Chapitre 8

Enveloppes convexes, Voronoi et Delaunay

8.1 Calculs d'enveloppes convexes

De manière générale, on veut résoudre

Problème 8.1 *Soit un ensemble $S = \{s_1, s_2, \ldots, s_n\}$ de n points de $\mathbb{R}^d$, calculer le treillis des faces de $Conv(S)$.*

Le tri de n nombres $\{x_1, x_2, \ldots, x_n\}$ se réduit en temps linéaire au calcul de l'enveloppe convexe des n points du plan $\{(x_1, x_1^2), (x_2, x_2^2), \ldots, (x_n, x_n^2)\}$ situés sur la parabole $\{y = x^2\}$. On en déduit

Lemme 8.2 *La complexité du calcul d'enveloppes convexes en dimension supérieure ou égale à 2 admet pour borne inférieure $\Omega(n \log n)$.*

En utilisant le modèle de l'arbre binaire de décision algébrique pour l'analyse des algorithmes (cf. Preparata et Shamos, 1984, pp 101-104), on montre également que la seule sélection des points extrêmes de $Conv(S)$ (sans leurs adjacences) admet pour borne inférieure $\Omega(n \log n)$.

8.1.1 Algorithmes naïfs

Par sélection des points extrêmes

Pour vérifier qu'un point $s \in S$ est extrême il suffit de vérifier qu'il n'est contenu dans aucun des d-simplexes formés sur $S \setminus s$. Il y a $\binom{n-1}{d+1}$ tels simplexes, ce qui induit un algorithme en temps $O(n\binom{n-1}{d+1}) = O(n^{d+2})$.

Par sélection des facettes

Un sous-ensemble de d points de S engendre une facette si son enveloppe affine ne sépare pas les points de S (i.e. est support pour S). On en déduit un algorithme en temps $O(\binom{n}{d}n) = O(n^{d+1})$.

Par propagation à partir d'une facette

Définition 8.3 *L'*angle dièdre *de deux demi-espaces est l'angle compris entre 0 et π formé par les normales aux hyperplans supports tournées vers l'extérieur.*

Soit F une facette de $P = Conv(S)$ et G une facette de F, i.e. une $(d-2)$-face de P. Pour tout $s \in S \setminus F$ on note H_s le demi-espace bordé par G et s et contenant F. Le demi-espace support de la facette adjacente à F le long de G coïncide avec l'hyperplan H_s lorsque celui-ci forme un angle dièdre minimal avec le demi-espace support de F.

Connaissant F et G, on peut donc déterminer la facette adjacente en temps linéaire. En supposant P simplicial (i.e. S en position générale) toute facette de F est déterminée par un sous-ensemble de $d-1$ des d sommets de F. Cela fournit un algorithme en $O(nh)$ – où h est le nombre de facettes de P – pour déterminer toutes les facettes et leurs adjacences. D'après le théorème de la borne supérieure cela fournit un algorithme en $O(n^{\lfloor d/2 \rfloor+1})$.

Il reste à trouver une première facette : prendre le point $s_0 \in S$ de coordonnées minimales pour l'ordre lexicographique, puis l'arête s_0s_1 formant un angle minimal avec sa projection sur l'hyperplan $H_0 = \{x_1 = (s_0)_1\}$, puis le triangle $s_0s_1s_2$ formant un angle minimal avec sa projection sur l'hyperplan contenant s_0s_1 dans le faisceau définit par H_0 et l'hyperplan normal à s_0s_1. De manière générale si on a déterminé une k-face $F_k = Conv(\{s_0, s_1, \ldots, s_k\})$ et un hyperplan support H_k de F_k on obtient $F_{k+1} = Conv(F_k \cup \{s_{k+1}\})$ en cherchant s_{k+1} tel que $Conv(F_k \cup \{s_{k+1}\})$ forme un angle minimal avec sa projection sur H_k. Et on obtient H_{k+1} - contenant s_{k+1} - dans le faisceau définit par H_k et l'hyperplan normal à s_ks_{k+1} contenant s_k. Cette étape prend un temps $O(dn)$ et on obtient finalement un algorithme en $O(n^{\lfloor d/2 \rfloor+1})$ pour la construction de toutes les facettes. Notons que le treillis des faces de P étant coatomique, la connaissance des facettes de P détermine entièrement ce treillis.

On peut remarquer que cet algorithme ne nécessite que des calculs de déterminants pour ordonner selon des angles.

8.1.2 Calcul en 2D

Marche de Jarvis

En deux dimensions l'algorithme précédent s'appelle la *marche de Jarvis* et fournit un algorithme de complexité $O(n^2)$.

Balayage de Graham

Soit o un point intérieur à $P = Conv(S)$ et soit s_0 un point extrême de S, par exemple le point de coordonnées minimales pour l'ordre lexicographique. On considère l'ordre cyclique sur les sommets s_i de S défini par l'angle polaire des demi-droites os_i avec os_0 (en fait l'ordre lexicographique sur la paire (angle, - distance à o)). P est un sous-ordre de cet ordre cyclique obtenu en ôtant récursivement tous les points formant un angle réflexe avec leur successeur et leur prédécesseur. D'un point de vu algorithmique, il suffit de parcourir la liste ordonnée des points à partir de s_o et de supprimer le point courant tant qu'il est le sommet d'un angle réflexe. Chaque sommet n'étant supprimer qu'un fois au plus, P peut être calculé en temps linéaire à partir de cette liste, ce qui fournit un algorithme en temps $O(n \log n)$.

Diviser pour régner

1. On partitionne les points de S en deux selon la médiane de leurs abscisses en temps $O(n)$.

2. On calcule récursivement les enveloppes convexes ECG et ECD de chaque moitié, et on fait leur fusion en temps $O(n)$.

Le temps total $T(n)$ requis pour le calcul de l'enveloppe convexe vérifie ainsi.

$$T(n) = O(n) + T(\lfloor n/2 \rfloor) + T(\lceil n/2 \rceil)$$

D'où $T(n) = O(n \log n)$, ce qui est optimal.

La fusion de ECG et ECD s'obtient en recherchant leurs tangentes, ou *ponts*, supérieure et inférieure. Pour cela on part d'une arête joignant le sommet de ECG (ECD) le plus à droite (gauche) qui est visible par ECD (ECG) puis on marche le long de ECD (ECG) vers le haut et vers le bas pour chacun des deux ponts haut et bas jusqu'à ce que les angles aux extrémités de chaque pont avec les chaînes correspondantes de ECG et ECD soient convexes.

balayage

Le calcul de l'enveloppe convexe de n points $S = S_n = \{s_1, s_2, \ldots, s_n\}$ du plan peut se faire à l'aide d'un algorithme par balayage. Cet algorithme repose sur trois étapes.

1. On commence par trier les n points s_i selon une direction donnée (par exemple l'axe des abscisses. Si plusieurs points ont même abscisse on utilise l'ordre lexicographique sur les couples de coordonnées). On supposera par la suite que les indices des points correspondent à l'ordre croissant.
2. On initialise $Conv(S_3)$ par le triangle $s_1 s_2 s_3$ dans l'ordre circulaire direct.
3. Itération : On construit $Conv(S_i)$ à partir de $Conv(S_{i-1})$ et du point s_i.

Clairement (pourquoi ?) $s_i \notin Conv(S_{i-1})$ et $Conv(S_i)$ s'obtient en supprimant les arêtes vues par s_i et en ajoutant deux arêtes issues de s_i et joignant les deux sommets aux extrémités de la chaîne des arêtes supprimées.

Le coût de parcours des arêtes supprimées peut être reporté (chargé) sur le coût de création de ces arêtes puisque toute arête créée n'est supprimée qu'une fois au plus. À chaque itération 2 arêtes sont créés donc le coût total des itérations est linéaire. En tenant compte du tri initial on obtient ainsi un algorithme en $O(n \log n)$.

Algorithme dynamique

- Dynamic Planar Convex Hull. G. S. Brodal and R. Jacob. FOCS'02. pp 617 - 626. Vancouver, Canada.
"There exists a data structure for the fully dynamic planar convex hull problem supporting INSERT and DELETE in amortized $O(\log n)$ time, and EXTREME POINT QUERY, TANGENT QUERY and NEIGHBORING-POINT QUERY in $O(\log n)$ time, where n denotes the size of the stored set before th operation. The space usage is $O(n)$."

8.1.3 Calcul en 3D

La taille d'une enveloppe convexe de n points en 3D est linéaire. Cela résulte directement du théorème de la borne supérieure ou bien de la formule d'Euler :
$n - A + F = 2$ et $3F \leq 2A \implies A \leq 3n - 6$ et $F \leq 2n - 4$.

Puisque le bord d'un polytope de dimension 3 est homéomorphe à une sphère on peut représenter le treillis des faces d'un tel polytope par une carte "planaire" (en fait sphérique). Dans ce qui suit on supposera que les enveloppes convexes sont représentées par une carte planaire. Une autre possibilité, qui a l'avantage de se généraliser en toute dimension, est de représenter une enveloppe convexe par le treillis de ses faces, c'est-à-dire par le graphe d'incidence de ses faces : chaque face est représentée par un élément qui pointe sur ses facettes et cofacettes.

balayage

Le calcul de l'enveloppe convexe de n points $S = S_n = \{s_1, s_2, \ldots, s_n\}$ de $\mathbb{R}^3$ peut se faire à l'aide d'un algorithme par balayage. Cet algorithme repose sur trois étapes.

1. On commence par trier les n points s_i selon l'ordre lexicographique sur leurs triplets de coordonnées. On supposera par la suite que les indices des points correspondent à l'ordre croissant.
2. On initialise $Conv(S_4)$ par le tétraèdre $s_1s_2s_3s_4$, ou plus précisément par les quatre premiers points non coplanaires, orienté positivement.
3. Itération : On construit $Conv(S_i)$ à partir de $Conv(S_{i-1})$ et du point s_i.

Clairement $s_i \notin Conv(S_{i-1})$ et $Conv(S_i)$ s'obtient en supprimant les faces et arêtes vues par s_i et en ajoutant les faces et arêtes du cône de sommet s_i et joignant le bord des faces de $Conv(S_{i-1})$ supprimées.

Le coût de parcours des faces et arêtes supprimées peut être reporté (chargé) sur le coût de création de ces faces et arêtes. À chaque itération $O(n)$ faces et arêtes peuvent être

créées donc le coût total des itérations est quadratique. En tenant compte du tri initial on obtient un algorithme de complexité $O(n^2)$.

Exercice 8.4 *Donnez un exemple de S pour lequel l'algorithme est un $\Omega(n^2)$.*

Algorithme randomisé incrémental

Il s'agit d'une version randomisée de l'algorithme par balayage. On considère un tétraèdre inclus dans $Conv(S)$, par exemple en prenant quatre points de S affinement indépendants. On ajoute les points de S incrémentalement dans un ordre aléatoire. On note S_i l'ensemble des i premiers points insérés (hormis les 4 points du tétraèdre initial). Soit s_i le i-ième point inséré.

- si $s_i \in Conv(S_{i-1})$, alors $Conv(S_i) = Conv(S_{i-1})$,
- sinon $Conv(S_i)$ s'obtient en supprimant les faces et arêtes vues par s_i et en ajoutant les faces et arêtes du cône de sommet s_i tangent à $Conv(S_{i-1})$.

Pour savoir si $s_i \in Conv(S_{i-1})$ ou pour connaître les faces de $Conv(S_{i-1})$ vues par s_i on maintient un *graphe des conflits* entre les faces de $Conv(S_{i-1})$ et les points de $S \setminus S_{i-1}$: chaque face f pointe vers la liste $S(f)$ des points de $S \setminus S_{i-1}$ qui la voit (i.e. qui ne sont pas contenus dans le demi-espace support de f) et chaque point s de $S \setminus S_{i-1}$ pointe vers la liste $F(s)$ des faces de $Conv(S_{i-1})$ qu'il voit. On a $s_i \in Conv(S_{i-1})$ si et seulement si $F(s_i)$ est vide. Dans le cas contraire la mise à jour du graphe des conflits s'obtient en supprimant s_i et toutes les faces (et leurs pointeurs) de $F(s_i)$ et en ajoutant toutes les faces du cône de s_i ainsi que leur liste de conflits. Pour déterminer la liste des conflits d'une nouvelle face f incidente à une *arête d'horizon* e (i.e. du bord du cône), on remarque que tout point voyant f voyait au moins l'une des deux faces f_1 et f_2 de $Conv(S_{i-1})$ incidentes à e. Il suffit donc de parcourir $S(f_1)$ et $S(f_2)$ pour en extraire $S(f)$: pour chaque sommet s de $S(f_1) \cup S(f_2)$ on test en temps constant si $s \in S(f)$, i.e. si s est dans le demi-espace bordé par le plan support de f et ne contenant pas $Conv(S_i)$.

Analyse

Lemme 8.5 *L'espérance du nombre de faces créées au cours de l'algorithme est linéaire.*

Preuve : Le nombre de faces créées par l'insertion de s_i est égal au degré d_i de s_i dans $Conv(S_i)$. Or, sur l'ensemble des permutations de S dont on a fixé l'*ensemble* S_i des i premiers sommets, chaque sommet $s \in S_i$ est le sommet s_i avec la probabilité $1/i$, d'où

$$E(d_i|S_i) = \frac{1}{i}\sum_{s \in S_i} d(s) \leq \frac{1}{i}(6(i+4) - 12) = O(1)$$

En effet, $\sum_{s \in S_i} d(s)$ vaut 2 fois le nombre d'arêtes de $Conv(S_i)$ et ce dernier nombre est majoré par $3(i+4) - 6$ (voir le début de cette section). On en déduit de manière inconditionnelle $E(d_i) = O(1)$, d'où le résultat par linéarité de l'espérance. □

La mise à jour des listes de conflits $F(.)$ et $S(.)$ à l'instant i prend clairement un temps proportionnel à la somme

– du nombre de listes supprimées, i.e. du nombre de faces supprimées,
– du nombre de listes créées, i.e. du nombre de faces créées,
– du nombre de conflits supprimés dans les listes, et
– du nombre de conflits ajoutés dans les listes.

Puisque le nombre de listes ou conflits supprimés est respectivement inférieur au nombre de listes ou conflits ajoutés, il suffit de comptabiliser ces derniers, ce qui correspond au coût de construction des listes $S(f)$ pour les nouvelles faces f ajoutées à l'instant i.

On construit initialement les listes de conflits des 4 faces du tétraèdre de départ, ce qui prend un temps $O(n)$ par simple inspection de chaque sommet. Il reste finalement à évaluer le coût de construction des $S(f)$ après initialisation. Pour cela on définit $S(e) = S(f_1) \cup S(f_2)$ la liste de conflits d'une arête e incidente à deux faces f_1 et f_2.

D'après ce qui précède le paragraphe d'analyse, le coût de création des $S(f)$ est proportionnel à la somme $\sum_e |S(e)|$ pour toutes les arêtes d'horizon apparaissant au cours de l'algorithme. On veut donc majorer l'espérance de cette somme.

Pour poursuivre l'analyse on définit la *région* associée à un quadruplet de points (p, q, r, s) comme l'union du demi-espace ouvert bordé par p, q, r et ne contenant pas s et du demi-espace ouvert bordé par p, q, s et ne contenant pas r. Les points de S en conflit avec une région R sont par définition les points contenus dans cette zone. On note $S(R)$ l'ensemble de ces points. **On supposera pour simplifier l'exposé que les points de S sont en position générale**.

À chaque arête $e = pq$ de $Conv(S_{i-1})$ on peut associer la région (p, q, r, s) où pqr et qps sont les deux triangles incidents à e dans $Conv(S_{i-1})$. Réciproquement les régions formées sur les points de S_{i-1} et sans conflit avec les points de S_{i-1} (les régions *actives* à l'instant $i-1$) sont précisément obtenues à partir des arêtes de $Conv(S_{i-1})$ comme précédemment indiqué.

Si e est une arête d'horizon de $Conv(S_{i-1})$ pour s_i, alors $S(e)$ est l'ensemble des sommets en conflit avec la région associée à e. Pour majorer $\sum_e |S(e)|$, il suffit donc de majorer $\sum_R |S(R)|$, la somme portant sur l'ensemble de toutes les régions actives au cours de l'algorithme. Par le lemme 13.8 des algorithmes randomisés incrémentaux (voir section 13.3.1), l'espérance de cette somme est majorée par

$$\sum_{i=1}^{n} 4^2 \frac{n-i}{i^2} E(|\mathcal{A}_i|)$$

où $\mathcal{A}_i$ est l'ensemble des régions actives à l'étape i. Dans le cas présent cet ensemble correspond aux arêtes de $Conv(S_i)$, d'où $E(|\mathcal{A}_i|) \leq 3(i+4) - 6$. On en déduit

$$E(\sum_e |S(e)|) = O(n \log n)$$

et par conséquent

Proposition 8.6 *L'enveloppe convexe de n points dans $\mathbb{R}^3$ peut être calculée par un algorithme randomisé en temps moyen $O(n \log n)$.*

8.1.4 Calcul en dimension quelconque

On se place dans le cadre dual du calcul du treillis des faces d'un polytope P donné par une intersection de demi-espaces dans $\mathbb{R}^d$. On s'appuie pour cela sur l'algorithme randomisé de programmation linéaire en temps linéaire due à Seidel [Sei91] et présenté dans la section 7.4.

On suppose que les hyperplans bordant les demi-espaces de P sont en position générale. En conséquence, P est simple et chacun de ses sommets a d arêtes incidentes.

Théorème 8.7 *Soit un entier $d > 3$ fixé. Il existe un algorithme randomisé pour calculer le treillis du polytope intersection de n demi-espaces donnés en position générale en temps moyen $O(n^{\lfloor d/2 \rfloor})$.*

Preuve : Soit $H_1^0, \ldots, H_n^0$ les n hyperplans bordant les demi-espaces $H_1, \ldots, H_n$ donnés. Pour simplifier l'exposé on commence par ajouter les $2d$ demi-espaces supports $H_{1-2d}, H_{2-2d}, \ldots, H_0$ d'un grand cube contenant l'intersection des H_i. Ceci permet de supposer que l'intersection des hyperplans est bornée. Remarquons que l'intersection des H_i est un polytope simple par hypothèse de position générale. Chaque sommet est donc incident à d arêtes et chaque arête est incidente à $d-1$ faces de dimension 2 (une arête peut former une 2-face avec chacune des $d-1$ arêtes partageant l'une fixée de ses extrémités).

On considère que les H_i, $i > 0$, sont dans un ordre aléatoire pour la loi uniforme. On maintient incrémentalement le 2-squelette de $P_i := \bigcap_{-2d<j\leq i} H_j$. Par la remarque précédente, ce 2-squelette a une taille proportionnelle à son nombre de sommets (d est fixé !). Le 2-squelette du cube P_0 est directement calculé en temps constant. Supposons avoir calculé le 2-squelette de P_i pour un certain $i < n$. On détermine si H_{i+1} contient P_i ou non. Pour cela on détermine le sommet p_i de P_i qui maximise la forme linéaire correspondant à H_{i+1}. D'après le théorème 7.15, p_i peut être calculé en temps $O(i)$. Si p_i est dans H_{i+1}, alors H_{i+1} contient P_i et $P_{i+1} = P_i$. Sinon, il faut mettre à jour le 2-squelette de P_{i+1} en coupant la partie de P_i dans le complémentaire H_{i+1}^+ de H_{i+1}. Pour cela on parcourt le 1-squelette de P_i à partir de p_i pour trouver la partie (connexe) contenue dans H_{i+1}^+ et les arêtes $a_1, \ldots, a_k$ coupées par H_{i+1}^0. Les nouveaux sommets de P_{i+1} sont les intersections de H_{i+1}^0 avec les a_j, et les nouvelles arêtes sont les intersections de H_{i+1}^0 avec les 2-faces incidentes aux a_j. Ceci permet de construire le 1-squelette de P_{i+1}. Le 2-squelette est obtenu en temps proportionnel au nombre de sommets par la propriété de coatomicité des polytopes : en identifiant chaque facette avec son hyperplan support on peut associer à chaque sommet (resp. arête) les d (resp. $d-1$) facettes qui le définissent, tandis que les 2-faces correspondent à des sous ensembles de $d-2$ facettes... Puisque chaque sommet ne peut être supprimé qu'une fois après avoir été créé, le coût total de construction des 2-squelettes est proportionnel au coût de créations des sommets. Ce coût est lui-même majoré par le nombre de sommets créés au cours de l'algorithme auquel s'ajoute $\sum_i O(i) = O(n^2)$ pour le calcul des p_i. Soit m_{i+1} le nombre de sommets créés à l'étape $i+1$, c'est-à-dire contenu dans H_{i+1}^0. Fixons les $(i+1)$ premiers demi-espaces, et considérons toutes les permutations commençant par ces $(i+1)$ premiers éléments pris dans un ordre quelconque. Ceci fixe P_{i+1}. Puisque chaque sommet de P_{i+1} appartient à d hyperplans, la probabilité qu'un de ces sommets soit créé à l'étape $i+1$ est au plus

$d/(i+1)$. Par le théorème de la borne supérieure, la valeur moyenne de m_{i+1} est donc un $O(\frac{d}{i+1}(i+1)^{\lfloor d/2 \rfloor}) = O((i+1)^{\lfloor d/2 \rfloor - 1})$. On en déduit que le coût total est

$$\sum_i O((i+1)^{\lfloor d/2 \rfloor - 1}) = O(n^{\lfloor d/2 \rfloor})$$

□

8.2 Diagramme de Voronoi

Soit $S = \{s_1, s_2, \ldots, s_n\}$ un ensemble de n points ou *sites* de $\mathbb{R}^d$. Pour tout $X \subset S$ on considère la région V_X formée des points de $\mathbb{R}^d$ à égale distance des sites de X et strictement plus proches de X que de $S \setminus X$. Les V_X sont des polyèdres (relativement ouverts), car intersections de demi-espaces "médiateurs" de couples de sites, et partitionnent $\mathbb{R}^d$ (pourquoi ?). Cette partition est appelée le *diagramme de Voronoi* de S.

8.2.1 Lien avec les enveloppes supérieures

On note (x, x_{d+1}) un point de $\mathbb{R}^d \times \mathbb{R} = \mathbb{R}^{d+1}$. On considère dans $\mathbb{R}^{d+1}$ le paraboloïde $U := \{x_{d+1} = |x|^2\}$. On relève chaque site s de S en un point $u(s) = (s, |s|^2)$ sur U et on note $T_U(s)$ l'hyperplan tangent à U en $u(s)$ de sorte que
$T_U(s) = \{x_{d+1} = 2\langle s, x\rangle - |s|^2\}$. Enfin, on note $H_U(s)$ le demi-espace bordé par $T_U(s)$ et tourné vers le haut : $H_U(s) = \{x_{d+1} \geq 2\langle s, x\rangle - |s|^2\}$.

Lemme 8.8 *Soit $x \in \mathbb{R}^d$ et $s \in S$ un site, alors la distance algébrique verticale de $T_U(s)$ à $u(x)$ vaut $|x - s|^2$.*

Voir figure 8.1 pour une illustration.

Preuve : le projeté vertical y de x (ou $u(x)$) sur $T_U(s)$ vérifie $y_{d+1} = 2\langle s, x\rangle - |s|^2$. D'où $(u(x) - y)_{d+1} = |x|^2 - 2\langle s, x\rangle + |s|^2 = |x - s|^2$. □

Proposition 8.9 *Le diagramme de Voronoi de S est la projection verticale sur $\mathbb{R}^d$ du bord du polyèdre $\cap_{s \in S} H_U(s)$, enveloppe supérieure des $T_U(s)$.*

Dit autrement, les cellules du diagramme de Voronoi sont les projections des faces de l'enveloppe supérieure. Notons que les faces (possiblement vides) de $\cap_{s \in S} H_U(s)$ sont précisément de la forme $\cap_{s \in X} T_U(s) \bigcap \cap_{s \in S} H_U(s)$ pour $X \subset S$. [1]

1. Chaque $T_U(s)$ est support d'une facette de l'enveloppe supérieure car s est intérieur au polyèdre $\cap_{s' \neq s} H_U(s')$ puisque $s \in int(H_U(s'))$ pour $s' \neq s$. Le treillis d'un polytope P étant coatomique (proposition 6.33), chaque face de P est l'intersection de facettes, i.e. l'intersection avec P de leurs hyperplans supports. On étend sans mal à un polyèdre non borné.

Preuve : Soit $X \subset S$.

$$x \in V_X \Leftrightarrow \forall s_i, s_j \in X, \forall s_k \in S \setminus X : |x - s_i|^2 = |x - s_j|^2 < |x - s_k|^2.$$

Ce qui équivaut à dire d'après le lemme précédent que l'intersection de la verticale en x avec l'enveloppe supérieure des $T_U(s)$ est dans $\cap_{s \in X} T_U(s)$. C'est-à-dire que x est dans la projection verticale de la face de l'enveloppe supérieure, obtenue comme intersection de $\cap_{s \in X} T_U(s)$ avec $\cap_{s \in S} H_U(s)$. □

Notons que cette projection est non dégénérée puisqu'aucun $H_U(s)$ ne contient la direction verticale.

Corollaire 8.10 *Pour d fixé, la complexité totale en nombre de cellules du diagramme de Voronoi de n sites est un $O(n^{\lceil \frac{d}{2} \rceil})$.*

Preuve : D'après le théorème de la borne supérieure (version duale) la complexité d'une intersection de n demi-espaces en dimension $d+1$ est $O(n^{\lfloor \frac{d+1}{2} \rfloor})$. □

En dimension 3 cela donne en particulier une borne supérieure en $O(n^2)$. Cette borne est atteinte pour $2n$ points répartis uniformément pour moitiés sur deux droites non sécantes et non parallèles de $\mathbb{R}^3$.

8.2.2 Calcul en 2D

D'après ce qui précède la taille du diagramme de Voronoi de n sites du plan est linéaire.

Algorithme de Fortune

Intéressant du point de vue historique ?

Diviser pour régner

1. On divise les points de S en deux selon la médiane de leurs abscisses en temps $O(n)$.

2. On calcul récursivement le diagramme de Voronoi de chaque moitié ainsi que l'enveloppe convexe, et on fait la fusion (expliquée si dessous) en temps $O(n)$.

$$T(n) = O(n) + T(\lfloor n/2 \rfloor) + T(\lceil n/2 \rceil)$$

D'où $T(n) = O(n \log n)$, ce qui est optimal.

On note $Vor(S)$ le diagramme de Voronoi de S et $V_X(S)$ la région de Voronoi associée à $X \subset S$ dans $Vor(S)$. On note également G et D les moitiés gauche et droite de S

calculées à l'étape 1. On pose enfin

$$\begin{aligned} U_G &= \{p \,|\, d(p,G) < d(p,D)\} \\ I &= \{p \,|\, d(p,G) = d(p,D)\} \\ U_D &= \{p \,|\, d(p,G) > d(p,D)\} \end{aligned}$$

Lemme 8.11

$$Vor(G) \cap U_G = Vor(S) \cap U_G \text{ et } Vor(D) \cap U_D = Vor(S) \cap U_D.$$

Preuve : pour la gauche :

$$\forall X \in S : p \in V_X(S) \cap U_G \implies X \subset G$$

d'où

$$p \in V_X(S) \cap U_G \Leftrightarrow p \in V_X(G) \cap U_G$$

□

Lemme 8.12 *I est une chaîne (verticalement) monotone simple composée des arêtes de $Vor(S)$ du type $V_{\{g,d\}}(S)$ où $g \in G$ et $d \in D$. Comme g est à gauche de d et que $V_{\{g,d\}}(S)$ est orthogonal à la droite gd, $V_{\{g,d\}}(S)$ n'est pas horizontale et peut être orientée vers le haut de sorte que g (d) est à sa gauche (droite).*

Preuve : Soit $p \in I$. Par définition il existe $g \in G$ et $d \in D$ tels que $d(p,g) = d(p,d) = d(p,S)$, donc $p \in V_{\{g,d\}}(S)$. Soit ℓ une droite horizontale. On veut montrer que ℓ intersecte I en un unique point. Clairement, il existe un point x de ℓ suffisamment à gauche tel que $d(x,G) < d(x,D)$ et il existe un point y de ℓ suffisamment à droite tel que $d(y,G) > d(y,D)$. Donc $f(p) = d(p,G) - d(p,D)$ change de signe sur ℓ et s'annule nécessairement, fournissant ainsi un point q dans $\ell \cap I$. Soit $\mathcal{C}$ le cercle de centre q et de rayon $d(q,G) = d(q,D)$. $\mathcal{C}$ est vide de points de S et toute droite h de séparation de G et D coupe $\mathcal{C}$. Supposons que h est à droite de q. Clairement tout $x \in \ell$ à droite de $\mathcal{C}$ est plus proche de d que de tout point à gauche de h (et hors de $\mathcal{C}$), donc $f(x) > 0$. Si q est le point le plus à gauche sur ℓ vérifiant $f(q) = 0$, c'est donc le seul. Par conséquent $|\ell \cap I| = 1$. □

La fusion des enveloppes convexes de G et D s'obtient à l'aide des deux ponts haut et bas comme décrit section 8.1.2. Les médiatrices de ces ponts supportent nécessairement les segments (infinis) supérieur et inférieur de I. Notons g_0 et d_0 les sites extrémités du pont supérieur. Pour calculer $Vor(S)$ à partir de $Vor(G)$ et $Vor(D)$, et donc pour calculer I, on commence par la partie "supérieure" de la médiatrice I_0 de g_0 et d_0 qui est nécessairement contenue dans $V_{\{g_0\}}(G)$ et dans $V_{\{d_0\}}(D)$. On parcourt en parallèle le bord de ces régions pour trouver le segment de leur bord qui intersecte I_0 le plus haut. Ce segment, que l'on supposera border $V_{\{g_0\}}(G)$ est une région $V_{\{g_0,g_1\}}(G)$. Le point $i_1 = V_{\{g_0,g_1\}}(G) \cap I$ vérifie $d(i_1,g_0) = d(i_1,g_1) = d(i_1,d_0) = d(i_1,S)$. Donc i_1 est un sommet de I et le segment suivant sur I ne peut être que $V_{\{g_1,d_0\}}(S)$. On poursuit sur ce segment jusqu'à intersecter un segment de $V_{\{g_1\}}(G)$ ou $V_{\{d_0\}}(D)$. On itère ce procédé jusqu'à rencontrer la médiatrice du pont inférieur. Le coût de construction de I est de l'ordre de la taille totale des régions parcourues qui est elle même majorée par la somme des tailles de $Vor(G)$ et $Vor(D)$ et est donc linéaire.

8.2.3 Variantes du diagramme de Voronoi

Diagramme de puissance. Rappelons que par définition la *puissance* d'un point x par rapport à une sphère $S(c, r)$ de centre c et de rayon r vaut $|x - c|^2 - r^2$ et que l'ensemble des points ayant même puissance relativement à deux sphères de centres distincts est un hyperplan (dit *radical*). Étant donné un ensemble de sphères de $\mathbb{R}^d$, ou de manière équivalente de points pondérés, le *diagramme de puissance* de ces sphères est la partition de $\mathbb{R}^d$ en régions telle que dans chaque région tout point ait une même puissance par rapport à un sous-ensemble fixe de sphères et une puissance plus grande par rapport aux autres sphères. Comme pour le diagramme usuel, les régions sont des polyèdres (intersections de demi-espaces bordés par des hyperplans radicaux). De manière analogue au diagramme de Voronoi usuel, un diagramme de puissance s'obtient comme projection de l'enveloppe supérieure d'hyperplans obtenus en translatant les hyperplans tangents aux relevés des sites sur le paraboloïde U (cf. section 8.2.1). Certains sites peuvent cependant avoir des régions vides contrairement au cas du diagramme usuel.

Diagramme d'ordre k. Soit une famille S de n sites de $\mathbb{R}^d$ et un entier $k < n$. On associe à chaque k-set (sous-ensemble de taille k) $T \subset S$ la cellule formées de l'ensemble des points qui sont plus proches des sites de T que de $S \setminus T$, i.e. $\{x \mid \forall t \in T, \forall s \in S \setminus T : d(x, t) \leq (x, s)$. Les intérieurs relatifs des intersections de ces cellules constituent une partition de R^d appelée *diagramme de Voronoi d'ordre k*. On montre aisément que c'est la projection verticale du k-ième *niveau* de l'arrangement des $T_U(s)$ (cf. section 8.2.1 et figure 8.1) relativement à la direction verticale vers le bas : c'est l'ensemble des cellules

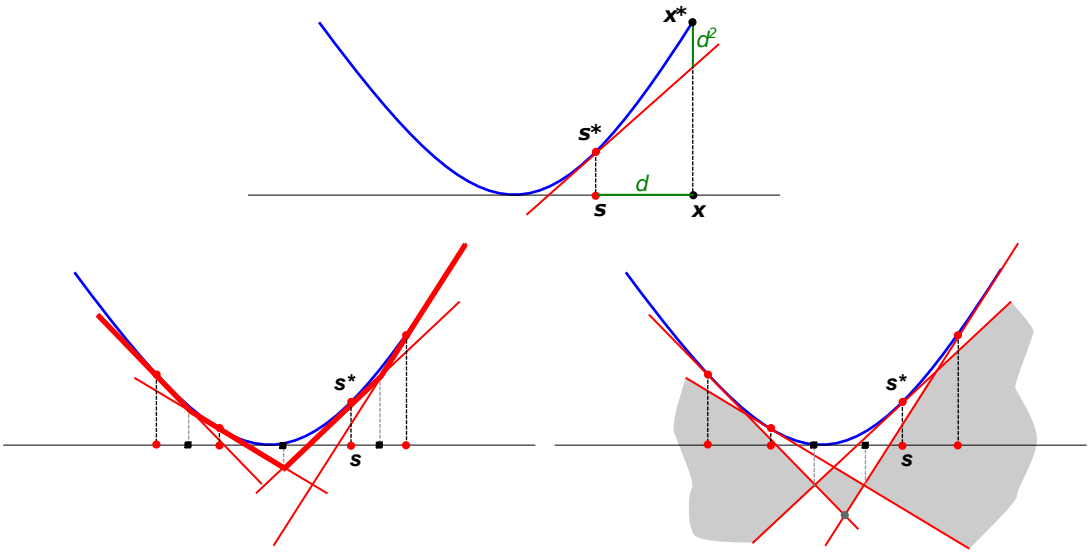

FIGURE 8.1 – En haut : Lien entre la distance entre les sites s et x et la distance verticale du relevé x^* sur le paraboloïde au plan tangent à s^*. Ici $U(s)$ est noté s^*. En bas à gauche : Le diagramme de Voronoi des points sur la droite horizontale apparaît comme la projection de l'enveloppe supérieure des plans tangents aux relevés. En bas à droite : Le diagramme d'ordre 2 du même ensemble de points.

de cet arrangement qui ont k hyperplans au dessus d'elles (au sens large). Le diagramme d'ordre $k = n-1$ s'appelle encore le *diagramme du voisin le plus loin.* La cellule d'un site (i.e. des $n-1$ sites complémentaires) est l'ensemble des centres des sphères contenant tous les autre sites. C'est encore la projection verticale de l'enveloppe *inférieure* des $T_U(s)$.

Autres diagrammes. On peut encore considérer des diagrammes de Voronoi pour des distances autres que la distance Euclidienne. On peut également remplacer les sites ponctuels par des objets plus variés comme des segments, des convexes, des droites, etc. . .La littérature sur ce sujet est sans fin [AK00].

8.3 Triangulation de Delaunay

Soit $S = \{s_1, s_2, \ldots, s_n\}$ un ensemble de n points ou *sites* de $\mathbb{R}^d$. On considère les $X \subset S$ tels qu'il existe une boule fermée B_X, circonscrite à X et vérifiant $B_X \cap S = X$. On pose $D_X = Conv(X)$. Le *diagramme de Delaunay* de S est la collection des D_X ainsi formés. Le bord d'une boule B_X comme ci-dessus s'appelle une *sphère de Delaunay.*

8.3.1 Lien avec les enveloppes convexes

Lemme 8.13 *Soit $X = \{x_1, x_2, \ldots, x_{d+1}\}$ un ensemble de $d+1$ points de $\mathbb{R}^d$ affinement indépendants. Alors, il existe une unique sphère $\mathcal{S}(X)$ circonscrite à ces points et elle a pour équation*

$$\left\| \begin{array}{cccc} |x|^2 & |x_1|^2 & \ldots & |x_{d+1}|^2 \\ x & x_1 & \ldots & x_{d+1} \\ 1 & 1 & \ldots & 1 \end{array} \right\| = 0$$

Preuve : En développant ce déterminant selon la première colonne on trouve une équation de la forme $\alpha|x|^2 + \langle\beta|x\rangle + \gamma = 0$ avec $\alpha = \left\| \begin{array}{ccc} x_1 & \ldots & x_{d+1} \\ 1 & \ldots & 1 \end{array} \right\|$ non nul puisque les points de X sont affinement indépendants. Cette équation est donc celle d'une sphère. Cette sphère est circonscrite aux x_i puisque ce déterminant s'annule pour $x = x_i$, pour tout i.

Par ailleurs, si $|x|^2 + \langle\beta|x\rangle + \gamma = 0$ et $|x|^2 + \langle\beta'|x\rangle + \gamma' = 0$ sont les équations de deux sphères circonscrites à X, alors $\langle\beta - \beta'|x\rangle + \gamma - \gamma'$ s'annule sur X ce qui implique $\beta = \beta'$ et $\gamma = \gamma'$ puisque X n'est contenu dans aucun hyperplan.

□

On rappelle la notation $u(x) = (x, |x|^2)$ de la section 8.2.1. Soit $X = \{x_1, x_2, \ldots, x_{d+1}\}$ un ensemble de $d+1$ points affinement indépendants. On note $H(X)$ l'unique hyperplan contenant
$u(X) = \{u(x_1), u(x_2), \ldots, u(x_{d+1})\}$. On déduit du lemme précèdent que

$$x \in \mathcal{S}(X) \Leftrightarrow u(x) \in H(X).$$

Plus précisément si $\mathcal{B}(X)$ est la boule fermée bordée par $\mathcal{S}(X)$ et $Int\,\mathcal{B}(X)$ est son intérieur, on vérifie que

Lemme 8.14

$$x \in Int\,\mathcal{B}(X) \Leftrightarrow u(x) \text{ est en dessous de } H(X)$$

et

$$x \notin \mathcal{B}(X) \Leftrightarrow u(x) \text{ est au dessus de } H(X)$$

Corollaire 8.15 *Soit $X \subset S$. Il existe une boule fermée B_X circonscrite à X et vérifiant $B_X \cap S = X$ si et seulement s'il existe un hyperplan H_X de $\mathbb{R}^{d+1}$ tel que $u(S)$ est au dessus de H_X et $H_X \cap u(S) = u(X)$.*

Preuve : $\Rightarrow$ Soit Y un ensemble de $d+1$ points de B_X affinement indépendants. D'après ce qui précède on peut prendre $H_X = H(Y)$.
$\Leftarrow$ Soit Y un ensemble de $d+1$ points de $H_X \cap U$ tels que leurs projections soit affinement indépendantes. On peut poser poser $B_X = \mathcal{B}(Y)$. □

Donc D_X est une cellule de Delaunay si et seulement s'il existe un hyperplan non vertical au dessous de $u(S)$ et qui contient précisément $u(X)$, ce qui équivaut encore à dire que $Conv(u(X))$ est une face de $Conv(u(S))$ possédant un demi-espace support tourné vers le haut. En appelant *enveloppe convexe inférieure* la collection des faces d'un polytope qui possèdent un demi-espace support tourné vers le haut, on obtient finalement,

Proposition 8.16 *$Del(S)$ est la projection verticale de l'enveloppe convexe inférieure de $u(S)$. En particulier, $Del(S)$ constitue une subdivision polytopale*[2] *de $Conv(S)$.*

Notons que cette projection est non dégénérée puisque chaque face de l'enveloppe convexe inférieure est contenue dans un hyperplan non vertical.

Corollaire 8.17 *Pour d fixé, la complexité totale en nombre de cellules du diagramme de Delaunay de n sites est un $O(n^{\lceil \frac{d}{2} \rceil})$.*

On dira que S est en *position générale* si $d+1$ points quelconques de S sont affinement libres et si $d+2$ points de S ne peuvent être cosphériques. En particulier, il passe une unique sphère par $d+1$ points quelconques de S.

Remarque 8.18 *Si S est en position générale, les D_X sont des simplexes. Dans ce cas $Del(S)$ est une triangulation de $Conv(S)$.*

2. Une subdivision polytopale d'un espace est un complexe polytopal dont cet espace est l'espace total. Un complexe polytopal est une collection de polytopes telle que toute face de ces polytopes est dans la collection et telle que deux polytopes de la collection s'intersectent selon une face commune (possiblement vide).

Exercice 8.19 *Montrer un résultat analogue au lemme 8.13 lorsqu'on suppose seulement* $\dim\ aff(X) = d-1$ *et que les points de* X *ne sont pas cocycliques (on trouve une équation qui se ramène à celle d'un hyperplan).*

Exercice 8.20 *Démontrer le lemme 8.14.*

En deux dimensions, la liste ordonnée des angles des triangles de Delaunay est maximale pour l'ordre lexicographique sur toutes les triangulations de S.

Une arête est dite *localement de Delaunay* si la sphère circonscrite à l'un de ses deux triangles incidents ne contient pas le troisième sommet de l'autre triangle incident. Si une arête n'est pas localement de Delaunay alors l'arête obtenue par *flip* est localement de Delaunay.

Proposition 8.21 *Une triangulation est de Delaunay si et seulement si toutes ses arêtes sont localement de Delaunay.*

8.3.2 Dualité entre les diagrammes de Delaunay et de Voronoi

Soit S un ensemble fini de points de $\mathbb{R}^d$.

Proposition 8.22 *Pour tout* $X \subset S$, $V_X \in Vor(S) \Leftrightarrow D_X \in Del(S)$.

Preuve : $V_X \neq \emptyset \Leftrightarrow \exists y\ (\in V_X)$ tel que y est le centre d'une sphère de Delaunay pour $X \Leftrightarrow D_X \in Del(S)$. □

On note $\bar{V}_X$ l'adhérence de V_X. Comme V_X est l'intérieur relatif d'un polytope, $\bar{V}_X$ désigne l'union de V_X avec ses faces.

Théorème 8.23 *Il existe une dualité entre le diagramme de Voronoi et le diagramme de Delaunay de* S *qui associe toute cellule* V_X *de Voronoi à la cellule* D_X *de Delaunay de sorte que*

$$\dim V_X = i \Leftrightarrow \dim D_X = d - i$$

$$\bar{V}_X \subset \bar{V}_Y \Leftrightarrow D_Y \subset D_X.$$

Preuve : On donne ici une preuve s'appuyant sur la dualité des cônes polyédriques et sur le lien entre les diagrammes de Voronoi ou Delaunay et la projection de certains polyèdres.

Soit S un ensemble de sites de $\mathbb{R}^d$. On considère à nouveau le paraboloïde U et le relèvement $u(x) = (x, |x|^2)$ dans $\mathbb{R}^{d+1}$. Par la proposition 8.9, le diagramme de Voronoi de S est la projection verticale sur $\mathbb{R}^d$ du bord du polyèdre $P = \cap_{s\in S} H_U(s)$. Soit $P' = e_{d+2} + P$ le translaté de P par e_{d+2} dans $\mathbb{R}^{d+2}$. On considère le cône C sur P' de sommet $\mathbf{0}$. Rappelons que $H_U(s) = \{2\langle s, x\rangle - x_{d+1} - |s|^2 \leq 0\}$, de sorte que C est l'intersection des demi-espaces vectoriels $\{h_s(x, x_{d+1}, x_{d+2}) \leq 0\}$, $s \in S$, où $h_s(x, x_{d+1}, x_{d+2}) = 2\langle s, x\rangle - x_{d+1} - |s|^2 x_{d+2}$.

Le lemme 6.52 indique que les faces de P sont en correspondance avec les faces de C qui intersectent l'hyperplan $E = \{x_{d+2} = 1\}$. Par la proposition 6.50, l'intérieur relatif de ces faces est de la forme

$$intF = \{z \mid \forall s' \in S', \forall s'' \in S'' : h_{s'}(z) = 0 \text{ et } h_{s''}(z) < 0\}$$

pour une partition $S' \cup S''$ de S telle que $E \cap intF$ est non vide (cf. exercice ci-dessous), c'est-à-dire telle qu'il existe un point $(x, x_{d+1}, 1)$ de E vérifiant

$$\forall s' \in S', \forall s'' \in S'' : 2\langle s', x\rangle - |s'|^2 = x_{d+1} \text{ et } 2\langle s'', x\rangle - |s''|^2 < x_{d+1}$$

Ce qui s'écrit encore, en notant $g_x(y, y_{d+1}) = 2\langle y, x\rangle - y_{d+1}$:

$$\forall s' \in S', \forall s'' \in S'' : g_x(u(s')) = x_{d+1} \text{ et } g_x(u(s'')) < x_{d+1}$$

On en déduit que le demi-espace $\{g_x \leq x_{d+1}\}$ de $\mathbb{R}^{d+1}$ est support pour la face $Conv(u(S'))$ du polytope $Q = Conv(u(S))$. On note de plus que ce demi-espace est tourné vers le haut (le coefficient de u_{d+1} dans h_x est -1). Réciproquement, toute face de Q admettant un hyperplan support tourné vers le haut, donc de la forme $H = \{y_{d+1} \geq \langle a, y\rangle - b\}$, correspond à une partition $S' \cup S''$ comme ci-dessus avec $S' = S \cap H$, puisque

$$\forall s' \in S', \forall s'' \in S'' : g_a(u(s')) = b \text{ et } g_a(u(s'')) < b$$

Par la proposition 6.54, le cône C^* dual de C est l'enveloppe conique des vecteurs $v_s = (2s, -1, -|s|^2), s \in S$. De plus, par la proposition 6.58 le treillis des faces de C est opposé au treillis des faces de C' et par le lemme 6.56 une face F comme ci-dessus a pour duale la face $F^\#$, enveloppe conique des v_s, $s \in S'$. Mais en appliquant l'isomorphisme linéaire de matrice

$$M = \begin{pmatrix} \frac{1}{2}I_d & 0 & 0 \\ 0 & 0 & -1 \\ 0 & -1 & 0 \end{pmatrix}$$

aux vecteurs v_s, on déduit que C^* est isomorphe à l'enveloppe conique des vecteurs $Mv_s = (s, |s^2|, 1) = (u(s), 1), s \in S$. Par le lemme 6.52, les faces de C^* sont donc en correspondance avec les faces de l'enveloppe convexe Q des points $u(s), s \in S$. Cette correspondance envoie la face $F^\#$ de C^* sur la face $Conv(u(S'))$ de Q.

Pour conclure, les cellules de Voronoi de dimension k correspondent aux faces de P de même dimension qui correspondent aux faces de C de dimension $k+1$ intersectant E. Ces faces sont elles mêmes en correspondance avec certaines faces de dimensions $d + 2 - (k + 1) = d - k + 1$ de C^* qui correspondent précisément aux faces de Q de dimension $d - k$ possédant un hyperplan support tourné vers le haut. Ces dernières se projettent sur les cellules de Delaunay de même dimension par la proposition 8.16. □

Exercice 8.24 *Soit C un cône et E un hyperplan ne contenant pas l'origine. Montrer que $C \cap E$ est non vide si et seulement si E intersecte l'intérieur relatif de C.*

8.3.3 Algorithme randomisé incrémental pour Delaunay

1. On commence par calculer un grand triangle contenant S.

2. On calcul incrémentalement $Del(S_r)$ en insérant s_r dans $Del(S_{r-1})$.

Après avoir trouvé le triangle Δ de $Del(S_r)$ contenant s_r on étoile ce triangle à partir s_r. Si s_r est sur une arête alors on supprime cette arête et on étoile le quadrilatère restant à partir s_r. Les 3 (ou 4) arêtes incidentes à s_r sont nécessairement (localement) de Delaunay : on peut trouver une sphère circonscrite à ces arêtes et incluse dans la sphère (initialement vide) de Δ. Seules les arêtes de Δ peuvent ne plus être localement de Delaunay. On flippe les arêtes qui ne le seraient plus, ce qui crée autant d'arêtes incidentes à s_r et deux fois plus de nouveaux triangles. Les arêtes opposées à s_r dans ces nouveaux triangles peuvent ne plus être localement de Delaunay et on continu ainsi jusqu'à ce que toutes les arêtes soient localement de Delaunay. Le nombre de triangles créés est au plus $2d_r - 3$ où d_r est le degré de s_r dans $Del(S_r)$.

Afin de localiser efficacement s_r dans $Del(S_{r-1})$ – i.e. de trouver le triangle de $Del(S_{r-1})$ qui contient s_r – on maintient une structure de recherche sous forme d'un DAG dont les noeuds sont les triangles créés durant l'algorithme et les feuilles sont les triangles de la triangulation courante. Chaque triangle pointe vers les triangles qui le remplacent et l'intersectent. Le degré sortant maximal dans ce DAG est donc 3.

Le coût total de l'algorithme est proportionnel au nombre de triangles créés plus le nombre de triangles détruits plus la somme des longueurs des chemins de recherche dans le DAG. Notons que le nombre de triangles détruits est inférieur au nombre de triangles créés. Une borne sur ce nombre est donnée par le lemme suivant.

Lemme 8.25 *Le nombre moyen de triangles créés durant l'algorithme est majoré par* $9n + 1$.

Preuve : On note $\mathcal{A}(S_r)$ le nombre de triangles de $Del(S_r)$. D'après ce qui précède le nombre de triangles créés à l'étape r est majoré par $2d_r - 3$. À S_r fixé, s_r est l'un quelconque des points de S_r de manière équiprobable donc

$$E(|\mathcal{A}(S_r) \setminus \mathcal{A}(S_{r-1})| \mid S_r) \leq \frac{1}{r} \sum_{s \in S_r} (2d(s) - 3)$$

Or $Del(S_r)$ étant une triangulation sur $r + 3$ sommets bordée par un triangle on a $|\mathcal{A}(S_r)| = 3(r + 3) - 6 = 3r + 3$. Si on omet les 3 sommets du triangle initial qui ont degré deux au moins on obtient

$$\sum_{s \in S_r} d(s) \leq 2|\mathcal{A}(S_r)| - 6 = 2(3r + 3) - 6 = 6r.$$

Par conséquent le nombre moyen de triangles créés à l'étape r est majoré par 9. Si on tient compte du triangle initial on peut conclure par le résultat annoncé. □

Il reste à estimer la somme des longueurs des chemins de recherche dans le DAG. Le chemin de recherche de s_r visite tous les triangles créés avant l'étape r et contenant s_r.

Chaque noeud ayant au plus trois enfants, ceci représente bien le coût de la recherche à un facteur 3 près. Si on compte à part le triangle contenant s_r dans $\mathcal{A}(S_{r-1})$, le coût de la recherche est proportionnel à 1 plus le nombre de triangles créés *et* détruits avant l'étape r et contenant s_r. Ces triangles étaient soient des triangles de Delaunay à une étape antérieure soit des triangles intermédiaires créés et détruits durant une même étape. Dans ce dernier cas le triangle adjacent qui a servi au flip était effectivement un triangle de Delaunay et son cercle circonscrit contenait le triangle détruit et donc s_r. On peut ainsi reporter dans tous les cas le coût de visite d'un triangle dans le DAG sur un triangle de Delaunay dont le cercle circonscrit contenait s_r. Notons que ces triangles de report sont tous distincts. Un triangle de Delaunay dont le cercle circonscrit contient un site est dit *en conflit* avec ce site. Le coût total de la recherche est donc borné par n (1 par site) plus le nombre total de tous les conflits entre des triangles de Delaunay créés (et détruits) au cours de l'algorithme et des sites insérés par la suite. Par le lemme 13.8, ce nombre est borné en moyenne par

$$\sum_{r=1}^{n} 3^2 \frac{n-r}{r^2} E(|\mathcal{A}(S_r)|).$$

Comme $|\mathcal{A}(S_r)| = O(r)$, on en déduit que ce coût moyen est un $O(n \log n)$.

Proposition 8.26 *La triangulation de Delaunay de n sites dans le plan peut être calculée par un algorithme randomisé en temps moyen $O(n \log n)$.*