Resumen Artículo científico: Generación de β-esqueletos con cómputo paralelo [Kowaluk and Majewska, 2015]

Camilo Andrés Rivera

12 de Junio de 2015

Los β-esqueletos son grafos, en la mayoría de casos 2-D, en el cual dos puntos estarán conectados dependiendo de su ubicación relativa (satisfaciendo ciertas condiciones geométricas) y del parámetro β. Este tipo de algoritmos es usado en una gran variedad de aplicaciones, entre ellas el aprendizaje de máquinas para problemas de clasificación geométrico, y para la generación de mallas de triangulación, por ejemplo para simulaciones de elementos finitos. Este método también es usado ampliamente en el tratamiento de imágenes para la reconstrucción de formas a partir de puntos sobre la frontera, por ejemplo reconstrucción de rostros.

Dado que la cantidad de puntos puede aumentar de gran manera dependiendo de la aplicación, es importante la generación de algoritmos óptimos para la obtención del grafo. Este es el caso de [Lingas, 1994] en donde se muestra un algoritmo óptimo para la construcción del *Relative Neighborhood Graph* para el caso $1 \leq \beta \leq 2$ con un tiempo de $\mathcal{O}(n)$, usando una triangulación Delaunay, en la cual ningún punto puede estar dentro de la circunferencia circunscrita por cada triángulo, como se muestra en la Figura 1. Por otro lado, en [Hurtado et al., 2003], se muestra un algoritmo $\mathcal{O}(n^2)$ para el caso de $\beta < 1$.

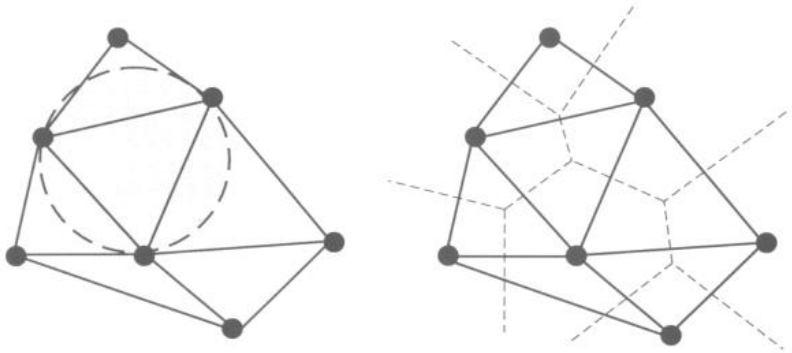

Figura 1: Triangulación de Delaunay con su estructura de Volonoi

Para el caso del artículo elegido [Kowaluk and Majewska, 2015], se tiene que los procesadores usados para la programación en paralelo seguían un modelo CREW_PRAM, el cual quiere decir que múltiples procesadores podían leer una célda de memoria simultáneamente pero sólo uno podía escribirla (*Concurrent read exclusive write*: CREW). El término PRAM viene de Parallel RAM o *parallel random access memory*. Se nombran dos usos de los algoritmos paralelos, de las cuales la primera trata de encontrar un β-*spectrum*, que se le llama al conjunto de todos los vértices que pertenecen al β-esqueleto. El pseudo-algoritmo diseñado para la generación del β-*spectrum* fue: construir la triangulación de Delaunay, construir el conjunto de árboles de eliminación (bosque de eliminación), extraer los polígonos conectados a cada árbol de eliminación, extracción de polígonos centrales y finalmente la estructura de Voronoi. Esta última, como se ve en la parte derecha de la Figura 1, se refiere a las fronteras a partir de las cuales se está más cerca de un punto en particular en vez de otro.

Para el caso del uso de paralelismo para la generación de esqueletos, se asignan los vértices de los árboles de eliminación a procesadores, alternando una clasificación entre procesadores "blancos" y "negros", los cuales están determinados de cierta manera en la que un procesador blanco puede tener dos predecesores negros pero sólo un

sucesor negro. La idea es que en cada iteración para cada árbol de eliminación, se revisa algún vértice correspondiente a un procesador blanco elimina todos los bordes del siguiente grupo de negros, en dado caso se elimina el grupo de procesadores negro y se juntan los blancos con los siguientes. El proceso se realiza similarmente con los conjuntos de procesadores negros. Al eliminar un grupo, se realiza la conexión de los grupos adyacentes (ya sean blancos o negros). Para el caso a $\beta \in [1, 2]$ se demuestra que este algoritmo tiene una velocidad de convergencia de $\mathcal{O}(\log^2 n)$.

Referencias

[Hurtado et al., 2003] Hurtado, F., Liotta, G., and Meijer, H. (2003). Optimal and suboptimal robust algorithms for proximity graphs. *Computational Geometry*, 25(1-2):35–49.

[Kowaluk and Majewska, 2015] Kowaluk, M. and Majewska, G. (2015). New sequential and parallel algorithms for computing the β-spectrum. *Theoretical Computer Science*, 590(0):73 – 85. Fundamentals of Computation Theory.

[Lingas, 1994] Lingas, A. (1994). A linear-time construction of the relative neighborhood graph from the delaunay triangulation. *Computational Geometry*, 4(4):199–208.