Fútbol, geometría y otros problemas.

Leandro Tortosa.

Universidad de Alicante, Alicante (España).
Departamento de Ciencia de la Computación e Inteligencia Artificial.
Ap. Correos 99, E-03080.
Alicante, España.
tortosa@ua.es, www.dccia.ua.es/~tortosa

RESUMEN. Una colección de puntos en plano o en el espacio puede parecer un conjunto o estructura matemática no demasiado interesante. Nada más lejos de la realidad. Con un simple conjunto de puntos podemos construir diagramas de Voronoi, realizar triangulaciones, construir complicados conjuntos geométricos como las α-formas, envolventes convexas, etc. La geometría computacional constituye una rama de las ciencias de la computación que trata de resolver problemas geométricos por medio de algoritmos. La incorporación de ciertas funciones relacionadas con la geometría computacional en la versión 4 de GeoGebra nos abre un amplio abanico de posibilidades a la hora de proponer ciertos problemas "reales" que pueden resultar motivadores para el estudiante y que nos permite enfocar la docencia de algunos conceptos geométricos desde la práctica. En este trabajo se analizan diversos ejemplos en los que se aplican técnicas de geometría computacional sobre conjuntos de puntos en el plano.

1. Introducción

Pensemos en un conjunto de puntos en el plano. A primera vista, parece que desde el punto de vista matemático no tiene mucho interés. Estamos acostumbrados a trabajar con estructuras matemáticas complejas como grupos, anillos, matrices, espacios vectoriales, etc. Sin embargo, nada más lejos de la realidad. Trataremos de demostrar en este breve artículo que un conjunto de puntos en el plano constituye una estructura matemática apasionante, tanto por sus posibilidades como por los resultados obtenidos, como vamos a tratar de visualizar ayudándonos, en la medida de lo posible, de GeoGebra (www.geogebra.org/).

La pregunta de la que partimos es: ¿qué podemos hacer con un conjunto de puntos en el plano?

La respuesta a esta pregunta está relacionada con una especialidad reciente en el campo de las matemáticas, como es la geometría computacional.

Podemos decir, de una manera más o menos formal, que la geometría computacional trata de resolver problemas geométricos mediante la aplicación de algoritmos. Por eso, se trata de una especialidad reciente, ya que se engloba dentro de la computación y resulta imprescindible el uso de computadoras, en las que se implementen y ejecuten los algoritmos diseñados para la resolución de los diferentes problemas.

Dado *S* un conjunto finito de puntos en el plano, podemos realizar las siguientes operaciones sobre ellos, entre otras muchas.

- ✓ Cálculo de su **envolvente convexa**, que denotamos por *CH(S)*.
- ✓ Cálculo del **diagrama de Voronoi**, que denotamos por *Vor(S)*.
- ✓ **Triangulación de Delaunay**, que escribimos como *DT(S)*.

Para una descripción más detallada de los conceptos y algoritmos básicos de geometría computacional, podemos consultar [1,2].

2. La envolvente convexa

Supongamos que *S* es un conjunto finito de puntos del plano. Nos interesa determinar el menor conjunto convexo que contiene a *S*. Dicho conjunto siempre existe y se llama la **envolvente convexa** de *S*.

Así pues, podemos afirmar que la envolvente convexa de un conjunto de puntos *S* es el polígono convexo *P* que contiene a todos los elementos de *S* con menor área (o perímetro) posible.

La envolvente convexa es un polígono convexo cuyos vértices son elementos de *S*. Si de forma intuitiva consideramos los puntos de *S* como clavos sobre un panel de madera, entonces podemos pensar en la frontera de la envolvente convexa como la forma que toma una liga elástica que encierra a todos los clavos.

Entre las aplicaciones más importantes de la envolvente convexa podemos citar:

- ➢ Cálculo del diámetro de un conjunto.
- ➢ Cálculo de la anchura de un conjunto.
- ➢ Cálculo del círculo vacío máximo.

Dado un conjunto de puntos *S* hablamos de el diámetro del conjunto de puntos para referirnos a la mayor distancia entre sus puntos. Cuando hablamos de anchura de *S* nos estamos refiriendo a la menor longitud de entre todas las de dos rectas paralelas que contengan en su interior todos los puntos de *S*.

El cálculo del diámetro de un conjunto de puntos nos sirve para resolver problemas reales en los que sea necesario obtener medidas de dispersión. Una aplicación real que requiere el cálculo de la anchura de un conjunto de puntos es el cálculo de trayectorias de vehículos móviles o robots.

En el problema del cálculo del círculo vacío máximo es necesario el cálculo de la envolvente convexa del conjunto de puntos, ya que nos interesa conocer su intersección con el diagrama de Voronoi del conjunto.

3. Los diagramas de Voronoi

Partimos, como ya venimos haciendo a lo largo de este capítulo, de un conjunto *S* de *n* puntos en el plano.

Para cada pareja de puntos del plano, por ejemplo, p_i y p_j, dibujamos la mediatriz del segmento que los une p_ip_j, de manera que el plano queda dividido por dos semiplanos.

Denotamos por $H(p_i, p_j)$ al semiplano compuesto por los puntos del plano más cercanos a p_i que a p_j.

Ahora, para cada punto p_i de S, denotamos por V_i al conjunto de puntos del plano más cercano al punto p_i que a los restantes puntos del conjunto S - $\{p_i\}$.
Entonces, se puede demostrar que

$$V_i = \cap_{j \neq i} H(p_i, p_j).$$

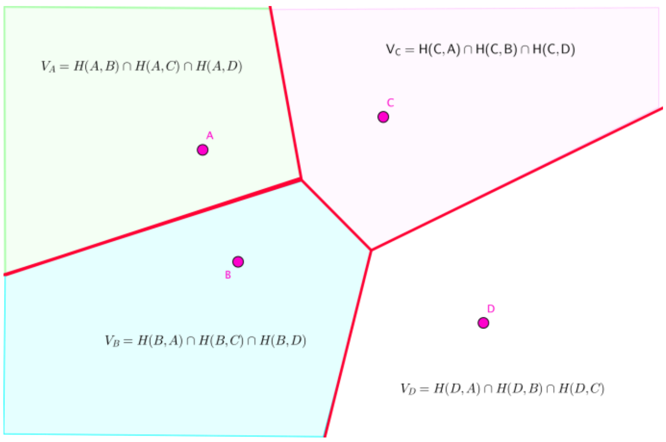
Esto significa que tenemos una forma de calcular el conjunto de puntos más cercano a un punto concreto p_i del conjunto S. Basta con calcular inicialmente todas las mediatrices de ese punto con el resto para obtener los conjuntos $H(p_i,pj)$, con j distinto de i. Después, es necesario calcular la intersección de todos esos semiplanos. Lo vemos más claramente con un ejemplo.
Supongamos que tenemos un conjunto S formado por cuatro puntos, es decir, $S=\{p_1, p_2, p_3, p_4\}$. Calculamos V_1, V_2, V_3, V_4 para el conjunto S.
Después de dibujar los puntos en el plano trazamos los segmentos p_1p_2, p_1p_3, p_1p_4, p_2p_3, p_2p_4, p_3p_4.
Una vez obtenidos los segmentos, podemos trazar las correspondientes mediatrices de esos segmentos, utilizando el siguiente criterio de notación: denotamos por m_{ij} a la mediatriz del segmento p_ip_j. Representamos las mediatrices m_{12}, m_{13}, m_{14}, m_{23}, m_{24}, m_{34}. Una vez dibujada la mediatriz m_{ij} ya estamos en disposición de dividir el plano en dos semiplanos, $H(p_i, p_j)$ y $H(p_j, p_i)$. Precisamente la intersección de estos semiplanos es la que nos proporciona los conjuntos de puntos más próximos a uno u otro punto.
Entonces se pueden calcular los conjuntos V_i.

Para $i=1$, calculamos $V_1 = H(p_1,p_2) \cap H(p_1,p_3) \cap H(p_1,p_4)$.
Para $i=2$, calculamos $V_2 = H(p_2,p_1) \cap H(p_2,p_3) \cap H(p_2,p_4)$.
Para $i=3$, calculamos $V_3 = H(p_3,p_1) \cap H(p_3,p_2) \cap H(p_3,p_4)$.
Para $i=4$, calculamos $V_4 = H(p_4,p_1) \cap H(p_4,p_2) \cap H(p_4,p_3)$.

Una vez construido V_1 realizamos un proceso similar con el resto de los puntos. En consecuencia, podemos definir el diagrama de Voronoi $Vor(S)$ de la forma

$$Vor(S) = \cup_{i=1}^{n} V_i \,.$$

En la figura siguiente se muestra un ejemplo del diagrama de Voronoi de un conjunto de puntos.



Una primera observación del diagrama de Voronoi nos permite obtener las siguientes conclusiones:

- Si tenemos un punto p_i en S, entonces su vecino más próximo se halla en alguno de los polígonos de Voronoi adyacentes a V_i.

- Si ordenamos en una lista cada punto p_i con su vecino más cercano, entonces podemos buscar en dicha lista el par de elementos de *S* más cercanos.

4. **Aplicaciones**.

Entre las aplicaciones más importantes de los diagramas de Voronoi podemos citar las siguientes:

- Cálculo del vecino más cercano.
- Encontrar los dos puntos más próximos.
- Cálculo del mayor círculo vacío posible.
- Cálculo del círculo de recubrimiento mínimo.
- Trayectorias de robots.

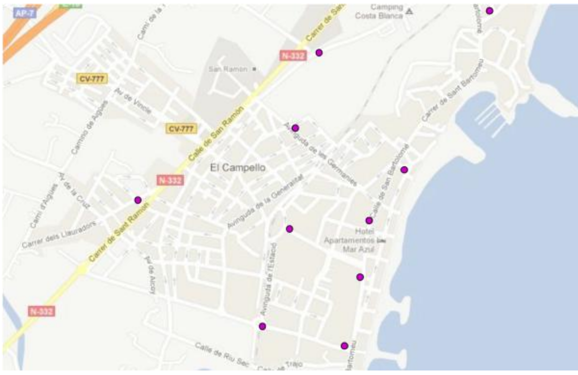
La siguiente tabla resume en qué consiste cada uno de estos problemas.

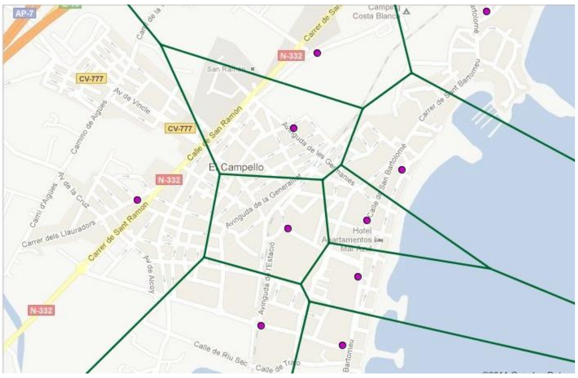
Cálculo del vecino más cercano.	Sea *S* un conjunto de puntos en el plano y un punto p que no pertenece a *S*. El problema consiste en determinar qué punto de S se encuentra más próximo a *p*.
Encontrar los dos puntos más próximos.	Dado *S*, consideramos el problema de determinar los dos puntos más próximos en dicho conjunto. Para ello, cada punto localiza su vecino más cercano y después se comprueba cuál de estos pares se encuentran a menor distancia.
Cálculo del mayor círculo vacío posible.	Se trata de encontrar el mayor círculo que no contenga puntos de *S*, cuyo centro se encuentre dentro de la envolvente convexa del conjunto de puntos. El centro de este círculo es un vértice de Voronoi o de su intersección con la envolvente convexa.
Cálculo del círculo de recu-brimiento mínimo.	Se trata en este caso de encontrar el menor círculo que engloba a un conjunto de puntos del plano.
Trayectorias de robots.	Si un robot debe moverse por una escena en la que existen obstáculos, nos interesa encontrar una trayectoria para el mismo lo más alejada posible de ellos. Dicha trayectoria se diseña sobre las aristas del diagrama de Voronoi generado a partir de los obstáculos.

A continuación se plantean algunos problemas cuya resolución se basa en el concepto y propiedades de los diagramas de Voronoi.

Ejemplo 1. Cálculo del vecino más próximo.

Tenemos en la imagen siguiente un mapa de una ciudad con la localización de los grandes supermercados existentes en la misma. Queremos saber, de acuerdo con la posición en la que nos encontremos, cuál de ellos es el que tenemos más cerca.

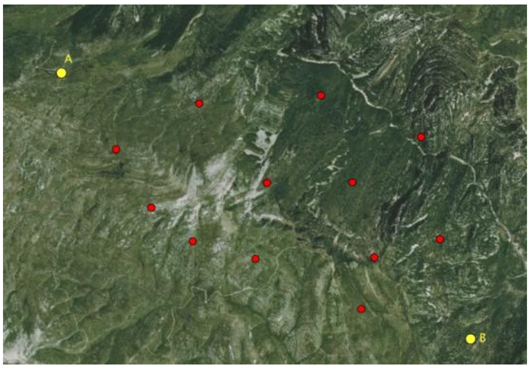




El diagrama de Voronoi para este conjunto de puntos nos proporciona las regiones más próximas a cada uno de los puntos, que en este caso representan supermercados (imagen de la derecha).

Ejemplo 2. Cálculo de trayectorias que eviten obstáculos.

Supongamos que queremos establecer una ruta a pie a través de un valle y una sierra que hemos representado por medio de la figura:



En dicha figura hemos representado por *A* y *B* el punto inicial y final, respectivamente, de la ruta. En dicho valle existe un tipo de vegetación autóctona que queremos proteger y preservar en la medida de lo posible puesto que se trata de especies protegidas. La presencia de dichas especies se ha marcado con puntos en el dibujo. Establecer una ruta que pase lo más lejos posible de estos puntos.

Ejemplo 3. Trayectorias de robots.

Supongamos el problema de un robot que debe atravesar una zona con obstáculos, que debe evitar a toda costa. La idea es establecer una trayectoria de manera que pase lo más lejos posible de los obstáculos.

Ejemplo 4. Cálculo del círculo vacío más grande.

Consideremos el problema de la instalación de una planta de residuos químicos en nuestra Comunidad Autónoma. Se trata de una instalación que conlleva un riesgo y supone un peligro evidente para la población en caso de accidente. Evidentemente, nadie desea una instalación de este tipo en su proximidad. Las autoridades responsables deciden que su localización geográfica debe ser tal que se encuentre lo más alejada posible de cualquier núcleo urbano cuya población supere los 10 000 habitantes. Sin embargo, por motivos logísticos, su construcción no debe situarse en zonas demasiado alejadas de la civilización, puesto que los costes de infraestructuras serían insostenibles. Nuestro problema consiste en buscar una localización geográfica para dicha instalación.

Ejemplo 5. El fútbol y Voronoi.

Puesto que un juego como el fútbol es un juego de posiciones en un área determinada, podemos plantear el determinar las posiciones de los jugadores y el área que les corresponde mediante el diagrama de Voronoi. Existen páginas web dedicadas a deportes en las que podemos encontrar estadísticas de todo tipo, entre las que se encuentran las posiciones medias de los jugadores en el transcurso de un partido. A partir de estas posiciones medias, podemos construir en GeoGebra el diagrama de Voronoi y extraer conclusiones sobre la forma de juego de un jugador individual o del equipo en conjunto.

Bibliografía.

[1] Devadoss, Satyan L., O'Rourke, Joseph. "Discrete and Computational Geometry". Princeton University Press, 2011.

[2] Sack, J.R., Urrutia, J. "Handbook of Computational Geometry". North-Holland, 1999.

SOCIEDAD DE LA INFORMACION

www.sociedadelainformacion.com

Edita:

Director: José Ángel Ruiz Felipe

Jefe de publicaciones: Antero Soria Luján

D.L.: AB 293-2001

ISSN: 1578-326x