

Une nouvelle structure de données pour la gestion efficace de complexes polyédriques

Alexandre MARIN^{1,2},
Alexandra BAC¹,
Laurent ASTART²

¹Université Aix-Marseille

²IFP Énergies nouvelles

Résumé

Nous présentons ici les travaux préliminaires d'une thèse relative à la génération de maillages polyédriques. Le premier objectif est donc d'avoir à disposition une structure robuste capable de représenter, construire au fur et à mesure, parcourir et modifier des complexes polyédriques contenant un grand nombre de mailles polyédriques quelconques, qui servira d'outil pour charger et éditer les futurs maillages destinés à de nouvelles méthodes numériques.

Mots-clés : Modélisation géométrique, maillages polyédriques, quasi-variétés

polyédriques, qui soit relativement *compacte* et dont on peut manipuler aisément la topologie.

1. Introduction

Il s'agit de présenter ici un travail en cours, qui se situe au début d'une thèse dont l'un des objectifs est de générer des maillages *polyédriques*, qui serviront de supports à de nouvelles méthodes numériques pour résoudre des E.D.P.s modélisant des phénomènes géologiques. Ces méthodes sont soit des Volumes Finis non-linéaires, soit des Éléments Virtuels, qui généralisent les Éléments Finis en permettant notamment l'utilisation de maillages non plus simpliciaux mais polyédriques. Élargir le type de mailles aux polyèdres améliorera les performances lors des calculs car on pourra plus facilement borner le nombre de mailles, alors que des maillages tétraédriques fins pour des modèles géologiques comportent souvent au moins un million de mailles.

Aujourd'hui il n'existe pas de bibliothèque proposant une structure représentant des maillages contenant des mailles polyédriques *quelconques*, offrant à la fois des primitives topologiques rapides, induisant un coût mémoire raisonnable et correspondant à nos besoins.

Les présents travaux ont pour but de définir et mettre en œuvre une structure permettant de charger des maillages po-

2. État de l'art

L'état de l'art concernant les maillages polyédriques est très limité. Pour garder toute l'information topologique d'un maillage, il est nécessaire d'encoder les relations d'incidence entre cellules de toutes les dimensions. Cela revient à être capable de retrouver le graphe des incidences. Il existe deux grandes familles de modèles topologiques : les structures inspirées des cartes généralisées, qui contiennent des chemins ordonnés du graphe des incidences, et ceux utilisant les demi-arêtes, qui contiennent des arcs du graphe.

2.1. Structures de type G-cartes et dérivés

Dans ce type de structure, on retient quatre involutions $(\alpha_i)_{0 \leq i \leq 3}$ définies sur un ensemble B de brins. Une i -cellule[†] est alors vue comme une orbite d'un brin $b \in B$ (en considérant que le groupe engendré par $\{\alpha_j\}_{0 \leq i \leq 3, j \neq i}$ agit sur B). La figure 1 illustre la structure de cartes généralisées.

Les cartes combinatoires et les n -cartes fonctionnent sur

[†]. Une i -cellule est un élément topologique de dimension i .

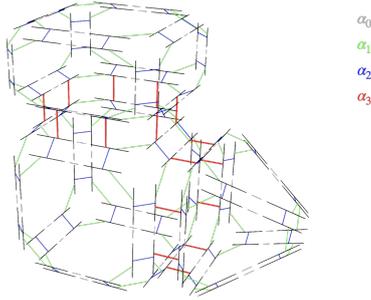


FIGURE 1: Figure 2.20 (a) extraite de [DAM10] : le domaine est décomposé pour faire apparaître les associations α_i .

le même principe, seuls la définition des brins et les axiomes sur les α_i changent. Ces structures ont l'avantage de fournir un cadre très mathématique et des fonctions topologiques rapides, en contrepartie elles prennent beaucoup de place en mémoire. Pour fixer les idées, considérons un maillage régulier hexaédrique formant un domaine hexaédrique. Pour chaque hexaèdre intérieur il faut encoder 24 associations de brins par involution α_i . En choisissant une structure de liste de paires d'entiers (économe en mémoire) pour retenir toutes les associations, on doit conserver 24 couples d'entiers par hexaèdre intérieur pour une seule involution α_i . Avec des entiers codés sur 8 bytes, le coût d'un hexaèdre intérieur s'élève à 1 536 bytes. Quand la taille d'un tel maillage tend vers l'infini, cette taille devient équivalente à celle de l'ensemble des hexaèdres intérieurs seuls. Donc pour un tel maillage contenant $20 \cdot 10^6$ hexaèdres et en excluant l'information géométrique, 28, 6 Gb est une borne inférieure approximative de l'empreinte mémoire d'une mise en œuvre de la structure de cartes généralisées.

La structure décrite dans [FWWT13] utilise des cartes combinatoires pour mémoriser les incidences. Afin de compresser un peu la structure, un ensemble minimal de tables décrivant des connexions est encodé. Cependant il n'est pas toujours possible de changer dynamiquement le maillage, et bien que la structure puisse enregistrer des polyèdres arbitraires, il faut définir au préalable chaque type de polyèdre utilisé.

2.2. Demi-arêtes

En dimension n , une demi-arête est la donnée de $n + 1$ cellules de chaque dimension incidentes les unes aux autres, ainsi que de n références vers d'autres demi-arêtes dites incidentes. Ces références permettent par exemple de tourner autour d'une face en itérant sur ses arêtes. En dimension 2, les demi-arêtes semblent faire l'unanimité mais en dimension supérieure il existe plusieurs façons de les généraliser, à commencer par les brins décrits plus haut.

2.2.1. Structure CIEL

La structure CIEL (Circular Incident Edge Lists) est présentée dans [CLCP05] : on y enregistre toutes les demi-arêtes, les 3-cellules et les sommets. Les faces sont déduites par des boucles utilisant les demi-arêtes. La spécificité de cette structure vient du fait que l'on encode une opération permettant d'orbiter autour d'un sommet afin de trouver tous ses voisins et demi-arêtes incidentes. Ces informations d'incidences accélèrent le calcul d'isovaleurs d'une fonction scalaire, rendant possible une visualisation efficace.

Des grilles polyédriques génériques peuvent être construites avec cette structure, mais son inconvénient majeur est que *toutes* les demi-arêtes sont encodées.

2.2.2. OpenVolumeMesh

Cette structure décrite dans [KBK12] n'encode par défaut que les incidences du haut vers le bas et peut représenter des domaines qui ne sont pas des variétés et qui comportent des éléments de dimensions différentes. Les relations d'incidences du bas vers le haut peuvent être ajoutées mais le gain de rapidité des opérations topologiques n'est pas toujours garanti puisqu'il faut alors mettre à jour ces nouvelles informations d'incidences. Le parcours des éléments reste par contre très efficace car ces derniers sont stockés dans des zones mémoires contiguës. Le remplacement des pointeurs par des entiers compacte aussi la structure.

2.2.3. Structure servant de base aux présents travaux

La structure décrite dans [VCMS12] a été conçue pour diminuer de façon drastique la consommation de mémoire des maillages polyédriques. Un ensemble *minimal* de demi-arêtes est utilisé pour garder l'information topologique complète. Chaque cellule d'une certaine dimension est représentée par une sous-structure qui contient une ou deux références vers un seul polyèdre incident. L'essentiel de l'information est gardé dans la structure de polyèdre : on y met les indices globaux de chaque cellule incidente ainsi qu'une suite ordonnée de demi-arêtes.

L'avantage certain de cette approche est la taille des maillages : il est possible de charger des grilles polyédriques possédant vingt millions de polyèdres pour un coût d'environ 8 Gb, géométrie comprise. Cependant seules les quasi-variétés peuvent être représentées avec cette structure et les requêtes topologiques passent nécessairement par une visite des polyèdres, d'où de moindres performances temporelles par rapport à d'autres structures. Cette remarque est à nuancer car les requêtes topologiques sont très rapides tant qu'elles concernent un seul polyèdre.

2.3. Intérêt des travaux

Cette dernière structure pourrait être satisfaisante puisqu'elle a permis de charger de grands modèles géologiques.

| Sous-structure | Données |
|-------------------------|--|
| Sommet | Une référence vers un polyèdre incident |
| Arête | Une référence vers un polyèdre incident |
| Polygone | Références vers les deux polyèdres incidents (ou un seul s'il s'agit d'une face de bord) |
| Demi-arête « minimale » | Indice local du sommet origine et indice local de l'arête correspondante |

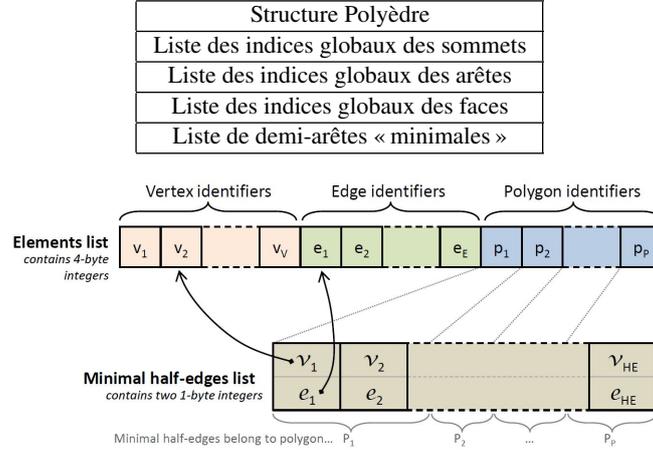


FIGURE 2: Mise en œuvre de la structure [VCMS12]. Le schéma est la figure 3 de [VCMS12] : on y voit les associations entre indices locaux de sommets et arêtes de demi-arêtes et indices globaux. Les indices globaux de polygones et ensembles correspondants de demi-arêtes sont ordonnés de la même façon.

| Opération | Signification |
|-------------------------|--|
| Next around polygon (a) | Demi-arête suivante au sein du même polygone dont l'origine est la cible de he |
| Mate polygon (b) | Demi-arête opposée liée au polygone adjacent à celui de he |
| Mate polyhedron (c) | Demi-arête opposée liée au polygone attaché à he, mais liée au polyèdre adjacent à celui de he |

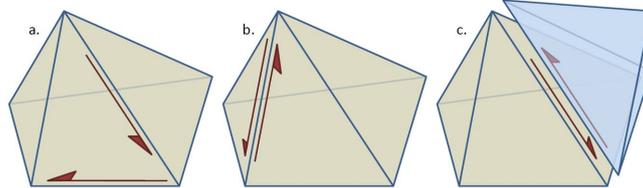


FIGURE 3: Opérations sur une demi-arête he et figure 1 extraite de [VCMS12].

Mais la génération de maillages peut faire intervenir des domaines tridimensionnels intermédiaires qui ne sont pas des quasi-variétés, par exemple lors d'une méthode d'avancée de front ou bien tout simplement en ajoutant une par une les cellules d'un diagramme de Voronoï.

Nous avons donc décidé d'améliorer cette dernière structure en insérant des informations supplémentaires pour encoder des connexions entre polyèdres qui ne relèvent plus seulement des quasi-variétés.

3. Description de la structure de données

3.1. Structure de base [VCMS12]

Les sous-structures nécessaires à la mise en œuvre de la base sur laquelle s'appuie ce travail sont données dans la figure 2.

Chaque élément de chaque dimension possède une identité, à savoir un identifiant sous forme d'entier. Pour tout élément e de dimension i comprise entre 0 et 2 inclus dans un polyèdre P , l'indice local à P de e est la position de l'indice de e dans la liste d'indices d'éléments de dimension i

de P . Cette conversion entre indices globaux et locaux permet d'utiliser des entiers « courts » dans les demi-arêtes et donc de compresser la structure.

Dans chaque polyèdre P , on enregistre pour chaque face f incidente à P une liste l_f de demi-arêtes qui réalise un cycle et permet donc de tourner autour de la face f , i.e. l_f contient uniquement et de façon ordonnée une demi-arête associée à chaque arête de f . On concatène ensuite ces listes l_f de toutes les faces incluses dans P pour obtenir la liste des demi-arêtes minimales. La condition sur cette liste est que le choix des l_f doit faire apparaître exactement deux demi-arêtes opposées par arête incluse dans P .

L'ordre des demi-arêtes constitue une information implicite qui allège la structure.

Trois opérations sont possibles sur les demi-arêtes afin de parcourir les éléments du maillage. Leur signification est donnée sur la figure 3.

À partir d'un polyèdre donné, il n'est permis de se déplacer qu'en passant à travers les faces. La figure 4 illustre un domaine incorrectement représenté par cette structure : on y voit une pseudo-variété, c'est-à-dire un complexe tel que pour tout couple de polyèdres, on puisse trouver un chemin de polyèdres deux à deux adjacents (par une face). La jonction du sommet n'est pas détectée par la structure de base.

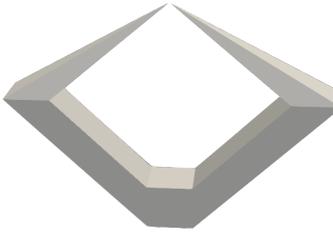


FIGURE 4: Exemple de pseudo-variété : il n'est pas possible de passer de l'un des polyèdres du haut à l'autre en ne passant qu'à travers des faces incidentes au sommet partagé.

Le cas des pseudo-variétés nous rappelle qu'il faut bien une contrainte supplémentaire sur les domaines que l'on peut représenter avec cette structure : si deux polyèdres P_1 et P_2 partagent un élément e , il doit être possible de traverser uniquement des faces incidentes à e pour passer de P_1 à P_2 . Ce sont donc seulement les quasi-variétés qui nous sont accessibles avec cette structure de base. La structure de données que nous introduisons dans ce travail (et que nous présentons au paragraphe suivant) apporte une solution à ce problème.

3.2. Sommets et arêtes charnières

La structure précédente est pourtant suffisante pour garder en mémoire un domaine sur lequel on appliquera des Vo-

lumes Finis par exemple, car on résout souvent des E.D.P.s sur des quasi-variétés. Le besoin d'adapter la structure aux complexes polyédriques homogènes[‡] est imposé par l'étape de génération de maillages.

Afin de permettre des connexions entre polyèdres selon des faces, des arêtes ou des sommets, deux tables de hachage sont ajoutées à la structure précédente. Dans la première table T_V , on associe à certains sommets dits charnières un ensemble d'indices de polyèdres incidents.

Dans la seconde table T_E on enregistre les associations de faces de bord partageant une arête dite charnière. Ce type d'association permet de passer d'une face de bord f à une autre qui serait inaccessible depuis f en orbitant autour de leur arête commune et en ne se déplaçant que par passage à un polyèdre adjacent. Un ensemble possible d'associations entre faces de bord est illustré à la figure 5. L'idée consiste à pouvoir récupérer les indices de tous les éléments incidents à une arête grâce à une boucle algorithmique.

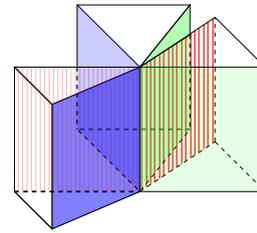


FIGURE 5: Trois prismes connectés selon une arête charnière : deux faces de bord appariées sont de la même couleur.

Ces deux tables peuvent être vides quand le maillage devient une quasi-variété. On retrouve alors la complexité spatiale de la structure précédente.

3.3. Demi-arêtes et orbiteurs

Pour faciliter les développements et l'intégration future du code, deux classes ont été programmées pour orbiter autour des sommets et arêtes (qui ne sont pas nécessairement des charnières).

La classe « orbiteur autour d'une arête » se sert des demi-arêtes pour parcourir les morceaux de maillages qui sont localement des quasi-variétés, tandis que la classe « orbiteur autour d'un sommet » dépend du premier orbiteur.

Le principe consiste à simplifier l'écriture de boucles permettant de parcourir les éléments du maillage incidents à un même sommet ou une même arête. Par exemple, en ce qui

[‡]. C'est-à-dire des complexes où tout élément de dimension inférieure à 2 est incident à au moins un polyèdre.

concerne un orbiteur autour d'une arête e , la classe détermine d'abord une demi-arête attachée à un polygone de bord et liée à e [§]. Ensuite on utilise la deuxième opération de la figure 3 : si le polygone associé à la nouvelle demi-arête n'est pas au bord, on utilise la dernière opération de la figure 3 puis on recommence. Sinon, on se sert de la table T_E pour trouver la face de bord correspondante. Si cette face de bord n'existe pas, le parcours est terminé. Sinon on continue le parcours et on recommence les étapes précédentes. Lorsque l'on revient sur la demi-arête de départ, l'algorithme se termine.

Les demi-arêtes permettent donc de parcourir les parties du maillage qui forment *localement* une quasi-variété tandis que la table T_E permet de passer d'un morceau de quasi-variété à un autre.

3.4. Optimisation

L'idéal serait de ne retenir dans la table T_V que des ensembles contenant un nombre minimal d'indices. Les ajouts et suppressions répétées de polyèdres nous obligent à considérer l'éventualité selon laquelle un sommet devienne une charnière. De temps en temps une optimisation doit être réalisée pour enlever de la table T_V les sommets qui sont isolés ou qui ne sont plus des charnières. De plus il peut être intéressant d'en profiter pour supprimer pour chaque sommet charnière v les indices de polyèdres que l'on peut déduire à partir des autres polyèdres incidents par rotation autour d'une arête (incidente à v) ou en passant à travers une face (incidente à v). Sur la figure 6, on s'aperçoit que l'information concrétisée par la flèche noire est facultative.

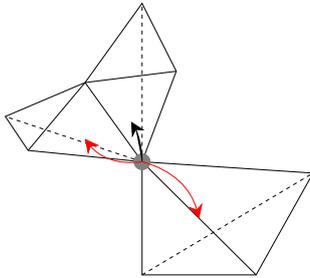


FIGURE 6: Un sommet charnière incident à trois tétraèdres : les flèches pointent vers les polyèdres que l'on peut enregistrer dans l'ensemble associé au sommet dans la table T_V . Comme l'un des tétraèdres en haut peut être déduit de l'autre en orbitant autour de l'arête partagée, il n'est nécessaire de retenir que deux indices de polyèdres.

[§]. Si une telle face de bord n'existe pas, n'importe quelle demi-arête associée à e convient.

Soit v un sommet et notons $p \stackrel{v}{\equiv} p'$ si deux polyèdres p et p' partagent une face ou une arête commune, incidente à v . Notons $\stackrel{v}{\sim}$ la fermeture transitive de la relation $\stackrel{v}{\equiv}$. Pour chaque sommet charnière v , l'étape d'optimisation consiste à calculer les classes d'équivalence pour $\stackrel{v}{\sim}$, dans l'ensemble des polyèdres incidents à v . Il suffit alors de garder l'indice d'un seul représentant de chaque classe d'équivalence dans l'ensemble $T_V(v)$ associé à v par T_V . Si cet ensemble $T_V(v)$ est un singleton, le sommet v n'est alors plus une charnière et peut être retiré de la table.

Diminuer le nombre d'entrées de la table T_V accélérera les futures recherches dans cette table. Remplacer les ensembles d'indices de polyèdres de cette même table par des ensembles plus petits diminuera l'empreinte mémoire de la structure et pourrait accélérer légèrement certaines requêtes topologiques. Une première stratégie d'optimisation consiste à alléger la table T_V dès que le nombre de sommets charnières à vérifier dépasse un seuil, que l'on étalonnera mieux avec des tests de performance.

4. Exemples

L'exemple à la figure 7 est composé de quatre tétraèdres se touchant seulement par un sommet. La table T_V associe donc au sommet central l'ensemble contenant les quatre indices des tétraèdres.



FIGURE 7: Quatre tétraèdres et un sommet charnière.

En insérant un tétraèdre comme sur la figure 8, une arête charnière apparaît. Puisque l'on peut déterminer les trois polyèdres du haut à partir d'un seul, le seul ensemble de T_V pourrait contenir 3 indices. Quant à l'arête charnière e , elle nécessite deux couplages de faces de bord. Par exemple ¶,

¶. Il existe dans ce cas deux façons de mettre à jour la table T_E . On pourrait très bien aussi coupler deux faces de bord qui ne sont pas du même côté (mais qui appartiennent à deux polyèdres distincts).

on peut coupler les deux faces de bord du devant (incidentes à e) ainsi que les deux faces de bord (incidentes à e) à l'arrière.

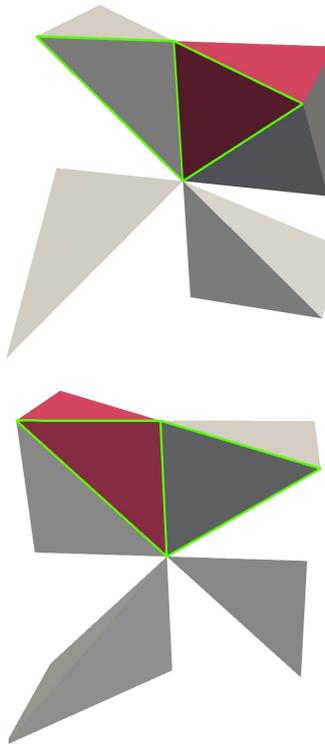


FIGURE 8: Cinq tétraèdres, un sommet charnière et une arête charnière : vue de face et de dos. Les faces de bord qui sont contenues dans la table T_E ont leur périmètre colorié en vert.

D'ailleurs, si on insère encore un tétraèdre à l'arrière comme sur la figure 9, l'arête charnière disparaît puisque l'on peut passer d'une face de bord à l'autre en orbitant autour de e par franchissement de polygones intérieurs. Le contenu de la table T_V ne change pas nécessairement.

5. Conclusion et perspectives

La mise en œuvre des charnières peut encore changer car l'on peut opter pour d'autres manières d'enregistrer les informations des tables de hachage. En effet, si le nombre moyen de polyèdres incidents à un élément charnière est limité, il serait judicieux d'utiliser un simple tableau ou une liste plutôt qu'une autre table de hachage ou un arbre ordonné.

Il reste encore à programmer quelques fonctions de division et fusion de cellules pour compléter la bibliothèque. Il est prévu également d'ajouter une fonction de simplification géométrique qui détecte les éléments alignés ou co-



FIGURE 9: Six tétraèdres et un sommet charnière : vue de face et de dos.

planaires que l'on pourrait fusionner pour obtenir quelque chose qui possède un sens géométrique. Des tests seront prochainement mis en œuvre pour mettre à l'épreuve la structure. Dès que la structure sera apte à charger des maillages volumineux, des tests de performance temporelle et spatiale permettront une meilleure comparaison avec les structures citées.

Remerciements

Nous remercions Wassim AHMED-BELKACEM pour l'image de pseudo-variété de la figure 4.

Références

- [CLCP05] CAUMON G., LÉVY B., CASTANIÉ L., PAUL J.-C. : Visualization of grids conforming to geological structures : a topological approach. *Computers & Geosciences*. Vol. 31, Num. 6 (2005), 671 – 680.
- [DAM10] DAMIAND G. : Contributions aux cartes combinatoires et cartes généralisées : Simplification, modèles, invariants topologiques et applications. Thèse d'habilitation à diriger des recherches en informatique, Septembre 2010.

- [FWWT13] FENG X., WANG Y., WENG Y., TONG Y. : Compact combinatorial maps : A volume mesh data structure. *Graphical Models*. Vol. 75, Num. 3 (2013), 149 – 156. Computational Visual Media Conference 2012.
- [KBK12] KREMER M., BOMMES D., KOBBELT L. : OpenVolumeMesh - A Versatile Index-Based Data Structure for 3D Polytopal Complexes. In *21st International Meshing Roundtable* (San Jose, United States, octobre 2012), Proceedings of the 21st International Meshing Roundtable, Springer, pp. 531–548.
- [VCMS12] VIARD T., CAVELIUS C., MALLISON B., SWORD C. H. : Data structure improvements for 3D polyhedral grids with application to unstructured discrete fracture models. . In *Proc. 32nd Gocad Meeting, Nancy* (2012).