

# *DVALon*: una herramienta para diagramas de Voronoi y grafos de proximidad de alcance limitado\*

Manuel Abellanas<sup>†</sup>    Gregorio Hernández<sup>‡</sup>    José Luis Moreno<sup>§</sup>  
Sergio Ordóñez<sup>¶</sup>    Vera Sacristán<sup>||</sup>

## Resumen

Presentamos un estudio de los diagramas de Voronoi de alcance limitado de un conjunto finito de puntos del plano y *DVALon*, una herramienta para la manipulación de dichos diagramas, los grafos de proximidad de alcance limitado y su aplicación a los caminos de desviación mínima y de separación máxima.

## 1 Introducción

Recientemente, los diagramas de Voronoi y, más en general, los grafos de proximidad están siendo objeto del interés de comunidades científicas tan dispares como las surgidas alrededor del análisis y el desarrollo de redes de sensores inalámbricos [8] o del estudio de la evolución de los bosques [1].

En ingeniería forestal se emplea el concepto de región potencialmente disponible de un árbol (*APA: Area Potentially Available*) para estudiar la influencia entre los árboles vecinos en el desarrollo de un bosque. La primera referencia de la que hay constancia es del ingeniero alemán Koenig y se remonta a 1864. Existen varias descripciones de dicha región. Smith en 1987 [11] hace un repaso de las primeras aproximaciones y llega a la conclusión de que los diagramas de Voronoi y los algoritmos que proporciona la Geometría Computacional son los instrumentos adecuados para su modelización. En este caso, las regiones de influencia son necesariamente limitadas por la capacidad de crecimiento de cada especie vegetal. En [10] se emplean las propiedades de los diagramas de Voronoi obtenidos como resultado de la expansión de círculos para estudiar las islas verdes que se crean en los incendios forestales (zonas rodeadas de fuego).

Sean éstas *ad hoc* o fijas, las redes de sensores tienen aplicaciones en la monitorización y el control de procesos industriales, el seguimiento de parámetros medioambientales, el control del tráfico, la automatización y la vigilancia domésticas, de edificios públicos y de oficinas, los cuidados sanitarios y sociales, la detección precoz de incendios, el seguimiento de movimientos sísmicos, las redes de radares y sónares, la detección de partículas químicas o nucleares en el aire, el control de robots móviles y, por descontado, las antenas de telefonía móvil.

En ninguno de estos casos sería realista asumir que los sensores sean capaces de detectar calor, presión, sonido, luz, campos electromagnéticos, vibración o lo que quiera que sea su especialización, a distancias arbitrariamente grandes, como tampoco que sean capaces de comunicarse independientemente de la distancia a que se encuentren los unos de los otros.

---

\*Parcialmente subvencionado por los proyectos MEC MTM2008-05043, MEC MTM2006-01267 y DURSI 2005SGR00692

<sup>†</sup>Universidad Politécnica de Madrid, [manuel.abellanas@upm.es](mailto:manuel.abellanas@upm.es)

<sup>‡</sup>Universidad Politécnica de Madrid, [gregorio@fi.upm.es](mailto:gregorio@fi.upm.es)

<sup>§</sup>Universitat Politècnica de Catalunya, [jlmd86@gmail.com](mailto:jlmd86@gmail.com)

<sup>¶</sup>Universitat Politècnica de Catalunya, [sarchio86@gmail.com](mailto:sarchio86@gmail.com)

<sup>||</sup>Universitat Politècnica de Catalunya, [vera.sacristan@upc.edu](mailto:vera.sacristan@upc.edu)

Es en estos contextos donde cobra interés el diagrama de Voronoi de alcance limitado: una descomposición del plano en regiones, cada una de ellas más cercana a un sensor -a un árbol- que a cualquier otro, pero limitadas por cierta distancia a éste. Por ejemplo, en [5] se propone un método de eliminación de redundancia sin pérdida de cobertura en redes de sensores basado en el reconocimiento de los sensores cuya región de Voronoi de alcance limitado coincide con su región de Voronoi ordinaria, y se caracterizan los sensores que intervienen en la frontera del diagrama de Voronoi de alcance limitado. En el mencionado trabajo no se propone una terminología específica, pero en [12] se bautiza el diagrama de Voronoi de alcance limitado con el nombre de *Aberrant Voronoi Graph*.

Del mismo modo, tienen interés y aplicación los grafos de proximidad de alcance limitado: triangulación de Delaunay, grafo de Gabriel, grafo de vecinos relativos, árbol generador mínimo, etc. Estos grafos han sido introducidos independientemente en [6] y [7], donde reciben los nombres de *Unit Delaunay Triangulation*, *Constrained Gabriel Graph* y *Constrained Relative Neighborhood Graph*.

En particular, los diagramas de Voronoi de alcance limitado y sus grafos de proximidad asociados tienen aplicación directa al estudio de caminos de desviación mínima y de separación máxima, tal como se expone en [2, 4]. La importancia de estos caminos deriva de su uso como herramientas de medida de la calidad de una red de sensores para cubrir un cierto territorio o *área de interés* [9].

En este trabajo presentamos un somero estudio de los diagramas de Voronoi de alcance limitado (Apartado 2) y *DVALon*, una herramienta para la manipulación de dichos diagramas, los grafos de proximidad de alcance limitado más relevantes y su aplicación a los caminos de desviación mínima y de separación máxima (Apartado 3).

## 2 Definiciones y propiedades

### 2.1 Definiciones

Dado un conjunto  $S = \{s_1, \dots, s_n\}$  de  $n$  puntos del plano y un número real  $r \geq 0$ , la *región de Voronoi de alcance  $r$*  de un punto  $s_i \in S$  es:

$$Vor(s_i, r) = \{x \in \mathbb{R}^2 \mid d(x, s_i) \leq d(x, s_j) \forall j \neq i \wedge d(x, s_i) \leq r\}.$$

La región  $Vor(s_i, r)$  es, pues, la intersección de la región de Voronoi ordinaria  $Vor(s_i)$  con el círculo de centro  $s_i$  y radio  $r$ .

El *diagrama de Voronoi de alcance  $r$*  del conjunto  $S$ ,  $Vor(S, r)$ , es la descomposición del plano en las regiones  $Vor(s_i, r)$ . Si  $r = 0$ , el diagrama coincide con  $S$ . Si  $r = +\infty$ , coincide con el diagrama de Voronoi ordinario,  $Vor(S)$ . En el primer caso, consta de  $n$  componentes conexas, en el último de una sola. Si  $r \neq +\infty$ ,  $Vor(S, r)$  puede constar de una o más componentes conexas, cada una de las cuales con posibles agujeros. La Figura 1 ilustra un ejemplo.

### 2.2 Estructura combinatoria de los diagramas

Pese a la continuidad con la que el valor del alcance puede variar a lo largo de la semirrecta real positiva, la estructura combinatoria de  $Vor(S, r)$  se mantiene constante a lo largo de ciertos intervalos y sólo experimenta cambios en determinados valores de  $r \in [0, +\infty)$  en los que se produce la fusión de componentes conexas, la aparición o desaparición de agujeros en éstas, etc. Estos cambios combinatorios, que también llamamos eventos, son de cuatro tipos, que se describen a continuación.

**Lema 2.1.** *Cuando  $r$  alcanza la mitad de la longitud de una arista  $s_i s_j$  del árbol generador mínimo euclídeo de  $S$ ,  $EMST(S)$ , se conectan dos de las componentes conexas de  $Vor(S, r)$ .*

*Demostración.* Es inmediato ver que cuando  $r$  alcanza la mitad de la distancia entre dos puntos  $s_i$  y  $s_j$  sus regiones  $Vor(s_i, r)$  y  $Vor(s_j, r)$  quedan conectadas entre sí, por lo que lo único que queda por demostrar es que  $s_i$  y  $s_j$  pertenecen a componentes conexas distintas antes de este evento. Supongamos

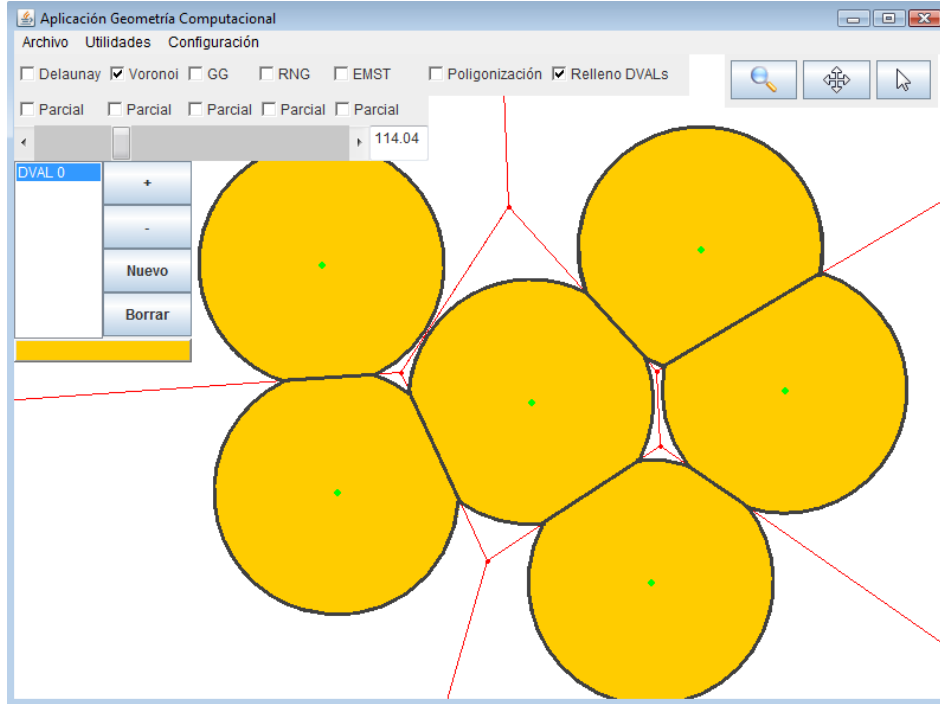


Figura 1: Ejemplo de  $Vor(S, r)$  con sus distintas regiones  $Vor(s_i, r)$ .

que  $s_i$  y  $s_j$  se encuentran en la misma componente conexa. En tal caso, tendría que existir un camino entre  $s_i$  y  $s_j$  formado únicamente por aristas de tamaño menor a  $2r$  y, si esto fuera cierto, la arista  $s_i s_j$  nunca habría sido elegida para formar parte de  $EMST(S)$  por formar un ciclo. Por lo tanto, si la arista  $s_i s_j$  pertenece a  $EMST(S)$ , significa que  $s_i$  y  $s_j$  pertenecen a componentes conexas distintas en  $Vor(S, r')$ , para todo  $r' < r$ .  $\square$

**Lema 2.2.** Cuando  $r$  alcanza la mitad de la longitud de una arista  $s_i s_j$  de Gabriel que no pertenece al árbol generador mínimo euclídeo de  $S$ , se produce un agujero en el diagrama  $Vor(S, r)$ .

*Demostración.* El hecho de que la arista  $s_i s_j$  no pertenezca a  $EMST(S)$  implica que  $s_i$  y  $s_j$  pertenecen a una misma componente conexa de  $Vor(S, r)$  para algún valor de  $r$  menor que la mitad de la longitud de la arista  $s_i s_j$ , porque existe un camino en  $EMST(S)$  que los une, con aristas de longitud estrictamente inferior a  $2r$ . Por otra parte, el hecho de que la arista  $s_i s_j$  sea de Gabriel implica que corta a su arista dual de Voronoi, de modo que los circuncentros de los dos triángulos adyacentes a la arista de Delaunay  $s_i s_j$  están uno a cada lado de ésta (son los dos extremos de la arista de Voronoi). Cuando  $r$  alcanza la mitad de la longitud de la arista  $s_i s_j$ ,  $Vor(S, r)$  contiene tanto la arista  $s_i s_j$  como el camino entre  $s_i$  y  $s_j$  que pasa por aristas de  $EMST(S)$  formando un ciclo, pero no contiene un punto interior como es el circuncentro del triángulo que queda rodeado por el ciclo, y que está en el interior de éste, pero a una distancia mayor que  $r$  de  $s_i$ ,  $s_j$  y de cualquier otro punto de la nube (por ser la triangulación de Delaunay). Resulta así que, al cerrar la arista  $s_i s_j$  un ciclo en  $Del(S, r)$ , queda un conjunto de puntos interiores al ciclo y que no pertenecen a  $Vor(S, r)$ , es decir, se produce un agujero en  $Vor(S, r)$ .  $\square$

**Lema 2.3.** Cuando  $r$  alcanza el radio de la circunferencia circunscrita a un triángulo de Delaunay cuyo centro es interior al triángulo, es decir, a un triángulo de Delaunay acutángulo, desaparece uno de los agujeros de  $Vor(S, r)$  y aparece un nuevo vértice de Voronoi.

*Demostración.* El radio de la circunferencia circunscrita al triángulo es mayor que la mitad del lado mayor de éste. Cuando  $r$  alcanza la mitad de la longitud del mayor de los lados, los tres lados del triángulo quedan contenidos en  $Vor(S, r)$  y se crea un agujero que consiste en cierta porción del interior del triángulo. El circuncentro del triángulo es uno de los puntos incluidos en el agujero, por ser interior al triángulo pero estar a una distancia mayor que  $r$  de sus tres vértices. Cuando  $r$  alcanza la longitud

del radio de la circunferencia circunscrita, todo el interior del triángulo se encuentra a distancia menor o igual que  $r$  de alguno de sus tres vértices y, por tanto, todo el triángulo está contenido en  $Vor(S, r)$  y desaparece el agujero. Allí donde estaba el circuncentro aparece un vértice de Voronoi, por ser este punto equidistante a los tres vértices del triángulo.  $\square$

**Lema 2.4.** *Cuando  $r$  alcanza el radio de la circunferencia circunscrita a un triángulo de Delaunay cuyo centro se encuentra en la frontera o en el exterior del triángulo, es decir, a un triángulo de Delaunay rectángulo u obtusángulo, desaparece un arco de circunferencia en la frontera de  $Vor(S, r)$  y aparece un nuevo vértice de Voronoi.*

*Demostración.* En el caso obtusángulo, cuando  $r$  alcanza la longitud de la mitad del mayor de los lados del triángulo de Delaunay, los tres vértices delimitan tres arcos de circunferencia en la frontera de  $Vor(S, r)$ . Al incrementar el alcance  $r$  en  $Vor(S, r)$ , el arco correspondiente al vértice que forma el ángulo obtuso se va haciendo más pequeño en la porción que se acerca al circuncentro del triángulo. Cuando  $r$  finalmente alcanza el radio de la circunferencia circunscrita al triángulo, el arco desaparece justamente en el punto equidistante a los tres vértices del triángulo (el circuncentro) y, a partir de ese momento, cualquier incremento de  $r$  supone la inclusión de nuevos punto en el  $Vor(S, r)$  que están más cerca de alguno de los otros dos vértices del triángulo que del vértice obtuso. Allí donde estaba el circuncentro aparece un vértice de Voronoi, por ser este punto equidistante a los tres vértices del triángulo. El razonamiento anterior se adapta de forma inmediata al caso de un triángulo rectángulo, para el cual el punto medio del mayor de los lados coincide con el circuncentro.  $\square$

En conclusión, hemos obtenido el siguiente resultado:

**Proposición 2.5.** *Dado un conjunto  $S = \{s_1, \dots, s_n\}$  de  $n$  puntos del plano, el número de diagramas de Voronoi de alcance limitado,  $Vor(S, r)$ , combinatoriamente distintos es  $m = O(n)$  y los valores del alcance  $r_1, \dots, r_m$  en que se producen los cambios combinatorios en la estructura de  $Vor(S, r)$  son los descritos en los lemas anteriores.*

### 2.3 Construcción de $Vor(S, r)$

En este apartado se estudia la forma de construir  $Vor(S, r)$  para un radio  $r$  dado y de obtener  $Vor(S, r_i)$  para todos los radios  $r_i$  que determinan algún cambio combinatorio en  $Vor(S, r)$ .

**Proposición 2.6.** *Para cualquier valor  $r \geq 0$  y cualquier conjunto  $S$  de  $n$  puntos del plano, es posible calcular el diagrama de Voronoi de alcance  $r$ ,  $Vor(S, r)$ , a partir del diagrama de Voronoi ordinario,  $Vor(S)$ , en tiempo  $O(n)$ .*

*Demostración.* Teniendo ya calculado  $Vor(S)$ , el cálculo de cada región  $Vor(s_i, r)$  se puede realizar cortando la propia región  $Vor(s_i)$  con el círculo  $B(s_i, r)$ , cosa que puede hacerse en tiempo proporcional a la complejidad de la región  $Vor(s_i)$ , dando lugar a un algoritmo de coste acumulado  $O(n)$ .  $\square$

**Proposición 2.7.** *El problema de obtener todos los diagramas  $Vor(S, r_i)$  combinatoriamente distintos de un conjunto  $S$  de  $n$  puntos dados puede resolverse en tiempo  $O(n \log n)$ .*

*Demostración.* Se construyen la triangulación de Delaunay  $Del(S)$  y el diagrama de Voronoi  $Vor(S)$  en tiempo  $O(n \log n)$  y se extraen los grafos  $EMST(S)$  y  $GG(S)$  en tiempo  $O(n)$ . A partir de aquí se pueden obtener los valores de todos los alcances  $r_1, \dots, r_m$  que producen cambios combinatorios en  $Vor(S, r)$ . Obsérvese que  $m = O(n)$  y que los valores  $r_1, \dots, r_m$  se pueden obtener en tiempo  $O(n)$  recorriendo las estructuras  $Del(S)$ ,  $Vor(S)$ ,  $EMST(S)$  y  $GG(S)$ . Se ordenan estos valores en tiempo  $O(n \log n)$  y a cada uno de ellos se le asigna el cambio combinatorio que le corresponde. Cada uno de estos cambios se puede insertar sucesivamente a partir del anterior en tiempo  $O(\log n)$ , de modo que el conjunto de todos ellos se construye en tiempo  $O(n \log n)$ .  $\square$

## 3 Descripción de la aplicación

*DVALon* es una aplicación cuyo objetivo es facilitar el estudio y la manipulación de diagramas de Voronoi de alcance limitado y su aplicación a los caminos de desviación mínima. La aplicación permite, dada una nube de puntos, obtener, visualizar y almacenar la información de los diagramas de Voronoi de alcance limitado y de diversos grafos de proximidad del mencionado conjunto de puntos. En los apartados siguientes se ofrece una breve explicación de las funcionalidades más relevantes de la aplicación y de su funcionamiento.

### 3.1 Funcionalidades

Para comenzar con el uso de la aplicación, es necesario introducir una nube de puntos. Para ello, el usuario dispone de tres opciones. La más intuitiva es introducir puntos con el ratón mediante *clicks* sobre la pantalla. La segunda opción es introducir el conjunto de puntos mediante un fichero que contenga las coordenadas de los mismos, lo que permite al usuario precisar la posición exacta de éstos. La tercera posibilidad es la de generar el número deseado de puntos distribuidos aleatoriamente en la pantalla de visualización.

Una vez introducido el conjunto de puntos  $S = \{s_1, \dots, s_n\}$ , el usuario puede visualizar distintos grafos de proximidad entre los que se encuentran los diagramas de Voronoi de alcance limitado calculados a partir de  $S$ .

Inicialmente la aplicación parte de un único  $Vor(S, r)$  con alcance igual a cero. El usuario dispone de tres mecanismos para modificar el alcance: una barra deslizante que permite modificar el valor de  $r$  de manera continua, un campo en el que se puede introducir un valor determinado del alcance, y un botón con el que ir moviéndose de manera discreta entre los alcances que producen un cambio combinatorio en  $Vor(S, r)$ .

La aplicación permite, además, crear tantos diagramas de Voronoi de alcance limitado como el usuario desee, así como borrarlos, con lo que es posible visualizar simultáneamente varios diagramas con alcances distintos. Todos los diagramas aparecen de un color distinto en la pantalla y en orden creciente de alcance en una lista. Gracias a esta lista, el usuario puede seleccionar el diagrama que quiera y modificar únicamente el diagrama seleccionado. En la Figura 2 puede verse cómo la aplicación permite visualizar simultáneamente varios diagramas de Voronoi de alcance limitado.

Además de los diagramas de Voronoi de alcance limitado, la aplicación ofrece la posibilidad de visualizar los siguientes grafos de proximidad: la triangulación de Delaunay del conjunto de puntos,  $Del(S)$ , su grafo de Gabriel,  $GG(S)$ , su grafo de los vecinos relativos,  $RNG(S)$ , y su árbol generador mínimo euclídeo,  $EMST(S)$ . La Figura 3 muestra algunos de estos grafos calculados por la aplicación.

Dado que el interés de este proyecto es trabajar con diagramas de alcance limitado, la aplicación permite visualizar estos grafos no sólo en su versión ordinaria, sino también de manera parcial, es decir, limitados por el alcance de un  $Vor(S, r)$  determinado. Esto puede hacerse seleccionando la opción de visualización parcial del grafo (o grafos) deseado, de forma que lo veremos limitado por el alcance del  $Vor(S, r)$  seleccionado.

A la hora de guardar un proyecto, la aplicación genera también un fichero de texto con toda la información descrita anteriormente. De esta manera, el usuario, una vez ha guardado su proyecto, dispone también de un fichero de texto con los datos concretos de éste. Asimismo, el usuario siempre puede abrir proyectos ya guardados y recuperarlos tal y como los tenía en el momento en que fueron almacenados.

### 3.2 Funcionamiento

Como ya se ha visto, para que la aplicación comience a funcionar es necesario introducir una nube de puntos. Una vez introducidos como mínimo dos puntos, el programa calcula todos los grafos de

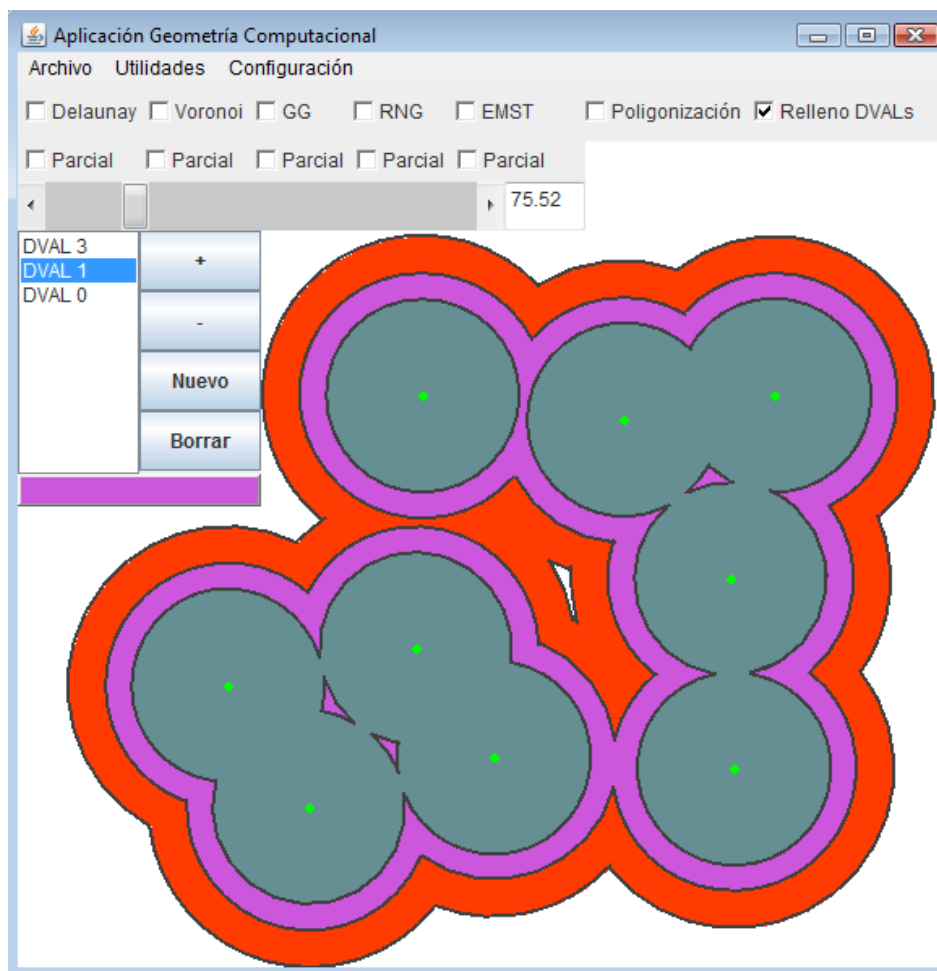


Figura 2: Ejemplo de situación con varios diagramas de Voronoi de alcance limitado.

proximidad descritos anteriormente. Al introducir un nuevo punto, la aplicación recalcula localmente la triangulación de los puntos y su diagrama de Voronoi, es decir, sólo recalcula los elementos de ambos que resultan afectados directamente por la inclusión del nuevo punto. El caso de modificación de un punto es análogo al de inclusión de un nuevo punto a efectos del recálculo de elementos. En cambio, al eliminar un punto la aplicación recalcula de nuevo todos los grafos de proximidad.

Los grafos de Gabriel, de vecinos relativos y al árbol generador mínimo, se recalculan siempre en función del grafo ya calculado en el que están contenidos. Es decir, el grafo de Gabriel se calcula a partir de la triangulación de Delaunay, el grafo de vecinos relativos a partir del grafo de Gabriel, y el árbol generador mínimo a partir del grafo de vecinos relativos.

Tal como se desprende de la descripción de las funcionalidades, la aplicación también calcula, clasifica y almacena los cuatro tipos de eventos, correspondientes a los cambios combinatorios que se producen en los diagramas  $Vor(S, r)$  descritos en el apartado 2, para la nube dada de puntos.

Para su fácil uso, la aplicación almacena la triangulación de Delaunay y el diagrama de Voronoi en sendas estructuras DCEL (Doubly-Connected Edge List) y los grafos de Gabriel, de vecinos relativos y el árbol generador mínimo en listas de adyacencias. Esto permite acceder de manera rápida a la información de estos grafos y facilita que funciones de la aplicación que se ejecutan frecuentemente, como por ejemplo el dibujado de todos los elementos calculados, sean lo más eficientes posible. En cuanto a las regiones formadas por los diagramas de Voronoi de alcance limitado, las fronteras y agujeros de sus componentes conexas están descritos como una lista de segmentos y arcos. Los elementos de las fronteras están ordenados en sentido antihorario, mientras que los de los agujeros lo están en sentido horario.

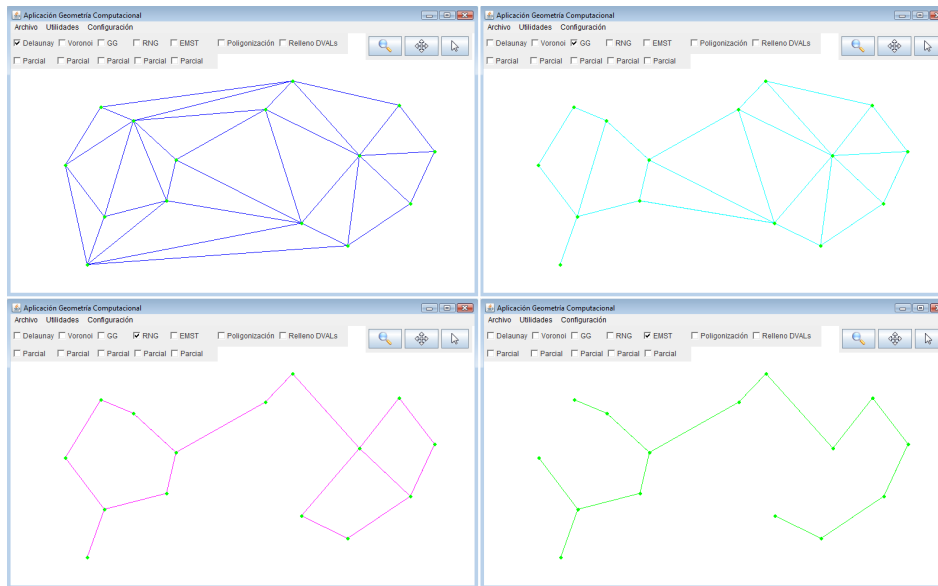


Figura 3: Ejemplo de algunos de los grafos de proximidad calculados.

El fichero de texto generado al guardar un proyecto contiene, en primer lugar, las DCEL correspondientes a la triangulación de Delaunay y al diagrama de Voronoi de la nube de puntos. A continuación, aparece toda la información relativa a los distintos  $Vor(S, r)$  creados por el usuario, ordenados siempre por valor creciente del alcance. Esta información consta del radio del diagrama, el número de componentes conexas y, para cada una de ellas, los puntos de la nube que contiene y las descripciones de su frontera exterior, de sus agujeros y de su poligonización. También aparece en este archivo el número de eventos (cambios combinatorios) que se producen en el diagrama de Voronoi de alcance limitado de la nube de puntos y, para cada evento, su tipo y el valor del radio  $r$  que lo produce. Para terminar, el fichero guarda la información relativa a los grafos de Gabriel y de vecinos relativos y al árbol generador mínimo de la nube. En concreto, para cada uno de los tres encontramos sus vértices, los vértices adyacentes a éstos y, para cada una de estas aristas, su longitud al cuadrado.

### 3.3 Ampliación en curso

La segunda etapa de este proyecto, que en el momento de escribir este texto se encuentra en fase de desarrollo, consiste en implementar funcionalidades destinadas al estudio y la optimización de caminos de desviación mínima y de separación máxima [2, 9, 4], basadas en diagramas de Voronoi de alcance limitado. Para ello, se prevé implementar algoritmos de decisión para detectar si un camino dado es o no de desviación mínima, de desviación mínima local, de separación máxima, algoritmos para el cálculo y comparación de la longitud entre caminos, etc.

Para obtener información actualizada sobre *DVALon*, consúltese la página [3].

## Referencias

- [1] B. Abellanas, M. Abellanas, C. Vilas. VOREST: Modelización de bosques mediante diagramas de Voronoi. *Actas de los XII Encuentros de Geometría Computacional (EGC'07)*, pp. 249–256, 2007.
- [2] M. Abellanas, G. Hernández. Optimización de rutas de evacuación, *Actas de los XII Encuentros de Geometría Computacional (EGC'07)*, pp. 273–280, 2007.
- [3] M. Abellanas, G. Hernández, J. L. Moreno, S. Ordóñez, V. Sacristán. *DVALon*. <http://www-ma2.upc.es/~geoc/DVALon/>.

- [4] M. Abellanas, G. Hernández, V. Sacristán. Caminos de desviación mínima local, presentado a los *XIII Encuentros de Geometría Computacional (EGC'09)*, 2009.
- [5] B. Cărbunar, A. Grama, J. Vitek, O. Cărbunar. Redundancy and Coverage Detection in Sensor Networks, *ACM Transactions on Sensor Networks*, Vol. 2, N. 1, pp. 94–128, 2006.
- [6] J. Gao, L. J. Guibas, J. Hershberger, L. Zhang, A. Zhu. Geometric Spanner for Routing in Mobile Networks, *Proc. 2nd ACM International Symposium on Mobile and Ad Hoc Networking and Computing (MobiHoc'01)*, pp. 45–55, 2001.
- [7] X.-Y. Li, G. Calinescu, P.-J. Wan, Y. Wan. Localized Delaunay Triangulation with Application in Ad Hoc Wireless Networks, *IEEE Transactions on Parallel and Distributed Systems*, Vol. 14, N. 10, pp. 1035–1047, 2003.
- [8] S. Meguerdichian, F. Koushanfar, M. Potkonjak, M. B. Srivastava. Coverage Problems in Wireless Ad-hoc Sensor Networks, *Proc. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies (INFORCOM 2001)*, Vol. 3, pp. 1380–1387, 2001.
- [9] S. Megerian, F. Koushanfar, M. Potkonjak, M. B. Srivastava. Worst and Best-Case Coverage in Sensor Networks, *IEEE Transactions on Mobile Computing*, Vol. 4, N. 1, pp. 84–92, 2005.
- [10] W. L. Roque, H. Choset. The Green Island Formation in Forest Fire Modeling with Voronoi Diagrams, *Proc. 3rd CGC Workshop on Computational Geometry*, 1998.
- [11] W. R. Smith. Area Potentially Available to a Tree: A Research Tool, *Proc. 19th Southern Forest Tree Improvement Conference, SFTIC*, pp. 22–29, 1987.
- [12] H. Xu, L. Huang, Y. Wan, K. Lu. Localized Algorithm for Coverage in Wireless Sensor Networks, *Proc. IEEE Sixth International Conference on Parallel and Distributed Computing, Applications and Technologies (PDCAT'05)*, pp. 750–754, 2005.