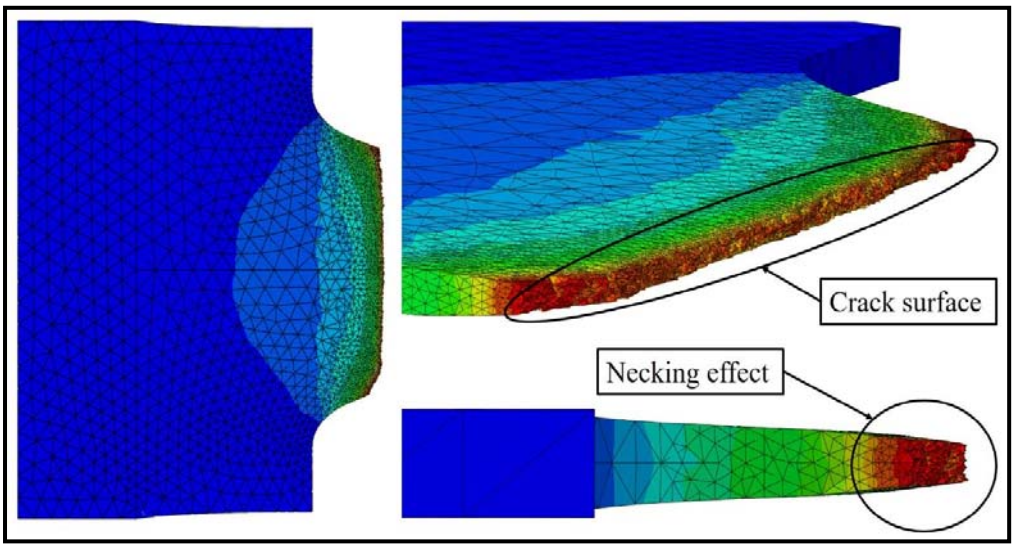
Simulation of continuous damage and fracture in metal-forming processes with 3D mesh adaptive methodology

Fangtao Yang

Par **Fangtao YANG**

Simulation of continuous damage and fracture in metal-forming processes with 3D mesh adaptive methodology

Thèse présentée
pour l'obtention du grade
de Docteur de l'UTC



Soutenue le 10 novembre 2017
Spécialité : Mécanique Numérique : Unité de recherche en Mécanique - Laboratoire Roberval (UMR-7337)
D2385

utc Recherche Roberval

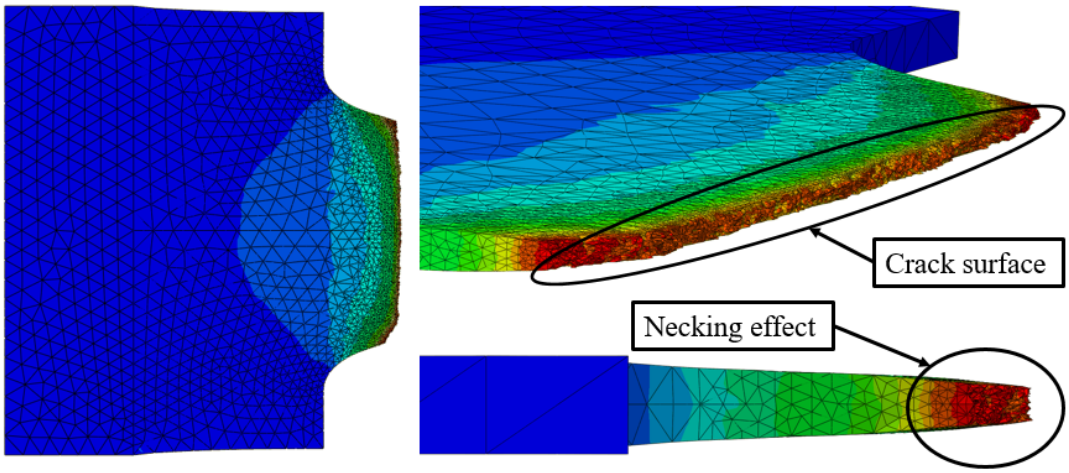
SORBONNE UNIVERSITES

par **Fangtao YANG**

Simulation of continuous damage and fracture in metal-forming processes with 3D mesh adaptive methodology

Thèse présentée pour l'obtention

du grade de Docteur de l'UTC



Soutenue le 10 novembre 2017

Spécialité: Mécanique Numérique

Simulation of continuous damage and fracture in metal-forming processes with 3D mesh adaptive methodology

Thèse soutenue le 10 novembre 2017 devant le jury composé de :

Prof	Jean-Michel Bergheau	École nationale d'ingénieurs de Saint-Étienne	Rapporteur
Prof	David Dureisseix	INSA de Lyon	Rapporteur
Dr	Frédéric Boitout	ESI Group Lyon	Examinateur
MCF HDR	Delphine Brancherie	Université de Technologie de Compiègne	Examinatrice
DR(Dir Rech)	Anne-Marie Habraken	Université de Liège	Examinatrice
Prof	Khemais Saanouni	Université de Technologie de Troyes	Examinateur
Prof	Carl Labergère	Université de Technologie de Troyes	Co-directeur de thèse
Prof	Alain Rassineux	Université de Technologie de Compiègne	Directeur de thèse

Acknowledge

Today is the day that the thesis ended successfully under the direction of Professor Alain RASSINEUX and Professor Carl LABERGERE. Remember these 3 years, I am so lucky that I have received help from many people who are at the base of this success.

First of all, I would like to thank all the members of the jury of the defense of my thesis who took the time to come to CompiÃĺgne from Lyon, Troyes and Li'ege. Special thanks to Dr. Jean-Michel Bergheau and Dr. David Dureisseix who are the reporters of my thesis and give a lot of valuable advice on the topics. I would like to express my gratitude to my supervisor, Dr. Alain RASSINEUX, whose expertise, understanding and patience have greatly enriched my professional experience. I appreciate his extensive knowledge and skills in meshing techniques (eg, Octree, mesh generation, bi-section subdivision, surface smoothing), and his help in writing reports (grant proposals, published articles and this thesis). Moreover, it helps me not only in my work, but also in my life. Whenever I have problems, for example, a toothache suddenly, it's always him who can find a solution for me. He drive me to Paris to see a dentist.

I would like to thank the other members of my laboratory, Dr. Piotr Breitkopf, and Dr. Delphine Brancherie for their help at all levels of the research project. Finally, I would like to thank Dr. Carl Labergere, as co-director, and Dr. Khemais Saanouni from Troyes University of Technology for all their efforts to help me build the elastoplastic model and its numerical application.

I would also like to thank my family for the support they have given me throughout my life and in particular, I must thank my wife, without whose help, love, and encouragement, I would not have completed this thesis.

In conclusion, I recognize that this research would not have been possible without the financial assistance of the China Scholarship Council (CSC), Compiegne University of Technology (UTEAM) and I express my gratitude to these agencies.

Résumé des travaux

Ces travaux s'inscrivent dans le cadre des recherches menées dans le cadre d'une collaboration entre le laboratoire Roberval de l'Université de Technologie de Compiègne et l'équipe dans le cadre du projet ANR-14-CE07-0035 LASMIS de l'Institut Charles Delaunay de l'Université de Technologie de Troyes. Nous présentons dans ces travaux une h-méthodologie adaptative tridimensionnelle des éléments finis afin de représenter l'initiation et la propagation des fissures dans des matériaux ductiles. Un modèle elasto-plastique couplé à l'endommagement isotrope proposé par l'équipe du LASMIS/UTT est utilisé. Les applications visées à terme concernent principalement la mise en forme des métaux. Dans ce contexte, une formulation Lagrangienne actualisée est employée et des remaillages fréquents s'avèrent essentiels afin d'une part d'éviter les fortes distorsions d'éléments dues aux grandes déformations plastiques et d'autre part de suivre les modifications de la topologie résultant de la création de fissures. La taille du nouveau maillage doit permettre à moindre coût représenter avec précision l'évolution des gradients des quantités physiques représentatives des phénomènes étudiées (plasticité, endommagement...). Nous proposons des indicateurs empiriques de taille d'éléments basés sur la déformation plastique ainsi que sur l'endommagement. Une courbe définie par morceau représente l'évolution de la taille d'élément suivant la sévérité de la plasticité et le cas échéant de l'endommagement. Les fissures sont représentées par une méthode de destruction d'éléments qui permet une description aisée de la géométrie de ces dernières et une gestion simplifiée de la fissuration sans nul besoin de critères additionnels. En revanche, pour permettre une description réaliste des fissures, ces dernières doivent être représentées par l'érosion des éléments de plus petite taille. Un solveur ABAQUS/Explicit® est utilisé avec des éléments tétraédriques quadratiques (C3D10M) évitant notamment les problèmes de verrouillage numérique survenant lors de l'analyse de structures en matériau compressible ou quasi-incompressible. Le contrôle de la plus petite taille du maillage est important dans un contexte explicite. De surcroît, pour les phénomènes adoucissant, la solution dépend de la taille de maille considérée alors comme un paramètre intrin-

sèque. Une étude nous a permis de constater que lorsque le maillage est suffisamment raffiné, les effets de la dépendance au maillage se réduisaient. Dans la littérature, les coûts de maillage ou de remaillage fréquents sont souvent considérés comme prohibitifs et de nombreux auteurs s'appuient sur cet argument pour introduire, avec succès certes, des méthodes alternatives qui limitent le coût des opérations de remaillage sans toutefois les éliminer (XFEM par exemple). Nos travaux montrent que le coût d'un remaillage local est négligeable par rapport au calcul. Compte tenu de la complexité de la géométrie et de la nécessité de raffiner le maillage, la seule alternative à ce jour est d'utiliser un mailleur en tétraèdres. La stratégie de remaillage local en tétraèdre s'appuie sur une méthode de bisection suivie si nécessaire d'une optimisation locale du maillage proposé par A. Rassineux en 2003. Le remaillage, même local, doit s'accompagner de procédures de transfert de champ des variables nodales et aux points d'intégration. Les variables nodales sont, comme le fait la plupart des auteurs, transférées en utilisant les fonctions de forme éléments finis. Le transfert de champ en 3D aux points de Gauss et les nombreux problèmes sous-jacents ont été relativement peu abordés dans la littérature. Les principales difficultés à résoudre pour assurer la « qualité » du transfert concernent la limitation de la diffusion numérique, le manque d'information près des frontières, le respect des conditions aux limites, de l'équilibre, les coûts de calcul, le filtrage du support d'information, des problèmes cruciaux en dimension 3 où le nombre de points de Gauss utilisés se chiffre en plusieurs centaines. Nous proposons une méthode dite « hybride » qui consiste, dans un premier temps, à extrapoler les données aux points de Gauss, aux noeuds par interpolation diffuse (ID) puis à utiliser les fonctions de forme éléments finis pour obtenir la valeur au point considéré. Un point clé du transfert par Moindres Carrée Mobiles (interpolation diffuse) réside dans la détermination des points d'information sur lesquels est construit l'approximation afin de limiter, d'une part, la diffusion numérique et, d'autre part, d'éviter de lisser et donc de réduire localement la valeur des champs. Afin de capter avec précision les forts gradients (distance et direction) dans les zones fortement localisées, nous proposons des techniques de sélection de points d'information permettant d'obtenir une distribution isotrope de points autour du point d'évaluation. La

contribution d'un point d'information à l'approximation est un compromis entre la distance au point d'évaluation et la direction radiale autour du point d'évaluation. Des poids locaux basés sur la distance et la direction évaluant ces contributions ont été développés. Au voisinage du point d'évaluation, on peut considérer que les points d'informations sont contenus dans une sphère et la méthode repose en particulier sur un découpage original de la surface en zones de même surface, dont la forme est la plus compacte possible minimisant l'écart entre zones adjacentes. Le transfert est validé par des exemples analytiques avec de très forts gradients et de fortes variations du maillage. L'approche globale est validée avec succès à l'aide de tests de traction sur des éprouvettes entaillées en acier laminé à froid (SPCC) pour lesquels on a fait varier le rayon d'entaille. Les résultats en termes d'initiation et de propagation des fissures, suivant la forme de l'éprouvette et des courbes force-déplacement, sont en très bon accord avec l'expérience. Les perspectives concernent principalement la simulation de divers procédés de mise en forme : découpage, pliage, forgeage.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The study of the initiation and propagation of cracks inside structures subjected to various monotonically increasing loading paths or mixed loading modes is of important concern in numerous industrial domains. A number of authors have proposed classifications of different techniques to perform crack simulation. In general, under monotonical loading path, cracks occur by brittle or by ductile mechanisms. For example, there are two distinct branches using 'smeared crack model' or the 'discrete crack model' to simulate brittle cracks. In this thesis, we focus on the simulation of ductile cracks. Authors study the crack in the structures by two basic theories, namely fracture mechanics and the damage mechanics. Usually, the development of the fracture mechanics is based on one or several small initial cracks. This kind of study mainly focus on the direction of crack propagation. However, the damage mechanics is based on the cumulative damage of the materials which is usually linked to the cumulative plastic strain. In this kind of study, initial cracks are not necessarily predefined and the cracks occur at the zones where the materials are totally damaged. The advantages of damage based crack are that the crack initiation zones can be predicted and several cracks can be treated in the same time without increasing the implementation difficulties.

The representation of the cracks has been widely studied in the literature. In terms of the representations, three main numerical methods are applied by authors, namely finite element methods with pre-refined mesh; displacement enriching ele-

ment or cohesive boundary without remeshing; and pure finite element methods with frequent remeshing.

In the first group, crack initiation and propagation can be represented in a pre-refined constant mesh without remeshing. In this group the discontinuity of the displacement across cracks surface are implemented either at interface of elements or within elements. LUIZ F. Martha [55] introduced a 2D direct boundary element method which works in a linear elastic domain. Full cubic polynomials are chosen to interpolate the discontinuity displacement function on both sides of the crack surface. G.T.Camacho and M.Ortizt [16] have proposed 2D cohesive-law fracture dynamic models which used to propagate multiple brittle cracks. Crack surfaces are created by splitting nodes according to a fracture criterion for conical cracks and a continuum damage criterion for radial cracks. Propagation direction of cracks are restricted along the element edges. Working with standard finite elements are the principle advantages of these methods, due to easy integration in the standard FEM code. However, cracks should propagate along the interface between elements so that mesh alignment have a significant influence on such method as explained by Martin G.A. Tijssens [87]. Furthermore, to guarantee the accuracy of direction of crack propagation, fracture zones should be predicted primarily and mesh is then pre-refined in vicinity of these zones which usually cost too much or sometimes even impossible. In order to eliminate the influence of mesh alignment and reduce calculation costs (relatively coarsened mesh), Dolbow [25] proposed a method which allows cracks to propagate in an arbitrarily direction including traversing elements. This crack representation is independent from mesh element. Discontinuity is integrated into nodes near crack surface by jump functions and nodes around crack tip by crack tip functions so that remeshing is not necessary. Some other authors [57], [14], [15] used X-FEM to introduce discontinuity in the fracture zones. Nodes around cracks are enriched and number of integration points in each element there are multiplied. Cracks are represented by a fictitious line across the enriched crack patch. An approach called Thick Level Set (TLS) [15] [4] is also proposed in which the damage variable is represented by a level set function. The damaged and undamaged zones are therefore separated

which allows the use of a non-local formulation and takes internal size effects into account. The cracks can be represented within the damaged zone.

In the second group, cracks are presented with a little modification of mesh. Due to the singularity at crack tips by fracture mechanism, it is attractive to deal with the zone surrounding crack tips separately from other zones. Otherwise, damage model should be performed. Nishioka et al. [60] and Atluri [2] introduced a moving mesh technique in order of avoid frequent remeshing. The elements around crack tip are replaced by singular-elements within which a large number of analytical eigen-functions corresponding to a steadily propagating crack are used as basic functions for displacements. However, it is much too complicated to transfer this method from 2D structured mesh to 3D unstructured mesh. Rashid [68] introduced an arbitrary local mesh replacement method, in which the finite element interpolant in the vicinity of the crack tip is replaced with one that is derived from a moving mesh patch. The moving mesh patch immerses in the background mesh according to several pre-defined patterns which limits the method to be used only to small deformations and planar problems as well as mono-crack propagation. Some other authors used damage model with little modification along crack path. In order to simulate cracks in quasi brittle materials, Kuutti and Kolari [43] have proposed a delete-and-fill 3D local remeshing procedure which in fact splits full damaged elements and triangularizes the sub-portion of them. The cohesive elements with zero thickness are used as interface of inter-elements and sub-portion elements. However, saw-tooth shape crack surface from split elements may cause shear failure. Dolbow and Moes [26] apply extended finite element method in modeling crack growth with frictional contact on the crack faces. The enriched crack path elements are triangularized with the new boundary, namely fictitious crack line which split the original crack path elements.

In the third group, pure finite element method with frequent changing of mesh topology is performed in most of the work. Adaptive mesh seems to be attractive, because it makes both initiation and propagation of cracks automatic without prediction of crack path and pre-refinement of mesh. The distortion of finite element is usually solved by remeshing technique. Bouchard et al. [7] [6] simulate 2D crack

propagation with both remeshing and nodal relaxation. The crack propagation direction is firstly computed, then nodes and edges are split along the calculated crack path, then the ring domain surrounding the crack tip is refined and remeshed. Several crack growth criteria are tested, such as maximum/minimum circumferential stress criterion, maximum strain energy release rate, the strain energy release rate and so on. Field transfer is indispensable after remeshing to form a new physical field based on the new mesh. Mediavilla [56] proposed a 2D remeshing-transfer operator for ductile fracture. The crack propagates along the predicted crack path which is based on maximum hoop stress and maximum tangential stress. A full remeshing process is carried out in the whole domain with the new boundary (crack path). Usually the distortion of elements occurs only along the crack path so that local remeshing can be used to largely reduce the cost of calculation compared to the full remeshing [41]. Khaoulani and Bouchard [40] proposed an anisotropic adaptive mesh strategy based on the proper combination of metrics driven by both damage and damage rate variables. The crack is represented by killing fully damaged elements. Labergere et al. [45] proposed an 2D h-adaptive remeshing procedure dedicated to the simulations of macroscopic ductile cracks initiation and propagation during metal forming process in which cracks are represented based on fully damaged elements deletion. The mesh located inside highly localized zones is adapted by both cumulative plastic strain and plastic strain rate. The loading sequence is also adapted by number of elements killed at each step.

Predicting the initiation and propagation of 3D cracks inside elastoplastically deforming structures that are subjected to various loading paths remains an open problem. In this work we shall focus on 3D FEM simulations of metallic ductile fracture under monotonically increasing loading paths, with an emphasis on adaptive aspects including size indicators, local remeshing, and field transfer, as well as on various related topics. For this reason we shall not discuss the large amount of published works relating to crack propagation in brittle and quasi-brittle structures.

In plastically deforming structures, large plastic or viscoplastic strains are localized within narrow bands, leading to the formation of macroscopic cracks. Since in

elastoplasticity there are no clear definitions of singularity at the crack tip, the finite element method is the most effective method for describing crack initiation and propagation. In 2D, many works have been proposed to address this problem [45]. Among these techniques, the deletion of fully damaged elements, together with an adaptive methodology, has been widely used with success. However, in a 3D FEM context, this is not a trivial task, mainly because of the remeshing difficulties. It is often remarked that unlike in 2D, the cost of 3D adaptive remeshing is prohibitive. Among the main objectives of the present work is to show that a framework can be proposed in which the computational cost of 3D adaptive remeshing may be negligible in relation to the total computation time.

Concerning the initiation and propagation of 3D cracks in elastoplasticity, several works have addressed issues relating to the deletion of fully damaged elements. Molinari et al. [58] proposed an adaptive methodology for predicting the localization of thermomechanical fields inside intensive shear bands. The authors used Rivara's [74] longest bisection technique together with mesh coarsening based on edge collapsing as a remeshing technique. They used an energy norm error indicator to refine the mesh. Mesh quality is not optimized during the simulation, which may affect the accuracy of the numerical solution. Also, the authors did not discuss field transfer and crack propagation in details. Pires et al. [65] proposed an explicit framework for describing fracture initiation based on damage growth. The authors used linear tetrahedron elements in which volumetric locking is enforced over patches of simplex elements. A full remeshing is performed at each loading step in which element size is driven by ZZ Error estimates [89] based on the damage rate together with the damage energy release rate. State variables at integration points are transferred via a moving least-squares technique based diffuse approximation method [12]. The authors did not address the problem of numerical diffusion [64] during fields transfer using a diffuse approximation method directly, and did not investigate the process of crack propagation. In an implicit context, some authors have proposed specific elements to avoid the problem of volumetric locking, especially for quasi-incompressible tetrahedral elements. Feulvarch et al. [30] have used P1/P1 element. Khaoulani

and Bouchard [40] investigated an anisotropic mesh adaptation strategy to describe ductile fracture and have used a bubble P1+/P1 tetrahedral element which indeed reduces the cost of the simulaton compared to a higher degree element. To regularize mesh dependency they proposed a simple nonlocal damage formulation based on continuum damage mechanics. In their remeshing technique, mesh is refined according to error estimates based on the damage and damage rate Hessian matrix. Fracture is represented through the deletion of fully damaged elements. However, the authors gave no details regarding the technical aspects of field transfer and associated 3D process of crack propagation. Following their work, Cao [17] modified the classical Lemaitre damage model and introduced the effects of the Lode angle on the damage evolution relationship. Cao used an element erosion technique in order to represent the crack accurately while avoiding mesh adaptation. Instead, a pre-refined mesh is created in the regions where cracks are expected to appear. Consequently, no error estimates or size indicators are needed, but the cost of the simulation is increased, since the number of elements required to represent the crack may be considerably overestimated. Javani [38], in an implicit context, performed a full remeshing in order to avoid element distortion and to represent the crack while improving the quality of the field transfer. The transfer is based on an extension of Mediavilla's work [56], meaning that the mechanical fields in the new mesh satisfy the equilibrium equations (static admissibility), the boundary conditions, and the yield function (plastic admissibility). As a consequence, the cost of a full remeshing and of the related transfer procedures is high. The author proposed a method controlling both the crack surface and the crack front to represent the propagation of the crack based on non-local damage. In order to perform a full tetrahedron remeshing, the surface of the domain must be rebuilt, including the crack surfaces and the boundaries of the domain. Although the author addressed meshing issues, the development of complex-shaped cracks and crack branching presents difficulties.

The 3D adaptive methodology that we propose in the present work is a 3D extension of the an 2D h-adaptive remeshing procedure proposed by Labergere et al. [45]. The methodology is applied for simulating initiation and propagation of macroscopic

ductile cracks in metal forming processes. Cracks are represented by deleting the fully damaged elements with the smallest mesh size. The main goals of the present approach are:

- Local robust remeshing at a negligible computational cost with controlled mesh gradation only when and where needed
- Hybrid field transfer (mesh and meshless) based on improved point selection, which limits the number of selected points, reduces numerical diffusion and preserved peak value of the field
- Size-controlled fully damaged element deletion, which retains the manifold topology of the domain with negligible volume loss
- Easy implementation within any explicit finite element code (quasi-independent from elastoplastic model)
- Smooth the crack surface (in order to coarsen the mesh or deal with contact problem in the future).

The overview of the h-adaptive methodology is given in chapter 2 and schemed to show each part in this methodology. The whole simulation of crack propagation is carried out by interating the adaptive loading sequences. The elastoplastic damage model is briefly introduced in chapter 3. The meshing and remeshing processes are presented from chapter 4 to 7 including element deletion, remeshing and field transfer. In each loading sequence, a computation loop is performed consisting of five sub-processes: (1) the equations related to the model are solved with the boundary and initial conditions at the beginning of the loading sequence; (2) fully damaged elements are deleted; (3) the size of the mesh is determined by size indicators; (4) the local remeshing technique is used to perform mesh adaptation; (5) all the physical fields are transferred from old mesh (before adaptation) to the new mesh (after adaptation). Since we find the field transfer process has a great influence on the adaptive methodology, we shall concentrate on field transfer process in section 7 and a detailed

state of art is made in the beginning of the section. Numerical simulation results are presented in chapter 8 and are compared with experimental results at last.

Chapter 2

Overview of the adaptive methodology

In the context of an updated Lagrangian-based formulation, frequent remeshing is required in order to avoid mesh entanglement following major topological changes, such as the creation of new surface boundaries resulting from newly created cracks and large plastic strains. As a consequence, the mesh must be adapted - where possible, at the smallest possible computational cost - to take account of topological changes in the domain and changes in physical fields. The two main drawbacks of remeshing that are most often cited are its prohibitive computational cost and its lack of robustness. Solving these issues would create new opportunities for improving the effectiveness of the updated Lagrangian FE formulation under finite transformations. Determining an appropriate mesh size in relation to either geometrical or physical criteria is a crucial issue in finite element analysis. In a context in which developing cracks are reproduced through the deletion of fully damaged elements based on local mechanical fields (plastic strain rate, damage rate, or dissipation) the choice of the smallest element size and the gradation of the mesh is crucial. Some of the above mentioned authors proposed error indicators based on total energy norm [58][8] or damage and damage rate [65][40]. Others chose an empirical mesh size distribution [45][17] to represent the evolution of the physical fields and to obtain a realistic crack path. The goal is always to assign the smallest size to the elements which reproduce

expected crack paths. Cao [17] separated the mesh domain into, first, potential crack regions in which a constant smallest mesh size is used and, second, other regions where a larger mesh size is used. Labergere et al. [45] used plasticity and damage to divide the mesh domain into sub-regions in which mesh size as well as mesh gradation is controlled.

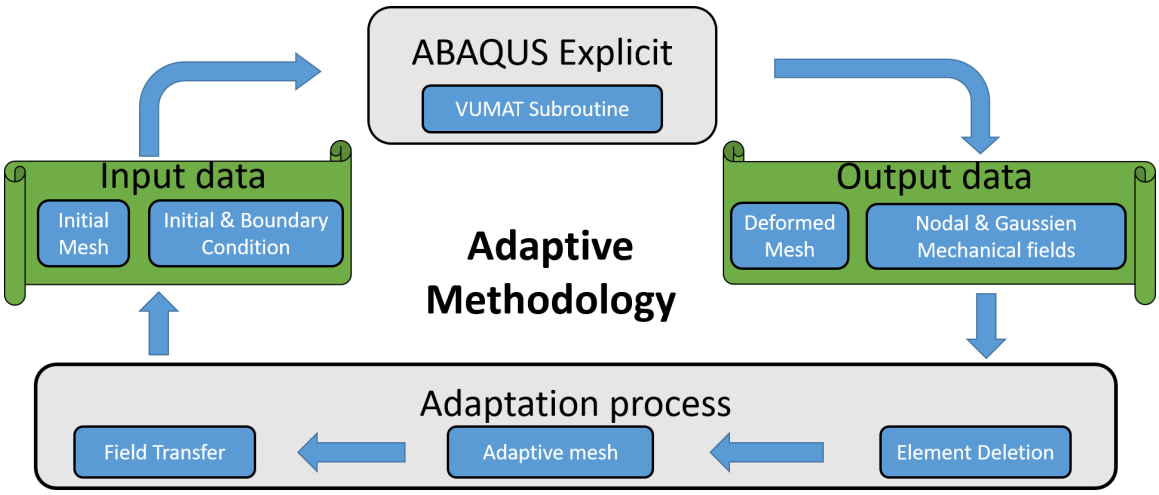


Figure 2-1: Overview scheme of the 3D adaptive methodology of ductile crack simulation

The whole ductile crack simulation (initiation and propagation) is then performed by iterating a calculation loop as illustrated in Fig. 2-1.

The elastoplastic constitutive equations are developed in VUMAT user defined subroutine which is called by ABAQUS/Explicit® with double precision. The input data for the ABAQUS/Explicit® includes 4 parts: the initial mesh data, the initial condition, the boundary conditions and the material parameters. In terms of iterating the calculation loop, there are two situations: the beginning of the simulation and in the beginning of one loading step (except from the first one). At the beginning of the simulation, the geometry of the specimens (parts) is discretized into a tetrahedron mesh. The geometry of the holders (tools) is discretized into a triangle mesh. If there is a contact between parts and tools, a geometrical adaptive remeshing process is performed. In this process, the meshes of the parts are refined according to the geometrical curvature of the tools. At the beginning of one loading sequence, the connectivity of the tool mesh is not changed and only the position of the tool

node is updated by boundary condition (displacement load). The mesh information of the parts are obtained from remeshing process which changes all the mesh information automatically in the last loading step. The mesh information includes: the coordinates of all the nodes, the connectivity of all the element, the node sets and the element sets of the user defined surfaces. In terms of initial conditions, at the beginning of the simulation, all the state variables are null. At the beginning of one loading sequence, the initial condition is obtained from field transfer process which is discussed in detailed in the chapter 7. Boundary conditions include 2 parts: the load and the constrains. The load can be displacements or forces exerted on either nodes or surfaces. The constrains can be clamped or contact surface pairs (a part contacts with a tool). During the simulation, the surfaces involved in the boundary condition can be remeshed. Therefore, the information in these sets should be updated after each loop in order to guarantee the correct boundary condition. For example if a specific surface is refined, all the nodes and the elements defining this surface should be included in the node and element sets defining this surface. In addition, the curvature of the surface should be kept even if the mesh is refined and the contact surface pairs should be carefully verified so that no penetration occurs especially in active or likely to be active areas. The material parameters are constant during all the simulation loading steps. ABAQUS/Explicit® solves the constitutive equations and gives the solutions including the values of state variables at both nodal and integration points.

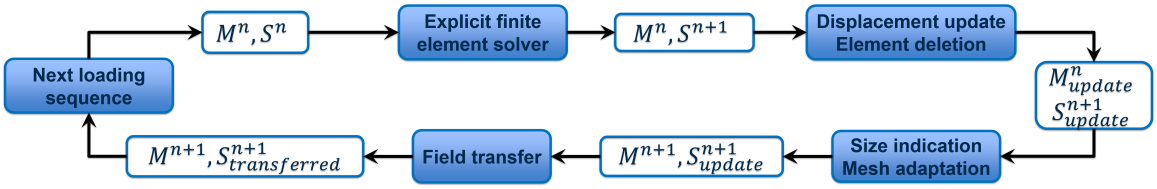


Figure 2-2: The data flow of both mesh and state variables in the calculation loop

In practice, after obtaining all the output data from ABAQUS/Explicit®, the meshing and remeshing process begins. This process is carried out in C language with different independent executable files: output data interpretation works as an interface which reads both the mesh data and the state variable fields data from ASCII data given by ABAQUS/Explicit® in order to be used directly by executable

files geometry update which adds the displacement of the parts and the tools on their nodes; element deletion which removes the totally damaged elements from the geometry of the parts and also removes the field values of state variables corresponding to these elements; a size indication procedure builds a referenced size map of the the mesh of the parts at each integration points of all the elements; the remeshing procedure refines the mesh in the zones where it is needed and active according to the size map (if the simulation has contact constrains: smoothing the crack surface and remeshing the zones around the smoothing surface; coarsening the mesh in the zones where there are needed); the field transfer procedure projects the values of the state variables from the mesh after element deletion to the mesh after remeshing and input data preparation writes all the binary data from the meshing and remeshing process into ASCII file in order to be read by ABAQUS/Explicit® as input data. The overview of the data flow in the calculation loop is schemed in the Fig. 2-2 in which, M^n is the mesh at the time t_n, S^n is the mechanical solution at the time t_n, M^{n+1} is the mesh at the time t_{n+1}, S^{n+1} is the mechanical solution at the time t_{n+1}.

Chapter 3

Elastoplastic constitutive equations

The model used is detailed in appendice A and only a brief description is given here. This model is mostly the work carried out by Khemais Saanouni and his team at UTT (Université de Technologie de Troyes). Our approach has been validated successfully with this model. Our approach has also been carried out with an elastoplastic model from ABAQUS and the shortcomings of this model are presented. In particular, we show that the cracks obtained by this second model are unrealistic.

The fully coupled elastoplastic behavior is modeled in the framework of the thermodynamics of irreversible processes with state variables [51][80][44], assuming the small elastic strain and large plastic strain hypothesis for metallic materials. Three couples of internal state variables are used for this isothermal case: $(\underline{\varepsilon}_e, \underline{\sigma})$ for plastic flow, (r, R) for isotropic hardening and (d, Y) for isotropic ductile damage. When the current configuration exhibits ductile damage, the concept of the effective state variables based on the hypothesis of total energy equivalence [80] is used. The state relations between the strain variables and the dual stress variables are derived from the state potential, classically taken as the Helmholtz free energy $\Psi(\underline{\varepsilon}_e, r, d)$ additively decomposed and written in the present isotropic case as:

$$\rho\Psi(\underline{\varepsilon}_e, r, d) = \frac{1}{2}(1-d)\underline{\varepsilon}_e : \underline{\underline{\Lambda}} : \underline{\varepsilon}_e + \frac{1}{n+1}(1-d^{\omega})^{\frac{n+1}{2}} Q r^{n+1} \tag{3.1}$$

Where ρ is the material density, Q and n are respectively the modulus and non

linear parameter associated to isotropic hardening. $\underline{\underline{\Lambda}} = \frac{\nu E}{(1+\nu)(1-2\nu)}\underline{1}\otimes\underline{1} + 2\frac{E}{2(1+\nu)}\underline{\underline{1}}$ is the fourth rank order symmetric elastic stiffness tensor for the isotropic material where E is the Young modulus and ν is the Poissons' coefficient. The parameter ω is used to adjust the influence of the isotropic damage effect on the isotropic hardening. By introducing the above state potential into the Clausius-Duhem basic inequality the state relations are obtained. Application of the generalized normality rule with a specific pseudo dissipation potentials leads to the definition of the required complementary evolution of the state variables. All the constitutive equations are summarized in Table 3.1.

Table 3.1: State relations and evolution equations

	state relation	Evolution equation
Cauchy stress	$\underline{\sigma} = \rho\frac{\partial\Psi}{\partial\underline{\varepsilon}_e} = (1-d)\underline{\underline{\Lambda}} : \underline{\varepsilon}_e$	$\underline{D}_p = \dot{\delta}\underline{n}$
Isotropic hardening	$R = \rho\frac{\partial\Psi}{\partial r} = (1-d^\omega)^{\frac{n+1}{2}}Qr^n$	$\dot{r} = \dot{\delta}(\frac{1}{\sqrt{1-d^\omega}} - br)$
Damage	$Y = Y_e^+ + Y_r$ $Y_e = \frac{(1+\nu)[(\langle\sigma_I\rangle_+^2+\langle\sigma_{II}\rangle_+^2+\langle\sigma_{III}\rangle_+^2)]}{2(1-d)E} - \frac{9\nu[\langle\sigma_H\rangle_+^2]}{2(1-d)E}$ $Y_r = \frac{1}{2Q^{\frac{1}{n}}}\omega d^{(\omega-1)}(1-d^\omega)^{-\frac{3n+1}{2n}}R^{\frac{n+1}{n}}$	$\dot{d} = \frac{\dot{\delta}}{(1-d)^\beta}[\frac{Y}{S}]^s$

In the equations of Table 3.1, $\underline{D}_p$ is the finite plastic strain rate, $\sigma_H = \frac{1}{3}\sigma_{ii}$ is the hydrostatic stress and σ_I, σ_{II} and σ_{III} are the three principal stresses with the notation $\langle x\rangle_+ = \frac{(x+|x|)}{2}$. A classical von Mises yield function is used to determine the plastic domain:

$$f = \frac{J_2(\underline{\sigma})}{\sqrt{1-d}} - \frac{R}{\sqrt{1-d^\omega}} - \sigma_y = 0 \tag{3.2}$$

where σ_y is the yield stress in uniaxial tension. The normal to the yield surface is $\underline{n} = \frac{3}{2\sqrt{1-d}}\frac{\underline{\sigma}^{dev}}{J_2(\underline{\sigma})}$, where $\underline{\sigma}^{dev}$ is the deviatoric part of the Cauchy stress and $J_2 = \sqrt{\frac{3}{2}\underline{\sigma}^{dev} : \underline{\sigma}^{dev}}$ is its second invariant. The parameter b characterizes the non linearity of the isotropic hardening, while S, s and β characterize the ductile damage evolution. The scalar $\dot{\delta}$ is the plastic multiplier solution of the consistency condition given by $\dot{f} = 0$ when $f = 0$.

It is worth mentioning that the fully coupled constitutive equations derived above, are valid under the small strain hypothesis. To be used in metal forming by large plastic deformation, they should be extended to the finite strain hypothesis in such a manner that the objectivity requirement is fulfilled. This is made using an Eulerian description for which the total small strain rate tensor $\underline{\dot{\varepsilon}}$ transforms to its Eulerian counterpart $\underline{D}$. If the elastic part of the total strain tensor is supposed very small compared to the plastic one, the total strain rate can be approximated by the following additive decomposition:

$$\underline{D} = \underline{\dot{\varepsilon}}_e^J + \underline{D}_p \tag{3.3}$$

in which $\underline{\dot{\varepsilon}}_e^J$ stands for the Jaumann derivative of the elastic small strain tensor, while $\underline{D}_p$ is the finite plastic strain tensor defined as the symmetric part of the velocity gradient tensor. The objectivity requirement is ensured by using the so called rotational frame formulation (RFF). It leads to express the constitutive equations in a rotated deformed and damaged (Eulerian) configuration obtained from the current one by the orthogonal rotation tensor $\underline{Q}$, solution of the following kinematic equation:

$$\underline{\dot{Q}}^T \cdot \underline{Q} = \underline{W_Q} \quad \text{with} \quad \underline{Q}\,(t=0) = \underline{1} \tag{3.4}$$

where W_Q is the rotation rate tensor which can be chosen in order to obtain some classical rotational objective derivatives as Jaumann, or Green-Naghdi derivation. This model has been implemented into ABAQUS/EXPLICIT® FE codes using the VUMAT subroutine in order to simulate any cold or hot sheet or bulk metal forming processes using Dynamic Explicit (DE) resolution procedure.

The computation of the internal force vectors requires the knowledge of the stress tensor at the end of the time step. This passes through the numerical integration of the overall set of constitutive equations developed above written in the rotated deformed and damaged configuration. The overall loading path is discretized into several non-constant loading steps defined by $t_n, t_{n+1} = t_n + \Delta t$ and all the variables are known at t_n i.e.: $\underline{\sigma}_n$, $\underline{D}_{p_n}$, $\underline{Q}_n$, R_n and d_n and values at t_{n+1} should be computed.

Let's mention that the rotation tensor $\underline{\underline{Q}}_{n+1}$ at t_{n+1} is independently determined before performing the integration of the constitutive equation in order to rotate all the tensorial variables as discussed above. Now, if the fully implicit Euler integration scheme is used for all the differential equations excepting the hardening equation for which an asymptotic scheme is applied, the equations to be solved over the current load increment can be written under the following incremental form:

$$\underline{\underline{D}}_{p_{n+1}} - \underline{\underline{D}}_{p_n} - \Delta\delta \frac{\underline{\underline{N}}_{n+1}}{\sqrt{1-d_{n+1}}} = 0 \quad \text{with} \quad \underline{\underline{N}}_{n+1} = \frac{3}{2}\frac{\underline{\underline{Z}}_{n+1}}{\|\underline{\underline{Z}}_{n+1}\|}, \quad \underline{\underline{Z}}_{n+1} = \underline{\underline{\sigma}}^{dev}_{n+1} \tag{3.5}$$

$$r_{n+1} - r_n e^{-b\Delta\delta} - \frac{\left(1-e^{-b\Delta\delta}\right)}{b\sqrt{1-(d_{n+1})^{\omega}}} = 0 \tag{3.6}$$

$$d_{n+1} - d_n - \frac{\Delta\delta}{(1-d_{n+1})^{\beta}}\left[\frac{\langle Y_{n+1} - Y_0\rangle}{S}\right]^s = 0 \tag{3.7}$$

These equations should be solved thanks to the well-known elastic prediction and iterative "plastic" correction procedure using the radial return algorithm applied to a reduced number of equations, in such a manner that the following plastic flow conditions are fulfilled [80]

$$f_{n+1} = \frac{\|\underline{\underline{Z}}_{n+1}\|}{\sqrt{1-d_{n+1}}} - \frac{R_{n+1}}{\sqrt{1-(d_{n+1})^{\omega}}} - \sigma_y = 0 \tag{3.8}$$

Chapter 4

Fully damaged element deletion methodology

4.1 Element deletion criterion

In our methodology, the damage variable d takes values ranging from 0.0 (no damage) to 1.0 (total damage). In practice, we consider that an element is fully damaged if the damage variable exceeds a critical damage threshold $d_c = 0.99$ at all of the element's integration points. The value of d_c is tuned to obtain the clean path of the cracks meanwhile considering a minimum number of elements as fully damaged. To our experience, with our model, when d_c is bigger than 0.99 but less than 1.0, the parameter sensitivity is not that obvious. We remove this fully damaged element from the mesh if and only if its size does not exceed the smallest mesh size h^d_{min}, in order to limit volume loss and avoid severely distorted elements which can prevent the calculation from going on. If the size of some totally damaged elements is greater than h^d_{min}, the loading sequence must be reduced. The process is iterated until the size of all the elements to be removed is less than the given threshold h^d_{min}.

4.2 Numerical disturbance control

In the simulation analysis, if only the critical damage value criterion is used to perform element deletion, it may cause topological problems around crack surface as shown in Fig. 4-1. The green elements have damage value $d < d_c$ and should be kept while red elements have damage value $d > d_c$ should be deleted. The blue line indicates the crack surface. Because of numerical disturbances, some deleted elements result in voids in the material and some kept elements form independent volumes. These numerical disturbances come from complicated topology of unstructured tetrahedron mesh. These unrealistic configurations are also caused as a result of the field transfer process based on shape function and diffuse approximation discussed in chapter 7. These disturbances are removed with an automatic crack surface cleaning process.

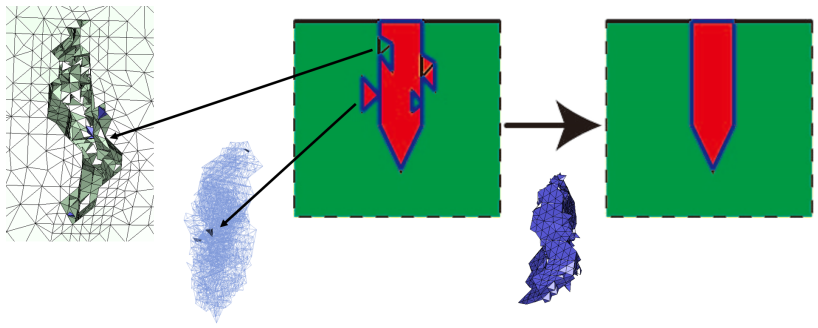


Figure 4-1: The numerical disturbance control process

Before crack surface cleaning process is discussed, we describe two sub-processes: Bulk Identification (Fig. 4-2) and Layer Separation (Fig. 4-3). Bulk Identification process works on a set of elements selecting bulks of the elements from the set. A bulk of elements is a set of elements in which every element has at least one common face with another element and the total number of elements in it must more than a threshold number. For instance, if we use Bulk Identification on the configuration in Fig. 4-2, the set of elements composed of black cube is selected. Because the elements in red, yellow and green have no common face with another element and the total number of elements composed of blue cube is smaller than a given threshold number. Layer Separation process asks for a set of nodes and the number of layers as input

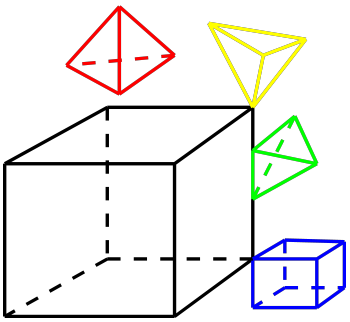


Figure 4-2: Illustration of bulk identification

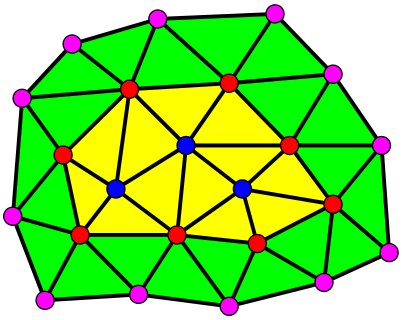


Figure 4-3: Illustration of layer separation

data. It gives a set of elements which derives from the input set of nodes depending on the number of layers. For example, in Fig. 4-3, number of layers equals to 2. The first layer of nodes are blue ones which used as input for Layer Separation. The first layer of elements are the yellow ones which attached to any one of the blue nodes. The second layer of nodes is composed of the red nodes from the the yellow elements other than blue nodes. The same process is iterated and red nodes are used to select the second layer of elements which attached to red nodes except from yellow elements. Finally, all the elements in yellow and green are included in the output element set of Layer Separation. Based on the same process, set of elements can be used as input data.

Bulk Identification process is defined as $E_{bulk}^{sum} = \bigcup_{i=1}^{k} E_{bulk}^{i} = \mathrm{BI}(E, n_{bulk})$ with E is an arbitrary set of elements, n_{bulk} is the threshold number, E_{bulk}^{i} is one of the k bulks of elements and E_{bulk}^{sum} is also a sum set of element of all k subsets E_{bulk}^{i}. If n is omitted, then a default value is given by $n = \mathrm{card}(A)/10$ in which A is the total geometry and card() is the number of elements in the geometry. This default value usually works well because the bulk part is always much bigger than the rest parts. Layer Separation process is defined as $E_{layer} = \mathrm{LS}(N, n_{layer})$ with N is an arbitrary set of nodes, n_{layer} is the number of layers and E_{layer} is the set of elements of n_{layer} layers deriving from N. Suppose that E_c is a set of elements which have a damage value $d > d_c$, E_k is a set of elements which should be killed and E_{all} is an element set including all the elements in the model, then crack surface cleaning process is carried out through Eq. 4.1.

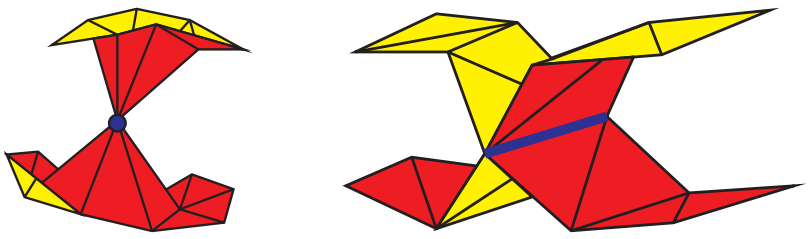
$$\begin{cases} \mathrm{LS}^3 = \mathrm{LS}(\mathrm{BI}(E_c), 3) \\ E_k = (E_{all} - \mathrm{BI}(\mathrm{LS}^3 - \mathrm{BI}(E_c))) \cap \mathrm{LS}^3 \end{cases} \tag{4.1}$$

in which $n_{layer} = 3$ is a compromise between calculation cost and cleaning effect. If n_{layer} is big enough, then $\lim_{n_{layer}\to\infty} \mathrm{LS}^n = E_{all}$ and Eq. 4.1 becomes to Eq. 4.2. This Eq. 4.2 is more direct than Eq. 4.1 and it drastically cleans the crack surface, however the drawback is that calculation costs increase dramatically with increasing of card(E_{all}). In the algorithm, the Bulk Identification process is based on Layer separation process, the Layer separation process is based on the inverse connectivity and anti-duplication of attached element. This anti-duplication is aimed to remove the same elements in a layer which are collected by inverse connectivity. In practice, a maximum of 3 layers is used which greatly reduces the calculation costs.

$$E_k = (E_{all} - \mathrm{BI}(E_{all} - \mathrm{BI}(E_c))) \cap E_{all} = \mathrm{BI}(E_{all} - \mathrm{BI}(E_c)) \tag{4.2}$$

4.3 Non-Eulerian configuration removal

Element killing changes the topology of the mesh, sometimes may make the mesh not comply with Euler polyhedron formula. Two typical configuration are illustrated in Fig. 4-4(a) “single-node” and in Fig. 4-4(b) “single-edge”. The removal process begins with “single-edge” identification and edge split followed by “single-node” identification and node split. Both “single-node” and “single-edge” are included in the surface of the 3D mesh.



(a) Single node (b) Single edge

Figure 4-4: Non-manifold configurations

"Single-edge" identification is performed based on a definition that a surface edge to which more than 2 surface faces attached is "single-edge". In order to remove this Non-Eulerian configuration, "single-edges" are split as following:

1) We build an element set E_{atch} including all the elements having this "single-edge" as a common edge.

2) Bulk Identification process is used $\bigcup_{i=1}^{k} E_{bulk}^{i} = \mathrm{BI}(E_{atch}, 1)$ to get k sub bulks of elements E_{bulk}^{i}

3) The "Single-edge" is split $k-1$ times and $k-1$ new edges are created so that k edges overlap in the space but belongs to k bulks of elements independently.

"Single-edge" split does not split nodes, however it transforms some Eulerian nodes to be "single-nodes". These new "single-nodes" can be coped with the same as old "single-nodes" which exist before "single-edge" split. Similar to "single-edge" split, "single-node" split works as following:

1) For each surface node, we built an element set E_{atch} including all the elements having this node as a common node

2) Bulk Identification process is used $\bigcup_{i=1}^{k} E_{bulk}^{i} = \mathrm{BI}(E_{atch}, 1)$ to get k sub bulks of elements E_{bulk}^{i}

3) If $k > 1$, the "Single-node" is split $k-1$ times and $k-1$ new nodes are created so that k nodes overlap in the space but belongs to k bulks of elements independently. If $k = 1$, we do nothing because this is an Eulerian node.

4.4 Physical field consistence

Element deletion process (numerical disturbance control and Non-Eulerian configuration removal) may change the topology of the 3D mesh. All the physical fields (integration or nodal) should be projected from old mesh to new mesh. In terms of integration fields, all values of killed elements are removed directly. However, some

totally damaged elements ($d > d_c$) are kept by element deletion process. For these elements, the integration values are transferred using diffuse approximation [13] and Shape function of element. The procedure is detailed in chapter 7. Herein, the information points are only the integration points of the elements which are not killed. In terms of nodal fields, we only copy the values from old "single-node" to the two nodes created by splitting it. In addition, due to element deletion, some nodes which attached only to the killed elements are also killed. The nodal field value of these nodes are removed directly. Above all, the physical fields are consistent between old topology and new topology.

The volume loss is often pointed out in an element deletion procedure but rarely quantified. Since the size of the deleted elements are limited to be no more than minimal size, only a minimal volume of the geometry is removed. We experienced that volume loss is far less than 0.1% and can be therefore neglected.

Chapter 5

Mesh size determination

In our methodology, mesh size is driven by the cumulative plastic strain and damage. The behavior of the ductile material can be separated into different phases: purely elastic zones without plasticity, homogeneous plastic zones with low damage values, localized plastic zones with moderate damage values, and highly localized plastic zones with severe damage values. We propose empirical mesh size indicators based on the cumulative plastic strain $p = \int_0^t \dot{p} dt$ with $\dot{p} = \sqrt{\frac{2}{3}\underline{D}_p : \underline{D}_p}$ and the damage value d. In the purely elastic zones, the cumulative plastic strain p is zero and the mesh size is set to the maximum mesh size h_{max}. Within the homogeneous plastic zone where $p \leq p^*$, the mesh size decreases linearly with respect to p from h_{max} to h^p_{max}. Here p^* is a threshold value satisfying a given criterion which maximizes the value of the second invariant of the Cauchy stress, i.e. $J_2(\underline{\sigma}(p^*, \ldots)) = max(J_2(\underline{\sigma}(p, \ldots)))$. Finally, within the highly localized plastic zone where $p > p^*$, high ductile damage values may appear, and the mesh gradation needs to be tuned to take account of how the severe damage develops. The change in mesh size with respect to the cumulative plastic strain p is therefore described by an exponential function decreasing from h^p_{max} to h^p_{min}. When the damage variable d exceeds a given threshold d_{min}, mesh size is driven by the damage variable d and reduces as an exponential function from h^p_{min} to h^d_{min}. Finally, when the damage variable d exceeds a threshold d_{max}, the smallest damage-related mesh size h^d_{min} is prescribed. All the related functions are listed in Table 6.1 and change in mesh size with respect to physical quantities is plotted in

Fig. 6-1.

Table 5.1: Equations of empirical size indicators

Behavior phases	Evolution of plasticity p and damage d	Size indicators
Purely elastic zone	$p = 0$	$h^p = h_{max}$
Homogeneous plastic zone	$p \leq p^*$	$h^p = \frac{(h^p_{max} - h_{max})}{p^*} p + h_{max}$
Localized plastic zone	$p > p^*$	$h^p = (h^p_{min} - h^P_{max})(1 - e^{-\kappa_1 (p - p^*)}) + h^P_{max}$
Moderate damage zone	$d_{min} < d \leq d_{max}$	$h^d = (h^d_{min} - h_p)(1 - e^{-\kappa_2 (d - d_{min})}) + h^d_{max}$
Severe damage zone	$d > d_{max}$	$h^d = h^d_{min}$

In Table 6.1, κ_1 and κ_2 are two user-tuned parameters which control the gradient of the mesh size. The choice of the smallest mesh size h^d_{min} is important, and when making this choice a number of factors must be considered:

- The accurate geometrical representation of cracks
- The computational cost of the simulation, which depends on the number of elements and on stability requirements for the loading sequences in an explicit framework
- Mesh dependency, given that mechanical behavior is strongly linked to local ductile damage that leads to softening, can be controlled. The idea is to choose the smallest mesh size as an intrinsic parameter when identifying the material parameters.

In the metal forming process, the mesh distortion and the entanglement are often caused by large geometry deformation. Therefore, the level of the deformation can be used to indicate the level of mesh distortion and entanglement. The cumulative plastic strain is a variable which describes the geometry deformation in FEM. Therefore, cumulative plastic strain p is chosen as one of the size indicators. As a result, when

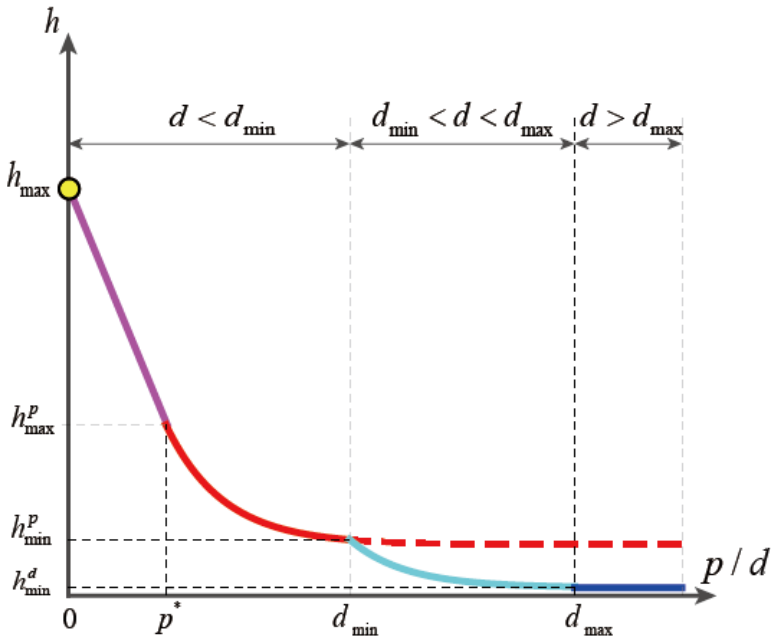


Figure 5-1: Evolution of the empirical size indicators

p increases to a certain level in a zone, the elements in the zones are refined in order to solve the distortion and the entanglement. As described above, we choose a quadratic tetrahedron element which has four nodes at four vertices and six nodes at the middle of the six edges to discretize the geometry. If the element is deformed, the edges of the element will be bended and kinked at the middle points as illustrated in Fig. 6-2. The mesh refinement (using bisection technique as will be described in the following chapter 6) is performed, the kinked edges are divided into two segments for the refined elements. We can see also in the last sub-figure in Fig. 6-2, none of the edges of the all elements is kinked and the elements are no longer distorted. When the mesh is repaired by refinement, the topology changes so that field transfer process is introduced. As discussed in chapter 7, the numerical errors in the field transfer process cannot be avoided, but should be reduced and controlled. Therefore, the mesh size should also be suitable to limit the numerical error in the field transfer in a minimum level. For this reason, the damage variable is also considered as a size indicator. Because in the large geometrical deformed zones, the damage variable is also highly located. The mesh should be refined enough so that transferring this

high gradient field with a high precision. In addition, as will be discussed in the chapter 7, in the field transfer process (transfer the value of state variables from the old mesh (deformed) to the new mesh (refined)), the size of the old mesh should be small enough before the gradient of field localizes. Therefore, for both cumulative plastic strain and damage variable, we propose a exponential function to describe the size map so that the mesh is refined earlier than the linear function during the metal forming process.

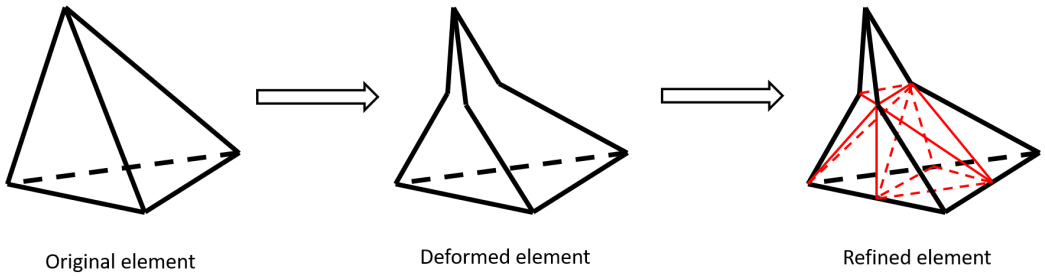


Figure 5-2: Example of largely deformed element and its refinement

Chapter 6

Adaptive local remeshing

6.1 Mesh size determination

In our methodology, mesh size is driven by the cumulative plastic strain and damage. The behavior of the ductile material can be separated into different phases: purely elastic zones without plasticity, homogeneous plastic zones with low damage values, localized plastic zones with moderate damage values, and highly localized plastic zones with severe damage values. We propose empirical mesh size indicators based on the cumulative plastic strain $p = \int_0^t \dot{p} dt$ with $\dot{p} = \sqrt{\frac{2}{3}\underline{D}_p : \underline{D}_p}$ and the damage value d. In the purely elastic zones, the cumulative plastic strain p is zero and the mesh size is set to the maximum mesh size h_{max}. Within the homogeneous plastic zone where $p \leq p^*$, the mesh size decreases linearly with respect to p from h_{max} to h^p_{max}. Here p^* is a threshold value satisfying a given criterion which maximizes the value of the second invariant of the Cauchy stress, i.e. $J_2(\underline{\sigma}(p^*, \ldots)) = max(J_2(\underline{\sigma}(p, \ldots)))$. Finally, within the highly localized plastic zone where $p > p^*$, high ductile damage values may appear, and the mesh gradation needs to be tuned to take account of how the severe damage develops. The change in mesh size with respect to the cumulative plastic strain p is therefore described by an exponential function decreasing from h^p_{max} to h^p_{min}. When the damage variable d exceeds a given threshold d_{min}, mesh size is driven by the damage variable d and reduces as an exponential function from h^p_{min} to h^d_{min}. Finally, when the damage variable d exceeds a threshold d_{max}, the smallest

damage-related mesh size h^d_{min} is prescribed. All the related functions are listed in Table 6.1 and change in mesh size with respect to physical quantities is plotted in Fig. 6-1.

Table 6.1: Equations of empirical size indicators

Behavior phases	Evolution of plasticity p and damage d	Size indicators
Purely elastic zone	$p = 0$	$h^p = h_{max}$
Homogeneous plastic zone	$p \leq p^*$	$h^p = \frac{(h^p_{max} - h_{max})}{p^*} p + h_{max}$
Localized plastic zone	$p > p^*$	$h^p = (h^p_{min} - h^P_{max})(1 - e^{-\kappa_1 (p - p^*)}) + h^P_{max}$
Moderate damage zone	$d_{min} < d \leq d_{max}$	$h^d = (h^d_{min} - h_p)(1 - e^{-\kappa_2 (d - d_{min})}) + h^d_{max}$
Severe damage zone	$d > d_{max}$	$h^d = h^d_{min}$

In Table 6.1, κ_1 and κ_2 are two user-tuned parameters which control the gradient of the mesh size. The choice of the smallest mesh size h^d_{min} is important, and when making this choice a number of factors must be considered:

- The accurate geometrical representation of cracks
- The computational cost of the simulation, which depends on the number of elements and on stability requirements for the loading sequences in an explicit framework
- Mesh dependency, given that mechanical behavior is strongly linked to local ductile damage that leads to softening, can be controlled. The idea is to choose the smallest mesh size as an intrinsic parameter when identifying the material parameters.

In the metal forming process, the mesh distortion and the entanglement are often caused by large geometry deformation. Therefore, the level of the deformation can be used to indicate the level of mesh distortion and entanglement. The accumulative

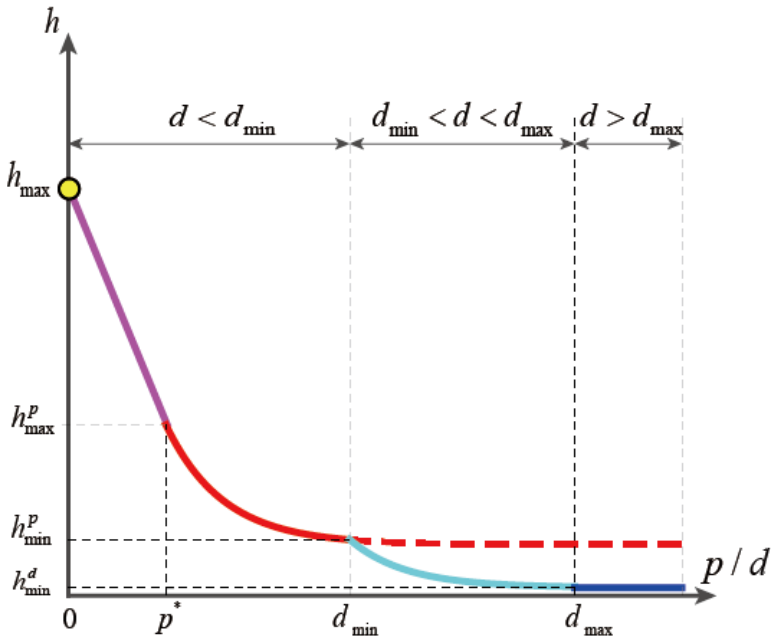


Figure 6-1: Evolution of the empirical size indicators

plastic strain is a variable which describes the geometry deformation in FEM. Therefore, accumulative plastic strain p is chosen as one of the size indicators. As a result, when p increases to a certain level in a zone, the elements in the zones are refined in order to solve the distortion and the entanglement. As described above, we choose a quadratic tetrahedron element which has four nodes at four vertices and six nodes at the middle of the six edges to discretize the geometry. If the element is deformed, the edges of the element will be bended and kinked at the middle points as illustrated in Fig. 6-2. The mesh refinement (using bisection technique as will be described in the following chapter 6) is performed, the kinked edges are divided into two segments for the refined elements. We can see also in the last sub-figure in Fig. 6-2, none of the edges of the all elements is kinked and the elements are no longer distorted. When the mesh is repaired by refinement, the topology changes so that field transfer process is introduced. As in chapter 7, the numerical errors in the field transfer process cannot be avoided, but should be reduced and controlled. Therefore, the mesh size should also be suitable to limit the numerical error in the field transfer in a minimum level. For this reason, the damage variable is also considered as a size indicator. Because

in the large geometrical deformed zones, the damage variable is also highly located. The mesh should be refined enough so that transferring this high gradient field with a high precision. In addition, as will be discussed in the chapter 7, in the field transfer process (transfer the value of state variables from the old mesh (deformed) to the new mesh (refined)), the size of the old mesh should be small enough before the gradient of field localizes. Therefore, for both cumulative plastic strain and damage variable, we propose a exponential function to describe the size map so that the mesh is refined earlier than the linear function during the metal forming process.

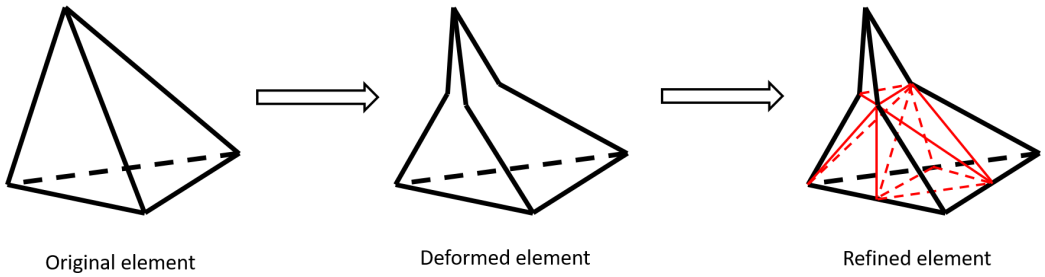


Figure 6-2: Example of largely deformed element and its refinement

6.2 Adaptive local mesh refinement with bisection technique

Ideally, the remeshing technique should adopt the prescribed mesh size and mesh gradation given by the size indicators detailed in chapter 5 above. In practice, in order to capture the gradient of the fields during the transfer process with accuracy, the mesh size must be smaller than that given by the size indicators, especially along the crack path. That means the size indicators give the maximum size of element on the mesh. Here we propose a local remeshing method based on a bisection technique [71]. The main goal of the proposed technique is to avoid global remeshing and therefore to modify the current mesh as little as possible. This local remeshing requirement is important for two reasons. First, the usual argument against 3D remeshing is its computational cost, often deemed to be prohibitive, but a local remeshing has a considerably lower cost than a global remeshing. Second, a local remeshing restricts

topology changes to the active regions, thus avoiding spurious and unnecessary field transfer (reducing the numerical diffusion). The numerical diffusion in a 3D context is a crucial issue during the field transfer process which is more obvious than in 2D context. A local remeshing can significantly limit the numerical diffusion in small zones and also reduces the cost of field transfer procedures themselves.

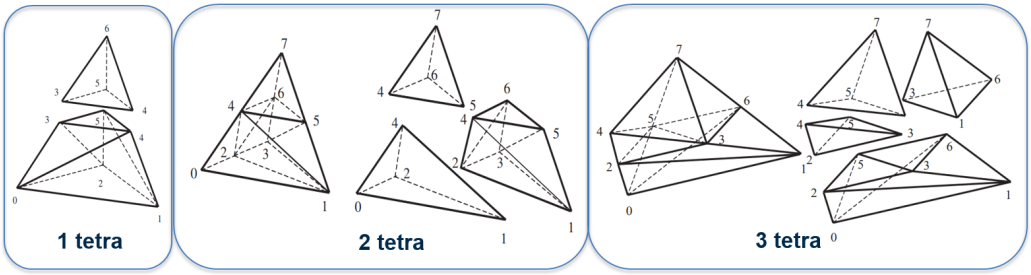


Figure 6-3: Three basic types of bisection refinement configurations[73]

The bisection technique is detailed in [71] and only a brief reminder of the main steps is given here. The whole process is based on edge subdivision. Element size distribution from mesh size indicators is provided by a background octree. If the length of a tetrahedron edge is smaller than the minimum information size obtained from the octree at each vertex and at the middle of the edge, the edge must be split. Once all the edges to be split have been identified, the process consists in remeshing one by one the tetrahedra in which at least one edge must be split. As all existing subdivision patterns of tetrahedra have been pre-calculated as illustrated in Fig. 6-3, this first remeshing step is very fast. However, experience shows that a further tetrahedron shape optimization is needed after all subdivisions. This optimization step consists in local remeshing around a node, an edge or a face of a tetrahedron in order to improve the mesh quality. To ensure the computational efficiency of the remeshing, this treatment is not performed on all elements, but limited to the elements that do not satisfy a given quality criterion [70]. During these local modifications, new nodes may be added, but the location of the nodes remains unchanged.

Since the process is based on bi-section subdivision, there is a size ratio of 2 between adjacent elements around the split elements. In our work we remarked that in order to prevent the creation of ill-shaped elements and to ensure accuracy

when transferring highly localized physical fields, the size ratio should not exceed 2. Consequently, in each loading step, the overall splitting process can be carried out only once. If there are still a large number of elements that need to be divided again according to size indicator, then the loading sequence should be reduced. Attention should also be paid to the initial mesh in which the size ratio should be less than 2.

6.3 Crack surface smoothing

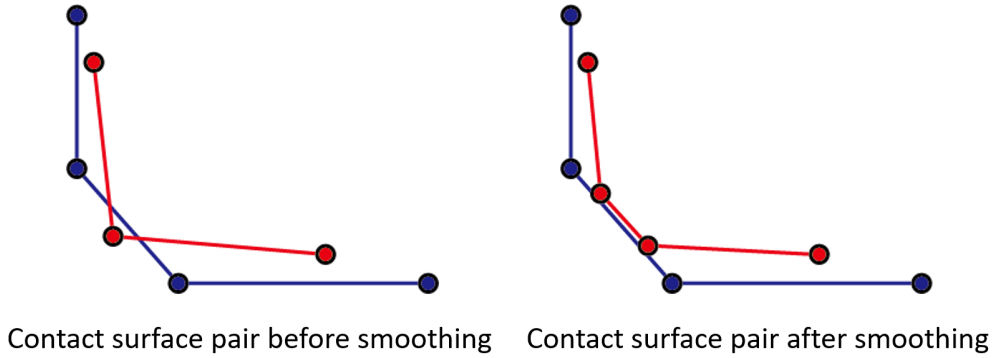


Figure 6-4: The removal of contact surface pair penetration

The representation of cracks based on element deletion process generates the cracks with serrated (saw-tooth) crack surfaces and tips. In the metal forming processes like tensile test, such serrated crack surfaces have no influence on the numerical simulation, because the loading is usually exerted on a free boundary which has no contact with the crack surfaces. Besides, in a tensile test, the cracks are commonly opened by the loading force which means the two crack surfaces of the same crack usually move away from each other so that there will not be contact between these two surfaces. Therefore, if the goal is just to simulate a tensile test, we can leave the crack surfaces serrated. However, if the numerical simulation of a compression test or some other tests related to friction are carried out, the two surfaces of the crack may contact with each other and relative displacement may happen. In such cases, the serrated crack surfaces must be smoothed so that the contact surface pairs can be clearly defined. A clear definition of the contact surface pairs is that the nodes on the two surfaces are located on or near the other surface and none of them penetrate into

the other surface as shown in Fig. 6-4. In addition, if an indentation test is carried out, the surfaces of the specimen will contact with the surface of the indenter. Even if no crack appears, the clear definition of the contact surface pair is still needed. As a consequence, smoothing the surface is the premise of a clear definition of the contact surface pairs. Furthermore, if the coarsening process is taken into account in order to reduce the computational cost, a smoothed surface is required to enable the coarsening process.

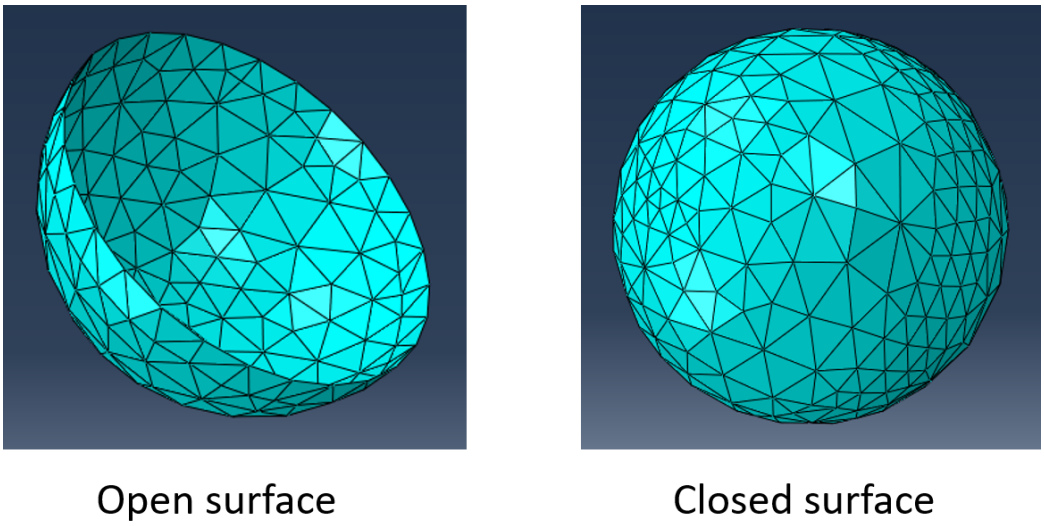


Figure 6-5: An example of closed and open surface

A method called Constrained Catmull-Clark Surface Smoothing Method (CCC-SSM) is proposed to perform local smoothing process on the crack surfaces. This method is based on the Catmull-Clark Surface Subdivision (CCSS) [9] which has the advantage of strong robustness in dealing with arbitrary shape of surface no matter whether it is open or closed in which all the segments are shared by two triangles exactly as in Fig. 6-5. It converges very fast without major changes of the surface mesh topology and it can be adopted to smooth any surfaces composed of polygons, such as triangles or quadrangles. The geometrical quality of the mesh after smoothing is enhanced because the method tends to transform polygons into regular polygons. Even if the satisfying smoothed state should be attained by iterating CCCSSM several times, the computational cost is still at a negligible level. In our work, the surfaces to be smoothed are only the crack surfaces. Given that the cracks propagate step by

step and in each step the serrated crack surfaces are only small parts of the entire crack surfaces, we can assume that the serrated crack surfaces formed in one step are quasi planar as long as the length of the step is small. The method does not preserve the sharp features of the surface, because the crack surfaces are supposed to have no sharp features as in other problems. The CCCSSM is different from CCSS that the number of nodes and elements are constrained to be constant and therefore no new node or element is added into the surface mesh. This global method can be used locally by fixing a chain of nodes which enclose a closed patch of surface and only changing the position of the nodes within the patch. In this section, a brief introduction of the method is first given and then the implementation of the method in the finite element simulation is explained.

6.3.1 Introduction of CCCSSM

The CCCSSM is introduced in this section. Suppose a 3D surface mesh composed of arbitrary polygons (triangles, quadrangles or mixed) is going to be smoothed. This smoothing process is then carried out in 6 steps:

1) For each face in the mesh, a face point FP is added as shown in Fig. 6-6(a). It is defined as the average of all the original vertices of this face.

2) For each edge, an edge point EP is added as shown in Fig. 6-6(b). It is defined as the average of its two original vertices and two face points in the adjacent faces.

3) For each node N, the average of the face points in all of its n adjacent faces is calculated and denoted as CFP $= \frac{1}{n}\sum_1^n$ FP as shown in Fig. 6-6(c).

4) For each node N, the average of the edge points in all of its n adjacent edges is calculated and denoted as CEP $= \frac{1}{n}\sum_1^n$ EP as shown in Fig. 6-6(d).

5) The new position N$'$ of this node is defined as the barycenter of the average of

FP, EP and N as following:

$$N' = \frac{\mathrm{CFP} + 2\mathrm{CEP} + (n-3)N}{n} \tag{6.1}$$

6) The smooth mesh is finally generated by inducing the new positions N′ for all the nodes N, however keep the topology of the mesh unchanged.

7) The above 6 steps are iterated if needed.

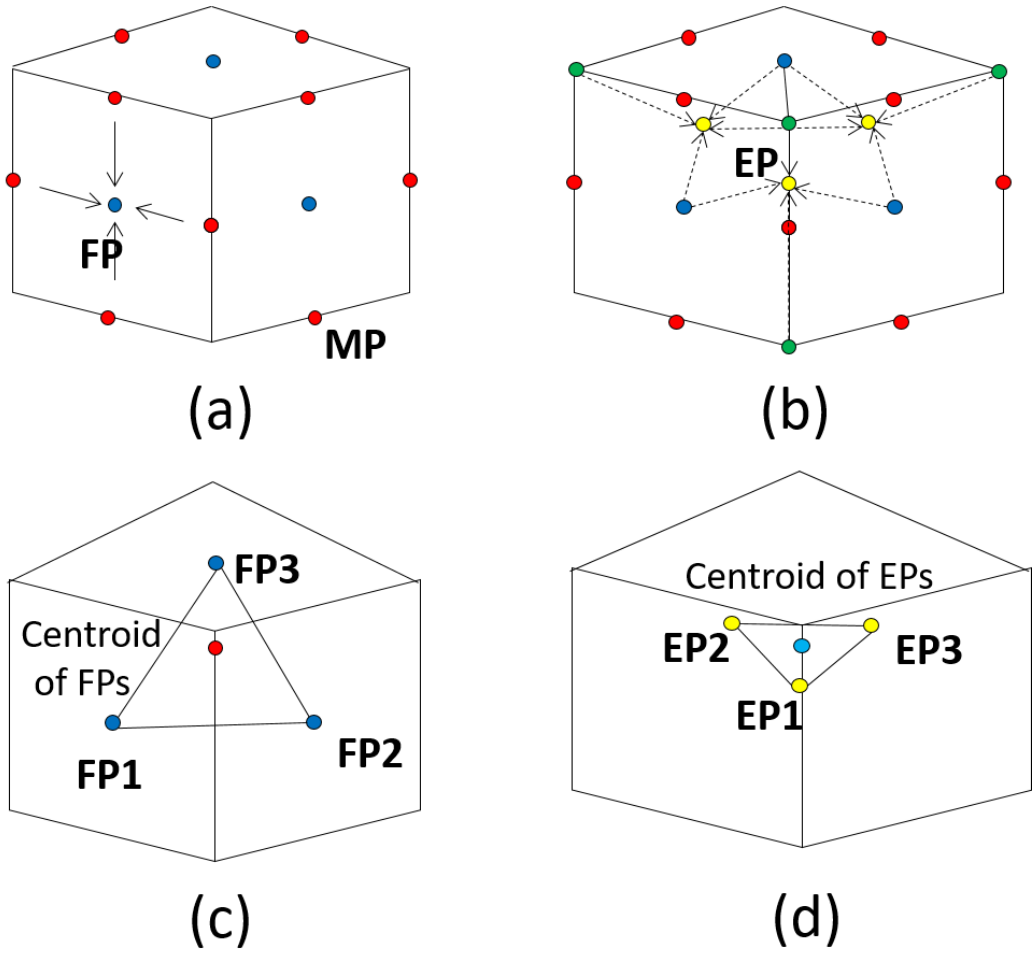


Figure 6-6: The process of Catmull-Clark surface subdivision

6.3.2 Implementation of CCCSSM

As explained above, the smoothing is a local process which is only performed on the cracked surface. In practice, the problem is how to determine the surfaces which

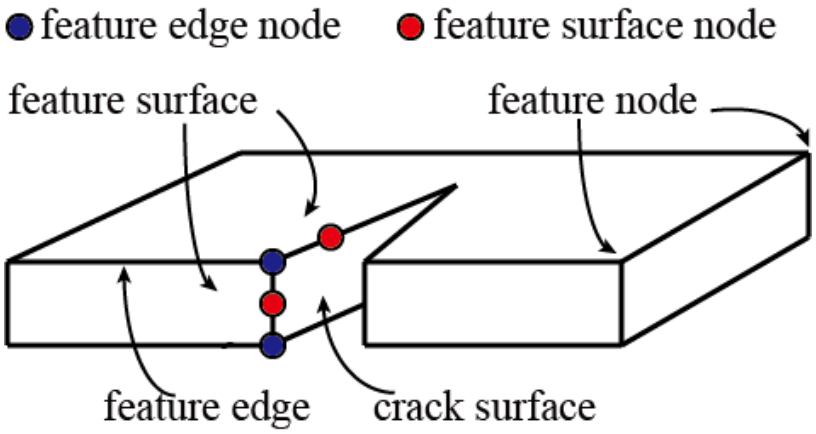


Figure 6-7: The feature topology around the crack

must be smoothed at each loading step. Even if the smoothing process changes the position of the nodes on the mesh surface without following a physical equation, the original features of the geometry should be preserved, such as feature node, feature edge and feature surface (Fig. 6-7) so that no unrealistic crack shape or deformation happen. Therefore, the surfaces to be smoothed should not include such original features even if some of their nodes are included in the crack surface. In addition, damage or cumulative plastic strain fields are always highly localized (high gradient) in the zones around the crack tips which at the same time results in gravely decreasing of element sizes according to the size indicators. The smoothing of the crack tip which is followed by a volume mesh generation can disturb the size of the mesh (making the size of elements somewhere bigger than the size limit given by the size indicator) around the crack tip. The physical fields such as cumulative plastic strain are highly localized in the vicinity of crack tips. If the mesh in these zones are changed, then the mesh size should respect the refinement size indicators so that when the highly localized field are transferred, the numerical diffusion can be minimized. However, the smoothing process does not follow the refinement size indicators. The solution is to smooth only the crack surfaces but not the crack tips. Above all, the crack surfaces to be smoothed should be carefully determined as well as the related node sets.

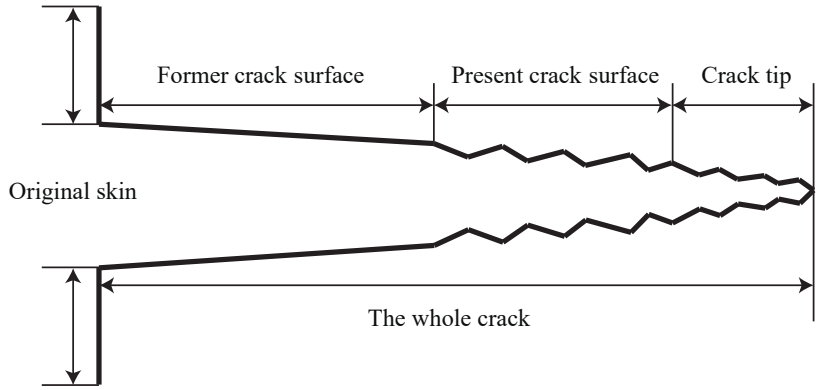


Figure 6-8: The seperation of the crack

6.3.2.1 Seperation of the geometry surface

During tensile test, when cracks propagate with loading sequence increasing, we can divide cracks into two parts, namely crack surfaces and crack tips. The interfaces of elements deleted and kept in present loading step are defined as crack tips and the ones in the former loading step are defined as crack surfaces. In the simulation, we only smooth crack surfaces and keep crack tips intact. Suppose we get a 3D mesh denoted as Ω in which elements in damage zone have been deleted so that cracks appear. The boundary of this mesh (the whole surfaces of the geometry) denoted as $\partial\Omega$ can be partitioned into four parts original skin (OS) denoted as $\partial\Omega_{os}$, former crack surfaces (FCS) denoted as $\partial\Omega_{fcs}$, present crack surface (PCS) denoted as $\partial\Omega_{pcs}$ and crack tip (CT) denoted as $\partial\Omega_{ct}$. This partition is illustrated in Fig. 6-8 and can be expressed as in Eq. 6.2. If we partition the whole surface into two parts: the ones to be smoothed denoted as $\partial\Omega_{smth}$ and the ones to be fixed denoted as $\partial\Omega_{fix}$, the whole surface can then be expressed as in Eq. 6.3a. The surfaces to be smoothed is the present crack surface as in Eq. 6.3b and the surfaces to be fixed is the other part as in Eq. 6.3c.

$$\partial\Omega = \partial\Omega_{os} + \partial\Omega_{fes} + \partial\Omega_{pcs} + \partial\Omega_{ct} \tag{6.2}$$

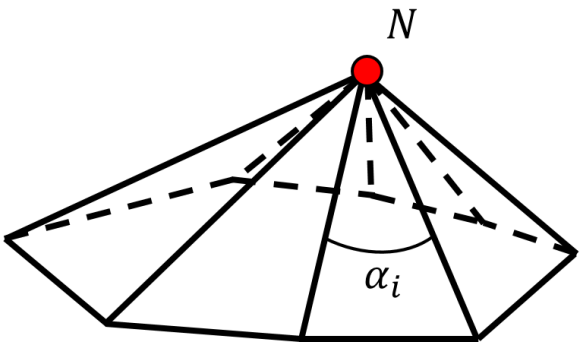


Figure 6-9: Example of sharp node

$$\begin{cases} \partial\Omega = \partial\Omega_{smth} + \partial\Omega_{fix} & \text{(6.3a)} \\ \partial\Omega_{smth} = \partial\Omega_{pcs} & \text{(6.3b)} \\ \partial\Omega_{fix} = \partial\Omega_{os} + \partial\Omega_{fcs} + \partial\Omega_{ct} & \text{(6.3c)} \end{cases}$$

Before explaining the determination of the surface separation, we define two sets which help to formulate the surface separation. Firstly, planar node set is defined and denoted as $(\partial\Omega)_p^N$. This set includes all the nodes which are not defined as sharp nodes. A node N on the surface mesh is regarded as a sharp node, if the summation of the surrounding angles is smaller than a threshold angle. The planar node set can be expressed as in 6.4 in which $(\partial\Omega)^N$ is a node set including all the nodes of $\partial\Omega$. The angle α_i^N is a surrounding angle between two adjacent edges attached to the node N, I is the total number of surrounding angles and ξ^N is the defect angle. This threshold ξ^* is the threshold value which is usually set to 1.8π (strictly planar when it is 2π). Next, planar face set is defined and denoted as $(\partial\Omega)_p^F$. This set includes all the faces (triangles) which have at least one planar node as its vertex. The planar face set can be expressed as in Eq. 6.5 in which $(\partial\Omega)^F$ is a face set including all the faces of $\partial\Omega$.

$$(\partial\Omega)_p^N = \left\{ N | N \in (\partial\Omega)^N; \xi^N \geq \xi^*, \xi^N = \sum_{i=1}^{I} \alpha_i^N \right\} \tag{6.4}$$

$$(\partial\Omega)_p^F = \left\{ F | F \in (\partial\Omega)^F; \exists N \in F, N \in (\partial\Omega)_p^N \right\} \tag{6.5}$$

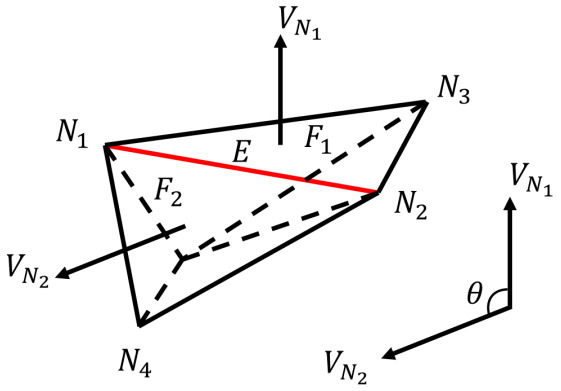


Figure 6-10: Example of sharp edge

As explained, the former crack surfaces $(\partial\Omega)_{fcs}$ are smoothed in the last loading step so that the faces composed of $(\partial\Omega)_{fcs}$ are included in the planar face set $(\partial\Omega)_p^F$. Because the original skin is adapted according to the geometry curvature at the beginning of the simulation, the original skin of the geometry is usually composed of only the planar faces. Therefore, if we extract the complete planar face set $(\partial\Omega)_p^F$ from the whole surfaces $\partial\Omega$), it becomes to the union of $(\partial\Omega)_{fcs}$ and $(\partial\Omega)_{os}$ as expressed in 6.6a. The crack tip as its definition can be easily stored during the totally damaged element deletion process in the same loading step as expressed in Eq. 6.6b. Finally, we substitute Eq. 6.3c, 6.6a and 6.6b into Eq. 6.3a obtaining 6.6c which indicates the determination of the final surfaces to be smoothed.

$$\begin{cases} (\partial\Omega)_p^F \Leftrightarrow \partial\Omega_{os} + \partial\Omega_{fcs} & \text{(6.6a)} \\ (\partial\Omega)_{ct}^F \Leftrightarrow \partial\Omega_{ct} & \text{(6.6b)} \\ \partial\Omega_{smth} = \partial\Omega - \partial\Omega_{fix} \Leftrightarrow (\partial\Omega)^F - (\partial\Omega)_p^F + (\partial\Omega)_{ct}^F & \text{(6.6c)} \end{cases}$$

Thirdly, a ridge edge set is defined as Eq. 6.8 in which $\mathrm{F}_{1,2}$ are two faces attached to edge E and $V_{N_{1,2}}$ are normal vectors of $\mathrm{F}_{1,2}$ which always point outwards from elements (Fig. 6-10).

$$\partial\Omega_{smth} = \partial\Omega - \partial\Omega_{fix} :\Rightarrow (\partial\Omega)^F - (\partial\Omega)_p^F - (\partial\Omega)_{ct}^F \tag{6.7}$$

$$(\partial\Omega)_r^E = \left\{\mathrm{E}|\mathrm{E} \in (\partial\Omega)^E; \mathrm{F}_{1,2} \in (\partial\Omega)_p^F; \theta^E > \theta^*, \theta^E =< V_{N_1}, N_{N_2} >\right\} \tag{6.8}$$

6.3.2.2 Determination of the node set to be smoothed

Before determining the node set to be smoothed, the determination of sharp edges is illustrated in the Fig. 6-10. Two boundary faces F_1 and F_2 have a common edge E. The normal vectors of these two faces point outwards the volume and denotes as $\overrightarrow{V_{n_1}}$ and $\overrightarrow{V_{n_2}}$. The angle between these two normal vectors is denoted as θ. if the angle θ is bigger than a threshold value θ^*, then the common edge E is determined as a sharp edge.

The smoothing process changes the position of the nodes so that before performing it, the exact set of nodes to be smoothed should be determined. As explained in the section 6.3.2.1, even if a node is included in the surface to be smoothed, it may not be moved. As illustrated in Fig. 6-7, some sharp nodes on the original skin of the geometry are denoted as feature nodes N_{ftr}. Some sharp edges on the original skin are denoted as feature edges E_{ftr}. The sharp nodes located along the feature edges are denoted as feature edge nodes N_{fe}. The sharp nodes located on the original skin are denoted as feature surface nodes N_{fs}. In order to avoid unrealistic deformation, the feature surface nodes N_{fs} should be kept on the original skin, the feature edge nodes N_{fe} should keep located along the feature edges E_{ftr} and the feature nodes N_{ftr} should be fixed.

In practice, all the nodes of the present crack surface are smoothed except from feature nodes. After smoothing, the feature edge nodes are projected onto the feature edge and the feature surface nodes are projected onto the original skin. As a results, the original features of the geometry are not destroyed by the smoothing process.

The CCCSSM changes the density of the nodes and tends to relocate the nodes into an isotropic distribution as illustrated in Fig. 6-11. The variation of the density of the nodes changes the element size near the surface in the further step. However, this is not a problem in our case. Because only the crack surface is smoothed during

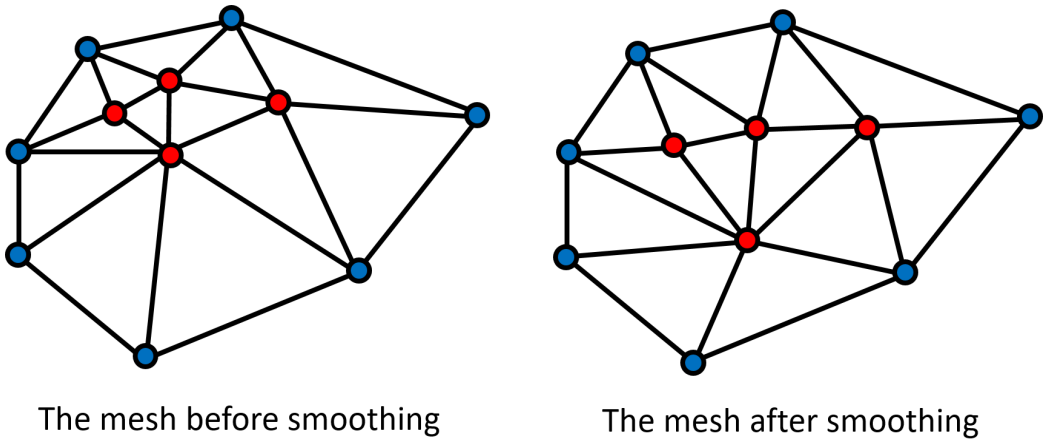
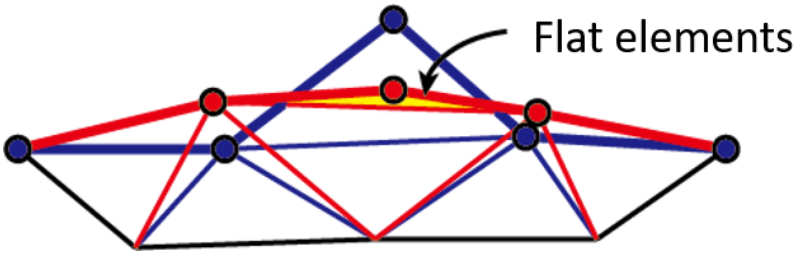


Figure 6-11: The density of the nodes changes caused by smoothing process

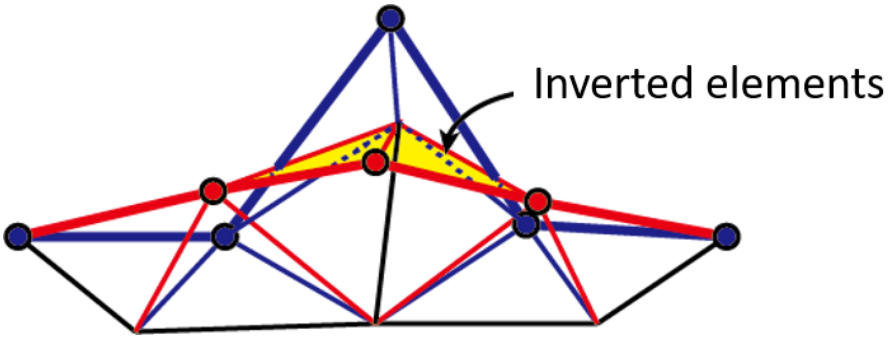
the adaptive process and the mesh size is slightly modified along the crack surfaces only. As explained in section 4, the cracks are represented by deleting a number of totally damaged elements which have sizes smaller than minimum sizes h_{min}. On the one hand, the zones near the crack surface (except from crack tip) are the inactive zones, which means that the mesh size does not influence the crack initiation and propagation so that it is not necessary to keep the node density in these zones; on the other hand, even if the element sizes are changed, they are still smaller than minimum sizes h_{min} which does not deteriorate the quality of our hybrid field transfer process.

6.3.3 Reliability of crack surface smoothing

The smooth process flatens the serrated surfaces. A relatively large displacement of the nodes along the surface of the 3D mesh occurs while the nodes elsewhere are fixed. The topology of the mesh may change at the boundary of the smoothed surface. A 2D example is shown in Fig. 6-12, the blue points and lines are respectively the nodes and edges of the mesh before smoothing; the red points and lines are the nodes and edges of the mesh after smoothing; the black lines are the edges which are not moved during the smoothing; the thick lines are the boundary of the mesh. The Fig. 6-12(a) shows that an element becomes very flat after smoothing. Fig. 6-12(b) shows that some elements are inverted which means one or more vertices of these elements move across their own edges and overlap with the other elements. These



(a) Flat element



(b) Inverse element

Figure 6-12: Example of bad element caused by smoothing

flat or inverted elements prevent the FE calculation from proceeding. Therefore, local surface remeshing should be performed near the boundary of the smoothed surface. This local surface remeshing corrects the topology of the surface mesh and ameliorates the quality of the volume mesh which is generated from the surface mesh.

6.3.4 Surface mesh quality improvement

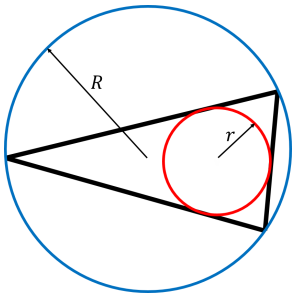


Figure 6-13: Definition of triangle quality

After smoothing the crack surface, we need to generate the tetrahedron mesh near the crack surface and therefore to extract closed triangle meshes. A mesh generator

based on advancing front technique is used [70] and the quality of the tetrahedron mesh is mostly dependent on the quality of the triangle surface mesh. Therefore, the quality of the smoothed surface mesh must be optimized. A number of authors have proposed quality criteria for triangles [84]. In our work, as illustrated in Fig. 6-13, the quality of the triangle is based on radius of its inscribed circle (r) and circumscribed circle (R). Because the ratio r/R equals 0.5 for a regular triangle, the quality of a arbitrary triangle can be defined as the ratio $\frac{r/R}{0.5}$

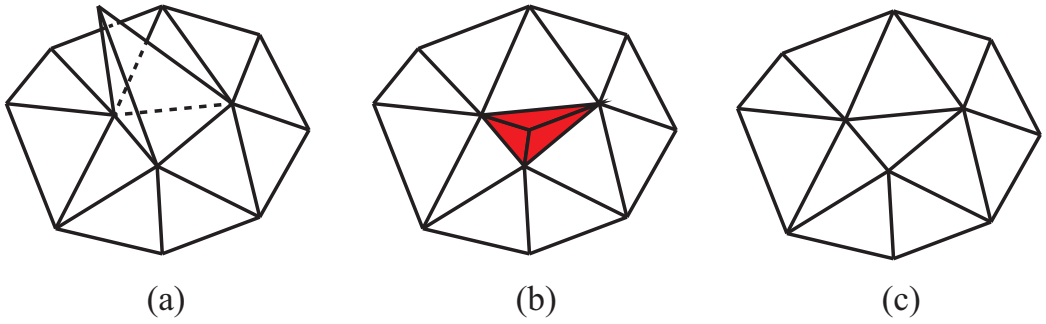


Figure 6-14: Improvement of surface mesh by deleting small elements

The CCCSSM tends to form an isotropic distribution of the nodes on the surface, but does not change the topology of the mesh. This isotropic distribution of the nodes optimizes the mesh quality on the surface in a current topology, but do not optimize the mesh quality in a global view (i.e. if topology is changed, the mesh quality can still be improved). For example, as illustrated in Fig. 6-14 (b), the nodes are distributed isotropically, however, there are 3 triangles in red which are evidently smaller than their neighbors. Such node is then regarded as a "bad node". These 3 small triangles are created by smoothing a sharp node as in Fig. 6-14 (a). Even if the quality of these 3 triangles can be accepted, deleting them may ensure a better quality of tetrahedron mesh. In practice, each node on the surface is checked. If there are 3 triangles connected to this node, then the sizes of the 3 adjacent triangles are calculated. If these 3 triangles have the similar sizes (none of them is 2 times larger than the other) and are smaller than the other neighboring triangles (at least two time smaller), then this node is a "bad node" and is deleted (the adjacent triangles are also deleted). A new triangle is created directly as illustrated in Fig. 6-14 (c).

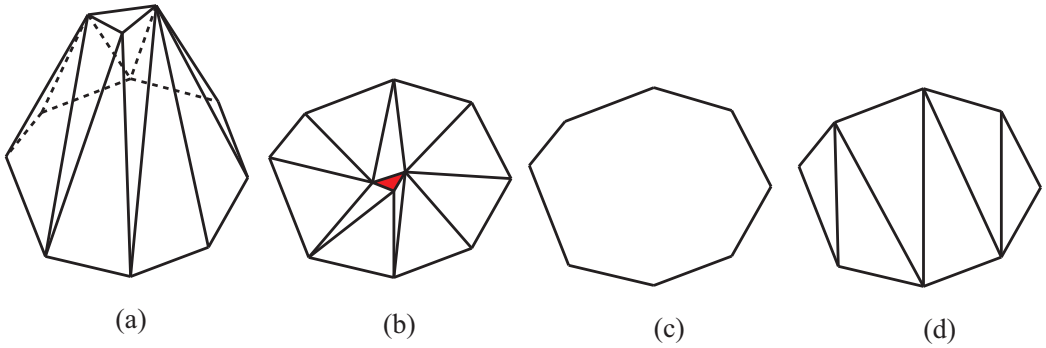


Figure 6-15: Improvement of surface mesh by remeshing

The similar configuration is illustrated in Fig. 6-15 (b). When there is a very small triangle enclosed by larger triangles, we can regard such triangle as "bad" triangle (the red triangle in Fig. 6-15 (b)). This configuration can be generated by smoothing the serrated mesh in Fig. 6-15 (a). If a "bad triangle" is deleted, the topology of the surface mesh is changed. A chain of edges is formed as in Fig. 6-15 (c). We suppose that such edge chain is nearly in a plane after enough smoothing. The patch within this edge chain is triangulated by just connecting the nodes along the edge chain. In practice, each original triangle is checked. If the sizes of all neighboring triangles are twice larger than this center triangle, then this triangle is denoted as "bad triangle". The "bad triangle" as well as the its neighboring triangles are deleted. The triangulation of a planar contour of n nodes into $n-2$ triangles has been widely discussed [23]. The technique is based on the determination of all triangulation configurations. The idea is that a n-node contour can be decomposed into $n-2$ triangles and a $(n-1)$-node contour as shown in Fig. 6-16. When n reaches 10, the number of triangles to compute is 120 and the number of triangulations is 1430. In practice, the determination of the configuration stops at $n = 10$.

All of the basic triangulation configurations for each possible value of n are predesigned. The qualities of triangles in all possible triangulation configurations are calculated. Among them, the configuration which has the highest minimum quality value is chosen as the final triangulation configuration as in Fig. 6-15 (d).

Edge swapping is also an efficient method to locally improve the quality of the surface mesh as illustrated in Fig. 6-17. If the red edge in 6-17 (a) is swapped as in

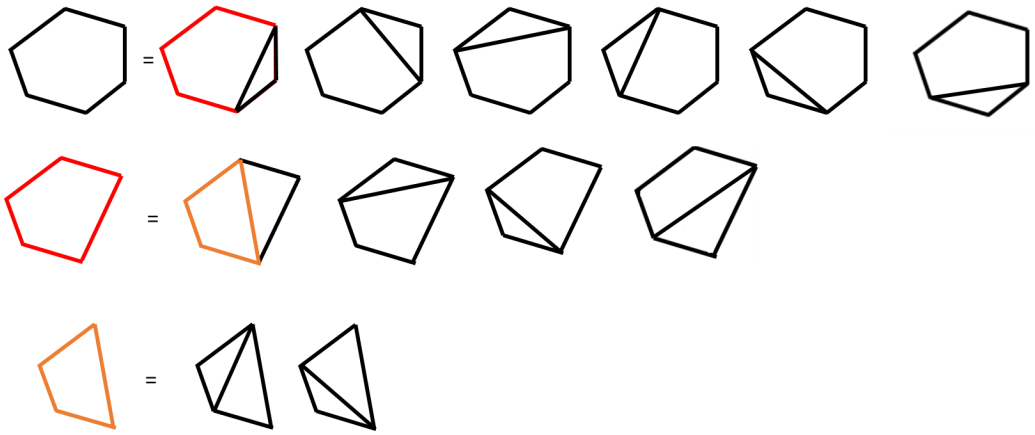


Figure 6-16: The illustration of planar contour meshing

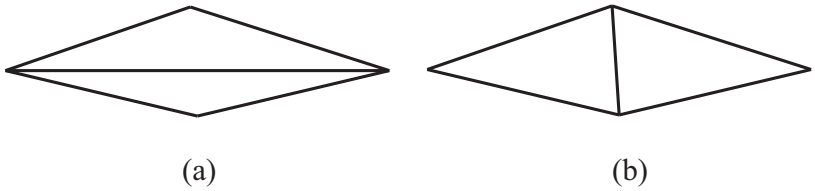


Figure 6-17: Improvement of surface mesh by edge swapping

6-17 (b), then the quality of both adjacent triangles are improved. In this case, the edge is swapped.

6.4 Surface relocating

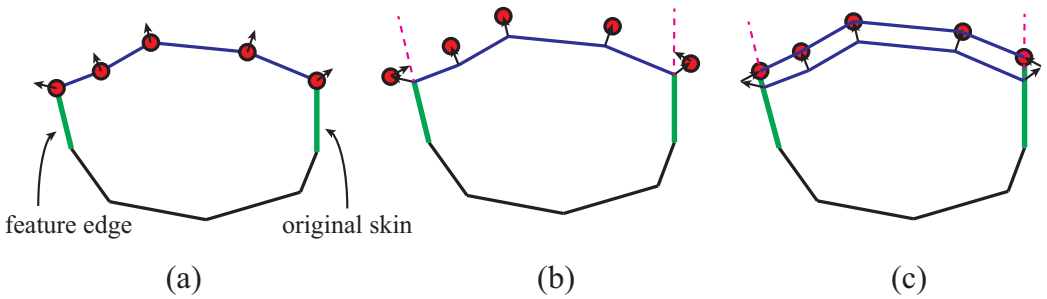


Figure 6-18: Illustration of surface relocation

As explained in section 6.3, the CCCSSM does not control the volume conservation. As a result, after the crack surface is smoothed by CCCSSM, the volume of the specimen is shrunk caused by the geometrical displacement of the nodes on the

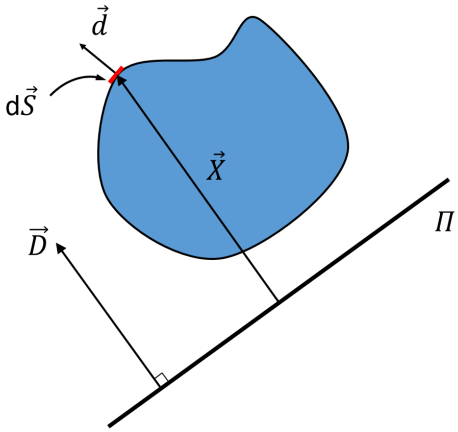


Figure 6-19: Calculation of the volume enclosed by an arbitrary oriented closed surface

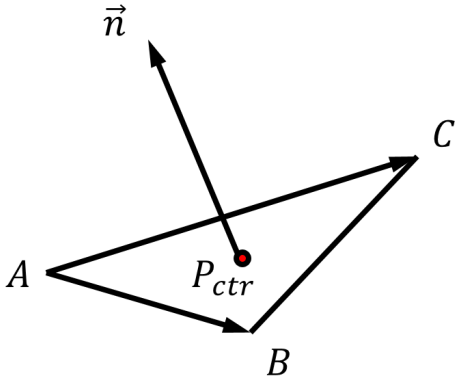


Figure 6-20: A triangle on the surface mesh

surface mesh. However, this shrinkage of the volume is nearly negligible compared to the total volume of the specimen. In our case the crack surface is rather small compared to the whole surface of the specimen. Therefore, the shrinkage of the volume of the specimen can be compensated by relocating the smoothed surface with a small identical displacement along the local normal direction of the surface at each node on the crack surface, as illustrated in Fig. 6-18 (a). The relocation only concerns about the nodes on the crack surface, whereas not all of them. As mentioned in section 6.3.2.2, the feature node, feature edge and original skin should not deformed artificially when the surface is relocated. The same projection for feature edge nodes and feature surface nodes are made after relocating the surface, as illustrated in Fig. 6-18 (b). This surface relocating method can be easily implemented into numerical calculation and does not destroy the geometry of the crack surface, as illustrated in 6-18 (c).

In practice, the surface relocation is the problem to calculate the displacement on the smoothed surface. The volume before surface smoothing is calculated and denoted as V_0 which is the volume needed after surface relocating. The volume after surface smoothing is calculated and denoted as V_{smth}. Based on the small volume variation assumption, we can suppose that the difference between these two volume

has a linear relationship with the identical displacement $\overrightarrow{d}$. The volume variation caused by feature edge node and feature surface node projection can be regarded as infinitesimal and neglected. Therefore, the length of identical displacement can be obtained by Eq. 6.9.

$$||\overrightarrow{d}|| = \frac{V_0 - V_{smth}}{||\overrightarrow{S}_{smth}||} \tag{6.9}$$

in which $||\overrightarrow{S}_{smth}||$ is the area of the smoothed surface and $|\overrightarrow{d}|$ is the length of identical displacement $\overrightarrow{d}$. For a finite element surface mesh using triangles, the area the smoothed surface and $|\overrightarrow{d}|$ is the sum of all the areas of the triangles.

In order to calculate the length of the identical displacement, we introduce a simplified way to calculate the volume V enclosed by an arbitrary oriented closed surface $\partial\Omega$. All the surfaces are outer-pointing normal surfaces. As illustrated in Fig. 6-19, we chose an arbitrary referenced direction and its unit direction vector is denoted as $\overrightarrow{d}$ and set a referenced plane $\prod$ which is vertical to the referenced direction $\overrightarrow{d}$ and has the whole surface $\partial\Omega$ at one side of itself. We suppose that $d\overrightarrow{S}$ is a small oriented piece on the surface $\partial\Omega$. If the area of the small piece is denoted as dA and the unit normal vector of the small piece is denoted as $\overrightarrow{n}$, then the small piece can be expressed as: $d\overrightarrow{S} = \overrightarrow{n}dA$. The center point of the small piece $d\overrightarrow{S}$ is denoted as P_S. The vector starting from the referenced plane $\prod$, going along the referenced direction $\overrightarrow{d}$ and ending at the point P_S is denoted as $\overrightarrow{X}$. We call the vector $\overrightarrow{X}$ as position vector. Because the piece is very small, we can assume that the distance from this small piece $d\overrightarrow{S}$ to the referenced plane is the distance between its center and the referenced plane. If this distance is denoted as l, then this position vector can be expressed as: $\overrightarrow{X} = l\overrightarrow{d}$. The volume of this surface V can be expressed as in Eq. 6.10.

$$V_{\partial\Omega} = \oiint_{\partial\Omega} \overrightarrow{X} \cdot d\overrightarrow{S} = \oiint_{\partial\Omega} l\overrightarrow{d} \cdot \overrightarrow{n}dA \tag{6.10}$$

$$\overrightarrow{V_n} = \overrightarrow{AB} \times \overrightarrow{AC} = 2A_i\overrightarrow{n_i} \tag{6.11}$$

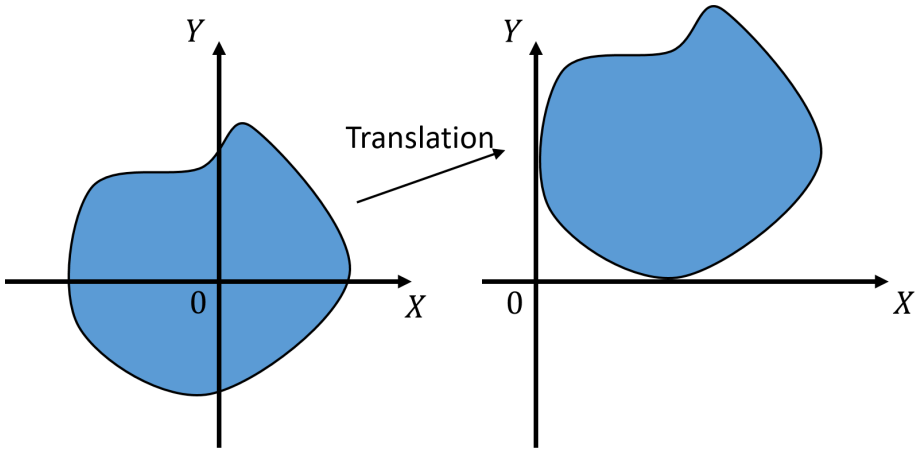


Figure 6-21: Translation of the mesh to the first quadrant

In terms of finite element triangle surface mesh, the volume V can be calculated by discretizing the Eq. 6.10. Firstly, in the Cartesian coordinate system, the surface is translated to the first quadrant ($x >= 0, y >= 0, z >= 0$) as illustrated in Fig. 6-21. Secondly, as illustrated in Fig. 6-20, the triangle ABC is the ith element of the surface mesh which has 3 vertexes A(x_A^i, y_A^i, z_A^i), B(x_B^i, y_B^i, z_B^i) and C(x_C^i, y_C^i, z_C^i). The point $P_{ctr}^i(x_{ctr}^i, y_{ctr}^i, z_{ctr}^i)$ is the center of the triangle ABC in which $x_{ctr}^i = (x_A^i + x_B^i + x_C^i)/3$, $y_{ctr}^i = (y_A^i + y_B^i + y_C^i)/3$ and $z_{ctr}^i = (z_A^i + z_B^i + z_C^i)/3$. The vector $\overrightarrow{V_n}^i(x_{V_n}^i, y_{V_n}^i, z_{V_n}^i)$ is the normal vector of the triangle ABC. If the area of the triangle ABC is denoted as A_i, then the normal vector $\overrightarrow{V_n}$ can be expressed as in Eq. 6.11. Thirdly, the position vector of the triangle ABC is denoted as $l_i \overrightarrow{d}$ and the discretization of Eq. 6.10 can be expressed in Eq. 6.12. Finally, the axis oz is chosen as the referenced direction and therefore the plane xoy is set as the referenced plane. In this case, we can substitute the term $A_i \overrightarrow{d} \cdot \overrightarrow{n_i}$ by $\frac{1}{2} z_{V_n}^i$ and substitute l_i by z_{ctr}^i in the Eq. 6.12 so that to get Eq. 6.13.

$$V_{\partial\Omega} = \sum_{i=1}^{N_{tri}} l_i A_i \overrightarrow{d} \cdot \overrightarrow{n_i} \tag{6.12}$$

$$V_{\partial\Omega} = \sum_{i=1}^{N_{tri}} z_{ctr}^i \frac{1}{2} z_{V_n}^i = \frac{1}{6} \sum_{i=1}^{N_{tri}} z_{ctr}^i (z_A^i + z_B^i + z_C^i) \tag{6.13}$$

6.5 Example of surface smoothing process with an analytical crack

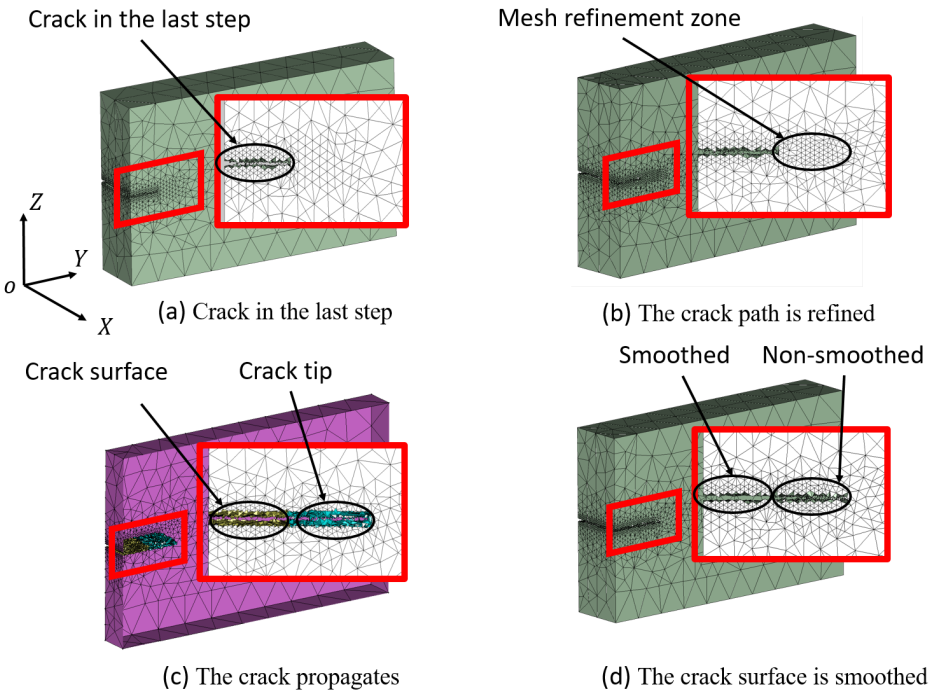


Figure 6-22: An example of crack smoothing in a loading step

In this section, an analytical example of crack smoothing integrated in a loading step is displayed, as illustrated in Fig. 6-22. In this example, the geometry is a cuboid which is discretized by tetrahedron elements and the crack is set to propagate along the middle plane of the cuboid parallel to the XoY plane. In Fig. 6-22 (a), a crack initiates at last step is displayed. The mesh is then refined along the crack propagation direction analytically as illustrated in Fig. 6-22 (b). The crack propagates by deleting the elements which are intersected by the central plane of this cube and the whole surface mesh of the geometry is illustrated in Fig. 6-22 (c) with a cutaway view. As discussed in section 6.3.2.1, the surface of the crack in last step is the crack surface which must be smoothed. The surface of the crack in this step is the crack tip which must be keep untouched. The final result (after smoothing and relocating), as shown in Fig. 6-22 (d), is detailed thereafter.

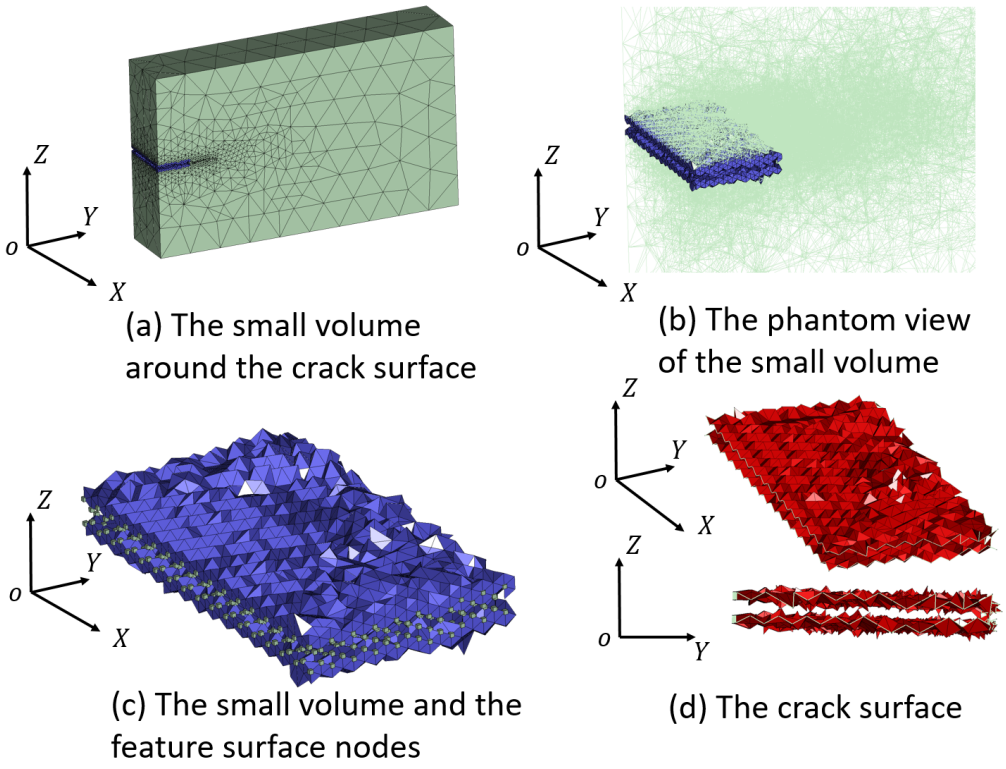


Figure 6-23: Implementation of CCCSSM in the analytic example

Our surface smoothing is a local process. It is not necessary to consider the whole geometry surface at the same time even if the surface separation process is performed. In the example, we only consider a small part around the crack. As illustrated in the Fig. 6-23 (a), the small part around the crack is composed of the set of blue elements. This small part is obtained by searching two layers of elements around the nodes of the crack surface by $E_{blue} = \mathrm{LS}(\mathrm{N}_{cs}, 2)$ which can be seen clearly in the Fig. 6-23 (b). The nodes (grey cubes) which are going to be smoothed is illustrated in Fig. 6-23 (c) and the surface which is finally smoothed by CCCSSM and the nodes (white lines) which are going to be projected on the feature edges and feature surfaces are illustrated in Fig. 6-23 (d).

The CCCSSM is implemented on the red surface in Fig. 6-23 (d) and the smoothed surface is displayed in Fig. 6-24 (a). If we look at the smoothed surface along the oz axis as in Fig. 6-24 (b), we can find that there are some "bad" elements which have low quality and the small elements which more than two times smaller than

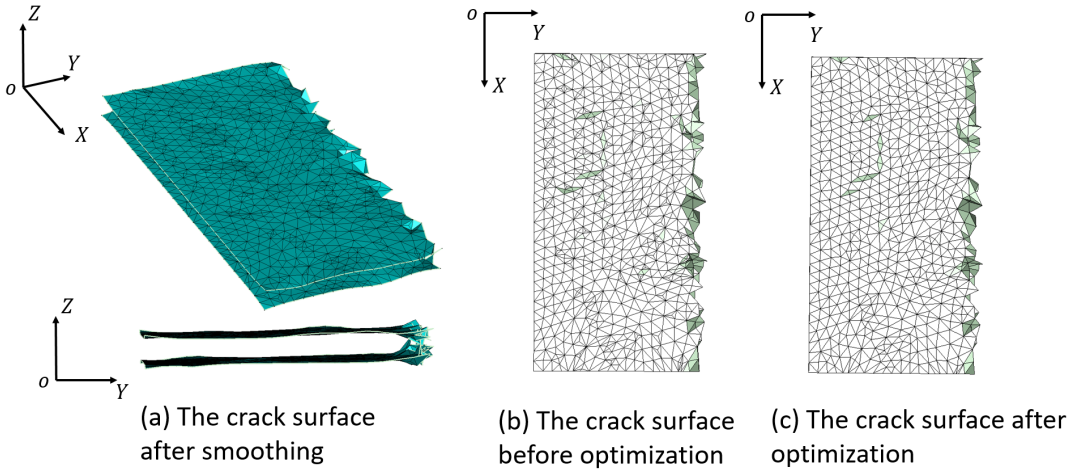


Figure 6-24: Illustration of surface quality optimization

the others. We optimize the smoothed surface as explained in section 6.3.4 and the optimized result is displayed in Fig. 6-24 (c).

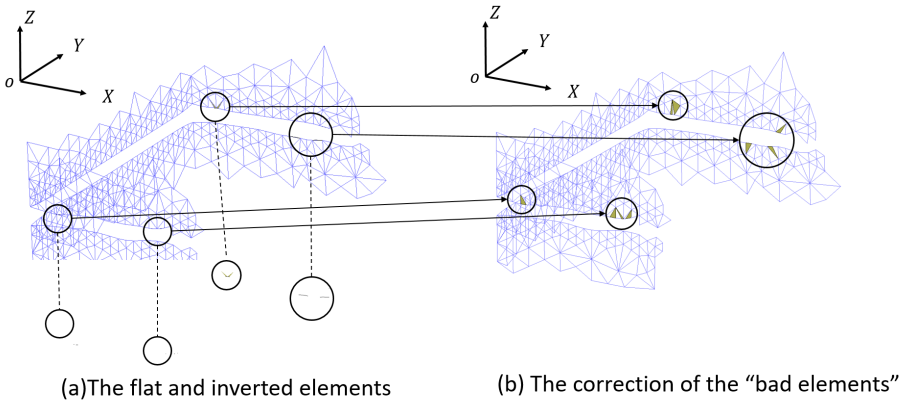


Figure 6-25: Reliability verification of the mesh topology after smoothing and relocating

The crack surface is relocated after being smoothed in order to keep the conservation of the volume. The feature edge nodes and the feature surface nodes are still projected onto the feature edge and the original surface. The mesh topology around the interface between crack surface and the original surface may be destroyed by the projection of the nodes. We need to check the topology. As illustrated in Fig. 6-25 (a), in this example we find 4 places where there are inverse elements. These inverse elements are repaired and the new topology of the mesh is corrected and displayed in

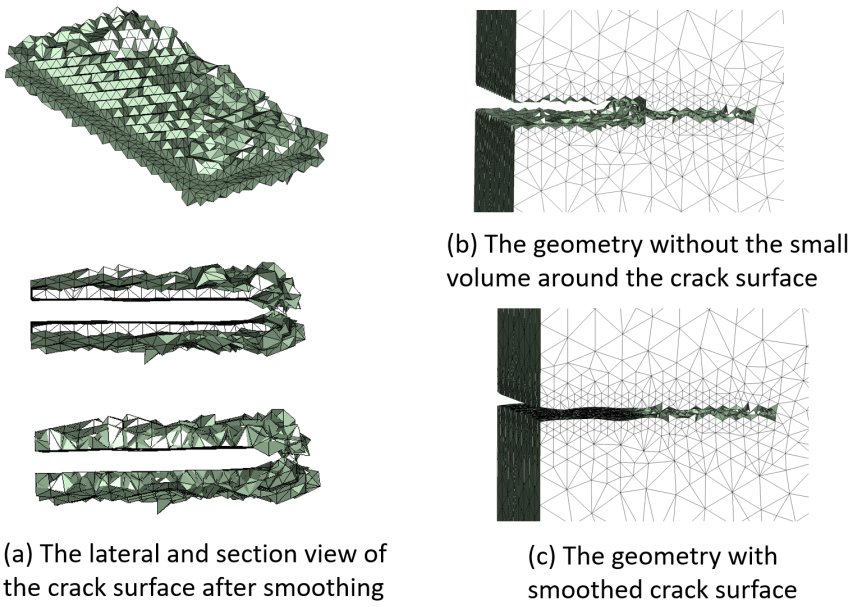


Figure 6-26: Mesh generation within the small part around the smoothed crack surface

Fig. 6-25 (b).

The last step is the mesh generation [72] within the closed surface of the small part around the smoothed crack surface, as illustrated in Fig. 6-26 (a). The other part of the whole geometry is displayed in Fig. 6-26 (b). The two parts are assembled together and the final result is illustrated in Fig. 6-26 (c) in which we can see that the original surfaces and the feature edges are not spoiled by the smoothing or the relocating process and the mesh quality remains good.

Chapter 7

Field transfer

7.1 Introduction of field transfer implemented in local remeshing

Finite Element Method (FEM) in context of pure Lagrangian simulations has been widely discussed in the literature. In the simulation of metal forming processes involving non-linear, elastic-plastic large deformation, the mesh based on pure Lagrangian equations is usually severely distorted and encounters large entanglement. Such deterioration of the mesh may lead to a large loss of accuracy in the subsequent analysis or even prevent from converging. Even if arbitrary Lagrangian-Eulerian (ALE) simulation [27] can be used to avoid frequent remeshing, the discontinuity such as crack propagation must still be handled. Numerous authors propose adaptive methodology with a rezoning process to solve the problem. The rezoning process is usually composed of three steps: firstly, geometrical or physical error indicators are chosen to determine the features of the new mesh (the zones to be remeshed and the mesh size). Secondly, the new mesh is generated either locally or globally. Finally, the necessary physical state variables both at nodal points and at integration points (if needed) are transferred from the old mesh to the new mesh. In this work, we give the main emphasis on the transfer step. When dealing with all Lagrangian finite element analysis, the following general requirements should be followed during the transfer process:

self-consistency, conservation of equilibrium and respect of constitutive equations. In practice, in terms of different analysis, different requirements are needed, for example minimization of numerical diffusion [64] and two directions of transfer between dissimilar meshes [28]. In this work, we focus on field transfer process implemented in metal forming process with crack initiation and propagation. In this analysis, on the one hand, some physical fields have large gradient such as cumulative plastic strain field and localize in a small zone such as the vicinity of a crack tip. The numerical diffusion caused by field transfer should be strictly limited, otherwise, the large gradients and high localization cannot be captured after the transfer process. On the other hand, because the path of the crack is always along the trajectory of the maximum damage value during the simulation, the crack propagation, especially in 3D, is very sensitive to this maximum damage which determines the direction of the crack propagation. As a consequence, in our work, the preservation of extrema of the field is of great importance.

Different transfer processes have been investigating in order to deal with various adaptive methodologies. The evaluated space is discretized by finite elements. We define 4 characteristic discretized positions, namely, nodal points of the old mesh denoted as NP^{old}, integration points of the old mesh denoted as IP^{old}, nodal points of the new mesh denoted as NP^{new} and integration points of the new mesh denoted as IP^{new}. With respect to the data flow during the integration field transfer process in which the data flow begins at IP^{old} and ends at IP^{new}, transfer methods can be classified into 3 groups.

In the first group, the state variables at integration points of the old mesh are extrapolated to the nodal points of the old mesh and the values at the integration points of the new mesh are interpolated using element shape function of old mesh or by meshless interpolation based on NP^{old}. This process can be expressed as: $IP^{old} \rightarrow NP^{old} \rightarrow IP^{new}$ [88][49][39]. The extrapolation at NP^{old} can be obtained by either averaging [49] or by weighted averaging[21] the values at IP^{new} around each nodal point. These methods are quick. The extrapolation can be performed continuously by finite element function based least square methods [35] in which

the discrete state variables at integration points are smoothed continuously at the nodal points. The number of integration points should be sufficient to interpolate the field to the element nodal points. The continuous fields can also be obtained after the discrete extrapolation at nodal points [54]. Most of the authors [39][63] use finite element shape function interpolation to calculate the values at IP^{new} from NP^{old}. The principle is to locate an integration point of the new mesh in a containing element of the old mesh [88].

In the second group of transfer, the nodal points in the new mesh are added in the scheme transfer and this process can be expressed as: $IP^{old} \rightarrow NP^{old} \rightarrow NP^{new} \rightarrow IP^{new}$. Peric et al. [64] propose such kind of transfer process: the first part $IP^{old} \rightarrow NP^{old}$ is performed by finite element shape function based least squares [35]. The second part $NP^{old} \rightarrow NP^{new}$ and the third part $NP^{new} \rightarrow IP^{new}$ are carried out using finite element shape functions interpolation in the old and new mesh respectively. J. Mediavilla et al. [56] follow this kind of transfer process but propose to transfer only a minimal set of variables in order to eliminate the inconsistencies [29]. This minimal set should be composed of fully representative variables while the remaining ones are derived from the constitutive equations. Javani et al. [38] propose a transfer process based on the work of Mediavilla et al.. Instead of local extrapolation in the first part, the authors use a global least square smoothing based on shape functions. After the reconstruction of all the fields, balancing iterations are done to restore global equilibrium by assuming the material behavior to be elastic in order to guarantee convergence. In the context of metal forming processes, the cumulative plastic strain is very large in the zone where the fracture occurs, the elements of the old mesh in such zones are usually severely distorted which ruins the shape function based interpolations. As a solution, instead of using shape functions, to perform $NP^{old} \rightarrow NP^{new}$, Dureisseix et al. [28] propose to use a mortar-like procedure in which the energy conservation is preserved piece-wisely. Ortiz and Quigley [62] uses a transfer process based on Hu-Washizu principle in order to keep consistency together with issues on conservation of equilibrium.

In the third group, state variables at integration points in old mesh are di-

rectly transferred to integration points of the new mesh which can be expressed as: $IP^{old} \rightarrow IP^{new}$. Some authors [33][85] use a weighted averaging method based on the square of inverse distance to transfer the state variables directly from old mesh to new mesh. However, when the mesh has a big gradation or a higher order element is used, such averaging methods encounter a decline of extrema of the fields after transfer which is rarely discussed in the literature. Pires et al. [65] propose a moving least square (MLS) technique using cosine shaped approximation weights. In this technique, the selection of information points only depends on distance. A similar method is proposed by Labergere et al. [45]. The approximation weights are replaced by diffuse interpolation weights which also only depends on distance. The main advantages of these MLS based methods are that the transferred results are not influenced by the quality of the elements in old mesh and any type of elements can be used. We remark that the topology of the mesh contains the distribution of information points with respect to the evaluation point on both distance and direction aspects. Neglecting the topology of the mesh and selecting information points only by the nearest distance in these meshless techniques lead to a loss of direction information. The experience shows that much attention must be paid to the direction aspect when selecting information points, especially when the gradient of the field and the mesh size gradation become large. Otherwise, much numerical diffusion is generated. This topic is detailed in the following section. In addition, according to most authors, the use of meshless methods in the transfer scheme $IP^{old} \rightarrow IP^{new}$ reduces the numerical diffusion and preserves the extrema values. As a drawback, consistency and equilibrium may be lost [63] [56]. In order to get consistent fields after transfer, Rashid [69] propose a method based on weak enforcement of equality between piecewisely constant fields on two meshes. The efficiency of the technique is based on an approximated determination of the volume intersection between the two meshes by boxes. In this scope, Brancherie et al. [10] propose a technique to ensure consistency and equilibrium. The equilibrium is satisfied locally by imposing null divergence of stress field when choosing the basis function to construct the interpolation. The respect of the constitutive equations is obtained at a computational

cost which could be prohibitive in a 3D context. Khoei et al. [42] propose a modified superconvergent patch recovery (MSPR) technique combining with a global 3D remeshing with moderate mesh size gradation. The basis functions to construct the interpolation around the crack tip are based on local polar coordinated trigonometric functions. Even if the gradation of the mesh can be controlled, a global remeshing together with a global transfer increase the cost of the simulation and the potential effects of numerical diffusion.

The above brief survey shows that a reliable transfer process should maintain the consistency of the constitutive equations and satisfy the equilibrium equations, while minimizing numerical diffusions[64]. However, if all the state variables are transferred to the new mesh independently, an unsatisfied equilibrium state at the beginning of the next adaptive step might happen. Cao [17] proposed a general method to solve this problem and ensure consistency and equilibrium. In this method, an additional small step is inserted before the next adaptive loading step. In this small step, the large viscous forces are imposed to the geometry until the consistency and equilibrium are retained (the extra forces vanish) and all the fields are smoothed by the constitutive equations. Cao notes that the possibility of convergence and the speed of convergence depend on the numerical diffusion caused by the transfer process. This additional step converges very fast, if the numerical diffusion is small. Following on from Cao's work, we propose a transfer operator which minimizes the numerical diffusion, leading to a very rapid convergence of consistency and recovery of equilibrium. Given the advantages of minimizing numerical diffusion in MLS-based meshless methods and of maintaining as much consistency as possible in shape function interpolation, we propose a hybrid transfer operator as a fourth group incorporating the two methods: $IP^{old} \rightarrow NP^{new} \rightarrow IP^{new}$: first, state variables at integration points in the old mesh are transferred by the Diffuse Interpolation technique to nodal points in the new mesh, and a new information point selection strategy is used to improve the robustness of the Diffuse Interpolation method; second, the state variables are interpolated by shape function from the element in the new mesh to the integration points in the new mesh in order to obtain the fields with greater consistency [56]. The consistency

and equilibrium are recovered using Cao's method. We discuss the details of our transfer operator in Section ??. In Section 7.6 we transfer an analytic field using different transfer operators and compared the results. In Section ?? we then use different transfer operators to simulate the metal-forming process and we compare the initiation and propagation of cracks are compared to experimental results.

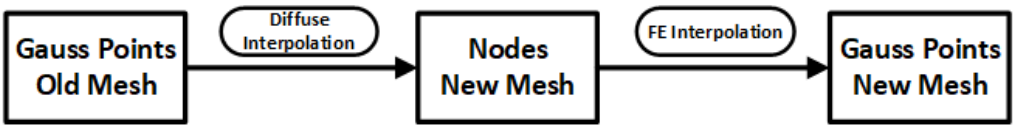


Figure 7-1: The procedure of hybrid field transfer operator

7.2 The Diffuse Approximation method

The Diffuse Interpolation method has been widely discussed in the literature [44][11] and only a short reminder is given in this section. The method is aimed to reconstruct the local scalar field $S(\mathbf{X}-\mathbf{X}_0)$ around the vicinity of the evaluation point $M(\mathbf{X}_0)$ by a polynomial function based approximation as expressed in Eq.7.1 in which $\mathbf{P}(\mathbf{X}-\mathbf{X}_0)$ is the basis of the approximation, $\mathbf{A}(\mathbf{X}_0)$ is the unknown coefficient vector which depends on the location of the evaluation point $\mathbf{X}_0$ and $\mathbf{X}$ is the global coordinate components vector.

$$S(\mathbf{X}-\mathbf{X}_0) = \mathbf{P}^T(\mathbf{X}-\mathbf{X}_0)\cdot\mathbf{A}(\mathbf{X}_0) \tag{7.1}$$

In this work, we propose a linear approximation in 3D Cartesian coordinate system to perform the approximation. Because this linear approximation works more robust than quadratic or higher order approximation, when the mesh size is well reduced according the the plastic strain and damage. Therefore, Eq.7.1 can be developed as:

$$S(\mathbf{X}-\mathbf{X}_0) = \begin{bmatrix} 1 & x-x_0 & y-y_0 & z-z_0 \end{bmatrix} \cdot \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} \tag{7.2}$$

The coefficient vector $\mathbf{A}$ is obtained by minimizing the following objective function Eq.7.3:

$$\begin{aligned} \text{Min} : J_{\mathbf{A}}\left(\mathbf{X}-\mathbf{X}_0\right) &= \sum_{i=1}^{n} w_i^{\infty}\left(S\left(\mathbf{X}_i-\mathbf{X}_0\right)-S_i\right)^2 \\ &= \sum_{i=1}^{n} w_i^{\infty}\left(\mathbf{P}^T\left(\mathbf{X}_i-\mathbf{X}_0\right)\cdot\mathbf{A}-S_i\right)^2 \end{aligned} \tag{7.3}$$

in which n denotes the number of information points to build the approximation, $\mathbf{X}_i$ is the coordinates of the information point M_i and S_i is the value at M_i. The interpolating weight w_i^{∞} is given by Eq.7.4, in which $R\left(\mathbf{X}_0\right)$ denotes the radius of the vicinity. This interpolating weight is close to zero when the information point is located near the boundary of the vicinity and becomes singular when the information point is located close to M_0. In practice, in order to enclose all information points, the radius is slightly increased and defined as $R\left(\mathbf{X}_0\right)=1.05\text{max}\left(\left\|\mathbf{X}_i-\mathbf{X}_0\right\|\right)$ [45].

$$w_i^{\infty}=\frac{w_i}{1-w_i}, \quad \text{with } w_i=\left(1-\frac{\left\|\mathbf{X}_i-\mathbf{X}_0\right\|}{R\left(\mathbf{X}_0\right)}\right)^2\cdot\left(1+\frac{2\left\|\mathbf{X}_i-\mathbf{X}_0\right\|}{R\left(\mathbf{X}_0\right)}\right) \tag{7.4}$$

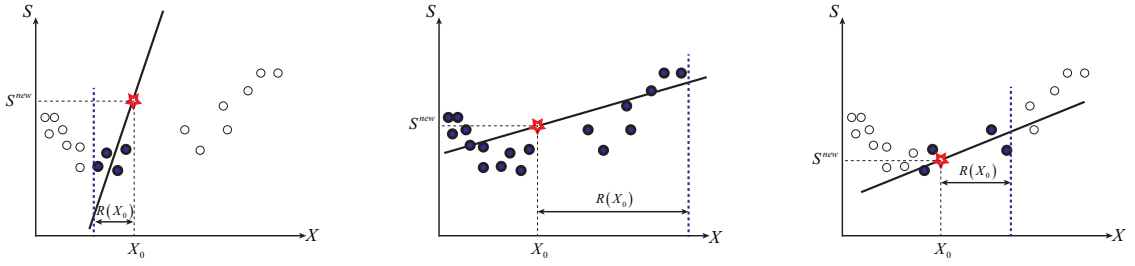
The minimization is carried out by solving linear equations: $\frac{\partial J_{\mathbf{A}}}{\partial \mathbf{A}}=0$. Coefficient vector $\mathbf{A}$ is obtained by solving Eq. 7.5 where $\widetilde{\mathbf{X}}$ is the local coordinate matrix centered at M_0. $\mathbf{W}$ is the diagonal interpolating weight matrix. The value at the evaluation point M_0 is given by α_0.

$$\left(\widetilde{\mathbf{X}}^T\mathbf{W}\,\widetilde{\mathbf{X}}\right)\mathbf{A}=\widetilde{\mathbf{X}}^T\mathbf{W}\,\mathbf{S} \tag{7.5}$$

$$\text{with } \mathbf{X}=\begin{bmatrix} 1 & x_1-x_0 & y_1-y_0 & z_1-z_0 \\ 1 & x_2-x_0 & y_2-y_0 & z_2-z_0 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & x_n-x_0 & y_n-y_0 & z_n-z_0 \end{bmatrix},$$

$$\mathbf{w}=\begin{bmatrix} w_1^{\infty} & 0 & \cdots & 0 \\ 0 & w_2^{\infty} & 0 & \vdots \\ \vdots & 0 & \ddots & 0 \\ 0 & \cdots & 0 & w_n^{\infty} \end{bmatrix}, \mathbf{A}=\begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix}, \mathbf{S}=\begin{bmatrix} S_1 \\ S_2 \\ \vdots \\ S_n \end{bmatrix}.$$

The results of the Diffuse Interpolation method strongly depends on the distribution of the n information points M_i. A linear 1D example with different information point selection strategies is illustrated in Fig.7-2. The coordinates of the points are plotted on the horizontal axis and the values at the points on the vertical axis. The points in the vicinity of the evaluation point M_0 are plotted as circles. The selected information points are plotted as the blue circles. The black line denotes the interpolation function which is used to estimate the value S^{new} at the evaluation points $M_0\,(\mathbf{X}_0)$ denoted as red star. The estimated value from the selection based only on distance (the closest points [45][10]) is illustrated in Fig. 7-2(a). Only the points on the left side of the evaluation point are selected as information points and the ones on the right side are missed because they are relatively far from the evaluation point. As a result, even if a sufficient number of points to build the approximation is selected, the estimated value is overestimated and the gradient of the field may not be reproduced faithfully. Another information point selection strategy selects all the neighboring points within the vicinity of the evaluation point, no matter how far they are from M_0. Even if the points in all directions (left and right side) are considered, a high number of information point is selected. Using a linear basis, only 2+1 information points [11] are necessary to build this 1D interpolation. This selection strategy also leads to an overestimation of the value at M_0 therefore numerical diffusion arises, because the interpolation is no longer local.



(a) Based on distance only (b) Based on direction only (c) Based on distance and direction

Figure 7-2: 1D linear diffuse interpolations using different information point selection strategies

The numerical diffusion caused by these two selection strategies is more severe as

the mesh gradation increases, especially in a 3D context. In order to minimize the numerical diffusion and get a more reliable interpolation result, an isotropic information point selection based on both distance and direction is proposed. The points in every direction (right and left side) are considered and only the nearer ones are selected as information points as shown in Fig.7-2(c). The numerical diffusion is limited at a large extent while capturing the gradient of the fields.

7.3 Local vicinity of the evaluation point—Determination of neighboring points

The information points should be as close as possible to the evaluation points. In addition, a visibility criterion [10][71] should be considered. A visibility criterion prevents for instance from choosing points on different sides of the crack lips as illustrated in Fig. 7-3. In our transfer method, the potential information points are the integration points of the old mesh. Therefore, it is efficient to use element connectivity, which provides an efficient way to determine a local set of neighboring points while satisfying a visibility criterion, to determine the local vicinity of the evaluation point. An evaluation point is first chosen. The element which contains the point is then determined. This containing element is also defined as the first element layer around the evaluation point. The second layer is composed of all elements which share any one of the vertices of the first layer element except from the containing element. Other layers can be defined using the same process. All the integration points of the elements in these layers are searched as the neighboring points within the vicinity. In practice, no more than three layers are used, otherwise the neighboring points are no longer local. Some authors regard all of these neighboring points directly as information points in a 2D context [10][45]. In this work, a 4-monomial-linear basis (Eq.7.2) is chosen to build the interpolation and therefore 4+1 information points are theoretically enough to create a continuous interpolation [11]. However, the number of elements in the first two layers can be as many as between 50 and 100, even for a

good quality tetrahedron mesh. As a result, the number of neighboring points can be far more than the necessary number (5) of information points, especially when higher order element which has more than one integration points are used. As explained in section 7.2 (Fig.7-2(b)), if all of the neighboring points are regarded as information points, the severe numerical diffusion can be induced. Therefore, a selection from these neighboring points should be performed to reduce the number of information points to an appropriate.

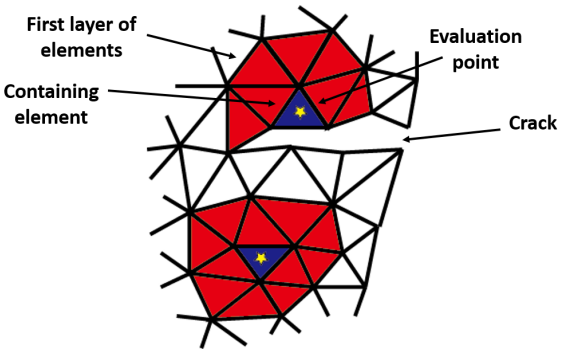


Figure 7-3: Visibility criterion

7.4 Isotropic information point selection strategy

As explained in section 7.2, both distance and relative direction between neighboring points and the evaluation point should be considered quantitatively at the same time (Fig.7-2(c)) and therefore a compromise should be made between these two aspects. We propose first to construct a distance weight and a direction weight for each neighboring point assessing these two aspects respectively. Then a compromise between them is made by a selection weight. The information points are finally selected according to the selection weight. In this work, the weight w_i in Eq. 7.4 is used as the distance weight. Emphasis is given on the construction of direction weight which is the crux of the selection strategy. We first only consider direction and the goal is to select a set of information points which are located in every radial direction starting from the evaluation point. Several radial direction vectors are chosen as referenced directions which are emitted from the evaluation point and point evenly to its

vicinity. If information points can be selected along every referenced direction, then we may agree that the information points are located in every radial direction. We define such set of information points as isotropically distributed information points and call the process as isotropic information point selection.

In practice, the neighboring points are not exactly along the referenced directions but located near them. The idea is to locally select the nearest neighboring point with respect to every referenced directions then the information points can be isotropically distributed in the vicinity of the evaluation point. As a consequence, the direction weight is constructed locally for each referenced direction independently in its own sub-region. The sub-region of a referenced direction is used to categorize a group of neighboring points which are considered around this referenced direction. In addition, such a local direction weight should be monotonically decreasing from referenced direction to the boundary its sub-region. To simplify the process, we can first project all the neighboring points onto a unit sphere centered at the evaluation point. Then a referenced direction becomes a referenced point on the sphere surface and its sub-region becomes a referenced patch on the sphere surface centered at a referenced point. As a consequence, the process to choose a set of isotropically distributed referenced directions and define their sub-regions becomes the partition of a sphere surface into isotropically distributed sub-regions. In addition, in each sub-region, at most one neighboring point (the nearest to the referenced point) is selected as information point, the number of information points is usually linked to the number of sub-regions. Based on these facts, we can qualitatively list 4 requirements for the sphere partition as following:

1) If the sphere is partitioned into N_{tot} sub-regions, the possible value of N_{tot} should be as many as possible

2) The area of each sub-region should be as identical as possible (equal area)

3) The shape of each sub-region should be as identically compact as possible (equal shape).

4) The distances between a sub-region and its neighboring sub-regions should be as identical as possible (locally dispersive distribution).

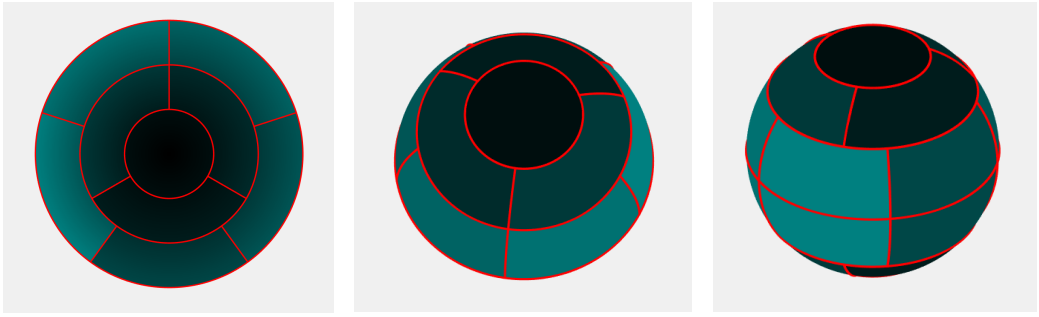
In this section, the method to partition the sphere is first discussed. The construction of direction weight is then explained. Finally, the selection of the information points and the related problems are presented.

7.4.1 Partition of a unit sphere surface

Different criteria are used to carry out the partitioning. Sometimes several criteria are given such as arbitrary number of partitions, equal area partitions or square-like partitions. However, usually these criteria cannot be satisfied completely at the same time and most authors give priority order to them. Leopardi [53] proposed a recursive zonal equal area sphere partitioning algorithm which guarantees that the areas of sub-regions are equal and makes the diameters of sub-regions as small as possible. This algorithm can be extended to partition higher dimensional sphere. Beckers et al. [3] propose a general rule to cut a disk into an imposed number of equal-area cells then using a mapping process to cut a hemisphere into similar sub-regions as the disk. In this method, the aspect ratio is enforced in order to make the sub-regions close to the square. Most of sphere partition techniques cannot impose an arbitrary number of sub-regions before partitioning. However, the technique should give as many possibilities of partitions as possible. Based on Beckers' work, we propose a method to partition a disk into several sub-regions and project this partition patterns onto sphere partition with the feasible numbers of sub-regions as many as possible. In section 7.4.1.1, 2D disk partition is discussed. In section 7.4.1.2, a projection technique from 2D disk partition to 3D sphere partition is proposed. In section 7.4.2, the construction of a selection weight is given. The selection of information points by a selection weight is explained finally in section 7.4.3.

7.4.1.1 Partition of a unit disk

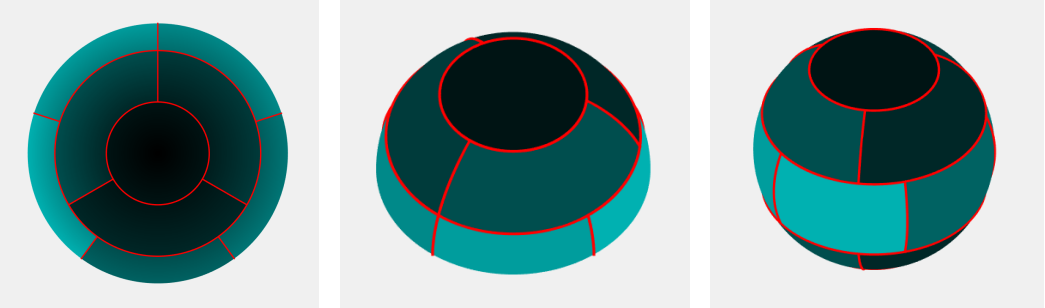
In order to project a 2D disk partition onto isotropic 3D sphere partition, the 2D disk partition should also be isotropic. We have 4 requirements for isotropic partition: the possible number N_{tot} is as many as possible, equal area, equal square-like shape and dispersive location. Since these requirements cannot be satisfied completely at the same time, a priority is given to the equal area. Following the basic partition patterns proposed by Beckers, a disk is first partitioned into several layers which have the same width by circles and each layer is then partitioned into several sub-regions which have the same area by radial straight segments. Examples of partition of a disk are shown in Fig.7-4(a). The advantages of this method is the convenience to control the area and shape of each sub-region and the full use of the disk boundary. This method also provides a transformation way when the partition on a disk is projected onto hemisphere, for instance, on a hemisphere the separation curves can be meridians and small circles parallel to the bottom circle of the hemisphere, as shown in Fig.7-4(b) and 7-5(b).



(a) Disk partition configuration (b) Hemisphere partition configuration (c) Sphere partition configuration

Figure 7-4: Basic configuration of partition when the total number of layers on a sphere is even

We propose a new sphere partition method here. In this method, disk partition is first done and the hemisphere partition is projected from the disk partition. The whole sphere partition is obtained by making a mirror symmetry of the hemisphere partition with respect to its equator plane. Given that the possible numbers of total sub-regions should be as many as possible and the identicalness of the area of all the



(a) Disk partition configuration (b) Hemisphere partition configuration (c) Sphere partition configuration

Figure 7-5: Basic configuration of partition when the total number of layers on a sphere is odd

sub-regions are given the most priority, we propose two kinds of configurations of the disk partition: first configuration is that the area of all the sub-regions on a disk is the same as illustrated in Fig.7-4(a); second configuration is that the areas of all the sub-regions along the boundary of the disk are only half of the others, as illustrated in Fig.7-5(a). We can see that the sphere partition derived from the first configuration of disk partition has even number of layers as in Fig.7-4(c). The sphere partition derived from the second configuration of disk partition has odd number of layers as in Fig.7-5(c). The total number of layers on the sphere is a characteristic parameter of the partition, denoted as K. The number of layers on a disk is denoted as T, which is a function of K and can be expressed in Eq. 7.6. The partitions corresponding to these two configurations are discussed in detail below.

$$T = \begin{cases} K/2, & \text{if } K \text{ is even} \\ (K-1)/2, & \text{if } K \text{ is odd} \end{cases} \tag{7.6}$$

When K is even and equals to 6, as in Fig.7-4(c), T equals to 3. The number of sub-regions in t layer is denoted as $N_t|_1^T$ and t increases from the center to the boundary on the disk. To be mentioned, the span lengths of all T layers are the same [3] and denoted as Δr in Fig.7-6. The equal area requirement can be expressed as $A_1 = A_t$.

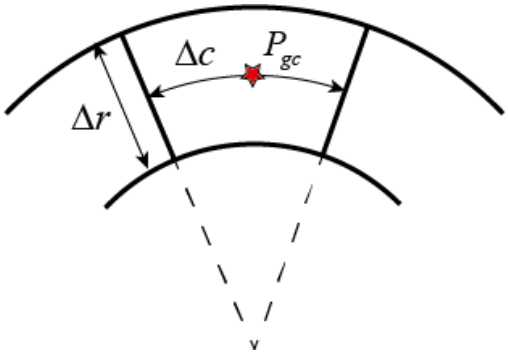


Figure 7-6: Variables in a sub-region

$$A_1 = A_t \Rightarrow \frac{\pi \Delta r^2}{N_1} = \frac{\pi (t\Delta r)^2 - \pi[(t-1)\Delta r]^2}{N_t} \Rightarrow N_t = (2t-1)N_1 \qquad (7.7)$$

According to this requirement, we can develop that the number of sub-regions in t layer N_t is only determined by the number in the first layer N_1 as in Eq.7.7. The number in the first layer N_1 is then another characteristic parameter of the partition. We can see later that it determines the other partition requirements. When K is even, the equal area requirement can be satisfied completely. However, when K is odd, this requirement is discussed later.

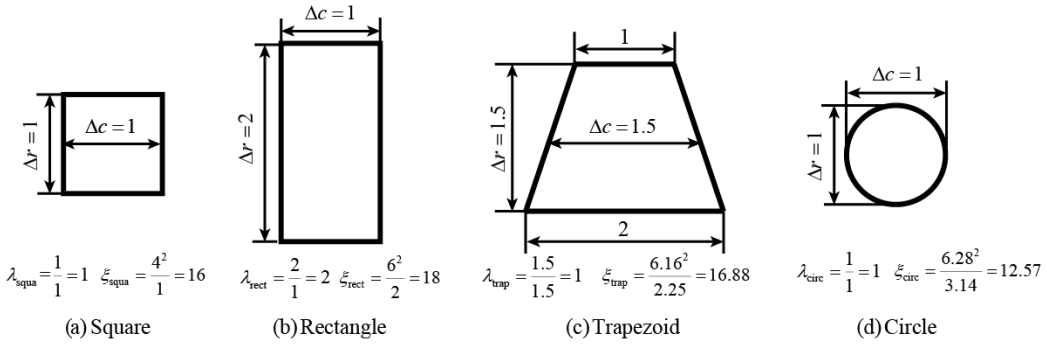


Figure 7-7: Aspect ratio and shape coefficient of different polygons

Before the square-like shape requirement is discussed, two coefficients to describe the 2D shape are introduced, namely aspect ratio and shape coefficient. Aspect ratio of a shape (sub-region) as in Fig.7-6 is defined as $\lambda = \Delta r / \Delta c$ in which Δr is span length and Δc is the chord of the shape which is an arc intersecting with the middle point of the span length. Shape coefficient is defined as $\xi = \rho^2 / A$ in which ρ is the

perimeter and A is the area. Different values of these two coefficients for different shapes are given in Fig.7-7. We can see from the Fig. 7-7(a) that the aspect ratio of a square λ_{squa} equals to 1 and its shape coefficient ξ_{squa} equals to 16. Therefore, if we define $\lambda_t|_1^T$ as the aspect ratio and $\xi_t|_1^T$ as the shape coefficient of sub-regions in the t layer, then the identically square-like shape requirement can be expressed as: aspect ratio $\lambda_t \approx \lambda_{squa} = 1$ and shape coefficient $\xi_t \approx \xi_{squa} = 16$. With respect to the even K basic partition configuration, the aspect ratio in t layer can be expressed in Eq.7.8.

$$\lambda_t = \frac{\Delta r}{2\pi(t-1/2)\Delta r/N_t} \tag{7.8}$$

Since the equal area is given the priority, it should be satisfied first so that we cannot directly set $\lambda_t = \lambda_{squa}$. Given the Eq.7.7 which is derived from equal area requirement, substitute Eq. 7.7 into Eq.7.8 we can obtain Eq.7.9.

$$\lambda_t = \frac{\Delta r(2t-1)N_1}{2\pi(t-1/2)\Delta r} = \frac{N_1}{\pi} \tag{7.9}$$

Under the basic partition configuration and equal area requirement, the aspect ratio in t layer λ_t is only determined by the number in the first layer N_1 as in Eq.7.9. Since $\pi \approx 3.14$ and $N_1 \in \mathbb{N}^+$, we can imagine that when $N_1 = 3$, the value of aspect ratio $\lambda_t = 0.955$ is the closest to 1 and when N_1 decreases or increases, λ_t goes away from 1. If square-like shape is only measured by the aspect ratio, we can conclude that the number of sub-regions in the first layer is close to 3, the shape of the sub-regions are more square-like. However, aspect ratio close to 1 is not enough to describe the square-like shape. As illustrated in Fig. 7-7(c), even if the aspect ratio of the trapezoid equals exactly to 1 but it is not a square, because the shape coefficient is not equals to 16. The shape coefficient in t layer denoted as $\rho_t|_1^T$ can be expressed as in Eq. 7.10.

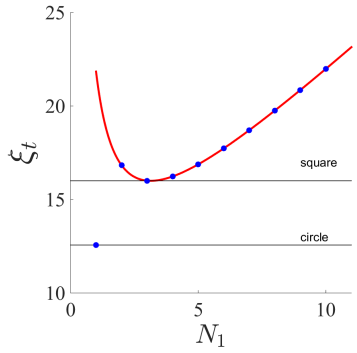


Figure 7-8: ξ_t is determined only by N_1

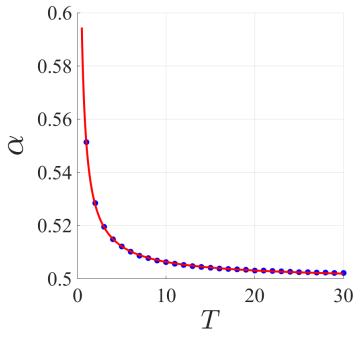


Figure 7-9: α is determined only by T

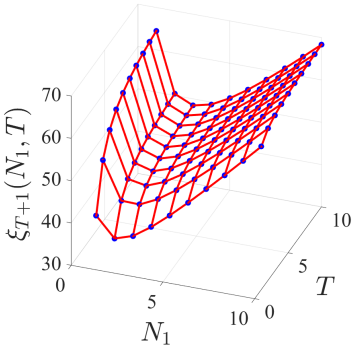


Figure 7-10: ξ_{T+1} is determined by both N_1 and T

$$\xi_t = \frac{\rho_t^2}{A_t} = \left(2\Delta r + \frac{2\pi t\Delta r}{N_t} + \frac{2\pi(t-1)\Delta r}{N_t}\right)\frac{N_t}{\pi(t\Delta r)^2 - \pi[(t-1)\Delta r]^2} = \frac{4N_1}{\pi}(1+\frac{\pi}{N_1})^2 \tag{7.10}$$

According to Eq.7.10, shape coefficient in t layer ξ_t is also only determined by the number in the first layer N_1. The Eq.7.10 is plotted in Fig.7-8. We can see from the Fig.7-8, when $N_1 = 3$, the shape coefficient ξ_t is nearly equal to $\xi_{squa} = 16$. When N_1 decreases or increases, the value of ξ_t goes away from 16. However, there is an exception when $N_1 = 1$ there is only one sub-region in the first layer and this sub-region becomes a circle. As illustrated in Fig.7-7(d), the shape coefficient of a circle is expressed as: $\xi_1(N_1 = 1) = \frac{(2\pi\Delta r)^2}{\pi\Delta r^2} = 12.57 \neq \frac{4(1+\pi)^2}{\pi} = 21.84$. There is also an exception of aspect ratio when there is only one sub-region in the first layer: $\lambda_1(N_1 = 1) = \frac{2\Delta r}{2\Delta r} = 1 \neq \frac{1}{\pi} = 0.3185$. Even if the shape of this sub-region is no longer identical square-like ($\xi_1(N_1 = 1) = 12.57 \ll 16$) as the other sub-regions, it is still compact ($\lambda_1(N_1 = 1) = 1 = \lambda_{squa}$) so that the partition patterns in which $N_1 = 1$ are still possible to be considered.

When K is odd, for example as in Fig.7-5(c) $K = 5$, there are $T = (5-1)/2 = 2$ layers with equal span length Δr and one layer with shorter span length $\alpha\Delta r (0 < \alpha < 1)$ within the disk. The coefficient α is termed the span length coefficient. In this configuration, the sub-regions in all T layers have the same features as when K

is even. The only difference appears in the $T+1$ layer. Therefore, we focus only on the $T+1$ layer and discuss all the partition requirements the same as K is even. The area of the sub-regions in $T+1$ layer A_{T+1} can be computed as in Eq. 7.11

$$A_{T+1} = \frac{2\pi[(T+\alpha)\Delta r]^2 - 2\pi(T\Delta r)^2}{N_{T+1}} = \frac{2\pi\alpha\Delta r^2(2T+\alpha)}{N_{T+1}} \tag{7.11}$$

The equal area requirement $A_1 = A_{T+1}$ is then expressed as in Eq.7.12:

$$A_1 = A_{T+1} \Rightarrow \frac{\pi\Delta r^2}{N_1} = \frac{2\pi\alpha\Delta r^2(2T+\alpha)}{N_{T+1}} \Rightarrow N_{T+1} = 2\alpha N_1(2T+\alpha) \tag{7.12}$$

In contrast to the layers 1 through T, where N_t depends only on N_1, in layer $T+1$ the number of sub-regions N_{T+1} depends on N_1, α, and T. The aspect ratio of the sub-regions in the $T+1$ layer λ_{T+1} can be computed as in Eq.7.13:

$$\lambda_{T+1} = \frac{2\alpha\Delta r}{\Delta c_{T+1}} = \frac{2\alpha\Delta r}{2\pi(T+\alpha)\Delta r/N_{T+1}} = \frac{2\alpha^2(2T+\alpha)}{\pi(T+\alpha)} \tag{7.13}$$

In layers 1 through T, the equal aspect ratio $\lambda_t = \lambda_1$ is directly satisfied when there is an equal area $A_t = A_1$. In layer $T+1$, if the equal aspect ratio is considered, we can obtain Eq.7.14.

$$\lambda_1 = \lambda_{T+1} \Rightarrow \frac{N_1}{\pi} = \frac{2\alpha^2(2T+\alpha)N_1}{\pi(T+\alpha)} \Rightarrow 2\alpha^3 + 4T\alpha^2 - \alpha - T = 0 \tag{7.14}$$

Coefficient α depends only on the number of layers T, so we can define α as $\alpha(T)$. When T increases, α decreases and converges to 0.5. As shown in Fig.7-9, the blue points represent the implicit solution of $\alpha(T)$, and the red line is the explicit fitting solution of $\alpha(T) = 0.0524T^{-0.8638} + 0.4991$. The number of layers T can be determined by the characteristic parameter K as $T(K)$. Therefore, the span length coefficient depends only on K as $\alpha(K)$.

Then the perimeter ρ_{T+1} can be calculated as in Eq.7.15:

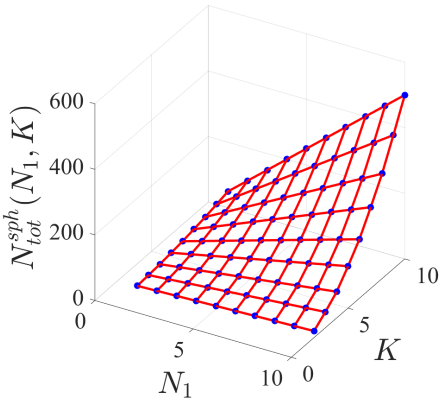


Figure 7-11: N_{tot}^{sph} is determined by both N_1 and K

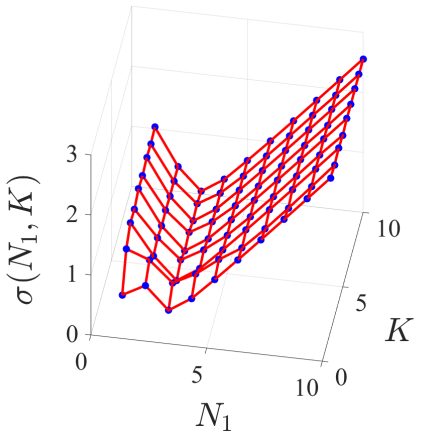


Figure 7-12: σ is determined by both N_1 and K

$$\rho_{T+1} = 4\alpha\Delta r + \frac{4T\pi\Delta r}{N_{T+1}} \tag{7.15}$$

Substitute Eq. 7.12 and Eq. 7.14 into Eq. 7.15, we can obtain Eq. 7.16:

$$\rho_{T+1} = 4\Delta r[\alpha + \frac{\pi T}{2\alpha N_1(2T+\alpha)}] = 4\alpha\Delta r[1 + \frac{\pi T}{N_1(\alpha+T)}] \tag{7.16}$$

Given Eq. 7.12 in which $A_{T+1} = A_1 = (\pi\Delta r^2)/N_1$, the shape coefficient ξ_{T+1} can be expressed as in Eq.7.17:

$$\xi_{T+1} = \frac{\rho_{T+1}^2}{A_{T+1}} = 16\alpha^2\Delta r^2[1 + \frac{\pi T}{N_1(\alpha+T)}]^2\frac{N_1}{\pi\Delta r^2} = \frac{16N_1\alpha^2}{\pi}[1 + \frac{\pi T}{(\alpha+T)N_1}]^2 \tag{7.17}$$

The shape coefficient ξ_{T+1} depends on the number of layers T and number of sub-regions in the first layer N_1. The variation of ξ_{T+1} with respect to N_1 and T is plotted in Fig.7-11. Above all, the characteristic parameters N_1 and K determine the shape of all the sub-regions as well as the number of sub-regions in all the layers, which in fact determines the total number of sub-regions. We only discuss the total number of the sub-regions on a sphere denoted $N_{tot}^{sph}(N_1, K)$. In the odd K configuration, the number of sub-regions in the $T+1$ layer is not an integer because, as expressed in Eq.7.12,

the span length coefficient is not an integer. In practice, we round N_{T+1} up and down to integers $\lceil N_{T+1} \rceil$ and $\lfloor N_{T+1} \rfloor$ which satisfy the relation $\lceil N_{T+1} \rceil - \lfloor N_{T+1} \rfloor = 1$. As a consequence, in the odd K configuration, one pair (N_1, K) corresponds to two N_{tot}^{sph}, as in Eq.7.18. All the possible values of N_{tot}^{sph} with $1 \leq N_1, K \leq 10$ are plotted as blue points in Fig.7-11, in which no rounding is done for the odd K configuration. Even if not all the positive integers are feasible, our partition method includes as many positive integers as possible for N_{tot}^{sph}.

$$N_{tot}^{sph}(N_1, K) = \sum_{k=1}^{K} N_k = \begin{cases} 2\sum_{t=1}^{T} N_t, & \text{if } K \text{ is even} \\ 2\sum_{t=1}^{T} N_t + \lceil N_{T+1} \rceil & \\ \text{or} & \text{if } K \text{ is odd} \\ 2\sum_{t=1}^{T} N_t + \lfloor N_{T+1} \rfloor, & \end{cases} \tag{7.18}$$

We can see from the figure that some of the N_{tot}^{sph} may correspond to more than one characteristic parameter pair (N_1, K). For example, $N_{tot}^{sph} = 12$ characteristic parameter pairs can be either $(N_1 = 1, K = 5)$ or $(N_1 = 6, K = 2)$. Our goal is to select a *better* characteristic parameter pair (N_1, K) for each possible corresponding total number of sub-regions N_{tot}^{sph}. Because the area of each sub-region is nearly the same, *better* here means that the shape of each sub-region is more square-like (aspect ratio $\lambda_{squa} = 1$ and the shape coefficient $\xi_{squa} = 16$). One coefficient, namely shape deviation $\sigma(N_1, k)$, is defined as the sum of unified standard deviations of these two parameters λ and ξ, as in Eq.7.19, in order to measure the deviations of all the sub-regions from squares. The pair (N_1, K) corresponding to the same N_{tot}^{sph} which minimizes the shape deviations among all the possible (N_1, K) pairs is finally used.

$$\sigma(N_1, K) = \sqrt{\frac{1}{N_{tot}} \sum_{i=1}^{N_{tot}} \left(\frac{\lambda_i - \lambda_{squa}}{\lambda_{squa}} \right)^2} + \sqrt{\frac{1}{N_{tot}} \sum_{i=1}^{N_{tot}} \left(\frac{\xi_i - \xi_{squa}}{\xi_{squa}} \right)^2} \tag{7.19}$$

In the example of $N_{tot}^{sph} = 12$, based on the definition of shape deviation, we have $\sigma(1, 5) = 0.68 < 1.02 = \sigma(6, 2)$ which determines the better solution for $N_{tot}^{sph} = 12$

is the pair $(N_1 = 1, K = 5)$. The shape deviation $\sigma(N_1, K)$ with $1 \leq N_1, K \leq 10$ is then plotted in Fig. 7-12. We can see from the figure that for the same number of total layers K, the better solution is usually at $N_1 = 3$. In addition, for the same number of sub-regions as in the first layer N_1, K does not greatly influence the decision, other than where $N_1 = 1$. As we explained above, when $N_1 = 1$, the only sub-region becomes a circle, which substantially modifies the aspect ratio and the shape coefficient.

Up to now, a partition pattern N_{tot}^{sph} has corresponded to only one parameter pair (N_1, K) that determines the number of layers and the number of sub-regions in each layer. However, the relative positions of the sub-regions between adjacent layers are still flexible; we may imagine that the sub-regions in a layer are located along a circle and that they may be moved along the circle. In order to satisfy the fourth partitioning requirement, namely that the distribution of every sub-region should be locally dispersive, these relative positions need to be carefully determined. The location of a sub-region can be indicated by its geometric center, which is defined as the middle point of its chord Δc, as illustrated in Fig.7-6. Since the relative positions between sub-regions in the same layer are fixed (when the number of sub-regions in a layer is determined), only the relative positions between sub-regions in adjacent layers can be optimized. Relative positions between pairs of adjacent layers are determined in two steps as follows.

In the first step we consider on two arbitrary adjacent layers t^{int} and t^{ext} ($t^{int} < t^{ext}$). As illustrated in Fig.7-13, we denote layer t^{int} as the interior layer and its sub-regions as interior sub-regions whose geometric centers are shown as red dots. Layer t^{ext} is the exterior layer, and its sub-regions are exterior sub-regions whose geometric centers are shown as blue dots. One of the straight boundaries of a sub-region within a layer is regarded as its reference. The relative position can be defined as an angle denoted as φ between two references of the two adjacent layers. This angle φ is optimized, in order to satisfy the locally dispersive distribution requirement for all of the sub-regions in the exterior layer. In the next step, beginning from the first layer to the T layer (even K) or the $T + 1$ layer (odd K), the angles $\varphi_t (t > 1)$ between

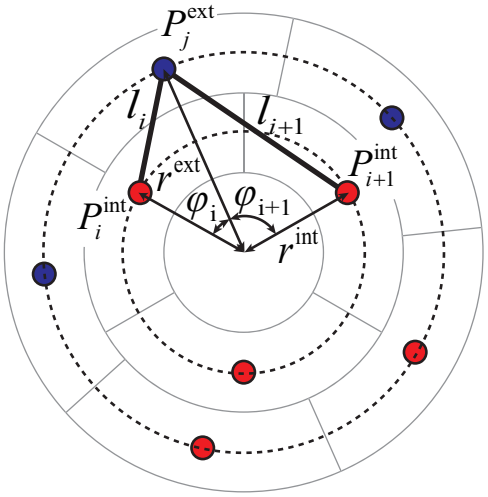


Figure 7-13: The locally dispersive location is determinted by an angle φ between exterior and interior layers

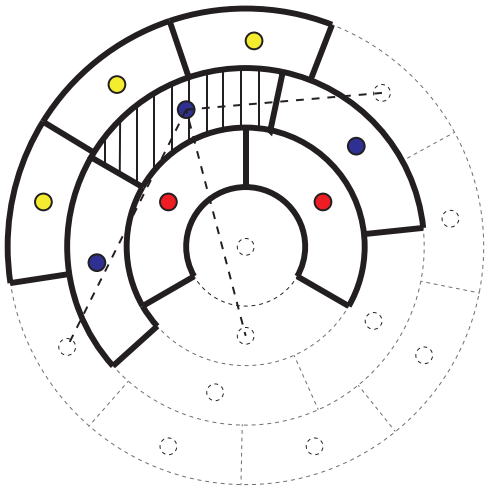


Figure 7-14: Neighboring sub-regions of a dashed sub-region

each pair of adjacent layers are optimized successively in order to satisfy the fourth requirement for sub-regions in all layers.

Let us focus on this first step and suppose there are two adjacent layers as illustrated in Fig.7-13. The number of sub-regions in the interior layer is denoted as N^{int} and the number of sub-regions in the exterior layer is denoted as N^{ext}. First, we concentrate on one exterior sub-region with geometric center P_j^{ext}, together with its neighboring sub-regions. As illustrated in Fig.7-14, the target sub-region might for example be the sub-region indicated by dashed lines and with a blue dot as its geometric center. In this case the neighboring sub-regions are the two sub-regions in the same layer, two (red geometric center) in the inner layer, and three (yellow geometric center) in the outer layer. The locally dispersive distribution for this sub-region means that the distances between its geometric center and the geometric centers of neighboring sub-regions should be as equal as possible. As described above, we only consider two layers (red and blue) and the distances between geometric centers in the same layer are constant. Therefore, only the distance between the geometric center of target sub-region and the geometric center of the neighboring sub-regions in the interior layer are considered. In practice, if we draw a segment between the geometric center of j exterior sub-region P_j^{ext} and the center of the disk denoted as P^{ctr}, we

will always find that this segment $\overline{P^{ctr}P_j^{ext}}$ intersects with an arc $\widehat{P_i^{int}P_{i+1}^{int}}$ whose two extremes are two geometric centers of the two interior adjacent sub-regions i and $i+1$, namely P_i^{int} and P_{i+1}^{int}. These two sub-regions are then defined as the neighboring sub-regions and the corresponding distances are denoted as $l_{i,j}$ and $l_{i+1,j}$. Above all, the locally dispersive distribution requirement for this target sub-region can be expressed as the minimization of the objective function $J(\varphi) = l_{i,j}^2 + l_{i+1,j}^2$. In terms of all the exterior sub-regions, the same objective function can be built as Eq.7.20. The relative positions between the two layers determine the angle φ^* that minimizes Eq.7.20.

$$J(\varphi) = \sum_{j=1}^{N^{tot}} (l_{i,j}^2 + l_{i+1,j}^2) \tag{7.20}$$

The length of the segment $\overline{P^{ctr}P_j^{ext}}$ is denoted as r^{ext}, and the segment $\overline{P^{ctr}P_i^{int}}$ ($\overline{P^{ctr}P_{i+1}^{int}}$) is denoted as r^{int} in Fig.7-13. The angle between $\overline{P^{ctr}P_i^{int}}$ and $\overline{P^{ctr}P_j^{ext}}$ is denoted as $\varphi_{i,j}$ and the angle between $\overline{P^{ctr}P_{i+1}^{int}}$ and $\overline{P^{ctr}P_j^{ext}}$ is denoted as $\varphi_{i+1,j}$. The two distances $l_{i,j}$ and $l_{i+1,j}$ are then expressed as Eq. 7.21, where r^{int} and r^{ext} are constant when t^{int} and t^{ext} are determined.

$$\begin{cases} l_{i,j}^2 = (r^{int})^2 + (r^{ext})^2 - 2r^{int}r^{ext}cos(\varphi_{i,j}) \\ l_{i+1,j}^2 = (r^{int})^2 + (r^{ext})^2 - 2r^{int}r^{ext}cos(\varphi_{i+1,j}) \end{cases} \tag{7.21}$$

As illustrated in Fig. 7-15, the angle between the point P_j^{ext} and the reference of exterior layer is denoted as $\varphi_{ref,j}^{ext}$. The angle between the point P_{i+1}^{int} and the reference of interior layer is denoted as $\varphi_{i+1,ref}^{int}$. These two angles are expressed as Eq.7.22 in which the variable $N_{i+,ref}^{int}$ is the number of interior sub-regions within the angle $\varphi_{i+1,ref}^{int}$ and the variable $N_{ref,j}^{ext}$ is the number of exterior sub-regions within the angle $\varphi_{ref,j}^{ext}$.

$$\begin{cases} \varphi_{i+1,ref}^{int} = (2N_{i+1,ref}^{int} + 1)\pi/N^{int} \\ \varphi_{ref,j}^{ext} = (2N_{ref,j}^{ext} + 1)\pi/N^{ext} \end{cases} \tag{7.22}$$

Given that the sub-regions i and $i+1$ are adjacent, the two angles $\varphi_{i,j}$ and $\varphi_{i+1,j}$

can be expressed as Eq.7.23.

$$\begin{cases} \varphi_{i,j} = 2\pi/N^{int} - \varphi_{i+1,j} \\ \varphi_{i+1,j} = \varphi^{int}_{i+1,ref} + \varphi^{ext}_{ref,j} - \varphi \end{cases} \tag{7.23}$$

Substituting Eq.7.22 into Eq.7.23 we obtain Eq.7.24:

$$\begin{cases} \varphi_{i,j} = (1 - 2N^{int}_{i+1,ref})\pi/N^{int} - (2N^{ext}_{ref,j} + 1)/N^{ext} + \varphi = \varepsilon_{i,j} + \varphi \\ \varphi_{i+1,j} = (2N^{int}_{i+1,ref} + 1)\pi/N^{int} + (2N^{ext}_{ref,j} + 1)/N^{ext} - \varphi = \varepsilon_{i+1,j} - \varphi \end{cases} \tag{7.24}$$

in which the two variables $\varepsilon_{i,j}(N^{int}, N^{ext})$ and $\varepsilon_{i+1,j}(N^{int}, N^{ext})$ correspond simply to the numbers of sub-regions in the interior layer and the exterior layer. If we substitute Eq.7.21 and Eq.7.24 into Eq.7.20, we find that the objective function also depends only on the two numbers N^{int} and N^{ext} as in Eq.7.25:

$$\begin{aligned} J(\varphi) &= \sum_{j=1}^{N^{tot}} (l^2_{i,j} + l^2_{i+1,j}) \\ &= \sum_{j=1}^{N^{ext}} \{2(r^{int})^2 + 2(r^{ext})^2 - 2r^{int}r^{ext}[\cos(\varepsilon_{i,j} + \varphi) + \cos(\varepsilon_{i+1,j} - \varphi)]\} \\ &= \sum_{j=1}^{N^{ext}} \{2(r^{int})^2 + 2(r^{ext})^2 - 2r^{int}r^{ext}[(\cos\varepsilon_{i,j} + \cos\varepsilon_{i+1,j})\cos\varphi - (\sin\varepsilon_{i,j} - \sin\varepsilon_{i+1,j})\sin\varphi]\} \\ &= \sum_{j=1}^{N^{ext}} [2(r^{int})^2 + 2(r^{ext})^2 - 2r^{int}r^{ext}(\psi^{cos}_j \cos\varphi - \psi^{sin}_j \sin\varphi)] \end{aligned} \tag{7.25}$$

where ψ^{cos}_j is defined as $\psi^{cos}_j = \cos\varepsilon_{i,j} + \cos\varepsilon_{i+1,j}$ and ψ^{sin}_j is defined as $\psi^{sin}_j = \sin\varepsilon_{i,j} - \sin\varepsilon_{i+1,j}$. With respect to the definition of the angle φ^* which gives a locally dispersive distribution between these two layers and where N^{ext} is always bigger than N^{int}, we know that the value of φ^* is limited in the interval $[0, 2\pi/N^{ext}]$. The derivation of the objective function is expressed in Eq.7.26.

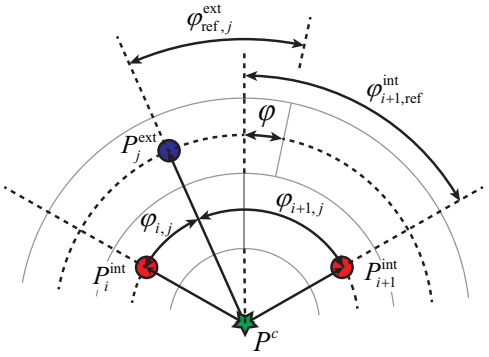


Figure 7-15: Illustration of different angles related to determination of locally dispersive location φ

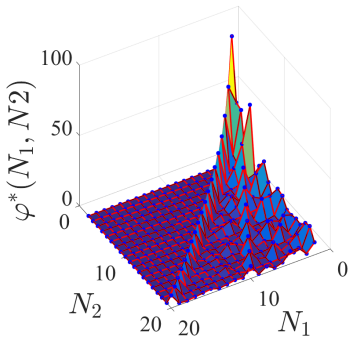


Figure 7-16: φ^* is determined by both N^{int} and N^{ext}

$$\frac{dJ(\varphi)}{d\varphi} = 2r^{int}r^{ext}[(\sum_{j=1}^{N^{ext}} \psi_j^{cos})\sin\varphi + (\sum_{j=1}^{N^{ext}} \psi_j^{sin})\cos\varphi] \tag{7.26}$$

In order to minimize the objective function, we set $\frac{dJ(\varphi^*)}{d\varphi} = 0$. The angle φ^* is finally determined by solving this equation as in Eq.7.27.

$$\varphi^*(N^{int}, N^{ext}) = -\arctan(\frac{\sum_{j=1}^{N^{ext}} \psi_j^{sin}}{\sum_{j=1}^{N^{ext}} \psi_j^{cos}}) \tag{7.27}$$

With respect to Eq.7.27, φ^* is only depend on the number of sub-regions in interior layer N^{int} and the number of sub-regions in exterior layer N^{ext}, as illustrated in Fig. 7-16. The relative position between $t-1$ and t layers φ_t is the cumulative value of φ^* within t layers as expressed in Eq. 7.28.

$$\varphi_t = \begin{cases} 0, & \text{if } t = 1 \\ \sum_{i=2}^{t} \varphi^*(N_{t-1}, N_t), & \text{if } t \geq 2 \end{cases} \tag{7.28}$$

Above all, a characteristic parameter pair (N_1, K) corresponds a unique configuration of disk partition and the relative positions of all the sub-regions are fixed. All the possible solutions of disk partition with $6 \leq K \leq 20$ are displayed in Fig.7-17. We can see that there are no solution for $N_{tot} = 7$, $N_{tot} = 11$ and $N_{tot} = 17$, because with these values no configuration satisfying the 4 requirements can be found.

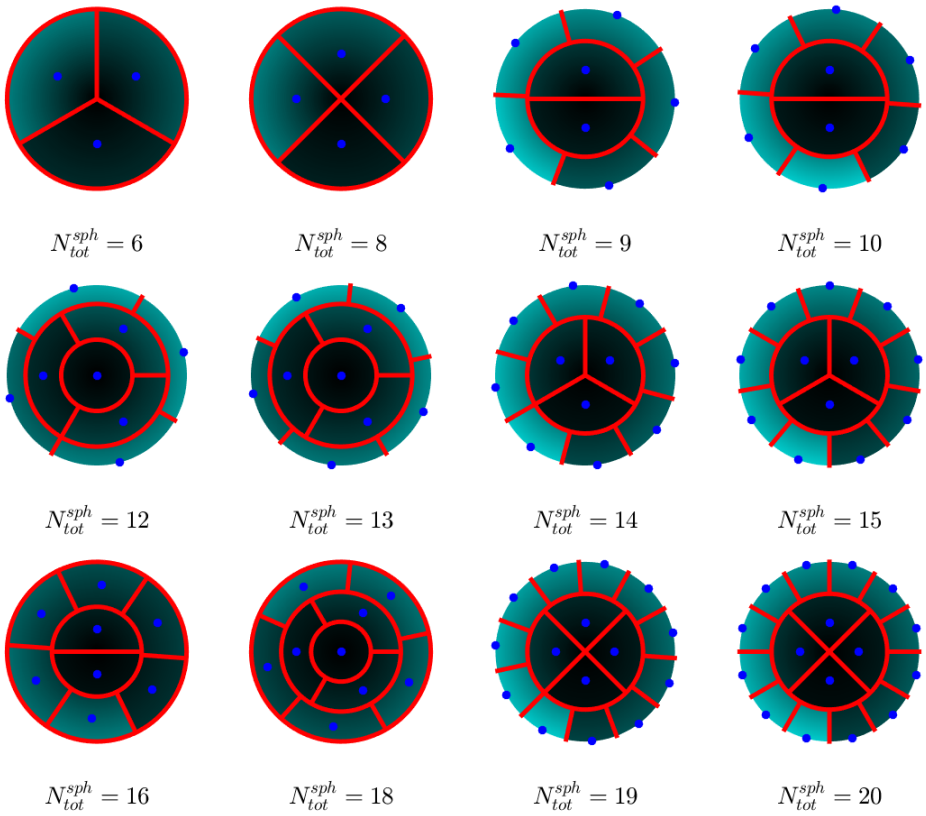


Figure 7-17: All possible disk partition patterns with $N_{tot}^{sph} \leq 20$

7.4.1.2 Getting sphere partition from disk partition

A 3D sphere partition is obtained by making a hemisphere partition symmetry with respect to its equator plane. A hemisphere partition can be projected from a disk partition. To be mentioned, the equal area requirement has also the priority in sphere partition. Therefore, the projection process should keep the area ratio identical between each sub-region on the disk and its projection on the sphere. In order to get equal ratio requirement in sphere partition, Lambert azimuthal equal-area projection [86] is used. In practice, a disk can be partitioned in polar coordinate system $(\varphi^{disk}, r^{disk})$ and a sphere partition can be expressed in spherical coordinate system (φ, θ, r). The Lambert azimuthal equal-area projection is performed as in Eq.7.29 in which is the radius of the partitioned disk.

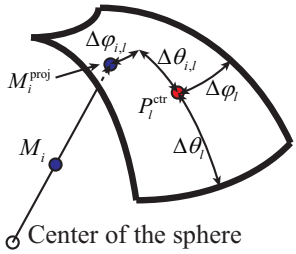


Figure 7-18: Local angles within a sub-region

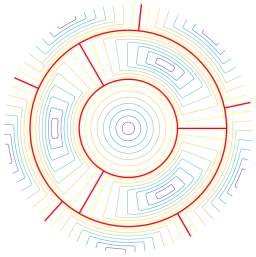


Figure 7-19: The contour lines of uniformed angle $\beta_{i,j}^{unif}$

$$\begin{cases} \varphi = \varphi^{disk} \\ \theta_1 = 2\arcsin(\dfrac{\sqrt{2}r^{disk}}{2R^{disk}}) \\ r = 1 \end{cases} \tag{7.29}$$

This equation in which $0 \leq \varphi \leq 2\pi$ and $0 \leq \theta_1 \leq \pi/2$ projects a disk partition onto a hemisphere. The corresponding whole sphere partition is then expressed by the symmetry process as: $\theta_2 = \pi - 2\arcsin(\frac{\sqrt{2}r^{disk}}{2R^{disk}})$ in which $\pi/2 \leq \theta_2 \leq \pi$.

7.4.2 Local direction weight construction

To be mentioned, the information point selection strategy is aimed to select a set of information points which are isotropically distributed around the evaluation point. As discussed in section 7.4, this isotropic distribution of the information point set is based on the isotropic distribution of sub-regions, because in each sub-region, an information point is selected. Therefore, the ideal information point to be selected in each sub-region is the neighboring point whose radial projection point on the sphere is located just on the geometry center of this sub-region. If this is happened in all sub-regions, the information points are distributed isotropically as the sub-regions. However, the projection of a neighboring point is usually not located on the geometry center of the sub-regions. Therefore, a local direction weight should be constructed in order to measure the selection priority of a neighboring point with respect to the distance between its projection and geometry center. On a sphere surface, it is better to use angular distance instead of length distance between two points.

A local angular distance here is defined in a local spherical coordinate system $(\Delta\varphi, \Delta\theta, r)$ and the lth sub-region on the unit sphere is illustrated in Fig.7-18, in order to denote the variables. The red point $P_l^{ctr}(0,0,1)$ is the geometry center of the sub-region and is also the reference point of the local spherical coordinates. The blue point $M_i^{proj}(\Delta\varphi_{i,j}, \Delta\theta_{i,j}, 1)$ is the projection of a neighboring point on the unit sphere with $0 \leq \Delta\varphi_{i,j} \leq \Delta\varphi_l$ and $0 \leq \Delta\theta_{i,l} \leq \Delta\theta_l$. The local angular distance $\beta_{i,j}^{loc}$ of the projection M_i^{proj} can be defined as a function of the two angular components of its coordinates, as expressed in Eq.7.30:

$$\beta_{i,l}^{loc} = h(\Delta\varphi_{i,l}, \Delta\theta_{i,l}) \quad \text{with} \quad 0 \leq \beta_{i,l}^{loc} \leq \Delta\beta_l^{loc} \tag{7.30}$$

With this definition, the local direction weight $w_{i,l}^{dir}$ can be defined as in Eq.7.31.

$$w_{i,l}^{dir}(M_i^{proj}, P_l^{ctr}) = \begin{cases} f(\beta_{i,l}^{loc}) & \text{if } 0 \leq \beta_{i,l}^{loc} \leq \Delta\beta_l^{loc} \\ 0, & \text{otherwise} \end{cases} \tag{7.31}$$

In our partition technique, all the sub-regions are regarded as the same importance. Therefore, the local direction weight should be the same along the boundary of different sub-regions. For example, when M_i^{proj} is located at the boundary of the l_1 and l_2 sub-regions, we have $\beta_{i,l_1}^{loc} = \Delta\beta_{l_1}^{loc}$ and $\beta_{i,l_2}^{loc} = \Delta\beta_{l_2}^{loc}$. The local direction weight of M_i^{proj} should be equal in the two sub-regions. This can be expressed as an equivalence criterion in Eq.7.32 so that no difference is generated for different sub-regions. This equivalence criterion is the same as the C^0 continuous criterion for the finite element shape function.

$$w_{i,l_1}^{dir} = f(\beta_{i,l_1}^{loc}) = f(\Delta\beta_{l_1}^{loc}) = f(\beta_{i,l_2}^{loc}) = f(\Delta\beta_{l_2}^{loc}) = w_{i,l_2}^{dir} \tag{7.32}$$

As the partition method explained in section 7.4.1.1, we can know that the shape of the sub-regions are similar but not totally the same which results in different local angular distance for the same point on the boundary of different sub-regions as $\Delta\beta_{l_1}^{loc} \neq \Delta\beta_{l_2}^{loc}$. The simplest way to guarantee the equivalence criterion is to unify the local angular distance $\beta_{i,l}^{loc}$ into the uniformed angular distance $\beta_{i,l}^{unif}$ in each sub-

region as in Eq.7.33, in order to get the equivalence as $\Delta\beta_{l_1}^{unif} = \Delta\beta_{l_2}^{unif} = \Delta\beta^{unif}$.

$$\beta_{i,l}^{unif} = g(\beta_{i,l}^{loc}) \quad \text{with } 0 \leq \beta_{i,l}^{unif} \leq \Delta\beta^{unif} \tag{7.33}$$

As a consequence, the equivalence criterion (Eq.7.32) can be satisfied as expressed in Eq.7.34.

$$w_{i,l_1}^{dir} = f(\beta_{i,l_1}^{unif}) = f(\Delta\beta^{unif}) = f(\beta_{i,l_2}^{unif}) = w_{i,l_2}^{dir} \tag{7.34}$$

If we substitute Eq.7.30 into Eq.7.33 and consider that there is no special requirements for function $g(\bullet)$ and $h(\bullet)$, we can know that the uniformed angular distance is directly the function of two local angular coordinates, as expressed in Eq.7.35.

$$\beta_{i,l}^{unif} = g(h(\Delta\varphi_{i,l}, \Delta\theta_{i,l})) = gh(\Delta\varphi_{i,l}, \Delta\theta_{i,l}) \tag{7.35}$$

The simplest way to build a function $gh(\bullet)$ in an interval $[0, \Delta\beta^{unif}]$ is a linear function as in Eq.7.36. This equation gives a continuous function for the uniformed angular distance in a sub-region. The contour lines of this function in a sub-region is illustrated in the Fig.7-19.

$$\beta_{i,l}^{unif} = \max\left(\frac{\Delta\beta_l^{unif}}{\Delta\beta_l^{loc}}\Delta\varphi_{i,l}, \frac{\Delta\beta_l^{unif}}{\Delta\beta_l^{loc}}\Delta\theta_{i,l}\right) \tag{7.36}$$

According to the meaning of direction weight, the maximum value should be given to the geometry center, since the most prior selection is the neighboring point whose projection is located in the geometry center. The minimum value should impose to the boundary, since the least prior one is located on the boundary. Therefore, the direction weight should monotonically decreases from geometry center ($\beta_{i,l}^{unif} = 0$) to the boundary ($\beta_{i,l}^{unif} = \Delta\beta^{unif}$). In a first study, we choose a cosine function of uniformed angular distance which is satisfied to the requirement as the direction weight function, as expressed in Eq.7.37. Since the weight should be positive so that the uniformed direction weight equals to $\pi/2$.

$$w_{i,l}^{dir}(M_i^{proj}, P_l^{ctr}) = \begin{cases} \cos(\beta_{i,l}^{unif}) & \text{if } 0 \leq \beta_{i,l}^{unif} \leq \pi/2 \\ 0, & \text{otherwise} \end{cases} \tag{7.37}$$

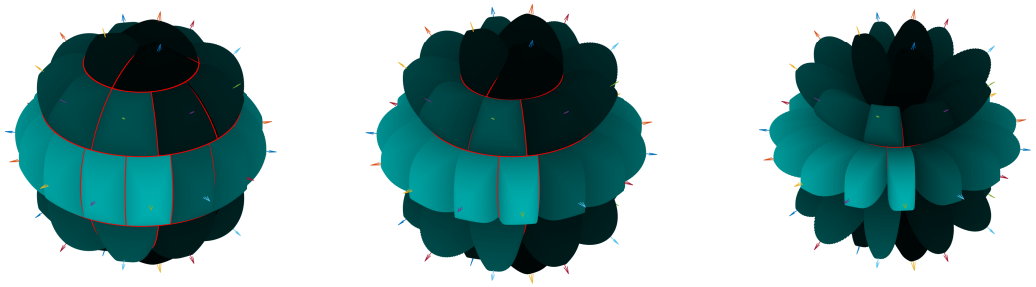
To be mentioned, the direction weight should be combined with distance weight in order to get a compromise as final selection weight. Therefore, we want to scale the direction weight and control the variation of direction weight. Based on this idea, we introduce two parameters to ameliorate the Eq.7.37, such as: shape parameter γ and scale parameter ν, as expressed in Eq. 7.38.

$$w_{i,l}^{dir}(M_i^{proj}, P_l^{ctr}) = \begin{cases} \nu[\cos\gamma(\cos\beta_{i,l}^{unif} - 1) + 1] \text{ if } & 0 \leq \beta_{i,l}^{unif} \leq \pi/2 \\ 0, & \text{otherwise} \end{cases} \tag{7.38}$$

As its name, shape parameter influence the shape of direction weight function. In order to illustrate the direction weight function, we plot partitioned sphere-like surfaces with the total number of sub-regions $N_{tot} = 38$ as in Fig.7-20. The radii of these sphere-like surfaces are the direction weight function with $\gamma = 0.45\pi$, $\gamma = 0.4\pi$ and $\gamma = 0.3\pi$. We can see from the Fig.7-20, when γ decreases, more priority is given near the geometry center which means the neighboring points there are much more likely to be selected than near the boundary. In fact, when scale parameter $\nu = 1$, the maximum of direction weight $\max(w_{i,l}^{dir}) = 1$ and the minimum $\min(w_{i,l}^{dir}) = \cos\gamma$. The other value of scale parameter is just used to scale the function linearly.

7.4.3 Selection of Information points from neighboring points through selection weight

Diffuse interpolation should be performed locally which means the information points should be as near to the evaluation point as possible. If a neighboring point is nearly located on the evaluation point, it should be prior selected as an information point no matter where its projection is on the sub-region. That is the reason why we should consider the interpolating distance weight as well as the local direction



(a) $\gamma = 0.45\pi$ (b) $\gamma = 0.4\pi$ (c) $\gamma = 0.3\pi$

Figure 7-20: The influence of coefficient γ on the shape of direction weight in all sub-regions

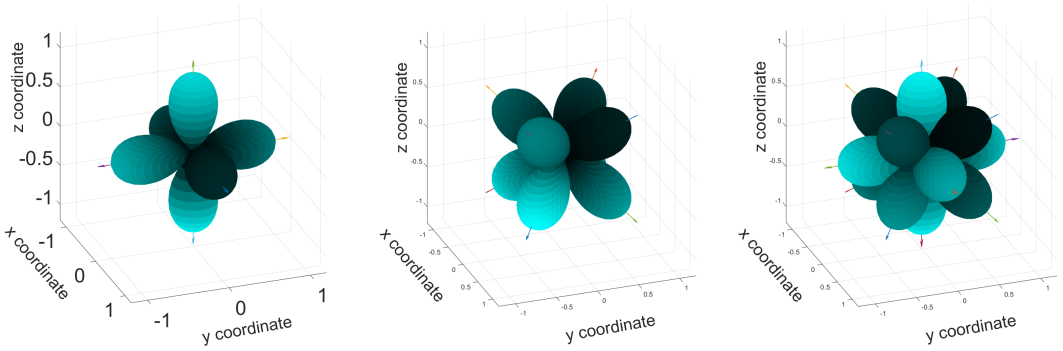
weight. The final selection weight $w_{i,l}$ is defined as a product of direction weight and distance weight in each sub-region respectively in Eq.7.39. The neighboring point which maximizes the selection weight $\max(w_{i,l})$ in each sub-region is then selected as an information point. In practice, the parameters in direction weight are tuned with respect to the distance weight in order to choose the most representative set of information points for Diffuse Interpolation calculation.

$$w_{i,l} = w_{i,l}^{dir} \cdot w_i^{\infty} \tag{7.39}$$

7.4.4 Another way to select isotropic information point

We present here another way to select an isotropic distribution of information points within the smallest vicinity of the evaluation point. This method is a previous work. Even if in this work, the number of sub-region is fixed at 14, it has an advantage that each sub region has more similar compact shape in the space. In this method, the determination of neighboring points can be the same as in section 7.3.

Our information point selection method is aimed to select an appropriate number of information points which are distributed isotropically in the space around the evaluation point. The contribution of any neighboring point to the information point selection is a compromise between its distance to the evaluation point and its radial direction to the evaluation point. Therefore, two types of weights are considered



(a) 6 sub-regions (b) 8 sub-regions (c) 14 sub-regions

Figure 7-21: Partitioning the space into sub-regions with respect to referenced direction vectors

corresponding to these two aspects, distance weight w^{∞} based on distance only and local direction weight w^{dir} based on radial direction only. According to this basic idea, we find another way which is more general. In this method, the vicinity space is also partitioned into several sub-region, however it is no longer based on partition of the sphere but based on a group of the direction vectors.

The partition process is carried out by choosing a number of referenced direction vectors which are isotropically distributed around the evaluation point. Fig. 7-21 shows three kind of space partition processes with different numbers of referenced direction vectors (sub-regions). As a major requirement, the sub-regions must cover the whole vicinity space, so that all neighboring points are taken into account. As we shall see, these sub-regions overlap and as a consequence, the local direction weight of this neighboring point must be computed in each sub-region. The sub-region to which a neighboring point is categorized maximizes the direction weight.

For the sake of clarity, the construction of local direction weights and the process of selecting a set of isotropically distributed information points are described with a 2D configuration as illustrated in Fig. 7-22. The evaluation point M_0 is plotted as a red star. Neighboring points M_i are represented with black circles.

The referenced direction vector of one sub-region l among q sub-regions is denoted as $\overrightarrow{dv}_l$. The angle between $\overrightarrow{M_0M_i}$ and direction vector $\overrightarrow{dv}_l$ is denoted as $\beta_{i,l}$ and given by $\beta_{i,l}(M_i, \overrightarrow{dv}_l) = \langle \overrightarrow{M_0M_i}, \overrightarrow{dv}_l \rangle$. The local direction weight $w_{i,l}^{dir}$ at point M_i with

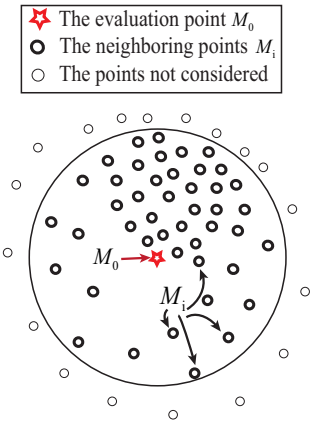


Figure 7-22: A 2D configuration of point cloud used to discribe the information point selection strategy

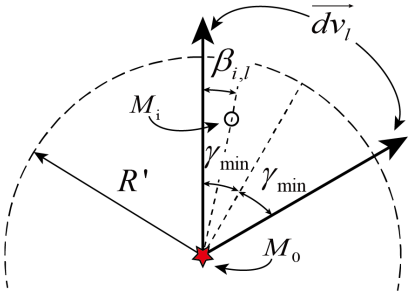


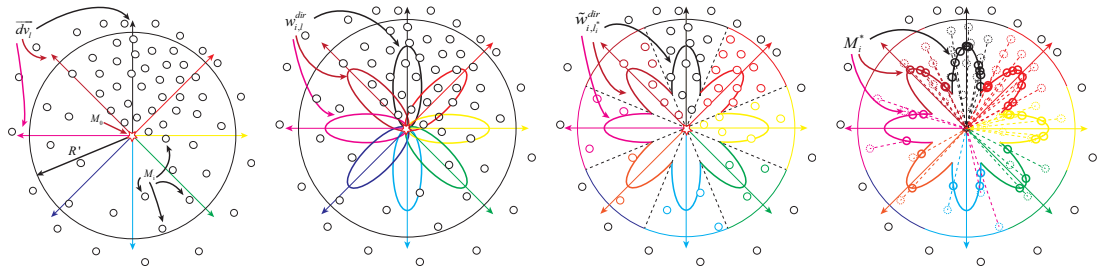
Figure 7-23: Variables related to the computation of direction weights

respect to the referenced direction vector $\overrightarrow{dv_l}$ is computed using Eq. (7.40a)(7.40b).

$$w_{i,l}^{dir}(M_i, \overrightarrow{dv_l}) = \begin{cases} \dfrac{cos(\beta_{i,l}) - cos(\gamma)}{1 - cos(\gamma)}, & \text{if } cos(\beta_{i,l}) \geq cos(\gamma) \quad (7.40a) \\ 0, & \text{otherwise} \quad (7.40b) \end{cases}$$

where γ is a parameter witch controls the boundary of each sub-regions. In order to cover all the vicinity space of the evaluation point, γ must be greater than a threshold γ_{min}. γ_{min} is a priori calculated and depends on the chosen set of referenced direction vectors. At point M_i, the effective direction weight is defined by a maximum criterion $\widetilde{w}_{i,l_i^*}^{dir} = max(w_{i,l}^{dir} \mid_{l=1}^{q})$, where l_i^* is the number of the sub-region that satisfies this criterion. This definition of effective direction weight can be expressed as: $\widetilde{w}_{i,l_i^*}^{dir} = w_{i,l}^{dir} \cdot \delta_{l_i^*}^{l}$, ($\delta_{l_i^*}^{l}$ is the Kronecker delta), which means that effective direction weight $\widetilde{w}_{i,l_i^*}^{dir}$ has no influence outside the sub-region numbered as l_i^*. Therefore, a surface which covers all the vicinity space and divides the space into sub-regions can be formed by the effective direction weight function when $\beta_{i,l}$ varies continuously. This surface is the partition surface illustrated in Fig. 7-21.

A step-by-step illustration of the process of categorizing all the neighboring points into the sub-regions by effective direction weights is given in Fig. 7-24 using the same



(a) Referenced direction vectors (b) Direction weight surfaces (c) Effective direction weight surface (d) Neighboring points categorization

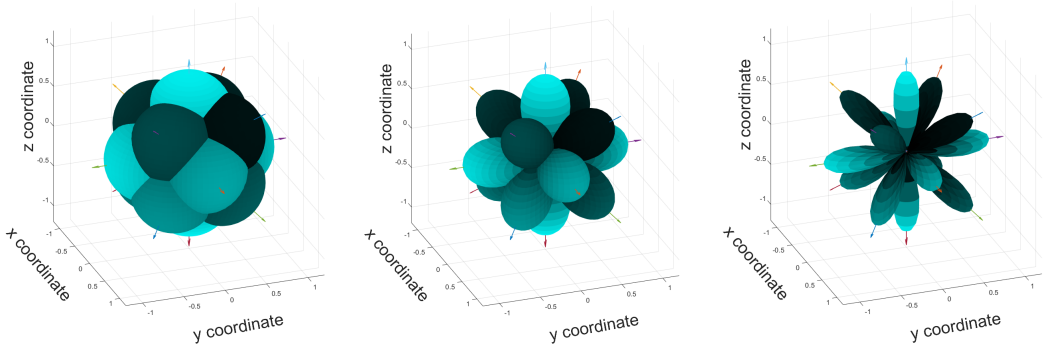
Figure 7-24: Categorizing all the neighboring points into the sub-regions by effective direction weights

configuration as the Fig. 7-22. The eight referenced direction vectors distributed isotropically around the evaluation point are built (Fig. 7-24(a)). Eight surfaces are then formed around eight referenced direction vectors by the direction weights as β varies (Fig. 7-24(b)). These surfaces are indicated by different colors in the figure and intersect with each other. The eight surfaces are merged into one surface which is the effective direction weight surface (Fig. 7-24(c)). The eight sub-regions are then defined by the eight radial regions which are separated by dash lines. The radial direction $\overrightarrow{M_0M_i}$ of each neighboring point M_i has an intersection point M_i^* with the effective direction weight surface (Fig. 7-24(d)). We can now categorize all the neighboring points into the eight sub-regions by M_i^*. In addition, the value of effective direction weight at each neighboring point is represented by the distance between M_i^* and M_0.

The same process can be performed in 3D. The only difference is the choice of direction vectors. In practice, the number of sub-regions is linked to the number of points needed to build the approximation. We chose a partition based on $q = 14$ direction vectors and therefore 14 sub-regions. This value is well adapted to the density of neighboring points when 10-node tetrahedral elements are used. Six directions are given by the global coordinate axis, and the other eight directions are given by the diagonals of eight virtual quadrants centered at the evaluation point with edges parallel to the global coordinate axis, as shown in Fig. 7-21. The selection weight matrix $\mathbf{w}^{sel} = w_{i,l}^{sel}$, which provides the selection weight of ith neighboring

point in the lth sub-region, is expressed as in Eq. (7.41):

$$w_{i,l}^{sel} = \widetilde{w}_{i,l_i^*}^{dir} \cdot w_i^{\infty} \tag{7.41}$$



(a) $\gamma = \pi/2$ (b) $\gamma = \pi/4$ (c) $\gamma = \pi/8$

Figure 7-25: Influence of γ on the effective direction weight function

Remark: In Eq. (7.41), the interpolating distance weight w_i^{∞} is only depends on the position of the neighboring points. However, effective direction weight $\widetilde{w}_{i,l_i^*}^{dir}$ depends on both the position of the neighboring points and the parameter γ. As illustrated in Fig. 7-25, the smaller the parameter γ is, the more the value of effective direction weight $\widetilde{w}_{i,l_i^*}^{dir}$ varies from the center to the border in each sub-region. As a result, if γ decreases, more influence is given to the effective direction weight $\widetilde{w}_{i,l_i^*}^{dir}$ in relation to the distance weight w_i^{∞}. In addition, in order to make sub-regions cover all the neighboring points, it is necessary that $\gamma \geq \gamma_{min}$. The local direction weight is given more prominence, if the gradation of the mesh is stiff. The parameter γ exerts more influence on the final results, if the number of referenced vector q is small (for instance $q = 6$). We experienced that with q equals to 14, the variation of parameter γ has little influence on the interpolation results. In practice, $\gamma = 1.05\gamma_{min}$.

When the selection weight matrix $\mathbf{w}^{sel}$ is constructed for all sub-regions at all neighboring points, the information points are chosen in each sub-region respectively. These information points maximize the selection weights in their own sub-regions. The final selection is displayed in Fig. 7-26. The selected information points are

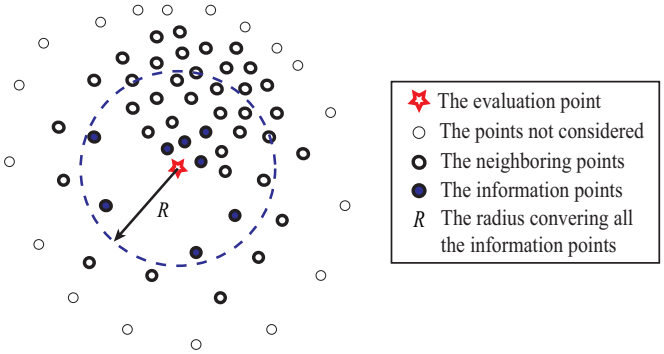
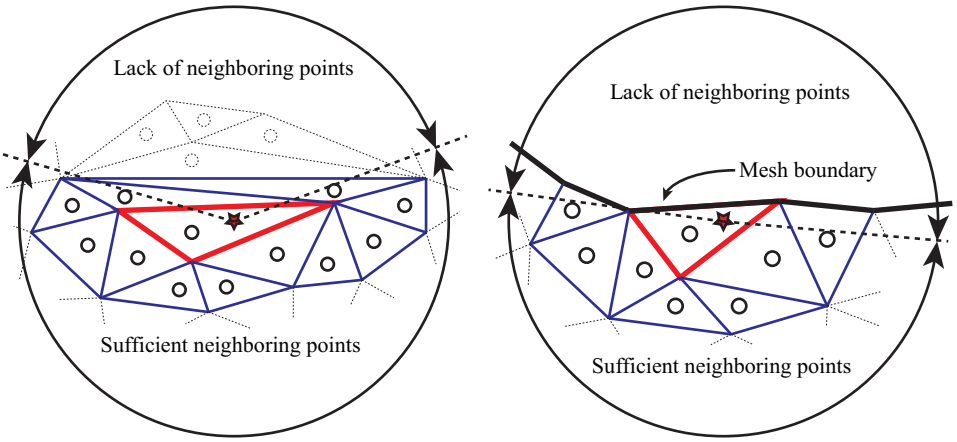


Figure 7-26: The set of final selected isotropically distributed information points

plotted as blue points which are not the nearest neighboring points but isotropically distributed around the evaluation points, meanwhile as close to the evaluation point as possible.

7.5 Iterative strategy to guarantee the isotropic distribution of the information points

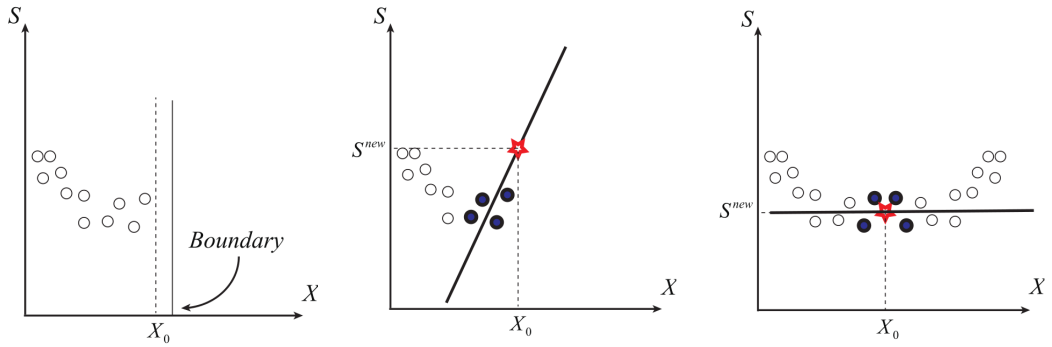


(a) Lack of neighboring points (b) Boundary effect

Figure 7-27: Lack of information points situation

The ideal situation is to select one information point in each sub-region so that the number of information points equals to the total number of sub-regions. However, this is not always the case. Sometimes, there is no neighboring point located in a sub-region and no information point can be selected. In this situation, the number

of selected information points is less than the total number of sub-regions. As a consequence, on the one hand, the number of information points may not be sufficient to build the Diffuse Interpolation. On the other hand, even if the number is enough, their distribution is no longer isotropic what may spoil the accuracy of the Diffuse Interpolation. This lack of neighboring point problem may be caused by two reasons, namely, the elements in certain directions are rather flat so that no integration points within them can be captured as neighboring points in a first element layer or the evaluation point is located at the surface of the mesh (as illustrated in a 2D example in Fig. 7-27(a)) and no element is located in certain directions so that no integration point can be taken into account (as illustrated in a 2D example in Fig. 7-27(b)).



(a) 1D situation of a point X_0 located near the boundary (b) Interpolation without information compensation (c) Interpolation with information compensation

Figure 7-28: 1D example of the influence of the boundary on the interpolation results

In order to solve these problems, we propose to enlarge the searching range for neighboring points. In practice, as explained in section 7.3, one more layer (three layers in total) of elements are considered and all of the integration points of the elements in these three layers are searched as neighboring points. Even if this process obtains much more neighboring than considering only two layers, the number of selected information points is still controlled. In a 3D tetrahedron mesh, as explained in last paragraph, either relatively planar elements or the geometry boundary cause the lack of information points problem. If the problem can be solve by enlarging one more layer of elements, then the problem is caused by relatively planar elements (no need to enlarge two layers of elements). Otherwise, the problem is caused by geom-

etry boundary. The lack of information points deteriorates the interpolation results. This problem is illustrated in a 1D example in Fig. 7-28 in which the neighboring points are denoted as small circles, the information points are denoted as blue circles and the evaluation point is denoted as a star. In Fig.7-28(b), only the neighboring points located at the left side of the evaluation points can be selected as information points what causes the lack of information points at the right side. As a consequence, the value at the evaluation points is overestimated. We propose here an information compensation process which reconstructs a set of virtual neighboring points which are the symmetrical set of real neighboring points. The information points are then selected from all the neighboring points including both real and virtual as illustrated in Fig.7-28(c). We can see that the technique provides a better approximation at the boundaries.

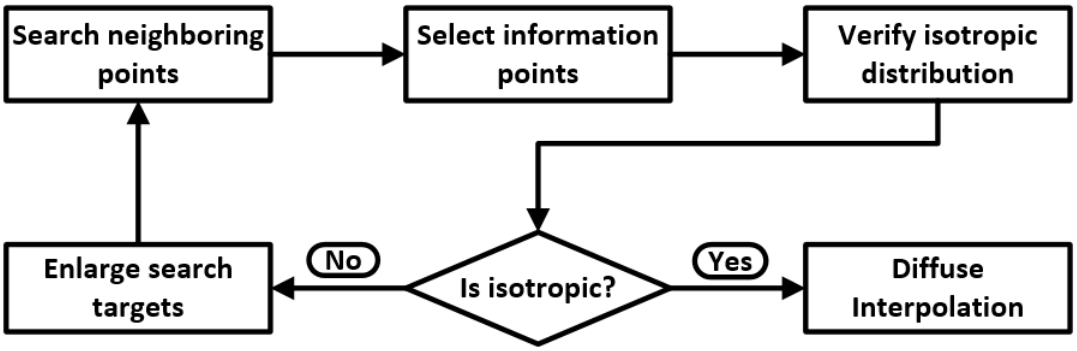


Figure 7-29: Iterative process of improved point selection strategy implemented in Diffuse Interpolation

In order to check the isotropy of the information point set, we transfer all the points from the Cartesian coordinate system $X(x, y, z)$ to a spherical coordinate system $\Theta(r, \varphi, \theta)$ centered at the evaluation point M_0, where φ denotes the precession angle between 0 and 2π, and θ denotes the nutation angle between $-\pi/2$ and $\pi/2$. The precession interval is then divided evenly into eight sections and the nutation interval is divided evenly into four sections. If there are two or more empty sections in φ, or one or more empty sections in θ, then the third layer of elements is used, in order to obtain more neighboring points. Otherwise, to estimate the value at the evaluation point, the Diffuse Interpolation is performed based only on the distance weight w_i^{∞}.

The whole iterative strategy is schemed in Fig. 7-29.

7.6 Analytic field transfer examples

Our 3D hybrid field transfer is aimed to be implemented in 3D h-adaptive finite element methodology [45], especially for the simulation of ductile damage and crack propagation. The importance of such implementation is to transfer highly localized field (cumulative plastic, stress and damage) from a coarse mesh to a finer mesh (after adaptation). Because the cracks are linked and sensitive to the highly localized damage field, the extrema value of the field should be preserved as much as possible with a minimum numerical diffusion. We notice that the more localized the field is, the more important for the field transfer operator is to preserved extrema value. In order to validate the advantages of extrema value preservation and numerical diffusion minimization of the field transfer in this work, different transfer operators are used to transfer a given analytic field $f(x, y, z)$ with several extrema values having different level of localization from coarse mesh to fine mesh. The localization level of field $f(x, y, z)$ at a point $P_0(x_0, y_0, z_0)$ describes the average variation of the field in all directions within a vicinity denoted as Ω_{P_0} of the point P_0. In order to give a standard concept to arbitrary fields, the variation is proposed to be uniformed by the difference of maximum and minimum value of a field. Therefore, the definition of localization level is given in Eq.7.42 and denoted as $\zeta(f)$.

$$\zeta(f) = \frac{1}{\text{area}(\partial\Omega)} \oiint_{\partial\Omega} \frac{f(x_0,y_0,z_0)-f(x_j,y_j,z_j)}{\max f(x,y,z)-\min f(x,y,z)} \cdot \frac{1}{l^{eig}(x_0-x,y_0-y,z_0-z)} d\partial\Omega \tag{7.42}$$

in which $l^{eig}(x_0 - x, y_0 - y, z_0 - z)$ is the eigen length defining the vicinity Ω_{P_0} in the direction $\overrightarrow{P_0P}$ (the point $P(x, y, z)$ is an arbitrary point in the vicinity). The boundary of the vicinity Ω_{P_0} is denoted as $\partial\Omega_{P_0}$. The integration can be discretized for finite element mesh in which the value of the fields only located at the integration points of the element. In practice, the element E_0 which contains the point P_0 is searched. Then the vicinity Ω_{P_0} can be defined as the volume composed of all the n

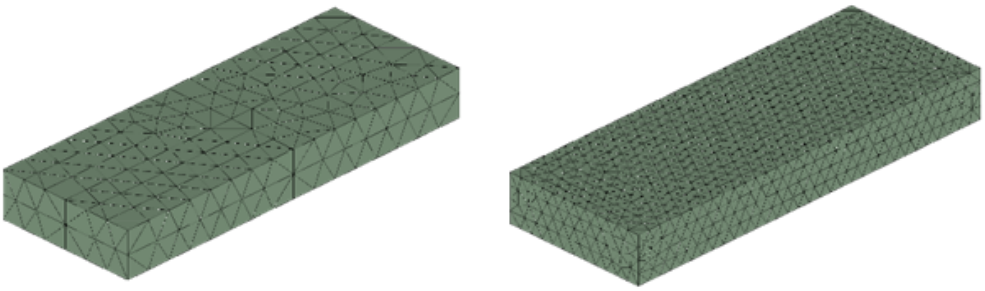
elements E_i attached to this element E_0 including itself. All the other points $P(x, y, z)$ are discretized into all the integration points of all the $n-1$ attached elements. In order to eliminate the influence of various element sizes, the eigen-length in the E_i element direction, it is defined as in Eq.7.43 and denoted as l_i^{eig}.

$$l_i^{eig} = \frac{2\sqrt{(x_0 - x_j)^2 + (y_0 - y_j)^2 + (z_0 - z_j)^2}}{h(E_0) + h(E_i)} \tag{7.43}$$

In which $h(\bullet)$ denotes the size of the element. Above all, assume that each element has m integration points, the discretized definition of localization level in finite element can be expressed in Eq.7.44

$$\begin{aligned} \zeta(f) = \; & \frac{1}{n \times m - 1} \sum_{i=1}^{n} \sum_{j=1}^{m} \frac{f(x_0, y_0, z_0) - f(x_{i,j}, y_{i,j}, z_{i,j})}{\max f(x,y,z) - \min f(x,y,z)} \\ & \cdot \frac{h(E_0) + h(E_i)}{2\sqrt{(x_0 - x_{i,j})^2 + (y_0 - y_{i,j})^2 + (z_0 - z_{i,j})^2}} \end{aligned} \tag{7.44}$$

in which (i, j) means the jth integration point in the ith element and $x_0 \neq x_{i,j}$, $y_0 \neq y_{i,j}$ and $z_0 \neq z_{i,j}$.



(a) The cuboid with mesh size equals to 10 (b) The cuboid with mesh size equals to 5

Figure 7-30: The coarse and the fine mesh on a cuboid

The analytic field transfer processes are performed on a 3D cuboid ($160 \times 60 \times 20$) using different transfer operators. The cuboid is discretized by linear tetrahedron elements with one integration point. The element size equals to 10 in the coarse mesh (Fig.7-30(a)) and 5 in the fine mesh (Fig.7-30(b)). The function $f(x, y, z)$ in Eq.7.45 is chosen as an analytic function of the field in which the localization level $\zeta(f)$ changes only with x. The original point of the Cartesian coordinate system is located at the geometry center of the cuboid. The shape of the function $f(x, y, 0)$

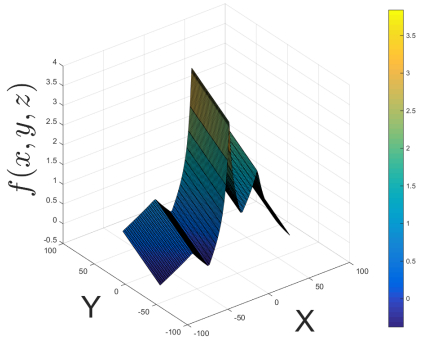


Figure 7-31: The shape of the analytic field $f(x, y, z)$

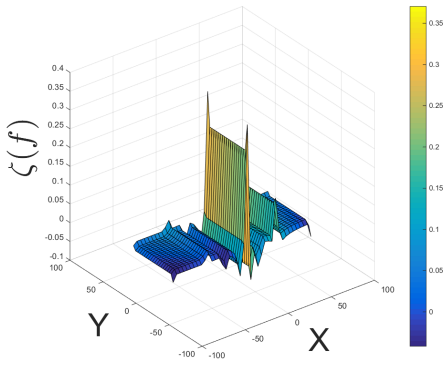


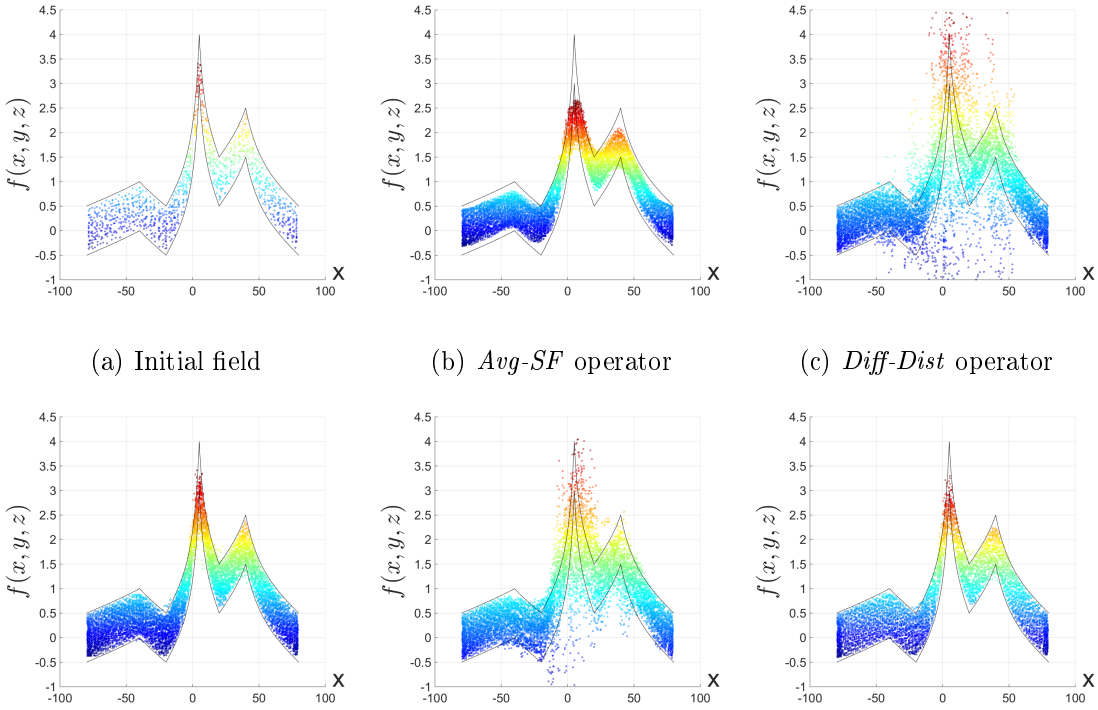
Figure 7-32: The localization level of the initial analytic field $\zeta(f)$

and $\zeta(f)$ are plotted respectively in Fig.7-31 and Fig. 7-32.

$$f(x,y,z) = \begin{cases} \tan((x+80)\text{atan}(0.5)/40) + (z+y)/80, & \text{if} \quad -80 \leq x < -40, \\ \tan((x+20)/20 * \text{atan}(0.5)) + (z+y)/80, & \text{if} \quad -40 \leq x < -20, \\ \tan((x+20)/25 * \text{atan}(3.5)) + (z+y)/80, & \text{if} \quad -20 \leq x < 5, \\ \tan((x-20)/15 * \text{atan}(2.5)) + (z+y)/80 + 1, & \text{if} \quad 5 \leq x < 20, \\ \tan((x-20)/20 * \text{atan}(1.0)) + (z+y)/80 + 1, & \text{if} \quad 20 \leq x < 40, \\ \tan((x-80)/40 * \text{atan}(2.0)) + (z+y)/80, & \text{if} \quad 40 \leq x < 80. \end{cases} \tag{7.45}$$

In order to represent the analytic field within the mesh clearly, the x coordinate of an integration point is used as its position so that the 3D field can be plotted in a 2D plane. The initial field denoted as f_{init} on the coarse mesh is created at integration points of all the elements. The 2D representation of f_{init} is plotted in Fig.7-33 (a) in which the black lines are the upper and lower analytic limit of the field. Five transfer operators are used to transfer the field f_{init} from the coarse mesh to the fine mesh: the second group transfer ($IP^{old} \rightarrow NP^{old} \rightarrow NP^{new} \rightarrow IP^{new}$) with averaging extrapolation and shape function interpolations abbreviated as *Avg-SF* operator, transfers in the third group ($IP^{old} \rightarrow IP^{new}$) based on meshless moving least square method with and without isotropic information point selection abbreviated as

Diff-DD and *Diff-Dist* operator, transfers in the forth group ($IP^{old} \rightarrow NP^{new} \rightarrow IP^{new}$) in which the first part $IP^{old} \rightarrow NP^{new}$ is carried out by Diffuse Interpolation with and without isotropic information point selection; the second part $NP^{new} \rightarrow IP^{new}$ is carried out by shape function interpolation abbreviated as *Hyb-DD* and *Hyb-Dist*. The fields transferred by these five operators are given in Fig.7-33 (b)-(f). The corresponding localization level of all these fields are displayed in Fig.7-34.

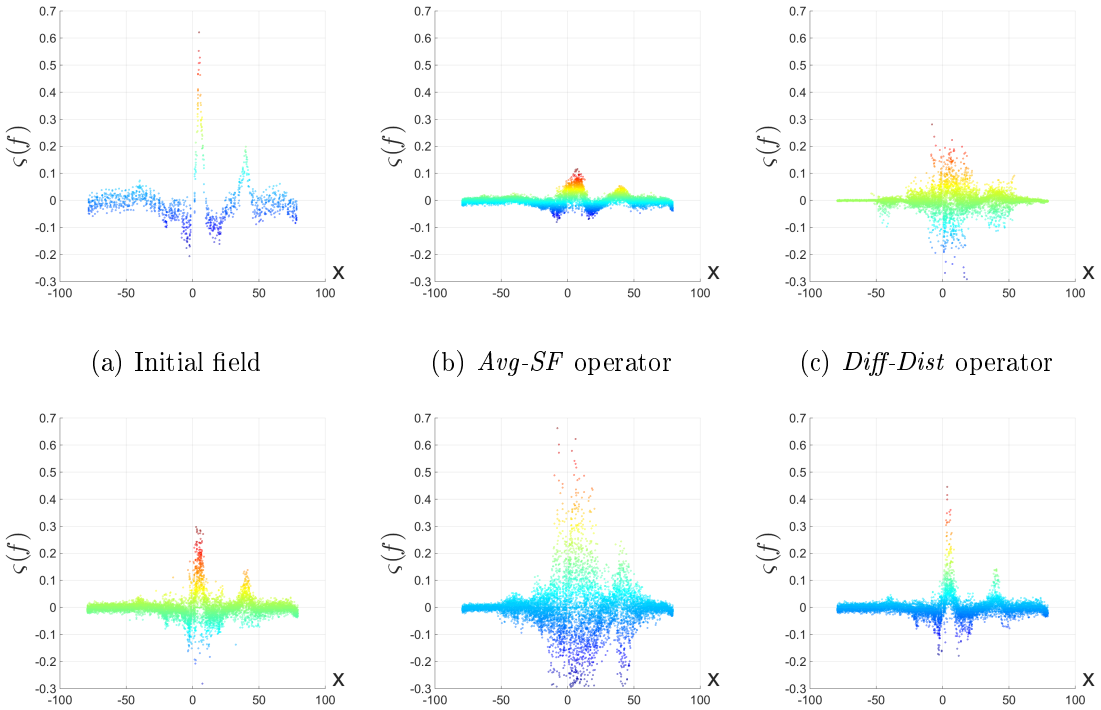


(a) Initial field (b) *Avg-SF* operator (c) *Diff-Dist* operator

(d) *Diff-DD* operator (e) *Hyb-Dist* operator (f) *Hyb-DD* operator

Figure 7-33: A 2D illustration of initial and transferred fields by different transfer operators

We can see in Fig.7-33(a) and Fig.7-34(a) that the field $f(x, y, z)$ on the initial coarse mesh has 5 localized zones: 3 maxima and 2 minima with different localization levels. The ideal transfer operator is expected to reconstruct the field as the initial field which is exactly distributed between two black lines (no blank space inside and no overflow outside). We can see from the Fig.7-33(c) and 7-33(e) that as long as the Diffuse Interpolation without isotropic information point selection is implemented in 3D field transfer, the severe numerical diffusion happens in the localized zones which



(a) Initial field (b) *Avg-SF* operator (c) *Diff-Dist* operator

(d) *Diff-DD* operator (e) *Hyb-Dist* operator (f) *Hyb-DD* operator

Figure 7-34: A 2D illustration of localization level of the corresponding initial and transferred fields

spoils the field. However, the $Hyb - Dist$ operator reduces the numerical diffusion compared to $Diff - Dist$ operator. In Fig.7-33(b) and Fig.7-34(b), we notice that the shape function operator limits the numerical diffusion but declines the extrema of the field and reduces the localization level of the field. The transfer operators based on Diffuse Interpolation with isotropic information point selections, as shown in Fig.7-33(d) and 7-33(f) preserves the extrema of the fields. In addition, $Hyb - DD$ operator has less numerical diffusion than $Diff - DD$ operator.

In an adaptive methodology, the mesh is adapted many times and the fields are transferred after each remeshing step. The errors caused by remeshing and transferring are cumulated during the simulation and should be minimized. However, the errors must be controlled during transfer process. The L_2 error between the analytic field denoted as f on the coarse mesh and the transferred field denoted as $\hat{f}$ on the fine mesh is used and expressed in Eq.7.46.

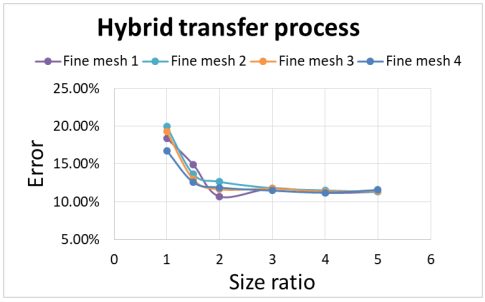


Figure 7-35: The errors from hybrid field transfer in function of size ratio between coarse and fine mesh

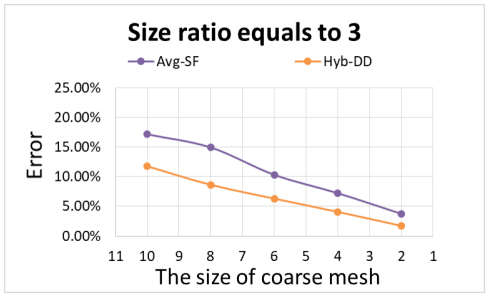
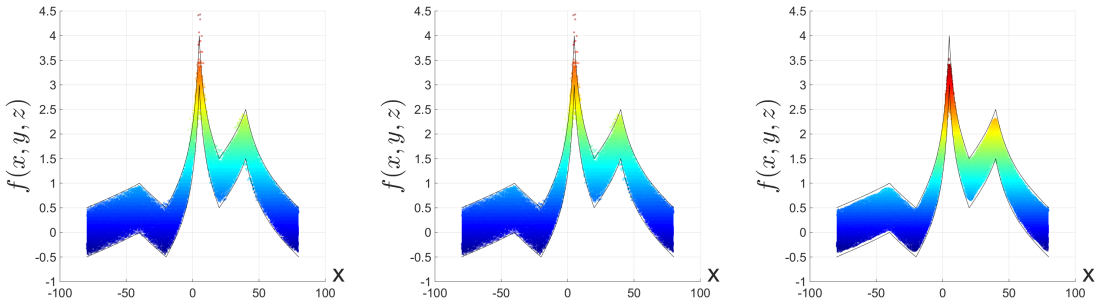


Figure 7-36: The errors from *Avg-SF* and *Hyb-DD* transfer operators in function of the size of the coarse mesh

$$e_{L_2} = \left[\int_{\Omega} \left(f(x,y,z) - \hat{f}(x,y,z)\right)^2 \frac{d\Omega}{\Omega}\right]^{1/2} \tag{7.46}$$

Two transfer operators, $Avg-SF$ and $Hyb-DD$, are used to transfer the same analytic field (Eq.7.45) on the same cuboid geometry (7-30). The size ratio between coarse mesh and fine mesh is fixed to 3. We can see from the Fig.7-36 that with the size of the coarse mesh declining, the error e_{L_2} caused by both transfer operators declines. In addition, this error caused by *Avg-SF* operator is always higher than *Hyb-DD* operator. We do not compare the error cause by *Diff-Dist*, because without isotropic information point selection the error is much bigger than the other and cannot be controlled by reducing the coarse size mesh. Furthermore, we fix the size of the coarse mesh equals to 10. Four different discretization ways are use to generate the fine meshes for different size ratio. The analytic field (Eq.7.45) is transferred from the coarse mesh to these fine meshes by *Hyb-DD* operator. We can see from Fig.7-35 that no matter how the geometry is discretized, the error e_{L_2} depends only on the size ratio. when size ratio is bigger than 2, the error caused by *Hyb-DD* operator converges to a constant value. This property provides a robustness way to limit the error if local remeshing is implemented in the simulation which leads to the size ratio no smaller than 2 in a local zone between two meshes.

As explained in section 7.4.3, the transfer process near the boundary should be specially treated by an information compensation process. In order to show the effect



(a) Initial field

(b) Without information compensation

(c) With information compensation

Figure 7-37: Information compensation process improves the transferred field while reducing the numerical diffusion

of this compensation process, the *Hyb-DD* operator is performed with and without the compensation respectively. The cuboid geometry (7-30) is discretized by a quadratic tetrahedral element with 10 nodes and 4 integration points. The analytic field on the coarse mesh is displayed in Fig.7-37(a). The transferred fields on the fine mesh without and with the compensation are displayed in Fig.7-37(b) and Fig.7-37(c). We can see that the compensation process reduces the numerical diffusion near the boundary of the geometry, especially in the localized zones.

Chapter 8

Numerical and experimental results

8.1 Comparison of crack propagation with different transfer operators

During the metal forming processes, several integration fields, such as: stress, cumulative plastic strain and damage are highly localized in the zones where cracks appear. The representation of cracks is usually dependent on damage variable. Saanouni [80] propose an element deletion method in which the cracks are represented by deleting the totally damaged elements in the mesh. An element is considered as totally damaged when the damage value exceeds a maxima threshold value at all of its integration points. The element deletion method is widely discussed in h-adaptive method [17][40]. The technique is proved to be sensitive to the extrema value of the field. Therefore, preservation of the extrema of the field such as: damage and cumulative plastic strain fields is of the great importance for the crack initiation and propagation with respect to field transfer.

In order to validate the implementation of *Hyb-DD* field transfer operator in the metal forming analysis, a tensile test is carried out on a specimen made of $2mm$ thick commercial cold-rolled SPCC steel [61]. The dimension of the specimen is illustrated in Fig.8-1. It was clamped by two rigid grippers: Gripper 1 was moved at a constant velocity and Gripper 2 was fixed. A h-adaptive methodology with local

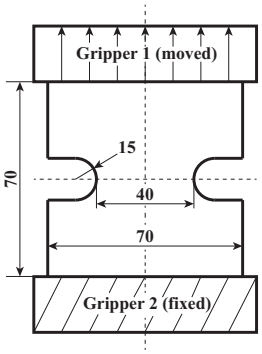


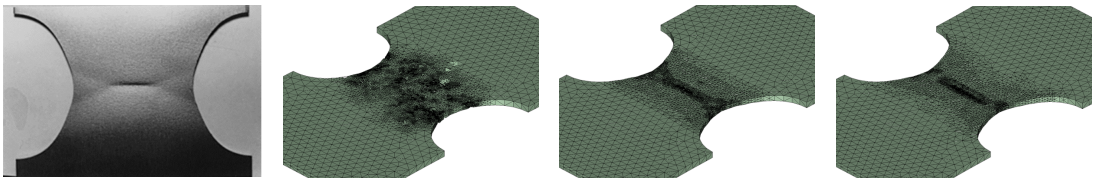
Figure 8-1: Dimension of the specimen used to perform the tensile test

remeshing [71] and element deletion technique is used to perform the simulation. The velocity is stored at the nodes and it is transferred from the old mesh to the new mesh directly by finite element shape function interpolation. For the state variables stored at integration point, three different transfer operators ($Diff-Dist$, $Avg-SF$ and $Dyb-DD$) are implemented in the simulation respectively. In each loading sequence, only a minimum set of variables at integration point are transferred, namely cumulative plastic strain p, isotropic hardening r, kinematic hardening $\underline{\alpha}$, isotropic damage d and plastic dissipation R. ABAQUS/Explicit® is used as a solver and the specimen is discretized by a quadratic 10-node tetrahedral element C3D10M proposed by ABAQUS which avoids volumetric locking problems. The numerical results are compared to the experimental results reported by reported by Niu et al. [61]. The parameters used to describe the material SPCC steel in the model are detailed in the Table 8.1.

Table 8.1: The values of material parameters for cold-rolled SPCC steel

$E(GPa)$	ν	$\sigma_y(MPa)$	$Q(MPa)$	b	C	a
220	0.29	165	720	1.2	2600	40.0
S	s	β	Y_0	ω	h_{clo}	h^d_{min}
8.5	1.0	0.5	0.0	4.0	0.0	0.1

The initiations of the crack on the specimen are shown in Fig.8-2. In our tensile test (Fig.8-2(a)) the crack initiated at the center of the specimen and ran along a line symmetrically between the two notches. In our simulations, where the *Diff-Dist*



(a) Experiment (b) *Diff-Dist* operator (c) *Avg-SF* operator (d) *Hyb-DD* operator

Figure 8-2: Experiment and numerical results with different transfer operators of crack initiation on the specimen

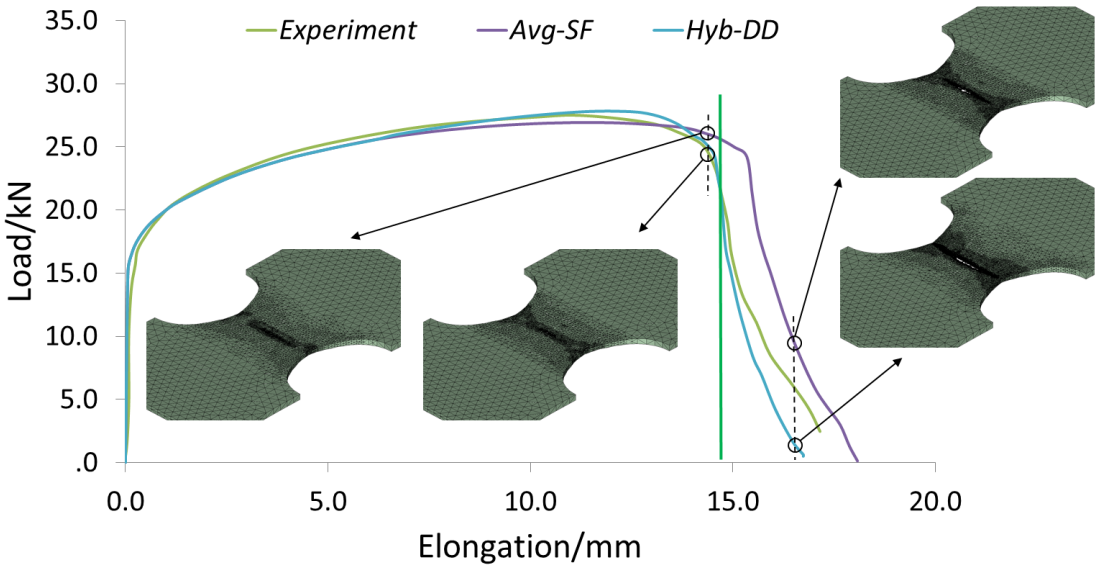


Figure 8-3: The elongation-load curves from experiment and numerical simulation of the whole tensile test

operator is used, the crack initiates at a different place (Fig.8-2(b)) from the experimental results. Because of the substantial numerical diffusion resulting from the transfer process, the fields are damaged, and consequently the mesh is damaged too. Where the $Avg-SF$ and $Hyb-DD$ operators are used, on the other hand, the cracks initiate at the same place (Fig.8-2(c) and Fig.8-2(d)) as in the experimental results. This is because the numerical diffusion is limited. However, we remark that due to a contraction of the extrema of the damage field, the $Avg-SF$ operator delays the initiation of the crack, which reduces the reliability of the simulation, especially in an industrial context. This delay in initiation can be seen more clearly when the ex-

perimental and numerical elongation-load curves are compared. As shown in Fig.8-3, with the *Avg-SF* operator the greatest load is less than in the experimental results, and crack initiation is delayed, meaning that the total elongation time is longer in the simulation than in the experiment, before the specimen becomes totally fractured. In the simulation using the *Hyb-DD* operator, even though the time required for the specimen to become totally fractured is slightly less than in the experiment results, this operator describes faithfully the initiation of the crack. Hybrid transfer is therefore seen to give more reliable simulation results than *Avf-SF* transfer when crack initiation is the most important concern.

8.2 Tensile test on a series of double notched specimens

In order to validate the robustness of our h-adaptive methodology, we use the same material as in section 8.1 table 8.1. The similar geometry as in Fig. 8-5 is used but with a series of various radii of the U notch denoted as R and chosen as $2mm$, $6mm$, and $15mm$ for the three specimens.

Fig. 8-7 ($R = 2mm$), Fig. 8-8 ($R = 6mm$) and Fig. 8-9 ($R = 15mm$) show the results of the tensile tests, from both experiments and simulations. The figures include photos from the experiments, numerical deformed geometries, and adapted meshes at three different moments: just after maximum load, intermediate, and just before rupture. On the specimen with acute notches ($R = 2mm$), the crack initiates at both notch roots and then propagates along the ligament to its center. On the specimen with obtuse notches ($R = 15mm$), the crack initiates at the center of the ligament and then propagates along the ligament to both notch roots. On the specimen with intermediate notch ($R = 6mm$), the crack initiates at a point between the center of the ligament and notch root and then propagates to the center of the ligament and notch root at the same time. We can see from the deformed geometries of the specimens that the necking phenomenon and shape of the crack are well captured by the numerical

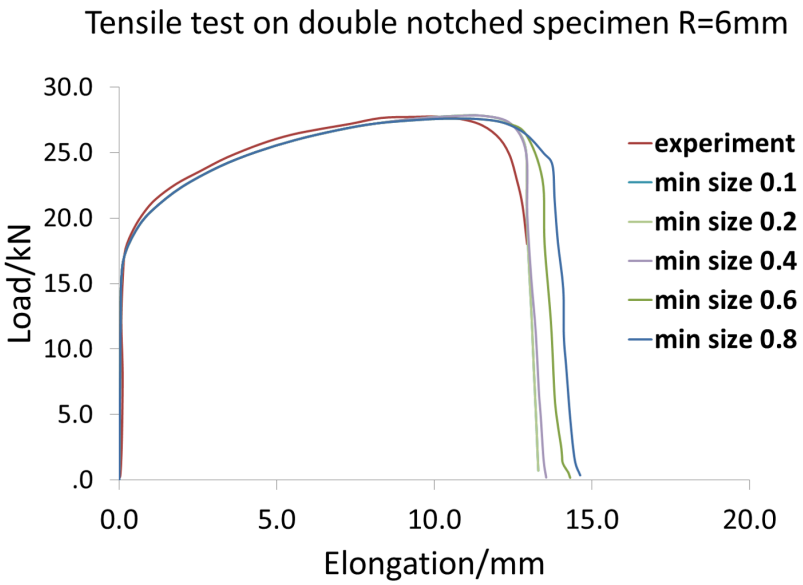


Figure 8-4: The elongation-load curves of the tensile tests on the specimen $R = 6mm$ using various minimum mesh size and the corresponding experimental result

model. Furthermore, the meshes are adapted as the crack propagates. In order to see the 3D crack clearly, a full crack on the specimen $R = 6mm$ is displayed in Fig. 8-10. The crack surface is composed of several layers of elements and the necking effect is obviously observed. The Elongation-Load curves for the three tensile tests, both experimental (Niu et al.) and numerical simulation, are shown in Fig. 8-6. For the specimens $R = 6mm$ and $R = 15mm$, the Load-Elongation curves from the simulations have a good agreement with those from the experiments.

Table 8.2: The value of parameters for size indicators

h_{max}	h^p_{max}	h^p_{min}	h^d_{min}	d_{min}	d_{max}	κ_1	κ_2
5.0	2.0	0.3	0.1	0.1	0.9	10.0	10.0

Our h-adaptive methodology is based on local elastoplastic damage model which inevitably yields mesh-dependent numerical solutions. The plastic flow leads to a highly localized narrow damage (shear) bands, which makes the results sensitive to the smallest mesh size. The simplest way to control this mesh dependency in terms of

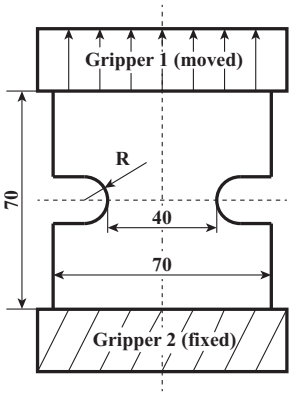


Figure 8-5: Tensile test: specimen dimensions (in mm) and boundary conditions

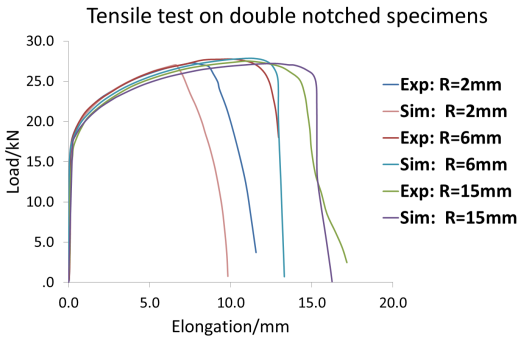


Figure 8-6: Comparison of experimental and numerical elongation-load curves for three tensile tests

our damage model is to identify all the material damage parameters by imposing the minimum mesh size h^d_{min} during the identification procedure. The minimum mesh size h^d_{min} therefore becomes an intrinsic material parameter. As a consequence, the whole h-adaptive methodology is based on the identification of the material parameters, and the crack is controlled by minimum mesh size. The parameters of size indicators used in this simulation are given in Table 8.2. The maximum mesh size h_{max} is tuned according to the geometry dimension of the specimen. In our case, the thickness of the specimen is only $2mm$ so that h_{max} can not be too much bigger than $2mm$, otherwise the quality of the elements in the original mesh can be very low. The determination of the minimum mesh size h^d_{min} is a crucial issue in this methodology. This parameter is identified by decreasing the value and comparing the elongation-loading curve from numerical simulation with the one from experiment until the difference between these curves converge. The different values of minimum mesh size is used to simulate a tensile test on the double notched specimen with $R = 6mm$. As shown in Fig. 8-4, the curves with different values of h_{min} is displayed and compared with the experimental curve. When the $h^d_{min} = 0.1mm$ and $h^d_{min} = 0.2mm$ the curves are nearly the same and give a good agreement with the experimental curves. Therefore, the minimum mesh size in this tensile test is set to $h^d_{min} = 0.1mm$. The two parameters κ_1 and κ_2 are the parameters to control the shape of the size indicator functions (κ_1 for plastic strain and κ_2 for damage). The bigger the κ_1 and κ_2 are the mesh size decreases more

quickly with the plastic strain and the damage increase.

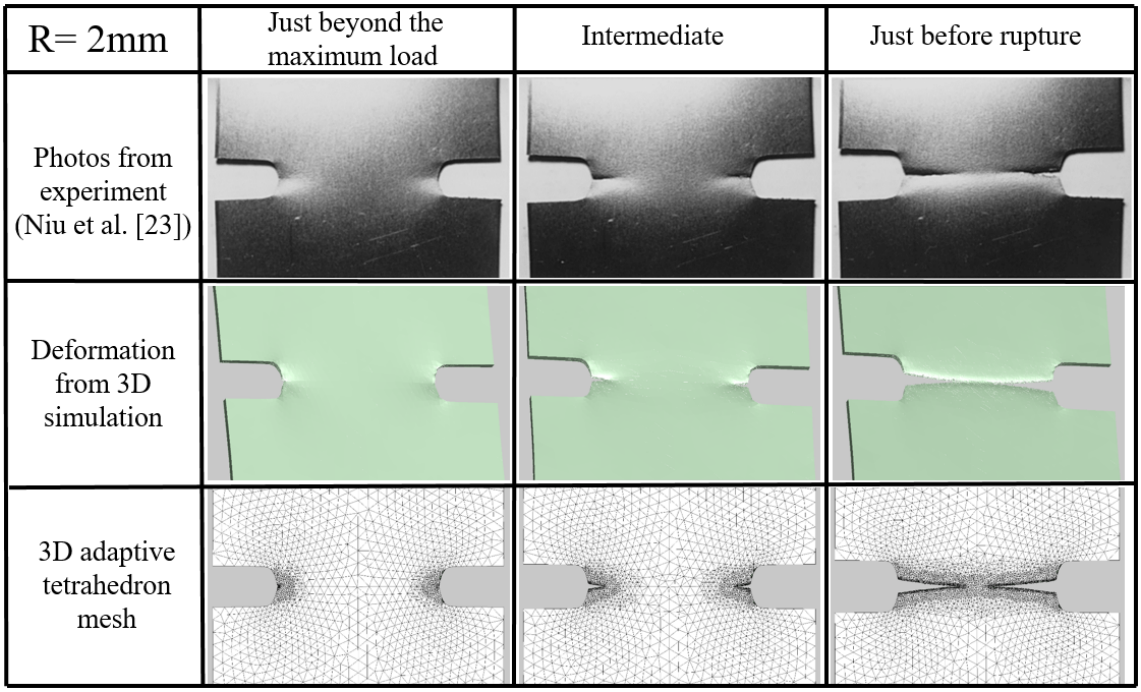


Figure 8-7: Comparison of experimental and 3D numerical results for specimen $R = 2mm$

We use the numerical simulation for specimen $R = 6mm$ as an example to give some details about the h-adaptive methodology in relation to meshing and remeshing aspects. The minimum size h^d_{min} is identified by using different values and comparing Elongation-Load curves with experimental results. This identification is performed with the specimen $R = 6mm$. As shown in Fig. 8-4, we can see that when the minimum size h^d_{min} decreases, the elongation-load curves get closer to the experimental curve. When minimum size $h^d_{min} = 0.2mm$ and $h^d_{min} = 0.1mm$ the curves nearly overlap the experimental curve. Therefore, for this material, h^d_{min} is fixed at 0.1.

Fig. 8-11 shows the increase in the numbers of vertex nodes, total nodes and the elements during the simulation for the $R = 6mm$ specimen. As we explained in chapter 6 above, the remeshing process, including mesh refinement, mesh optimization, element deletion and field transfer, has a very low computational cost in comparison to the solver. In Fig. 8-12 ($R = 6mm$) the x axis is the number of loading sequence and the y axis is the cumulative CPU time at the end of each loading sequence: in

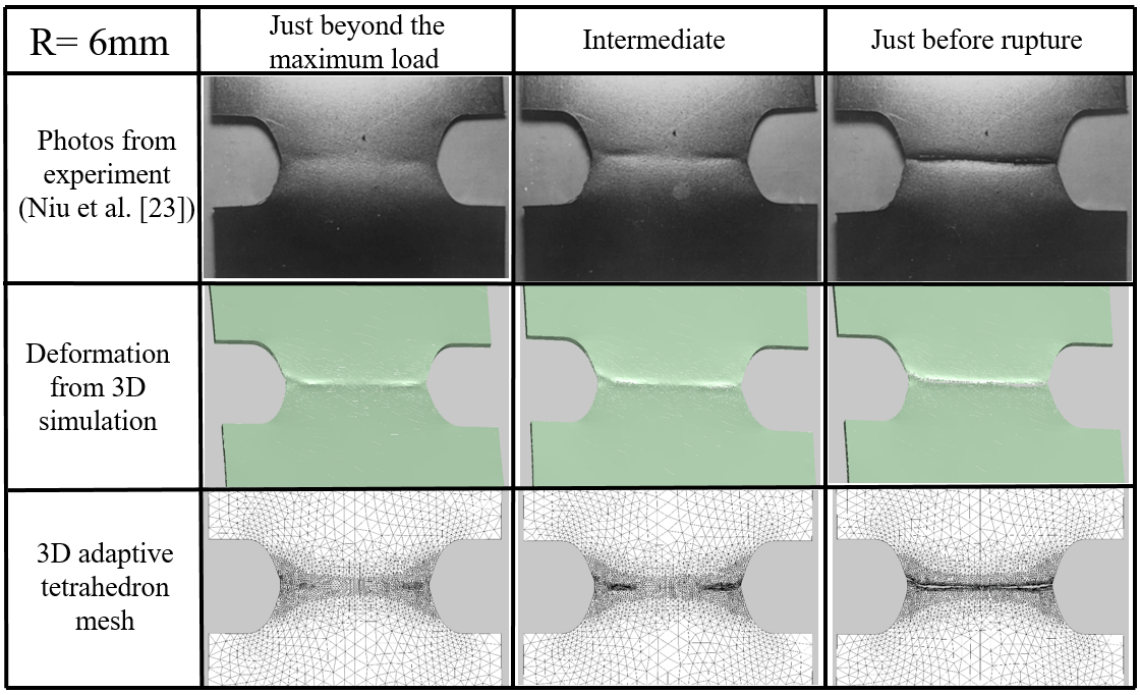


Figure 8-8: Comparison of experimental and 3D numerical results for specimen $R = 6mm$

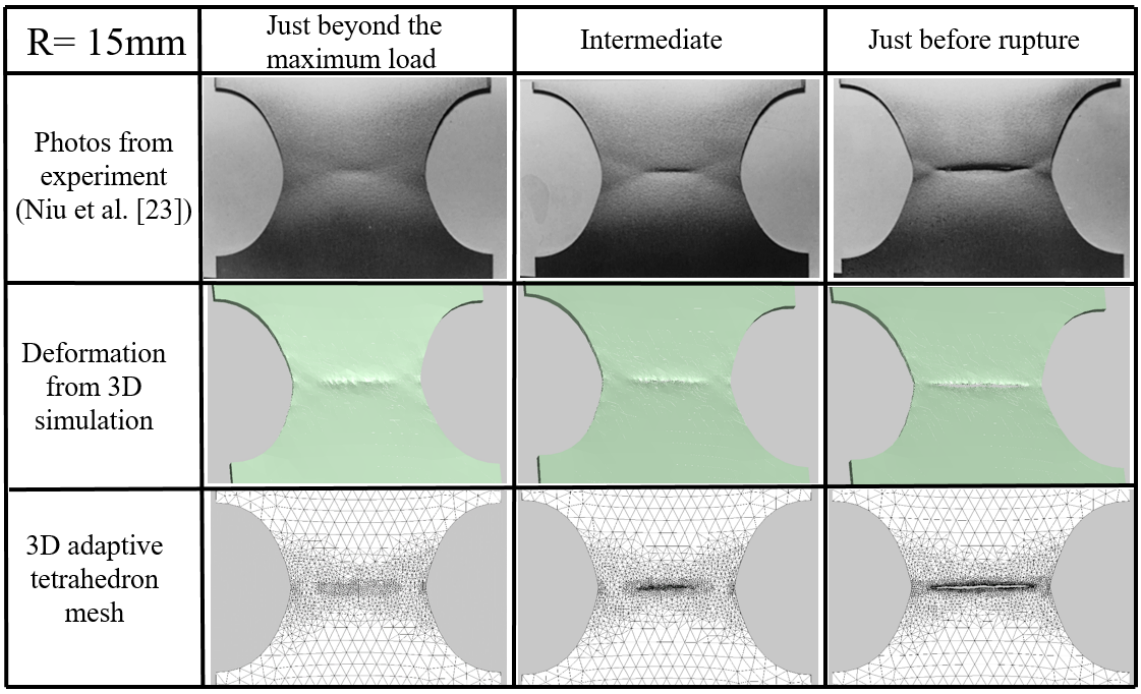


Figure 8-9: Comparison of experimental and 3D numerical results for specimen $R = 15mm$

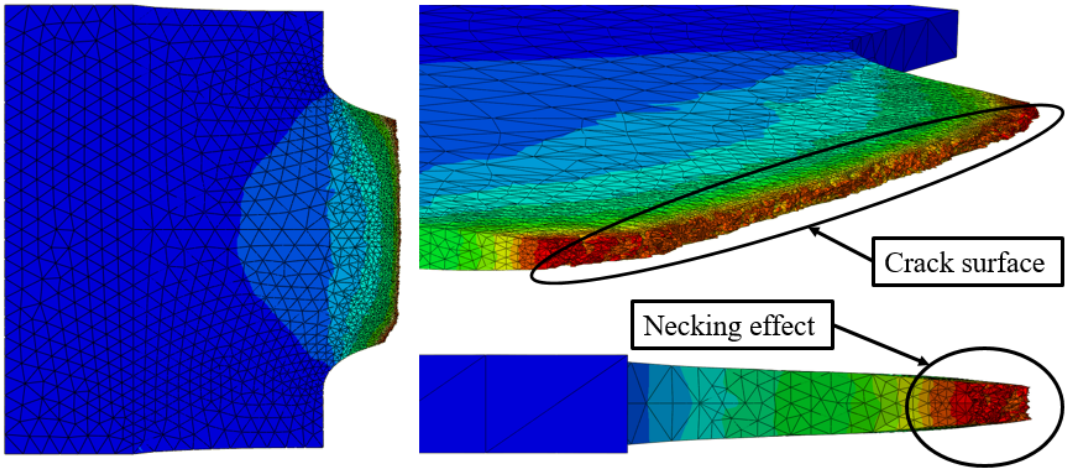


Figure 8-10: The full 3D numerical crack surface and necking effect on the specimen $R = 6mm$

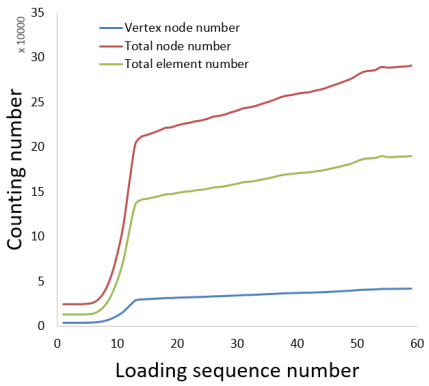


Figure 8-11: Numbers of nodes and elements variation during a simulation

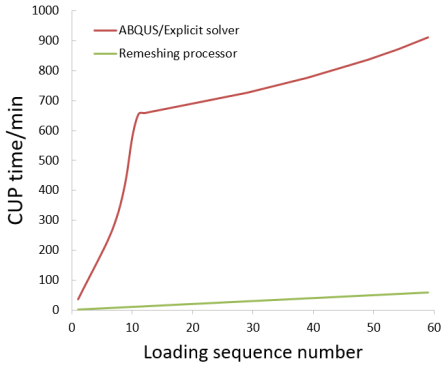


Figure 8-12: CPU time for solver and remeshing processor during a simulation

the simulation 50 CPUs are used at the same time for the solver, and only 1 CPU is used for the remeshing process.

8.3 Using the methodology with different elastoplastic models

An elastoplastic model proposed by Hooputra et al. [36] and integrated in ABAQUS has been first used to validate our adaptive tools at a lower computational cost (relatively 5 times lower than the model explained in chapter 3). The material used in this

simulation is an extruded aluminum alloy EN AW-7108 T6. This material behaves in an elastoplastic manner and can undergo damage due to either one or a combination of the following damage mechanisms: nucleation and coalescence of voids, shear band formation, and necking instability. In this simulation, a ductile damage (d_d) initiation criterion (Eq. B.1) and a shear damage (d_S) initiation criterion (Eq. B.4) are used to measure the damage value which is defined as: $d = \max(d_d, d_S)$. This damage criterion is used in our methodology. We can see from the Fig. 8-13 and 8-14, the crack initiates in the middle of the speciment and propagates to the two sides. However, the model is built for the material having a thin-wall structure, when the thickness of the specimen increases, the damage zone is not that localized. As a result, the crack is wide and unrealistic.

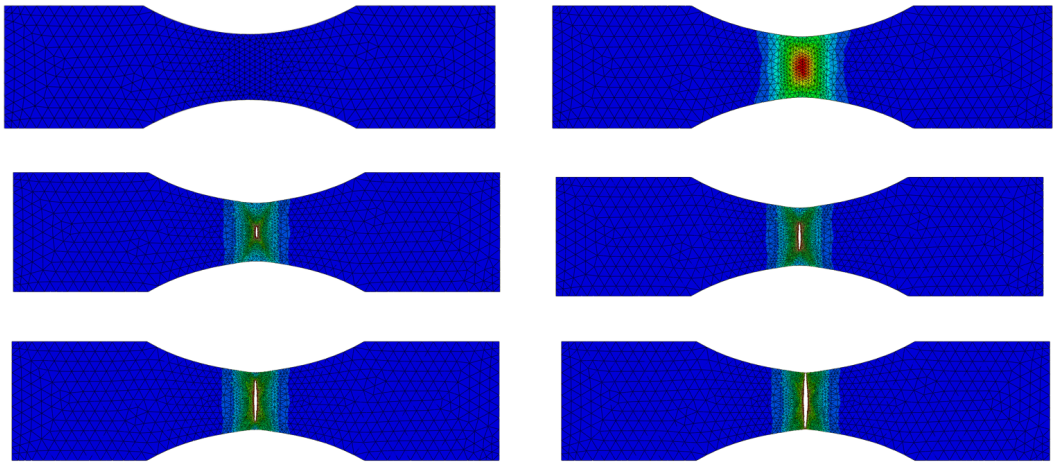


Figure 8-13: The crack initiation and propagation on the specimen

8.4 2D simulation and experiment example of blanking process

No blanking process has been carried out in a 3D context (volume element) in our work. Hereinafter, we present a 2D example of simulating a blanking test which

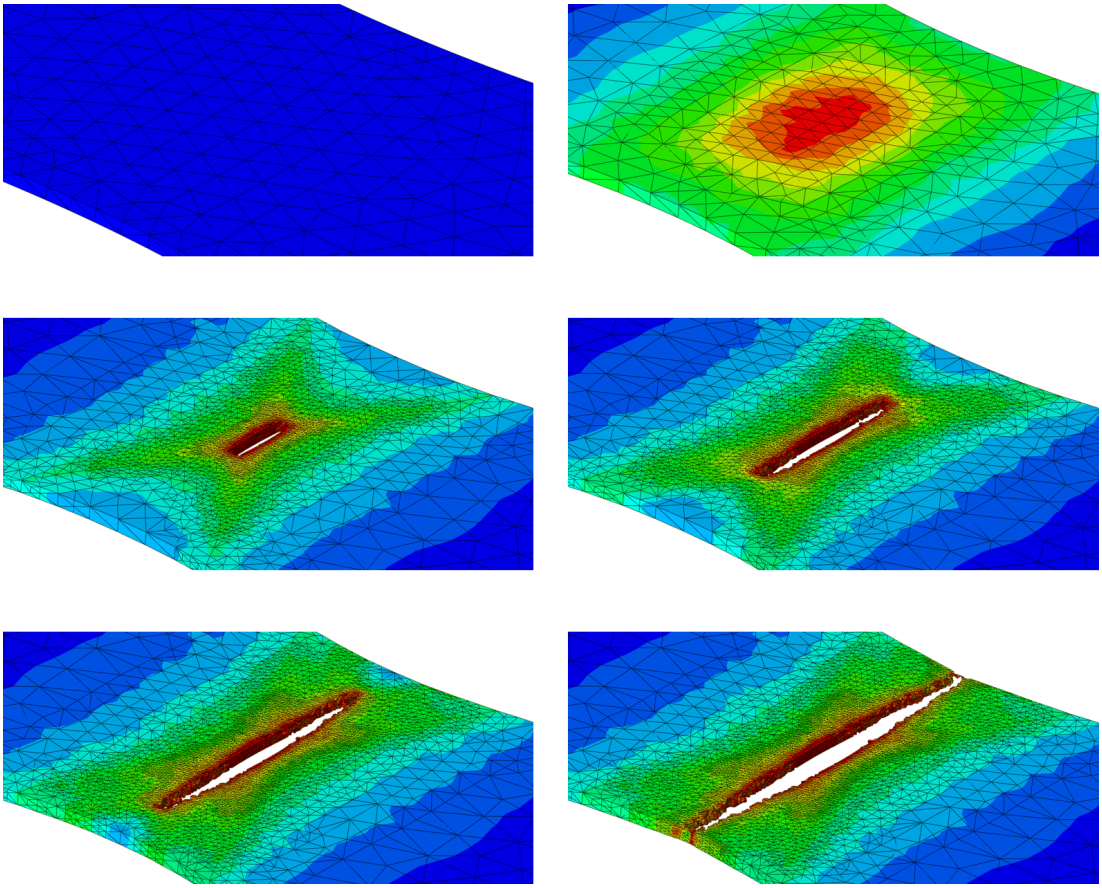


Figure 8-14: The crack initiation and propagation on the specimen

has been done on a circular thin sheet [45]. This 2D example is presented in order to point out the difficulties and the differences that can be found in a 3D context. This most concerns the cost of a global meshing and coarsening problem. The problem is proposed to be solved in perspective part.

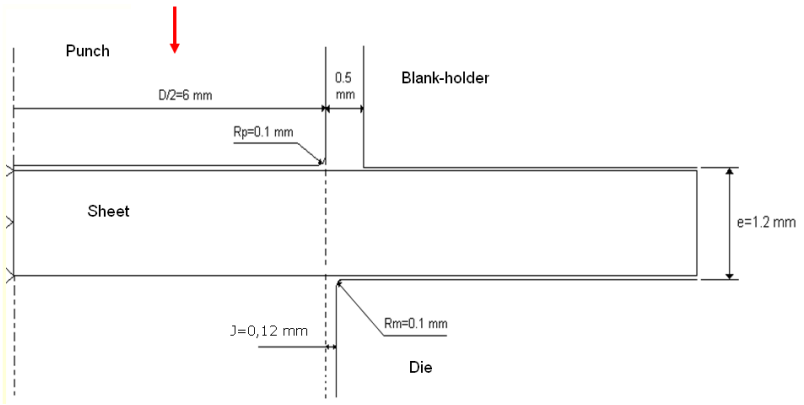


Figure 8-15: The geometry of the cutting tool

In this 2D blanking simulation, the axisymmetric sheet has an initial external radius $R = 200mm$ and a thickness $e = 1.2mm$. Punch diameter is $12mm$ and the clearance between the fixed die and the moving punch is $0.06mm$. The material steel DP600 is used in this simulation and the corresponding material parameters are given in the Table 8.3.

Table 8.3: The values of material parameters for the steel DP600

$E(GPa)$	ν	$\sigma_y(MPa)$	$Q(MPa)$	b	C	a
220	0.29	410	1010	2.0	7000	40.0
S	s	β	Y_0	γ	h	h^d_{min}
22	1.2	2.0	0.8	4.0	0.0	0.05

The Fig. 8-16 shows the spatial discretization of the sheet steel with the quadrangle axisymmetric CAX4R and the refinement in the zones near the contact with the rays matrix and the punch. We can visualize the state of the sheet steel at different displacement U of the punch.When the displacement is $U = 0.439mm$, we can find a plastic flow of the sheet steel without fracture but a highly localized field of cumulative plastic strain in the contact zone with the tool leaving. The size of the elements in these zones are gradual and the refinement is concentrated according to the band of the localization. When the displacement reaches to $0.457mm$, two macro-cracks occur on both sides of the thickness of the specimen which finally meet with each other and result in a completely fractured sheet steel.

During the whole period of the propagation of the two cracks, the size of the minimum mesh size h^d_{min} is concentrately located in the highly damaged zones. Since the cracks propagate, the mesh is automatically coarsened in the inactive zones in order to recover the elements in an acceptable number. In this 2D example, quadrangle element is used to discretize the geometry and a global remeshing process is performed. However, in a 3D context, a global remeshing is not practical in terms of both the precision and the computational cost. Therefore, a new coarsening method is needed to be performed in a 3D context.

The Fig. 8-16 shows a good agreement of the crack surface obtained by numerical model to the experimental micrography of the cutting zone of the DP600 sheet steel.

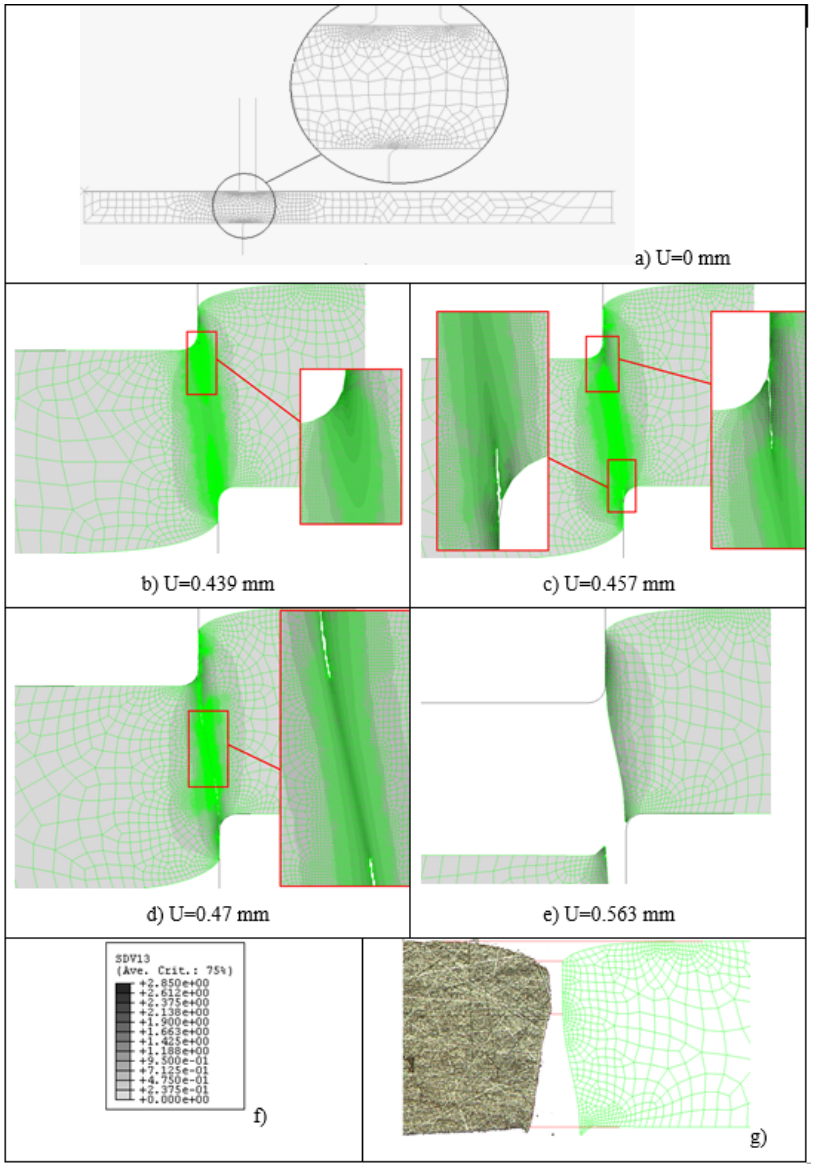


Figure 8-16: Evolution of the mesh and the plastic deformation at different displacement of the punch U and comparison between numerical and experimental fracture surfaces

The numerical crack surface is conform to a profile of the classic cutting behevior — a domed surface, a sheared smooth zone, a torn area and a smudge. The realistic behavior of the numerical model is comfort when we compare Fig. 8-17 the normalized cutting force (F/F_{max}) measured experimentally and calculated numerically. The results are conformed with the crack surfaces and the present cutting force in the literature [67].

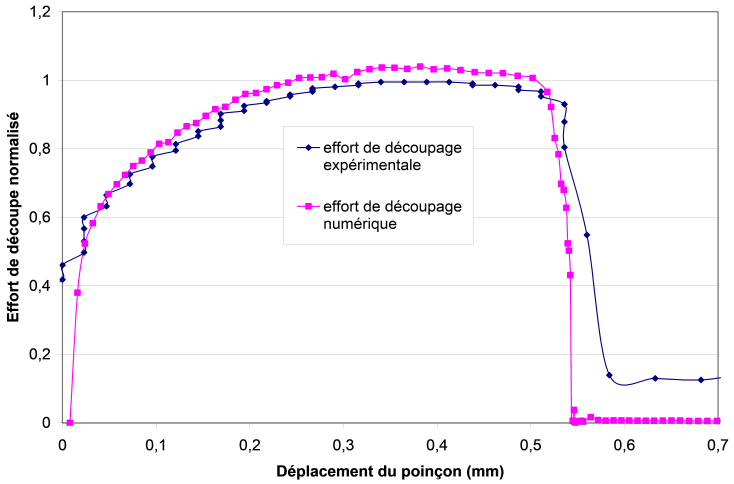


Figure 8-17: The cutting force as function of the punch displacement

Chapter 9

Conclusion and perspectives

A 3D h-adaptive methodology is proposed especially for the simulation of crack initiation and propagation during ductile metal forming processes. Both efficiency and robustness of this methodology are validated with a series of tensile tests in which the location of crack initiation and the path of crack propagation were well observed on the specimens. In addition, the elongation-load curves from the simulation also give a good agreement with the ones from the experiments. From these numerical results, we can also see that the mesh gradation captures the evolution of the plastic flow with accuracy which validates our size indicators.

Our 3D h-adaptive methodology demonstrates that the time spent in the remeshing process which is of concern for most authors is not an important issue here due to a local remeshing. Compared to the CPU time cost by ABAQUS/Explicit® solver, the cost of remeshing process is negligible, a point which is rarely mentioned or even prevent authors from using a standard finite element method. The hybrid field transfer with improved point selection strategy avoids the potential shortcomings (numerical diffusion and peak value decline) occurring after numerous remeshing and field transfer processes, especially in the highly localized zones. In addition, all these problems altogether are not mentionned in the litterature, especially the decline of the peak value. The adaptive length of each loading step is controlled by the rate of plastic strain as well as the severely distorted elements.

Based on the validation of tensile tests, our h-adaptive methodology can be ex-

tended to various 3D metal forming tests, bending, blanking, deep drawing or forging. These metal forming processes may be more complex with respect to the meshing and remeshing process. In processes like blanking or deep drawing, there is relative motion between the contact surface pairs of parts and tools. Therefore, the contact surface pair should be carefully identified in each loading step to ensure that there is no penetration between these two surfaces. In order to achieve this, two aspects should be taken into consideration. On the one hand, the mesh size of the part should be small enough to follow the curvature of the tool, otherwise, the two discretized surfaces would not be into perfect contact with each other. A size indicator based on the curvature of the geometry is needed to guarantee the suitable mesh size. This size indicator can be built as A. Rassineux proposed in 2000 [73]. The curvature of the geometry can be expressed as a function of the defect angle at the surface node of the tool. The referenced mesh size proposed by geometric size indicator can be expressed as a function of the curvature. This referenced size can be given to each triangle on the part surface at their geometry center. The curvature value at the geometry center of part surface triangle can be interpolated by improved diffuse approximation method. This referenced size proposed by geometric size indicator should be considered together with the referenced size proposed by physical size indicator. The simplest way to do this is to choose a final referenced size which is the smaller one among these two sizes. On the other hand, a criterion used to judge whether the penetration occurs should be given. Because of the smoothing process, the two surfaces in the contact pair can be regarded as smoothed surface. As a result, we check all the nodes on the two surfaces to verify whether penetration occurs. All the nodes on the two surfaces should be outside each other and located with a minimum distance from each other.

An error indicator based on the evolution of the plastic strain $\dot{p}$ and the evolution of damage value $\dot{d}$ can be designed in order to identify the optimized parameters of the size indicators in the simulation. The determination of the length of each loading step can also be optimized. For example, the length of each loading step may not only depend on the number of totally damaged elements or the sizes of these

elements but also depends on the evolution of the plastic strain $\dot{p}$ and the evolution of damage value $\dot{d}$. In order to reduce the computational cost, a new linear tetrahedron element inspired from the work of Feulvarch [31] and which avoid locking problems can be developped or used in the context of explicit simulations instead of the present quadratic tetrahedron element.

The proposed elastoplastic model based on local formulation leads to constitutive equations (for elasticity, plasticity, mixed hardening, damage, friction) with material parameters having a clear intrinsic character. The strong coupling between the plastic behavior and the ductile damage leads to an induced softening which generates inevitably mesh-dependent numerical solutions of the initial and boundary value problems (IBVP). In that case, the plastic flow with damage localizes inside narrow (shear) bands having a finite width which is highly mesh sensitive. The simplest way to control this mesh dependency, in this case of a fully local formulation, consists of assuming that all the damage material parameters (S, s, β and Y_0) are identified by imposing a relevant minimal mesh size which provides the best fit of the reference (experimental) curve. This smallest mesh size becomes a kind of intrinsic material parameter. As a consequence, the whole h-adaptive process is controlled by this smallest mesh size imposed in the fully damaged zones. It is worth mentioning that this approximate way to trigger the mesh dependency may be replaced by a more straightforward non-local formulation as the micromorphic models [75]. As our method is quasi-independent to the model, it can be implemented in the non-local formulation model directly without being changed too much. A strong discontinuity method [5] may also be used with our methodology to simulate the crack initiation and propagation.

In terms of the present methodology, the critical zones are only the active zones where the plastic strain accumulates. In these zones, the mesh is refined. In fact, after the crack tip passes through these zones, the plastic strain is constant. In such zones, in most cases, no deformation will occurs so that the mesh will not be distorted or entangled unless in cases in which the crack may close for instance. Therefore, we can coarsen the mesh in such zones. The coarsening process can reduce significantly

the number of the elements in the model. For example, if we double the size of elements in a 3D zone, the number of the elements may be 8 times less. For the plate specimens which have a relative small thickness, the coarsening of the volume mesh can be done by first coarsening the surface mesh and then generating the volume mesh in the surface mesh. The coarsening of the surface mesh can be an iterative process: firstly, the small triangles are deleted from the mesh and the holes caused by removing these triangles are repatched by the optimized surface meshing technique explained in section 6.3.4; secondly, the surface mesh is smoothed so that the quality of the mesh is enhanced; finally, the first two steps are iterated until the mesh size requirements are fulfilled with respect to the input size map. Based on our surface meshing technique, it is better not to delete more than one small elements which are adjacent at the same time so that holes created by deleting these small triangles are not too big. Otherwise, the boundary of the holes is composed of too many segments which leads to a huge number of configurations to be considered when we generate surface mesh in the hole and the huge number of configurations costs too much. Since the mesh refinement causes a mesh gradation within the geometry, we could not only coarsen the zones with minimum mesh size. The main goal of the mesh coarsening is mostly to avoid a full remeshing or at least a limited number of times. The determination of the coarsening zones and the size indicator for coarsening the mesh should be made in order to coarsen the mesh automatically at each loading step.

Appendix A

APPENDICES

A.1 Modelling of elasto-plastic constitutive equation coupled with isotropic damage

The advanced elasto-plastic behavior model fully coupled with ductile damage in large deformation is introduced, which has been used and developed in several theses of LASMIS (citer des papiers du LASMIS). This adaptive 3D numerical metal forming methodology is based on the ductile damage at finite strain and its effect on the other mechanical fields.

A.1.1 Mathematical Description of finite transformations

In the beginning, the mathematical description of the finite transformations is illustrated briefly, which consider the continuum deformation gradient, introduce a discrete version of the deformation gradient, and finally present the strain tensors. For simplicity, we assume a fixed Cartesian coordinate system and a continuous body composed of particles that are identified by the position vectors which specify their place in the initial configuration C_0. A motion maps a particle from its initial position $\overrightarrow{X}$ in the reference configuration C_0, to its position $\overrightarrow{x}$ in the current configuration C_t as shown in Fig. A-1.

The deformation includes volume changes, shape changes and rigid body motions.

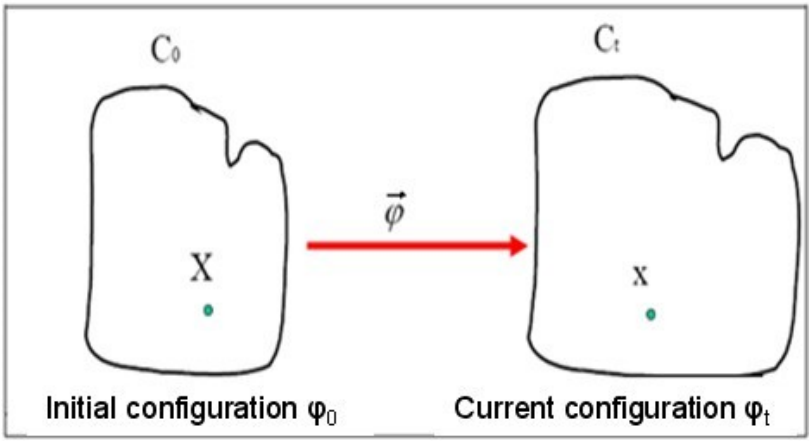


Figure A-1: Illustration of the solid deformation

A deformation is the relationship between two configurations and is usually denoted by $\overrightarrow{\varphi}$:

$$\overrightarrow{\varphi}(\overrightarrow{X}) = \overrightarrow{x}(\overrightarrow{X}, t) \tag{A.1}$$

The displacement vector and the velocity vector for each material point are usually denoted by Eq. A.2 and Eq. A.3:

$$\overrightarrow{u} = \overrightarrow{x} - \overrightarrow{X} \tag{A.2}$$

$$\overrightarrow{v} = \frac{\partial \overrightarrow{u}}{\partial t} = \frac{\partial \overrightarrow{x}}{\partial t} \tag{A.3}$$

For quantifying the change of material point in a solid body, the relationship between $d\partial\overrightarrow{X}$ and $d\partial\overrightarrow{x}$ is defined as follows:

$$d\partial\overrightarrow{x} = \frac{\partial \overrightarrow{\varphi}(\overrightarrow{X}, t)}{\partial \overrightarrow{X}} d\partial\overrightarrow{X} \tag{A.4}$$

where the deformation gradient tensor $\underline{F}$ is given by:

$$\underline{F} = \frac{\partial \overrightarrow{\varphi}(\overrightarrow{X}, t)}{\partial \overrightarrow{X}} = \underline{1} + \frac{\partial \overrightarrow{u}}{\partial \overrightarrow{X}} \tag{A.5}$$

For any physically admissible deformation, the volume of the deformed element

must be positive (no matter how much you deform a solid, you canâĂŹt make material disappear). Therefore, all physically admissible displacement fields must satisfy the conditions in Eq. (3-6):

$$J = \det \underline{F} > 0 \tag{A.6}$$

where J is the measure of the volume change which generates by the deformation.

In order to measure the rate of deformation, the velocity gradient $\underline{L}$ in configuration C_t is introduced and can be expressed in terms of the deformation gradient:

$$\underline{L} = \frac{\partial \overrightarrow{v}}{\partial \overrightarrow{x}} = \frac{\partial \overrightarrow{v}}{d\overrightarrow{X}} \cdot \frac{d\overrightarrow{X}}{\partial \overrightarrow{x}} = \dot{\underline{F}} \cdot \underline{F}^{-1} \tag{A.7}$$

The velocity gradient $\underline{L}$ can be decomposed into the stretch rate tensor D and the spin tensor $\underline{\Omega}$ as follow:

$$\underline{D} = \underline{L}^S = \frac{1}{2}(\underline{L} + \underline{L}^T) \tag{A.8}$$

$$\underline{\Omega} = \underline{L}^A = \frac{1}{2}(\underline{L} - \underline{L}^T) \tag{A.9}$$

To quantify the interaction force between two material points, the Cauchy stress tensor is introduced and denoted as $\underline{\sigma}$. The other stress tensor can be defined in C_t which through the Kirchhoff tensor as in Eq. A.10.

$$\underline{\tau} = J\underline{\sigma} \tag{A.10}$$

with:

$$J = \det \underline{F} = \frac{\rho_0}{\rho} \tag{A.11}$$

where ρ_0 and ρ represent separately the mass density of the material at a point in the current configuration and in the initial configuration.

Make $d\overrightarrow{f}$ as the force on the element of surface dS in the current configuration, we have Eq. A.12:

$$d\vec{f} = \underline{\sigma} \cdot \vec{n}\, dS \tag{A.12}$$

where $\vec{n}$ denotes a unit vector normal to the surface dS.

A.1.2 Total Strain rate decomposition

In finite deformation mechanics, it is useful to assume that the total transformation gradient F is multiplicatively decomposed into elastic (recoverable) and plastic (irreversible) parts giving as:

$$\vec{F} = \vec{F}^e \cdot \vec{F}^p \tag{A.13}$$

where $\vec{F}^e$ is the gradient of elastic transformation and $\vec{F}^p$ is the gradient of plastic transformation.

Lee [48] introduced the notion of relaxed configuration C_r which is between the initial configuration C_0 and the current configuration C_t in order to decompose the total transformation into plastic and elastic parts as shown in Fig. A-2.

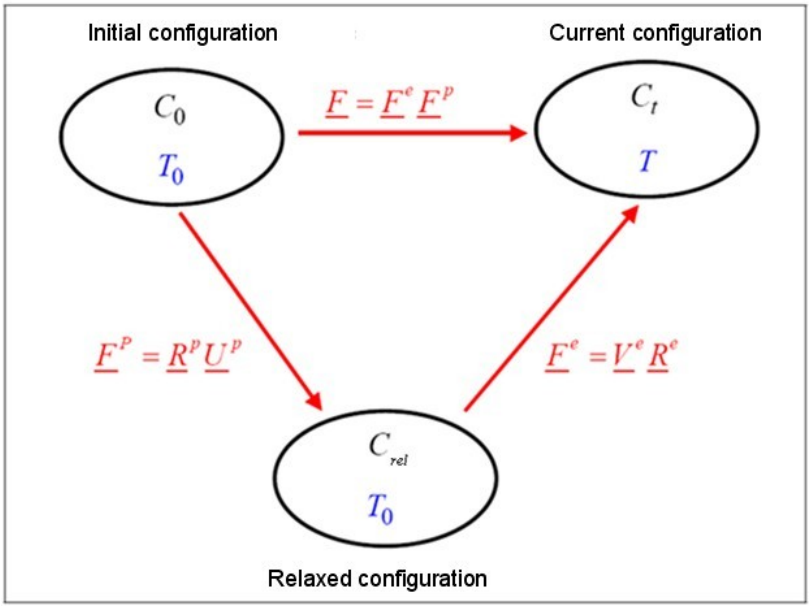


Figure A-2: Decomposition of total transformation gradient

It is worth noting that the fully coupled constitutive equations derived above, are

valid under the small strain assumption. To be used in metal forming by large visco-plastic deformations, they should be extended to the finite strain framework in such a manner that the objectivity requirement is fulfilled. This is made using an updated Lagrangian description for which the total small strain rate tensor $\underline{\dot{\varepsilon}}$ transforms to its Eulerian counterpart $\underline{D}$ defined as a symmetric part of the Eulerian velocity gradient tensor. If the elastic part of the total strain tensor is assumed very small compared to the plastic one, the total strain rate can be approximated by the following additive decomposition:

$$\underline{D} = \underline{\dot{\varepsilon}}_e^J + \underline{\dot{\varepsilon}}^p \tag{A.14}$$

in which $\underline{\dot{\varepsilon}}_e^J$ stands for the Jaumann derivative of the small elastic strain tensor and $\underline{\dot{\varepsilon}}^p$ represents the finite plastic strain tensor. To simplify the notation, in the following, the elastic strain is expressed by $\underline{\varepsilon}^e$ and the plastic strain is $\underline{\varepsilon}^p$ where $\underline{\varepsilon}^p = \int \underline{\dot{\varepsilon}}^p dt$.

A.1.3 Principle of indifference and material objectivity

On the other hand, to fulfill the objectivity requirement at finite strain, the rotational objective rates are used to calculate the derivatives of any tensorial variable. The objectivity requirement is ensured by using the so called rotational frame formulation [!!Dougui 1989!!]. It leads to express the constitutive equations in a rotated, deformed and damaged (Eulerian) configuration obtained from the current one by the orthogonal rotation tensor $\underline{Q}$, solution of the following kinematic equation:

$$\underline{T}^Q = \underline{Q}^T \cdot \underline{T} \cdot \underline{Q} \tag{A.15}$$

in which $\underline{Q}$ is the orthogonal rotation tensor, solution of the following "kinematical" constitutive equation:

$$\underline{W}_Q = \dot{\underline{Q}}^T \cdot \underline{Q}^T \quad \text{and} \quad \underline{Q}(t=0) = \underline{1} \tag{A.16}$$

where $\underline{W}_Q$ is the rotation rate tensor which can be chosen in order to obtain some classical rotational objective derivatives as Jaumann, or Green-Naghdi derivation [Dougui 1989]. This rotated frame formulation keeps unchanged the basic structure of the constitutive equations as formulated in small strain hypothesis while ensuring the objectivity requirement. Using this "rotated" objective formulation a complete set of constitutive equations can be formulated for metal forming simulation.

A.2 Modelling of material behaviour with damage

Damage in materials can be defined as an effective surface density of microcracks or microcavities present in any plane of a representative volume element. Damage, in the context of continuum damage mechanics (CDM), is a thermodynamic state variable. It accounts for the progressive deterioration of the load-carrying capacity of the material. The damage process develops in material microstructure as result of micro-void formation. The understanding and measurement of the basic damage process in ductile materials is important to predict failure of material behaviour ([50] [59] [20] [18] [77]).

A.2.1 State and effective state variables

A.2.1.1 State variables

Modelling in the thermodynamics structure requires the introduction of state variables. These variables could be divided into observable variables and internal variables. Observables are the total strain $\underline{\varepsilon}$ and temperature T. Internal variables represent the different dissipative phenomena to be modelled as isotropic hardening, kinematic hardening, heat flow and damage. So r represents the isotropic hardening strain, r_c is isotropic dynamic recrystallisation strain, $\underline{\alpha}$ is kinematic hardening strain, d is the isotopic damage, s is entropy and $\overrightarrow{q}$ is heat flux vector. All observable and internal variables are summarized in Table 3-1 with their associated variables.

Table A.1: Experimentally determined ductile failure parameters

State variables		Associated variables
Observable	Internal	
Total strain $\underline{\varepsilon}$	Elasticity $\underline{\varepsilon}^e$	$\underline{\sigma}$
	Isotropic hardening r	R
	Isotropic damage d	Y

A.2.1.2 Effective variables

The damage can not be measured directly, so it is only be tested by its effect on the mechanical behaviour of material. To describe the effect of damage behaviour, effective variables should be used ([19] [50] [77] [76]).

These state variables are defined according to three hypotheses. The first assumes strain equivalence [19]; the second implies the stress equivalence [34] and the third involves the elastic energy equivalence [22]. The elastic energy equivalence can affect not only the stress though damage but also elastic strain. Later, this theory is introduced into total energy equivalence [77] which can apply to damage of elastic and inelastic material behaviour.

For an EVR containing micro-crack damage is opened in tension state and even leads to a failure. If now the same EVR is loaded in compression, due to a reversal of the stress, micro-cracks can close gradually until to a total closure if the solicitation of compression is sufficient. In this case the modulus of elasticity and hardening could be able to restore to their initial values before damage gradually. This recovery effect of physical properties after closure of micro-cracks is called quasi-unilateral effect ([47] [52] [24] [66]). With an isotropic damage, we can define effective variables in unilateral condition following Ladevèze and Lemaitre approach. For all tensor variable $\underline{T}$ written in its principal reference, we define one positive part $< \underline{T} >_+$ and one negative part $< \underline{T} >_-$ as $\underline{T} =< \underline{T} >_+ + < \underline{T} >_-$ and trace $\underline{T} : \underline{1} =< \underline{T} : \underline{1} >_+ + < \underline{T} : \underline{1} >_-$ or $< \underline{T} : \underline{1} >_+$ and $< \underline{T} : \underline{1} >_-$ are positive part and negative part of trace $\underline{T}$ with:

$$< \underline{T} : \underline{1} >_+ = \begin{cases} \underline{T} : \underline{1} & \text{if} \quad \underline{T} : \underline{1} > 0 \\ 0 & \text{if} \quad \underline{T} : \underline{1} \leq 0 \end{cases} \tag{A.17}$$

Eq. A.17 expresses the positive part of the trace $\underline{T}$, when this $\underline{T}$ is strictly positive. Similarly, we introduce:

$$< \underline{T} : \underline{1} >_- = \begin{cases} \underline{T} : \underline{1} & \text{if} \quad \underline{T} : \underline{1} \leq 0 \\ 0 & \text{if} \quad \underline{T} : \underline{1} > 0 \end{cases} \tag{A.18}$$

Eq. A.18 expresses the negative part of the trace $\underline{T}$, when this $\underline{T}$ is strictly negative.

According to the theory of energy equivalence with general relations of isotropic damage between the state variables and the effective variables, applying the decompositions above to the stress tensor and elastic strain tensor, the following equations can be obtained:

$$< \tilde{\underline{\sigma}} >_+ = \frac{< \underline{\sigma} >_+}{\sqrt{1-d}} \qquad < \tilde{\underline{\sigma}} >_- = \frac{< \underline{\sigma} >_-}{\sqrt{1-hd}} \tag{A.19}$$

$$< \tilde{\underline{\sigma}} : \underline{1} >_+ = \frac{< \underline{\sigma} : \underline{1} >_+}{\sqrt{1-d}} \qquad < \tilde{\underline{\sigma}} : \underline{1} >_- = \frac{< \underline{\sigma} : \underline{1} >_-}{\sqrt{1-hd}} \tag{A.20}$$

$$< \tilde{\underline{\varepsilon^e}} >_+ = < \underline{\varepsilon}^e >_+ \sqrt{1-d} \qquad < \tilde{\underline{\varepsilon^e}} >_- = < \underline{\varepsilon}^e >_- \sqrt{1-d} \tag{A.21}$$

$$< \tilde{\underline{\varepsilon^e}} : \underline{1} >_+ = < \underline{\varepsilon^e} : \underline{1} >_+ \sqrt{1-d} \qquad < \tilde{\underline{\varepsilon^e}} : \underline{1} >_- = < \underline{\varepsilon}^e : \underline{1} >_- \sqrt{1-d} \tag{A.22}$$

$$\begin{cases} \tilde{R} = \frac{R}{\sqrt{1-d^\gamma}} \\ \tilde{r} = r\sqrt{1-d^\gamma} \end{cases} \tag{A.23}$$

The parameter $h (0 \leq h \leq 1)$ is a constant which characterizes the effect of the closure of micro-cracks in a loading state of compression.

A.2.2 State potential and state relations

After defining the state variables and the effective variables, we can define the potential state and deduce the state relations. The Helmholtz free energy is used as a state potential in the strain space. Written under additive form of a elastic and an inelastic term, this free energy is written assuming:

$$\rho\psi(\underline{\varepsilon}^e, r, \underline{\alpha}; d) = \rho\psi_{el}(\underline{\varepsilon}^e; d) + \rho\psi_{inel}(r, d) \tag{A.24}$$

Then the thermoelastic part of the Helmholtz energy is written as:

$$\begin{array}{rl} \rho\psi_{el}(\underline{\varepsilon}^e, T, d) = & \frac{\nu E(1-d)}{2(1+\nu)(1-2\nu)} < \underline{\varepsilon}^e : \underline{1} >_+^2 + \frac{E(1-d)}{21+\nu} < \underline{\varepsilon}^e >_+ < \underline{\varepsilon}^e >_+ \\ & \frac{\nu E(1-d)}{2(1+\nu)(1-2\nu)} < \underline{\varepsilon}^e : \underline{1} >_-^2 + \frac{E(1-d)}{21+\nu} < \underline{\varepsilon}^e >_- < \underline{\varepsilon}^e >_- \end{array} \tag{A.25}$$

Concerning the inelastic part it is given by:

$$\rho\psi_{inel} = \frac{1}{2}(1 - d^{\omega})Qr^2 \tag{A.26}$$

where Q is the isotopic hardening modulus and C is the kinematic hardening modulus.

From the state potential defined above, we can easily derive the state relations as following:

$$\underline{\sigma} = \rho\frac{\partial\psi}{\partial\underline{\varepsilon}^e} = < \underline{\sigma} >_+ + < \underline{\sigma} >_- \tag{A.27}$$

with:

$$\left\{ \begin{array}{l} < \underline{\sigma} >_+ = \frac{\nu E(1-d)}{(1+\nu)(1-2\nu)} < \underline{\varepsilon}^e : \underline{1} >_+ < \underline{1} >_+ + \frac{E(1-d)}{(1+\nu)} < \underline{\varepsilon}^e >_+ \\ < \underline{\sigma} >_- = \frac{\nu E(1-d)}{(1+\nu)(1-2\nu)} < \underline{\varepsilon}^e : \underline{1} >_- < \underline{1} >_- + \frac{E(1-d)}{(1+\nu)} < \underline{\varepsilon}^e >_- \end{array} \right. \tag{A.28}$$

$$R = \rho\frac{\partial\psi}{\partial r} = Q(1 - d^{\omega})r \tag{A.29}$$

$$\begin{cases} Y(\underline{\varepsilon}^e, r, \underline{\alpha}, d) = -\rho\frac{\partial\psi}{\partial d} = Y_e + Y_r \\ Y_e = < Y_e >_+ + < Y_e >_- \end{cases} \tag{A.30}$$

with:

$$\begin{cases} < Y_e >_+ = \frac{\nu E}{2(1+\nu)(1-2\nu)} < \underline{\varepsilon}^e : \underline{1} >_+^2 + \frac{E}{2(1+\nu)} < \underline{\varepsilon}^e >_+ : < \underline{\varepsilon}^e >_+ \\ < Y_e >_- = \frac{\nu E}{2(1+\nu)(1-2\nu)} < \underline{\varepsilon}^e : \underline{1} >_-^2 + \frac{E}{2(1+\nu)} < \underline{\varepsilon}^e >_- : < \underline{\varepsilon}^e >_- \\ Y_r = \gamma d^{(\gamma-1)}\frac{1}{2}Qr^2 \end{cases} \tag{A.31}$$

where Y_e is the elastic part of the damage energy release rate, Y_r is isotropic hardening part.

A.2.3 Time-independent intrinsic dissipation

The framework of the so called Generalised Standard Material [32] is used to analyse the mechanical dissipation. The von Mises yield function and pseudo pasltic dissipation potentials are introduced [77] [78] [81] [83] [46] [37] [82]:

$$f(\underline{\sigma}, R; d) = \frac{||\underline{\sigma} - \underline{X}||}{\sqrt{1-d}} - \frac{R}{\sqrt{1-d^\gamma}} - \sigma_y \leq 0 \tag{A.32}$$

$$\begin{aligned} F(\underline{\sigma}, R, d) = \quad & f + \frac{1}{2}\frac{bR^2}{Q(1-d^\gamma)} \\ & + \frac{S}{(S+1)(1-d)^\beta} < \frac{Y-Y_0}{S} >^{S+1} \end{aligned} \tag{A.33}$$

where, $||\underline{\sigma}|| = \sqrt{(3/2)\underline{\sigma}^{dev} : \underline{\sigma}^{dev}}$ defines the well-known von Mises equivalent stress in which $\underline{\sigma}^{dev}$ is the deviatoric part of the stress tensor. The parameters a and b reflect the nonlinear property of the kinematic and isotropic hardening respectively and S, s and β characterize the nonlinear evolution of the ductile damage. Applying the generalized normality rule to the above defined local yield function and dissipation potential leads to the following evolution equation:

$$\underline{\dot{D}}^p = \dot{\lambda}\frac{\partial F}{\partial \underline{\sigma}} = \dot{\lambda}\underline{\tilde{n}} = \dot{\lambda}\frac{\underline{n}}{\sqrt{1-d}} \tag{A.34}$$

$$\dot{r} = -\dot{\lambda}\frac{\partial F}{\partial R} = \dot{\lambda}\left(1 - \frac{bR}{Q(1-d^{\omega})}\right) = \dot{\lambda}\left(\frac{1}{\sqrt{1-d^{\omega}}} - br\right) \tag{A.35}$$

$$\dot{d} = \dot{\lambda}\frac{\partial F}{\partial Y^{*}} = \frac{\dot{\lambda}}{(1-d)^{\beta}} < \frac{Y - Y_0}{S} >^{S} \tag{A.36}$$

where, $\underline{n}3(\underline{\sigma}^{dev}\underline{X})/2||\underline{\sigma} - \underline{X}||$ is the outside normal to the yield surface in the stress space and $\dot{\lambda}$ is the plastic multiplier. The plastic multiplier $\dot{\lambda}$ appearing in all the evolution equations will be deduced from the consistency condition associated to the yield function Eq. A.32: $\dot{f}(\underline{\sigma}, \underline{X}, R; d) = 0$

A.3 Models of contact and friction between surfaces

Then, in this paragraph the contact problem with friction between solids in the modelling of process will be illustrated. It is well known, that the friction contact between the billet and the die plays an important role in forging process. The modeling of the contact with friction under mechanical conditions is not a very easy task due to the wears affecting deeply the friction condition at the interface. The contact with friction is simply described according to the very classical Coulomb model available in ABAQUS.

A.3.1 Coulomb friction model

The Coulomb friction model is used for most contact problems with friction. This model is based on the concept of threshold and the intervention of the contact pressure. The Coulomb's friction model is defined by Eq. (3-58):

$$\begin{array}{lllll} \text{Sticking contact :} & \tau < \mu p & \Rightarrow & V_g = 0 & \\ \text{Sliding contact :} & \tau = \mu p & \Rightarrow & \exists \lambda \geq 0, & V_g = -\lambda\tau \\ \text{Impossible contact :} & \tau > \mu p & \Rightarrow & \textit{impossible} & \end{array} \tag{A.37}$$

where μ is the friction coefficient which is between 0.01 (thick film lubrication) and 0.5 (without lubrication: dry contact); τ is the equivalent tangential stress; v_g is the

relative tangential velocity at the contact point lying on the contact boundary and p is the normal contact pressure. Fig. A-3 is the graphic representation of Coulomb contact model.

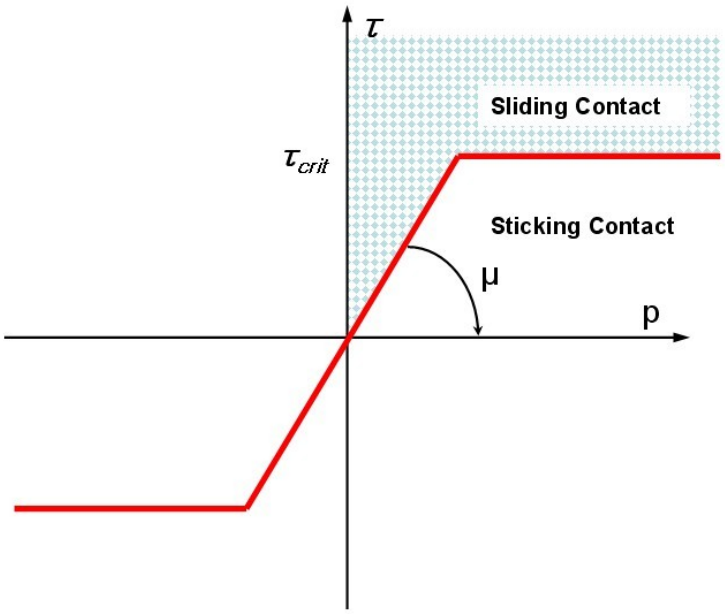


Figure A-3: Representation of Coulomb contact model

A.3.2 Numerical aspect and adaptive FE methodology

In this section, the fully coupled elasto-plastic constitutive equations presented previously are implemented into Abaqus/Explicit FE code using the Vumat user subroutine. The global resolution scheme is Dynamic Explicit and it is associated with an automatic time step control. The computation of the stress tensor as well as the state variables is worked out by numerical integration of the fully coupled damage/plasticity equations.

A.3.2.1 Global resolution scheme

The problem proposed above is defined by a classical variational problems of the equilibrium problem. By using the displacement based (Galerkin) finite element

method, for each element (e) in its current (deformed, heated and damaged) configuration with volume V and boundaries Γ, can be defined:

$$I_M^e\left(\{u_{nod}^e\},\{\delta u_{nod}^e\}\right) = \langle [M^e]\{\ddot{u}_{nod}^e\} - \{F_{int}^e\} + \{F_{ext}^e\}\rangle \{\delta \dot{u}_{nod}^e\} \tag{A.38}$$

with:

$$\begin{cases} \{F_{ext}^e\} = & \int_{V^e} [N^e]^T \{f_d^e\}\, dV + \int_{S_f^e} [N^e]^T \{t^e\}\, dV + \int_{S_c^e} [N^e]^T \{t_c^e\}\, dV \\ \{F_{int}^e\} = & \int_{V^e} [B^e]^T \{\sigma\}\, dV \end{cases} \tag{A.39}$$

where $[N^e]$ is the matrix of interpolate shape displacement function of the element 'e'; $\{u_{nod}^e\}$ is the vector of nodal elementary displacement; $\{\delta u_{nod}^e\}$ is the virtual nodal elementary displacement; $[M^e]$ is the mass matrices of element (e):

$$[M^e] = \int_{V^e} \rho\, [N^e]^T [N^e]\, dV \tag{A.40}$$

To simplify the resolution scheme, the elementary mass matrix $[M^e]$ is generally diagonalised by using a lumped mass matrix algorithm. $[B^e]$ is the matrix of interpolation for the total strain tensor:

$$[B^e] = \frac{\partial [N^e]}{\partial \underline{x}} \tag{A.41}$$

By using a classical highly nonlinear algebraic system expressing the dynamical equilibrium of the work piece and tools at each time step, Eq. A.38 can be easily written as the following:

$$[M]\{\ddot{u}\} - \{F_{ext}\} + \{F_{int}\} = \{0\} \tag{A.42}$$

The system (Eq. A.42) is solved successively over each time step $t_n, t_{n+1} = t_n + \Delta t$ as following: First, the mechanical equation (3-66.a) is solved at constant temperature T_n thanks to the explicit scheme available in ABAQUS. The displacement field is obtained as well as the stress and state variables at t_{n+1}. Second, the thermal equation

(3-66.b) is solved with respect to the temperature in order to obtain T_{n+1} knowing all the mechanical fields at t_{n+1}. The solution of the IBVP by using the element presented above is performed using the dynamic explicit resolution scheme in a standard way in ABAQUS/Explicit. The Eq.((39)a) that gives the stress and state variables at t_{n+1} and the Eq.((39)b) are solved in parallel for each time increment $[t_n, t_{n+1}]$ in order to compute the micromorphic damage variable at t_{n+1}. Note that the stability of the dynamic explicit solution is governed by the following stability condition: $\Delta t \leq \frac{2}{\omega_{max}}(\sqrt{1+\xi^2}-\xi)$ in which ω_{max} is the highest eigenvalue of the mechanical system and $\xi \leq 1$ is a damping parameter. An approximation of the stability time limit for the solution of the mechanical problem is given by: $\Delta t \leq \frac{2}{\omega^e_{max}}$ where ω^e_{max} is the biggest elementary eigenvalue. The stability time limit is defined as follows and can be written under the form $\Delta t \leq \min(\frac{L^e}{C_d})$ with L^e being the size of the element and C_d is the velocity of a solid wave passing through the element and is given by $C_d = \sqrt{\frac{\lambda_e+2\mu_e}{\rho}}$ so that $\Delta t \leq \min\left(L^e\sqrt{\frac{\rho}{\lambda_e+2\mu_e}}\right)$. For the same reasons, the stability for the solution is also governed by a time step defined as: $\Delta t \leq \frac{2}{\omega^{\hat{d}}_{max}}$ where $\omega^{\hat{d}}_{max}$ is the biggest eigenvalue of the micromorphic problem regarding the nonlocal damage. As every problem has a critical time step, for assuring the stability of the IBVP we define the critical time step as follows: $\Delta t \leq \min(2/\omega_{max}, 2/\omega^{\hat{d}}_{max})$.

A.4 Local integration scheme

The computation of $\{F^e_{int}\}$ (see Eq. A.39) needs the calculation of the stress tensor at the end of the time step. This passes through the integration of the overall set of constitutive equations developed above. They are fully coupled non-linear ODEs (Ordinary Differential Equations) to be linearized by using appropriated time discretization scheme. In this work, a fully implicit return mapping algorithm is used together with an exponential (asymptotic) development applied to the hardening equations.

Moreover, special developments have been made in this fully isotropic case, in order to reduce the ODEs number to only two scalar equations coming from the time

discretization of the yield function (Eq. A.43a) and the damage equation (Eq.A.43b) [79]:

$$\begin{cases} f(\Delta\lambda, d_{n+1}) = & ||\underline{Z}^*_{n+1}||_M - \frac{1-d_n}{\sqrt{1-d_{n+1}}} \\ & \left[\frac{Q}{b}(1-e^{-b\Delta\lambda}) + \frac{\sqrt{1-d^\omega_{n+1}}}{1-d^\omega_n} R e^{-b\Delta\lambda}\right. \\ & \left. + \frac{3E}{2(1+\nu)}\Delta\lambda + \frac{C}{a}(1-e^{-a\Delta\lambda}) + \sigma_y\right] \quad \text{(A.43a)} \\ g(\Delta\lambda, d_{n+1}) = d_{n+1} - d_n - \Delta\lambda Y^*_{n+1} \quad \text{(A.43b)} \end{cases}$$

in which the deviatoric tensor Z^* is expressed as a function of the trial stress tensor $\underline{\sigma}^*$:

$$\underline{Z^*_{n+1}} = \left(\underline{\sigma}^{*dev}_{n+1}\right) \tag{A.44}$$

However, in order to save the CPU time from engineering point of view, the damage equation can be approximated by the following explicit formula:

$$d_{n+1} = d_n + \frac{\Delta\lambda}{(1-d_n)^\beta}\left(\frac{Y_{n+1}(\Delta\lambda, d_n) - Y_0}{S}\right) \tag{A.45}$$

Accordingly, only the Eq. A.43a is solved iteratively thanks to the Newton method in order to compute the single unknown $\Delta\lambda$.

Appendix B

APPENDICES

An elastoplastic model integrated with damage is proposed by ABAQUS [1] in which Hooputra et al. [36] have shown that the extruded aluminum alloy EN AW-7108 T6 displays plastic orthotropy due to the nature of the extrusion processing and have used the Barlat symmetric yield locus (Barlat et al., 1991) to fit the experimental data. In this example we neglect the orthotropy and assume both the elastic and the plastic behavior to be isotropic with the yield surface described by the Mises yield function. The assumption of isotropic plasticity may appear to be too restrictive for the accurate prediction of failure in extruded alloys. However, in crashworthiness simulations the assumption of isotropy usually yields acceptable results when compared with experimental observations.

Two main mechanisms can cause the fracture of a ductile metal: ductile fracture due to the nucleation, growth, and coalescence of voids; and shear fracture due to shear band localization. Based on phenomenological observations, these two mechanisms call for different forms of the criteria for the onset of damage [36].

The ductile criterion is a phenomenological model for predicting the onset of damage due to nucleation, growth, and coalescence of voids. The model assumes that the equivalent plastic strain at the onset of damage $\bar{\varepsilon}_d^{pl}$, is a function of stress triaxiality and strain rate: $\bar{\varepsilon}_d^{pl}(\eta, \dot{\bar{\varepsilon}}^{pl})$ where $\eta = -p/q$ is the stress triaxiality, p is the pressure stress, q is the Mises equivalent stress, and $\dot{\bar{\varepsilon}}^{pl}$ is the equivalent plastic strain rate. The criterion for damage initiation is met when the following condition

is satisfied:

$$\omega_d = \int \frac{\mathrm{d}\bar{\varepsilon}^{pl}}{\bar{\varepsilon}_d^{pl}(\eta, \dot{\bar{\varepsilon}}^{pl})} = 1 \tag{B.1}$$

where ω_d is a state variable that increases monotonically with plastic deformation. At each increment during the analysis the incremental increase in ω_d is computed as following:

$$\omega_d = \frac{\Delta\bar{\varepsilon}^{pl}}{\bar{\varepsilon}^{pl}(\eta, \dot{\bar{\varepsilon}}^{pl})} \geq 0 \tag{B.2}$$

For the ductile damage initiation criterion the equivalent plastic strain is given by the following function of the stress triaxiality η:

$$\bar{\varepsilon}_d^{pl}(\eta, \dot{\bar{\varepsilon}}^{pl}) = \frac{\varepsilon_T^+ \sinh[k_0(\eta^- - \eta)] + \varepsilon_T^- \sinh[k_0(\eta - \eta^+)]}{\sinh[k_0(\eta^- - \eta^+)]} \tag{B.3}$$

where ε_T^+ and ε_T^- correspond to the equivalent plastic strain at ductile damage initiation for equibiaxial tensile and equibiaxial compressive deformation, respectively. For isotropic materials the stress triaxiality in equibiaxial tensile deformation state η^+, is $2/3$, and in equibiaxial compressive deformation state η^-, is $-2/3$. The definition of η in Abaqus, as a ratio of the equivalent mean stress to the Mises equivalent stress, differs from that used by [36] by a factor of $1/3$. Consequently, the value of k_0 used in the above expression is three times the value used in [36]. The above expression has three parameters that must be obtained experimentally: ε_T^+, ε_T^-, and k_0. These parameters depend on the material, strain rate, and possibly the temperature. For each strain rate of interest, three experiments are needed at different values of stress triaxiality to obtain the three material parameters. ε_T^+ can be obtained directly from the Erichsen test ($\eta = \eta^+$). Three-point bending of sheet coupons (with width/thickness > 4) under plane strain tension ($\eta = 1/\sqrt{3}$) and fracture at the notch root of waisted tensile coupons in uniaxial tension ($\eta = 1/3$) may provide two additional experiments to determine ε_T^- and k_0. In the Erichsen and three-point bending experiments the local fracture strain can be derived by placing a grid on the specimen's surface; in

the waisted tensile experiment the fracture strain can be obtained from the sheet thickness in the fracture plane [36]. For the aluminum alloy used in this example the experimentally obtained ductile failure parameters at quasi-static and dynamic strain rates ($250s^{-1}$) are listed in Table B.1. The other material parameters are given by Hooputra et al. [36].

Table B.1: Experimentally determined ductile failure parameters

Parameter	Dynamic ($\dot{\varepsilon} = 250s^{-1}$)
ε_T^+	0.44
ε_T^-	1494.0
k_0	8.6304

The shear criterion is a phenomenological model for predicting the onset of damage due to shear band localization. The model assumes that the equivalent plastic strain at the onset of damage, $\underline{\varepsilon}_S^{pl}$, is a function of the shear stress ratio and strain rate: $\underline{\varepsilon}_S^{pl}(\theta_S, \dot{\underline{\varepsilon}}^{pl})$. Here $\theta_S = (q + k_s p)/\tau_{max}$ is the shear stress ratio, τ_{max} is the maximum shear stress, and k_s is a material parameter. A typical value of k_S for aluminum is $k_S = 0.3$ [36]. The criterion for damage initiation is met when the following condition is satisfied:

$$\omega_S = \int \frac{\mathrm{d}\overline{\varepsilon}^{pl}}{\overline{\varepsilon}_S^{pl}(\theta_S, \dot{\overline{\varepsilon}}^{pl})} \tag{B.4}$$

where ω_S is a state variable that increases monotonically with plastic deformation. At each increment during the analysis the incremental increase in ω_S is computed as following:

$$\omega_S = \frac{\mathrm{d}\overline{\varepsilon}^{pl}}{\overline{\varepsilon}_S^{pl}(\theta_S, \dot{\overline{\varepsilon}}^{pl})} \geq 0. \tag{B.5}$$

For the shear damage initiation criterion the equivalent plastic strain at the onset of damage is given by the following function of the shear stress ratio, θ_S [36]:

$$\underline{\varepsilon}_S^{pl}(\theta_S, \dot{\underline{\varepsilon}}^{pl}) = \frac{\varepsilon_S^+ \sinh[f(\theta_S - \theta_S^-)] + \varepsilon_S^- \sinh[f(\theta_S^+ - \theta_S)]}{\sinh[f(\theta_S^+ - \theta_S^-)]} \tag{B.6}$$

where $\theta_S = (1 - k_S\eta)/\psi$ with $\psi = \tau_{max}/\sigma_{eq}$, and ε_S^+ and ε_S^- correspond to the equivalent plastic strain at shear damage initiation for equibiaxial tensile and equibiaxial compressive deformation, respectively. The parameters θ_S^+ and θ_S^- correspond to the values of θ_S at $\eta = \eta^+$ and $\eta = \eta^-$, respectively. This expression has four parameters that must be determined experimentally: k_S, ε_S^+, ε_S^-, and f. These parameters depend on the material and strain rate. Hooputra et al. have used tensile specimens with a groove (rectangular cross-section and groove depth=half the sheet thickness) at 45° to the loading direction ($\theta_S = 1.469$), specially designed tensile specimens with a groove parallel to the loading direction (pure shear, $\theta_S = 1.732$), and Erichsen tests ($\theta_S = 1.6$) in conjunction with the above expression to determine ε_S^+, ε_S^-, and f. The value of the material parameter k_S is taken as 0.3. For the aluminum alloy used in this example the experimentally obtained shear failure parameters at quasi-static and dynamic strain rates ($250s^1$) are listed in Table B.2.

Table B.2: Experimentally determined shear failure parameters

Parameter	Dynamic ($\dot{\varepsilon} = 250s^{-1}$)
ε_S^+	0.35
ε_S^-	1.2
f	2.05
k_S	0.3

Bibliography

[1] Ver ABAQUS. 6.14 documentation. Dassault Systemes Simulia Corporation, 2014.

[2] Satya N Atluri and T Nishioka. Numerical studies in dynamic fracture mechanics. In Dynamic fracture, pages 119–135. Springer, 1985.

[3] Benoit Beckers and Pierre Beckers. A general rule for disk and hemisphere partition into equal-area cells. Computational Geometry, 45(7):275–283, 2012.

[4] Paul-Emile Bernard, Nicolas Moës, and Nicolas Chevaugeon. Damage growth modeling using the thick level set (tls) approach: Efficient discretization for quasi-static loadings. Computer Methods in Applied Mechanics and Engineering, 233:11–27, 2012.

[5] Virginie Bonnaillie-Noël, Delphine Brancherie, Marc Dambrine, Sébastien Tordeux, and Grégory Vial. Effect of micro-defects on structure failure coupling asymptotic analysis and strong discontinuity. European Journal of Computational Mechanics/Revue Européenne de Mécanique Numérique, 19(1-3):165–175, 2010.

[6] Pierre-Olivier Bouchard, François Bay, and Yvan Chastel. Numerical modelling of crack propagation: automatic remeshing and comparison of different criteria. Computer methods in applied mechanics and engineering, 192(35):3887–3908, 2003.

[7] Pierre-Olivier Bouchard, François Bay, Yvan Chastel, and Isabelle Tovena. Crack propagation modelling using an advanced remeshing technique. Computer methods in applied mechanics and engineering, 189(3):723–742, 2000.

[8] Ramzy Boussetta, Thierry Coupez, and Lionel Fourment. Adaptive remeshing based on a posteriori error estimation for forging simulation. Computer methods in applied mechanics and engineering, 195(48):6626–6645, 2006.

[9] Wade Brainerd. Catmull-clark subdivision surfaces. GPU Pro 7: Advanced Rendering Techniques, 2016.

[10] Delphine Brancherie, Pierre Villon, and Adnan Ibrahimbegovic. On a consistent field transfer in non linear inelastic analysis and ultimate load computation. Computational Mechanics, 42(2):213–226, 2008.

[11] P Breitkopf, A Rassineux, J-M Savignat, and P Villon. Integration constraint in diffuse element method. Computer Methods in Applied Mechanics and Engineering, 193(12):1203–1220, 2004.

[12] Piotr Breitkopf, Alain Rassineux, Gilbert Touzot, and Pierre Villon. Explicit form and efficient computation of mls shape functions and their derivatives. International Journal for Numerical Methods in Engineering, 48(3):451–466, 2000.

[13] Piotr Breitkopf, Alain Rassineux, and Pierre Villon. An introduction to moving least squares meshfree methods. Revue européenne des éléments finis, 11(7-8):825–867, 2002.

[14] P Broumand and AR Khoei. The extended finite element method for large deformation ductile fracture problems with a non-local damage-plasticity model. Engineering Fracture Mechanics, 112:97–125, 2013.

[15] P Broumand and AR Khoei. X-fem modeling of dynamic ductile fracture problems with a nonlocal damage-viscoplasticity model. Finite Elements in Analysis and Design, 99:49–67, 2015.

[16] Godofredo T Camacho and M Ortiz. Computational modelling of impact damage in brittle materials. International Journal of solids and structures, 33(20-22):2899–2938, 1996.

[17] T-S Cao. Numerical simulation of 3d ductile cracks formation using recent improved lode-dependent plasticity and damage models combined with remeshing. International Journal of Solids and Structures, 51(13):2370–2381, 2014.

[18] Jean-Louis Chaboche. Continuum damage mechanics: Part iiâĂŤdamage growth, crack initiation, and crack growth. Journal of applied mechanics, 55(1):65–72, 1988.

[19] JL Chaboche. Viscoplastic constitutive equations for the description of cyclic and anisotropic behaviour of metals. Bull. Acad. Polon. Sci., Ser. Sci. Tech., 25(1):33–42, 1977.

[20] JL Chaboche and G Rousselier. On the plastic and viscoplastic constitutive equationsâĂŤpart i: Rules developed with internal variable concept. Journal of Pressure Vessel Technology, 105(2):153–158, 1983.

[21] Jung-Ho Cheng and Noboru Kikuchi. A mesh re-zoning technique for finite element simulations of metal forming processes. International Journal for Numerical Methods in Engineering, 23(2):219–228, 1986.

[22] JP Cordebois and F Sidoroff. Damage induced elastic anisotropy. In Mechanical Behavior of Anisotropic Solids/Comportment Méchanique des Solides Anisotropes, pages 761–774. Springer, 1982.

[23] Eric Briere de LâĂŹisle and Paul Louis George. Optimization of tetrahedral meshes. In Modeling, Mesh Generation, and Adaptive Numerical Methods for Partial Differential Equations, pages 97–127. Springer, 1995.

[24] Rodrigue Desmorat, Fabrice Gatuingt, and Frédéric Ragueneau. Nonlocal anisotropic damage model and related computational aspects for quasi-brittle materials. Engineering Fracture Mechanics, 74(10):1539–1560, 2007.

[25] JOHN Dolbow and Ted Belytschko. A finite element method for crack growth without remeshing. International journal for numerical methods in engineering, 46(1):131–150, 1999.

[26] John Dolbow, Nicolas Moës, and Ted Belytschko. An extended finite element method for modeling crack growth with frictional contact. Computer methods in applied Mechanics and engineering, 190(51):6825–6846, 2001.

[27] Jean Donea, J-Ph Ponthot, and Antonio Rodriguez Ferran. Encyclopedia of computational mechanics, 2004.

[28] David Dureisseix and Henri Bavestrello. Information transfer between incompatible finite element meshes: application to coupled thermo-viscoelasticity. Computer Methods in Applied Mechanics and Engineering, 195(44):6523–6541, 2006.

[29] Horacio D Espinosa, Pablo D Zavattieri, and Gordon L Emore. Adaptive fem computation of geometric and material nonlinearities with application to brittle failure. Mechanics of Materials, 29(3):275–305, 1998.

[30] E Feulvarch, H Amin El Sayed, J-C Roux, and J-M Bergheau. A stabilized p1/p1 finite element for the mechanical analysis of solid metals. International Journal of Material Forming, pages 1–12, 2015.

[31] E Feulvarch, J-C Roux, J-M Bergheau, and P Gilles. A stable p1/ p1 finite element for finite strain von mises elasto-plasticity. Computer Methods in Applied Mechanics and Engineering, 324:537–545, 2017.

[32] Paul Germain, P Suquet, and QS Nguyen. Continuum thermodynamics. ASME Transactions Series E Journal of Applied Mechanics, 50:1010–1020, 1983.

[33] AM Habraken and Serge Cescotto. An automatic remeshing technique for finite element simulation of forming processes. International Journal for Numerical Methods in Engineering, 30(8):1503–1525, 1990.

[34] DRu Hayhurst and FA Leckie. The effect of creep constitutive and damage relationships upon the rupture time of a solid circular torsion bar. Journal of the Mechanics and Physics of Solids, 21(6):431–432, 1973.

[35] E Hinton and JS Campbell. Local and global smoothing of discontinuous finite element functions using a least squares method. International Journal for Numerical Methods in Engineering, 8(3):461–480, 1974.

[36] H Hooputra, H Gese, H Dell, and H Werner. A comprehensive failure model for crashworthiness simulation of aluminium extrusions. International Journal of Crashworthiness, 9(5):449–464, 2004.

[37] Mazen Issa, Khemais Saanouni, Carl Labergère, and Alain Rassineux. Prediction of serrated chip formation in orthogonal metal cutting by advanced adaptive 2d numerical methodology. International Journal of Machining and Machinability of Materials, 9(3-4):295–315, 2011.

[38] HR Javani, RHJ Peerlings, and MGD Geers. Three-dimensional finite element modeling of ductile crack initiation and propagation. Advanced Modeling and Simulation in Engineering Sciences, 3(1):1–25, 2016.

[39] KR Jayadevan and R Narasimhan. Finite element simulation of wedge indentation. Computers & structures, 57(5):915–927, 1995.

[40] R. El khaoulani and P.O. Bouchard. An anisotropic mesh adaptation strategy for damage and failure in ductile materials. Finite Elements in Analysis and Design, 59:1–10, 2012.

[41] AR Khoei, H Azadi, and H Moslemi. Modeling of crack propagation via an automatic adaptive mesh refinement based on modified superconvergent patch recovery technique. Engineering Fracture Mechanics, 75(10):2921–2945, 2008.

[42] AR Khoei, H Moslemi, and M Sharifi. Three-dimensional cohesive fracture modeling of non-planar crack growth using adaptive fe technique. International Journal of Solids and Structures, 49(17):2334–2348, 2012.

[43] Juha Kuutti and Kari Kolari. A local remeshing procedure to simulate crack propagation in quasi-brittle materials. Engineering Computations, 29(2):125–143, 2012.

[44] Carl Labergere, Bruno Guelorget, and Manuel Francois. Strain rate distribution and localization band width evolution during tensile test. International Journal of Solids and Structures, 51(23):3944–3961, 2014.

[45] Carl Labergere, Alain Rassineux, and Khemais Saanouni. Numerical simulation of continuous damage and fracture in metal-forming processes with 2d mesh adaptive methodology. Finite Elements in Analysis and Design, 82:46–61, 2014.

[46] Carl Labergère, Alain A Rassineux, and Khemais K Saanouni. 2d adaptive mesh methodology for the simulation of metal forming processes with damage. International journal of material forming, 4(3):317–328, 2011.

[47] P Ladeveze and J Lemaitre. Damage effective stress in quasi unilateral conditions. In 16th International congress of theoretical and applied mechanics, Lyngby, Denmark, 1984.

[48] Erastus H Lee. Elastic-plastic deformation at finite strains. ASME, 1969.

[49] Nam-Sua Lee and Klaus-Jürgen Bathe. Error indicators and adaptive remeshing in large deformation finite element analysis. Finite Elements in Analysis and Design, 16(2):99–139, 1994.

[50] Jean Lemaitre. A continuous damage mechanics model for ductile fracture. Transactions of the ASME. Journal of Engineering Materials and Technology, 107(1):83–89, 1985.

[51] Jean Lemaitre, Jean-Louis Chaboche, Ahmed Benallal, and Rodrigue Desmorat. Mécanique des matériaux solides-3eme édition. Dunod, 2009.

[52] Jean Lemaitre, Rodrigue Desmorat, and Maxime Sauzay. Anisotropic damage law of evolution. European Journal of Mechanics-A/Solids, 19(2):187–208, 2000.

[53] Paul Leopardi. A partition of the unit sphere into regions of equal area and small diameter. Electronic Transactions on Numerical Analysis, 25(12):309–327, 2006.

[54] Gilles Loubignac, G Cantin, and G Touzot. Continuous stress fields in finite element analysis. AIAA journal, 15(11):1645–1647, 1977.

[55] Luiz F Martha, LJ Gray, and AR Ingraffea. Three-dimensional fracture simulation with a single-domain, direct boundary element formulation. International journal for numerical methods in engineering, 35(9):1907–1921, 1992.

[56] J Mediavilla, RHJ Peerlings, and MGD Geers. A robust and consistent remeshing-transfer operator for ductile fracture simulations. Computers & structures, 84(8):604–623, 2006.

[57] Nicolas Moës and Ted Belytschko. Extended finite element method for cohesive crack growth. Engineering fracture mechanics, 69(7):813–833, 2002.

[58] JF Molinari and M Ortiz. Three-dimensional adaptive meshing by subdivision and edge-collapse in finite-deformation dynamic-plasticity problems with application to adiabatic shear banding. International Journal for Numerical Methods in Engineering, 53(5):1101–1126, 2002.

[59] S Murakami. Mechanical modeling of material damage. ASME, Transactions, Journal of Applied Mechanics, 55:280–286, 1988.

[60] T Nishioka, RB Stonesifer, and SN Atluri. An evaluation of several moving singularity finite element models for fast fracture analysis. Engineering Fracture Mechanics, 15(1-2):205–218, 1981.

[61] Li-Bin Niu, Hiroshi Takaku, and Mitsuyuki Kobayashi. Tensile fracture behaviors in double-notched thin plates of a ductile steel. ISIJ international, 45(2):281–287, 2005.

[62] M Ortiz and JJ Quigley. Adaptive mesh refinement in strain localization problems. Computer Methods in Applied Mechanics and Engineering, 90(1-3):781–804, 1991.

[63] Ch PavanaChand and R KrishnaKumar. Remeshing issues in the finite element analysis of metal forming problems. Journal of Materials Processing Technology, 75(1):63–74, 1998.

[64] D Perić, Ch Hochard, M Dutko, and DRJ Owen. Transfer operators for evolving meshes in small strain elasto-placticity. Computer Methods in Applied Mechanics and Engineering, 137(3):331–344, 1996.

[65] FM Andrade Pires, EA de Souza Neto, and DRJ Owen. On the finite element prediction of damage growth and fracture initiation in finitely deforming ductile materials. Computer Methods in Applied Mechanics and Engineering, 193(48):5223–5256, 2004.

[66] A Pirondi, N Bonora, D Steglich, W Brocks, and D Hellmann. Simulation of failure under cyclic plastic loading by damage models. International Journal of Plasticity, 22(11):2146–2170, 2006.

[67] M Rachik, JM Roelandt, and A Maillard. Some phenomenological and computational aspects of sheet metal blanking simulation. Journal of Materials Processing Technology, 128(1):256–265, 2002.

[68] MM Rashid. The arbitrary local mesh replacement method: an alternative to remeshing for crack propagation analysis. Computer Methods in Applied Mechanics and Engineering, 154(1-2):133–150, 1998.

[69] MM Rashid. Material state remapping in computational solid mechanics. International Journal for Numerical Methods in Engineering, 55(4):431–450, 2002.

[70] A Rassineux. 3d mesh adaptation. optimization of tetrahedral meshes by advancing front technique. Computer Methods in Applied Mechanics and Engineering, 141(3):335–354, 1997.

[71] A Rassineux, P Breitkopf, and P Villon. Simultaneous surface and tetrahedron mesh adaptation using mesh-free techniques. International journal for numerical methods in engineering, 57(3):371–389, 2003.

[72] Alain Rassineux. Generation and optimization of tetrahedral meshes by advancing front technique. International Journal for Numerical Methods in Engineering, 41(4):651–674, 1998.

[73] Alain Rassineux, Pierre Villon, J-M Savignat, and Olivier Stab. Surface remeshing by local hermite diffuse interpolation. International Journal for numerical methods in Engineering, 49(1-2):31–49, 2000.

[74] Maria-Cecilia Rivara. Mesh refinement processes based on the generalized bisection of simplices. SIAM Journal on Numerical Analysis, 21(3):604–613, 1984.

[75] K Saanouni. Damage mechanics in metal forming. Advanced Modeling and Numerical Simulation John Wiley, London, 2012.

[76] K Saanouni, N Belamri, and P Autesserre. Finite element simulation of 3d sheet metal guillotining using advanced fully coupled elastoplastic-damage constitutive equations. Finite Elements in Analysis and Design, 46(7):535–550, 2010.

[77] K Saanouni, CH Forster, and F Ben Hatira. On the anelastic flow with damage. International Journal of Damage Mechanics, 3(2):140–169, 1994.

[78] K Saanouni, K Nesnas, and Y Hammi. Damage modeling in metal forming processes. International Journal of Damage Mechanics, 9(3):196–240, 2000.

[79] Khémais Saanouni. Numerical modelling in damage mechanics. Kogan Page Publishers, 2003.

[80] Khémais Saanouni. Damage mechanics in metal forming: Advanced modeling and numerical simulation. John Wiley & Sons, 2013.

[81] Khémais Saanouni, Abdelhakim Cherouat, and Youssef Hammi. Numerical aspects of finite elastoplasticity with isotropic ductile damage for metal forming. Revue européenne des éléments finis, 10(2-4):327–351, 2001.

[82] Khemais Saanouni and Mohamed Hamed. Advanced micromorphic elatoplastic constitutive equations with various length scales for metal forming simulation. In USNCCM12: 12th US National Congress on Computational Mechanics, Raleigh, July 23-25, 2013 (USA), 2013.

[83] Khémaïs Saanouni and Y Hammi. Numerical simulation of damage in metal forming processes, 2000.

[84] J Sarrate, J Palau, and Antonio Huerta. Numerical representation of the quality measures of triangles and triangular meshes. International Journal for Numerical Methods in Biomedical Engineering, 19(7):551–561, 2003.

[85] R Schneiders, W Oberschelp, R Kopp, and M Becker. New and effective remeshing scheme for the simulation of metal forming processes. Engineering with computers, 8(3):163–176, 1992.

[86] John P Snyder. An equal-area map projection for polyhedral globes. Cartographica: The International Journal for Geographic Information and Geovisualization, 29(1):10–21, 1992.

[87] Martin GA Tijssens, Bert LJ Sluys, and Erik van der Giessen. Numerical simulation of quasi-brittle fracture using damaging cohesive surfaces. European Journal of Mechanics-A/Solids, 19(5):761–779, 2000.

[88] Henry TY Yang, Martin Heinstein, and J-M Shih. Adaptive 2d finite element simulation of metal forming processes. International journal for numerical methods in engineering, 28(6):1409–1428, 1989.

[89] O. C. Zienkiewicz and J. Z. Zhu. The superconvergent patch recovery and a posteriori error estimates. part 1: The recovery technique. International Journal for Numerical Methods in Engineering, 33(7):1331âĂŞ1364, 1992.