



Université de Bourgogne
UFR Sciences et Techniques

9 avenue Alain Savary
BP 47870
21078 DIJON CEDEX

Master 2 recherche Informatique, instrumentation de l'image et imagerie médicale

Rapport de stage
Génération de maillages volumiques et adaptables pour la simulation
physique

Natacha MOREAU
<moreauna@gmail.com>

Maîtres de stage :
Vincent Vidal, maître de conférences
Fabrice Jaillet, maître de conférences

Université Claude Bernard Lyon 1 - LIRIS UMR 5205
Bâtiment Nautibus
43 boulevard du 11 novembre 1918
69622 VILLEURBANNE CEDEX

Dijon, le 21 juin 2014

Table des matières

Remerciements	4
Introduction	5
Présentation du laboratoire	5
Contexte	5
Objectifs	6
1 État de l'art	7
1.1 Simulation par la méthode des éléments finis	7
1.2 Classification des maillages et des éléments	8
1.3 Génération de maillages volumiques	8
1.3.1 Quels types d'éléments choisir ?	8
1.3.2 Génération adaptative	9
1.3.3 Génération de maillages hexaédriques	9
1.3.4 Génération de maillages tétraédriques	10
1.3.5 Qualité d'un élément	11
1.4 Techniques de remaillage volumique	12
1.5 Structures CGAL utilisées : Cartes combinatoires et Linear Cell Complexes	15
2 Méthode de remaillage proposée	18
2.1 Structure de données : gestion des entrées-sorties	18
2.1.1 Conversion d'un fichier VTK vers un Linear Cell Complex	19
2.1.2 Conversion d'un Linear Cell Complex vers un fichier VTK	20
2.2 Opérations de remaillage local implémentées	20
2.2.1 Opérations topologiques implémentées sur les cartes combinatoires	21
2.2.2 Fonctions de vérification de l'applicabilité d'une opération sur un élément	29
2.3 Premiers algorithmes de remaillage	33
2.3.1 Remaillage a posteriori	33
2.3.2 Perspectives	34

3 Résultats expérimentaux	35
3.1 Résultats des premiers tests des opérations de remaillage	35
3.2 Résultats des premières simulations et modifications envisagées	37
Conclusion	39
Annexes	42

Remerciements

Merci à Vincent Vidal et Fabrice Jaillet pour m'avoir donné l'opportunité de réaliser ce stage, pour le suivi régulier, pour la correction de ce rapport ainsi que pour l'aide apportée aux moments où je n'étais plus capable de m'extraire de certains problèmes, qu'ils soient théoriques ou techniques. Merci à Guillaume Damian (créateur des cartes combinatoires et des Linear Cell Complexes de CGAL) pour l'aide apportée dans la compréhension de la structure, des méthodes à adopter lors de la réalisation des algorithmes de flip, ainsi que pour le débogage spécifique à CGAL ainsi qu'à cette structure précise. Merci à Ricardo Uribe Lobello pour son aide lors du débogage de MEPP et à Xavier Faure pour le temps passé à travailler sur le couplage de la simulation mécanique sous SOFA et du programme de remaillage intégré à MEPP, ainsi que pour la réalisation des simulations et des images présentes dans le rapport, qui en sont issues.

Introduction

Présentation du laboratoire

Situé à Lyon et distribué entre plusieurs sites physiques différents, le LIRIS (Laboratoire d'InfoRmatique en Image et Systèmes d'information) est une unité mixte de recherche placée sous les tutelles du CNRS, de l'INSA de Lyon, de l'Université Claude Bernard Lyon 1, de l'Université Lumière Lyon 2 et de l'École Centrale de Lyon. Il est composé de 320 membres, et divisé en douze équipes couvrant en tout six pôles de compétences : Vision intelligente et reconnaissance visuelle (équipes Imagine et M2DisCo), Géométrie et modélisation (M2DisCo et GeoMod), Science des données (GRAMA, DM2L, BD), Interactions et cognition (SILEX, GRAMA), Services, systèmes distribués et Sécurité (SOC, DRIM) et Simulation, virtualité et sciences computationnelles (SAARA, R3AM et Beagle). L'intégralité du stage se déroule au sein du bâtiment Nautibus, situé sur le campus de l'Université Claude Bernard Lyon 1.

Contexte

La mission du stage, débutée le 17 mars et vouée à se prolonger jusqu'au 16 septembre 2014, se place dans le cadre d'un projet transversal entre les deux équipes SAARA et M2DisCo du LIRIS, lesquelles collaborent afin de mettre en commun les compétences de la première en termes de simulation physique et de la seconde, en termes de techniques de remaillage.

Les outils créés à l'occasion du stage ont dans un premier temps été implémentés au sein de la plateforme MEPP utilisée par l'équipe M2DisCo. MEPP permet le traitement et la visualisation de maillages CGAL de type « Polyhedron ». L'intégration de ces nouvelles fonctions permet d'étendre les capacités de MEPP au traitement des maillages volumiques, tandis qu'il ne gérait jusqu'à présent que les maillages surfaciques. La finalité recherchée est l'intégration des outils en question dans le logiciel de simulation physique en temps réel SOFA, utilisé au sein de l'équipe SAARA. On souhaite améliorer les résultats d'une simulation donnée en modifiant le maillage en entrée de sorte à diminuer les erreurs de simulation tout en conservant la rapidité d'exécution.

Objectifs

La simulation physique en temps réel réalisée au sein du framework SOFA a pour objectif d'appliquer une déformation la plus réaliste possible à un maillage volumique (composé d'éléments volumiques qui représentent l'intérieur de l'objet, et pas uniquement sa surface).

Au cours de la simulation, le maillage d'entrée est progressivement déformé. De mauvais éléments (présents initialement ou bien créés au cours de l'exécution) peuvent entraîner des erreurs qui dégraderont la qualité du résultat final. Afin d'éviter au maximum à ces erreurs d'apparaître, il est nécessaire de détecter les éléments du maillage qui sont susceptibles de les engendrer, et de les remplacer par des éléments de meilleure qualité. Pour cela, deux approches sont envisageables : la première consiste à réaliser une simulation entière, puis à calculer a posteriori certains critères sur les éléments résultants afin de mesurer leur qualité. Les éléments de mauvaise qualité doivent être remaillés dans le fichier d'origine, afin d'espérer améliorer la qualité des éléments résultants lors d'une nouvelle exécution. Une seconde approche dite de remaillage dynamique serait d'approximer la qualité des éléments en cours d'exécution, et d'éliminer ceux qui sont jugés susceptibles d'engendrer des erreurs lors des prochains pas de simulation. Dans un premier temps, on se concentre principalement sur l'option de remaillage a posteriori, en transférant en boucle un fichier de SOFA vers MEPP et inversement, de sorte à remailler en fonction des résultats d'une précédente simulation, puis d'effectuer en boucle une nouvelle simulation sur le résultat remaillé.

Dans un premier temps, il était nécessaire de développer des outils de remaillage volumique susceptibles d'améliorer la qualité pour la simulation de mauvais éléments du maillage, ainsi que les fonctions permettant de vérifier que l'application des dits outils était possible sans risquer la génération d'éléments invalides (plats, de mauvaise topologie), ou la création d'auto-intersections. L'objectif suivant consiste à trouver une façon d'utiliser ces opérations de façon pertinente de sorte à remailler intelligemment et à tendre vers une amélioration globale des résultats de la simulation.

Chapitre 1

État de l'art

1.1 Simulation par la méthode des éléments finis

La méthode des éléments finis permet d'approximer la solution d'une équation différentielle qui simule le comportement d'un objet physique soumis à certaines contraintes (il peut par exemple s'agir d'un fluide, ou bien d'un solide subissant des déformations mécaniques). La forme de l'objet correspond à ce qu'on nomme « domaine ». Le domaine est représenté par un maillage composé d'éléments volumiques, recouvrant l'intégralité du volume théoriquement pris par l'objet. L'idée est de prendre des approximations locales d'une fonction (niveau des éléments), et d'obtenir une approximation globale en assemblant toutes les pièces ensemble. L'approximation globale doit converger vers la véritable solution. Les éléments doivent être finis en taille. Les limites du domaine (conditions de bord) doivent être modélisées par les éléments [Ode91].

De façon générale et pour décrire succinctement le fonctionnement de la méthode, on a $\{F\} = [K] \{U\}$ où F sont les forces appliquées sur un élément, K est la matrice de raideur de l'élément (modélisant sa résistance contre ces forces), et U les déplacements que l'élément subit.

L'influence des différentes forces est interpolée sur le domaine. La matrice de raideur est une matrice $n*n$ où n est le degré de liberté de l'élément fini. En général le degré de liberté est calculé en faisant le produit du nombre de nœuds d'un élément, et du nombre de valeurs de la variable de champ (par exemple, trois valeurs pour les trois coordonnées x , y et z des nœuds du domaine). De petites différences de calcul peuvent apparaître selon le nombre d'éléments qui composent l'interpolation. Une définition de l'erreur d'interpolation est donnée comme étant la distance de la solution du problème au sous-espace des approximations [Ode91].

1.2 Classification des maillages et des éléments

Un maillage est une structure de données géométrique composée de sommets de l'espace, reliés entre eux par des arêtes et plus généralement, par des éléments. Ces éléments peuvent être des polygones (triangles, quadrilatères...) qui se contentent de recouvrir la surface de l'objet représenté (auquel cas on parlera de maillage surfacique). Un maillage composé d'éléments volumiques (comme des tétraèdres, ou des hexaèdres) qui recouvrent l'intégralité de son volume sera quant à lui appelé « maillage volumique ».

Les maillages peuvent être classifiés en deux autres catégories : ils peuvent être structurés (avec un motif de connectivité régulier, qui peut être retrouvé facilement sans nécessité de stocker l'information) ou bien non-structurés (les connectivités ne suivent aucun motif et doivent être stockées). Les maillages structurés sont bons pour représenter des domaines à la géométrie simple, proche d'une (ou composée de) région(s) hexaédrique(s). Pour des domaines complexes cependant, les maillages non structurés semblent être meilleurs (car plus adaptables) [GBF⁺97].

Un maillage régulier est un maillage ne contenant que des éléments équilatéraux. Il n'est généralement pas garanti que ce maillage existe pour un domaine aléatoire, mais il existe des méthodes qui permettent de générer des maillages « presque réguliers ». Il y a deux types de discrétisation de la frontière d'un domaine : les discrétisations uniformes (avec un pas de taille constant), et les discrétisations géométriques (avec un pas de taille variant avec la courbure de la frontière) [GBF⁺97].

1.3 Génération de maillages volumiques

Les discrétisations uniformes sont utiles lorsqu'il s'agit de créer des maillages réguliers, mais elles peuvent ne pas se conformer à la forme originale de la frontière. Les discrétisations géométriques respectent mieux la forme de la frontière mais la régularité des maillages générés en partant d'elles n'est pas aussi facile à atteindre [GBF⁺97].

1.3.1 Quels types d'éléments choisir ?

On se concentrera sur la génération de maillages volumiques, lesquels nous intéressent particulièrement puisqu'ils sont utilisés comme donnée dans le cadre de la simulation dont on souhaite améliorer le résultat.

Les éléments d'un maillage doivent former un complexe cellulaire (ils ne doivent s'intersecter que sur un sommet, une arête, ou une face entière). La génération de maillage est plus facile lorsqu'on autorise la présence d'éléments non-conformes (par exemple une face qui partage une arête avec deux autres faces plus petites).

Un maillage composé d'éléments tétraédriques sera logiquement nommé « maillage tétraédrique ». Pendant longtemps, la méthode des éléments finis a surtout été appliquée sur ces maillages car ils sont plus faciles à générer, et à adapter à beaucoup de formes de domaines différentes. Un maillage composé d'éléments hexaédriques sera nommé « maillage hexaédrique ». Pour des raisons topologiques, les maillages hexaédriques peuvent être très difficiles à générer pour des domaines à la géométrie complexe. De même, il peut s'avérer difficile de faire varier progressivement la taille des éléments de ce type de maillage sans générer d'éléments de mauvaise qualité. Il pourrait pourtant être intéressant de réussir à passer outre ces difficultés car pour un même domaine, avec le même degré de précision, un maillage hexaédrique sera composé d'un plus petit nombre d'éléments qu'un maillage tétraédrique, ce qui pourrait amener à une optimisation du temps de calcul [FM07]. Il n'y aurait a priori aucun algorithme véritablement fiable et suffisamment générique pour la génération des maillages hexaédriques [She11]. L'utilisation de structures hybrides peut pallier à cette difficulté : maillages à dominantes hexaédriques. D'autres types d'éléments peuvent encore être retrouvés dans les maillages mixtes : prismes et pyramides.

1.3.2 Génération adaptative

Un problème de génération de maillage est décrit de la façon suivante : Un domaine Ω dans R^2 ou R^3 est défini par sa frontière Γ . En 3D, Γ est approximé avec une surface constituée de polygones (triangles, quads...). Une approximation (maillage volumique) de Ω est construite. Dans un problème de génération de maillage classique, la qualité des éléments sera jugée sur leur forme et sur la gradation du maillage uniquement. Pour une génération de maillage adaptative ou contrôlée, une taille et une direction idéale seront définies pour chaque élément [GBF⁺97].

1.3.3 Génération de maillages hexaédriques

Les méthodes pour la génération de maillages hexaédriques peuvent être classées en trois catégories : les méthodes directes (par front avançant) qui peuvent produire des régions vides impossibles à combler à l'intérieur du maillage [SOB05], les méthodes indirectes, qui construisent un maillage tétraédrique avant de le convertir en maillage hexaédrique, et peuvent engendrer la création de très mauvais éléments, et les techniques structurées (qui génèrent des maillages structurés) qu'on ne peut appliquer qu'à des maillages de forme simple [LLD12].

Trois principales techniques de génération de maillages sont présentées dans l'article de Shewchuk et al. [She11] : les méthodes par front avançant, les algorithmes de raffinement de Delaunay, et les méthodes via grilles, quadrees et octrees.

1.3.4 Génération de maillages tétraédriques

Méthodes par Front Avançant

Les méthodes par front avançant consistent à partir de la surface du domaine, puis à avancer progressivement à l'intérieur de ce dernier en générant de nouveaux éléments jusqu'à ce qu'ils se rejoignent dans le maillage en construction. Les éléments à la surface sont de bonne qualité, contrairement à ceux qu'on trouve à l'intérieur du maillage, au niveau du point de collision entre les fronts. Ce type d'approche a été particulièrement fructueux en mécanique des fluides. Quatre points doivent être alors éclaircis : Comment choisir, sur le front avançant, la face à partir de laquelle on crée le nouvel élément ? Quels sommets préexistants, ou quels nouveaux sommets devront être liés à cette nouvelle face dans l'espoir de construire des éléments de qualité optimale ? Comment trouver rapidement les sommets candidats, et comment rapidement savoir si l'élément en cours de construction intersecte le maillage de manière valide ? Certains problèmes peuvent apparaître dans les maillages tétraédriques lorsqu'on rencontre une cavité qui ne peut pas être divisée en tétraèdres. Déterminer la quantité de sommets qui devront être insérés pour que ce genre de cavités puisse être divisé correctement est généralement un problème NP-difficile. Les générateurs de maillage par front avançant arrivent le plus souvent à leurs fins avec une complexité de $O(n \log n)$, à moins qu'une cavité non-tétraérisable soit rencontrée durant l'exécution.

Algorithmes de raffinement de Delaunay

Ce genre d'algorithmes permet de ne générer que des maillages triangulaires et tétraédriques [MMG00]. En premier lieu, une triangulation de Delaunay (en deux dimensions ou plus) est réalisée, puis raffinée en insérant de nouveaux sommets pour éliminer les éléments trop fins ou trop gros, tout en maintenant les propriétés de Delaunay du maillage. La triangulation de Delaunay crée le moins d'arêtes courtes et d'éléments fins possible. Les meilleurs éléments se trouvent à l'intérieur du domaine, tandis qu'on trouve les moins bons sur sa surface. La triangulation de Delaunay d'un ensemble de points existe toujours et est unique, à quelques cas exceptionnels près (correspondant au cas où pour une triangulation de dimension d , on trouve $d+1$ sommets dans le même hyperplan et $d+2$ sommets dans la même hypersphère). La triangulation en elle-même ne garantit pas qu'aucun élément trop fin n'apparaîtra, ni que les limites du domaine seront respectées. Une triangulation de Delaunay sous contrainte peut aider à résoudre le second problème en fixant les arêtes qui ne devront pas être traversées. En 3D, il peut être nécessaire de rajouter des sommets afin de parvenir à achever la triangulation sous contrainte. L'étape de raffinement suite à la triangulation pose plusieurs problèmes : comment choisir la position des points à insérer pour l'élimination des mauvais éléments ? Et pour renforcer la conformité au domaine ?

Générateurs de maillages par grille

Ces générateurs créent une grille de fond constituée de cellules qui correspondent à de simples, petites parties du domaine. La génération de maillages gradués peut être réalisée grâce à l'exploitation d'octrees et de quadtrees. Des très bons éléments sont placés à l'intérieur du domaine mais les éléments à sa surface sont spécialement mauvais (plus mauvais qu'avec les deux autres types de méthodes présentées plus haut). La plupart de ces algorithmes, dans le cas de la génération de maillages non-structurés [She11] nécessitent que l'arbre ait plus d'un sommet dans chacune de ses feuilles, et qu'il soit équilibré. Afin d'assurer la qualité des éléments générés, certains sommets (la plupart du temps sur la surface du domaine) doivent être déplacés. Enfin, les feuilles de l'arbre sont triangulées. Les méthodes par arbre ou par grille ne sont généralement pas très adaptées à la gestion des cas anisotropes. [GBF⁺97]

1.3.5 Qualité d'un élément

Les éléments du maillage doivent avoir « la bonne forme ». Selon un critère de qualité géométrique, cette forme devra être la plus équilatérale possible (dans le cas d'éléments triangulaires ou tétraédriques). Dans certains cas, cependant, on pourra tirer parti d'éléments anisotropes (lorsqu'on est obligé d'utiliser un nombre fixe d'éléments pour approximer une fonction donnée, ou encore pour modéliser un matériau possédant des propriétés anisotropes). Les éléments ne doivent pas présenter d'angles trop larges (qui causent des erreurs d'interpolation). De trop petits angles peuvent aussi poser problème dans certains cas (dans le cadre de la méthode des éléments finis par exemple, de mauvaises valeurs propres dans la matrice de raideur d'un élément). La taille d'un élément évolue en fonction d'un autre paramètre d'entrée : le champ de taille [FPR⁺00], lequel varie en fonction de la position prise dans l'espace du domaine. De petits éléments peuvent être utiles dans certaines régions où une bonne précision est nécessaire à la représentation du domaine. Au contraire, on préférera utiliser de gros éléments dans des régions plus grossières (limiter le nombre d'éléments améliore l'efficacité de la méthode). Être capable de passer de petits éléments à de gros éléments sur une courte distance peut améliorer les résultats, mais introduire des éléments très fins (spécialement dans le cas des maillages hexaédriques). Les contraintes de conformité au domaine et de qualité des éléments ne sont pas évidentes à satisfaire simultanément.

Eléments tétraédriques contre éléments hexaédriques

De manière générale, les maillages tétraédriques sont plus faciles à générer et à remailler que les maillages hexaédriques. Un élément tétraédrique sera de plus forcément convexe, contrairement à un hexaèdre dont la potentielle forme concave pourrait engendrer des erreurs de simulation. Malgré tout, les éléments hexaédriques possèdent des propriétés intéressantes que n'ont pas les tétraèdres : ils sont indépendants de l'orientation du maillage, et plus faciles à déformer [FM07].

Critères de qualité d'un élément

Les travaux de Wicke et al. [WRK⁺10] décrivent aussi un bon élément (tétraédrique en l'occurrence) comme étant le plus équilatéral possible, à moins que des propriétés anisotropes soit explicitement demandées. La qualité d'un élément est ici calculée grâce à une mesure volume-longueur introduite dans les travaux de Parthasarathy et al. [ITF04]. Un tenseur 3x3 aide à définir l'anisotropie de l'élément idéal.

Une autre façon d'évaluer la régularité d'un élément est de calculer son *aspect ratio*, compris entre 0 et 1. Dans le cas d'un élément tétraédrique équilatéral, on obtiendra un *aspect ratio* de 1. Moins l'élément est régulier (aplatis, allongé), moins la valeur de son *aspect ratio* sera élevée. [GBF⁺97]

Le calcul de la matrice jacobienne d'un élément (et de son déterminant) peut donner une information supplémentaire sur sa validité. En effet, un élément de jacobien négatif sera un élément inversé, et donc forcément invalide (ce genre d'éléments peut facilement être généré lors d'un déplacement de nœud, par exemple via lissage laplacien). La fonction utilisée pour calculer la matrice jacobienne est celle qui permet d'associer aux coordonnées de chaque point de l'élément de référence, celles d'un élément déformé du maillage [BLPH10].

Dans le cas d'un élément hexaédrique, le jacobien n'existe pas. Afin de pallier cette difficulté, il est possible de calculer le déterminant de la matrice jacobienne en chaque nœud (sur le tétraèdre formé par le nœud et par les trois autres qui lui sont liés) et de prendre le maximum ou le minimum selon les cas. Si l'un des déterminants est négatif, alors l'élément est non valide car inversé [BLPH10].

D'autres fonctions géométriques pour l'évaluation de la qualité des éléments peuvent être utilisées [Fie00] [Knu99].

1.4 Techniques de remaillage volumique

Amélioration de maillage par modifications locales

Suite à la génération d'un maillage, une étape d'amélioration peut être réalisée [She11]. Elle consiste à appliquer un ensemble d'opérations de transformations locales afin de remplacer certains mauvais éléments par d'autres, meilleurs. Un inconvénient de ce type d'améliorations locales est qu'elles manquent de garanties mathématiques (mais elles peuvent au moins garantir qu'elles ne dégraderont pas la qualité du maillage). Les différentes opérations réalisables doivent être ordonnées au sein d'un programme qui cherchera des opportunités de les appliquer [KS07]. Un lissage laplacien peut être appliqué sur les sommets intérieurs du maillage afin d'améliorer la qualité des éléments qui les lient, mais dans le cas de maillages tétraédriques, quadrilatéraux et hexaédriques, c'est une

opération peu fiable, qui peut engendrer des éléments inversés [BLPH10]. Alternier entre ce lissage global et le calcul d'une nouvelle triangulation de Delaunay peut améliorer les résultats obtenus [ACSYD05]. L'algorithme d'optimisation non-linéaire donne des résultats encore meilleurs, mais il est particulièrement lent. D'autres types d'optimisations peuvent être réalisées au travers d'opérations telles que des flips d'arêtes, contraction d'arête, suppression d'arête, ou encore des flips bistellaires. D'autres types de flips existent pour les maillages quadrilatéraux et hexaédriques [BEE02]. Les flips topologiques sont les opérations les plus rapides à effectuer, tandis que des opérations telles que la contraction ou la suppression d'arêtes sont plus coûteuses. Ce type d'amélioration locale est une bonne méthode pour introduire de l'anisotropie dans un maillage.

La dite méthode est notamment appliquée dans le cadre des travaux de remaillage dynamique de Wicke et al. [WRK⁺10]. La méthode de remaillage en question améliore la précision d'une simulation dynamique par éléments finis sur des matériaux plastiques et élastiques subissant des déformations. Le remaillage est réalisé localement afin de limiter les erreurs d'interpolation et la diffusion artificielle de déformations plastiques sur des objets purement ou principalement élastiques. L'amélioration est effectuée via un algorithme de type « hill-climbing » (recherche locale au sein de laquelle la solution au problème est améliorée de façon itérative en modifiant un unique élément de la solution). Deux types d'approches pour l'amélioration de la qualité des méthodes par éléments finis sont cités : le raffinement hiérarchique (h-adaptivité [JP]), qui raffine les éléments dans les zones où l'erreur de simulation va au-delà d'un seuil d'acceptation donné), et les méthodes de maillage mobile (« relocation refinement », ou r-adaptivité [BHR09], qui ne modifient pas le nombre de noeuds dans le maillage mais bougent certains points dans le but de les concentrer dans les zones d'intérêt, où plus de précision est requis). Cette méthode est conçue pour les maillages tétraédriques. Elle applique plusieurs types d'opération, citées au préalable : flips d'arête, lissage de sommet, contraction d'arête, insertion de sommet, retrait multi-face [She]. Un programme d'application des dites opérations contrôle la priorité avec laquelle ces dernières seront appliquées selon les cas rencontrés. Celui qui est ici implémenté s'inspire des travaux de Freitag et al. [FOG97], ensuite étendus par ceux de Klingner et al. [KS07]. La qualité des tétraèdres est quantifiée grâce à un vecteur de qualité qui possède les propriétés suivantes : si une sous-partie (représentant un sous-maillage) du vecteur de qualité est améliorée, alors le vecteur entier l'est aussi. Il peut être nécessaire de raffiner ou de grossir les éléments pour que leur taille soit adaptée au gradient de la déformation et à la forme du maillage déformé.

Un autre type d'opérations de remaillage local dit de « couper coller » est présenté dans l'article de Boussetta et al. [BCF06]. Il s'agit de supprimer un sous-maillage et de le remplacer par un nouveau qui possède les mêmes frontières (les flips bistellaires semblent être un type particulier d'opération de couper coller). La qualité des sous-maillages « avant » et « après » peut être comparée via leur pire élément. Ici, le nouveau sous-maillage est généré via un opérateur étoilé. Soit une cavité obtenue suite à la suppression d'un sous-maillage (constitué de tous les éléments connectés à un sommet ou une arête donnée). L'opérateur étoilé connecte un noeud aux faces des bords de la

cavité qui ne le contiennent pas, et génère ainsi de nouveaux éléments. Pour obtenir un maillage conforme, en accord avec le théorème du volume minimal donné par Coupez et al. [Cou00], on considère comme étant les meilleurs candidats les éléments de volume minimal. Ainsi, le nouveau nœud ajouté à l'intérieur de la cavité est associé aux coordonnées du barycentre de cette cavité.

Carte de tailles optimisée

Les travaux de Boussetta et al. [BCF06] en remaillage dynamique évoquent une méthode qui permet de calculer la taille idéale d'un élément en fonction de l'évaluation a posteriori de l'erreur de simulation par éléments finis [ZZ92]. Cette technique permet non seulement d'optimiser le remaillage adaptatif, mais aussi de fixer le nombre d'éléments maximal qu'on autorise dans le maillage. Tant que le maximum d'éléments autorisés à l'intérieur du maillage n'a pas été atteint, la carte de tailles optimisée est obtenue grâce à la méthode AST1. Cette méthode permet d'obtenir un maillage optimal T_{opt} en définissant une valeur de précision arbitraire θ^{imp} qui quantifie l'erreur qu'on autorise la simulation à avoir. La contribution θ_e de chaque élément e à l'erreur globale doit être d'abord calculée via l'estimateur. Ensuite, l'erreur uniforme distribuée θ^{uni} est trouvée à partir de cette valeur. La carte de tailles optimisée h_e^{opt} qui prend en compte l'erreur uniforme distribuée peut de ce fait être calculée. Enfin le nouveau nombre d'éléments $Nbelt^{opt}$ est obtenu. L'ensemble du processus est réalisé avec une complexité $O(n)$, où n est le nombre d'éléments dans le maillage. Si le nombre d'éléments estimé $Nbelt^{opt}$ venant d'être calculé avec la première méthode est plus élevé que la quantité maximale d'éléments autorisés $Nbelt^{imp}$, la carte de tailles optimisée est de nouveau calculée, cette fois via la méthode AST2. Celle-ci permet de trouver le maillage optimal T_{opt} en définissant un $Nbelt^{imp}$ arbitraire. Un nouveau θ^{imp} est calculé en $O(n)$ puis les étapes précédentes (le calcul de θ^{uni} et h_e^{opt}) sont de nouveau réalisées. On s'assure donc que $Nbelt^{opt}$ sera approximativement égal à $Nbelt^{imp}$. Un processus de raffinement (ou de déraffinement) est exécuté afin d'obtenir un maillage qui respectera la carte de tailles optimisée.

1.5 Structures CGAL utilisées : Cartes combinatoires et Linear Cell Complexes

Une carte combinatoire est une structure de données constituée de cellules de dimensions d qui, ensemble, forment un complexe cellulaire. Ainsi, une carte combinatoire peut facilement représenter la topologie d'un maillage (pour l'instant, on ne parle pas de ses propriétés géométriques, telles que par exemple les coordonnées de ses sommets). On peut voir les cartes combinatoires comme l'extension logique de la structure de demi-arête à la dimension $d \geq 1$.

Parcours de la structure

L'élément constitutif de base d'une carte combinatoire est le **brin**. Un brin est un élément orienté qui peut être lié à d'autres brins par l'intermédiaire de pointeurs β_i (avec i compris entre 0 et d). Les cellules de la carte combinatoire sont constituées de brins. Afin de comprendre le fonctionnement de base de la structure de données, on va prendre l'exemple d'une carte combinatoire de dimension 2 formant une triangulation planaire.

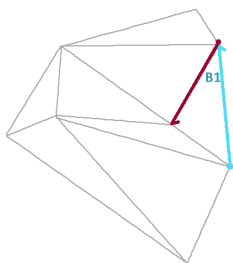


FIGURE 1.1 – Mise en évidence d'un brin (bleu) appartenant à une 1-cell, lié à un autre brin (et à une autre 1-cell) par l'intermédiaire d'un lien β_1

Une 1-cell représente une arête. Un brin peut être considéré comme une demi-arête (ou arête orientée). Un pointeur β_1 permet d'atteindre la 1-cell suivante. Si une 1-cell représente une arête (point d'origine du brin et point du brin pointé par β_1) alors une 0-cell représente un sommet (point d'origine du brin). Un pointeur β_0 permet d'atteindre la 1-cell précédente (sur l'image, le lien β_0 du brin rouge permettra d'accéder au brin bleu).

De la même façon une 2-cell, constituée d'un chemin de brins fermé représentera une face. Les pointeurs β_0 et β_1 permettront de « naviguer » sur les arêtes de la face en question. Le pointeur β_2 quant à lui permet d'atteindre la face adjacente à un brin particulier, si la facette adjacente existe. Si elle n'existe pas, alors on dit que le brin est « 2-free » (ce qui se généralise à toutes les dimensions).

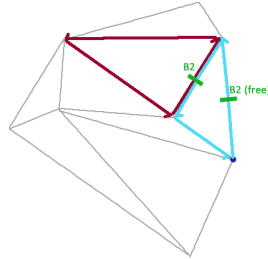


FIGURE 1.2 – Mise en évidence des brins d’une 2-cell (bleue) liée à une autre 2-cell par l’intermédiaire d’un lien β_2

Et ainsi de suite. Dans une carte combinatoire 3D, une 3-cell représentera un élément volumique, entre les faces duquel on pourra naviguer via les pointeurs β_2 . On pourra passer d’une arête à l’autre d’une face en particulier via les pointeurs β_0 et β_1 . Enfin, l’accès aux éléments volumiques voisins se fera via les pointeurs β_3 .

Un brin ne peut être lié qu’à une seule i -cell pour chaque dimension $i \leq d$, d étant la dimension de la carte. Il n’y a donc jamais d’ambiguïté sur le choix de cellule à coudre. Le nombre de brins contenu par une i -cell varie en fonction de la complexité des jonctions auxquelles elle participe. Une face triangulaire libre sur un élément volumique sera composée de trois brins. Une face triangulaire joignant deux éléments volumiques ensemble en sera composée de six (trois appartenant au premier élément volumique, trois au second, dont l’orientation ira à l’inverse des précédents). La 1-cell commune à trois éléments volumiques au niveau d’une jonction en T sera composée de six brins. Le choix des brins est important car il assure qu’on est bien en train de parcourir la bonne face, sur le bon élément volumique (deux faces possibles pour chacun des trois éléments volumiques impliqués dans ce genre de jonction).

Couture des cellules

Une carte combinatoire peut être composée d’un ou plusieurs ensembles de cellules liées entre elles par une relation de voisinage (on dit qu’elles sont cousues). Lors de la création d’une cellule de dimension d dans la carte, on obtient un ensemble de brins liés ensemble par des pointeurs β_0 à β_{d-1} (par exemple une 3-cell, dont les faces (2-cells) sont reliées via leurs β_2 et leurs arêtes via leurs β_1). La cellule étant indépendante, afin de l’associer à d’autres éléments de même dimension, on va devoir la coudre manuellement à ces derniers (c’est à dire associer aux pointeurs β_d de ses brins l’adresse d’un brin d’un autre élément). Des opérations de couture sont mises à disposition par les cartes combinatoires de CGAL dans toutes les dimensions. Une couture en dimension 3 permet de

coudre deux 3-cells ensemble en fusionnant une 2-cell de chaque (on lie deux éléments volumiques en fusionnant une face de l'un avec une face de l'autre). De même coudre deux 2-cells ensemble revient à fusionner une 1-cell de chaque (on lie deux faces en fusionnant une arête de l'un avec une arête de l'autre). Ainsi de suite dans toutes les dimensions.

Marques booléennes

Des marques booléennes à associer aux brins d'une carte combinatoire sont mis à disposition par CGAL. Elles peuvent notamment servir à rendre un brin reconnaissable, et offrent une information supplémentaire à exploiter dans les algorithmes travaillant dessus (marquage des brins d'une cellule particulière, marquage des brins ayant subi un traitement particulier...). Ces marques doivent tout d'abord être créées (chaque marque créée possédera une valeur différente de celles qui ont été générées précédemment, si bien qu'il sera possible de les différencier). Chaque marque peut ensuite être associée à un ou plusieurs brins (on peut par exemple choisir de marquer tous les brins d'un même élément volumique afin de se servir plus tard de cette information). A tout moment, une fonction **is_marked** prenant en paramètre un brin et une marque booléenne peut être appelée. Elle renvoie vrai si le brin possède la marque en question, et faux sinon.

Attributs des cellules

Enfin, des i -attributs (comme par exemple une couleur) peuvent être attribués à chaque i -cell ($i = 0$ à d), via un template « Cell_attribute » auquel on peut associer un type d'attribut particulier (double, booléen, classe particulière conçue sur mesure...), ainsi que deux foncteurs « OnMerge » et « OnSplit » qui définissent le comportement des attributs lorsque les cellules sur lesquels ils portent sont respectivement cousues ou décousues (Exemple lors d'une découpe : valeur d'attribut copiée sur les deux i -cell résultantes de la division de l' i -cell d'origine, ou encore valeur divisée par deux). Le type des attributs est unique pour une dimension i donnée. Afin de sauvegarder la géométrie associée à un maillage qu'on souhaite représenter sous la forme d'une carte combinatoire, il faut rajouter aux 0-cells (sommets) un attribut supplémentaire représentant ses coordonnées dans l'espace. Dans la librairie C++ CGAL, les cartes combinatoires sont étendues par une structure de données nommée « Linear Cell Complexes » qui sont un plongement de la structure des cartes combinatoires dans l'espace R^d . Les Linear Cell Complexes associent aux 0-cells un attribut de type Point en plus de leur 0-attribut standard. Les Linear Cell Complexes de CGAL offrent quelques outils supplémentaires vis-à-vis des cartes combinatoires d'origine (ex : insertion de barycentre dans une 1-cell ou une 2-cell).

Chapitre 2

Méthode de remaillage proposée

2.1 Structure de données : gestion des entrées-sorties

Soit un maillage volumique sur lequel on souhaite effectuer une simulation. On a pour objectif de le remailler non pas pour améliorer son aspect géométrique, mais pour éliminer les éléments qui engendrent de fortes erreurs dans la simulation. Ce maillage, d'abord livré sous la forme d'un fichier au format VTK, doit être lu et stocké dans une structure de données, sur laquelle on effectue les opérations de remaillage vouées à améliorer la qualité des éléments pour la simulation. Il a été dans un premier temps décidé que les éléments volumiques pourraient être modélisés par un chemin d'arêtes. L'utilisation des cartes combinatoires (permettant facilement ce type de parcours sur un élément, et garantissant de plus la conformité du maillage généré) a de ce fait été naturellement sélectionnée. La classe « Linear Cell Complex » de CGAL a été utilisée pour réaliser l'implémentation de la structure de données.

Des attributs sont fixés sur les sommets (0-cells) et sur les éléments (3-cells). On associe à chaque sommet un indice entier, notamment utilisé lors des opérations de lecture et écriture de la carte combinatoire depuis et vers un fichier VTK. Ce format de fichier a été choisi en raison de la facilité avec laquelle il permet de gérer des champs scalaires et des champs vectoriels associés aux cellules d'un maillage volumique. Les attributs des volumes représentent la valeur du critère de qualité généré par une exécution de la simulation (et par l'estimation de son erreur). Il s'agit d'un réel qu'on utilisera plus tard pour savoir si un élément donné doit être raffiné ou bien gardé tel quel.

Dans un premier temps, on choisit de travailler sur des maillages tétraédriques, afin de tirer parti de leur propriétés : forme invariablement convexe, rapidité des algorithmes de remaillage sur les maillages à base d'élément tétraédriques.

2.1.1 Conversion d'un fichier VTK vers un Linear Cell Complex

Une fonction fournie par MEPP permet de lire un fichier VTK à l'intérieur d'une structure de données intermédiaire, constituée d'une liste de sommets, d'une liste de facettes, d'une liste de volumes, d'une liste de normales de faces et d'une liste de champs d'attributs.

La liste des éléments volumiques est parcourue. Les tétraèdres peuvent facilement être générés car il existe une correspondance directe entre les valeurs que la liste contient et les indices des points de la liste de sommets. La création des attributs de chaque élément est de même triviale puisque l'indice de l'élément courant est aussi celui de sa valeur (scalaire) associée dans le champ d'attributs.

La seule véritable difficulté rencontrée lors de la génération du Linear Cell Complex consiste à détecter les facettes qui doivent être cousues entre elles (sachant qu'une facette ne peut être cousue qu'une seule fois à une autre). On va, pour chaque facette d'un nouvel élément inséré, vérifier si une autre facette constituée des mêmes sommets a déjà été insérée au sein d'un tableau associatif (`std::map`) prenant pour clé les adresses des sommets en question. Si oui, les facettes doivent être cousues ensemble. Sinon, l'un des brins permettant d'accéder à la nouvelle facette est enregistré dans le tableau pour être utilisé plus tard lors d'une opération de couture.

Pour coudre deux faces, on utilise l'opération `sew` des cartes combinatoires, qui prend en paramètre deux brins, lesquels doivent correspondre à la même 1-cell, et avoir des orientations opposées, sans quoi la topologie du maillage d'origine ne sera pas respectée (ce qu'on appellera « couture non valide »).

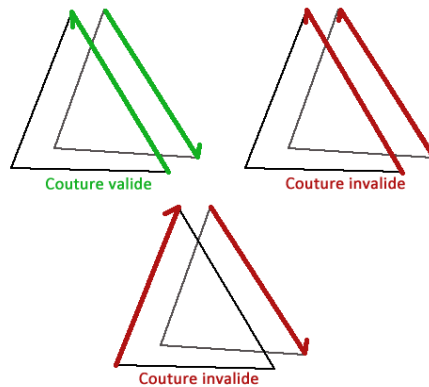


FIGURE 2.1 – Exemples de coutures de faces (entre deux 3-cells d'une carte combinatoire) valides et non valides

2.1.2 Conversion d'un Linear Cell Complex vers un fichier VTK

Les sommets du Linear Cell Complex sont parcourus afin d'être enregistrés dans la liste de points de la structure intermédiaire précédemment citée. Leur adresse est associée à un indice particulier. De cette façon on peut créer les listes de facettes et de volumes en parcourant l'ensemble des facettes du Linear Cell Complex puis l'ensemble de ses éléments. Les indices associés aux adresses des points constituant les 2-cells et 3-cells concernées sont stockés dans les fameuses listes. Le parcours des éléments permet aussi de remplir la liste de champs d'attributs. L'opération d'écriture ne présente aucun type de difficulté particulier.

2.2 Opérations de remaillage local implémentées

Dans un premier temps, afin d'étudier le comportement du maillage soumis à certaines opérations de raffinement et de déraffinement, ainsi que la pertinence des critères de qualité des éléments vis-à-vis de l'évolution de l'erreur, nous avons choisi de nous concentrer sur le remaillage a posteriori d'un maillage volumique sur lequel une simulation a été effectuée. L'estimation de l'erreur de simulation fournit une information sur les éléments de mauvaise qualité qu'on souhaiterait modifier. Cela dit, on envisage à plus long terme d'évoluer vers un remaillage dynamique (effectué en cours de simulation). Pour cette raison, l'idée d'un remaillage global (reconstruction du maillage entier « from scratch », par exemple grâce à une triangulation de Delaunay) est naturellement écartée.

Parmi les opérations de remaillage local existantes, certaines sont plus rapides et moins contraignantes que d'autres ce qui, dans une optique de temps réel, les rend particulièrement intéressantes. Dans un premier temps, nous avons donc choisi d'implémenter les opérations de flips topologiques basiques sur les tétraèdres avec les Linear Cell Complexes : Flip 4-4, 2-3, 3-2, 4-1 et 1-4. La combinaison judicieuse de ces flips peut aider à modifier la qualité des éléments internes à la frontière des tétraèdres concernés par l'opération. Néanmoins, comme ces flips ne modifient pas les faces externes de l'ensemble de tétraèdres en question, si l'on souhaite raffiner le maillage à certains endroits où des éléments de plus petite taille sont requis, on va devoir se tourner vers d'autres opérateurs capables de scinder ces faces. Une opération de flip 1-3 (ou 2-6 selon la configuration des éléments scindés) a donc été ajoutée à l'ensemble des outils de remaillage proposés. Pour le moment, son opération inverse (3-1 ou 6-2) n'a pas été implémentée. On affinera la stratégie de remaillage adoptée en fonction des problèmes rencontrés.

2.2.1 Opérations topologiques implémentées sur les cartes combinatoires

Flip bistellaire 4-4

Ce flip porte sur quatre tétraèdres regroupés autour d'une arête centrale. Il s'agit d'une opération symétrique (qui appliquée deux fois permet donc de retrouver la configuration d'origine). Elle ne modifie pas le nombre d'éléments dans le maillage, et ne peut être considérée ni comme une opération de raffinement, ni comme une opération de déraffinement. L'ajout de nouveaux tétraèdres en remplacement des anciens peut éventuellement permettre d'améliorer (ou de dégrader) la qualité des quatre éléments comblant l'espace concerné. La valence de certains points change, ce qui peut être considéré comme une donnée à prendre en compte puisque la qualité de la simulation s'améliore avec la régularité de la valence sur l'ensemble du maillage (elle doit rester la plus égale possible, et évoluer graduellement).

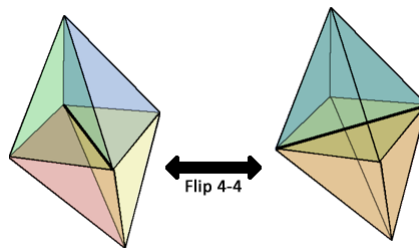


FIGURE 2.2 – Exemple de flip bistellaire 4-4 sur un ensemble basique de quatre tétraèdres

Algorithme pour réaliser un flip bistellaire 4-4 sur des éléments tétraédriques d'un Linear Cell Complex : D'abord, on part de l'arête centrale qui relie les quatre éléments ensemble. Les facettes (2-cells) liées à cette arête (1-cell qui contient quatre brins, liés à quatre facettes) sont supprimées. Ainsi, on a « vidé » la cavité dans laquelle les tétraèdres se trouvaient et on obtient une unique, grosse cellule, qu'il faut maintenant remailler.

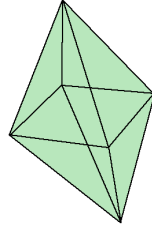


FIGURE 2.3 – Cellule obtenue suite à la suppression des quatre facettes reliées à l’arête centrale lors d’un flip 4-4

Une nouvelle facette est créée afin de scinder la cavité en deux, dans le sens où elle ne l’était pas au préalable. Une nouvelle arête centrale est insérée dans la dite facette.

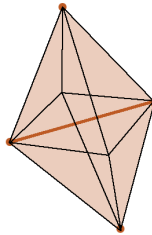


FIGURE 2.4 – Scission de la cavité en deux et génération de la nouvelle arête centrale lors d’un flip 4-4

Enfin, on est en mesure d’ajouter les deux facettes manquantes autour de l’arête centrale, qui permettront de diviser les deux cellules courantes en quatre nouveaux tétraèdres. On note qu’on ne peut ajouter une arête entre deux points que sur une facette qui comprend ces deux points. De même une facette ne peut être insérée dans un volume (pour le scinder) que s’il existe un chemin de brins appartenant au même volume et passant par les points de la facette en question. Ces contraintes permettent de garantir la validité du maillage, via la validité du Linear Cell Complex. Elles obligent à passer par certaines étapes particulières afin d’obtenir la division souhaitée, ainsi qu’on continuera de le voir par la suite.

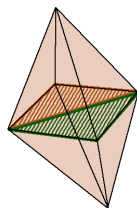


FIGURE 2.5 – Ajout des deux dernières facettes lors d'un flip 4-4

Flip bistellaire 2-3

Ce flip porte sur deux tétraèdres liés ensemble par une même face. L'idée est de les remplacer par trois tétraèdres collés ensemble par une arête. Cette opération est le symétrique du flip 3-2 dont on parlera plus tard. Elle peut notamment permettre d'éliminer des éléments trop plats. Si l'un des deux tétraèdres d'origine était dégénéré ou quasi dégénéré, à condition que l'arête formée par les deux sommets originellement indépendants soit suffisamment au centre de la facette commune aux tétraèdres, alors les trois éléments résultants de ce flip ne seront pas, ou moins aplatis. On remarque la différence d'orientation entre les tétraèdres originaux et résultants. Ici aussi, la valence de certains sommets change.

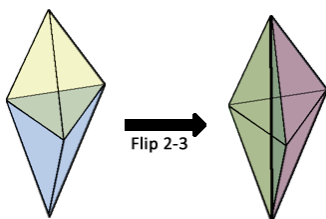


FIGURE 2.6 – Exemple de flip bistellaire 2-3 sur un ensemble basique de deux tétraèdres

La première étape de l'algorithme permettant d'effectuer ce flip sur une carte combinatoire consiste à supprimer la facette centrale. On obtient donc, comme pour l'opération précédente, une cavité « vidée ». Comme il n'est pas possible de directement insérer l'arête centrale dans le volume, il est d'abord nécessaire de créer une face entre quatre des points du volume (le fait qu'elle ne soit pas planaire ne pose pas de souci, comme les cartes combinatoires ne travaillent que sur la topologie des éléments qui la constituent). Sur cette nouvelle face, on insère l'arête. La face est scindée en deux et on obtient deux nouveaux volumes : un tétraèdre, et le « reste » de la cavité.

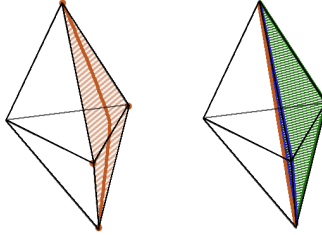


FIGURE 2.7 – À gauche, insertion d’une facette, ensuite scindée en deux par l’insertion de l’arête centrale lors d’un flip 2-3. A droite, aspect des éléments résultant de cette opération

Ne reste plus qu’à insérer une dernière facette afin de couper en deux le dernier volume non-tétraédrique.

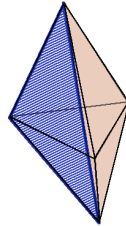


FIGURE 2.8 – Insertion de la dernière facette lors d’un flip bistellaire 2-3

Flip bistellaire 3-2

Ce flip est l’inverse de l’opération précédemment décrite. Il peut lui aussi aider à éliminer des éléments aplatis, dans le cas où l’arête centrale reliant trois tétraèdres passe sur (ou trop près) d’une de leurs autres arêtes. A condition que les sommets de l’arête centrale soient suffisamment éloignés des autres, le résultat de l’opération de flip 3-2 permettra d’obtenir des tétraèdres moins plats qu’au préalable.

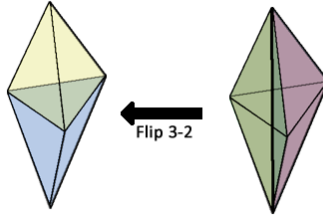


FIGURE 2.9 – Exemple de flip bistellaire 3-2 sur un ensemble basique de trois tétraèdres

La réalisation de cette opération sur un Linear Cell Complex est très simple. On supprime d’abord les faces (2-cells) autour de l’arête centrale (1-cell constituée de trois brins, liés à trois faces). On obtient ainsi la cavité « vidée ». On coupe cette cavité en deux en insérant dans le volume une facette constituée des sommets qui ne participaient pas à l’arête centrale.

Flip bistellaire 1-4

Ce flip permet de couper un tétraèdre en quatre autres, en insérant un point à l’intérieur de l’élément qu’on souhaite diviser. Il permet de réduire la taille des éléments mais risque de produire des tétraèdres trop plats puisque la distance entre les faces de l’élément d’origine (gardées intactes) et le quatrième sommet diminue. On peut aussi imaginer, grâce à la diminution de cette même distance, que dans le cas d’éléments trop allongés, cette opération pourrait permettre de générer au moins un tétraèdre plus régulier.

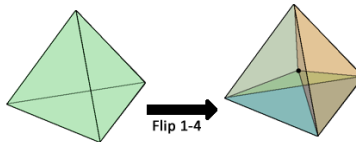


FIGURE 2.10 – Exemple de flip bistellaire 1-4 sur un tétraèdre

Sur les Linear Cell Complexes, cette opération est plus difficile à réaliser que les précédentes. D’abord, on commence par dupliquer les quatre faces du tétraèdre d’origine. Ensuite, le tétraèdre est « décousu » (on retire les liens β_2 qui relient ses faces entre-elles). On obtient donc un ensemble de huit faces égales deux à deux. Les faces d’origines sont gardées intactes. Par contre, le barycentre des faces copiées est inséré dans chacune d’entre elles, si bien qu’elles sont divisées en trois :

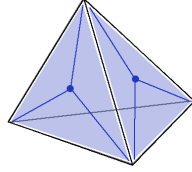


FIGURE 2.11 – Insertion du barycentre des faces copiées durant un flip bistellaire 1-4

On coud ensemble les arêtes des faces d'origine et de leur équivalent copié, ce qui permet d'obtenir quatre tétraèdres indépendants. Les faces de ces tétraèdres sont à leur tour cousues entre voisines, si bien que les sommets générés sur les quatre faces copiées lors de l'insertion de leur barycentre sont fusionnés. Enfin, on déplace le fameux sommet au niveau du barycentre du tétraèdre d'origine (il est possible de définir des poids pour chaque sommet, afin de contrôler la position du nouveau point).

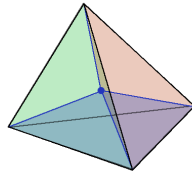


FIGURE 2.12 – Fusion des sommets de chaque tétraèdre généré durant un flip bistellaire 1-4 suite à la couture de leurs faces voisines, et positionnement du point central au niveau du barycentre du tétraèdre d'origine

Flip bistellaire 4-1

Ce flip est l'inverse du précédent. Il permet, à partir de quatre tétraèdres reliés à un même sommet central, de n'en obtenir qu'un seul. On est donc capable de générer des éléments de taille plus grossière (déaffinement), ce qui peut être utile dans des régions du maillage où peu de précision est requise. Il ne sera cependant pas forcément facile de trouver des cas dans lesquels appliquer cette opération, puisque la contrainte à respecter est particulièrement exigeante (il faut que les quatre éléments concernés forment une cavité elle-même tétraédrique).

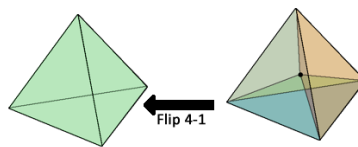


FIGURE 2.13 – Exemple de flip bistellaire 4-1 sur un ensemble basique de quatre tétraèdres

Là encore l'algorithme à appliquer est très simple : il suffit de supprimer toutes les faces (2-cells) qui sont reliées au sommet central (0-cell). Les six faces concernées retirées de la carte combinatoire, on obtient la cavité « vidée » qu'on garde en l'état puisqu'il s'agit d'un tétraèdre.

Raffinement 1-3

Ainsi qu'on l'a dit précédemment, les flips bistellaires implémentés ne permettent pas de subdiviser les faces du maillage d'origine (ou du moins pas celles qui forment l'extérieur d'une cavité). L'extérieur du maillage demeurera inchangé qu'importe l'opération appliquée sur ses éléments. Notamment dans le cas du flip 1-4, cette particularité rend le raffinement d'un maillage difficile à réaliser proprement. Afin d'avoir à disposition des outils de raffinement plus souples, et qui permettent la division des faces du maillage d'origine, on rajoute une opération de raffinement 1-3.

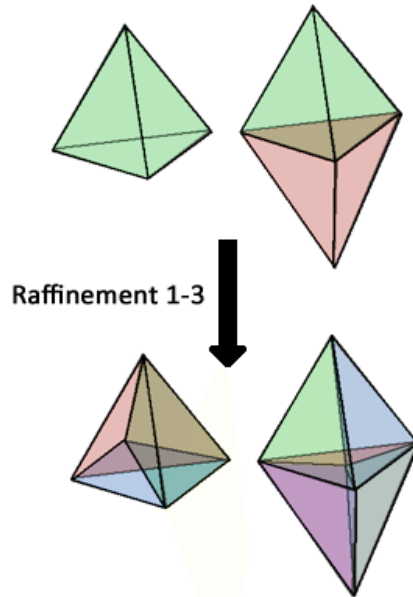


FIGURE 2.14 – Exemple de raffinement 1-3 à gauche sur un tétraèdre de surface, à droite sur deux tétraèdres internes partageant une face subdivisée

Une face à diviser est arbitrairement sélectionnée (la priorité est donnée aux faces externes). Si la face à diviser est partagée par deux éléments, ces deux éléments doivent être décousus. Dans tous les cas, on enregistre l'adresse d'un brin membre de la face en question, on calcule son barycentre, et on applique un flip 1-4 sur l'élément. Le brin précédemment stocké fera dorénavant partie d'un tétraèdre (parmi les quatre obtenus) qu'on souhaite supprimer, ce qu'on fait donc sans difficulté puisqu'on a un accès direct dessus. L'ancien « point central » généré par l'opération de flip 1-4 est déplacé sur la position du barycentre dont on a obtenu les coordonnées juste avant. On obtient trois tétraèdres qui remplissent l'espace de l'ancien.

Dans le cas où la face était partagée par deux tétraèdres, on doit faire exactement la même chose sur l'élément voisin. Trois coutures de face doivent enfin être réalisées pour rétablir les liens de voisinage entre les éléments.

2.2.2 Fonctions de vérification de l'applicabilité d'une opération sur un élément

Pour chacune des opérations précédemment décrites, on doit s'assurer (avant de l'appliquer) qu'elle est bien réalisable sur l'élément sélectionné. Dans certains cas, la topologie du sous-maillage concerné n'est pas compatible avec celle que sous-entend l'opération. On pourra aussi chercher à empêcher la génération d'éléments dégénérés, ou bien à détecter certains autres types de configurations géométriques dans lesquelles l'opération de flip ne sera pas valide, et ne doit donc pas être effectuée.

Validité du flip bistellaire 1-4

Comme elle ne porte que sur un unique élément, cette méthode est la plus simple parmi les cinq proposées. Elle met en œuvre deux mécanismes de vérification basiques qu'on retrouvera souvent dans les méthodes suivantes. En premier lieu on doit vérifier que l'élément auquel on souhaite appliquer l'opération est bien un tétraèdre (ce qui est réalisable grâce à une méthode fournie par les cartes combinatoires de CGAL). Si l'élément est dégénéré, il sera remplacé par quatre éléments dégénérés, ce qui n'est pas utile. On considère donc que l'opération n'est pas réalisable sur un élément plat. Si les quatre points qui le constituent sont coplanaires (facile à vérifier en utilisant les fonctions fournies par CGAL) alors le tétraèdre est plat, et n'est donc pas « flipable ».

Validité du flip bistellaire 4-4

On part d'une arête centrale, laquelle doit être partagée par exactement quatre éléments (sans quoi la topologie locale n'est pas la bonne est il ne sera de toute façon pas possible d'effectuer le flip). Les quatre volumes concernés doivent être des tétraèdres. Les quatre volumes qui résulteront de l'opération de flip ne doivent pas être dégénérés (ce qu'on vérifie en testant la coplanarité des sommets qui formeront les nouveaux éléments). Les sommets opposés deux à deux doivent être différents les uns des autres. Enfin on crée quatre marques de valeur différente qu'on applique aux brins de chacun des quatre volumes. Pour les quatre faces autour de l'arête centrale, on vérifie que les liens β_3 permettent bien d'accéder au tétraèdre attendu (on teste donc la valeur de la marque associée au brin pointé, afin de s'assurer que la façon dont sont « branchés » les éléments les uns aux autres correspond bien à ce qu'on attend).

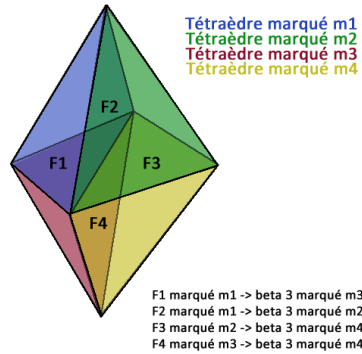


FIGURE 2.15 – Marquage des quatre tétraèdres d’origine et égalités de marque à vérifier pour qu’un flip bistellaire 4-4 soit réalisable dessus

Validité du flip bistellaire 2-3

On part d’une face qui doit être partagée par deux éléments exactement. Ces deux éléments doivent être des tétraèdres. Une fois qu’on s’en est assuré, on vérifie que les sommets « libres » des deux tétraèdres sont bien différents l’un de l’autre (sans quoi on se trouve dans le cas d’un élément doublé). On crée une marque qu’on associe aux brins d’un des deux éléments. On vérifie que chacun des liens β_3 du tétraèdre non marqué placé sur la face commune accède bien à un brin marqué (ainsi on s’assure que toutes les arêtes de la dite face sont partagées par les deux même tétraèdres).

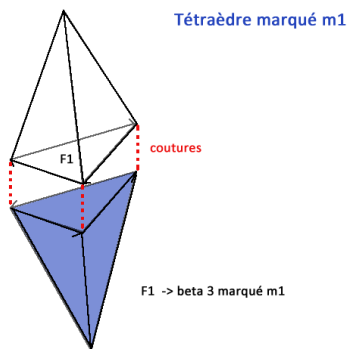


FIGURE 2.16 – Marquage des deux tétraèdres d’origine et égalités de marque à vérifier pour qu’un flip bistellaire 2-3 soit réalisable dessus

Enfin, l'opération de flip 2-3 n'est possible que si le segment formé par les sommets des tétraèdres qui ne sont pas placés sur leur face commune intersectent la dite face commune. Sinon, l'opération modifierait la forme de la cavité et risquerait de produire des cas d'auto-intersection dans le maillage, ce qui n'est pas souhaitable.

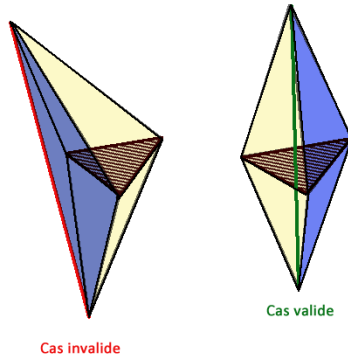


FIGURE 2.17 – Mise en évidence de l'importance de la position des sommets des tétraèdres d'origine pour la validité d'une opération de flip 2-3

Le segment en question ne doit par contre pas intersecter les arêtes de la face commune, sans quoi au moins un des tétraèdres résultant du flip serait dégénéré.

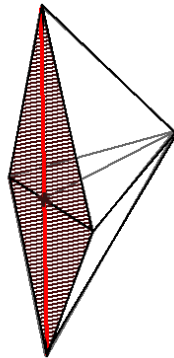


FIGURE 2.18 – Génération d'un tétraèdre dégénéré suite à un flip 2-3 si la nouvelle arête centrale intersecte l'une des arêtes de l'ancienne facette commune

Validité du flip bistellaire 3-2

On part d'une arête qui doit être partagée par exactement trois éléments. Ces éléments doivent être des tétraèdres. Les sommets formant les futurs éléments ne doivent pas être coplanaires (pour éviter les tétraèdres dégénérés). Une fois encore on va associer à chaque tétraèdre une marque différente et vérifier que leurs faces sont cousues de la bonne manière (tous les brins d'une des faces interne du volume 1 avec des brins du volume 2, tous les brins de l'autre face interne du volume 1 avec des brins du volume 3, et enfin les brins de la face restante du volume 2 avec des brins du volume 3).

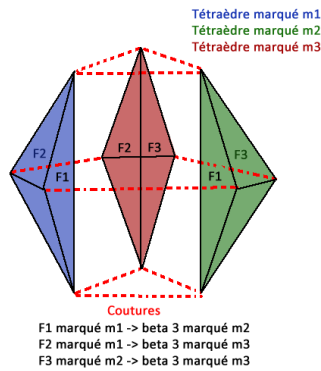


FIGURE 2.19 – Marquage des trois tétraèdres d'origine et égalités de marque à vérifier pour qu'un flip bistellaire 3-2 soit réalisable dessus

Validité du flip bistellaire 4-1

On part d'un sommet central. Ce sommet doit avoir une valence exactement égale à 4, sinon, on est d'ores et déjà sûr de ne pas se trouver dans une configuration compatible. Le sommet doit être relié à exactement quatre éléments, qui doivent être des tétraèdres. Ensuite, on associe une marque différente à chaque tétraèdre et on va vérifier que ces derniers partagent bien leurs faces selon le motif précis qui garantira la validité du flip (si ce motif est respecté, on est certain que la cavité formée par les quatre éléments est tétraédrique).

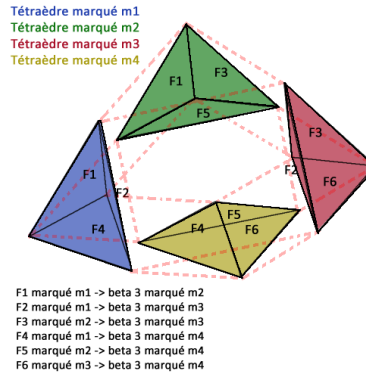


FIGURE 2.20 – Marquage des quatre tétraèdres d’origine et égalités de marque à vérifier pour qu’un flip bistellaire 4-1 soit réalisable dessus

2.3 Premiers algorithmes de remaillage

2.3.1 Remaillage a posteriori

Dans un premier temps, une simulation est réalisée sur un maillage original. La simulation calcule a posteriori un critère compris entre 0 et 1 pour chaque élément. Un fichier VTK contenant le maillage ainsi que la valeur du critère pour chaque élément est généré. C’est à partir de ce fichier qu’on va vouloir effectuer le remaillage, en prenant en compte la valeur du critère pour savoir si un élément donné a besoin d’être raffiné, ou non.

La valeur maximale de critère est extraite. On décide qu’un élément sera raffiné si sa valeur de critère dépasse un certain pourcentage de cette valeur maximale (80%). Les éléments du maillage sont parcourus une seule fois. Ceux qui doivent être raffinés sont listés. Des opérations de remaillage local sont effectuées sur les tétraèdres de la dite liste.

On a d’abord choisi de ne raffiner les éléments qu’avec des flips 1-4 (qui ne modifient qu’un seul élément à la fois). Cependant, ces flips gardent les faces communes entre les éléments d’origine intactes et ne permettent pas de raffiner le volume suffisamment, si bien qu’on a ensuite remplacé cette opération par le raffinement 1-3, qui permet de diviser certaines des faces d’origine des tétraèdres en trois.

Une fois le remaillage effectué, son résultat est enregistré dans un nouveau fichier .vtk sur lequel on effectuera une nouvelle simulation afin de recalculer le critère pour chaque élément. L’étape de remaillage pourra être réitérée avec ces nouvelles données jusqu’à convergence.

2.3.2 Perspectives

À l'heure actuelle, l'algorithme de raffinement 1-3 est appliqué sur le tétraèdre dans un sens arbitraire, et donne la priorité aux faces externes. Par la suite, afin d'optimiser la qualité du raffinement, on envisage de choisir la face à diviser en prenant en compte certains critères tels que l'*aspect ratio* des éléments générés et la valence résultante des sommets.

L'utilisation des autres opérations implémentées (flips 2-3, 3-2, 4-4, 4-1) sera éventuellement intégrée au processus. Une réflexion doit être portée sur la façon de combiner ces opérations afin d'être en mesure de les exploiter de la façon la plus pertinente possible. On doit réfléchir à la priorité avec laquelle les appliquer, ainsi qu'au choix des opérations qu'il sera en définitive utile de garder vis-à-vis de la qualité des résultats obtenus.

Pour le moment, on envisage d'évaluer la qualité des éléments via le calcul de leur *aspect ratio* et de leur jacobien. D'autres critères pourraient être intégrés, comme par exemple la valence qui, pour le bon déroulement de la simulation, doit rester la plus égale possible et varier graduellement plutôt que brutalement entre sommets voisins.

Chapitre 3

Résultats expérimentaux

3.1 Résultats des premiers tests des opérations de remaillage

L'implémentation des opérations de remaillage local précédemment citées a tout d'abord été testée sur des cas très simples afin de vérifier d'une part qu'elles effectuaient bien les modifications attendues (voir résultats visuels des tests en annexe), et d'autre part qu'elles étaient bien symétriques (ou symétriques deux à deux selon les cas). La connectivité des éléments ainsi que les coordonnées de leurs sommets ont été modifiées de sorte à vérifier que les fonctions de vérification « isFlipable » détectaient bien les cas à problèmes, et empêchaient le flip lorsqu'il n'était pas possible de le réaliser correctement. Enfin, les tétraèdres dont les faces sont flippées ont été « plongés » dans un maillage simple, de sorte à ce que les faces de la cavité qu'ils forment soient partagées avec d'autres éléments. Ces tests ont permis de vérifier que les liens de voisinage entre éléments à la frontière de la cavité remaillée étaient bien préservés.

Suite à ces vérifications atomiques, une boucle de tests a été codée afin de mettre à l'épreuve les opérations de remaillage local sur de véritables données, et sur un grand nombre de configurations différentes. L'algorithme permet à l'utilisateur de choisir le nombre de flips qu'il veut réussir à effectuer en tout (les tentatives ratées pour cause de configuration « non flippable » n'incrémentant pas le compteur). Les éléments du maillage sont parcourus. On tente tout d'abord d'effectuer un flip bistellaire 4-4. Si ce n'est pas possible, on tente un flip 4-1, puis un flip 3-2, puis un flip 2-3 et enfin un flip 1-4 (l'ordre de priorité a été défini de sorte à ce que les opérations prioritaires soient celles qui ne modifient pas le nombre d'éléments dans le maillage, puis celles qui le diminuent, puis celles qui l'augmentent. L'opération de flip 1-4 vient en dernier car elle est théoriquement toujours réalisable. De cette manière, même si on choisit d'effectuer un grand nombre de flips, on évite au possible de trop alourdir la structure de données). La boucle de test a été effectuée pour un nombre de flips $N = 10000$ sans rencontrer d'erreurs. L'opération de flip 1-3 / 2-6 apparaît dans une autre boucle de test.

Dans l'optique d'en venir progressivement à la première boucle de remaillage a posteriori (à base de flips 1-4), la boucle de tests a été modifiée pour se contenter de réaliser ce type de flips. Pour une meilleure visibilité du résultat et pour éviter d'effectuer le raffinement dans des régions trop proches les unes des autres on a modifié le parcours des éléments pour qu'il ne suive plus l'ordre des brins donné par les itérateurs, mais soit aléatoire. L'amélioration étant insuffisante, on a encore modifié la sélection successive des éléments en faisant en sorte de toujours raffiner l'élément de plus grand volume. Pour un maillage original de 573 éléments tétraédriques, au bout de 60 flips 1-4, on obtient un maillage raffiné de 753 éléments dont les plus grands éléments ont été scindés.

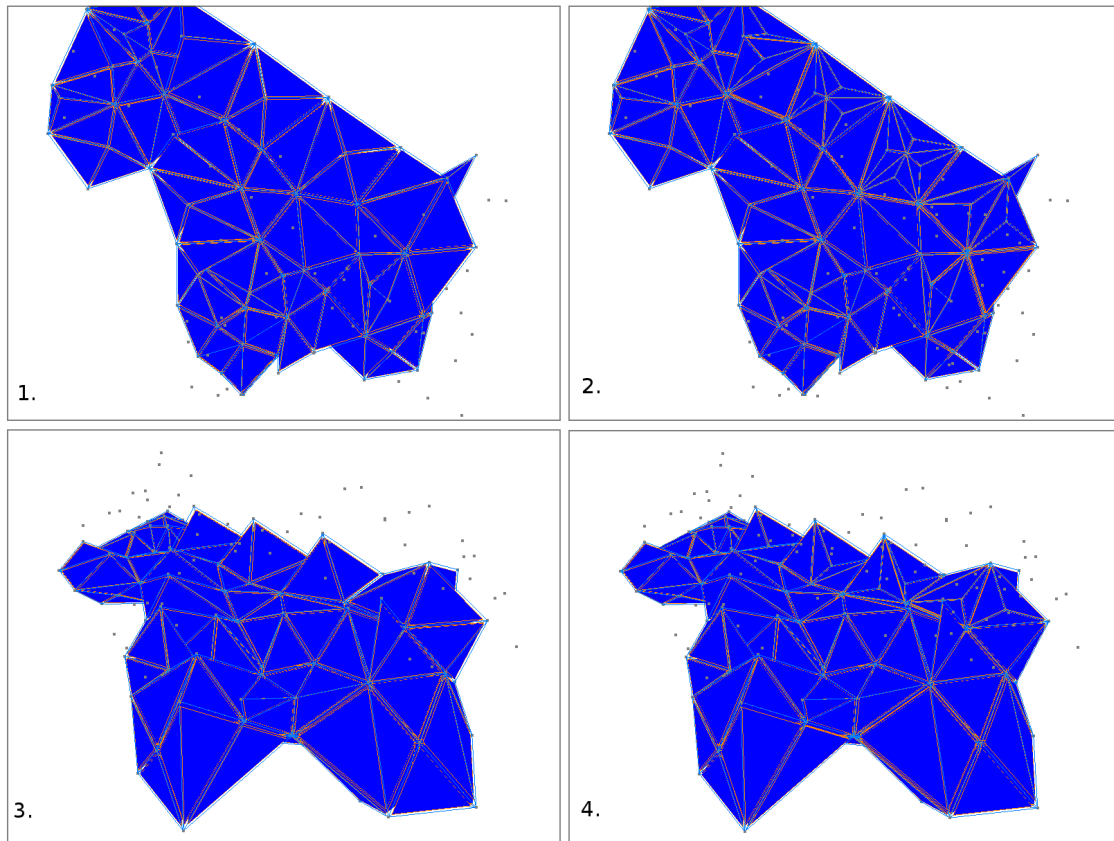


FIGURE 3.1 – 1. Première vue en coupe d'un maillage original - 2. Première vue en coupe du maillage après 60 flips 1-4 - 3. Seconde vue en coupe du maillage original - 4. Seconde vue en coupe du maillage après 60 flips 1-4

3.2 Résultats des premières simulations et modifications envisagées

On simule la déformation d'une poutre fixée sur un mur. On connaît la solution analytique de ce problème, si bien qu'on est capable de calculer l'erreur en comparant la dite solution avec le résultat de la simulation. De plus, cet exemple permet de tester quasiment tous les cas critiques qu'on pourrait rencontrer lors d'une autre simulation (compression de certains éléments, rotation). Une première boucle de remaillage a posteriori a été implémentée comme on l'a expliqué dans la section 2.3.1. Cette boucle n'effectuait que des flips 1-4 sur les éléments jugés mauvais en fonction du critère fourni par les résultats de la simulation. Le flip 1-4 ayant tendance à aplatir les éléments raffinés, dans une optique de remaillage adaptatif, un critère basé sur l'*aspect ratio* n'était pas pertinent (car les nouveaux éléments étant plus aplatis que les anciens, on aura tendance à toujours raffiner au même endroit). Un critère basé sur le stress (forces exercées sur un élément pour le déformer) pondéré par le volume des éléments permet une meilleure sélection des éléments à remailler. Cependant, le flip bistellaire 1-4 appliqué seul ne permet pas de diminuer l'erreur de simulation. Certaines valences particulières deviennent très élevées et les performances diminuent rapidement suite à quelques itérations successives de la simulation et de la boucle de remaillage. On remarque notamment que les facettes d'origine du maillage sont toujours gardées intactes et ne seront jamais raffinées. Le raffinement ne se produit qu'à l'intérieur d'un élément particulier, ce qui entraîne la génération de nombreux éléments mal formés car très irréguliers. L'idée de remplacer le flip 1-4 par un raffinement 2-3 (qui scinde une facette en trois) a été amenée pour tenter de corriger ce souci. On est dorénavant capable de scinder les faces originelles, même si on a pour l'instant pas optimisé le choix des faces à couper prioritairement. Des résultats devraient être obtenus rapidement, la nouvelle boucle de remaillage étant tout juste déboguée.

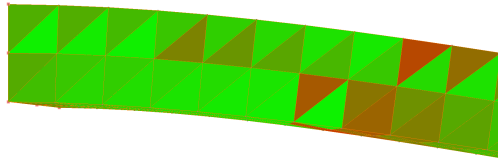


FIGURE 3.2 – Aspect extérieur du maillage suite à dix simulations et dix remaillages successifs (similaire à l'aspect extérieur du maillage original). En vert, les éléments jugés bon selon le critère utilisé par la simulation, et en rouge les éléments jugés mauvais.

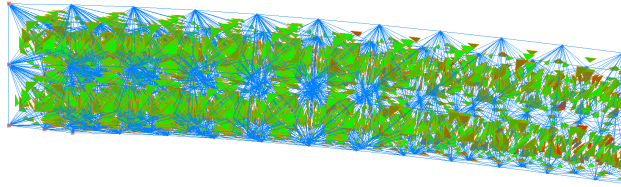


FIGURE 3.3 – Vue éclatée du maillage permettant de voir la répartition des éléments jugés mauvais par le critère utilisé par la simulation, suite à dix simulations et dix remaillages successifs

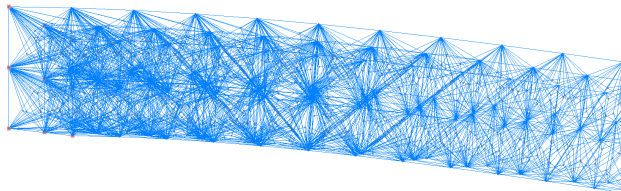


FIGURE 3.4 – Vue fil de fer du maillage suite à dix simulations et dix remaillages successifs. On constate qu'il y a eu plus de raffinement du côté gauche de la poutre, soumis à plus de contraintes physiques que le droit

Conclusion

Nous avons programmé des algorithmes de remaillage a posteriori pour réaliser des simulations mécaniques portant sur des maillages volumiques (et dans un premier temps, tétraédriques). Pour ce faire, nous avons utilisé les Linear Cell Complexes, une structure de données CGAL basée sur les cartes combinatoires (extension dans toutes les dimensions de la structure demi-arêtes). Cette structure est adaptée aux modifications topologiques locales et permet d'assurer que les données manipulées restent conformes. Plusieurs opérations de flip de faces ont été implémentées. Leur intérêt pour améliorer effectivement le maillage volumique utilisé dans une simulation mécanique a été partiellement étudié (on a notamment détecté certains cas dans lesquels ces flips peuvent éliminer ou générer des éléments dégénérés ou trop plats et incidemment, mal formés). Les critères permettant d'évaluer la qualité des éléments (et justement de dire si ils sont bien ou mal formés) ont été étudiés et pour certains implémentés : *aspect ratio*, jacobien. La simulation générant les valeurs de certains autres critères tels que le stress, nous avons pu étudier l'influence de cette donnée sur le remaillage adaptatif d'une poutre dans le cas d'un raffinement uniquement effectué via des flips bistellaires 1-4, jugés finalement peu adaptés.

La pertinence des différents critères cités doit encore être évaluée et le sera dès lors que nous serons en mesure d'effectuer un meilleur raffinement sur les mauvais éléments détectés par la simulation. Une boucle de remaillage appliquant arbitrairement des flips 1-3 sur les mauvais éléments remplacera d'abord la précédente. Elle sera ensuite modifiée de sorte à ce que le choix de la face scindée maximise la valeur minimale de l'*aspect ratio* des futurs éléments générés. On envisagera l'intégration d'autres types d'opérations de remaillage dans la boucle (flips 2-3, 3-2...). L'influence de certains autres critères tels que la régularité et l'évolution graduelle des valences, ou l'anisotropie des éléments, pourront par le futur être étudiés. Enfin, on tentera de passer d'un processus de remaillage a posteriori à un processus de remaillage dynamique en intégrant l'étape de raffinement à la simulation, pour améliorer les éléments jugés mauvais à chacun de ses pas

Bibliographie

- [ACSYD05] Pierre Alliez, David Cohen-Steiner, Mariette Yvinec, and Mathieu Desbrun. Variational tetrahedral meshing. *ACM Transactions on Graphics*, 24(3) :617, July 2005.
- [BCF06] 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(48-49) :6626–6645, October 2006.
- [BEE02] Marshall Bern, David Eppstein, and Jeff Erickson. Flipping cubical meshes. *Engineering with Computers*, (Theorem 5) :1–20, 2002.
- [BHR09] Chris J. Budd, Weizhang Huang, and Robert D. Russell. *Adaptivity with moving grids*, volume 18. May 2009.
- [BLPH10] Marek Bucki, Claudio Lobos, Yohan Payan, and Nancy Hitschfeld. Jacobian-based repair method for finite element meshes after registration. *Engineering with Computers*, 27(3) :285–297, November 2010.
- [Cou00] Thierry Coupez. Génération de maillage et adaptation de maillage par optimisation locale. *Revue Européenne des Éléments*, 9(4) :403–423, 2000.
- [Fie00] David A Field. Qualitative measures for initial meshes. 906(May 1998) :887–906, 2000.
- [FM07] J.L.M. Fernandes and P.a.F. Martins. All-hexahedral remeshing for the finite element analysis of metal forming processes. *Finite Elements in Analysis and Design*, 43(8) :666–679, May 2007.
- [FOG97] L. Freitag and C. Ollivier-Gooch. Tetrahedral Mesh Improvement Using Swapping and Smoothing. *Fourth International Meshing Roundtable*, 1997.
- [FPR⁺00] Sarah F Frisken, Ronald N Perry, Alyn P Rockwood, Thouis R Jones, Sarah F Frisken, Ronald N Perry, Alyn P Rockwood, and Thouis R Jones. Adaptively Sampled Distance Fields : A General Representation of Shape for Computer Graphics Abstract Adaptively Sampled Distance Fields (ADFs) are a unifying representation of shape that Adaptively Sampled Distance Fields : A General Representation of Shape for Computer Graphics. 2000.
- [GBF⁺97] P L George, H Borouchaki, P J Frey, P Laug, and E Saltel. Chapter 17 Mesh Generation and Mesh Adaptivity : Theory and Techniques. In *History*, pages 1–31. 1997.
- [ITF04] G Irving, J Teran, and R Fedkiw. Invertible Finite Elements For Robust Simulation of Large Deformation. 2004.
- [JP] Mark T Jones and Paul E Plassmann. FINITE-ELEMENT MESHES. pages 1–22.

- [Knu99] Patrick M. Knupp. Matrix norms and the condition number : A general framework to improve mesh quality via node-movement. In *Eighth International Meshing Roundtable (Lake Tahoe, California)*, pages 13–22, 1999.
- [KS07] BM Klingner and JR Shewchuk. Aggressive tetrahedral mesh improvement. *... of the 16th international meshing roundtable*, 2007.
- [LLD12] Hongwei Lin, Hongwei Liao, and Chongyang Deng. Filling Triangular Mesh Model with All-Hex Mesh by Volume Subdivision Fitting. Technical report, 2012.
- [MMG00] M Murphy, DM Mount, and CW Gable. A point-placement strategy for conforming Delaunay tetrahedralization. *International Journal of ...*, 7(0), 2000.
- [Ode91] J Tinsley Oden. Finite Elements : An Introduction. II(Part 1) :3–15, 1991.
- [She] Jonathan Richard Shewchuk. Two Discrete Optimization Algorithms for the Topological Improvement of Tetrahedral Meshes. pages 1–11.
- [She11] JR Shewchuk. Unstructured mesh generation. *Combinatorial Scientific Computing*, pages 259–299, 2011.
- [SOB05] ML Staten, SJ Owen, and TD Blacker. Unconstrained paving & plastering : a new idea for all hexahedral mesh generation. *Proceedings of the 14th International ...*, 2005.
- [WRK⁺10] Martin Wicke, Daniel Ritchie, Bryan M. Klingner, Sebastian Burke, Jonathan R. Shewchuk, and James F. O’Brien. Dynamic local remeshing for elastoplastic simulation. *ACM SIGGRAPH 2010 papers on - SIGGRAPH ’10*, page 1, 2010.
- [ZZ92] 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) :1331–1364, 1992.

Annexes

Résultat visuel des tests sur les opérations de remaillage local

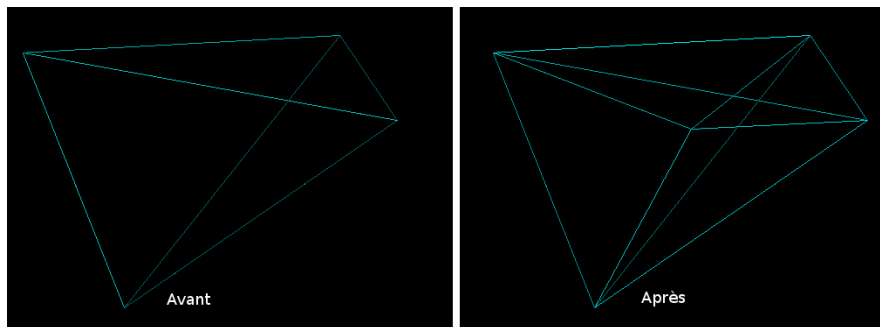


FIGURE 3.5 – Résultat d'un flip bistellaire 1-4

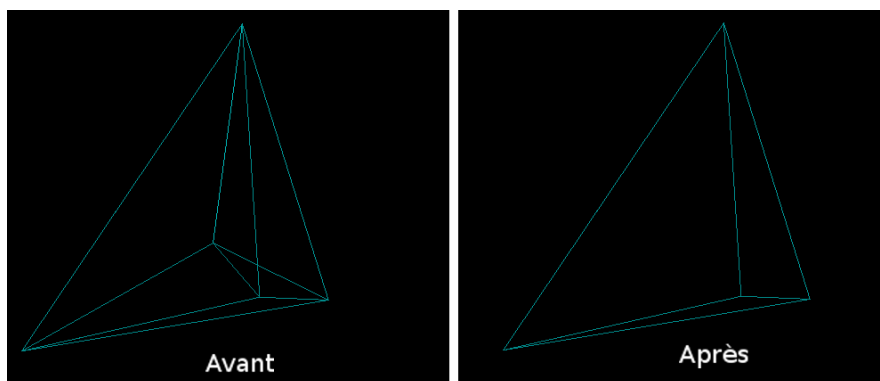


FIGURE 3.6 – Résultat d'un flip bistellaire 4-1

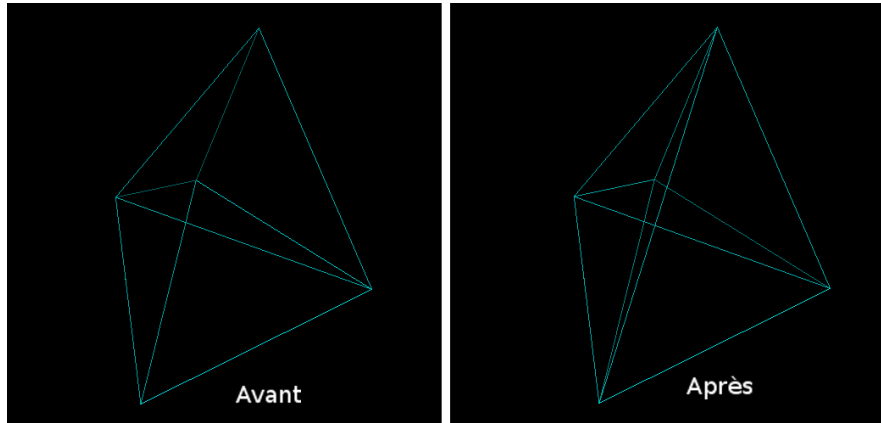


FIGURE 3.7 – Résultat d'un flip bistellaire 2-3

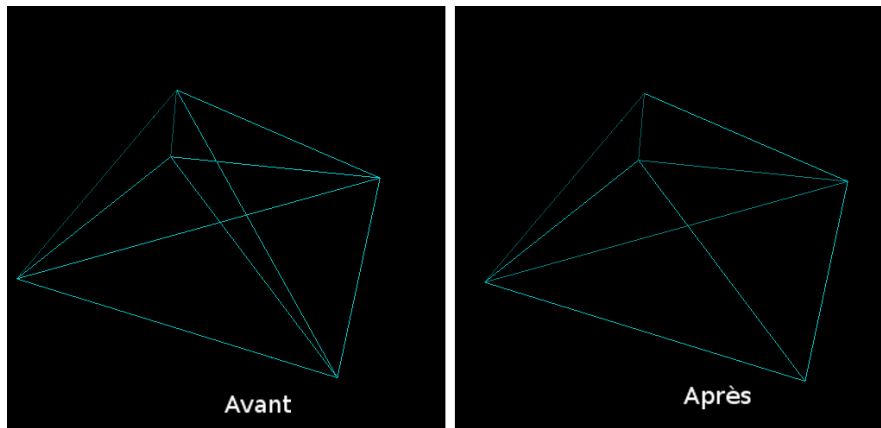


FIGURE 3.8 – Résultat d'un flip bistellaire 3-2

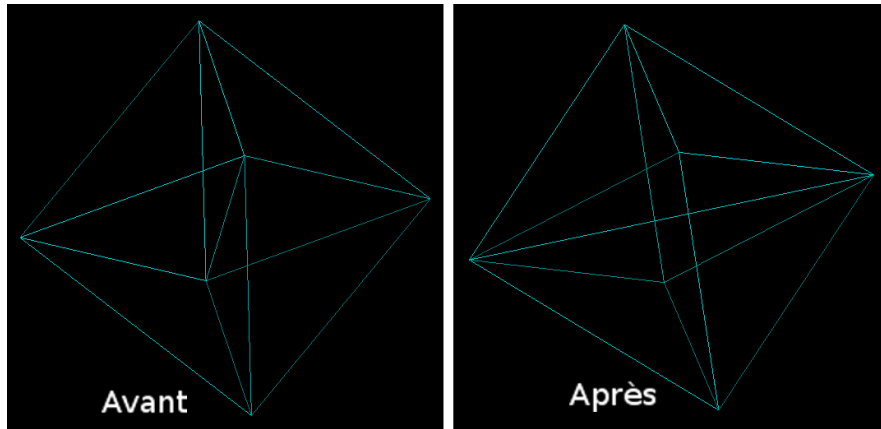


FIGURE 3.9 – Résultat d'un flip bistellaire 4-4

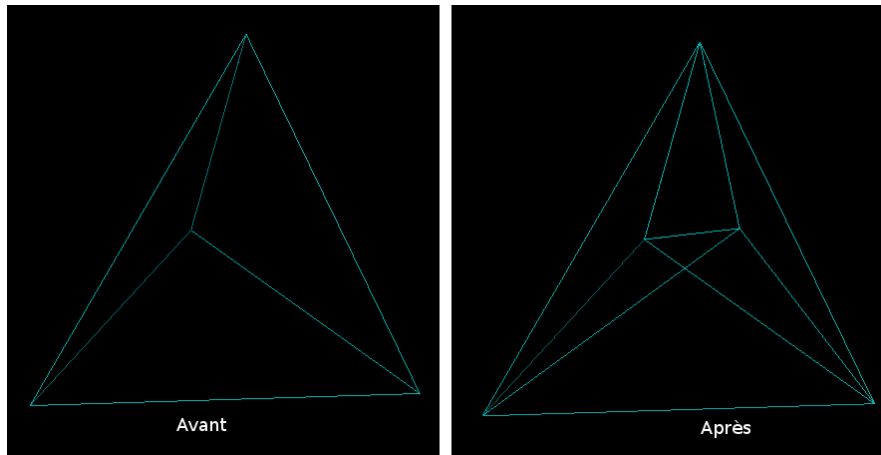


FIGURE 3.10 – Résultat d'un raffinement 1-3 sur une face externe

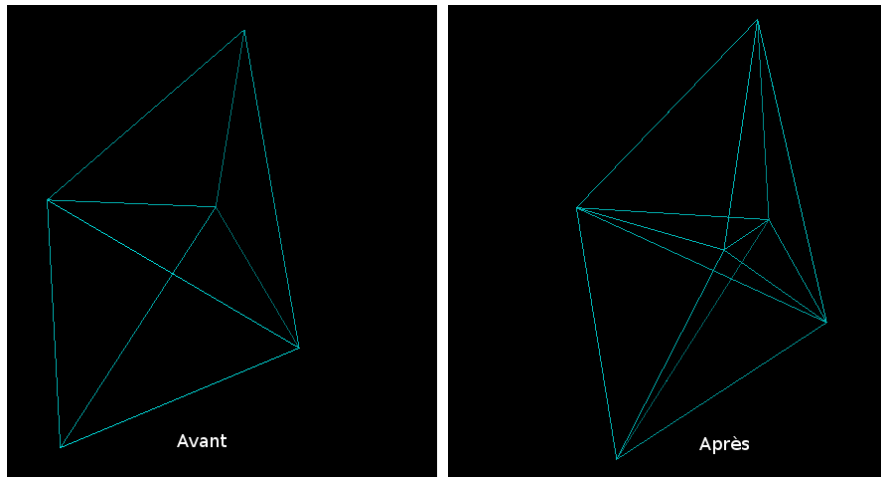


FIGURE 3.11 – Résultat d'un raffinement 1-3 sur une face partagée par deux tétraèdres