Calcul de l'expansion d'un polygone

D. Legland

17 juin 2009

Résumé

Pour certaines applications géométriques, on a besoin de déterminer un polygone situé à une certaine distance d'un autre polygone. Ce document décrit brièvement la méthode utilisée pour la bibliothèque Matlab « geom2d », et donne quelques exemples de résultats sur des polygones simplifiés ainsi que sur des polygones issus de cas réels.

Table des matières

1 Introduction

Un problème rencontré fréquemment est de déterminer un polygone qui soit « à une distance donnée » d'un autre polygone. On peut avoir deux approches.

On peut définir la forme parallèle à un polygone comme l'ensemble des points situés à une distance donnée de la frontière du polygone. En général, on distingue une distance positive si on va vers l'extérieur du polygone, et une distance négative si on va vers l'intérieur du polygone. Le résultat est un ensemble de courbes, composées de segments de droite et d'arcs de cercle. Ce type de forme est appelé « Buffer » avec les systèmes d'information géographique.

Une autre approche est de ne garder pour le résultat que des segments de droite parallèles aux segments d'origine (Figure 1). Le résultats est un polygone ou un ensemble de polygones. Des cas pathologiques peuvent cependant apparaître près des sommets dont les arêtes voisines sont quasiment parallèles : erreur sur le calcul de l'intersection, ou résultat très pointu avec des nouveaux sommets situés à une grande distance du polygone d'origine.

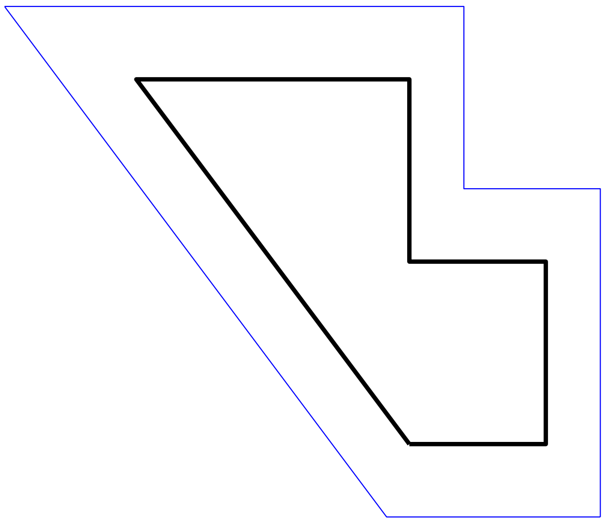

FIGURE 1 – Résultat de l'expansion d'un polygone dans un cas simple.

Dans certains cas (vers l'intérieur d'un polygone convexe) les deux approches aboutisent au même résultat. Les cas qui me préoccupent sont en général d'avoir des distances vers l'intérieur du polygone, j'ai donc privilégié l'approche avec des arêtes parallèles qui me paraissait plus simple à implémenter.

2 Principe

Le principe consiste à calculer le polygone « parallèle » au polygone d'origine, puis à rajouter un post-traitement pour supprimer des artefacts indésirables dus aux concavités du polygone.

2.1 Polygone parallèle

Le calcul du polygone parallèle ne pose pas de problème particulier. Pour chaque arête du polygone, on détermine la droite parallèle à cette arête. On calcule ensuite les points d'intersection des droites parallèles à deux arêtes consécutives. Le polygone expansé est défini par la succession des points d'intersection (Figure 2).

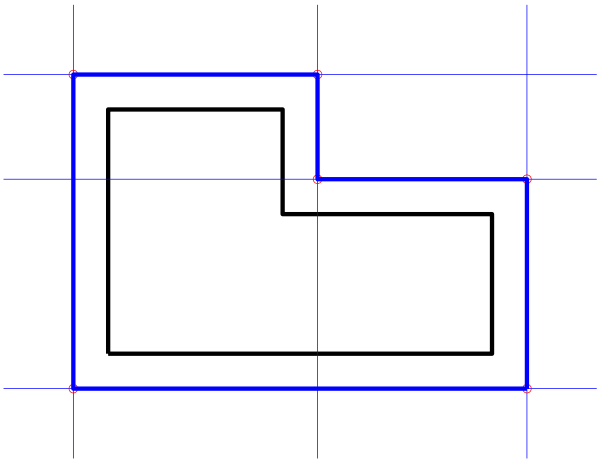

FIGURE 2 – Calcul de l'expansion d'un polygone dans un cas simple. En noir le polygone original, en bleu les droites parallèles à chaque arête, et en rouge les points d'intersection. Le polygone expansé est représenté en bleu et avec un trait épais.

Lorsque le polygone n'est pas convexe, des recouvrements du polygone expansé peuvent apparaître (Figure 2).

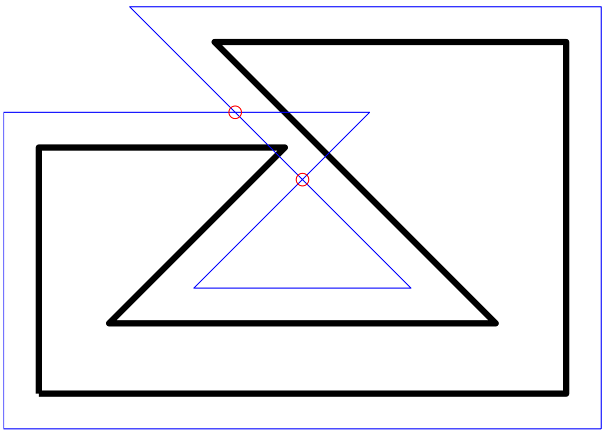

FIGURE 3 – Recouvrement du polygone expansé dans le cas non convexe.

Un traitement additionnel est nécessaire pour décomposer le résultat en polygones qui ne se recoupent pas.

2.2 Décomposition en polygones simples

Les points d'intersection du polygone sont calculés en itérant sur tous les couples d'arêtes, et en évitant les arêtes voisines (qui se touchent par définition).

On définit une position pour un point sur le polygone entre 0 et N_v (N_v étant le nombre de sommets du polygone). La partie entière de la position, entre 0 et $N_v - 1$, identifie l'arête à laquelle appartient le point. La partie fractionnaire (entre 0 et 1) définit la position relative entre les deux extrémités de l'arête.

Chaque point d'intersection peut être identifié par deux positions sur le polygone. Sur le polygone expansé de la figure 3, le point d'intersection du haut peut ainsi avoir la position 2.37, sur la deuxième arête, ou 5.36, sur la cinquième arête (les arêtes sont numérotées de 0 à 7, dans le sens inverse des aiguilles d'une montre, en commençant par l'arête du bas). De manière similaire, le point d'intersection du bas peut avoir la position 2.61 ou 4.61.

	pos1	pos2
1	2.37	5.36
2	2.61	4.61
3	4.61	2.61
4	5.36	2.37

Table 1 – Couples des positions des points d'intersection du polygone expansé.

On stocke ces couples de positions dans un tableau à deux colonnes, une fois en commençant par la première position, une autre fois en commençant par la deuxième. Le résultat est un tableau de $2 \times N_i$ lignes et 2 colonnes. On ordonne ensuite ce tableau en fonction de la première colonne. Pour le polygone de la figure 3, on obtient le tableau 1.

Algorithm 1 Extraction des boucles

```
tant que il reste des intersections faire
  créer une nouvelle boucle (vide)
  pos0 ← première valeur du premier couple
  pos ← deuxième valeur du premier couple
  supprimer la première ligne du tableau
  tant que on ne revient pas au départ faire
    ind ← indice de la première position supérieure à pos
    ajouter à la boucle la portion de courbe comprise entre pos et pos1(ind)
    pos ← pos2(ind)
    supprimer la ligne ind du tableau
  fin tant que
  ajouter la boucle créée à la liste des boucles
fin tant que
```

On décompose ensuite le polygone en plusieurs « boucles », qui sont des polygones simples (sans intersection). Le principe de cette étape est détaillé dans l'algorithme 1.

2.3 Sélection des polygones

L'ensemble de boucles obtenu à l'étape précédente comprend des boucles qui touchent le polygone original, ou qui sont trop proche de lui.

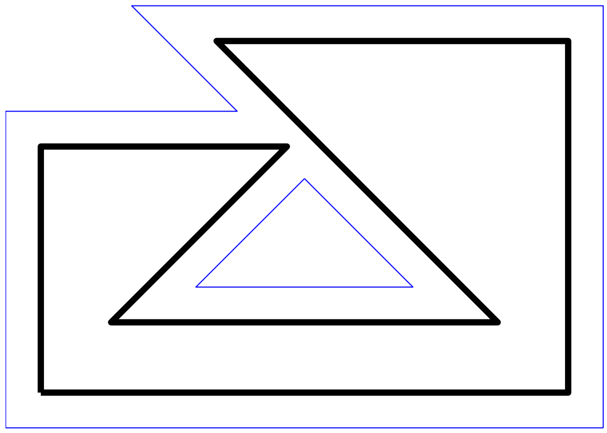

FIGURE 4 – Polygone expansé dans un cas non convexe.

On sélectionne les boucles finales en fonction de leur distance au polygone original. Pour calculer cette distance, on calcule la distance entre chaque sommet d'un polygone et chaque segment de l'autre polygone.

En ne gardant que les boucles situées à la bonne distance, on obtient finalement le polygone expansé (Figure 4).

3 Application à un contour de tige de maïs

L'objectif du développement était de pouvoir repérer des points à l'intérieur d'images de tiges de maïs acquises en macroscopie. Le contour des tiges a été identifié sur les images, et simplifié pour ne garder que quelques dizaines de points (Figure 5).

L'application de l'algorithme permet de définir différents polygones concentriques, pour différentes distances à partir du bord de la tige.

4 Conclusion

L'algorithme semble opérationnel sur les quelques polygones testés. Il n'a pas fait l'objet d'une procédure de tests intensive vis-à-vis des cas pathologiques (segments consécutifs colinéaires).

Les quelques précautions à prendre sont :

- éviter les polygones avec un nombre trop important de points, les algorithmes ne sont pas spécialement optimisés
- éviter les arêtes consécutives colinéaires
- garder en tête que tous les points du polygone expansé ne sont pas à la même distance du polygone d'origine : les sommets en particulier sont en général plus éloignés, du fait que l'on évite d'utiliser des arcs de cercle.

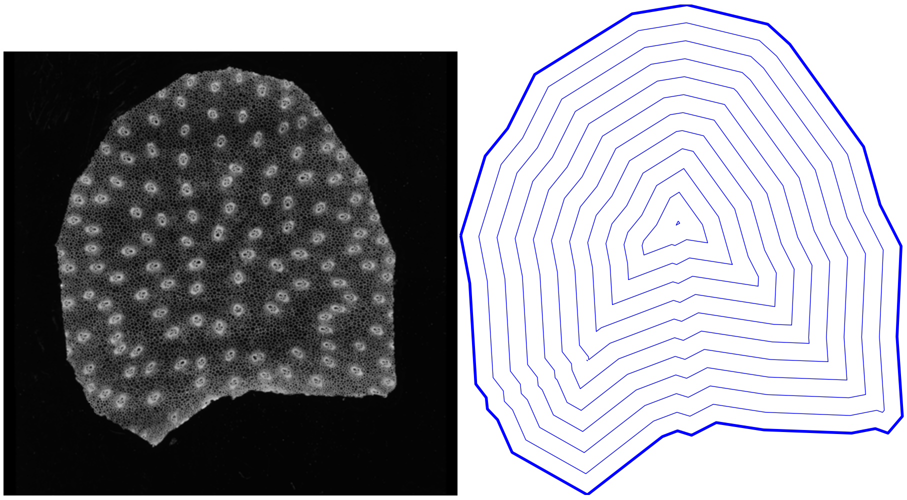

FIGURE 5 – Résultat d'une acquisition d'image de tige, et résultats de l'expansion vers l'intérieur d'un contour simplifié, en utilisant différentes distances.

Il pourrait être intéressant de faire quelques essais pour évaluer le temps d'exécution en fonction du nombre sommets du polygone de base.