



Couplage ossature-peau pour l'animation interactive d'objets déformables

Mathieu Coquerelle

► To cite this version:

Mathieu Coquerelle. Couplage ossature-peau pour l'animation interactive d'objets déformables. Synthèse d'image et réalité virtuelle [cs.GR]. 2004. inria-00598405

HAL Id: inria-00598405

<https://inria.hal.science/inria-00598405>

Submitted on 6 Jun 2011

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Couplage ossature-peau pour l'animation interactive d'objets déformables

Mathieu COQUERELLE

Rapport de stage

Evasion/GRAVIR/IMAG-INRIA

Composition du jury :

Augustin	LUX	
Gilles	DEBUNNE	
François	FAURE	Directeur de stage
John F.	HUGHES	Rapporteur externe



Table des matières

1	Motivations	4
1.1	Approche multi-résolution adaptative	4
1.2	Principales problématiques	5
1.2.1	Dynamique du milieu déformable	5
1.2.2	Affichage de la surface de l'objet	5
1.2.3	Gestion des collisions	6
1.3	Approche adoptée : FFD multi-résolutions	6
1.4	Organisation du document	6
2	État de l'art	7
2.1	Free-Form Deformations	7
2.2	FFD mécaniques	8
2.3	Axial Deformations, Wires et Skinning	9
2.4	Déformations multi-résolutions	9
3	Free-Form Deformations Multi-Résolutions Adaptatives	11
3.1	Octree déformable	11
3.1.1	Définition	11
3.1.2	Initialisation de l'octree	13
3.2	Application de l'octree aux objets déformables : réseau de masses-ressorts	14
3.2.1	Le quadtree : approche mécanique	14
3.2.2	Contraintes sur les sommets	16
3.2.3	Résultats	16
3.3	FFD par Skinning	17
3.3.1	Paramétrisation d'un point dans une cellule	18
3.3.2	FFD - Interpolation de type Skinning	20
3.4	Déformation multi-résolutions adaptative	29
4	Bilan et Conclusion	31
4.1	Bilan	31
4.2	Perspectives	32

Table des figures

2.1	Un grand nombre de cellules sont générées à l'édition d'un simple sommet (entouré en rouge) Ono et Al [14]	8
3.1	Manipulation d'une cellule	11
3.2	Contraintes topologiques	12
3.3	Manipulation multi-résolutions sur l'octree par déplacement d'un sommet	13
3.4	Répartition des masses sur les sommets : les poids sont à diviser par 16 (resp. 64) dans la figure de gauche (resp. droite)	15
3.5	Etude monodimensionnelle du comportement dû à la raideur	16
3.6	Etude du comportement dû à la raideur sur le quadtree	16
3.7	Animation de la sphère par manipulation de l'octree	17
3.8	Exemple de déformation en 2D par manipulation d'une cellule	18
3.9	Repère associé au "premier des sommets de la cellule" (entourés)	19
3.10	Repères et paramètres en différents sommets	19
3.11	Raccord entre deux cellules de même profondeur	21
3.12	Discontinuité au raccord due à la fonction de poids linéaire utilisée	22
3.13	Continuité C^1 , illustrations des tangentes	23
3.14	Illustration de l'influence des vecteurs formant les repères	24
3.15	Problème d'applatissage de la courbe avec un raccord C^2	25
3.16	Raccord entre deux cellules de profondeur différente	25
3.17	Propagation de l'influence du sommet central	27
3.18	Calcul du vecteur "moyen"	28
3.19	Effet de <i>popping</i> lors de la subdivision	29
4.1	Différentes déformations d'un plan défini par une triangulation de 100×100 points	33

Chapitre 1

Motivations

La déformation interactive et temps-réel d'objets dans un espace tridimensionnel est un sujet largement étudié dans le domaine de l'animation du fait de ses nombreuses applications, notamment dans les jeux vidéos et la chirurgie assistée par ordinateur. Les contraintes soulevées par le problème sont tout d'abord de travailler sur un modèle de déformations physiquement réalistes aux yeux de l'observateur et précises du point de vue mécanique. Lié à la dynamique ainsi qu'à sa représentation géométrique, l'aspect visuel de l'objet - images affichées sur l'écran - doit lui aussi satisfaire l'utilisateur, c'est-à-dire de minimiser les artefacts visuels induits par la discrétisation des informations au sein de l'ordinateur, autant que faire se peut. La dernière des contraintes est temporelle ; en effet, dans les domaines de la réalité virtuelle et de la synthèse d'images, les applications doivent être adaptées aux fréquences optiques et haptiques imposées par l'Homme, elles doivent rendre transparente l'utilisation de l'outil informatique pour immerger l'utilisateur dans un monde virtuel.

L'aspect géométrique dans le cadre de l'animation d'objet déformables a été peu développé, dans ce rapport, nous nous pencherons particulièrement sur la résolution de ce problème en proposant une solution de modélisation multi-résolution adaptative d'objets déformables.

1.1 Approche multi-résolution adaptative

Nous définissons ici la notion de multi-résolution (noté MR) comme la représentation d'une entité géométrique à plusieurs niveaux de détail. Cette notion de MR est proche des *Splines hiérarchiques* introduites par Forsey et Bartels [7] dans le sens où plusieurs niveaux de profondeurs différentes peuvent coexister au même moment sur différentes parties de l'entité. Cette représentation peut s'écrire structurellement sous forme d'un arbre où chaque noeud représente un niveau de détail, et est relié à ses niveaux de détail plus fins par une relation père-fils. Le terme *adaptatif* est utilisé ici pour définir le changement de résolutions au cours du temps en fonction d'un critère de décision quelconque. Ce changement doit être effectué de manière cohérente c'est-à-dire que cette transition doit s'effectuer de manière transparente du point de vue de l'utilisateur.

Cette approche est par exemple utilisée pour l'amélioration des performances de rendu d'objets 3D comme dans la technique des *Levels Of Details* (LOD) où l'objet est stocké dans une hiérarchie avec à chaque niveau une représentation avec nombre moins important de polygones ; s'il occupe beaucoup de pixels à l'écran, on utilise sa représentation la plus fine, et plus il s'éloigne, plus on utilisera une représentation grossière.

Nous cherchons, un modèle capable d'appliquer les LOD pour la visualisation (ajout de détails, simplification en fonction de la distance), la dynamique (focaliser les calculs sur les parties les plus déformées) et la détection de collisions (par l'utilisation de volumes englobants par exemple).

1.2 Principales problématiques

1.2.1 Dynamique du milieu déformable

Dans le domaine du graphisme lors des deux dernières décennies, les techniques d'animation physique d'objets déformables ont été largement étudiées. Les deux plus notables sont les réseaux de masse-ressort (*mass-spring* en anglais) et de particules ainsi que plus récemment par l'arrivée de plus fortes puissances de calcul, les éléments finis (*FEM : Finite Elements Method*).

La première technique se démarque par sa simplicité et la rapidité d'exécution, elle se base sur l'interaction de particules ou masses ponctuelles par des lois physiques simples telles que l'attraction-répulsion, l'élasticité, la viscosité, ...

Les éléments finis quant à eux consistent en la représentation de milieux continus par interpolation de valeurs définies en certains points. Plus complexe à résoudre, cette méthode est donc plus coûteuse que les réseaux de masses-ressorts mais est beaucoup plus précise. Dans les deux techniques, plus le nombre d'éléments devient important, plus la simulation est longue à calculer, et plus la contrainte de temps-réel devient difficile à respecter.

C'est dans le contexte d'objets complexes que l'approche multi-résolution adaptative apporte tout son intérêt car elle permet d'utiliser une version grossière du modèle dynamique lors de faibles déformations et de le complexifier lors de plus grandes déformations ou de manipulations plus locale de l'objet. Il est donc nécessaire d'adapter les techniques d'animations basées sur des structures statiques pour pouvoir les appliquer aux modèles multi-résolutions adaptatifs tout en respectant les contraintes liées aux modèles utilisés.

Ainsi, DeBunne et Al [5] proposent une représentation MR d'objets déformables par éléments finis. Le volume défini par l'objet est subdivisé en tétraèdres plus ou moins gros suivant la résolution, ces derniers servent d'éléments de base pour une simulation par FEM. A l'état stable, le découpage le plus simple est utilisé. Mais plus les forces appliquées au modèle sont importantes et plus l'on descendra dans la hiérarchie, c'est-à-dire plus il y aura d'éléments. Ceci permet d'obtenir des comportements plus complexes lors de manipulations précises et locales ou lors de grandes déformations tout en conservant une cohérence temporelle lors du passage à une résolution plus fine ou plus grossière.

1.2.2 Affichage de la surface de l'objet

Un aspect essentiel de la déformation d'objets 3D est la qualité des images présentées à l'utilisateur. En effet, il n'est pas suffisant de visualiser un comportement physiquement réaliste, il faut aussi que l'objet affiché soit visuellement et géométriquement correct et réaliste. Nous pouvons noter trois critères importants : la conservation des continuités géométriques ainsi que des discontinuités si elles existent, éviter les artefacts visuels lors d'un changement de résolution (effets de *popping*) et la manipulation de l'objet à n'importe quelle résolution doit respecter les contraintes précédentes.

De nombreuses techniques permettent de modifier dynamiquement la géométrie d'un objet, plus particulièrement une surface, à différentes résolutions.

Les *Splines Hierarchiques* décrite par Forsey et Bartels [7] permettent d'éditer une surface paramétrée de manière globale ou bien plus localement tout en assurant la conservation des continuités aux raccords entre les différentes résolutions. Très utilisées pour la modélisation d'objet 3D, elles permettent d'obtenir de très beaux résultats, mais sont contraintes par le fait qu'elles soient basées sur la paramétrisation de la surface ce qui n'est pas le cas d'un objet quelconque, ainsi que par le nombre de points de contrôle nécessaires (pour obtenir une continuité C2 entre patches, il faut un patch de 4×4 points de contrôle).

Les *Free Form Deformations* sont des techniques de déformation d'objets quelconques, qui ont été largement étudiées depuis l'article novateur de Sederberg et Parry [12]. Elles consistent à immerger l'objet dans une gelée déformable (de topologie quelconque mais simple à manipuler, par exemple un cube), de déformer cette gelée à partir de points de contrôles et ainsi la déformation est répercutée à l'objet initial. Les FFD sont largement utilisées dans le domaine de la modélisation 3D car elles permettent, assez intuitivement, de déformer tout type d'objet

de manière lisse. Malheureusement, peu de travaux traitent de FFD multi-résolutions adaptatives et les solutions proposées ne permettent pas une manipulation MR simple (du point de vue technique) et efficace (du point de vue de l'utilisateur).

1.2.3 Gestion des collisions

Dans l'optique de réaliser un modèle d'animation d'objets déformables complet, il est nécessaire de résoudre le problème des collisions de l'objet avec lui même (auto-collisions) ou avec d'autres objets (collisions externes). Ce problème est décomposé en deux parties : tout d'abord la détection des collisions, puis la réponse physique à associer à cette collision. La résolution du premier problème peut s'avérer très coûteuse car pour être complet, dans le cas d'objets quelconques, il convient de vérifier qu'aucun des polygones des modèles 3D ne s'intersectent ce qui a pour coût algorithmique $O(n^2)$. De nombreuses méthodes permettent de réduire grandement le coût de la phase de détection de collisions, notamment en utilisant une représentation de l'objet par une hiérarchie de volumes englobants (contenant des polygones aux feuilles et des boîtes englobantes aux noeuds) de forme simple (par exemple une sphère).

1.3 Approche adoptée : FFD multi-résolutions

Nous cherchons une méthode permettant de traiter efficacement et de manière multi-résolution les trois aspects du problème de l'animation d'objets déformables : la géométrie, la dynamique et la gestion des collisions. Nous avons adopté une approche hiérarchique (de structure régulière) de découpage spatial multi-résolution de l'objet. L'idée est d'utiliser cette structure comme base pour la dynamique, pour les collisions et comme champ de déformation pour la géométrie de l'objet à la manière des FFD. Ce champ de déformation est en conséquence entièrement indépendant de l'objet déformable.

Nous proposons une technique qui permet de palier les limitations des méthodes de déformation géométriques multi-résolutions actuelles, notamment la complexité provenant des opérations de raffinement et de simplification, le coût de la déformation de l'objet à des résolutions différentes, la rapidité d'affichage ainsi que le contrôle direct de l'animation physique sous-jacente. Nous validons aussi l'approche du point mécanique en présentant un réseau de masses-ressorts adapté à la multi-résolution.

1.4 Organisation du document

Nous commencerons par étudier les techniques actuelles de déformations lisses d'objets. Nous présenterons ensuite notre contribution qui permet de palier certaines limitations des méthodes de déformation géométriques multi-résolutions actuelles, notamment la complexité provenant des opérations de raffinement et de simplification, le coût de la déformation de l'objet à des résolutions différentes, la rapidité d'affichage ainsi que le contrôle direct de l'animation physique sous-jacente. Nous validerons aussi l'approche du point de vue mécanique en présentant un réseau de masses-ressorts adapté à la multi-résolution. Nous fournirons ensuite quelques exemples de résultats obtenus grâce à notre technique. Puis nous terminerons en effectuant un bilan nous fournirons plusieurs pistes pour des travaux futurs.

Chapitre 2

État de l'art

Depuis l'apparition des applications 3D dans le monde de l'informatique, la modélisation et l'animation d'objets 3D en est devenu un des sujets phares. Selon le type d'applications (temps-réel, interactif, ...), différentes techniques ont été introduites, certaines dans un objectif de création d'objets à la manière d'un sculpteur (tels que les *Splines Hierarchiques* de Forsey et Bartels [7]), d'autres dans le but de déformer un objet existant (particulièrement les *Free-Form Deformations*) et certaines visant à allier les deux (lors de manipulations directes).

2.1 Free-Form Deformations

Il y a une dizaine d'années, la modélisation assistée par ordinateur d'objets 3D demandait beaucoup de temps, de soin et de patience. L'arrivée de nombreux artistes dans le monde de l'infographie a imposé la nécessité de développer des outils divers et simples permettant aux non néophytes de l'informatique de s'abstraire au maximum du support sur lequel ils travaillent et ainsi de pouvoir laisser libre cours à leur imagination lors de la création d'images virtuelles. Le monde de la modélisation 3D s'est tout particulièrement rapproché de celui de la sculpture, notamment de la sculpture sur des matériaux tels que la glaise, permettant de déformer, ajouter, supprimer de la matière simplement avec les mains.

Originellement introduite par Sederberg et Parry [12], les FFD sont aujourd'hui un des outils les plus pratiques, car ils permettent très aisément de déformer de manière lisse tout type d'objet 3D. Son principe en est simple, il consiste à plonger l'objet dans un espace déformable, une sorte de gelée, défini par un ensemble de points de contrôle nommé **treillis** (*lattice* en anglais) de topologie simple et définissant la notion de cellules (l'espace défini par un ensemble de sommets du treillis en fonction de leur connexité). Le type de treillis présenté dans leur article provient du découpage régulier (mais possiblement différent dans chacune des dimensions) d'un parallélépipède. Un triplet de paramètres est associé à chaque point du maillage, définissant sa position dans l'espace. Lorsque le treillis est déformé par l'utilisateur, les déformations apportées sont répercutées aux points du maillage de l'objet 3D par l'évaluation simple de produits de polynômes de Bernstein apportant ainsi une déformation volumique de type Spline.

Ils montrent aussi qu'il est possible de raccorder deux treillis. En imposant des contraintes au raccord entre deux cellules, il est possible d'obtenir une continuité C^2 entre les deux treillis. Pour ce faire, il faut que chaque treillis possède au moins $4 \times 4 \times 4$ points de contrôle et que les segments orthogonaux à la face commune soient colinéaires. Cette contrainte est assez forte et il n'est donc pas envisageable de l'utiliser dans le cadre d'une animation physique. Un autre point important à noter est que plus le nombre de points de contrôle augmente, plus le degré des polynômes augmente et ainsi plus l'évaluation de la position d'un point dans l'espace devient coûteuse.

Par la suite, Coquillart [4] proposa les *Extended Free-Form Deformations* proposant une approche plus générale de la technique, permettant d'utiliser un treillis de topologie plus complexe. MacCracken et Joy [11] quand à eux introduisent une méthode de déformation à partir d'un

treillis de topologie arbitraire et la possibilité d'effectuer des opérations de subdivisions sur un treillis en se basant notamment sur les travaux de Catmull et Clark [2] pour définir la position des nouveaux points de contrôle tout en conservant la continuité.

Plus récemment, Hua et Qin ont proposé un type de FFD basé sur un ensemble d'interactions entre des champs d'influence définis par des sommets de contrôle (ainsi, l'on s'abstrait des problèmes de topologie), à la manière de la modélisation de surfaces implicites. Leur méthode permet d'obtenir une très grande variété de déformations car le domaine des fonctions d'influence est très large, ainsi qu'un raffinement du maillage déformé lorsque le maillage initial n'est pas assez fin pour conserver un aspect visuel lisse après déformation. Malheureusement, même si il est possible de générer un réseau de sommets de contrôle automatiquement, l'ajout dynamique de points de contrôles à des résolutions plus élevées n'est pas trivial et l'application d'un modèle de déformations dynamiques cohérentes n'est pas évident.

En se basant sur les travaux de MacCracken et Joy, Ono et Al [14] ont dernièrement introduit une méthode d'édition multi-résolutions d'un objet à partir de treillis hiérarchiques. Ils rendent l'utilisation des FFD plus intuitives en générant automatiquement les treillis à différents niveaux de résolutions autour de l'objet, par subdivisions des cellules d'un octree. Cette technique permet, par la manipulation des points de contrôle à différents niveaux de résolution, de déformer l'objet de manière locale ou bien plus globalement en conservant les modifications apportées à des niveaux inférieurs. Ils permettent à l'utilisateur de pouvoir subdiviser lui même une cellule donnée, ceci entraîne un octree de topologie plus complexe (comme présenté dans la figure 3.2(a)), mais l'édition locale d'un des nouveaux points de contrôle ainsi créés nécessite l'existence de $7 \times 7 \times 7$ cellules de même niveau autour pour pouvoir assurer une continuité C^2 de déformation de l'objet entre les cellules. Cela oblige, à l'édition d'un point de profondeur supérieure et proche (moins de 4 cellules) d'une cellule de plus au niveau, la subdivision de cette dernière au même niveau de profondeur (cf figure 2.1). En conséquence, bien que leur méthode soit améliorable en la rendant adaptative (c'est-à-dire sans stocker n niveaux de subdivisions uniformes de l'octree, en effectuant donc les subdivisions/simplifications dynamiquement), elle est toujours aussi coûteuse que la technique de MacCracken et Joy et elle ne permettra pas de pouvoir animer efficacement un objet déformé de manière globale et locale, simultanément.

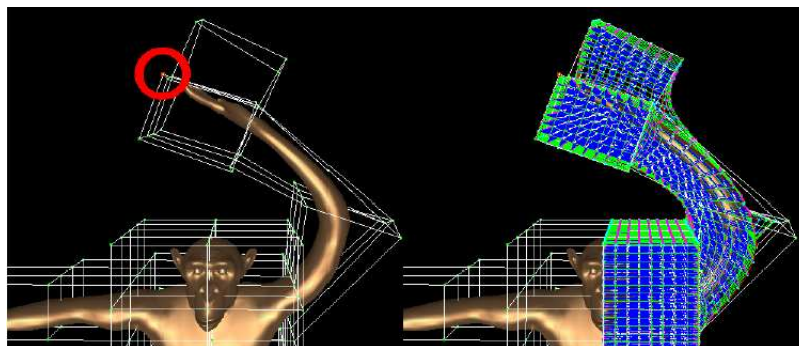


FIG. 2.1 – Un grand nombre de cellules sont générées à l'édition d'un simple sommet (entouré en rouge) Ono et Al [14]

2.2 FFD mécaniques

L'animation par simulation physique d'un objet quelconque (principalement non articulé) est un domaine ayant donné lieu à de nombreuses contributions. Le lien entre l'animation physique et la déformation géométrique de l'objet induite doit permettre d'obtenir des comportements réalistes aussi bien visuellement que physiquement. L'utilisation des FFD permet de déformer de manière très intuitive un maillage quelconque, ainsi Faloutsos et Al [6] présentent un modèle de déformation physique basé directement sur les treillis basé sur des réseaux des masses-ressorts. Ils obtiennent ainsi des résultats très convaincants en s'abstrayant du modèle géométrique. L'anima-

teur définit un ensemble de treillis spatialement indépendants à des niveaux de détail différents et par un couplage mécanique de ces treillis, il est ainsi possible de déformer l'objet de manière multi-résolutions. Ils utilisent des treillis découpés régulièrement, fournis par l'utilisateur, ceci ne permet pas d'obtenir un comportement multi-résolution adaptatif simple et automatique.

2.3 Axial Deformations, Wires et Skinning

L'utilisation de modèles articulés, comme des personnages par exemple, pour l'animation par manipulation directe ou indirecte d'un *squelette* pose le problème de la répercussion des mouvements sur la peau représentée par un maillage. En effet, il est nécessaire d'avoir une technique qui permette de déformer de manière lisse les points 3D sans faire apparaître de discontinuités au niveau du maillage.

Les articles de Lazarus et Al [10] (*Axial Deformations*) et de Singh et Fiume [13] (*Wires*) présentent une technique de déformation de maillage basée sur la paramétrisation des points du maillage par rapport à une primitive de dimension 1 : un segment ou une courbe. Ainsi, par toute manipulation de la courbe (par exemple) : torsions, étirements, ..., il est possible d'appliquer ces types de déformations aux points du maillage associé à cette courbe. Il est ainsi possible d'obtenir le même genre de comportements que Barr [1] obtenait en se basant uniquement sur un axe central, mais ici de manière plus générale avec la possibilité de limiter l'influence de la déformation aux points proches de la courbe.

Une simplification de cette méthode, nommée **Skinning**, permet d'obtenir un type de déformations similaires, plus limitées mais très efficaces (rapides à calculer et aujourd'hui accélérée sur les cartes graphiques) et non contraintes le long d'une courbe. Elle est basée sur l'association d'un point du maillage à plusieurs repères dans l'espace et pour chacun de ces repères est attribué un poids, généralement fonction de la distance du point à l'origine du repère. A l'état initial (avant l'animation), les coordonnées locales Q_i dans le repère R_i d'un point de position Q ainsi que les poids correspondants sont calculés. Ainsi, la position Q du point peut s'écrire en coordonnées homogènes :

$$Q \sim \sum_i w_i R_i Q_i$$

où le signe $\sim$ dénote l'égalité à un facteur d'échelle près et w_i représente le poids associé au repère R_i . Ainsi, par la manipulation des repères (position de l'origine, norme et direction des vecteurs), il est possible d'appliquer plusieurs types de déformations au maillage : grossissements, torsions, rotations et étirements.

De nombreux travaux ont été effectués concernant la déformation réaliste d'un être personnage virtuel, la principale difficulté est le calcul des poids et des repères associés aux points. Mohr et Gleicher [8] présentent une méthode permettant de régler les principaux problèmes introduits par le skinning dans ce cadre : l'applatissage ou le vrillement des membres.

2.4 Déformations multi-résolutions

Les méthodes classiques d'édition multi-résolutions possèdent ces principaux inconvénients :

- Les ondelettes : pour connaître la position d'un point défini à une profondeur n , il faut évaluer tous les niveaux inférieurs, ceci peut s'avérer coûteux en mémoire et temps de calcul si la profondeur est élevée.
- Les FFD multi-résolutions précédentes : pour conserver une continuité C^1 , les points du maillage sont définis par des patches de $3 \times 3 \times 3$ points de contrôle. Ceci s'avère très contraignant car à la limite entre deux patches de résolution différente, il est nécessaire d'appliquer un fastidieux changement de topologie des cellules voisines permettant de respecter cette contrainte.

Nous voulons éviter ces problèmes qui limitent l'utilisation de ces techniques dans le cadre d'applications temps-réel. Ainsi, nous nous fixons comme contrainte l'utilisation d'un minimum d'infor-

mations (paramétrisation et points de contrôles) pour pouvoir effectuer de la manière la plus libre et efficace qu'il soit, des déformations multi-résolutions en profitant des accélérations graphiques développées pour le skinning.

Chapitre 3

Free-Form Deformations Multi-Résolutions Adaptatives

Nous présentons maintenant notre contribution qui consiste au rapprochement entre les méthodes de FFD et de skinning présentées précédemment, permettant la déformation à différentes résolutions d'un objet 3D efficace, avec la possibilité de l'utiliser de manière adaptative pour l'animation d'objets déformables.

3.1 Octree déformable

3.1.1 Définition

Nous proposons une forme de treillis permettant l'édition MR de l'objet par FFD. Elle est basée sur une représentation hiérarchique de découpage de l'espace couramment utilisée l'**octree**. Un octree est un ensemble de cellules de forme cubique liées par des relations de parenté. Une cellule de l'octree est soit une feuille soit une cellule mère. Dans ce cas, elle possède exactement 8 cellules filles provenant du découpage régulier de la cellule mère en 2 (dans chacune des dimensions) (cf figure 3.1(b)).

Une cellule est composée de 6 faces, 12 arêtes et définie par 8 sommets. Les sommets communs aux cellules sont uniques, c'est-à-dire qu'ils sont partagés par toutes les cellules qui y sont connectées. Ainsi, logiquement, le déplacement d'un sommet entraînera le déplacement de toutes les faces et les arêtes lui étant connectées (cf figure 3.1(c)). Il n'existe pas de contrainte sur la forme des faces, entres autres, elle peuvent ne pas être planaires.

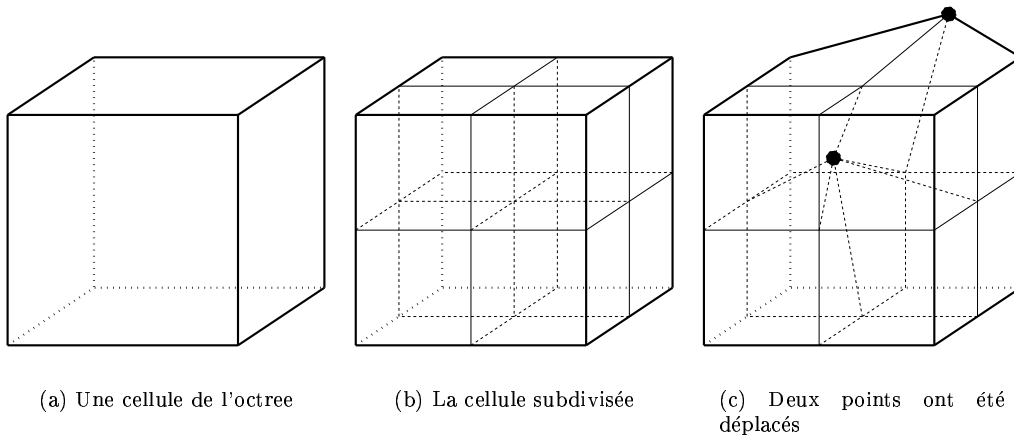


FIG. 3.1 – Manipulation d'une cellule

La particularité de l'octree est que des cellules voisines peuvent être de profondeur différente, il en résulte alors une forme géométrique de topologie contrainte car topologiquement, les faces d'une cellule feuille (une cellule n'ayant pas de fille) ne sont paramétrées que par les 4 sommets les formant. Ainsi, nous définissons la notion de *sommet libre* comme étant un sommet non soumis aux contraintes de topologie, un sommet de ce type peut être déplacé librement ; dans le cas contraire, nous le notons comme *sommet contraint*. Par exemple, dans la figure 3.2(a) nous voyons que si le sommet entouré est déplacé horizontalement, alors il violera l'intégrité de la cellule voisine de plus haut niveau. Par la suite lorsque cela sera nécessaire, dans les schémas, nous noterons les sommets libres par des ronds pleins et les sommets contraints par des cercles.

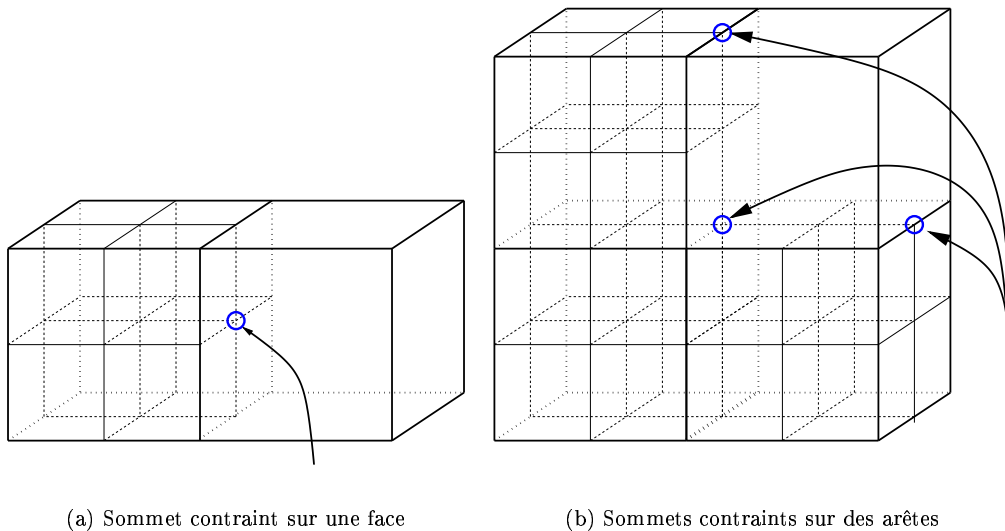


FIG. 3.2 – Contraintes topologiques

Lors de la subdivision d'une cellule, 19 sommets sont créés ($3 \times 9 = 27, 27 - 8 = 19$) que l'on peut classer dans 3 types différents :

- un *sommet de cellule* est le barycentre des 8 points de la cellule génitrice. Par construction, ce type de sommet est toujours libre. Il n'en existe qu'un.
- un *sommet de face* est le barycentre des 4 points formant une des 6 faces de la cellule génitrice. Si, dans la cellule de l'autre côté de la face ce sommet n'existe pas déjà, alors ce nouveau sommet est contraint par ses 4 sommets parents.
- un *sommet d'arête* est le barycentre des 2 points formant une des 12 arêtes de la cellule génitrice. Il est libre si et seulement si toutes les cellules de même niveau possédant cette arête ne sont pas des feuilles. Dans le cas contraire, il est contraint par ses 2 sommets parents.

Il faut noter que le déplacement d'un sommet contraint ne casse pas obligatoirement la contrainte topologique, par exemple si l'on déplace un sommet d'arête contraint le long de l'arête, la topologie de la cellule restera inchangée, mais nous verrons plus tard que pour des raisons géométriques, même ce genre de déplacement est interdit. Également, la cellule principale (la première dans l'arborescence) n'a pas de voisine, ainsi, si une cellule se trouve au "bord", les faces incluses dans les faces de la cellule principale n'ont elles non plus pas de voisine.

Lors de la simplification d'une cellule, il est nécessaire de :

- supprimer les sommets connectés uniquement aux cellules filles.
- conserver les autres sommets qui sont utilisés par des cellules voisines et qui deviennent ainsi contraints. Si le sommet était libre, alors il devient contraint est redevient alors le barycentre de ses parents.

Ainsi, lors des opérations de subdivisions et simplifications, l'arbre des relations mère-fille des cellules ainsi que les liens topologiques et géométriques (présentés plus haut) liant les sommets à

leurs parents sont mis à jour.

Il est ainsi possible de manipuler les sommets de l'octree au niveau de profondeur désiré, nous appelons cette méthode l'édition multi-résolutions. Nous définissons le niveau d'un sommet comme étant la profondeur de la plus grosse des cellules y étant connectées. Un exemple de déformation multi-résolutions est donné dans la figure 3.3.

Si l'on déplace un sommet S de niveau n à un niveau m ($m \geq n$), il faudra alors déplacer tous les sommets contraints par ce sommet (ils sont tous de niveau supérieur à n) et tous les sommets libres de profondeur supérieure ou égale à m descendants de ce sommet. Les sommets contraints restent le barycentre de leurs parents et les sommets libres sont déplacés de manière rigide, c'est-à-dire de manière cohérente avec leur position barycentrique initiale : si $S_{initial}$ est la position initiale d'un sommet : $S_{initial} = \sum_{j=1}^{N_P} S_j / N_P$ (où N_P représente le nombre de parents défini par le type de sommet : cellule, arête ou face), alors si l'on ajoute un vecteur V à un des sommets parents S_j , nous obtenons : $S = \sum_{j=1}^{N_P} S_j / N_P + V / N_P$. Ainsi, par un raisonnement récursif basé sur les liens de parenté des sommets, la propagation du déplacement s'effectue de manière correcte. En conséquence, tous les sommets libres ou non descendants du sommet S de niveau inférieur à n ne sont bien sûr pas concernés par ce déplacement.

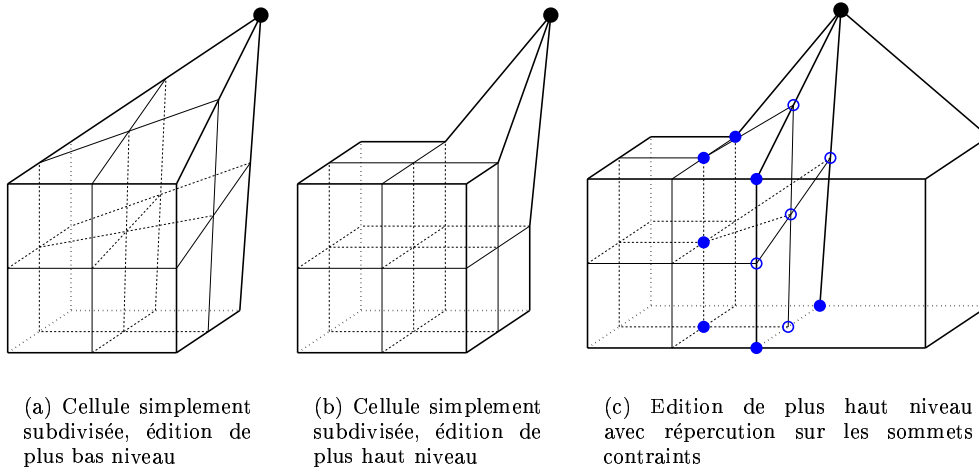


FIG. 3.3 – Manipulation multi-résolutions sur l'octree par déplacement d'un sommet

Grâce aux contraintes données précédemment et aux opérations de simplification et subdivision, il est possible d'effectuer une édition multi-résolutions de la structure d'octree permettant de travailler localement et ensuite plus globalement tout en conservant les modifications appliquées aux branches les plus profondes de l'arborescence.

3.1.2 Initialisation de l'octree

Nous devons maintenant appliquer ce treillis à un objet 3D dans l'optique future de le déformer. Il est possible de voir l'objet comme étant le volume défini par une infinité de cellules de matière. Comme il est présenté dans [14], nous proposons un découpage de l'espace occupé par l'objet de manière hiérarchique. Partant d'une cellule englobant la totalité de l'objet, nous pouvons créer l'arborescence de cellule grâce à des critères simples tels que :

- la densité de matière contenue dans la cellule : si la cellule n'intersecte pas l'objet ou si elle est contenue entièrement dedans, alors il est inutile de la subdiviser, sinon il faut la subdiviser.
- la densité de points du maillage dans la cellule : si la cellule ne possède pas de point du maillage ou n'en possède qu'un seul, alors il est inutile de la subdiviser, sinon il faut la subdiviser.

Le problème de ce genre de découpage est qu'il peut entraîner un nombre très important de cellules, ce qui rendrait son utilisation prohibitive. Il est donc nécessaire de limiter le processus

de subdivision, les critères d'arrêt possibles sont : une densité de matière suffisante, une densité de points suffisante ou bien une profondeur maximum.

Le premier critère de subdivision proposé peut s'avérer complexe à évaluer, dans le cadre de ce stage nous avons donc utilisé le second critère avec une borne sur le nombre de points maximum contenus dans une cellule.

Les déformations appliquées aux points de l'octree se feront uniquement dans les feuilles, elles seront paramétrées localement par ces cellules. Nous présentons dans la partie 3.3 comment utiliser l'octree pour déformer un objet quelconque de manière lisse après en avoir réparti les points dans les cellules de l'octree.

3.2 Application de l'octree aux objets déformables : réseau de masses-ressorts

Nous présentons maintenant une méthode préliminaire concernant l'application d'un modèle physique, un réseau de masses-ressorts cohérent avec la structure d'octree étudiée précédemment. Pour plus de clarté, nous baserons notre explication sur le quadtree : l'équivalent de l'octree en 2D, concrètement nous l'avons appliqué à l'octree.

3.2.1 Le quadtree : approche mécanique

Nous allons maintenant définir et étudier les propriétés mécaniques que nous associons à un quadtree.

Les masses

Une cellule feuille du quadtree représente un carré de matière possédant une certaine masse. Par une opération de subdivision, 4 cellules feuilles sont créées, il faut donc répartir la masse de la cellule mère dans ses filles. Chaque cellule feuille peut répartir sa masse sur ses 4 sommets.

Nous prouvons maintenant la conservation de la masse totale d'un quadtree par subdivision des cellules. Soit une cellule C de masse $masse(C) = m$. Après subdivision, chaque cellule fille C_f possède donc une masse de $masse(C_f) = masse(C)/4 = m/4$. Chaque cellule fille contribue par un quart de sa masse en chacun de ses sommets. Un sommet s partagé par les cellules C_i a donc pour masse :

$$masse(s) = \frac{1}{4} \sum_i masse(C_i).$$

Par subdivision, nous obtenons 9 sommets dont :

- 4 sont partagés par une seule cellule, de masse $1/4 * m/4 = m/16$
- 4 sont partagés par deux cellules, de masse $1/4 * (m/4 + m/4) = m/8$
- 1 est partagé par les 4 cellules, de masse $1/4 * (m/4 + m/4 + m/4 + m/4) = m/4$.

Nous obtenons donc pour la masse totale m_{totale} après subdivision :

$$m_{totale} = \sum_s masse(s) = 4 * m/16 + 4 * m/8 + m/4 = m$$

qui est bien la masse de la cellule mère avant subdivision.

On peut vérifier ce résultat sur les illustrations de la figure 3.4.

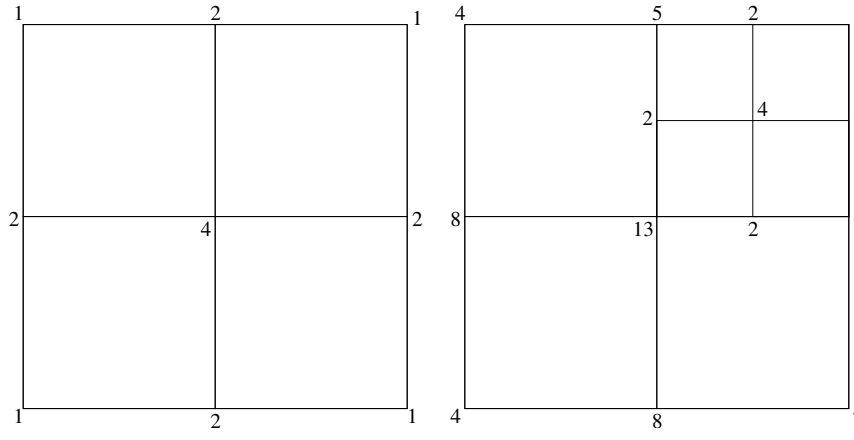


FIG. 3.4 – Répartition des masses sur les sommets : les poids sont à diviser par 16 (resp. 64) dans la figure de gauche (resp. droite)

Les ressorts

Au niveau de chaque cellule feuille, nous appliquons un réseau de masse-ressort-viscosité, un ressort pour chaque arête. Pour pouvoir pallier le problème d'écrasement, nous ajoutons des ressorts sur les diagonales.

Chaque ressort à une longueur à vide de la taille de la cellule, car par subdivision la longueur d'une arête est divisée par deux. Nous avons choisi une viscosité et une raideur constante pour tout le réseau. Nous expliquons pourquoi nous avons fait ce choix pour la raideur K .

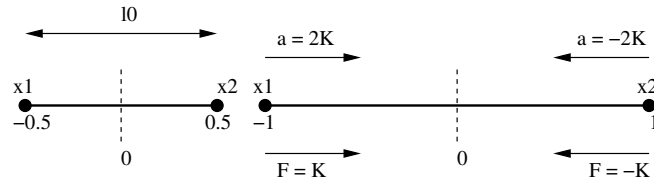
Si l'on se place dans le cas d'un ressort (sans amortissement) monodimensionnel de longueur à vide $l_0 = 1$ et raideur K entre deux points x_1 et x_2 de masse $1/2$, et que l'on tire les deux points de 0.5 de chaque côté (cf figure 3.5(a)), ils recevront chacun une force de valeur plus ou moins $F = -K((x_1 - x_2) - l_0) = -K$ et donc leur accélération sera de plus ou moins $a = -2K$. En subdivisant le segment, nous obtenons deux points de masse $1/4$ aux extrémités (x'_1 et x'_2) et un point de masse $1/2$ au milieu (x'_3) (cf figure 3.5(b)). Les deux ressorts associés possèdent une raideur de K' ainsi qu'une longueur à vide de $l'_0 = l_0/2 = 0.5$. De la même manière que précédemment, si l'on tire les deux extrémités de 0.5 , nous trouvons comme forces associées à x'_1 et x'_2 plus ou moins $F' = -K'((x'_1 - x'_3) - l'_0) = -K'/2$, et le point x'_3 une force nulle ($F' - F'$). Les accélérations pour les deux extrémités sont donc de plus ou moins $a' = -4 * K'/2 = -2K'$.

Maintenant, dans le cadre d'un quadtree, si l'on se place dans la dimension horizontale et que l'on tire de la même manière les deux arêtes verticales aux extrémités (cf figure 3.6), il est facile de voir que le point x_1 se voit toujours appliquer une force de $F = K$ et x'_1 une force de $F' = K'/2$. Mais leur masse ne diffère plus d'un facteur deux mais d'un facteur quatre ($1/4$ et $1/16$), leurs accélérations respective sont donc de : $a = 4K$ et $a' = 8K'$.

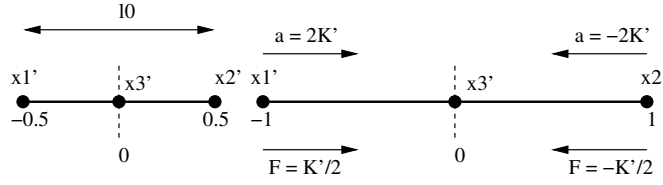
Si l'on choisit de prendre $K' = 2K$, nous obtenons $F' = F$ et $a' = 4a$: la force appliquée est la même, mais l'accélération a quadruplé.

Si l'on choisit de prendre $K' = K/2$, nous obtenons $F' = F/4$ et $a' = a$: la force appliquée est divisée par quatre, mais l'accélération est restée la même.

Ces deux choix offrent des comportements différents (non étudiés ici), et dans l'optique de trouver un compromis, nous avons choisi la solution $K' = K$.



(a) Un seul ressort



(b) Deux ressorts

FIG. 3.5 – Etude monodimensionnelle du comportement dû à la raideur

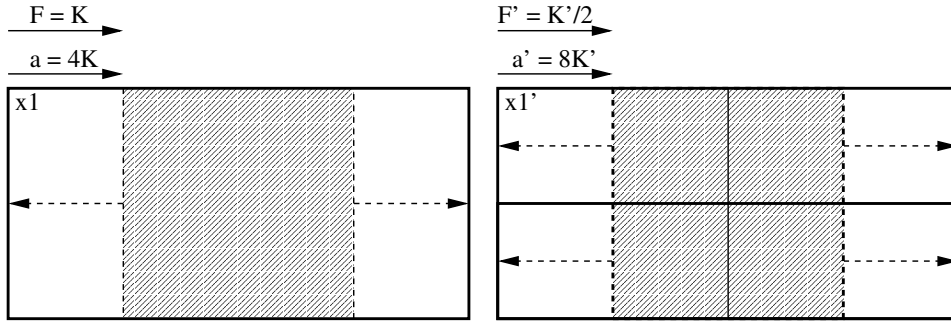


FIG. 3.6 – Etude du comportement dû à la raideur sur le quadtree

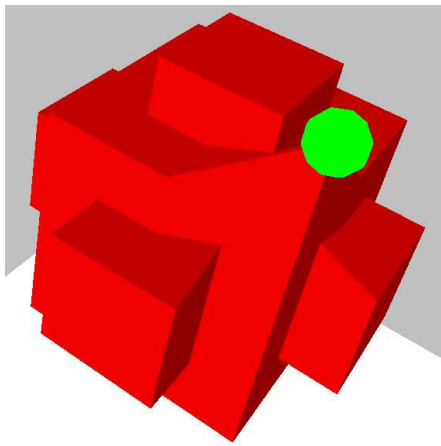
3.2.2 Contraintes sur les sommets

La structure de quadtree sur laquelle nous basons ce travail introduit la notion de sommets contraints (présentée plus haut). Comme nous l'avons expliqué, ces sommets n'apportent pas d'information concrète car ils sont purement dépendants de leurs parents.

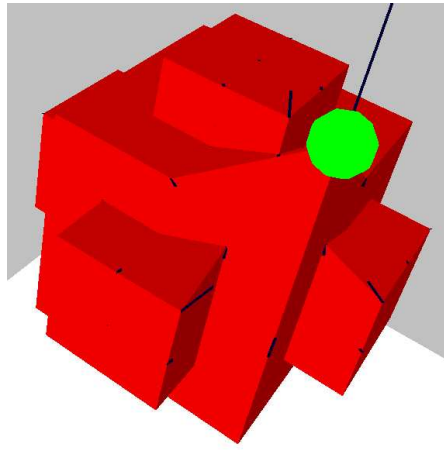
Ganovelli et Al [3] proposent comme solution pour prendre en compte la particularité de tels sommets, d'ajouter un ressort monodimensionnel (basé sur une face ou une cellule) entre ces sommets et leur parents. Notre approche consiste à ignorer le calcul de la dynamique en de tels sommets, mais pour cela il faut au préalable répartir leur masse équitablement chez leur parents. La conservation de la masse totale est assurée ainsi que la cohérence de l'octree.

3.2.3 Résultats

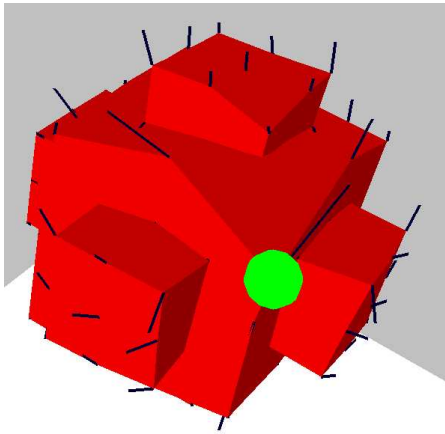
Nous présentons dans la figure 3.7 des captures d'écran effectuée lors de l'animation d'une sphère implicite plongée dans un octree, la manipulation consistait à tirer sur un des sommets de l'octree pour déplacer le volume. Il est à noter que la géométrie est très simple, nous avons simplement rempli en rouge les cellules contenant de la matière.



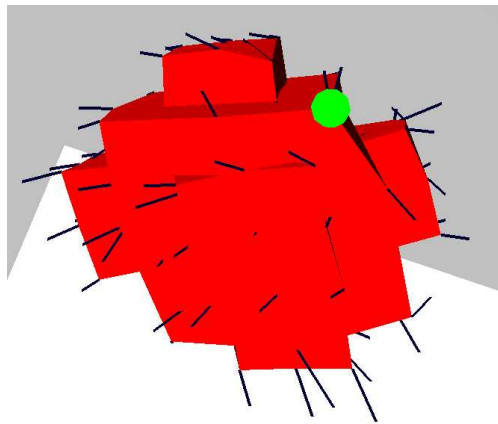
(a) Manipulation d'un sommet (ajout de vitesse vers le haut) sur un corps mou



(b) Idem, avec affichage des vecteurs vitesses



(c) La simulation après quelques itérations



(d) Ici, plusieurs manipulations ont été effectuées en peu de temps, le volume est très déformé et tourne sur lui même

FIG. 3.7 – Animation de la sphère par manipulation de l'octree

3.3 FFD par Skinning

Nous désirons tirer partie de cette structure multi-résolution contrainte pour permettre de déformer de manière lisse un objet quelconque en conservant les avantages des FFD. Nous présentons maintenant l'approche géométrique que nous avons développée, la **FFD par skinning**, approche axée sur les techniques présentées dans l'état de l'art et sur le type de treillis défini précédemment. Elle se base sur la paramétrisation des points dans une cellule de l'octree par rapport à ses sommets, et par déformation de cette cellule, les points sont déplacés et ainsi l'objet est déformé (cf illustration en 3.8 pour un exemple en 2D).

Nous commencerons par expliquer la paramétrisation de l'objet à l'intérieur des cellules de l'octree, puis nous montrerons comment déformer de manière lisse le maillage contenu dans une cellule unique. Ensuite nous réglerons le problème de raccord entre cellules voisines de même niveau puis élargirons la méthode au voisinage de cellules de profondeurs différentes. Nous terminerons en expliquant comment il est possible de déformer un objet plongé dans l'octree à plusieurs résolutions différentes de manière adaptative.

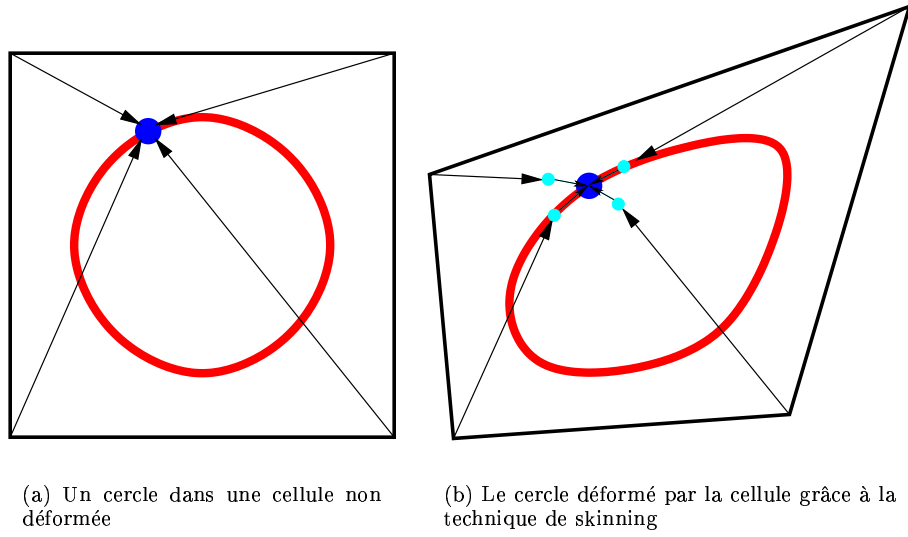


FIG. 3.8 – Exemple de déformation en 2D par manipulation d’une cellule

3.3.1 Paramétrisation d’un point dans une cellule

Chaque cellule feuille de l’octree englobe un certains nombre de points du maillage et chaque point du maillage est associé à une unique cellule feuille de l’octree. Pour l’instant, nous nous basons sur des cellules non déformées, ce sont donc des cellules de type “brique” c’est-à-dire avec les faces opposées parallèles deux à deux et les faces adjacentes orthogonales. Posons un repère principal R dont l’origine est le “premier” des sommets de la cellule (noté P_1) et les vecteurs formant la base sont les 3 arêtes de la cellule incidentes à ce sommet (notés U , V et W) (cf illustration en figure 3.9).

Un point Q de l’espace peut donc être exprimé dans ce repère grâce aux 3 paramètres (notés α , β et γ) associés à chacun des vecteurs :

$$Q = P_1 + \alpha U + \beta V + \gamma W$$

où les paramètres α , β et γ sont tous trivialement compris dans l’intervalle $[0..1]$. De manière plus compacte nous pouvons écrire en coordonnées homogènes :

$$\hat{Q} \sim \begin{pmatrix} U & V & W & P_1 \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \\ \gamma \\ 1 \end{pmatrix} \sim R \omega$$

où la tilda ($\sim$) signifie l’égalité à un facteur d’échelle près (notation conventionnelle en coordonnées homogènes).

Maintenant si l’on place un repère à chacun des 8 sommets de la cellule, de la même façon que précédemment, en se fixant les vecteurs comme les arêtes incidentes aux sommets. Nous noterons U_i (resp. V_i et W_i) le vecteur associé au paramètre α (resp. β et γ) dans le repère R_i d’origine P_i . Il est facile de trouver les paramètres associés à chacun de ces repères en fonction des paramètres obtenus dans la repère principal, concernant le paramètre α , nous obtenons $\alpha_i = \alpha$ dans les repères R_i pour $i = 1, 3, 5, 7$ et $\alpha_i = 1 - \alpha$ dans les repères R_i pour $i = 2, 4, 6, 8$ (cf figure 3.10). Le raisonnement est le même pour les deux autres paramètres. Nous définissons donc les fonctions $g_{\alpha_i}(\alpha)$ (resp. g_{β_i} et g_{γ_i}) transformant les paramètres associés au repère principal dans le repère i .

Ainsi, nous pouvons écrire :

$$Q_i = P_i + \alpha_i U_i + \beta_i V_i + \gamma_i W_i = P_i + g_{\alpha_i}(\alpha) U_i + g_{\beta_i}(\beta) V_i + g_{\gamma_i}(\gamma) W_i. \quad (3.1)$$

Tous les Q_i représente le même point, c’est-à-dire que pour $\forall i \in 1..8$: $Q = Q_i$ lorsque la cellule reste non déformée.

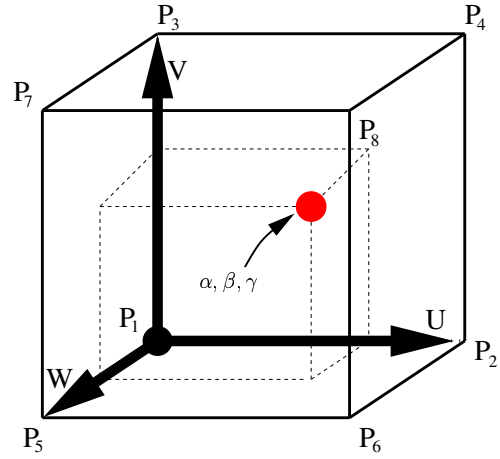
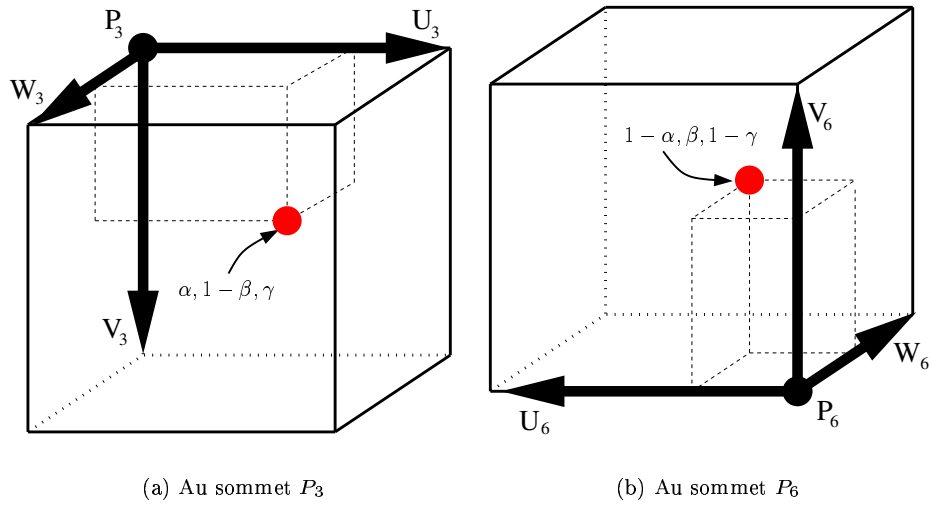


FIG. 3.9 – Repère associé au “premier des sommets de la cellule” (entourés)



(a) Au sommet P_3

(b) Au sommet P_6

FIG. 3.10 – Repères et paramètres en différents sommets

3.3.2 FFD - Interpolation de type Skinning

Maintenant, intéressons nous à l'application de la technique du skinning aux FFD basée sur la structure d'octree présentée précédemment. Nous présentons d'abord la méthode générale utilisée pour la déformation de l'objet à l'intérieur d'une seule cellule. Puis nous montrons comment adapter cette méthode pour pouvoir déformer un ensemble de cellules de même niveau (par exemple pour une subdivision uniforme de l'espace) de manière lisse. Finalement nous présentons notre principale contribution : la méthode générale permettant de déformer, toujours de manière lisse, un objet plongé dans un octree de hiérarchie quelconque.

Skinning à l'intérieur d'une cellule

Comme nous l'avons présenté dans l'état de l'art, la méthode de skinning consiste à calculer les coordonnées d'un point dans plusieurs repères, puis de combiner ces résultats pour obtenir un nouveau point. Nous proposons une manière de l'utiliser pour effectuer des déformations lisses du type FFD à l'intérieur d'une cellule hexaédrale.

Nous utilisons l'équation 3.1 qui définit la position Q_i d'un point Q dans le repère R_i (présenté dans la partie 3.3.1) en fonction des paramètres associés au point α_i , β_i et γ_i . Pour appliquer la méthode de skinning, il faut associer à chacun des points Q_i un poids k_i tel que (pour plus de clarté nous utiliserons dorénavant l'expression $\sum_i$ à la place de $\sum_{i=1}^8$) :

$$Q = \frac{\sum_i k_i Q_i}{\sum_i k_i}.$$

Ainsi, nous devons trouver ces poids tels qu'un point Q proche d'un sommet de la cellule P_i soit très influencé par ce sommet et peu par les autres. Nous imposons aussi la contrainte suivante : de la même manière que pour les interpolations de type Spline, un point de l'espace situé sur une face devra toujours rester sur cette face, quels que soient les déformations appliquées aux sommets de la cellule. En d'autres termes, l'influence des 4 sommets opposés à cette face doit être nulle sur un tel point.

Si, comme dans la plupart des méthodes de skinning, nous utilisons un poids k_i fonction de la distance du point Q au sommet P_i , il est facile de voir que nous ne pouvons pas respecter cette contrainte. Ainsi nous définissons les poids k_i associés à chacun des points Q_i (calculés dans les repères R_i) :

$$k_i = f(\alpha_i)f(\beta_i)f(\gamma_i) = f(g_{\alpha_i}(\alpha))f(g_{\beta_i}(\beta))f(g_{\gamma_i}(\gamma)) \quad (3.2)$$

où $\alpha_i = g_{\alpha_i}(\alpha)$ (similairement pour β et γ) et f est une fonction monotone décroissante telle que : $f(0) = 1$ et $f(1) = 0$. La première fonction qui vient à l'esprit est la fonction utilisée pour les interpolations linéaires : $f(t) = 1 - t$.

Il est facile de vérifier qu'un point Q dont les paramètres sont $\alpha = \beta = \gamma = 0$ ne sera influencé que par le sommet définissant le repère principal : P_1 (en fait, Q et P_1 sont ainsi confondus). De la même manière, tous les points définis par $\alpha = \beta = 1$ seront toujours situés sur l'arête P_4P_8 . Il en va de même pour les points situés sur les faces.

Remarquons que $f(1 - t) = 1 - f(t)$, ainsi il est facilement vérifiable par factorisation que $\sum_i k_i = \sum_i f(\alpha_i)f(\beta_i)f(\gamma_i) = 1$ (laissé au lecteur). L'équation définissant la position d'un point à l'intérieur d'une cellule en fonction des paramètres α , β et γ peut donc s'écrire plus simplement :

$$Q = \sum_i k_i Q_i.$$

Notons aussi que comme la fonction $g_{\alpha_i}(\alpha)$ est égale soit à α soit à $1 - \alpha$ (selon les i) (idem pour les fonctions g_{β_i} et g_{γ_i}), nous pouvons alors simplifier l'expression $f(g_{\alpha_i}(\alpha))f(g_{\beta_i}(\beta))f(g_{\gamma_i}(\gamma))$ par $f_{\alpha_i}(\alpha)f_{\beta_i}(\beta)f_{\gamma_i}(\gamma)$. Par exemple :

$$k_7 = f_{\alpha_7}(\alpha)f_{\beta_7}(\beta)f_{\gamma_7}(\gamma) = f(\alpha)f(1 - \beta)f(1 - \gamma).$$

Maintenant, l'on peut se demander si la déformation induite par le déplacement des sommets se fait de manière lisse. Pour le vérifier, nous définissons la fonction $Q(\alpha, \beta, \gamma)$:

$$Q(\alpha, \beta, \gamma) = \sum_i k_i(\alpha, \beta, \gamma) Q_i(\alpha, \beta, \gamma). \quad (3.3)$$

Comme cette fonction est le produit de polynômes, alors toutes ses dérivées sont continues, elle est donc trivialement C^∞ à l'intérieur d'une cellule.

Skinning adapté à un découpage uniforme

Maintenant que nous avons vu comment déformer de manière lisse un maillage contenu dans une cellule grâce à la technique de skinning et à une paramétrisation simple, intéressons-nous au problème du raccord entre deux cellules voisines. La déformation d'un sommet commun à deux cellules doit permettre d'assurer la continuité des points du maillage répartis dans chacune des cellules. Nous présentons comment adapter la technique de déformation par Skinning présentée précédemment pour atteindre ce but.

Pour plus de simplicité, nous nous plaçons dans le cas 2D, c'est-à-dire, lorsque nous avons fixé un des paramètres, par exemple ici, nous posons $\gamma = 0$. Nous étudions donc la fonction de mélange (3.3) au raccord entre deux cellules, par exemple le long de l'arête commune aux deux cellules présentées dans la figure 3.11 : l'arête définie par les sommets P_2P_4 dans la cellule C et $\tilde{P}_1\tilde{P}_3$ dans la cellule $\tilde{C}$. Les points sur cette arête ont uniquement le paramètre β de libre, nous les notons Q_β dans la cellule C et $\tilde{Q}_\beta$ dans la cellule $\tilde{C}$. Comme nous l'avons montré précédemment, seulement les sommets P_2 et P_4 auront un poids différent de zéro dans l'évaluation de la fonction de mélange pour la cellule C (resp. $\tilde{P}_1$ et $\tilde{P}_3$ pour la cellule $\tilde{C}$). Dans le cas présent, nous pouvons donc simplifier la fonction (3.3) :

$$\begin{aligned} Q_\beta(\alpha, \beta) &= k_2 Q_2 + k_4 Q_4 \\ \tilde{Q}_\beta(\tilde{\alpha}, \tilde{\beta}) &= \tilde{k}_1 \tilde{Q}_1 + \tilde{k}_3 \tilde{Q}_3 \end{aligned}$$

Il est facile de vérifier que $k_2 = \tilde{k}_1 = f(\beta)$ et $k_4 = \tilde{k}_3 = f(1 - \beta)$ ainsi que $Q_2 = \tilde{Q}_1$ et $Q_4 = \tilde{Q}_3$. Nous obtenons donc bien une continuité de position car le long de l'arête, $\forall \beta$ nous avons : $Q_\beta(\alpha, \beta) = \tilde{Q}_\beta(\tilde{\alpha}, \tilde{\beta})$. Le même raisonnement est extensible pour les faces communes à deux cellules, nous obtenons donc bien une continuité C^0 au raccord entre cellules.

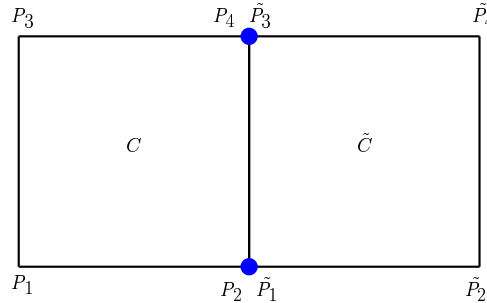


FIG. 3.11 – Raccord entre deux cellules de même profondeur

Quand est-il pour la continuité C^1 ? Intéressons nous aux dérivées premières de la fonction $Q(\alpha, \beta)$. La dérivée partielle par rapport au paramètre α est :

$$\begin{aligned} \frac{\partial Q(\alpha, \beta)}{\partial \alpha} &= \sum_i \left(\frac{\partial k_i(\alpha, \beta)}{\partial \alpha} Q_i(\alpha, \beta) + k_i(\alpha, \beta) \frac{\partial Q_i(\alpha, \beta)}{\partial \alpha} \right) \\ &= \sum_i (f'_{\alpha_i}(\alpha) f_{\beta_i}(\beta) Q_i(\alpha, \beta) + k_i(\alpha, \beta) g'_{\alpha_i}(\alpha) U_i) \\ &= \sum_i A_i(\alpha, \beta) + B_i(\alpha, \beta). \end{aligned}$$

De la même manière que pour la continuité de position, nous étudions le problème le long de l'arête P_2P_4 utilisée précédemment. Le raisonnement est bien sûr extensible au problème des faces en 3D.

Pour savoir si il y a une continuité C^1 au niveau de cette arête commune aux deux cellules, il convient d'étudier les dérivées de la fonction par rapport à α dans chacune des deux cellules C (où $\alpha = 1$) et $\tilde{C}$ (où $\alpha = 0$). Il est nécessaire que :

$$\begin{aligned} \frac{\partial Q(\alpha, \beta)}{\partial \alpha} &= \frac{\partial \tilde{Q}(\tilde{\alpha}, \tilde{\beta})}{\partial \tilde{\alpha}} \\ \sum_i A_i(\alpha, \beta) + B_i(\alpha, \beta) &= \sum_i \tilde{A}_i(\tilde{\alpha}, \tilde{\beta}) + \tilde{B}_i(\tilde{\alpha}, \tilde{\beta}) \end{aligned}$$

Si l'on essaie de travailler par identification, pour $\beta = \tilde{\beta}$:

- Dans la cellule C (resp. $\tilde{C}$), les termes B_1 et B_3 (resp. $\tilde{B}_2$ et $\tilde{B}_4$) seront nuls. Nous obtenons donc :

$$\sum_i B_i(\alpha, \beta) = B_2(\alpha, \beta) + B_4(\alpha, \beta) = -f(\beta)U_2 - f(1 - \beta)U_4 \quad \text{car } g'_{\alpha_2}(\alpha) = g'_{\alpha_4}(\alpha) = -1$$

et similairement

$$\sum_i \tilde{B}_i(\tilde{\alpha}, \beta) = B_1(\tilde{\alpha}, \beta) + B_3(\tilde{\alpha}, \beta) = f(\beta)\tilde{U}_1 + f(1 - \beta)\tilde{U}_3 \quad \text{car } g'_{\alpha_1}(\alpha) = g'_{\alpha_3}(\alpha) = 1.$$

Il est possible d'identifier ces deux expressions si et seulement si $\tilde{U}_1 = -U_2$ et $\tilde{U}_3 = -U_4$.

- Les termes A_i et $\tilde{A}_i$ nous gênent aussi car comme $f'_{\alpha_i}(\alpha) = \pm 1$, nous ne pouvant annuler $\sum_i A_i - \sum_i \tilde{A}_i$ car ils font intervenir des points Q_i et $\tilde{Q}_i$ qui ne sont pas communs aux deux cellules (le détail n'est pas donné ici mais est facilement vérifiable).

La dérivée des fonctions dans les deux cellules est donc différente ceci pose donc un problème de discontinuité indésirable que l'on peut remarquer sur la figure 3.12.

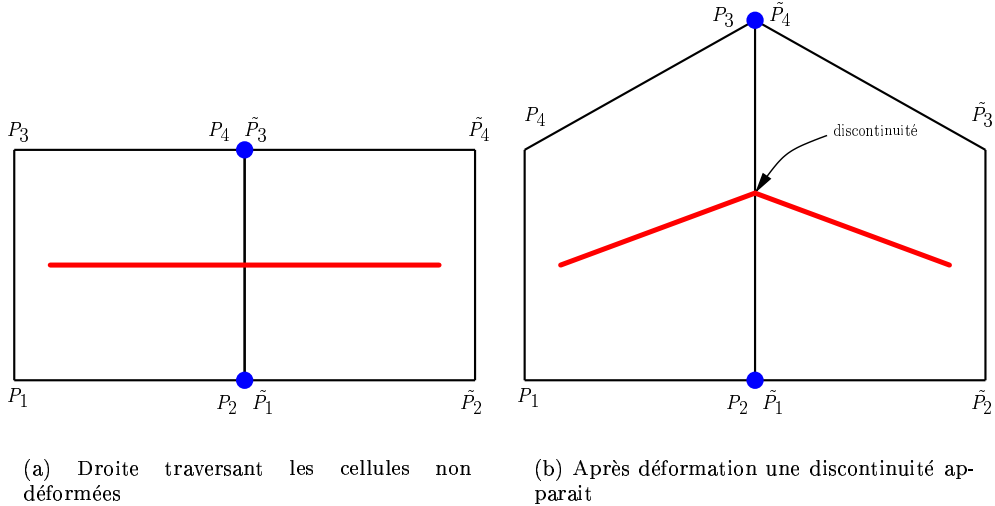


FIG. 3.12 – Discontinuité au raccord due à la fonction de poids linéaire utilisée

Pour remédier à ce problème de discontinuité de dérivée au raccord entre deux cellules, nous proposons la solution suivante : nous allons d'abord annuler tous les termes A_i et $\tilde{A}_i$. Nous souhaitons trouver une fonction, de plus bas degré possible, $f(t)$ monotone et strictement décroissante telle que $f'(0) = f'(1) = 0$, tout en conservant le fait que $f(0) = 1$ et $f(1) = 0$. Pour ce faire, nous utilisons la fonction dite d'*Hermite* du troisième ordre :

$$f(t) = 1 - 3t^2 + 2t^3.$$

Une autre fonction, paire celle-ci possédant les mêmes propriétés est la fonction $f(t) = (t^2 - 1)^2$. Grâce à l'annulation des A_i et $\tilde{A}_i$, nous pouvons maintenant simplifier l'expression des dérivées en (rappelons que $\beta = \tilde{\beta}$) :

$$\begin{aligned}\frac{\partial Q(\alpha, \beta)}{\partial \alpha} &= \frac{\partial \tilde{Q}(\tilde{\alpha}, \beta)}{\partial \tilde{\alpha}} \\ \sum_i B_i(\alpha, \beta) &= \sum_i \tilde{B}_i(\tilde{\alpha}, \beta) \\ -f(\beta)U_2 - f(1 - \beta)U_4 &= f(\beta)\tilde{U}_1 + f(1 - \beta)\tilde{U}_3\end{aligned}$$

Pour obtenir l'égalité, il suffirait que $\tilde{U}_1 = -U_2$ et $\tilde{U}_3 = -U_4$. Par la définition de ces vecteurs (qui sont en fait les arêtes des cellules), l'égalité est vraie lorsque les cellules sont à l'état initial mais devient fausse lorsqu'elles sont déformées. L'idée que nous proposons pour résoudre ce problème est, pour un sommet S partagé par plusieurs cellules, d'utiliser les mêmes vecteurs, donc les mêmes repères (au signe près). Ainsi, dans notre exemple, nous choisissons pour le sommet $P_2 = \tilde{P}_1$ le vecteur $\tilde{U}_2 = \tilde{U}_1 = \frac{U_2 - \tilde{U}_1}{2}$ pour la direction α (nous appliquons le même raisonnement pour les deux autres vecteurs). Nous obtenons ainsi :

$$\frac{\partial Q(\alpha, \beta)}{\partial \alpha} = \frac{\partial \tilde{Q}(\tilde{\alpha}, \beta)}{\partial \tilde{\alpha}} = f(\beta)\tilde{U}_2 + f(1 - \beta)\tilde{U}_4$$

Intuitivement, si l'objet à déformer est une droite définie par les points de β constant, il est possible de visualiser la tangente de la droite déformée au raccord entre deux cellules : c'est l'interpolation par la fonction f des tangentes qui sont les vecteurs $\tilde{U}_2$ et $\tilde{U}_4$. En effet, si $\beta = 0$ alors la tangente de la droite déformée sera en $\alpha = 1 : \tilde{U}_2$, si $\beta = 1$, elle sera alors $\tilde{U}_4$. Ainsi, pour un quelconque β elle sera : $f(\beta)\tilde{U}_2 + f(1 - \beta)\tilde{U}_4$. Deux illustrations sont présentées en figure 3.13.

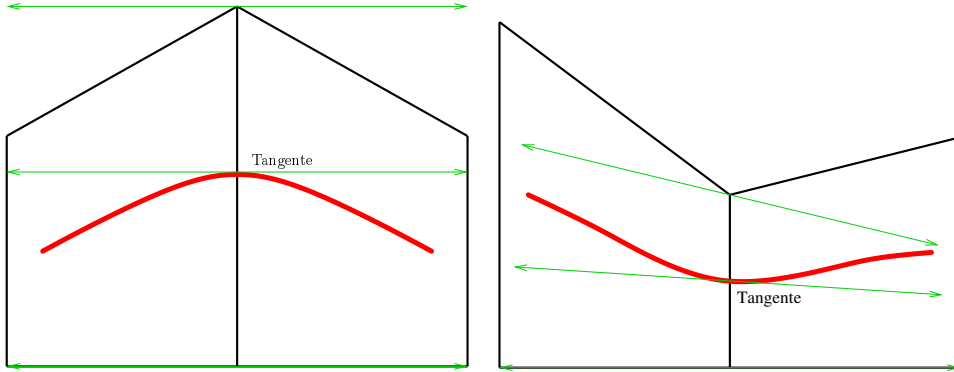


FIG. 3.13 – Continuité C^1 , illustrations des tangentes

Cette nouvelle définition de repères uniques associés aux sommets de l'octree possède un très gros avantage : il serait très simple pour un modelleur utilisant notre technique de FFD de manipuler l'aspect de l'objet par manipulation des sommets et aussi de pouvoir travailler directement sur les tangentes en manipulant les vecteurs des repères (cf figure 3.14).

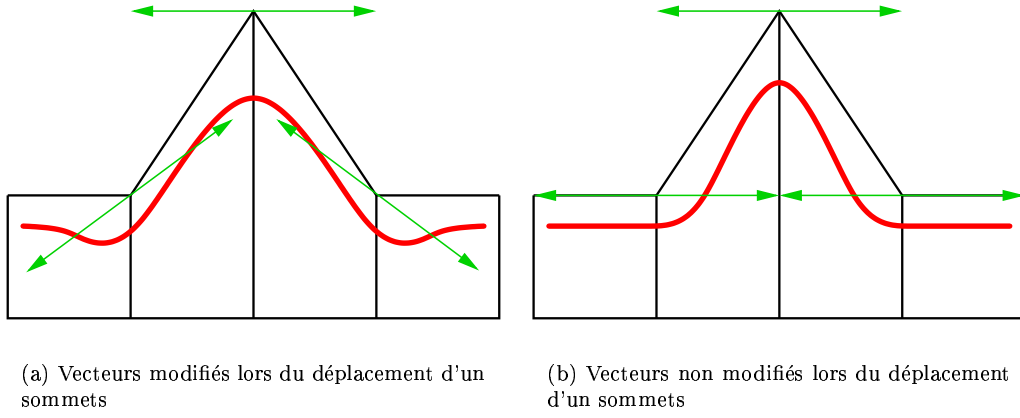


FIG. 3.14 – Illustration de l'influence des vecteurs formant les repères

Quand est-il pour la continuité C^2 ? Étudions tout d'abord les dérivées croisées par rapport à α et β :

$$\frac{\partial^2 Q(\alpha, \beta)}{\partial \alpha \partial \beta} = \frac{\partial^2 Q(\alpha, \beta)}{\partial \beta \partial \alpha}$$

$$\frac{\partial^2 Q(\alpha, \beta)}{\partial \alpha \partial \beta} = \sum_i (g'_{\alpha_i}(\alpha) g'_{\beta_i}(\beta) (f'(g_{\alpha_i}(\alpha)) f'(g_{\beta_i}(\beta)) + f'(g_{\alpha_i}(\alpha)) f(g_{\beta_i}(\beta)) V_i + f(g_{\alpha_i}(\alpha)) f'(g_{\beta_i}(\beta)) U_i)) .$$

Si l'on se place au raccord entre les deux cellules nous obtenons une continuité car dans la cellule C en $\alpha = 1$ nous obtenons :

$$\frac{\partial^2 Q(\alpha, \beta)}{\partial \alpha \partial \beta} = f'(\beta) U_1 - f'(1 - \beta) U_3$$

et dans la cellule $\tilde{C}$ en $\tilde{\alpha} = 0$ nous avons :

$$\frac{\partial^2 \tilde{Q}(\tilde{\alpha}, \tilde{\beta})}{\partial \tilde{\alpha} \partial \tilde{\beta}} = -f'(\beta) \tilde{U}_2 + f'(1 - \beta) \tilde{U}_4 = \frac{\partial^2 Q(\alpha, \beta)}{\partial \alpha \partial \beta}$$

comme $\beta = \tilde{\beta}$ et $\tilde{U}_2 = -U_1$ et $\tilde{U}_4 = -U_3$.

Étudions maintenant la dérivée seconde par rapport à α (la dérivée par rapport à β peut en être facilement déduite) :

$$\begin{aligned} \frac{\partial^2 Q(\alpha, \beta)}{\partial \alpha^2} &= \sum_i \left(\frac{\partial^2 k_i(\alpha, \beta)}{\partial \alpha^2} Q_i(\alpha, \beta) + 2 \frac{\partial k_i(\alpha, \beta)}{\partial \alpha} \frac{\partial Q_i(\alpha, \beta)}{\partial \alpha} + \frac{\partial^2 Q_i(\alpha, \beta)}{\partial \alpha^2} \right) \\ &= \sum_i f(g_{\beta_i}(\beta)) (f''(g_{\alpha_i}(\alpha)) Q_i(\alpha, \beta) + f'(g_{\alpha_i}(\alpha)) U_i) . \end{aligned}$$

En développant cette équation des deux côtés du raccord entre les cellules, on s'aperçoit qu'elles font intervenir des termes en fonction des 8 valeurs de i et en conséquence sont différentes. Par contre, il est possible d'obtenir un raccord C^2 de la déformation si l'on impose la contrainte $f''(1) = 0$:

$$\begin{aligned} \frac{\partial^2 Q(1, \beta)}{\partial \alpha^2} &= f''(0) (f(\beta) Q_2(0, \beta) + f(1 - \beta) Q_4(0, 1 - \beta)) \\ \frac{\partial^2 \tilde{Q}(0, \tilde{\beta})}{\partial \tilde{\alpha}^2} &= f''(0) (f(\beta) Q_1(0, \beta) + f(1 - \beta) Q_3(0, 1 - \beta)) \\ \frac{\partial^2 Q(1, \beta)}{\partial \alpha^2} &= \frac{\partial^2 \tilde{Q}(0, \tilde{\beta})}{\partial \tilde{\alpha}^2} \end{aligned}$$

Malheureusement, dans ce cas-là, nous observons un aplatissement de la déformation le long des faces dû au fait que la dérivée seconde ne soit fonction que des sommets de la face (cf figure 3.15). En pratique, nous avons observé qu'une fonction d'Hermite assurant la simple continuité C^1 de la déformation suffit amplement et permet d'obtenir des résultats convaincants.

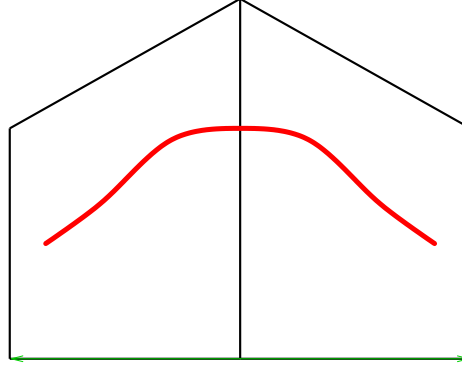


FIG. 3.15 – Problème d'aplatissement de la courbe avec un raccord C^2

FFD par Skinning sur un octree de topologie quelconque

Maintenant que nous avons présenté comment assurer un raccord lisse entre deux cellules de même niveau, nous allons nous intéresser au problème du raccord entre deux cellules de niveau différent (illustration en figure 3.16(a)).

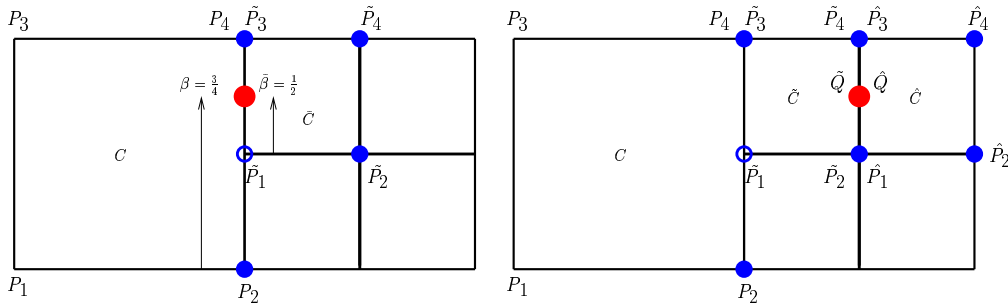


FIG. 3.16 – Raccord entre deux cellules de profondeur différente

Nous montrons d'abord pourquoi l'application de la méthode précédente ne permet pas d'assurer la continuité de position entre deux telles cellules. Pour ce faire, nous détaillons le calcul d'un point situé sur une sous-partie de l'arête P_2P_4 (sous-partie définie par l'arête $\tilde{P}_1\tilde{P}_3$) : dans l'exemple proposé, les paramètres du point Q dans la cellule C sont : $\alpha = 1$ et $\beta = 3/4$, et dans la cellule $\tilde{C}$ de plus bas niveau : $\tilde{\alpha} = 0$ et $\tilde{\beta} = 1/2$.

Grâce à la fonction (3.3), il est possible de calculer la position du point Q dans les deux cellules par ses paramètres ainsi que par les repères proposé plus haut :

$$\begin{aligned} Q(\alpha, \beta) &= k_2(\alpha, \beta)Q_2(\alpha, \beta) + k_4(\alpha, \beta)Q_4(\alpha, \beta) \\ \tilde{Q}(\tilde{\alpha}, \tilde{\beta}) &= \tilde{k}_1(\tilde{\alpha}, \tilde{\beta})\tilde{Q}_1(\tilde{\alpha}, \tilde{\beta}) + \tilde{k}_3(\tilde{\alpha}, \tilde{\beta})\tilde{Q}_3(\tilde{\alpha}, \tilde{\beta}). \end{aligned}$$

En développant et en supprimant les termes égaux à 1 ($f(0) = 1$) :

$$\begin{aligned} Q(\alpha, \beta) &= f(\beta)(P_2 + \beta\bar{V}_2) + f(1 - \beta)(P_4 + (1 - \beta)\bar{V}_4) \\ \tilde{Q}(\tilde{\alpha}, \tilde{\beta}) &= f(\tilde{\beta})(\tilde{P}_1 + \tilde{\beta}\tilde{V}_1) + f(1 - \tilde{\beta})(\tilde{P}_3 + (1 - \tilde{\beta})\tilde{V}_3). \end{aligned}$$

Trois problèmes se posent :

- Les sommets utilisés ne sont pas les mêmes. Mais par définition de la subdivision de l'octree, il est facile de remarquer que : $\tilde{P}_3 = P_4$ et comme le point $\tilde{P}_1$ est contraint : $\tilde{P}_1 = \frac{P_2 + P_4}{2}$.
- Les vecteurs utilisés ne sont pas les mêmes. On peut tout de même remarquer que $\tilde{V}_3 = \frac{\tilde{V}_4}{2}$ et $\tilde{V}_1 = \frac{\tilde{V}_2}{2} = -\tilde{V}_3$.
- Il est facile de noter, toujours à cause de la contrainte sur $\tilde{P}_1$ que $\beta = \frac{\tilde{\beta} + 1}{2} : \tilde{\beta} = 2\beta - 1$. Les poids appliqués à chacun des points ne permettent pas, malgré les remarques précédentes, d'égaliser les deux fonctions, en effet :

$$\begin{aligned}\tilde{Q}(\tilde{\alpha}, \tilde{\beta}) &= f(2\beta - 1)\left(\frac{P_2 + P_4}{2} + 2\beta - 1\frac{\tilde{V}_2}{2}\right) + f(2(1 - \beta))(P_4 + 2(1 - \beta)\frac{\tilde{V}_4}{2}) \\ &\neq Q(\alpha, \beta)\end{aligned}$$

(le détail n'est pas présenté ici).

Le principal problème est la non-cohérence des poids. En effet, si l'on sait exprimer dans les deux cellules les points situés sur l'arête uniquement en fonction des repères des sommets P_2 et P_4 , il faut encore appliquer exactement les mêmes poids aux positions du point calculés dans chacun de ces repères. En d'autres termes, il faut que les poids associés aux repères de P_2 et P_4 soient strictement les mêmes dans les cellules C et $\tilde{C}$. Pour ce faire, il suffit de connaître les paramètres du point $\tilde{Q}$ (situé dans la cellule $\tilde{C}$) dans sa cellule mère adjacente à la cellule C .

Dans le cas de cellules initialement non déformées, cette opération est très simple. Etant donné les paramètres α_n, β_n d'un point dans une cellule de niveau n , après subdivision, ce point se situe dans une des cellules filles de niveau $n + 1$: si $\alpha_n \leq 1/2$ alors il se situera dans une des cellules de gauche et ainsi : $\alpha_{n+1} = 2\alpha_n$, sinon il se situera dans une des cellules de droite et $\alpha_{n+1} = 2\alpha_n - 1$ (le même raisonnement est appliqué pour les autres paramètres). Par récursion, il est ainsi aisé de trouver les paramètres dans une cellule de niveau n d'un point paramétré dans une cellule de niveau $n + m$, et inversement.

Il est à noter qu'un point situé sur une arête (ou sur une face en 3D) sera placé dans une des cellules partageant cette arête, cela ne pose pas de problème puisque nous allons présenter comment il est possible d'assurer la continuité entre toutes les cellules.

Nous proposons donc une nouvelle méthode permettant de calculer la position d'un point dans une cellule tout en conservant la continuité de position et de dérivée au raccord entre des cellules de profondeur quelconque.

Notre approche se base sur le fait que le champ de déformation induit par la structure d'octree uniquement est entièrement et uniquement défini par des sommets *libres*. Les sommets *contraints* n'apportent pas d'informations supplémentaires car ils sont contraints par leurs parents.

Si par exemple un sommet est contraint sur une arête alors son influence et le repère associé seront entièrement liés à ceux de ses parents. Ainsi, en remontant l'arbre des dépendances de sommets, il est possible d'obtenir tous les sommets libres ascendants aux 8 sommets d'une cellule C , nous noterons cette liste de sommets $P_{libres}(C)$. Nous posons maintenant une nouvelle fonction permettant de connaître la position d'un point dans C uniquement en fonction de sommets libres :

$$Q(\alpha, \beta, \gamma) = \frac{\sum_j^{\text{nombre de } P_{libres}(C)} k_{influence}(P_{libres}(C)_j, \alpha, \beta, \gamma) Q(P_{libres}(C)_j, \alpha, \beta, \gamma)}{\sum_j^{\text{nombre de } P_{libres}(C)} k_{influence}(P_{libres}(C)_j, \alpha, \beta, \gamma)} \quad (3.4)$$

où α, β et γ sont les paramètres du point locaux à la cellule C , $P_{libres}(C)_j$ est le j ième sommet libre de la liste $P_{libres}(C)$, $k_{influence}$ donne le poids-influence associé au sommet $P_{libres}(C)_j$ en fonction des paramètres locaux, et Q la position du point calculé dans le repère de $P_{libres}(C)_j$ en fonction des paramètres locaux (rappelons que les repères sont ici tous orientés vers l'intérieur de la cellule C). Il faut bien sûr diviser par la somme des poids car cette dernière est maintenant différente de 1.

Ainsi, il est facile de vérifier que pour l'exemple présenté précédemment, le long de l'arête P_2P_4 , où P_2 et P_4 sont libres, et d'après l'équation (3.4), nous obtenons :

$$Q(\alpha, \beta, \gamma) = \tilde{Q}(\tilde{\alpha}, \tilde{\beta}, \tilde{\gamma}) = f(\beta)(P_2 + \beta\tilde{V}_2) + f(1 - \beta)(P_4 + (1 - \beta)\tilde{V}_4).$$

Il reste un problème : cette nouvelle méthode ne permet pas d'annuler l'influence d'un sommet libre aux autres sommets libres. Par exemple, si les sommets $\tilde{P}_2$ et $\tilde{P}_4$ sont libres, alors un point $\tilde{Q}$ situé sur cette arête ne doit en aucun cas subir l'influence d'autres sommets (cf figure 3.16(b)). En effet, dans la cellule voisine notée $\hat{C}$, comprenant cette arête, le point $\hat{Q}$ équivalent à $\tilde{Q}$ est uniquement fonction des points $\hat{P}_1$ et $\hat{P}_3$ définissant cette arête. Or, dans la cellule $\tilde{C}$, l'influence des sommets P_2 et P_4 n'y est pas nulle, il est donc nécessaire de trouver un moyen de l'annuler à cet endroit, tout en conservant les propriétés permettant d'obtenir un raccord lisse des cellules.

Nous définissons maintenant la notion de volume d'influence (ou de pondération) : Tout sommet libre P d'une cellule C ne possédant que des sommets libres a une influence (ou une pondération) sur tout l'espace des cellules du sous-arbre des cellules partant de C . L'influence de P est propagée à tous les champs de déformation définis par les cellules descendantes de C . Elle définit donc un volume d'influence qui est fonction de la topologie des cellules.

Nous illustrons la propagation de l'influence d'un sommet dans la figure 3.17. On peut ainsi se rendre compte qu'à chaque sommet est associé un doublet (un triplet en 3D) de paramètres permettant de connaître la valeur du poids en ce sommet, ces paramètres correspondent aux paramètres α et β (et γ) dans la cellule. Ces paramètres sont liés à la position du sommet dans la cellule ainsi qu'à ses liens de parenté.

Les lois régissant la propagation de l'influence sont les suivantes :

1. Le sommet libre P générateur possède le couple de paramètres $(0, 0)$ car l'influence en cet endroit est totale ($f(0)f(0) = 1$).
2. Les 3 autres sommets de la cellule possèdent les paramètres suivants : $(0, 1)$ pour le sommet connecté à P en dans la "direction" α , $(1, 0)$ pour celui dans la direction β et $(1, 1)$ pour celui à l'opposé.
3. Tout autre sommet libre ayant 4 parents (sommets de face) possède les paramètres $(1, 1)$, ainsi, l'influence du sommet P y sera de $f(1)f(1) = 0$. Les sommets libres ayant 2 parents (sommets d'arête) possèdent 1 pour le paramètre associé à la direction (α , β ou γ) de l'arête, et la moyenne des paramètres de ses deux parents pour les deux autres paramètres.
4. Un sommet contraint sur une arête par deux points aura pour paramètres la moyenne des paramètres deux.
5. Un sommet contraint sur une face par quatre sommets aura pour paramètres la moyenne des paramètres des quatres.

De manière récursive il est alors possible d'associer à chaque sommet issu de la cellule C un couple de paramètres.

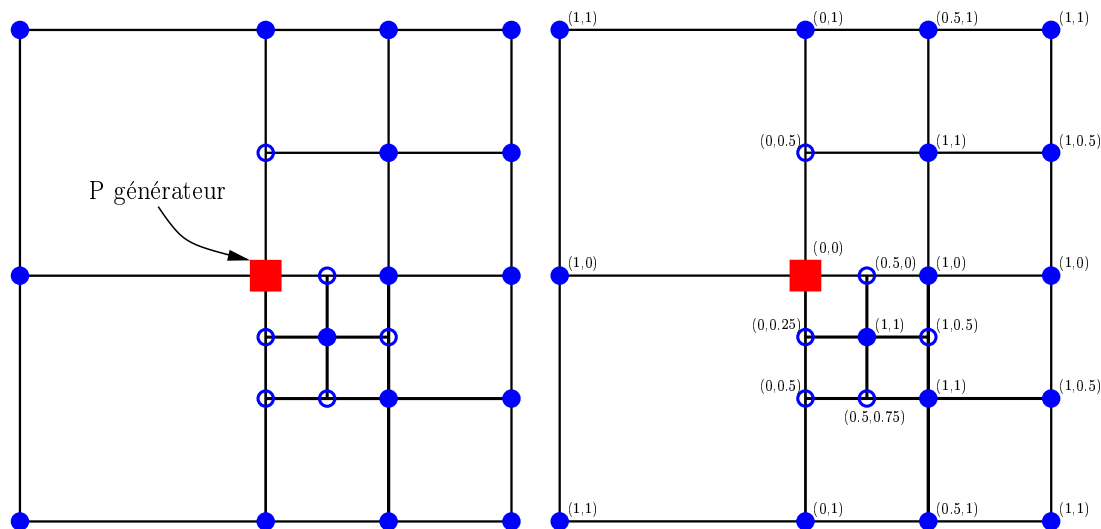


FIG. 3.17 – Propagation de l'influence du sommet central

Pour connaître l'influence du sommet P dans une cellule $\tilde{C}$ (descendante de C) en un point Q (de paramètres locaux $\tilde{\alpha}$ et $\tilde{\beta}$ (et $\tilde{\gamma}$ en 3D)), il suffit d'interpoler trilinéairement les paramètres associés aux 4 sommets de la cellule $\tilde{C}$ (8 en 3D), en fonction des $\tilde{\alpha}$ et $\tilde{\beta}$ (et $\tilde{\gamma}$ en 3D) de Q .

Soient $(\tilde{x}_i, \tilde{y}_i, \tilde{z}_i)$ les paramètres d'influence de P associés au sommet $\tilde{P}_i$ de la cellule $\tilde{C}$, alors l'influence à associer à P dans cette cellule est :

$$k_{influence}(P, \alpha, \beta, \gamma) = f\left(\sum_i w_i x_i\right) f\left(\sum_i w_i y_i\right) f\left(\sum_i w_i z_i\right) \quad (3.5)$$

où $w_i = g_{\alpha_i}(\alpha)g_{\beta_i}(\beta)g_{\gamma_i}(\gamma)$ permet d'interpoler les différentes influences du point P en chacun des sommets en fonction de la position du point Q dans la cellule (définie par α, β et γ). Il est à noter que la somme des w_i est égale à 1. Ainsi, $k_{influence}(P, \alpha, \beta, \gamma)$ représente bien la valeur du champ d'influence du sommet P au point Q .

Dans le cas où tous les sommets de la cellule sont libres (alors la liste $P_{libres}(C)$ est la liste des sommets de la cellule), il est facile de vérifier que nous retrouvons bien la même fonction $Q(\alpha, \beta, \gamma)$ présentée dans l'équation (3.3).

Précisions concernant les vecteurs associés aux repères

Il est nécessaire de clarifier quelques points, que nous avons brièvement introduit plus haut, concernant le choix des vecteurs des repères associés aux sommets libres de l'octree.

En effet, nous avons expliqué que le vecteur $\bar{U}$ associé à la direction α (et similairement pour β et γ), est défini comme : $\bar{U} = \frac{U_1 - U_2}{2}$ où U_1 et U_2 sont les vecteurs définis par les arêtes de direction α incidentes au sommet dans les cellules C_1 et C_2 (cf figure 3.18).

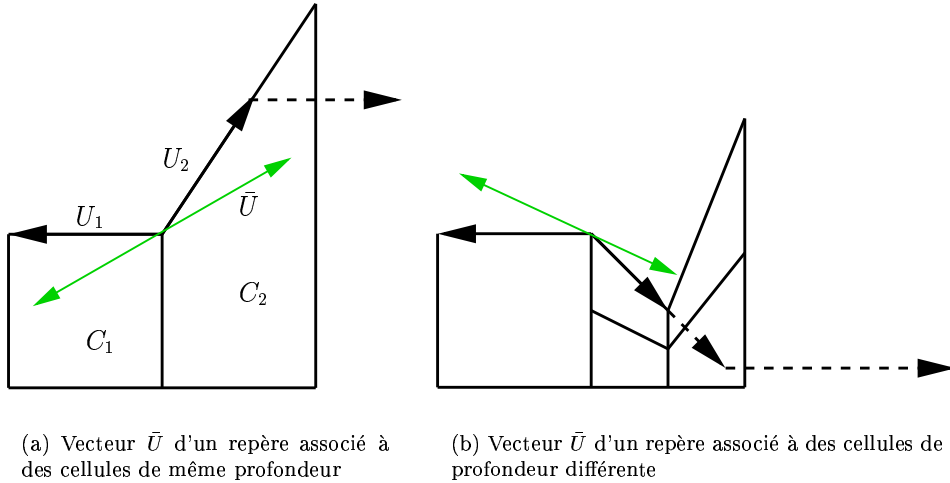


FIG. 3.18 – Calcul du vecteur “moyen”

Le lecteur peut poser deux remarques :

- Le fait de déformer la cellule C_2 tend à agrandir la norme du vecteur $\bar{U}$ et en conséquence risque d'apporter des déformations non désirées par l'utilisateur dans la cellule C_1 . Pour palier ce problème, nous proposons de conserver une norme constante pour les vecteurs U_1 et U_2 , cette norme est celle des vecteurs à l'état initial non déformé.
- Si la cellule C_2 est subdivisée et les cellules filles déformées, le vecteur $\bar{U}$ doit tenir compte de ces déformations appliquées aux niveaux plus profonds. Pour résoudre ce problème, nous proposons d'effectuer la “moyenne” des vecteurs associés aux arêtes de plus bas niveau en mettant les deux vecteurs à la même norme.

D'après ces remarques, nous pouvons définir la fonction de calcul du vecteur $\bar{U}$ dans une cellule C :

$$\bar{U}_C = \frac{U_1/\|U_1\| - U_2/\|U_2\|}{2\|U_C\|} \quad (3.6)$$

où $\|U_C\|$ est la norme initiale du vecteur de direction α dans la cellule C .

3.4 Déformation multi-résolutions adaptative

Nous avons présenté précédemment une technique de déformation lisse d'un objet placé dans un octree dont la topologie ne change pas. Il n'est pas possible d'utiliser la méthode que nous avons décrite si l'on veut effectuer une déformation adaptative (c'est-à-dire une dynamique de la structure d'octree), conserver les modifications apportées lors des opérations de simplification ou subdivision.

En effet, si après la déformation de l'octree donc de l'objet, on souhaite subdiviser ou simplifier une cellule, l'utilisation telle quelle de la méthode introduira un effet de "saut" (*popping*) des points du maillage de l'objet situés à l'intérieur de la cellule dont la topologie a été modifiée.

Ce problème est illustré dans la figure 3.19, ceci peut facilement se comprendre : le point Q a pour paramètres $\alpha = \beta = 1/2$ dans la cellule de gauche non subdivisée (C). Lors de la subdivision, il est placé dans la cellule fille en bas à gauche (C'), avec pour paramètres locaux $\alpha' = \beta' = 1$. Il est donc confondu avec le sommet (libre) partagé par toutes les cellules filles, ce qui n'était pas le cas avant la subdivision.

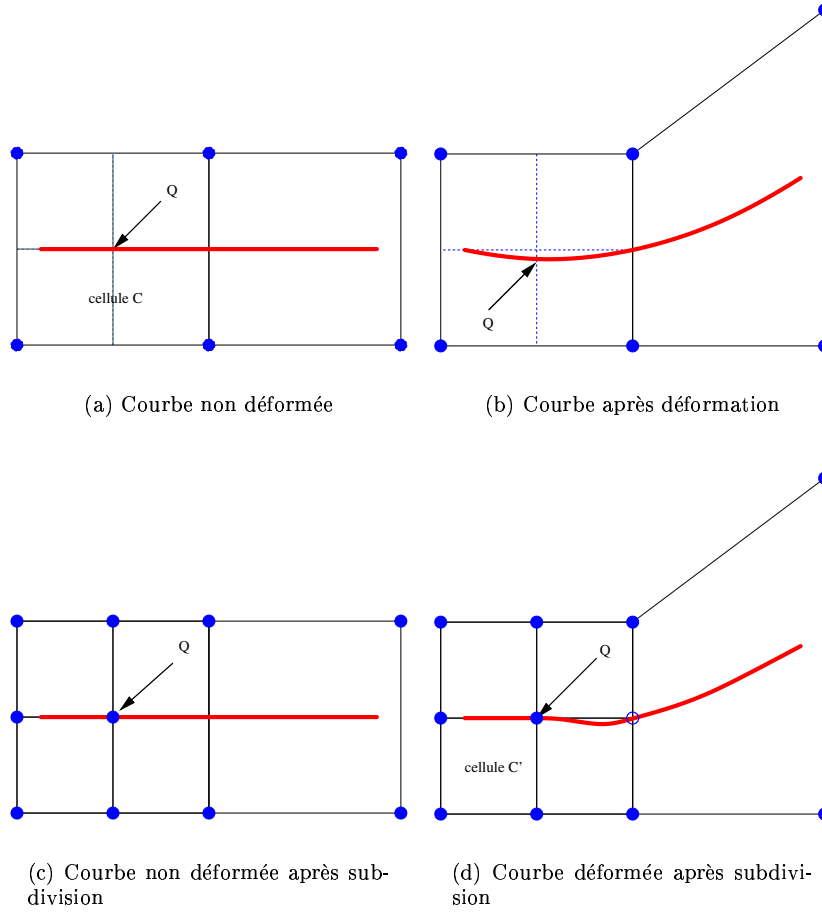


FIG. 3.19 – Effet de *popping* lors de la subdivision

Le problème dual apparaît lors de la simplification de cellules.

La solution à ce problème consiste à trouver les paramètres correspondants aux points situés à l'intérieur des cellules filles lors des opérations de subdivision et de simplification. Soit $\hat{Q}$ la

position dans l'espace d'un point du maillage, il convient de trouver α , β et γ tel que

$$Q(\alpha, \beta, \gamma) - \hat{Q} = 0$$

. Il est possible de trouver une approximation du triplet de paramètres par une méthode d'optimisation (de type Newton par exemple) en cherchant à minimiser :

$$\min_{\alpha, \beta, \gamma} \left(Q(\alpha, \beta, \gamma) - \hat{Q} \right)^T \left(Q(\alpha, \beta, \gamma) - \hat{Q} \right). \quad (3.7)$$

où T dénote la transposée du vecteur. Pour appliquer une méthode d'optimisation de type Newton, il convient de connaître le Jacobien de la fonction $Q(\alpha, \beta, \gamma) - \hat{Q}$ (en se basant sur la fonction Q de l'équation (3.4)), il est obtenu par le calcul des dérivées partielles par rapport aux paramètres. Les champs d'influence introduits précédemment peuvent entraîner l'apparition de minima locaux loins de la solution optimale. Pour éviter de "tomber" dans de tels minima il convient de fournir au processus d'optimisation une solution initiale la plus proche possible du minimum global. Lors de la subdivision il suffit, comme nous l'avons présenté plus tôt, de trouver la cellule parmi les 8 dans laquelle le point devrait se trouver et d'appliquer une fonction linéaire simple aux paramètres du point dans la cellule mère pour trouver une estimation des paramètres dans la cellule fille. Lors de la simplification, il suffit d'appliquer le processus inverse pour trouver les paramètres du point dans la cellule mère. Ainsi, l'effet de *popping* peut être résolu.

Deux problèmes principaux peuvent arriver lors d'opérations sur des cellules très déformées :

- La subdivision d'une cellule très déformée peut entraîner le fait qu'un point se trouve dans une autre cellule que celle intuitée par le choix selon les paramètres. Une idée consisterait à estimer les paramètres dans la première cellule intuitée et suivant les paramètres obtenus, rechercher ces paramètres dans une des cellules soeurs : par exemple si l'on se trouve dans la cellule fille à laquelle à l'état initial, nous aurions associé les points de paramètres (dans la cellule mère) dans l'intervalle $[0, 0.5]$ et que nous trouvons $\alpha > 1$, alors il faudra à nouveau estimer les paramètres dans la cellule voisine en α .
- Le problème dual : la simplification d'une cellule alors que l'on a effectué une grande déformation aux niveaux inférieurs, par exemple si l'on fait sortir les points du champ de déformation de la cellule mère. Il serait alors envisageable de chercher les paramètres dans les cellules voisines de la cellule mère ou alors en partant de la cellule principale, de parcourir la hiérarchie ; ceci n'est bien sûr possible que si le point se trouve à l'intérieur d'au moins une cellule et que les cellules de plus grande profondeur ne soit pas "trop" déformées.

Pour pouvoir effectuer ce raisonnement, il est nécessaire de travailler sur une fonction un peu plus complexe que la fonction d'Hermite présentée : la fonction doit aussi être paire et égale à 0 en dehors de l'intervalle $[-1, 1]$.

Il est alors possible d'effectuer deux types de déformations : des déformations **plastiques** où l'objet est déformé de la même manière qu'une pâte à modeler, ou des déformations **élastiques** où les déformations à des profondeurs plus élevées sont perdues à la simplification : on ne cherche pas à trouver les nouveaux paramètres dans la cellule mère, on utilise les précédents (avant la subdivision).

Le premier type sera le plus pratique pour un modelleur car ces déformations permettent d'éditer librement un objet à n'importe quelle résolution sans se soucier du niveau de résolutions des cellules voisines, parentes ou descendantes.

Le deuxième type sera le plus adapté à la déformation d'objets de type organique qui sont considérés comme très élastique. Par contre, lors d'une animation physique, il conviendra de simplifier une cellule uniquement si toutes ses descendantes sont en état de stabilité sous peine d'obtenir un effet de *popping* visuellement désagréable.

Nous n'avons malheureusement pas eu le temps d'implanter cette méthode et ainsi nous ne pouvons pas montrer de résultats pour l'instant.

Chapitre 4

Bilan et Conclusion

4.1 Bilan

Nous venons de présenter une méthode de FFD multi-résolutions adaptative permettant de faire de la modélisation et/ou de l’animation d’objets 3D quelconques. Il est ainsi possible d’effectuer des déformations locales à haute-résolution tout aussi bien qu’à un niveau plus bas pour appliquer des déformations plus globale de manière lisse tout en conservant un nombre de degrés de liberté (en nombre de sommets de contrôle) faible. Pour cela, nous avons introduit une structure de données complexe permettant de tenir compte des contraintes géométriques et topologiques de la structure d’octree sur laquelle nous basons notre travail. Cette dernière permet aussi de définir un système d’animation physique simple (directement lié à l’objet par les sommets libres de l’octree) et généré de manière automatique.

L’outil de déformation multi-résolutions adaptatif que nous venons de présenter semble comparable qualitativement aux techniques de FFD actuelles. Il permet de plus un contrôle très fin des déformations à des niveaux de profondeurs différentes simultanément, sans avoir à modifier la topologie de la structure ou à précalculer n niveaux de profondeur, puis de changer de résolution lorsqu’il est nécessaire.

De plus, le nombre de points de contrôle influant sur un point de l’espace est limité en fonction du nombre de sommets libres parents de la cellule et lors d’une subdivision uniforme est plus petit (égale à 8, le nombre de sommets libres de chaque cellule) que les techniques de FFD traditionnelles où il est nécessaire d’avoir des patches de $3 \times 3 \times 3 = 27$ points de contrôle pour assurer une continuité C^1 entre les patches.

Nous montrons dans les images de la figure 4.1 des exemples de déformation lisse d’une surface plane (à l’état non déformé) à des résolutions différentes avec ou non des contraintes topologiques (cellules voisines de profondeurs différentes).

Un inconvénient de notre méthode provient du fait même de la structure d’octree utilisée : le comportement des déformations sont entièrement dépendantes du découpage de l’espace par cellules cubiques. Pour être utilisé dans une phase de modélisation, l’infographiste doit être conscient de cela et doit veiller à positionner correctement l’objet dans la cellule englobante pour obtenir les résultats qu’il désire.

Ensuite, l’influence d’un sommet de contrôle est dans le cas général limitée aux cellules auxquelles il est directement connecté. Lorsque l’octree est très subdivisé cela peut introduire des déformations un peu “brutales” lors de l’édition multi-résolutions de ce dernier, ceci provient du fait que lorsqu’il y a une variation importante de dérivée au niveau de repères adjacents, ce changement n’est pris en compte que localement dans la cellule possédant ces deux repères. Ce problème n’arrive pas lors d’une simulation physique car les liens mécaniques entre les cellules tendent à se stabiliser et à éviter ce genre de déformation. Du point de vue du modélisateur, nous proposons deux solutions simples pour pallier ce problème : simplifier temporairement la cellule éditée, appliquer la déformation et resubdiviser la cellule ensuite, ou bien l’on pourrait déplacer les sommets descendants de la cellule déformée de la même manière que les points du maillage :

par FFD.

4.2 Perspectives

La méthode que nous présentons est facilement adaptable au cas 2D. Ceci pourrait permettre de déformer très simplement une surface dans l'espace, une application envisageable serait l'animation multi-résolutions efficace de vêtements qui pourrait suivre les mouvements d'un personnage tout en étant soumis à des lois physiques facilement modélisables par un type de structure similaire à celui que nous présentons.

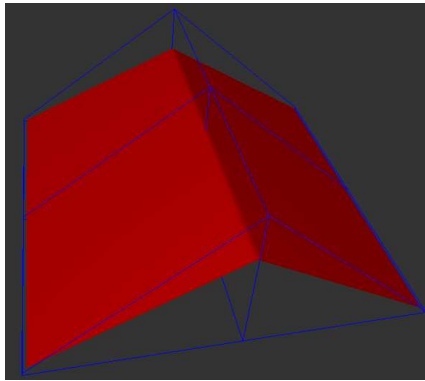
Nous envisageons aussi de pouvoir appliquer notre méthode à une structure hiérarchique de topologie plus libre, par exemple un réseau de tétraèdres, pour cela, il suffit de définir les lois sur les sommets contraints ainsi que les repères à associer aux sommets libres pour qu'ils permettent de conserver au moins une continuité C^1 entre de telles cellules.

Nous pensons aussi pouvoir créer une nouvelle méthode de simulation physique par éléments finis définissant le comportement d'une cellule comme étant le mélange de comportements simples (des éléments finis sur des parallélépipèdes à faces parallèles basés sur les repères de l'octree par exemple) faciles et rapides à calculer, mais qui nécessite toujours un modèle de gestion de collisions.

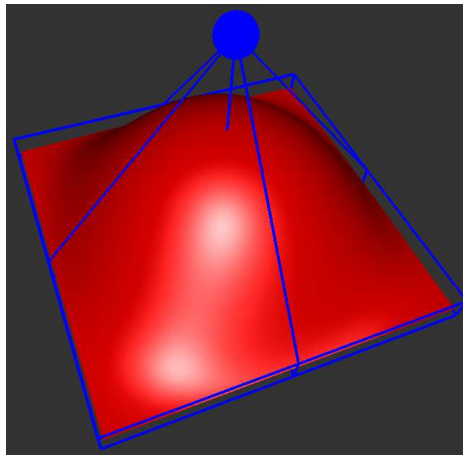
Les cellules feuilles de l'octree forment un sous-espace défini par l'ensemble des valeurs de la fonction $Q(\alpha, \beta, \gamma)$ (avec les paramètres compris dans l'intervalle $[0, 1]$). Ceci permet d'obtenir une boîte englobante du morceau d'objet dans cette cellule et ainsi de savoir rapidement si il est nécessaire d'effectuer des tests de collisions au niveau géométrique.

De nombreux articles expliquent comment raffiner un maillage soumis à une déformation de manière lisse, une solution pour notre problème consisterait pour raffiner un triangle, à moyenner les paramètres des 3 points formant le triangle et de trouver la position associée dans l'espace par simple évaluation de la fonction de déformation.

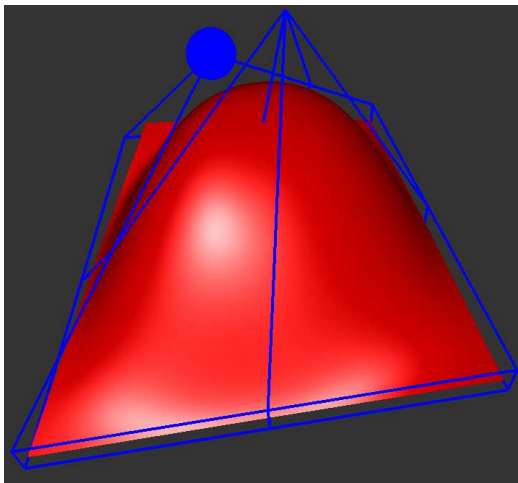
La technique de manipulation directe présentée dans [9] est aussi applicable à notre méthode, elle s'y ajouterait et proposerait un outil complet permettant la déformation lisse, interactive, multi-résolution adaptatif, physique et directe d'un objet 3D quelconque.



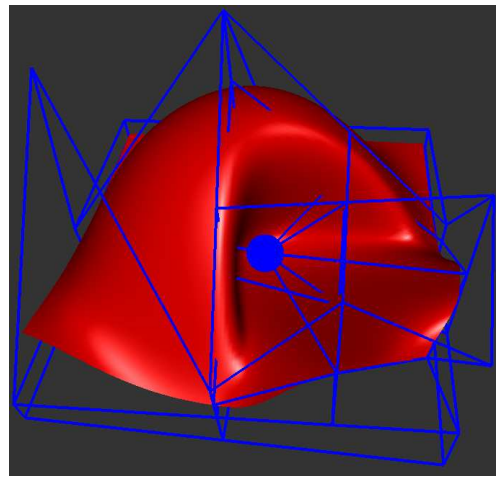
(a) Fonction d'interpolation linéaire : on voit apparaître la discontinuité de dérivée première



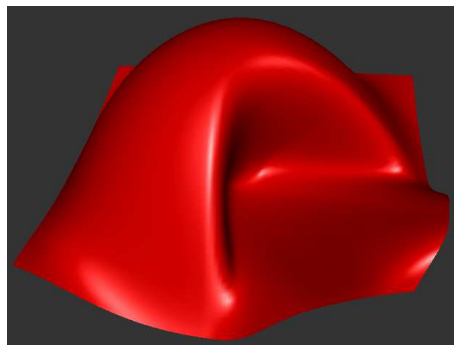
(b) Fonction d'interpolation d'Hermite, ici le point central à été élevé



(c) Édition multi-résolutions de l'octree : à partir de l'image précédente, nous avons déplacé un sommet de plus haut niveau



(d) Édition multi-résolutions : déformations de cellules voisines de profondeur différente



(e) La même image que précédemment, sans l'affichage de la structure d'octree

FIG. 4.1 – Différentes déformations d'un plan défini par une triangulation de 100×100 points

Bibliographie

- [1] Alan H. Barr. Global and local deformations of solid primitives. *IEEE Transactions on Visualization and Computer Graphics*, July 1984.
- [2] E. Catmull and J. Clark. Recursively generated b-spline surfaces on arbitrary topological meshes. *Computer Aided Design*, 10, 1978.
- [3] P. Cignoni, F. Ganovelli, and R. Scopigno. Introducing multiresolution representation in deformable modeling, 1999.
- [4] Sabine Coquillart. Extended free-form deformation : a sculpturing tool for 3d geometric modeling. In *Proceedings of the 17th annual conference on Computer graphics and interactive techniques*, pages 187–196. ACM Press, 1990.
- [5] Gilles Debunne, Mathieu Desbrun, Alan H. Barr, and Marie-Paule Cani. Interactive multi-resolution animation of deformable models. In *Eurographics Workshop on Computer Animation and Simulation*, Sep 1999.
- [6] P. Faloutsos, M. van de Panne, and D. Terzopoulos. Dynamic free-form deformations for animation synthesis. *IEEE Transactions on Visualization and Computer Graphics*, 3(3) :201–214, /1997.
- [7] David R. Forsey and Richard H. Bartels. Hierarchical b-spline refinement. In *Proceedings of the 15th annual conference on Computer graphics and interactive techniques*, pages 205–212. ACM Press, 1988.
- [8] Alex Mohr Michael Gleicher. Building efficient, accurate character skins from examples. *ACM Transactions on Graphics*, 10, July 2003.
- [9] William M. Hsu, John F. Hughes, and Henry Kaufman. Direct manipulation of free-form deformations. *Computer Graphics*, 26(2) :177–184, 1992.
- [10] Francis Lazarus, Sabine Coquillart, and Pierre Jancène. Axial deformations : an intuitive deformation technique. *Computer-Aided Design*, 26(8) :607–613, August 1994.
- [11] Ron MacCracken and Kenneth I. Joy. Free-form deformations with lattices of arbitrary topology. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 181–188. ACM Press, 1996.
- [12] Thomas W. Sederberg and Scott R. Parry. Free-form deformation of solid geometric models. In *Proceedings of the 13th annual conference on Computer graphics and interactive techniques*, pages 151–160. ACM Press, 1986.
- [13] Karan Singh and Eugene Fiume. Wires : a geometric deformation technique. 1998.
- [14] Tomoyuki Nishita Yutaka Ono, Bing-Yu Chen and Jieqing Feng. Free-form deformation with automatically generated multiresolution lattices. 2002.