
La modélisation de données océanographiques à l'aide du diagramme de Voronoï tridimensionnel

Hugo Ledoux — Christopher Gold

GIS Research Centre, School of Computing, University of Glamorgan
Pontypridd CF37 1DL, Wales, UK
hledoux@glam.ac.uk, christophergold@voronoi.com

RÉSUMÉ. L'intégration de données océanographiques dans les systèmes d'information géographique (SIG) conventionnels est problématique parce que ces systèmes ont été conçus principalement pour des applications terrestres. Le problème provient du fait que les éléments retrouvés en mer sont totalement différents des éléments « terrestres ». Pour remédier à la situation, nous proposons d'utiliser un nouveau modèle spatial qui permet de gérer efficacement les relations spatiales entre les éléments d'un jeu de données océanographiques. Ce modèle est basé sur le diagramme de Voronoï (DV) tridimensionnel et son dual la triangulation de Delaunay (TD). Nous présentons dans cet article les principales propriétés du DV et de la TD tridimensionnels, un survol des principaux algorithmes et structures de données requis pour les construire et les modifier, ainsi que les opérations d'analyse spatiale qu'il est possible de réaliser avec un tel modèle. Nous démontrons aussi, par quelques exemples, comment ce modèle peut être utile à plusieurs applications en océanographie.

ABSTRACT. Managing oceanographic data with traditional geographical information systems (GIS) is a difficult task because these systems have been primarily designed for land-based applications. The main problem is that the nature of objects at sea is completely different from the nature of objects found on the land. To solve this problem, we propose using a new spatial model to handle topological relationships between objects in oceanographic datasets. This model is based on the three-dimensional Voronoi diagram (VD) and its dual the Delaunay triangulation (DT). We present in this paper the main properties of the VD and the DT, algorithms to construct and modify them, and show how some 3D GIS operations are greatly simplified when a spatial model is built upon them. We also show some examples of applications in oceanography.

MOTS-CLÉS : SIG tridimensionnel, modélisation, océanographie, Voronoï, Delaunay.

KEYWORDS: Three-dimensional GIS, Modelling, Oceanography, Voronoi, Delaunay.

1. Introduction

L'océanographie est la science qui s'intéresse aux océans et à l'interaction des différents phénomènes biologiques et physiques qui y surviennent. Afin d'étudier la variation spatiale des propriétés de l'eau (par exemple la température, la salinité ou quelconques attributs), des échantillons doivent être recueillis. La collecte de ces échantillons, qui représentent en fait la valeur d'un attribut à une position donnée ($x, y, profondeur$), se fait habituellement à l'aide d'une sonde qui est bougée verticalement sous un bateau ; des « colonnes d'eau », où tous les échantillons ont la même position planimétrique mais différentes profondeurs, sont donc recueillies. Une représentation tridimensionnelle (volumétrique) d'une masse d'eau est construite à partir de plusieurs colonnes d'eau. Il en résulte des jeux de données qui ont une distribution fortement anisotrope : les données sont abondantes verticalement, mais clairsemées horizontalement. Les données océanographiques sont donc totalement différentes des données normalement retrouvées dans des applications sur la terre ferme, où les objets sont en deux dimensions, représentés par des points, des lignes et des polygones, et ont normalement une distribution isotrope. Une autre différence majeure est qu'en mer les composantes du milieu ont tendance à bouger continuellement, alors que sur la terre elles sont normalement fixes.

L'intégration de données océanographiques dans les systèmes d'information géographique (SIG) traditionnels¹ est problématique parce que ces systèmes ont été créés pour des applications terrestres. Un jeu de données doit par exemple être « réduit » d'une dimension, en le découpant selon plusieurs tranches, pour être intégré et analysé dans un SIG traditionnel. Les nombreuses disparités entre les données océanographiques et les SIG ont été décrites par de nombreux chercheurs (Davis *et al.*, 1988; Li *et al.*, 1993; Lockwood *et al.*, 1995). La plupart des problèmes peuvent être réglés en utilisant un modèle spatial basé sur le diagramme de Voronoï (DV), tel que Gold *et al.* (1995) le démontrent. Tel qu'expliqué à la section 3, le DV s'adapte naturellement à la distribution spatiale des données et crée une mosaïque qui peut servir à définir les relations spatiales entre des objets non connectés. Il peut de plus être modifié localement, ce que les structures des SIG traditionnels ne peuvent faire (Gold, 1991). Wright *et al.* (1997), dans une revue concernant l'application des SIG au domaine marin, affirment que le DV est la seule solution aux problèmes de traitement et modélisation des données océanographiques. La solution que Gold et Condal proposent ne concerne par contre que le DV en 2D, pour des applications reliées par exemple à la navigation maritime et à la bathymétrie.

Cet article se veut une extension des travaux de Gold *et al.* (1995). Nous proposons d'utiliser le diagramme de Voronoï tridimensionnel pour gérer les relations spatiales entre les éléments d'un jeu de données océanographiques. Tel que démontré à la section 3, les concepts et propriétés du DV peuvent être généralisés en 3D, et donc un modèle spatial basé sur celui-ci permet de résoudre la plupart des problèmes

1. Nous considérons comme « traditionnels » les SIG qui sont basés sur les architectures points-lignes-polygones et/ou raster, telles que développées dans les années 1970.

qui surviennent lors de la modélisation de données océanographiques. Même si la généralisation des concepts est relativement simple, leur implémentation est une tâche compliquée. Nous présentons donc à la section 4 un survol des principaux algorithmes et structures de données pour construire et modifier dynamiquement le DV et son dual géométrique la triangulation de Delaunay (TD). Un modèle spatial basé sur le DV et la TD en 3D possède plusieurs avantages que les autres méthodes n'ont pas, tel que démontré à la section 5. Un de ces avantages est que plusieurs opérations d'analyse spatiale 3D sont grandement simplifiées et optimisées ; nous en démontrons quelques-unes à la section 6. La section 7 présente finalement, à l'aide de quelques exemples, comment ce modèle pourrait résoudre des problèmes reliés à différents domaines de l'océanographie.

2. Travaux précédents

Comme c'est le cas pour leurs pendants en deux dimensions, les SIG tridimensionnels offrent deux représentations : vectorielle et matricielle. Ces deux représentations sont liées aux structures de données communément retrouvées dans les produits commerciaux.

Les modèles vectoriels permettent de représenter individuellement chaque objet à l'aide de primitives (points, lignes, polygones et polyèdres). En trois dimensions, la composante volumétrique d'un objet (par exemple une maison ou un tuyau d'égout) n'est que très rarement stockée explicitement : seulement ses frontières (*b-rep*) permettent de le définir. Les relations topologiques entre les objets sont basées, tout comme en 2D, sur les intersections et inclusions entre les objets (Egenhofer, 1995), et la construction de la topologie pour un ensemble d'objets est une opération *globale* qui doit être faite chaque fois qu'une modification survient (ajout, suppression ou mouvement d'un objet). Des méthodes récentes permettent de modifier seulement une certaine portion d'une carte, mais les opérations nécessaires sont quand même basées sur la visite d'un graphe – quoique efficace, la mise à jour de la topologie lorsqu'un ou plusieurs objets sont en mouvement est très lente, voire même impossible. Zlatanova *et al.* (2004) présentent une revue des principaux modèles vectoriels, qui sont pour la plupart basés sur les travaux de Molenaar (1990). Ces modèles sont très difficilement applicables dans un contexte océanographique où les éléments (échantillons) ne s'intersectent généralement pas et ont tendance à bouger (une reconstruction *globale* de la topologie est impensable).

Les modèles matriciels en 3D divisent l'espace en cellules régulières ; à chaque cellule, appelée *voxel*, est attribué un ou plusieurs attributs. Ces modèles sont particulièrement utiles pour représenter des phénomènes continus parce que les cellules occupent tout l'espace couvert par un jeu de données. Parce qu'ils offrent une définition simple et directe des relations spatiales entre les éléments, ces modèles ont été utilisés par plusieurs pour modéliser des données géoscientifiques (Raper, 1989; Jones, 1989; Mason *et al.*, 1994). Ils comportent par contre plusieurs désavantages : les objets (échantillons) ne peuvent pas être représentés individuellement (les données

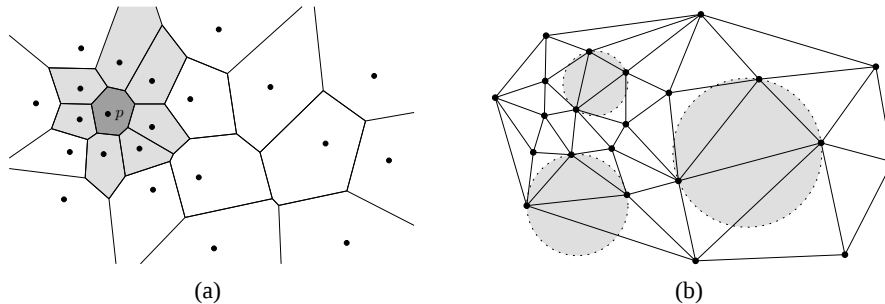


Figure 1. (a) le DV et (b) la TD pour un même ensemble de points en 2D

originales sont « perdues » lorsque converties au format voxel) ; ils supportent mal les transformations affines (rotation, translation et changement d'échelle) ; et finalement, la quantité de données peut devenir rapidement très grande si l'utilisateur désire avoir une représentation détaillée. Pour remédier au problème de la quantité de données, il est par contre possible de créer des mosaïques irrégulières, où l'espace est décomposé récursivement (Samet, 1984).

Pour la modélisation d'objets tridimensionnels, des triangulations 3D ont aussi été utilisées par certains (Lattuada, 1998; Pilouk, 1996).

3. Propriétés du diagramme de Voronoï et de la triangulation de Delaunay

Le diagramme de Voronoï pour un ensemble S de points, appelés sites, dans l'espace euclidien $\mathbb{R}^d$ de dimension d , est la subdivision de cet espace en régions, nommées cellules de Voronoï, qui contiennent tous les points de l'espace étant plus près du site générateur de la région que des autres sites. D'un point de vue mathématique, chaque cellule de Voronoï $\mathcal{V}_p$ générée par un site p est :

$$\mathcal{V}_p = \{x \in \mathbb{R}^d \mid \|x - p\| \leq \|x - q\|, \forall q \in S\}.$$

En deux dimensions, une cellule $\mathcal{V}_p$ est un polygone convexe ayant un certain nombre de cellules voisines (les cellules ayant une arête commune) ; par exemple la cellule du site p de la figure 1(a) possède sept cellules voisines. La forme et la grandeur des cellules dépendent de la configuration des sites. En trois dimensions, une cellule de Voronoï est un polyèdre convexe formé de faces elles aussi convexes (figure 2(a)). Il existe une dualité entre le DV et la triangulation de Delaunay (TD), et ce dans toutes les dimensions. La dualité signifie que les deux structures représentent la même chose, mais d'un point de vue différent. En 2D, la TD est la subdivision du plan en triangles (les sommets des triangles étant les sites générateurs des cellules de Voronoï) ayant un *cercle circonscrit vide* (figure 1(b)). Un cercle est vide s'il ne contient aucun site en son intérieur, mais plusieurs sites peuvent être directement sur le cercle. La TD est populaire pour la modélisation de terrain parce que, entre toutes les triangulations

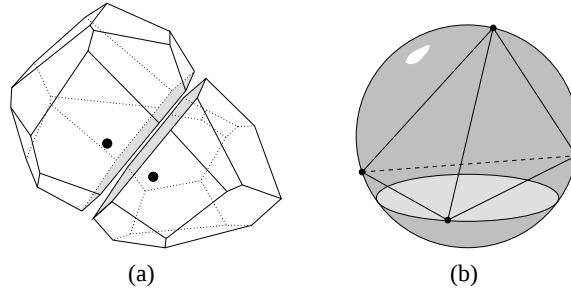


Figure 2. (a) Deux cellules de Voronoï adjacentes. (b) Un tétraèdre est Delaunay si sa sphère circonscrite ne contient aucun site en son intérieur

possibles d'un ensemble de points dans le plan, elle maximise l'angle minimal de chaque triangle. En d'autres termes, un triangle de Delaunay est le plus équilatéral possible, ce qui est une propriété utile pour l'interpolation par exemple.

En trois dimensions, la TD divise l'espace en tétraèdres qui ont chacun une sphère circonscrite vide (figure 2(b)). Les tétraèdres, qui remplissent complètement l'espace, ne se chevauchent jamais et chaque tétraèdre est adjacent à exactement quatre tétraèdres (sauf ceux qui sont situés sur la frontière de l'ensemble de points). La TD est unique pour un ensemble de points, sauf s'il y a des dégénérescences dans l'ensemble (cinq points cosphériques ou plus). Dans ce cas, plusieurs triangulations respectent le critère de Delaunay et un choix arbitraire doit être fait pour obtenir une triangulation valide. Contrairement à la TD en 2D où le nombre de triangles pour une triangulation de n points est toujours le même, le nombre de tétraèdres dans une TD en 3D dépend de la configuration des n points, et peut être jusqu'à $\mathcal{O}(n^2)$. La plupart des propriétés du DV/TD en 2D sont valides en 3D, sauf celles concernant l'angle minimal de chaque triangle. En effet, un tétraèdre respectant le critère de Delaunay peut être quasiment « plat », c'est-à-dire que ses quatre sommets se situent tous près de l'équateur d'une sphère, et donc le volume du tétraèdre tend vers zéro. Pour plusieurs applications où la TD est utilisée directement, par exemple en ingénierie pour des analyses d'éléments finis ou lorsque les tétraèdres servent de base à l'interpolation, ces tétraèdres peuvent créer des problèmes et doivent être éliminés. On peut se demander ici pourquoi utiliser des tétraèdres de Delaunay si leurs propriétés ne sont pas optimales ? Premièrement, dans la majorité des cas la TD tend à favoriser des tétraèdres « ronds » (Rajan, 1991). Deuxièmement, le DV n'est pas affecté par les tétraèdres plats, c'est-à-dire qu'une cellule de Voronoï 3D sera relativement ronde même si son point générateur fait partie d'un tétraèdre plat. Finalement, plusieurs opérations d'analyse spatiale utilisent directement les propriétés du DV, tel que décrit à la section 6.

La dualité entre la TD et le DV en 3D est simple, chaque élément d'une structure correspond à un et un seul élément de sa structure duale : un sommet correspond à un polyèdre, et vice versa ; une arête correspond à une face, et vice versa. Par exemple, un sommet de Delaunay p correspond à une cellule de Voronoï (figure 3(a)), une arête

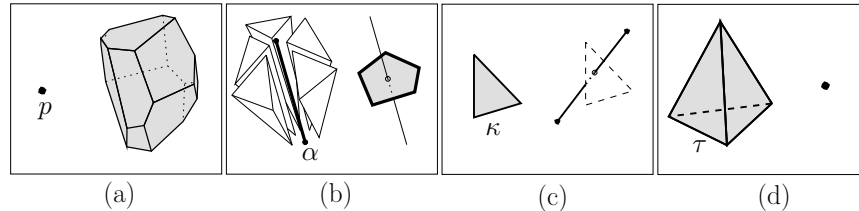


Figure 3. Dualité entre les éléments de la TD et du DV

de Delaunay α correspond à une face de Voronoï (figure 3(b)), une face de Delaunay κ correspond à une arête de Voronoï (figure 3(c)), et un tétraèdre de Delaunay τ correspond à un sommet de Voronoï (figure 3(d)).

4. Algorithmes et structures de données

Tel qu'expliqué à la section précédente, le DV et la TD sont géométriquement équivalents. Une conséquence directe de cette dualité est que si une structure est connue, l'autre peut toujours être déduite et construite. Parce que, du point de vue d'un algorithme ou d'une structure de données, la gestion de tétraèdres est plus simple que la gestion de polyèdres (un tétraèdre a un nombre fixe de sommets et de voisins), nous construisons et modifions un DV en n'utilisant que la TD.

Dans cette section, nous décrivons premièrement comment extraire une cellule de Voronoï à partir d'une TD, et ensuite nous présentons les opérations de base nécessaires pour construire et modifier une TD. Il est à noter ici que toutes les opérations décrites dans cet article sont effectuées directement en mémoire vive. Les arêtes, faces et tétraèdres ne sont jamais stockés dans une base de données – la sauvegarde d'un graphe dans une base de données est problématique (Güting, 1994) – seulement les points sont sauvegardés et le DV/TD est reconstruit à chaque fois. La reconstruction d'une TD est une opération très rapide, tel qu'expliqué à la section 4.4. Nous discutons aussi brièvement des structures de données possibles pour stocker efficacement la TD et/ou le DV.

4.1. Extraction du DV à partir de la TD

Lorsque la TD est connue, la construction du DV implique l'extraction de tous ses éléments (sommets, arêtes, faces et polyèdres). Un seul sommet, ou une seule arête, de Voronoï peut être extrait facilement : un sommet de Voronoï, dual à un tétraèdre τ , est situé exactement au centre de la sphère circonscrite à τ ; et une arête de Voronoï, qui correspond à une face (triangulaire) de Delaunay κ , est formée par les deux sommets de Voronoï duals aux deux tétraèdres partageant κ . L'extraction d'une face de Voronoï est par contre une opération plus complexe. Tel qu'illustré à la figure 3(b), une face de

Voronoi, qui correspond à une arête de Delaunay α , est formée par les sommets duals à tous les tétraèdres incidents à α (en d'autres termes, à tous les tétraèdres partageant α). Il suffit donc de « tourner » autour d'une arête et d'extraire les sommets de Voronoi de chaque tétraèdre pour en extraire une face. La construction d'une cellule de Voronoi $\mathcal{V}_p$ correspondant à un sommet p est similaire : elle est formée par tous les sommets de Voronoi de tous les tétraèdres incidents à p . Parce qu'une cellule de Voronoi est convexe par définition, il est possible de collecter tous les sommets de Voronoi et ensuite de calculer l'enveloppe convexe des points (voir Barber *et al.* (1996) pour plus de détails). Une méthode plus simple consiste à identifier toutes les arêtes de Delaunay incidentes à p , et à extraire chaque face de la cellule.

Construire le DV lorsque la TD d'un ensemble de n points est connue implique la visite de chaque vertex du diagramme, et donc a une complexité de $\mathcal{O}(n)$.

4.2. Localisation de points

Le problème de localisation de points consiste à identifier à l'intérieur de quel tétraèdre un point se situe. C'est une opération fondamentale en géométrie algorithmique qui est requise pour diverses opérations, par exemple pour insérer un nouveau point dans une triangulation ou pour interpoler. La méthode que nous décrivons ici, appelée « walk », est décrite dans les premiers articles concernant la construction de triangulations en 2D (Gold *et al.*, 1977; Green *et al.*, 1978). Sa généralisation en 3D est simple car elle utilise seulement les relations d'adjacence entre triangles pour naviguer. L'idée est la suivante. Soit τ un tétraèdre de départ, on marche vers l'un de ses voisins τ_v si et seulement si le point requête x et τ_v sont du même côté de la face partagée par τ et τ_v . La marche continue, de tétraèdre en tétraèdre, jusqu'à ce que τ n'ait plus aucun voisin qui satisfasse la condition ; τ est donc le tétraèdre qui contient x . Cette méthode est extrêmement simple à implémenter car seulement une fonction est requise (pour déterminer de quel côté d'un plan un point se situe), et elle ne requiert pas de stockage supplémentaire. Elle est aussi très efficace en pratique, comme Mücke *et al.* (1999) le démontrent.

4.3. Flips tridimensionnels

Un « flip » est une opération topologique qui modifie *localement* la configuration de quelques tétraèdres adjacents dans une triangulation ; c'est donc une opération très utile pour insérer, supprimer ou bouger un sommet dans une TD. Soit $S = \{a, b, c, d, e\}$ un ensemble de points en 3D, selon Lawson (1986) il existe trois façons différentes de le trianguler : avec deux, trois ou quatre tétraèdres. La figure 4(a) présente une de ces configurations : lorsque le point e se situe à l'intérieur du tétraèdre $abcd$. La figure 4(b) présente quant à elle les deux autres configurations possibles où le polyèdre $abcde$ est triangulé avec deux ou trois tétraèdres. Ceci permet de définir différents flips ; la notation correspond au nombre de tétraèdres avant et après un flip. Un *flip14* est l'opération pour insérer un nouveau sommet dans un tétraèdre (et

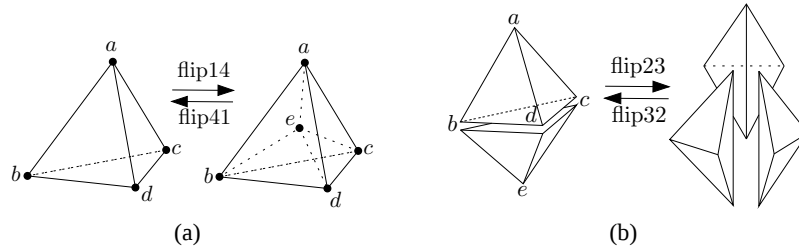


Figure 4. *Flips en trois dimensions*

créer quatre nouveaux tétraèdres), et un *flip41* est l'opération inverse pour supprimer le sommet. Un *flip23* remplace une triangulation de deux tétraèdres par une avec trois tétraèdres, et un *flip32* fait l'action inverse. Lorsqu'un flip est fait dans une triangulation, il n'affecte que localement les quelques tétraèdres concernés, c'est-à-dire que tous les tétraèdres « extérieurs » ne sont pas modifiés.

4.4. Construction et mise à jour de la TD

Construire une TD 3D peut se faire de différentes façons. Une des solutions consiste à construire l'enveloppe convexe de l'ensemble de points projetés sur un paraboloïde en quatre dimensions, et ensuite à reprojeter le résultat dans l'espace original pour obtenir la TD (voir (Edelsbrunner *et al.*, 1986) pour les détails). Les algorithmes et les logiciels permettant de construire une enveloppe convexe dans les dimensions supérieures sont nombreux, voir par exemple *Qhull* (Barber *et al.*, 1996). Une autre solution est l'algorithme *DeWall* qui est basé sur l'approche « diviser pour régner » (Cignoni *et al.*, 1998). Ces algorithmes ne sont par contre utiles que pour construire une TD lorsque tous les points sont connus à l'avance, les modifications locales (insertion, suppression ou mouvement) sont soit lentes ou tout simplement impossibles.

Les algorithmes permettant des modifications locales sont appelés « dynamiques », et la construction de la TD pour un ensemble de points se fait en insérant les points un par un, en mettant à jour la triangulation après chaque insertion (pour que tous les tétraèdres respectent le critère de Delaunay). L'insertion d'un seul point dans une TD ne modifie que localement la triangulation (voir figure 5 pour le cas en deux dimensions, mais la même idée s'applique aux dimensions supérieures). Seulement les tétraèdres dont la sphère circonscrite contient le nouveau point doivent être détruits et remplacés par d'autres. Cette mise à jour peut se faire à l'aide de l'algorithme de Watson (1981) ou bien par une séquence de flips, comme Joe (1991) le démontre. L'algorithme de Joe est par contre plus stable numériquement et plus simple à implémenter car une triangulation complète est préservée lors du processus de mise à jour. Ce qui suit est la description des principales étapes pour insérer un seul point x dans une TD, sans tout reconstruire. La première étape consiste à identifier le tétraèdre contenant x , et à y insérer x avec un *flip14*. Tous les nouveaux tétraèdres créés doivent être testés

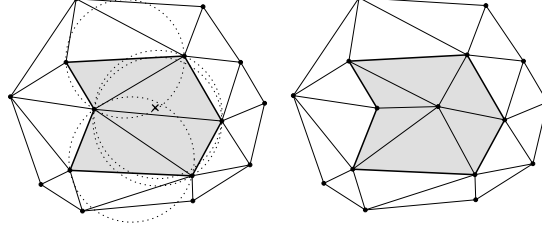


Figure 5. L'ajout ou la suppression d'un seul point dans une TD ne modifie que localement la triangulation. Seulement les triangles en gris sont modifiés

pour être sûr qu'ils sont Delaunay, ce test implique deux tétraèdres : le nouveau tétraèdre τ et son unique voisin τ_v qui n'est pas incident à x . Si τ n'est pas localement Delaunay, selon la configuration de τ , τ_v et des tétraèdres adjacents, il y a trois possibilités : flip23, flip32 ou aucune action. Après un flip, tous les nouveaux tétraèdres créés doivent aussi être testés. Le processus continue jusqu'à ce que tous les tétraèdres incidents à x soient Delaunay.

La complexité de cet algorithme est $\mathcal{O}(n^2)$ pour construire une TD d'un ensemble de n points. C'est donc un algorithme optimal dans le pire des cas, car le nombre de tétraèdres peut être théoriquement $\mathcal{O}(n^2)$ pour certaines configurations très dégénérées. Par contre, lorsque les points sont distribués uniformément le nombre de tétraèdres est $\mathcal{O}(n)$, et, par conséquent, on peut s'attendre en pratique à une complexité de $\mathcal{O}(n \log n)$ (Edelsbrunner *et al.*, 1996).

4.5. Suppression d'un sommet

Tout comme l'insertion d'un point dans une TD, la suppression d'un seul sommet x est une opération locale qui n'implique que quelques tétraèdres (voir la figure 5 pour le cas en 2D). Même si la suppression d'un point apparaît simple, c'est un problème qui est beaucoup plus compliqué que l'insertion, et très peu d'algorithmes sont disponibles.

L'algorithme de Devillers (2002), qui est valide en 2D et dans les dimensions supérieures, est le plus connu et le plus utilisé. En 2D, il implique la suppression de tous les triangles incidents au sommet x , et la retriangulation du polygone (un « trou » dans la triangulation) ainsi formé se fait en ajoutant, un à un, des triangles (une liste prioritaire des triangles potentiels est tenue à jour). L'algorithme est basé sur la construction de l'enveloppe convexe des points du polygone projetés sur un paraboloïde en trois dimensions (Edelsbrunner *et al.*, 1986). Le nombre d'arêtes k incidentes à x est utilisé pour analyser les différents algorithmes. La méthode de Devillers a une complexité de $\mathcal{O}(k \log k)$ en deux dimensions. En trois dimensions, le nombre t de tétraèdres nécessaires pour la retriangulation n'est pas le même que le nombre original de tétra-

èdres, et donc la complexité de l'algorithme est $\mathcal{O}(t \log k)$. Même si cet algorithme est théoriquement valide en 3D, son implémentation exige plusieurs modifications pour gérer les nombreuses dégénérescences et, à notre connaissance, il n'a jamais été implémenté. Une solution plus simple, et plus robuste contre les cas dégénérés, consiste à tester chaque tétraèdre avec chaque point formant le trou pour être sûr qu'il est Delaunay. L'algorithme, qui n'est par contre pas optimal, est disponible dans la librairie CGAL² (Devillers *et al.*, 2003).

Cependant, ces méthodes ne sont pas stables numériquement et détruisent temporairement la triangulation (il faut utiliser des structures de données externes pour représenter le « trou » dans la triangulation), ce qui cause des problèmes lors de l'implémentation. Nous avons donc développé une méthode qui permet la suppression d'un sommet à l'aide d'une série de flips (Ledoux *et al.*, 2005b). L'algorithme est robuste pour tous les cas dégénérés possibles, par exemple lorsque certains points sont à la fois cosphériques et dans un même plan.

4.6. Mouvement de points

Lorsqu'un point est en mouvement, l'insérer, le supprimer et le réinsérer continuellement n'a pas de sens compte tenu de la complexité des opérations à réaliser. Il est possible de réellement bouger un point et de mettre à jour les relations topologiques localement lorsqu'elles sont requises. Roos (1991) et Gold (1991) décrivent une méthode qui utilise les flips pour tenir à jour une triangulation de Delaunay en 2D qui contient des points qui bougent. Le mouvement d'un seul point, de sa position originale à sa destination, implique la mise à jour de toutes les relations topologiques qui ne seront pas valides entre le point de départ et la destination. Si la position d'un point dans une triangulation change de peu, la mise à jour de ses relations topologiques ne sera probablement pas requise. Par contre, dès que le point entre ou sort du cercle circonscrit à un triangle, un flip doit être fait pour s'assurer que tous les triangles respectent le critère de Delaunay.

Ces idées sont aussi valides en trois dimensions, pour la TD, quoique, à notre connaissance, elles n'ont jamais été implantées. Nous travaillons présentement au développement d'un algorithme permettant de garder à jour la TD, avec des flip23 et flip32, lorsqu'un point est en mouvement. Comme pour la suppression d'un sommet, les dégénérescences compliquent grandement l'implémentation d'un tel algorithme.

4.7. Structures de données

Le choix d'une structure de données pour stocker, en mémoire vive, une TD (ou un DV) est un compromis entre la mémoire requise et le nombre de relations topologiques entre les éléments (sommets, arêtes, faces et polyèdres). L'utilisation d'une structure

2. *Computational Geometry Algorithms Library* (www.cgal.org).

de données très simple implique que, lorsque certaines opérations seront faites, par exemple pour extraire les faces d'une cellule de Voronoï, plus d'opérations seront nécessaires que si une structure stockant la TD et son dual était utilisée.

La structure de données la plus simple pour stocker une TD représente explicitement chaque tétraèdre ; chaque enregistrement contient quatre pointeurs vers les quatre sommets et quatre pointeurs vers les voisins. Plusieurs implémentations de la TD (par exemple CGAL) utilisent cette structure parce qu'elle est simple et permet de construire une TD rapidement. Il est aussi possible d'utiliser une structure de données où l'élément de base est une face triangulaire, et d'avoir une liste chaînée de toutes les faces incidentes à chaque arête. Ces structures sont utiles et efficaces principalement pour la construction d'une TD, par contre, les opérations tels que la suppression ou le mouvement d'un point, l'analyse spatiale ou l'extraction du DV ne sont pas optimisées. Nous avons donc développé une nouvelle structure de données, appelée le *augmented quad-edge*, qui permet de stocker symétriquement la TD et le DV dans une même structure. La structure est basée sur le quad-edge (Guibas *et al.*, 1985) qui a été développé pour des surfaces 2D. Chaque cellule tridimensionnelle (tétraèdre ou cellule de Voronoï) est représentée individuellement à l'aide de quad-edges et les cellules sont reliées entre elles par l'arête duale (une arête de Voronoï) à chaque face partagée par deux cellules. La structure de données est simple à implémenter car seulement le quad-edge est requis, et une seule modification doit être apportée pour permettre la représentation d'un DV et d'une TD tridimensionnels. La structure peut aussi être modifiée localement (donc les flips peuvent être utilisés) et la navigation du DV et de la TD est aussi possible. Son principal désavantage est qu'elle utilise plus de mémoire que les autres structures stockant seulement la TD. Plus de détails sont disponibles dans (Ledoux *et al.*, 2005a).

5. Un modèle spatial basé sur le diagramme de Voronoï

Il existe deux façons de conceptualiser l'espace, et elles influencent directement la structure des SIG. La première conçoit l'espace comme étant vide et « peuplé » par différents objets. La seconde conçoit l'espace comme quelque chose de continu, c'est-à-dire que chaque point de l'espace possède un attribut quelconque. Un modèle spatial, aussi appelé un modèle de données spatiales, décrit comment une représentation d'un phénomène (ou de la réalité) doit être stockée dans un ordinateur. Dans le cas des phénomènes continus, sa définition est très semblable à celle de la discrétisation de l'espace. Il est à noter que les deux représentations offertes par les SIG (vectorielle et matricielle) sont souvent faussement utilisées comme synonymes des deux conceptualisations de l'espace. Selon Goodchild (1992), les termes « vectoriel » et « matriciel » sont ambigus parce que, par exemple, les primitives tels point, ligne et polygone peuvent servir à représenter des objets individuels et aussi un phénomène continu s'il est discrétisé, en différentes cellules (par exemple les points et arêtes d'une triangulation).

Un modèle spatial basé sur le DV permet en quelques sortes d'intégrer les deux conceptualisations de l'espace dans une même structure : il permet de représenter individuellement chaque objet (les échantillons dans le cas présent), et la mosaïque que le DV crée remplit complètement l'espace. De plus, chaque cellule de Voronoï possède un nombre défini de voisins et peut donc servir à définir les relations spatiales entre des objets non connectés. En somme, il combine les principaux avantages des représentations vectorielles et matricielles. Lorsque le DV et la TD sont stockés simultanément, deux modèles spatiaux différents, tous deux basés sur la subdivision de l'espace en cellules, sont disponibles pour représenter la variabilité spatiale d'un attribut. Le DV offre un modèle très simple qui consiste à assigner à tous les points à l'intérieur d'une même cellule de Voronoï la valeur de l'attribut du point générateur. La TD permet quant à elle une représentation où une fonction linéaire est assignée à chaque tétraèdre, c'est-à-dire que la valeur de l'attribut à un endroit est estimée par une fonction linéaire définie par les quatre sommets formant le tétraèdre.

Ce qui suit est une liste des principaux avantages de l'utilisation d'un modèle spatial basé sur le DV (et donc aussi sur la TD) :

- 1) la construction du DV/TD est un processus entièrement automatique, même lorsqu'il y a des dégénérescences dans un jeu de données, qui ne requiert pas de paramètre défini par un utilisateur ;
- 2) le stockage du modèle spatial dans une base de donnée est extrêmement simple et efficace car seulement les points peuvent être conservés. La reconstruction du DV est une opération rapide qui peut être effectuée partiellement si seulement une certaine région d'un jeu de données est d'intérêt ;
- 3) le DV ne fait pas que diviser arbitrairement l'espace. La forme et la grosseur de ces cellules s'adaptent à la distribution spatiale des objets, et donc c'est une excellente méthode pour représenter des données fortement anisotropes ;
- 4) les données originales sont conservées et non « perdues », comme c'est le cas lorsqu'un modèle voxel est construit ;
- 5) les mises à jour locales sont possibles. L'insertion, la suppression et même le mouvement d'objets n'impliquent pas la reconstruction totale des relations spatiales d'un jeu de données ;
- 6) l'affichage à l'écran de données est rapide car la majorité des cartes vidéos sont optimisées pour des triangles (lorsque la TD est utilisée) ;
- 7) plusieurs opérations d'analyse spatiale sont possibles lorsque le DV/TD sont disponibles (voir prochaine section).

6. Opérations d'analyse spatiale 3D

Plusieurs requêtes concernant le voisinage sont possibles lorsque le DV/TD d'un ensemble de points est construit. Okabe *et al.* (1994) en présentent plusieurs qui sont basées sur le DV généralisé (différentes métriques) en deux dimensions, mais la plu-

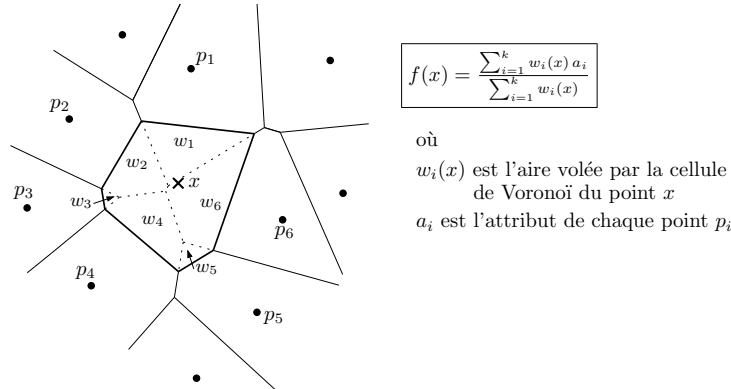


Figure 6. Interpolation par voisins naturels en deux dimensions

part de ces requêtes sont aussi valides en trois dimensions. En plus des requêtes de voisinage, plusieurs opérations d'analyse spatiale sont possibles, et même optimisées, lorsque le DV/TD est disponible. Cette section en présente quelques-unes.

6.1. Interpolation spatiale

L'interpolation consiste à estimer la valeur d'un attribut à un endroit non échantillonné. Elle est requise pour la modélisation, la visualisation, simplement pour avoir une meilleure « connaissance » d'un jeu de données, ou pour convertir des données d'un format à un autre format (par exemple de points clairsemés à voxels). Les méthodes normalement retrouvées dans les SIG commerciaux – qui sont basées sur les distances ou les triangles – peuvent être généralisées en 3D, mais elles s'adaptent mal aux jeux de données dont la distribution est fortement anisotrope (Watson, 1992). Ces problèmes surviennent parce qu'elles ne considèrent pas la configuration spatiale des données.

Il a été démontré par plusieurs que la méthode d'interpolation par *voisins naturels*³ (Sibson, 1981) évite les problèmes des méthodes traditionnelles lorsque l'anisotropie est forte (Gold, 1989; Watson, 1992). Cette méthode, qui est valide dans n'importe quelle dimension, est entièrement basée sur le diagramme de Voronoï pour sélectionner les voisins et pour leur assigner un poids (une importance relative). Tel que la figure 6 le montre, pour interpoler à la position x en 3D, un point temporaire x doit être inséré dans le DV. Les voisins utilisés pour l'interpolation sont ceux ayant une cellule de Voronoï adjacente à celle de x , et le poids de chaque voisin est défini par le volume que la cellule de Voronoï de x vole à la cellule du voisin en l'absence de x dans le DV. La méthode est *exacte* (elle « honore » les échantillons), *locale*, et la

3. *Natural neighbour interpolation method.*

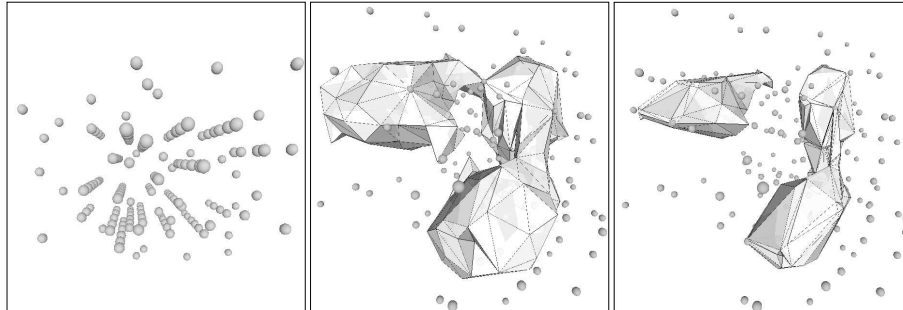


Figure 7. Un simple jeu de données tridimensionnelles (où chaque échantillon a comme attribut la température de l'eau), et deux isosurfaces extraites du jeu de données. Image du milieu : température de 10°C ; image de droite : température de 8°C

première dérivée de la fonction résultante est possible partout, sauf aux échantillons. Elle s'adapte de plus particulièrement bien aux données anisotropes parce que le DV permet de sélectionner les voisins qui sont à la fois près de x et qui l'entourent.

Quoique la méthode ait été utilisée avec succès pour des applications en deux dimensions (Watson, 1992), son utilisation en 3D est encore restreinte parce que son implémentation requiert la construction de deux DV – un sans x et un avec x – ainsi que le calcul du volume des cellules de Voronoï. Une méthode qui utilise les flips et qui est basée sur un algorithme dynamique pour l'insertion d'un point (voir les sections 4.3 et 4.4), a été récemment développée par les auteurs de cet article (Ledoux *et al.*, 2004). L'algorithme a une complexité qui est la même que pour l'insertion d'un seul point dans un DV, et nous croyons qu'il est considérablement plus simple à implémenter que les autres méthodes parce que seulement quelques modifications à l'algorithme d'insertion (qui est connu et robuste) sont requises.

6.2. Construction d'isosurfaces

Un des meilleurs moyens de visualiser un jeu de données tridimensionnelles consiste à construire une ou plusieurs isosurfaces. Une isosurface (voir figure 7), est l'équivalent tridimensionnel d'une isoligne (ou d'une courbe de niveau) : elle correspond à l'espace où un attribut a une valeur constante (par exemple où la température de l'eau est à 7°C). L'algorithme le plus connu pour extraire une approximation d'une isosurface est appelé « marching cubes » (Lorensen *et al.*, 1987), et a été développé pour des données en format voxel seulement. Il peut néanmoins être facilement modifié pour fonctionner avec une subdivision en tétraèdres : chaque tétraèdre d'une TD est visité et les intersections entre celui-ci et une isosurface d'une certaine valeur sont calculées par interpolation linéaire sur les arêtes. Il y a trois cas possibles pour un tétraèdre : aucune intersection ; trois arêtes sont intersectées et donc une face triangulaire

de la surface est créée ; quatre arêtes sont intersectées et donc deux faces triangulaires doivent être créées. La surface obtenue avec un tel algorithme ne contient pas de trou et est formée de faces triangulaires adjacentes, ce qui est idéal pour un affichage rapide.

6.3. Gestion de données temporelles

L'insertion, la suppression et le mouvement d'objets sont des opérations qui peuvent se faire localement avec le DV, et donc chaque opération est réversible. Comme Gold (1996) le démontre, en conservant seulement un fichier de l'historique des opérations faites sur une carte, il est possible de reconstruire une carte (avec les relations topologiques entre les objets à jour) à n'importe quel moment. Cette façon de faire résout un problème majeur que les SIG ont avec les données temporelles, et est valide en 2D et 3D. Il n'y a donc pas de raison pour stocker différentes versions d'une même carte à différentes dates. Lorsqu'une carte à une date donnée est requise, elle est seulement reconstruite à partir des données originales et du fichier historique. Une carte peut aussi être visualisée comme un « film » des changements survenus pendant un certain laps de temps.

7. Applications en océanographie

Valavanis (2002), dans sa revue exhaustive des différentes utilisations des SIG en océanographie, note que la grande majorité des projets décrits dans la littérature utilisent les SIG commerciaux, et que cela constitue un handicap à l'analyse des données. Comme plusieurs avant lui l'ont fait (Li *et al.*, 1993; Lockwood *et al.*, 1995), il affirme qu'un système dynamique et tridimensionnel est requis pour pouvoir modéliser adéquatement l'environnement marin. La principale conséquence de l'inexistence d'un tel système est que très peu de détails sont disponibles à propos de ses applications potentielles. Dans cette section, nous en décrivons néanmoins quelques-unes pour lesquelles un modèle spatial, basé sur le DV/TD, peut être utile. La plupart de ces applications, qui couvrent plusieurs domaines reliés à l'océanographie, requièrent la combinaison de plusieurs propriétés du DV et d'opérations d'analyse spatiale.

Visualisation des attributs de l'eau. Il est notoire que la visualisation de données tridimensionnelles est problématique, même lorsqu'un environnement offrant rotations, translations et zooms est disponible. De bons « outils » de visualisation peuvent aider grandement à comprendre la distribution spatiale d'un attribut, et donc la visualisation devient en quelques sortes une opération d'analyse spatiale qualitative. Différentes méthodes ont été proposées dans plusieurs domaines (géologie, océanographie, médecine, ingénierie, etc.) pour améliorer les fonctions de base des outils de visualisation. Avec les nouvelles techniques développées récemment en infographie, il est par exemple possible d'afficher à l'écran d'un utilisateur plusieurs isosurfaces en utilisant des surfaces transparentes, en leur assignant différentes couleurs, ou bien en

les « épluchant » partiellement. Une autre méthode consiste à découper un jeu de données selon un certain plan, à interpoler à différents endroits sur le plan, et à y afficher des courbes de niveau 2D. D'autres méthodes sont évidemment possibles, et Head *et al.* (1997) en présentent quelques-unes qui sont reliées aux particularités des jeux de données océanographiques.

Applications en temps réel. Le terme « temps réel » fait référence à l'utilisation d'un SIG en mer lorsque les données sont recueillies, par exemple pour évaluer la qualité d'un jeu de données. La collection de données océanographiques est une opération très dispendieuse, et il est donc préférable de s'assurer de la qualité des données recueillies lorsque le bateau est près de la zone étudiée. Avec un modèle spatial basé sur le DV/TD, les données pourraient être intégrées au système au fur et à mesure qu'elles sont recueillies, et, parce que les relations spatiales entre les éléments sont toujours à jour, à tout moment il serait possible de visualiser les données (par exemple à l'aide d'isosurfaces) ou de les analyser en utilisant n'importe quelles méthodes mentionnées précédemment. De plus, si la qualité d'un jeu de données n'était pas suffisante, ou bien si une erreur s'était glissée, de nouvelles données pourraient être intégrées ou supprimées sans que la reconstruction de toutes les relations ne soit nécessaire. Hatcher *et al.* (1999) présentent plusieurs exemples d'applications en mer en temps réel.

Étude des remontées des eaux profondes. La remontée des eaux profondes est un phénomène par lequel les eaux relativement froides et riches en nutriments remontent à la surface près des côtes. L'étude de ce phénomène est particulièrement important parce que 50 % de la pêche mondiale se fait dans les régions affectées par ce phénomène, et que ces régions ne comptent que pour 0.1 % de la surface de tous les océans (Valavanis, 2002). Su *et al.* (1999) ont étudié le phénomène dans la baie Monterey aux États-Unis en visualisant certaines propriétés de l'eau durant une certaine période de temps. Ils ont généré des isosurfaces pour chaque attribut, et les ont ensuite animées. La localisation et les patterns de ces isosurfaces aident à déterminer les caractéristiques des remontées des eaux profondes. Ils ont dû convertir toutes les données au format voxel, mais si un modèle spatial basé sur le DV/TD avait été utilisé, les isosurfaces auraient pu être générées et affichées directement à partir des données originales, et il aurait même été possible de les garder à jour lorsque les points sont en mouvement. La simulation tridimensionnelle aurait même pu être visualisée en temps réel, et stoppée à tout moment pour analyser les données.

Biogéographie. La biogéographie est l'étude de la distribution spatiale de la biodiversité. Elle nous aide à comprendre où les animaux et les plantes vivent, et tente d'en expliquer les raisons. Son lien avec les SIG est donc évident. Pour le moment, la majorité des travaux sont faits avec des systèmes 2D (par exemple Schick (2002)), mais, comme c'est le cas pour les autres domaines, un système 3D est souhaité. Comme le souligne Dawn Wright (2003), professeure au Department of Geosciences de la Oregon State University et spécialiste des applications SIG en océanographie,

un tel système pourrait servir à « suivre » un mammifère marin dans ses mouvements et à analyser par exemple quelles températures il rencontre. En combinant différents jeux de données, il serait possible d'expliquer le comportement de l'animal et même de prédire ses mouvements futurs.

Gestion des stocks de pêche. Même si la majorité des chercheurs reconnaissent les bénéfices de l'utilisation des SIG commerciaux pour les applications reliées à la pêche (Meaden (1999) et Valavanis (2002) donnent plusieurs exemples d'applications), ils critiquent ses limitations et souhaitent un système dynamique et 3D. Récemment, de nouvelles technologies ont été développées et c'est maintenant possible, à l'aide d'un sonar, de détecter des bancs de poissons, et donc d'étudier leur distribution et leur abondance. L'intégration de ces données et de celles concernant les attributs de l'eau et du fond marin (température, type d'algues, salinité, etc.) dans un même système aideraient grandement à comprendre leurs relations et à identifier les mesures nécessaires à leur exploitation durable.

8. Conclusion

Le principal objectif de cette recherche est de bâtir un modèle spatial complet pour la gestion et l'analyse de données océanographiques. Nous avons démontré qu'un modèle spatial basé sur le diagramme de Voronoï tridimensionnel (et son dual) comble de nombreuses lacunes des SIG et des systèmes utilisant des voxels. Le DV permet de discrétiser l'espace tridimensionnel couvert par un jeu de données océanographiques en cellules qui s'adaptent naturellement à la distribution des données (même les distributions les plus anisotropes). Le modèle spatial peut de plus être modifié localement, ce qui permet de modéliser la composante dynamique du milieu marin. Tel que démontré dans cet article, le DV n'est pas qu'une solution théorique puisque de nombreux algorithmes, pour le construire et le modifier, existent et ont été implémentés. De plus, plusieurs opérations d'analyse spatiale sont grandement facilitées lorsque le DV et la TD sont construits, par exemple les requêtes de voisinage, les fonctions d'interpolation et d'extraction d'isosurfaces.

Nous avons déjà implémenté plusieurs opérateurs pour la construction et la modification du DV/TD, et nous planifions d'implémenter tous les algorithmes décrits dans l'article. Nous avons aussi développé quelques fonctions d'analyse spatiale 3D et travaillons présentement à bâtir une liste plus complète.

Finalement, les résultats de cette recherche ne se limitent pas qu'à l'océanographie. Il existe en effet plusieurs similarités entre les données et les fonctions requises pour la modélisation en océanographie et celles nécessaires dans d'autres disciplines, par exemple la géologie, la météorologie et la géophysique.

Remerciements

Nous tenons à remercier la Chaire Marie Curie de l'Union européenne et le Hong Kong's Research Grants Council (projet PolyU 506800E) pour leur financement. Une partie de la recherche pour cet article a été effectuée lorsque nous étions à la Hong Kong Polytechnic University, et nous tenons à la remercier de son soutien financier.

9. Bibliographie

- Barber C. B., Dobkin D. P., Huhdanpaa H. T., « The Quickhull algorithm for convex hulls », *ACM Transactions on Mathematical Software*, vol. 22, n° 4, p. 469-483, 1996.
- Cignoni P., Montani C., Scopigno R., « DeWall : a fast divide & conquer Delaunay triangulation algorithm in E^d », *Computer-Aided Design*, vol. 30, n° 5, p. 333-341, 1998.
- Davis B. E., Davis P. E., « Marine GIS : Concepts and Considerations », *Proceedings GIS/LIS '88*, Falls Church, VA, USA, 1988.
- Devillers O., « On Deletion in Delaunay Triangulations », *International Journal of Computational Geometry and Applications*, vol. 12, n° 3, p. 193-205, 2002.
- Devillers O., Teillaud M., « Perturbations and Vertex Removal in a 3D Delaunay Triangulation », *Proceedings 14th ACM-SIAM Symposium on Discrete Algorithms (SODA)*, Baltimore, MD, USA, p. 313-319, 2003.
- Edelsbrunner H., Seidel R., « Voronoi Diagrams and Arrangements », *Discrete & Computational Geometry*, vol. 1, p. 25-44, 1986.
- Edelsbrunner H., Shah N. R., « Incremental Topological Flipping Works for Regular Triangulations », *Algorithmica*, vol. 15, p. 223-241, 1996.
- Egenhofer M. J., Topological relations in 3D, Technical report, University of Maine, Orono, USA, 1995.
- Gold C. M., « Surface Interpolation, spatial adjacency and GIS », in J. Raper (ed.), *Three Dimensional Applications in Geographic Information Systems*, Taylor & Francis, p. 21-35, 1989.
- Gold C. M., « Problems with Handling Spatial Data—the Voronoi Approach », *CISM Journal*, vol. 45, n° 1, p. 65-80, 1991.
- Gold C. M., « An Event-Driven Approach to Spatio-Temporal Mapping », *Geomatica, Journal of the Canadian Institute of Geomatics*, vol. 50, n° 4, p. 415-424, 1996.
- Gold C. M., Charters T. D., Ramsden J., « Automated contour mapping using triangular element data structures and an interpolant over each triangular domain », in J. George (ed.), *Proceedings Siggraph '77*, vol. 11(2) of *Computer Graphics*, p. 170-175, 1977.
- Gold C. M., Condal A. R., « A Spatial Data Structure Integrating GIS and Simulation in a Marine Environment », *Marine Geodesy*, vol. 18, p. 213-228, 1995.
- Goodchild M. F., « Geographical Data Modeling », *Computers & Geosciences*, vol. 18, n° 4, p. 410-408, 1992.
- Green P. J., Sibson R., « Computing Dirichlet tessellations in the plane », *Computer Journal*, vol. 21, n° 2, p. 168-173, 1978.

- Guibas L. J., Stolfi J., « Primitives for the Manipulation of General Subdivisions and the Computation of Voronoi Diagrams », *ACM Transactions on Graphics*, vol. 4, p. 74-123, 1985.
- Gütting R. H., « GraphDB : Modeling and Querying Graphs in Databases », *Proceedings 20th VLDB Conference*, Santiago, Chile, p. 297-308, 1994.
- Hatcher G. A. J., Maher N., « Real-time GIS for Marine Applications », in , D. J. Wright , D. Bartlett (eds), *Marine and Coastal Geographic Information Systems*, Taylor & Francis, London, p. 137-147, 1999.
- Head M. E. M., Luong P., Costolo J. H., Countryman K., Szczechowski C., « Applications of 3-D visualizations of oceanographic data bases », *Proceedings Oceans '97—MTS/IEEE*, vol. 2, p. 1210-1215, 1997.
- Joe B., « Construction of three-dimensional Delaunay triangulations using local transformations », *Computer Aided Geometric Design*, vol. 8, p. 123-142, 1991.
- Jones C. B., « Data structures for three-dimensional spatial information systems in geology », *International Journal of Geographical Information Systems*, vol. 3, n° 1, p. 15-31, 1989.
- Lattuada R., A triangulation based approach to three dimensional geoscientific modelling, PhD thesis, Department of Geography, Birkbeck College, University of London, London, UK, 1998.
- Lawson C. L., « Properties of n -dimensional triangulations », *Computer Aided Geometric Design*, vol. 3, p. 231-246, 1986.
- Ledoux H., Gold C. M., « An Efficient Natural Neighbour Interpolation Algorithm for Geoscientific Modelling », in , P. F. Fisher (ed.), *Developments in Spatial Data Handling—11th International Symposium on Spatial Data Handling*, Springer, p. 97-108, 2004.
- Ledoux H., Gold C. M., « Simultaneous Storage of Primal and Dual Three-Dimensional Subdivisions », *Computers, Environment and Urban Systems*, 2005a. Accepté.
- Ledoux H., Gold C. M., Baciú G., « Flipping to Robustly Delete a Vertex in a Delaunay Tetrahedralization », *Proceedings International Conference on Computational Science and its Applications — ICCSA 2005*, LNCS 3480, Springer-Verlag, Singapore, p. 737-747, 2005b.
- Li R., Saxena N. K., « Development of an Integrated Marine Geographic Information System », *Marine Geodesy*, vol. 16, p. 293-307, 1993.
- Lockwood M., Li R., « Marine Geographic Information Systems—What Sets Them Apart ? », *Marine Geodesy*, vol. 18, p. 157-159, 1995.
- Lorensen W. E., Cline H. E., « Marching Cubes : A High Resolution 3D Surface Construction Algorithm », *Computer Graphics*, vol. 4, p. 163-168, 1987.
- Mason N. C., O'Conaill M. A., Bell S. B. M., « Handling four-dimensional geo-referenced data in environmental GIS », *International Journal of Geographical Information Systems*, vol. 8, n° 2, p. 191-215, 1994.
- Meaden G. J., « Applications of GIS to Fisheries Management », in , D. J. Wright , D. Bartlett (eds), *Marine and Coastal Geographic Information Systems*, Taylor & Francis, London, p. 205-226, 1999.
- Molenaar M., « A Formal Data Structure for Three Dimensional Vector Maps », *Proceedings 4th International Symposium on Spatial Data Handling*, Zurich, Switzerland, p. 830-843, 1990.

- Mücke E. P., Saías I., Zhu B., « Fast randomized point location without preprocessing in two- and three-dimensional Delaunay triangulations », *Computational Geometry—Theory and Applications*, vol. 12, p. 63-83, 1999.
- Okabe A., Boots B., Sugihara K., « Nearest neighbourhood operations with generalized Voronoi diagrams : a review », *International Journal of Geographical Information Systems*, vol. 8, n° 1, p. 43-71, 1994.
- Pilouk M., Integrated Modelling for 3D GIS, PhD thesis, ITC, The Netherlands, 1996.
- Rajan V. T., « Optimality of the Delaunay triangulation in $\mathbb{R}^d$ », *Proceedings 7th Annual Symposium on Computational Geometry*, ACM Press, North Conway, New Hampshire, USA, p. 357-363, 1991.
- Raper J. (ed.), *Three Dimensional Applications in Geographic Information Systems*, Taylor & Francis, London, 1989.
- Roos T., Dynamic Voronoi diagrams, PhD thesis, Universität Würzburg, Germany, 1991.
- Samet H., « The Quadtree and Related Hierarchical Data Structures », *ACM Computing Surveys*, vol. 16, n° 2, p. 187-260, 1984.
- Schick R., « Using GIS to Track Right Whales and Bluefin Tuna in the Atlantic Ocean », in , D. J. Wright (ed.), *Undersea with GIS*, ESRI Press, Redlands, USA, p. 65-84, 2002.
- Sibson R., « A brief description of natural neighbour interpolation », in , V. Barnett (ed.), *Interpreting Multivariate Data*, Wiley, New York, USA, p. 21-36, 1981.
- Su Y., Sheng Y., « Visualizing Upweilling at Monterey Bay in an Integrated Environment of GIS and Scientific Visualization », *Marine Geodesy*, vol. 22, p. 93-103, 1999.
- Valavanis V. D., *Geographic Information Systems in Oceanography and Fisheries*, Taylor & Francis, 2002.
- Watson D. F., « Computing the n -dimensional Delaunay tessellation with application to Voronoi polytopes », *Computer Journal*, vol. 24, n° 2, p. 167-172, 1981.
- Watson D. F., *Contouring : A Guide to the Analysis and Display of Spatial Data*, Pergamon Press, Oxford, UK, 1992.
- Wright D. J., Communication personnelle, 2003.
- Wright D. J., Goodchild M. F., « Data from the Deep : Implications for the GIS Community », *International Journal of Geographical Information Science*, vol. 11, n° 6, p. 523-528, 1997.
- Zlatanova S., Abdul Rahman A., Shi W., « Topological models and frameworks for 3D spatial objects », *Computers & Geosciences*, vol. 30, p. 419-428, 2004.