

Triangulation de Delaunay

Louis Cohen & Léo Noël-Baron

École Normale Supérieure de Rennes

28 Octobre 2014

Résumé

Comment paver le plan de triangles à partir d'un ensemble de points donnés? La triangulation de Delaunay est une réponse à cette problématique, que nous étudierons avant d'y appliquer une adaptation du jeu de la vie.

MOTS-CLÉS — triangles, Delaunay, jeu de la vie

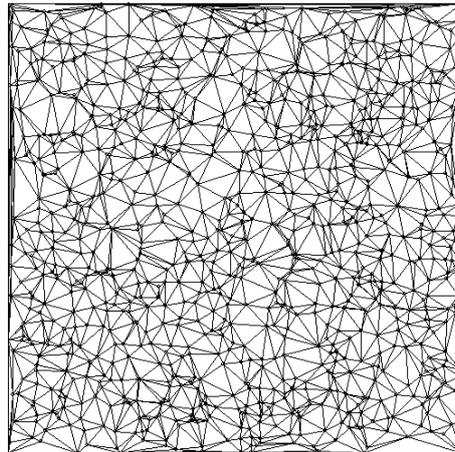


FIGURE 1 – Une triangulation de Delaunay

1 Triangulation de Delaunay

1.1 Principe de base

1.1.1 Définitions préliminaires

Triangulation — Décomposition d'une forme (ici le rectangle de la fenêtre graphique) en un ensemble de triangles.

Triangle le plus régulier possible — Triangle le plus proche possible d'un triangle équilatéral (*i.e.* dont le cercle circonscrit ne contient strictement aucun point de l'ensemble à trianguler).

Triangulation de Delaunay — Méthode de triangulation produisant un ensemble de triangles les plus réguliers possibles, comme en Figure 1.

1.1.2 Méthode générale

Nous allons procéder de manière itérative, c'est-à-dire partir d'une triangulation du plan (sous forme d'une liste l_0 de triangles) de n points pour construire une triangulation de $(n+1)$ points.

Pour commencer on ne considère que les points extrémaux de la fenêtre : $(0, 0)$, $(x_{\max}, 0)$, (x_{\max}, y_{\max}) et $(0, y_{\max})$ (les variables x_{\max} et y_{\max} correspondant aux dimensions de la fenêtre), et une triangulation triviale de cet ensemble, ici t_0 défini par $(0, 0)$, $(x_{\max}, 0)$ et (x_{\max}, y_{\max}) et t_1 défini par $(0, 0)$, $(0, y_{\max})$ et (x_{\max}, y_{\max}) .

Par la suite, à chaque étape, on procède comme suit pour ajouter un point p :

- détermination de l'enveloppe convexe de p (cette étape est détaillée par la suite) sous la forme d'une liste l_1 de triangles ;
- retrait de cette liste de triangles de la liste de base (l_0 devient $l_0 \setminus l_1$) ;
- conversion de cette liste de triangles en une liste de segments $(a_0, a_1), (a_1, a_2), \dots, (a_{n-1}, a_n)$ formant l'enveloppe convexe de p (de même, développée dans la suite) ;
- création d'une liste l_2 de nouveaux triangles à partir de la liste de segment, les triangles en question étant $(a_0, a_1, p), (a_1, a_2, p), \dots, (a_{n-1}, a_n, p)$;
- ajout de l_2 à l_0 .

1.2 Détail des fonctions

`in_circle t p`

Cette fonction vérifie si un point p est dans le cercle circonscrit à un triangle $t = (p_1, p_2, p_3)$, ceci au moyen du calcul d'un déterminant, que nous ne développerons pas dans ce rapport.

`border t_set`

Calcule l'enveloppe convexe d'un point dans le maillage. Pour cela, on passe en paramètre la liste `t_set` des triangles t tels que `in_circle t p = true` (comme dans la Figure 2 : le point p appartient aux cercles des triangles t_1 et t_2) et la fonction transforme cette liste de triangles en une liste de segments (couples de points). Il faut ensuite retirer les segments à l'intérieur de cette enveloppe pour n'en garder que la bordure ; cela revient à supprimer les segments apparaissant deux fois dans la liste (si un segment apparaît deux fois, il appartient à deux triangle de l'enveloppe et est donc intérieur à l'enveloppe).

Dans la Figure 3, le segment s_1 appartient à la fois au triangle t_0 et au triangle t_1 ; il n'apparaîtra pas dans la liste renvoyée par `border`.

`add_point t_set p`

Ajoute un point p au maillage actuel `t_set`. La fonction calcule la liste de segments formant l'enveloppe convexe de p (grâce à `border`), supprime les triangles contenus à l'intérieur de cette enveloppe pour obtenir un résultat analogue à celui de la Figure 4, et reforme les triangles de la Figure 5. Ceux-ci sont définis par les points p_1 , p_2 et p , où p_1 et p_2 sont les points d'une arête de l'enveloppe convexe (ce qui donne pour la Figure 5 les triangles t_2, t_3, t_4 et t_5).

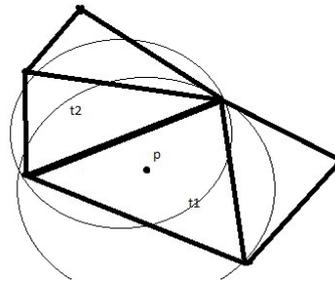


FIGURE 2 – Détermination de l'enveloppe convexe

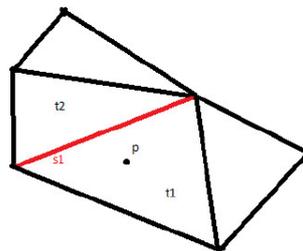


FIGURE 3 – repérage des segments intérieurs à la bordure

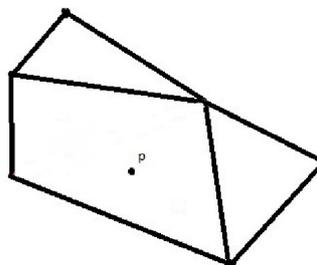


FIGURE 4 – Suppression des triangles internes

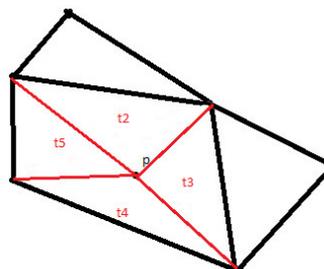


FIGURE 5 – Création des nouveaux triangles

2 Jeu de la vie

2.1 Principe général

Le jeu de la vie est un automate cellulaire, c'est-à-dire un automate constitué de cellules dont l'état évolue simultanément en fonction du temps selon des règles fixées à l'avance.

2.1.1 Règles classiques

Les règles du jeu de la vie classique, dans la version initialement proposée par John Conway sur un damier de cases carrées, sont, pour le nouvel état d'une cellule à un instant $n + 1$, une fonction du nombre de ses voisines vivantes à l'instant n :

voisines vivantes	état suivant
0 ou 1	la cellule meurt
2	la cellule reste dans le même état
3	la cellule vit ou naît
4 ou plus	la cellule meurt

REMARQUE — *Étant donné que nous travaillons ici avec des triangles, il faut modifier ces règles.*

2.1.2 Nouvelles règles

Il faut d'abord décider de ceux que l'on considérera comme voisins d'un triangle t . Différents choix sont possibles dans le cas de triangles :

- les triangles possédant un point en commun avec t ;
- les triangles possédant un côté en commun avec t .

Ces deux options sont illustrées en Figures 6 et 7, où les voisins sont à chaque fois coloriés en rouge.

La deuxième solution a été retenue pour des raisons de lisibilité du graphe, ainsi que de prévisibilité de l'évolution et des changements que l'utilisateur peut effectuer ; en effet, un des buts de ce projet était de créer une interface facile d'accès à l'utilisateur. Si les conséquences de chaque action de celui-ci étaient trop complexes, la lisibilité en souffrirait.

Il convient ensuite de créer une nouvelle fonction de changement d'état. Le choix a été fait d'une fonction très semblable à celle d'origine :

voisines vivantes	état suivant
0	la cellule meurt
1	la cellule reste dans le même état
2	la cellule vit ou naît
3	la cellule meurt

REMARQUE — *On remarquera que le nombre de voisins d'un triangle est exactement de 3, car il est impossible d'avoir plus d'un triangle partageant un côté d'un autre triangle, du fait de la méthode de création des triangles. Le cas où un point est sur un segment déjà présent n'est pas valide et donc refusé dans l'algorithme.*

2.2 Gestion du jeu de la vie

On utilise des tables de hachage, structure de données permettant d'associer à un élément quelconque un ensemble d'autres éléments. Ici, une table `bound` associe à un côté les deux triangles l'ayant en commun, et la table `life` fait correspondre à chaque triangle son état dans le jeu (vivant ou mort).

```
one_move t_set bound life
```

Il s'agit de la fonction principale, qui fait avancer le jeu de la vie d'une unité temporelle. Elle crée une nouvelle table `life2` qu'elle remplit par récurrence sur `t_set` comme suit (où `t` est le premier triangle de `t_set`) :

- à partir des côtés `s1`, `s2` et `s3` de `t`, `bound` permet d'obtenir `t1`, `t2` et `t3`, les voisins de `t` ;
- `life` donne ensuite les états de ces voisins ;
- la fonction calcule alors le nouvel état de `t` et le stocke dans `life2`.

Pour permettre une meilleure compréhension du jeu de la vie (et de la même manière, de la triangulation de Delaunay), nous avons principalement travaillé une interface utilisateur simple dans ce projet, permettant de contrôler l'évolution du jeu de la vie et de modifier à la souris la triangulation ou l'état d'une cellule.

2.3 Intérêt

2.3.1 Interface simplifiée

Une interface graphique, même simple, est un atout majeur pour la compréhension de tout type de projet ou de problématique en ce qu'elle permet de visualiser au plus près les mécanismes mis en jeu et, par exemple, le fonctionnement d'un algorithme. Elle peut donc être réutilisée pour un travail futur sur Delaunay ou le jeu de la vie (en une semaine, poser des bases pour une éventuelle suite peut être intéressant).

2.3.2 Le jeu de la vie

La question se pose de savoir comment le jeu de la vie, aussi connu que simple, peut se comporter une fois adapté à un environnement différent et plus complexe que la grille carrée de base. La topologie d'une triangulation de Delaunay permet d'envisager des configurations intéressantes, et peut-être même des figures comparables à celles du jeu original.

De plus, l'aspect ludique de l'interface interactive laisse envisager d'utiliser cette présentation dans une optique éducative pour des personnes ayant peu de contacts avec l'informatique. La facilité de manipulation permet d'influencer le jeu de manière palpable et aisée, et ainsi de mieux appréhender son évolution à tout instant. La compréhension est (comme dit précédemment) largement aidée par le choix de règles simples et visuelles, avec un nombre de voisins limité.

2.3.3 Le jeu dans la triangulation

Toujours dans une perspective didactique, l'introduction d'un « jeu » dans un problème aussi hermétique que celui de la triangulation de Delaunay, donne un angle d'approche ludique

et original au premier abord.

Mais qui plus est, l'aspect imprévisible des changements impliqués par l'ajout d'un point dans la triangulation pourrait permettre une visualisation de l'adaptation des structures du jeu aux modifications de l'environnement. En effet, l'adaptation du milieu à l'ajout d'un point laisse à penser qu'on pourrait ajouter une dimension supplémentaire au jeu de la vie en rajoutant des règles sur l'évolution du terrain. La principale limitation du jeu actuel est qu'il modifie les « ressources » (cellules vivantes ou mortes) présentes, mais pas le milieu en lui-même; on pourrait par exemple penser à rajouter un point aléatoirement dans un triangle quand celui-ci est entouré de trois triangles vivants (et mettre cela sur le compte de la détérioration du terrain...), ou imaginer d'autres règles dans cette idée.

Il ne semble pas déraisonnable d'envisager des modélisations intéressantes basées sur ce principe, bien que les réflexions présentées ici soient hors contexte.

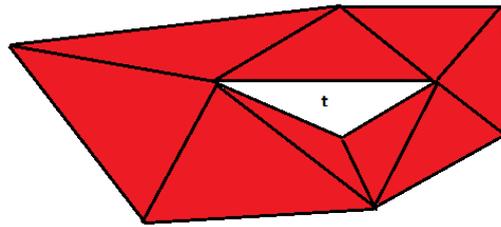


FIGURE 6 – Les voisins possédant au moins un point en commun

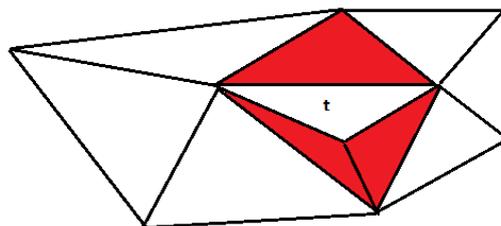


FIGURE 7 – Les voisins possédant un côté entier en commun

A Annexe : auto-évaluation

Points forts

- Travail en groupe positif, utilisation judicieuse de la forge
- Tentative d'originalité
- Sortie de l'idée de recherche pure et tentative d'application plus didactique

Points faibles

- Mauvaise gestion du temps : projet trop ambitieux
- Application ne marchant pas à la date de rendu, projet presque invalide

Améliorations possibles

- Approfondissement de l'idée d'évolution de l'environnement
- Modélisation de phénomènes réels
- Application réelle à l'enseignement (?)

Difficultés à anticiper

- Gestion d'un projet sur un très court laps de temps
- Meilleure anticipation des problèmes possibles