

Représentation efficace des maillages volumiques multirésolutions

Lionel Untereiner, David Cazier et Dominique Bechmann

LSIT, Université de Strasbourg, CNRS ; France
{lionel.untereiner, david.cazier, bechmann}@unistra.fr

Résumé

La représentation multirésolution d'objets modélisés est devenue monnaie courante ces dernières années en informatique graphique. On peut notamment citer les surfaces de subdivision régulièrement abordées en modélisation géométrique dans le cadre de la conception assistée par ordinateur. Ce type de représentation est encodé grâce à des structures de données adaptées à la manipulation d'une hiérarchie de maillages. À l'inverse, cette question de la représentation multirésolution et adaptative des volumes de subdivision reste ouverte pour des maillages volumiques décomposés en un ensemble de polyèdres.

Dans ce travail, nous proposons une nouvelle structure de données pour la représentation et la manipulation efficace de volumes de subdivision multirésolution. Notre modèle topologique permet une subdivision adaptative du maillage et une navigation instantanée et efficace à tous les niveaux de la hiérarchie de maillages. Sa représentation est formalisée dans le cadre des cartes combinatoires pour nous permettre de donner une formulation générale des changements topologiques.

Mots clé : informatique graphique, modélisation géométrique, modèles topologiques, cartes combinatoires, maillages multirésolution, volumes de subdivision

1. Introduction

La multiplication des applications liées à la modélisation de phénomènes physiques est une des raisons de la popularité croissante des maillages volumiques en informatique graphique. On rencontre des travaux portant sur la simulation de propriétés de matériaux tel la simulation élastoplastique utilisant des éléments finis dans un maillage tétraédrique [WRK*10], ou encore des éléments finis isogéométriques dans un maillage hexaédrique [BHU10b]. En animation, on trouve des travaux concernant la détection de collision qui exploitent un partitionnement volumique de l'espace pour en améliorer l'interactivité [JCD09]. Ce type de maillage est également utilisé comme maillage de contrôle dans le cadre de la déformation de forme libre [MJ96, SHW04]. Enfin, quelques travaux [BSWX02, MCQ04] se sont penchés plus spécifiquement sur la subdivision, et plus particulièrement sur la géométrie engendrée par la subdivision d'un maillage hexaédrique ou tétraédrique. Leurs auteurs se concentrent sur la convergence et la continuité des schémas de subdivision sans aborder les questions liées au raffinement des maillages.

Par ailleurs, les modèles multirésolutions comme les surfaces de subdivision [VZ01] ou les maillages progressifs [DFKP05] sont devenus très populaires. Dans ce mode de représentation, l'objet n'est plus défini par un maillage unique mais par une séquence de maillages imbriqués, ainsi que par les relations permettant d'évoluer dans cette séquence. Cette représentation est dite adaptative si le niveau de résolution n'est pas le même dans toutes les zones de l'objet. En comparaison, la représentation de volumes de subdivision a rarement été abordée. Une des raisons en est que les règles de raffinement sont plus difficiles à décrire et énumérer. En effet, il n'existe pas de méthode canonique pour subdiviser un polyèdre quelconque. Beaucoup de travaux se sont donc concentrés sur les modèles tétraédriques ou hexaédriques. Même dans ce cadre simplifié, l'adaptativité engendrerait la création de maillages non conformes qui ne sont pas représentables avec les structures de données usuelles.

Pourtant, de nombreuses applications [CGC*02, BHU10b] exploiteraient avantageusement une représentation générique de volumes de subdivision. Une telle structure de données volumique devrait être générale et efficace, permettre un accès direct et simple à tous les niveaux de résolution. Elle devrait supporter une subdivision

adaptative des volumes et une large gamme de schémas de subdivision.

1.1. État de l’art

S’il n’existe pas de modèle général pour des polyèdres multirésolution, de nombreux travaux se sont intéressés aux maillages tétraédriques.

Dans ce cadre, une hiérarchie régulière peut-être générée par décimation d’un maillage en utilisant un opérateur de contraction d’arête (cf. figure 1). Le half-edge tree proposé dans [DdFM*05] permet la représentation de ces hiérarchies. Les contractions d’arêtes sont encodées dans une forêt binaire de sommets. Historiquement ces travaux sont parmi les premiers à avoir proposé une véritable gestion multirésolution de maillages tétraédriques. Néanmoins, l’opérateur de contraction d’arête ne correspond pas à la subdivision d’un volume et de ce fait s’éloigne un peu de notre domaine d’application.

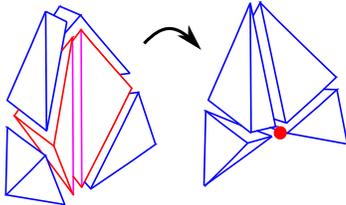


Figure 1: Opération de contraction d’une arête

D’autres approches permettent la construction de hiérarchies de tétraèdres, par exemple en exploitant la bisection des tétraèdres le long de leur plus longue arête [LWC*02]. La figure 2 illustre ce principe. De telles hiérarchies sont usuellement encodées dans des forêts binaires de tétraèdres (cf. figure 4a avec un exemple 2D) ou dans des diagrammes acycliques orientés (DAO) de diamants [WDF11]. Dans ces derniers, un diamant correspond à l’ensemble des simplexes incidents à l’arête bissectée (cf. figure 3). La relation entre les triangles (en 2D) ou tétraèdres (en 3D) générés par l’opération de bisection décrit une relation hiérarchique de dépendance entre les diamants, de laquelle est extraite une hiérarchie de diamants (cf. figure 4b avec un exemple 2D).

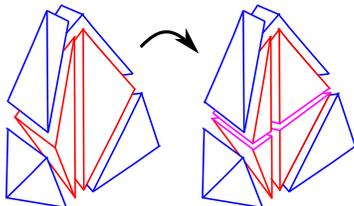


Figure 2: Opération de bisection d’une arête

D’autres structures sont utilisées dans le contexte précis

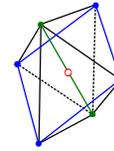


Figure 3: Exemple d’un type de diamant en dimension 3 encodant les tétraèdres autour de l’arête bissectée (en vert)

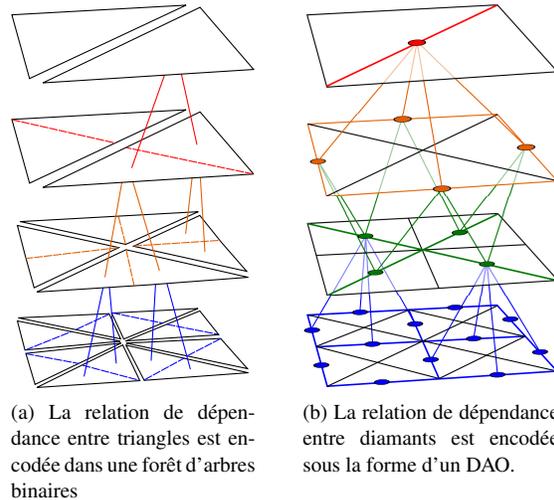


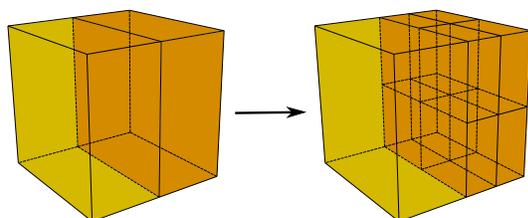
Figure 4: Comparaison en 2D entre deux modèles dont la hiérarchie est générée par une opération de bisection d’arête

de la subdivision de volumes mais sans la notion de niveaux de résolution. Seul le niveau de résolution le plus fin est représenté. McDonnell et al. [MCQ04] utilisent une version simplifiée de la structure radial-edge qui est une généralisation de la structure winged-edge aux complexes arbitraires. Bajaj et al. [BSWX02] utilisent une représentation récursive d’un cube de dimension d (un d -cube) où un 0-cube est un index de sommet et où un d -cube ($d > 0$) est une liste de deux $(d - 1)$ -cubes.

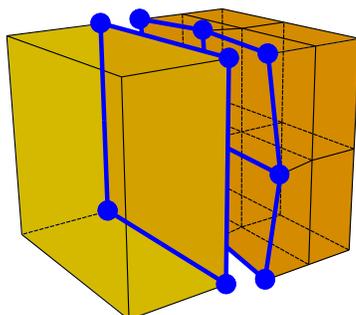
Parallèlement, une solution classique pour encoder les surfaces de subdivision multirésolution issues de schémas quaternaires consiste à utiliser un quadtree en tant que structure de données [DFKP05]. Cette solution peut être étendue au cas volumique pour encoder les volumes de subdivision multirésolution en utilisant un octree, qui est naturellement dérivé de la hiérarchie de volumes imbriqués générée par le processus de subdivision. Seulement, cette structure n’est pas utilisée en pratique pour les volumes de subdivision multirésolution car elle ne permet pas de représenter toutes les cellules du maillage.

Tous les modèles mentionnés sont limités à la représentation de polyèdres spécifiques (hexaèdres ou tétraèdres) dont le degré des faces est figé (carré ou triangle). D’une part, cela interdit des subdivisions adaptatives qui provoqueraient

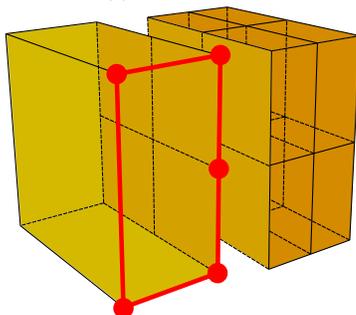
l'apparition de faces non conformes (cf. figure 5). D'autre part, cela limite l'éventail des règles de raffinement de la topologie du maillage car celles-ci doivent assurer que les volumes générés sont représentables par le modèle, c'est-à-dire en pratique des hexaèdres ou des tétraèdres. Toutes ces contraintes expliquent la difficulté de voir émerger des structures de données pour la représentation de volumes de subdivision générique.



(a) Raffinement adaptatif d'un hexaèdre



(b) Dans un octree



(c) Dans une 3-carte

Figure 5: Gestion de l'adaptativité pour le raffinement d'un hexaèdre (5a). Avec une représentation de type octree (5b), les faces non conformes générées provoquent l'apparition d'un trou topologique visible en bleu (les sommets incidents à ce trou ont été déplacés pour les rendre visible). Dans une 3-carte (5c) gérant des polyèdres quelconques, la conformité est maintenue. Un polyèdre à 9 faces est créé et naturellement géré. Ce polyèdre comprend 5 faces à 4 côtés et 4 faces à 5 côtés (dont une visible en rouge figure 5c).

1.2. Notre contribution

Dans cet article, nous présentons un modèle général pour des polyèdres multirésolution. Basé sur une extension multirésolution des cartes combinatoires de dimension 3, ce modèle permet de définir une série de maillages volumiques imbriqués, constituant des niveaux de résolution différents avec des règles permettant de passer de l'un à l'autre. Chaque niveau est défini par un ensemble de polyèdres quelconques assemblés le long de faces communes et encodé dans une 3-carte. Ce modèle est générique de par sa capacité à représenter une très large variété de maillages. Ce modèle est également adaptatif, c'est-à-dire qu'il permet un raffinement à une profondeur variable suivant les zones du maillage.

Il est possible d'accéder de manière indépendante à chaque niveau de la hiérarchie de polyèdres. Au sein de celle-ci, un niveau est construit à partir de niveaux plus grossiers en leur appliquant des opérateurs de subdivision. Ce modèle supporte une large gamme d'opérateurs de subdivision qui ne sont pas limités à une classe spécifique de polyèdres. Il est par exemple possible de subdiviser un tétraèdre en hexaèdres ou une pyramide en une combinaison de tétraèdres et de prismes.

La combinaison de ces propriétés nous permet de gérer élégamment des subdivisions adaptatives sans problèmes de conformité (cf. figure 5b et 5c). Il est même possible d'utiliser des opérateurs différents pour chacun des polyèdres composant le maillage volumique d'un niveau donné. Cela permet l'utilisation des 3-cartes multirésolution dans un large éventail d'applications.

1.3. Sommaire

Nous commencerons par présenter et définir la subdivision de volumes. Nous enchaînerons par la présentation de notre modèle multirésolution et de son application efficace à la représentation multirésolution et adaptative de volumes de subdivision. Ensuite, nous présenterons les avantages à utiliser cette nouvelle structure en terme de complexité en temps et en espace. Enfin, nous conclurons et donnerons quelques perspectives à ce travail.

2. Volumes de subdivision

La subdivision est un processus récursif définissant un maillage de plus en plus fin. Il peut être séparé en deux étapes. D'abord, la topologie du maillage est raffinée (par exemple en insérant des sommets), c'est-à-dire que des cellules (sommets, arêtes, faces, volumes) sont insérées ainsi que les relations de voisinage définissant le type de subdivision. Puis, la géométrie est mise à jour, c'est-à-dire que la position des cellules insérées est définie en appliquant le masque de subdivision aux sommets ajoutés pour les schémas interpolants ou à tous les sommets pour les schémas approximatifs. Les poids utilisés dans les masques de subdivision visent l'obtention de surfaces ou de volumes dont

la continuité est contrôlée. Ces considérations géométriques sortent du cadre de ce travail et ne seront pas discutées plus avant.

Nous nous intéressons ici uniquement à la première étape, c'est-à-dire le raffinement de la topologie. Ce processus de subdivision est utilisé pour générer une hiérarchie de maillages, on parle de maillage multirésolution. Notre contribution consiste principalement à proposer une représentation efficace de cette hiérarchie.

Dans cette section nous concentrons notre présentation sur les algorithmes de subdivision topologique, et plus particulièrement la famille des schémas primaux. Dans le cas des surfaces, les modèles primaux de référence sont ceux de Loop [Loo87] et Butterfly [DLG90, ZSS96] pour les faces triangulaires, de Catmull-Clark [CC78] pour les faces quadrangulaires.

Dans le cas volumique, l'extension naturelle des schémas primaux consiste à d'abord subdiviser toutes les faces du polyèdre, puis, à créer et connecter de nouvelles faces à l'intérieur de celui-ci afin de créer de nouveaux polyèdres. Les principaux schémas existants pour les volumes de subdivision sont : le schéma de Loop adapté aux tétraèdres [CMQ02] puis également aux octaèdres [SHW04], le schéma de Catmull-Clark adapté aux hexaèdres et généralisé aux autres polyèdres par McCracken et Joy [MJ96] ainsi qu'une adaptation aux maillages tétraédriques du schéma $\sqrt{3}$ pour les maillages triangulaires [BHU10a].

Dans le cas surfacique, des schémas duaux travaillant par éclatement de sommets, ont été proposés, comme le schéma de Doo-Sabin [DS78]. Lorsqu'on considère uniquement la topologie du maillage, réaliser une subdivision de Doo-Sabin revient à appliquer une subdivision de Catmull-Clark sur le maillage dual [KCB09]. Pour les maillages volumiques, le modèle que nous proposons supporterait une telle approche. Cependant, il n'existe pas, à notre connaissance, d'applications exploitant des schémas de subdivision duaux pour des volumes. Nous avons donc choisi de ne pas développer plus avant cette possibilité pour le moment.

La figure 6 illustre un maillage hexaédrique et un maillage tétraédrique, à deux niveaux consécutifs D^0 (cf figure 6a) et D^1 (cf figure 6b) de subdivision primale. Les cellules (arêtes et sommets) en rouge sont celles qui sont introduites au niveau D^1 (cf figure 6c).

Les opérations de raffinement utilisées par les schémas de subdivision primaux présentés ci-dessus ne reposent pas sur une structure topologique générique. Ceci implique qu'ils sont spécifiques à chaque type de maillage, tétraédrique/octaédrique dans un cas, hexaédrique ou quelconque dans l'autre. Ces algorithmes ne sont pas présentés dans une version adaptative, leurs auteurs n'ayant pas à leur disposition de modèle gérant les problèmes de conformité déjà mentionnés.

Enfin, si ces schémas permettent de générer une hiérar-

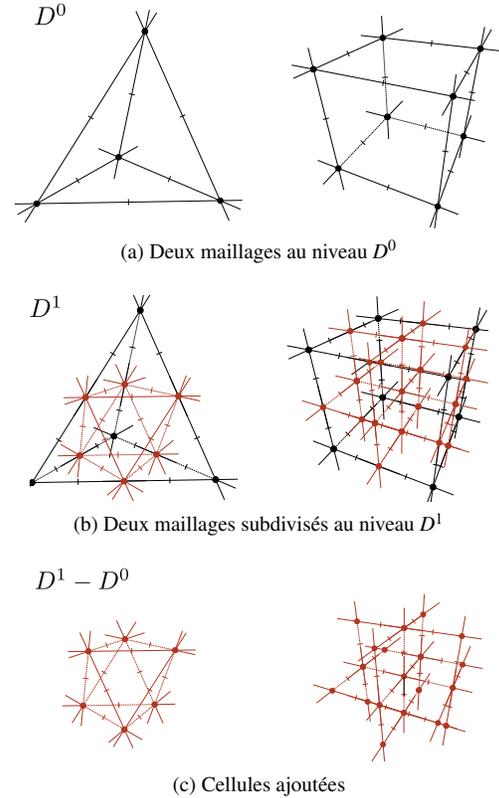


Figure 6: Subdivision d'un hexaèdre et d'un tétraèdre

chie complète de volumes, les modèles utilisés dans ces travaux ne représentent que le maillage le plus fin. En conséquence, les algorithmes exploitant des données multirésolution comme l'édition multirésolution ou la compression ne sont pas directement accessibles. Le modèle présenté ci-après vise à combler ces manques.

3. Modèle multirésolution

Un maillage tridimensionnel consiste en la discrétisation d'un volume en un complexe cellulaire composé de cellules de différentes dimensions (volumes, faces, arêtes, sommets) connectées entre elles par des relations d'adjacences.

L'une des premières structures de données manipulant la topologie de ce type de données est la structure de facet-edge [DL87]. Un complexe cellulaire est stocké comme un ensemble d'objets, chacun représentant une seule facet-edge. En pratique, cette structure est très peu utilisée car difficile à implémenter.

Une formalisation de cette structure a été proposée [Duf89, Lie91], conduisant au modèle des 3-cartes combinatoires défini par le quadruplet :

$$M = (D, \alpha_0, \alpha_1, \alpha_2) \quad (1)$$

où D est un ensemble fini de brins abstrayant la notion de demi-arête orientée, et, où α_0 , α_1 et α_2 sont des relations entre ces brins telles que :

- α_0 et α_1 soient des involutions sur D
- α_2 soit une permutation sur D
- $\alpha_2 \circ \alpha_0$ soit une involution sur D .

Ces relations permettent de parcourir les cellules composant le maillage : les sommets, les arêtes, les faces, les volumes (cf. figure 7). La contrainte sur α_0 et α_2 implique que les deux brins d'une arête (représentés en vert et rouge sur la figure 7c) sont cousus de manière symétrique.

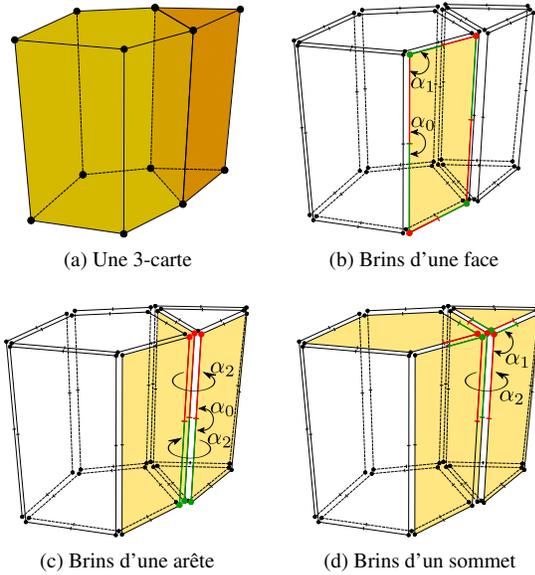


Figure 7: Schéma des principaux parcours de cellules topologiques dans une 3-carte. Ces parcours donnent accès aux cellules de dimensions supérieures incidentes à la cellule parcourue (soulignées en jaune ci-dessus).

Les 3-cartes multirésolution étendent le modèle des 2-cartes multirésolution introduit dans [KCB09]. De manière intuitive, chaque maillage volumique de la hiérarchie est représenté par une 3-carte. Ils héritent ainsi de leur généralité et de leur efficacité. Si on se contentait d'une implémentation naïve utilisant des cartes complètement déconnectées le coût mémoire d'une telle représentation deviendrait rapidement prohibitif. Pour éviter toute redondance, les brins des niveaux grossiers sont réutilisés dans les niveaux plus fins. Cela conduit à un modèle dont le coût mémoire est maîtrisé comme montré section 6. Cet aspect a déjà été illustré figure 6 où les brins rouges correspondent aux cellules insérées durant la subdivision et les brins noirs sont ceux réutilisés d'un niveau à l'autre.

Précisément, dans une carte multirésolution, les brins sont rangés dans une suite d'ensembles imbriqués les uns dans les autres comme illustré sur la figure 8. Comme les relations

topologiques (α_0 , α_1 et α_2) changent d'un niveau de résolution à l'autre, chaque brin stocke les valeurs de ses relations pour chacun des niveaux dans lequel il existe.

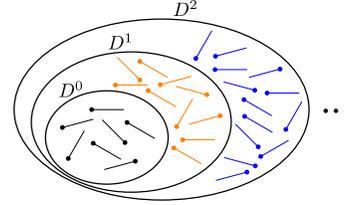


Figure 8: Ensembles de brins d'une 3-carte multirésolution

Concrètement, dans une carte contenant n niveaux de subdivision, chaque brin connaît le niveau i dans lequel il a été inséré et stocke un tableau de $n - i$ versions de ces relations. Ainsi les brins du niveau le plus fin, qui sont les plus nombreux (de l'ordre de 50% des brins) portent une seule version de ces relations, qui sont les moins nombreux, portent n versions de ces relations. C'est ce qui rend notre modèle si compétitif en ce qui concerne le coût mémoire. Cette notion est illustrée figure 9 en se limitant aux brins d'une face pour des questions de lisibilité.

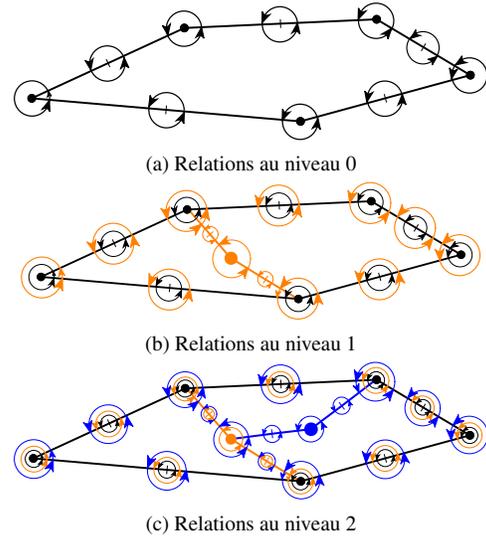


Figure 9: Illustration du stockage des différentes versions des relations topologiques dans une face de 3-carte multirésolution

La définition formelle d'une 3-carte multirésolution est donc :

$$M = (\{D^i\}_{i \geq 0}, \{\alpha_0^i\}_{i \geq 0}, \{\alpha_1^i\}_{i \geq 0}, \{\alpha_2^i\}_{i \geq 0}) \quad (2)$$

où pour tout $i \geq 0$, $M^i = (D^i, \alpha_0^i, \alpha_1^i, \alpha_2^i)$ est une 3-carte représentant le maillage au niveau i . Les D^i forment une

suite d'ensembles telle que pour tout $i \geq 0$, $D^i \subset D^{i+1}$. Ainsi, chaque D^i contient les brins insérés jusqu'au niveau i . Les $\{\alpha_k^i\}_{i \geq 0}$ représentent les versions des relations α_k ($k \in \{0, 1, 2\}$) pour les différents niveaux de résolution de la 3-carte. Elles sont construites de manière à ce que la restriction des α_k^i aux brins de niveau i forment une 3-carte valide.

4. Subdivision d'un polyèdre quelconque

Nous proposons un schéma de subdivision implémentant de manière générale et adaptative le schéma de Catmull-Clark adapté aux hexaèdres et généralisé aux autres polyèdres ainsi que le schéma de Loop adapté aux tétraèdres/octaèdres. Dans cette section, nous expliquons précisément la reconstruction topologique pour le premier schéma. L'objectif est de démontrer les qualités de généralité de notre modèle. Nous verrons par la suite comment cette généralité permet une subdivision adaptative de maillages volumiques quelconques, en résolvant convenablement tous les problèmes de conformité.

Le principe de la subdivision consiste à insérer un sommet au centre du polyèdre. Ce sommet est relié par des arêtes aux centres des faces. Ensuite, chaque sommet au centre d'une face est relié aux milieux des arêtes composant cette face. Après cette première phase de subdivision, il faut construire les faces reliant toutes ces arêtes puis les volumes bordés par ces faces. Enfin, il faut générer l'ensemble des relations d'adjacences et d'incidences entre toutes les cellules créées.

Lorsque l'on s'intéresse à des polyèdres quelconques, c'est-à-dire dont le nombre de faces et le degré de chaque face est quelconque, cette dernière phase est de loin la plus complexe combinatoirement. C'est pourquoi beaucoup de travaux se limitent à la subdivision d'hexaèdres en 8 octants, c'est-à-dire à un cas où la topologie des cellules insérées est invariante. Nous montrons ci-après comment l'information topologique présente dans notre modèle permet de retrouver efficacement l'ordre et l'orientation des cellules ajoutées.

4.1. Ordre des arêtes autour d'un sommet

Dans un premier temps nous considérons la subdivision d'une face et l'ordre des arêtes autour du sommet inséré. La figure 10a montre une face pentagonale avec les relations α_0 et α_1 en rouge et vert. La figure 10b montre cette face subdivisée avec un sommet central. Il s'agit de retrouver, autour de ce sommet, l'ordre des arêtes (représenté par la flèche bleue).

La figure 10c montre une vue éclatée des faces générées. On constate que la combinaison $\alpha_1 \circ \alpha_2$, de la relation α_2 reliant deux faces adjacentes et de la relation α_1 entre deux arêtes d'une face, permet de parcourir les arêtes de la manière souhaitée. Ainsi, pour reconstruire l'ordre des sommets, il suffit de générer correctement les relations α_2 entre

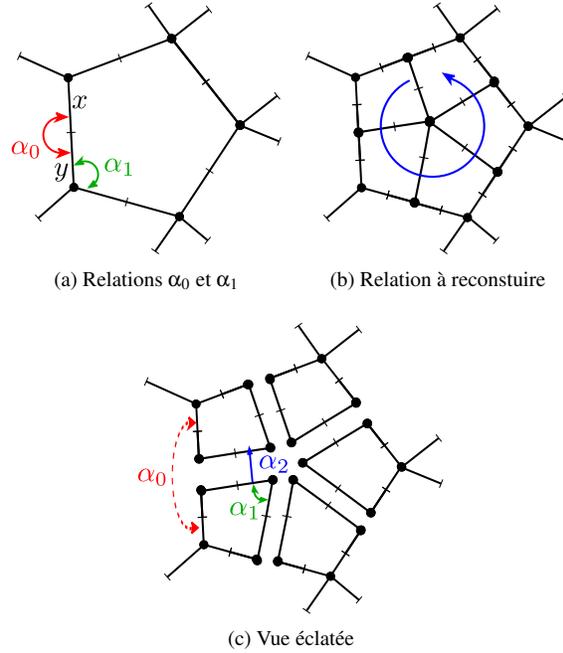


Figure 10: Relations à reconstruire dans une face subdivisée

les faces ajoutées. On remarque alors que ces relations sont directement liées aux relations par α_0 des brins des arêtes subdivisées (flèche en pointillée rouge sur la figure 10c). Toute la finesse de notre algorithme consiste à alterner entre représentation primale et duale pour propager cette information durant la phase de subdivision. Notons que, pour que cela soit possible, il faut que les deux niveaux de subdivision coexistent dans le modèle.

Nous détaillons maintenant l'algorithme de subdivision d'une face de niveau i . Dans un premier temps, toutes les arêtes sont coupées en deux en ajoutant deux brins de niveau $i+1$ représentés en orange sur la figure 11a. Durant cette opération les relations α_0^{i+1} sont construites. Puis, pour tous les niveaux i , cette étape reconstruit pour tout $x \in (D^{i+1} \setminus D^i)$ les relations de voisinage α_1^{i+1} comme suit (avec $y = \alpha_0^i(x)$) :

$$\alpha_1^{i+1}(\alpha_0^{i+1}(x)) = \alpha_0^{i+1}(y) \quad (3)$$

Les sommets de la face initiale forment ainsi des coins séparés visibles figure 11a. La seconde étape consiste à former les faces naissantes en reliant les sommets de niveau $i+1$ par deux arêtes, c'est-à-dire 4 brins de niveau $i+1$ (cf. figure 11b).

La dernière étape (cf. figure 11c) consiste à coudre par α_2 au niveau $i+1$ (flèche bleue), les paires d'arêtes attachées à des brins reliés par α_0 au niveau i (les brins noirs). Pour tous les niveaux i , cette étape reconstruit pour tout $x \in (D^{i+1} \setminus D^i)$

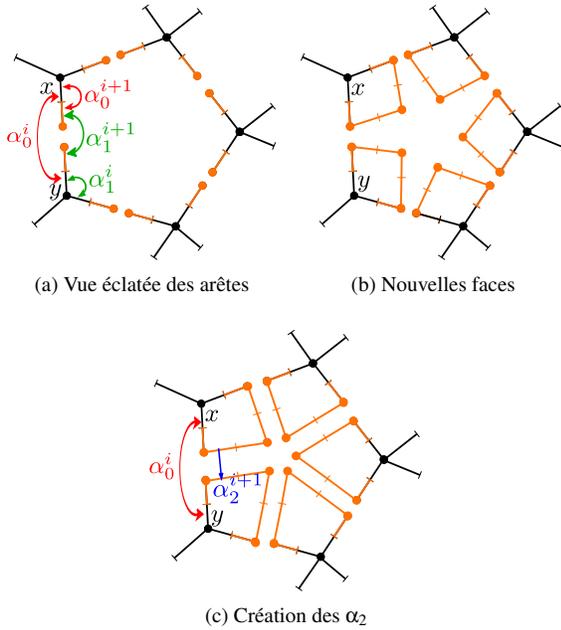


Figure 11: Étapes de subdivision d'une face

les relations de voisinage α_2^{i+1} comme suit (avec $y = \alpha_0^i(x)$) :

$$\alpha_2^{i+1}(\alpha_1^{i+1}(\alpha_0^{i+1}(x))) = \alpha_1^{i+1}(\alpha_0^{i+1}(y)) \quad (4)$$

On applique cette reconstruction de la relation α_2^{i+1} aux 4 autres brins de l'arête de manière symétrique. Ce processus est réitéré pour chaque nouvelle arête « centrale » du niveau $i+1$ en parcourant les arêtes de la face de niveau i . Au final, de manière complètement implicite, l'ordre des arêtes autour du sommet central, c'est à dire la relation $\alpha_1^{i+1} \circ \alpha_2^{i+1}$ est reconstruit au niveau suivant quelque soit le degré de la face initiale, ainsi cet algorithme est générique car il permet de subdiviser toute face quel que soit son degré. Ce principe est étendu au volume dans la section suivante.

4.2. Topologie d'un polyèdre subdivisé

Les outils présentés ci-dessus permettent de définir élégamment la subdivision des faces d'un polyèdre quelconque. En premier lieu, les faces sont subdivisées par l'ajout d'un sommet central (cf. figure 12a et 12b). Par contre, à cette étape la couture des arêtes autour des sommets centraux n'est pas effectuée. On obtient ainsi autour de chaque sommet initial un coin (ou ombrelle) composé de faces quadrangulaires (cf. figure 12c). Dans un deuxième temps, cette ombrelle est complétée automatiquement pour former le futur volume. Pour cela une face topologique (non planaire) est créée le long du bord de l'ombrelle (cf. figure 12d). Cette face topologique est ensuite subdivisée à son tour par l'ajout d'un point central qui sera positionné au centre du volume subdivisé (cf figure 12e).

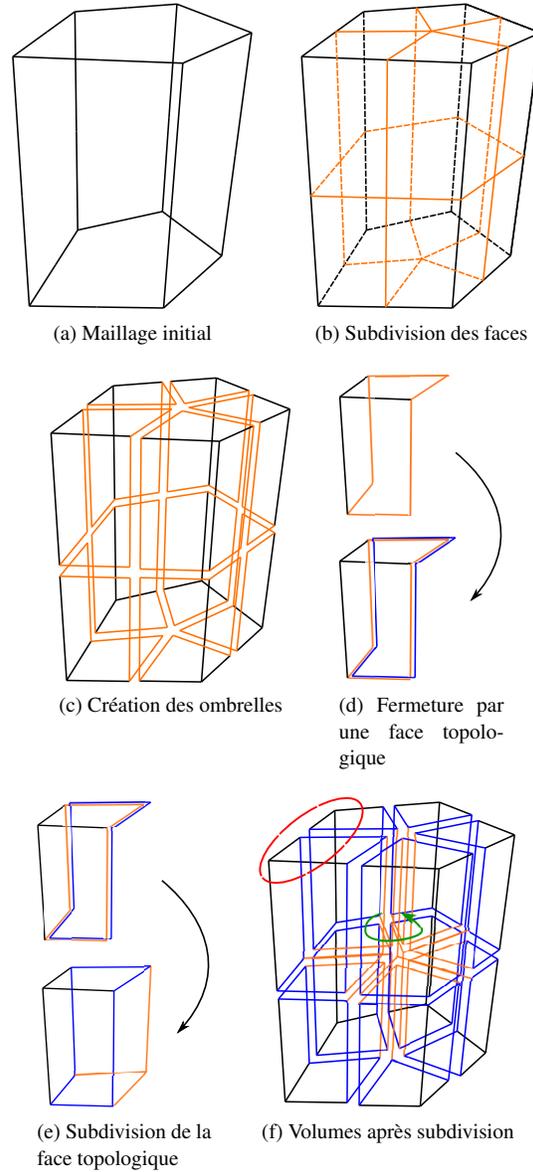


Figure 12: Subdivision topologique

À ce stade, nous avons généré un volume par sommet du polyèdre de départ (cf. figure 12f). Les relations laissées libres peuvent maintenant être complétées pour ordonner les faces internes des volumes créés autour de leur arête commune (cf. flèche verte figure 12f). Pour ce faire, intéressons nous aux relations entre les deux volumes dans l'ovale rouge de la figure 12f dont une vue dans le formalisme des cartes combinatoires est reproduite figure 13a. Précisément, pour chaque couple de sommets du polyèdre initial, on considère les deux brins x et y (en noir sur la figure 13a) cousus par α_0 au niveau i . Leurs images par α_0^{i+1} sont les sommets

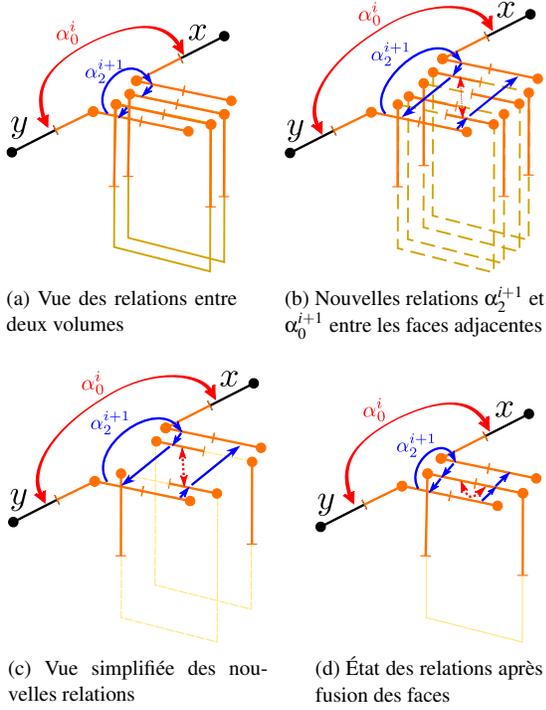


Figure 13: Étape de couture des volumes entre eux

des deux faces incidentes (visible en trait plein figure 13a et 13d). Ces deux faces se superposent et doivent être fusionnées pour ne former qu'une seule face liant les deux volumes entre eux. Lier ces deux volumes (cf. figure 13b et 13c) consiste à coudre un brin sur deux de chacune des deux faces avec la relation α_2^{i+1} (flèches bleues) pour chaque paire de faces internes, ainsi qu'un brin sur deux avec la relation α_0^{i+1} (flèche rouge pointillée) pour chaque paire d'arêtes internes. Pour reconstruire l'ordre des faces, il faut générer correctement les relations α_2^{i+1} et α_0^{i+1} entre les faces internes. Ainsi, pour tous les niveaux i , cette étape reconstruit pour tout $x \in (D^{i+1} \setminus D^i)$ les relations de voisinage comme suit :

$$\alpha_2^{i+1}(\alpha_1^{i+1}(\alpha_0^{i+1}(x))) = \alpha_1^{i+1}(\alpha_0^{i+1}(y)) \quad (5)$$

et

$$\alpha_2^{i+1}(\alpha_1^{i+1}(\alpha_0^{i+1}(x))) = \alpha_2^{i+1}(\alpha_0^{i+1}(\alpha_1^{i+1}(\alpha_0^{i+1}(y)))) \quad (6)$$

Cette formulation doit être appliquée à tous les brins de la face pour reconstruire le lien entre les différents volumes. Après cette étape, l'ensemble des relations topologiques entre les cellules (sommets, arêtes, faces, volumes) créées sont reconstruites (cf. figure 13d), ainsi cet algorithme est générique car il permet de subdiviser tout volume quel que soit son degré.

5. Subdivision adaptative générique

Le principe d'une subdivision adaptative est de choisir, en fonction de critères donnés, les polyèdres subdivisés et ceux qui ne le sont pas. Pour que cela puisse se faire, il faut traiter correctement les cellules, arêtes ou faces, incidentes à des polyèdres ayant des niveaux de subdivision différents. Dans le cas contraire, des problèmes de conformité déjà mentionnés peuvent apparaître. Comme décrit précédemment, les cartes combinatoires possèdent un atout fort pour maintenir cette cohérence. Contrairement aux modèles basés sur des graphes d'incidence ou aux modèles explicitant les cellules (comme les modèles tétraédriques), les cellules d'une carte sont définies implicitement. Ces définitions implicites sont illustrées section 3 sur la figure montrant les parcours de cellules topologiques.

Pour être plus précis, dans une 3-carte multirésolution, une face située entre deux polyèdres est commune aux deux. La subdivision d'un niveau i à un niveau $i+1$ d'un des polyèdres entraîne la subdivision de la face des deux côtés. Les brins de cette face vus par le polyèdre non subdivisé sont parcourus au niveau de résolution i . Cette face est donc perçue comme encore entière. À l'inverse, les brins de cette face vus par le polyèdre subdivisé sont parcourus au niveau $i+1$. Cette même face est donc perçue comme subdivisée. Précisément, le parcours des brins montrera n faces de niveau $i+1$, où n est le degré de la face initiale. Par exemple dans la figure 14, la face attenante aux deux polyèdres est vue comme 4 faces de degrés 4 du côté du polyèdre au premier plan (en rouge) mais également comme 1 face de degré 4 pour chacun des 4 hexaèdres au second plan (en bleu).

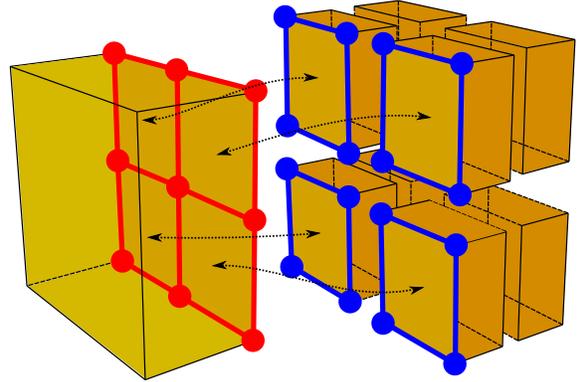


Figure 14: Une face à deux niveaux de subdivision

Comme exposé ci-dessus, une face peut-être vue de différentes manières en fonction du niveau duquel on la parcourt. Il est donc nécessaire de connaître et de savoir calculer le niveau de cette face. Soit $lev(b)$, le niveau d'un brin b . Pour déterminer le niveau d'une face f au sein de la 3-carte de niveau i , on part de l'un de ses brins b ayant le plus petit niveau d'insertion dans la carte $lev(b) \leq i$. En effet, le niveau de cette face est nécessairement compris entre le minimum

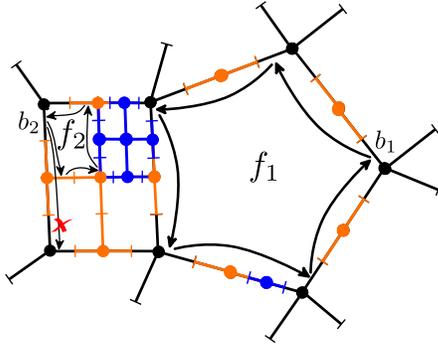


Figure 15: Exemple de calcul du niveau d'une face : illustration pour les faces f_1 et f_2 en partant respectivement des brins b_1 et b_2 .

des niveaux d'insertion de ses brins et le niveau i de la 3-carte considérée. Afin de tester les niveaux du plus faible au plus élevé, on démarre donc sur l'un des brins présentant le niveau d'insertion le plus faible. D'abord, on teste si les brins de la face du brin b au niveau $lev(b)$ sont contenus dans la face de b au niveau i . Si oui, f est une face de niveau $lev(b)$ dans la carte de niveau i . Dans le cas contraire, on teste si les brins de la face de b au niveau $lev(b) + 1$ sont contenus dans la face de b au niveau i . On itère sur ce test de niveau jusqu'à trouver le plus petit niveau pour lequel tous les brins de la face de b sont contenus dans la face f au niveau i . Au maximum, ce niveau est le niveau i lui-même.

Par exemple, pour la face f_1 (cf. figure 15), on part du brin b_1 inséré dans la carte au niveau 0, puis on commence à parcourir la face de b_1 au niveau 0 (brins en noir). Tous les brins de cette face sont contenus dans la face de b_1 au niveau 2 (brins en noir, orange et bleu). Autrement dit, on parcourt toute la face en ne rencontrant que des brins de niveau 0. On en conclut que f_1 est une face de niveau 0. Pour la face f_2 , on commence à partir du brin b_2 inséré dans la carte au niveau 0. Le brin suivant dans la face de b_2 au niveau 0 n'est pas inclus dans la face de b_2 au niveau 2. Nous continuons donc le parcours de la face de b_2 au niveau 1 et nous remarquons que tous les brins de la face sont contenus au niveau 2. Nous pouvons donc en conclure que f_2 est une face de niveau 1. Autrement dit, avec les brins de niveau 0 il y a un changement de face, mais pas avec les brins de niveau 1.

Ainsi, pour décrire sans ambiguïté ce qu'est une subdivision adaptative, il faut définir formellement la notion de niveau de subdivision d'une cellule dans une carte de niveau i . On appelle $cell_k^i(b)$ l'ensemble des brins de la k -cellule du brin b au niveau i . On définit le niveau de subdivision de la k -cellule du brin b comme étant égal au plus petit niveau j tel que $cell_k^j(b') \subset cell_k^i(b)$, avec $b' \in cell_k^i(b)$. Concrètement, l'inclusion dans la formule précédente signifie que le niveau d'une cellule est défini comme le plus petit niveau tel que le bord de la cellule peut être subdivisé mais pas la cellule.

Grâce à ces renseignements sur le niveau d'une cellule, la subdivision adaptative d'un polyèdre s'écrit assez simplement. Pour subdiviser un polyèdre du niveau i au niveau $i + 1$, l'algorithme commence par la subdivision de chaque face du polyèdre. La subdivision d'une face commence par la recherche du niveau de celle-ci. En effet, si la face est déjà de niveau supérieur à $i + 1$ parce que le polyèdre adjacent est déjà subdivisé alors l'algorithme passe à la face suivante. Dans le cas contraire (face de niveau i), cette face interroge à son tour les arêtes qui la composent pour connaître leur niveau. En effet, si une arête de cette face est déjà de niveau $i + 1$ parce qu'elle est adjacente à un polyèdre déjà subdivisé, l'algorithme passe à l'arête suivante. Les arêtes de niveau i , quant à elles sont subdivisées au niveau $i + 1$. La face peut alors être subdivisée à son tour au niveau $i + 1$. Lorsque cette première étape est effectuée, nous nous servons ensuite des outils présentés dans les sections précédentes pour construire et lier topologiquement au niveau $i + 1$ les faces intérieures des futurs volumes. Une fois le processus terminé nous obtenons des polyèdres liés entre eux par des relations topologiques au niveau i et $i + 1$. En appliquant l'algorithme de cette manière, l'adaptativité au sein de la hiérarchie est gérée de manière transparente.

6. Évaluation de la complexité

Comme il n'existe aucune structure de données utilisée pour représenter des volumes de subdivision multirésolution adaptatif en pratique, nous nous comparons ici à une structure de partitionnement de l'espace équivalente : une forêt d'octree. D'abord, nous comparerons la complexité en temps et en espace entre les deux structures. Puis, pour finir, nous ferons quelques observations concernant cette comparaison.

6.1. Complexité en temps

Les requêtes d'adjacences sont les opérations les plus exécutées avec des volumes de subdivision multirésolution. Elles sont utiles lors de l'application d'un schéma de subdivision aussi bien que pour l'exécution d'algorithmes plus complexes.

Dans une forêt d'octree, chaque hexaèdre ou tétraèdre du maillage initial représente la racine d'un arbre octal. Cette structure peut être vue comme un graphe acyclique non orienté. Les requêtes de voisinage dans ce type de structure sont résolues en temps constant amortis. L'algorithme classique consiste à remonter l'arbre jusqu'à la racine parente puis chercher la cellule adjacente. Les requêtes d'adjacence entre arbres sont résolues au niveau racine en $O(H(n))$ où $H(n)$ est le niveau maximum (c-à-d la hauteur de l'arbre le plus profond).

Pour une 3-carte multirésolution, chaque maillage de la hiérarchie représente une 3-carte directement accessible. Les requêtes d'adjacences sont exécutées en temps constant. La complexité des algorithmes plus évolués comme traverser

les arêtes voisines sont linéaires en la taille du voisinage considéré. Notre modèle est donc optimal pour tous les opérateurs courants.

6.2. Complexité en espace

Dans cette section nous comparons une estimation du coût d'une 3-carte multirésolution avec une forêt d'arbres octaux d'hexaèdres dans le cadre d'une subdivision régulière. Ceci étant le pire cas au niveau de l'espace mémoire nécessaire. Toutefois, afin de préciser notre propos, nous commencerons par dissocier le coût des structures elles-mêmes de leur extension multirésolution.

6.2.1. Coût standard

Soit n le nombre de brins dans une 3-carte. Pour compter le coût total de l'information topologique, il nous faut compter le nombre de pointeurs stockés pour chaque brin. Le calcul pour les relations α_0 , α_1 et α_2 est assez simple puisque chaque brin dispose d'une unique liaison. Il y a donc en tout n pointeurs stockés pour chacune des trois relations. Il nous faut compter également un pointeur pour chaque brin vers une structure stockant les plongements de sommet. Ce qui nous donne un coût de : $4 \times n$ pointeurs.

Dans une forêt d'octree, les racines représentent le maillage le plus grossier. Leur coût correspond au nombre d'hexaèdres présents dans le maillage. Cependant, il nous faut remarquer qu'à l'inverse des cartes combinatoires, ceci n'est pas une représentation topologique complète car il n'y a pas de relation liant entre elles toutes les cellules du maillage. En effet, les arêtes ne sont pas représentées dans ce type de structure. Ainsi, comme expliqué précédemment, des trous topologiques apparaissent aux frontières entre les zones de niveaux différents.

Afin de compléter les informations topologiques disponibles dans ce type de structures, nous nous devons d'ajouter un minimum d'informations décrivant la topologie d'un polyèdre pour que la structure de données définisse un objet volumique. Chaque hexaèdre contiendra ainsi une liste de ses faces, avec pour chaque face une liste des arêtes (4 pointeurs par face donc 24 pointeurs pour les 6 faces) et enfin pour chacune des 12 arêtes, il y a 2 pointeurs vers les sommets (24 pointeurs en plus). Il faut ajouter à cela les 8 pointeurs vers le tableau de plongement des sommets. Pour h le nombre d'hexaèdres du maillage, nous obtenons un coût de : $56 \times h$ pointeurs.

Un hexaèdre étant composé de 24 brins dans une 3-carte, nous savons que $h = \frac{n}{24}$. Le ratio entre les deux structures de données à cette étape est donc de :

$$\frac{4 \times h}{\frac{56}{24} \times h} \simeq \frac{12}{7} \simeq 1,7 \quad (7)$$

La structure des cartes combinatoires est donc à cette

étape 70% plus coûteuse en terme d'espace. Ce qui est normal, car contrairement aux cartes combinatoires, il manque à la forêt d'octree encore des informations topologiques. Nous y avons seulement ajouté le minimum pour décrire un volume. En effet, toutes les requêtes de voisinage ne sont pas disponibles immédiatement mais doivent être calculées. Par exemple, pour obtenir toutes les faces autour d'une arête, il faut commencer par chercher le volume adjacent à une des faces de cette arête. Puis, dans la liste des faces de ce volume, il faut chercher celle possédant l'identifiant d'arête correspondant. Enfin, il faut recommencer l'opération. Dans une 3-carte combinatoire, les brins des faces sont reliés entre eux par la relation volumique α_2 qu'il suffit de parcourir.

6.2.2. Cas multirésolution

Soit n le nombre de brins dans une 3-carte multirésolution. Soit n_0 le nombre de brins au niveau 0 du maillage, et k le nombre maximal de niveau de la structure multirésolution.

Chaque hexaèdre étant coupé en 8 nouveaux hexaèdres, le nombre de brins est multiplié par un facteur 8 à chaque étape de subdivision. Nous avons donc :

$$n = n_0 \times 8^k \quad (8)$$

Pour compter le coût total de l'information topologique, il nous faut compter le nombre de pointeurs stockés pour chaque brin.

Le calcul des relations α_1 et α_2 est assez simple puisque chaque brin dispose d'une unique liaison α_1 ou α_2 . Il y a donc en tout n (ou $n_0 \times 8^k$) pointeurs stockés pour chacune des deux relations.

Pour la relation α_0 , il faut additionner les tailles des tableaux de pointeurs contenu dans chaque brin. La taille de ce tableau est fonction du niveau d'insertion du brin. On note que $\frac{7}{8}$ des brins sont insérés au niveau de résolution maximum et n'ont donc qu'une seule liaison α_0 , $\frac{7}{8}$ des autres brins (c'est-à-dire $\frac{7}{64}$) ont deux liens ... ; pour i entre 1 et k il y a $n \times \frac{7^i}{8^i}$ brins dont le tableau a i éléments. Les brins décrivant le niveau 0 ont $k+1$ éléments dans leur tableau. Le nombre total d'éléments dans les tableaux de la relation α_0 de tous les brins est donc :

$$n_0 \times (k+1) + n_0 \times 7 \sum_{i=1}^k i \times 8^{k-i} \quad (9)$$

En comptant les pointeurs α_1 , α_2 et un pointeur pour les plongements de sommet, le nombre total de pointeurs est donc :

$$3 \times n_0 \times 8^k + \left(n_0 \times (k+1) + n_0 \times 7 \sum_{i=1}^k i \times 8^{k-i} \right) \quad (10)$$

La somme de l'expression précédente peut être identifiée à la série entière : $\sum_{n \geq 0} i \times x^i = \frac{x}{(1-x)^2}$, définie pour $|x| < 1$. En négligeant les termes de la série tels que $i > k$, on obtient :

$$\sum_{i=1}^k i \times 8^{k-i} \simeq 8^k \times \frac{\frac{1}{8}}{(1-\frac{1}{8})^2} \quad (11)$$

On substitue :

$$3 \times n_0 \times 8^k + \left(n_0 \times (k+1) + n_0 \times 8^k \times \frac{8}{7} \right) \quad (12)$$

Cette équation se simplifie en :

$$\frac{29}{7} \times n_0 \times 8^k \quad (13)$$

Les racines de la forêt d'octree stockent 8 pointeurs pour les informations hiérarchiques vers les fils. Dans la forêt d'octree, un nœud nécessite 9 pointeurs pour les informations hiérarchique (c'est-à-dire 8 pour les fils et 1 pour le parent). Comme nous l'avons exposé dans la section précédente, chaque racine et chaque nœud de la forêt d'octree stocke également 56 pointeurs pour les informations topologiques et géométriques. Pour résumer, nous obtenons 64 pointeurs pour chaque racine et 65 pointeurs par nœud.

Soit h_0 le nombre de volumes au niveau 0 (c'est à dire la racine de l'arbre) et k le niveau maximal de résolution. Comme le nombre de volumes est multiplié par un facteur 8 à chaque niveau de subdivision, le nombre total de pointeurs stockés est :

$$h_0 \times 64 + 65 \times h_0 \times \sum_{l=1}^k 8^l \quad (14)$$

La somme est identifiable à la série entière : $\sum_{i=0}^n x^i = \frac{x^{n+1}-1}{x-1}$. Nous pouvons l'exprimer de la manière suivante :

$$\sum_{l=1}^m 8^l \simeq \frac{8^{m+1}-1}{7} - 1 \quad (15)$$

Comme nous savons que $h_0 = \frac{n_0}{24}$, notre équation se simplifie donc en :

$$\frac{65}{21} \times n_0 \times 8^k \quad (16)$$

Le ratio entre les deux structures de données est de :

$$\frac{\frac{29}{7} \times n_0 \times 8^k}{\frac{65}{21} \times n_0 \times 8^k} \simeq \frac{87}{65} \simeq 1,3 \quad (17)$$

La structure des cartes combinatoire multirésolution est donc au final seulement 30% plus coûteuse en terme d'espace que la forêt d'octrees dans le cas multirésolution. Ce surcoût est largement compensé par la générique des polyèdres représentables grâce à une carte combinatoire ainsi que la complétude des informations topologiques.

7. Conclusion

Les 3-cartes combinatoires multirésolution définies dans cet article offrent un nouveau cadre pour la représentation de maillages volumiques multirésolution. Elles apportent un certain nombre d'avantages par rapport aux structures qui doivent se limiter à la représentation de polyèdres spécifiques (tétraèdres ou hexaèdres).

Elles permettent, au sein d'un même modèle, la représentation de différents polyèdres générés par un schéma générique de subdivision s'appuyant sur des opérations topologiques. Cette capacité à représenter des maillages polyédriques est un avantage supplémentaire dans le cas d'une subdivision adaptative, où la coexistence de différents niveaux de résolution produit des volumes non hexaédriques ou non tétraédriques, qui ne sont pas supportés par des structures de données classiques et y engendrent alors des trous topologiques.

Les requêtes d'adjacence sont effectuées plus efficacement étant donné qu'elles sont exécutées en temps constant quel que soit le niveau de résolution. Enfin, on peut remarquer que la différence de besoin en espace mémoire est intrinsèque au modèle et non à l'extension multirésolution puisque celle-ci s'amenuise. Cette différence est également à mettre en rapport avec la souplesse apportée par la générique de notre modèle.

Le cadre défini ici amène de nombreuses perspectives. Tout d'abord, la généralité des 3-cartes multirésolution nous permet d'envisager l'élaboration d'un algorithme générique mixant la subdivision de faces triangulaires (p.ex. les tétraèdres) et quadrangulaires (p.ex. les hexaèdres) pour les polyèdres ayant des faces triangulaires et quadrangulaires. Ainsi, un prisme ne serait plus subdivisé en 6 hexaèdres comme actuellement, mais en 8 prismes (cf. figure 16). Ceci peut-être intéressant pour garder une certaine régularité des éléments constituant le maillage. Celle-ci est nécessaire à des applications comme la simulation physique ou l'édition multirésolution basée sur la transformation en ondelettes.

En utilisant le caractère adaptatif de notre modèle et la subdivision topologique, nous pouvons également viser des applications de lissage topologique et géométrique du bord des volumes grossiers pour la visualisation de maillages utilisés dans différents domaines de la simulation physique.

Pour finir, nous allons également appliquer notre modèle des 3-cartes multirésolution à la représentation de volumes multirésolution non générés par des algorithmes de subdivi-

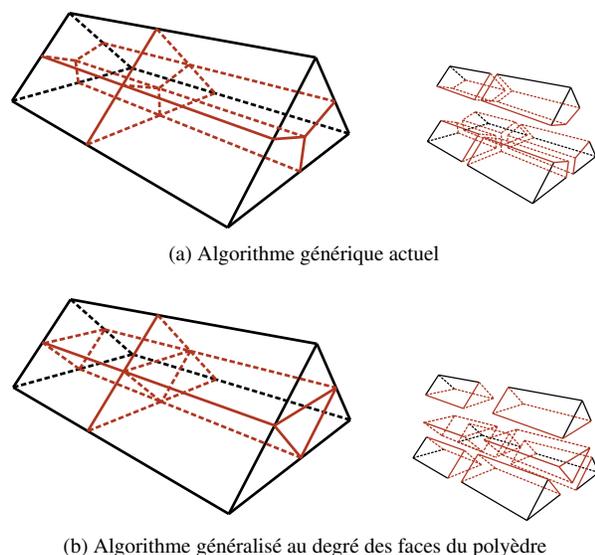


Figure 16: Subdivision d'un prisme

sion mais par des algorithmes d'éclatement et de fusion de sommets.

Références

- [BHU10a] BURKHART D., HAMANN B., UMLAUF G. : Adaptive and feature-preserving subdivision for high-quality tetrahedral meshes. *Computer Graphics Forum*. Vol. 29, Num. 1 (2010), 117–127.
- [BHU10b] BURKHART D., HAMANN B., UMLAUF G. : Iso-geometric finite element analysis based on Catmull-Clark subdivision solids. *Computer Graphics Forum*. Vol. 29, Num. 5 (2010), 1575–1584.
- [BSWX02] BAJAJ C., SCHAEFER S., WARREN J., XU G. : A subdivision scheme for hexahedral meshes. *The Visual Computer*. Vol. 18, Num. 5-6 (2002), 343–356.
- [CC78] CATMULL E., CLARK J. : Recursively generated b-spline surfaces on arbitrary topological meshes. *Computer-Aided Design*. Vol. 10, Num. 6 (1978), 350–355.
- [CGC*02] CAPELL S., GREEN S., CURLESS B., DUCHAMP T., POPOVIĆ Z. : A multiresolution framework for dynamic deformations. In *Proceedings of the 2002 ACM SIGGRAPH/Eurographics symposium on Computer animation* (New York, NY, USA, 2002), SCA '02, ACM, pp. 41–47.
- [CMQ02] CHANG Y.-S., McDONNELL K. T., QIN H. : A new solid subdivision scheme based on box splines. In *Proceedings of the seventh ACM symposium on Solid modeling and applications* (New York, NY, USA, 2002), SMA '02, ACM, pp. 226–233.
- [DdFM*05] DANOVARO E., DE FLORIANI L., MAGILLO P., PUPPO E., SOBRERO D., SOKOLOVSKY N. : The half-edge tree : A compact data structure for level-of-detail tetrahedral meshes. In *SMI '05: Proceedings of the International Conference on Shape Modeling and Applications 2005* (Washington, DC, USA, 2005), IEEE Computer Society, pp. 334–339.
- [DFKP05] DE FLORIANI L., KOBELT L., PUPPO E. : A survey on data structures for level-of-detail models. In *Advances in Multiresolution for Geometric Modelling*, Dodgson N. A., Floater M. S., Sabin M. A., (Eds.), Mathematics and Visualization. Springer Berlin Heidelberg, 2005, pp. 49–74.
- [DL87] DOBKIN D. P., LASZLO M. J. : Primitives for the manipulation of three-dimensional subdivisions. In *Proceedings of the third annual symposium on Computational geometry* (New York, NY, USA, 1987), SCG '87, ACM, pp. 86–99.
- [DLG90] DYN N., LEVINE D., GREGORY J. A. : A butterfly subdivision scheme for surface interpolation with tension control. *ACM Trans. Graph.* Vol. 9 (April 1990), 160–169.
- [DS78] DOO D., SABIN M. : Behaviour of recursive division surfaces near extraordinary points. *Computer-Aided Design*. Vol. 10, Num. 6 (1978), 356–360.
- [Duf89] DUFOURD J.-F. : Algebraic map-based topological kernel for polyhedron modellers : Algebraic specification and logic prototyping. In *Int. Conf. Eurographics'89* (1989), North-Holland, pp. 649–662.
- [JCD09] JUND T., CAZIER D., DUFOURD J.-F. : Particle-based forecast mechanism for continuous collision detection in deformable environments. In *SPM '09 : 2009 SIAM/ACM Joint Conference on Geometric and Physical Modeling* (New York, NY, USA, 2009), ACM, pp. 147–158.
- [KCB09] KRAEMER P., CAZIER D., BECHMANN D. : Extension of half-edges for the representation of multiresolution subdivision surfaces. *The Visual Computer*. Vol. 25, Num. 2 (2009).
- [Lie91] LIENHARDT P. : Topological models for boundary representation : a comparison with n-dimensional generalized maps. *Comput. Aided Des.* Vol. 23, Num. 1 (1991), 59–82.
- [Loo87] LOOP C. : *Smooth Subdivision Surfaces Based on Triangles*. Department of mathematics, University of Utah, Utah, USA, 1987.
- [LWC*02] LUEBKE D., WATSON B., COHEN J. D., REDDY M., VARSHNEY A. : *Level of Detail for 3D Graphics*. Elsevier Science Inc., New York, NY, USA, 2002.
- [MCQ04] McDONNELL K. T., CHANG Y., QIN H. : Interpolatory, solid subdivision of unstructured hexahedral meshes. *The Visual Computer*. Vol. 20, Num. 6 (2004).

- [MJ96] MACCRACKEN R., JOY K. I. : Free-form deformations with lattices of arbitrary topology. In Proceedings of the 23rd annual conference on Computer graphics and interactive techniques - SIGGRAPH '96 (New Orleans, 1996), pp. 181–188.
- [SHW04] SCHAEFER S., HAKENBERG J. P., WARREN J. : Smooth subdivision of tetrahedral meshes. In Proceedings of the 2004 Eurographics/ACM SIGGRAPH symposium on Geometry processing - SGP '04 (Nice, France, 2004), pp. 147—154.
- [VZ01] VELHO L., ZORIN D. : 4-8 subdivision. Computer Aided Geometric Design. Vol. 18, Num. 5 (2001), 397 – 427.
- [WDF11] WEISS K., DE FLORIANI L. : Simplex and diamond hierarchies : Models and applications. Accepted to Computer Graphics Forum (2011).
- [WRK*10] WICKE M., RITCHIE D., KLINGNER B. M., BURKE S., SHEWCHUK J. R., O'BRIEN J. F. : Dynamic local remeshing for elastoplastic simulation. In Proceedings of ACM SIGGRAPH 2010 (juillet 2010), pp. 49 :1–11.
- [ZSS96] ZORIN D., SCHRÖDER P., SWELDENS W. : Interpolating subdivision for meshes with arbitrary topology. In Proceedings of the 23rd annual conference on Computer graphics and interactive techniques (New York, NY, USA, 1996), SIGGRAPH '96, ACM, pp. 189–192.