

J.M. Escobar y R. Montenegro

INTRODUCCIÓN A LA GENERACIÓN DE MALLAS EN 3-D

1. INTRODUCCIÓN.

Sea $X = \{x_k \in E_3 / 1 \leq k \leq n\}$ un conjunto de n puntos de un espacio afín Euclideo tridimensional E_3 . Definimos el poliedro de Voronoi V_i asociado al punto x_i como: $V_i = \{x \in E_3 / d(x, x_i) \leq d(x, x_j), i \neq j, 1 \leq j \leq n\}$. En general, cada vértice de un poliedro de Voronoi es compartido por otros cuatro de estos poliedros (sólo en los casos degenerados hay más de cuatro); Cavendish [1985]. Uniendo los puntos asociados a cada uno de los cuatro poliedros contiguos a uno de estos vértices v_k se forma un tetraedro t_k . El conjunto de tetraedros $T = \{t_k / 1 \leq k \leq m_n\}$ constituye una triangulación tridimensional de Delaunay asociada al conjunto de puntos X . Una propiedad de gran importancia en la triangulación de Delaunay es que la esfera $b(t_k)$ circunscrita al tetraedro t_k no contiene en su interior ningún punto de X . El algoritmo de Watson utiliza esta propiedad para construir una triangulación de Delaunay del conjunto X .

Si consideramos $T_i = \{t_k / 1 \leq k \leq m_i\}$ el conjunto de tetraedros de la triangulación asociada a los i primeros puntos, y t_0 un tetraedro inicial que contiene a todos los puntos de X , el algoritmo de Watson puede resumirse:

0. Comienzo, $i = 0$.
1. Si $i < n$ introducimos el punto x_{i+1} en la triangulación; si no, fin.
2. Hallamos el conjunto $T1_i = \{t_k / b(t_k) \supset x_{i+1}\}$.
3. Eliminamos los tetraedros de $T1_i$.
4. Construimos el nuevo conjunto de tetraedros $T2_i$, formado mediante las caras exteriores de los tetraedros de $T1_i$ y el punto x_{i+1} .
5. $T_{i+1} = (T_i - T1_i) \cup T2_i$, hacemos $i = i+1$ y retornamos al punto 1.

2. PROBLEMAS ASOCIADOS A ERRORES DE REDONDEO.

El algoritmo de Watson presenta dificultades de implementación debido al error que comete el ordenador en el tratamiento de números reales. Los problemas más frecuentes que aparecen son:

a) *Cruces de tetraedros*: haremos las discusiones considerando una triangulación en 2-D, si bien la situación para 3-D es enteramente análoga.

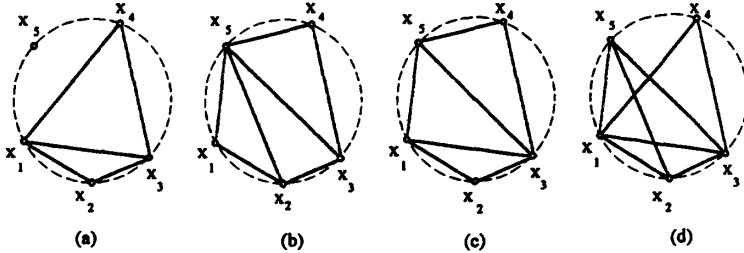


Figura 1. Diferentes triangulaciones posibles al insertar x_5 .

Los puntos x_1, x_2, x_3, x_4 y x_5 están situados sobre la misma circunferencia; al insertar el punto x_5 en la triangulación, Figura 1 (a), y por errores de redondeo en el cálculo de dicha circunferencia, se pueden presentar los siguientes casos:

1. Los triángulos $\{x_1, x_2, x_3\}$ y $\{x_1, x_3, x_4\}$ pertenecen a T_1 ; Figura 1 (b).
2. El triángulo $\{x_1, x_2, x_3\}$ no pertenece a T_1 , y el $\{x_1, x_3, x_4\}$ sí; Figura 1 (c).
3. El triángulo $\{x_1, x_2, x_3\}$ sí pertenece a T_1 , y el $\{x_1, x_3, x_4\}$ no; Figura 1 (d).

Todas las situaciones anteriores son admisibles a excepción de la última en donde aparece un cruce entre triángulos, George [1992].

b) *Formación de tetraedros planos*: en la Fig. 2 suponemos que los puntos x_2, x_4, x_5 y x_6 son coplanarios, el tetraedro $t_1 = \{x_1, x_2, x_4, x_5\}$ no pertenecen a T_1 , pero el tetraedro $t_2 = \{x_2, x_3, x_4, x_5\}$ sí pertenece.

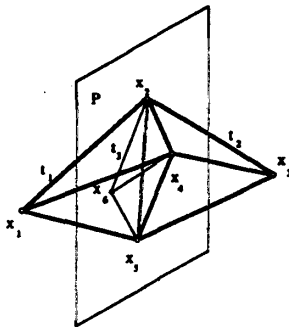


Figura 2. Ejemplo de tetraedro plano al insertar x_5 .

Se formaría el tetraedro plano $t_3 = \{x_2, x_4, x_3, x_6\}$, no admisible en la triangulación. Es más, si el punto x_6 está ligeramente desplazado hacia el

semiespacio izquierdo, se formarán cruces entre los tetraedros t_i y t_j ; en este último caso diremos que el conjunto T_1 , no es en forma de estrella respecto a x_i .

3. CONSTRUCCIÓN DEL CONJUNTO T_1 .

La construcción exacta del conjunto T_1 , no es posible debido a los errores de redondeo antes señalados. Es muy importante determinar este conjunto de forma que se eviten situaciones inadmisibles. Así mismo, la eficacia en la localización de los tetraedros de T_1 , hará que el programa funcione rápidamente.

a) *Búsqueda rápida del tetraedro que "contiene" al punto x_{i+1} (Núcleo).*

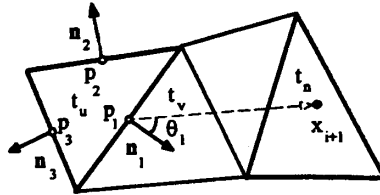


Figura 3. Búsqueda del núcleo, t_u .

La búsqueda del núcleo comienza en el último tetraedro, t_u , instalado en la triangulación T_i . Si llamamos \bar{n}_k ($k=1,2,3,4$) a los vectores unitarios normales salientes a las caras de un tetraedro t , p_k ($k=1,2,3,4$) a los centros geométricos de las caras del tetraedro t , y siendo $\cos(\theta_k) = \bar{n}_k \cdot \overline{p_k x_{i+1}} / \|\overline{p_k x_{i+1}}\|$ ($k=1,2,3,4$), el algoritmo de búsqueda queda de la siguiente forma:

1. $t = t_u$
2. Si $\bar{n}_k \cdot \overline{p_k x_{i+1}} \leq 0$ ($k=1,2,3,4$) $t_u = t$ fin;
si no, encontramos la cara c de t / $\cos(\theta_c) = \text{Max}_{k=1,2,3,4} [\cos(\theta_k)]$.
3. Sea t_c el tetraedro que comparte c con t . Hacemos $t = t_c$ y volvemos a 2.

Si el punto x_{i+1} está muy próximo a una cara o a una arista del tetraedro que lo contiene, consideramos como núcleo los dos tetraedros contiguos a la cara o el conjunto de tetraedros que comparten la arista, respectivamente; George [1992].

b) *Localización del resto de tetraedros de T_1 , a partir del núcleo.*

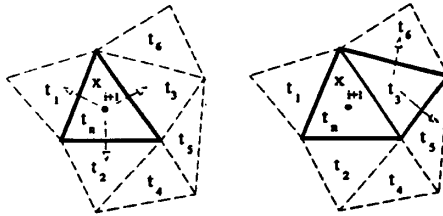


Figura 4. Construcción de T_1 , a partir del núcleo.

Una vez localizado el núcleo t_n , construimos $T1$, aplicando el algoritmo:

1. Sea t_n el núcleo, consideramos $T1_1 = \{t_n\}$.
2. Hallamos $T1_1^* = \{t_k / 1 \leq k \leq n_s\}$, conjunto de tetraedros adyacentes a $T1_1$,
3. Para $k = 1, 2, \dots, n_s$. Si $b(t_k) \supset x_{i+1}$, entonces $T1_2 = T1_1 \cup \{t_k\}$.
4. Si $\exists t_k \in T1_1^* / b(t_k) \supset x_{i+1}$, retornamos al punto 2; si no, fin.

Para evitar la construcción de tetraedros planos o conjuntos $T1$, que no sean en forma de estrella respecto de x_{i+1} , debemos incluir en el algoritmo anterior ciertos controles. Éstos serán explicados mediante el siguiente ejemplo.

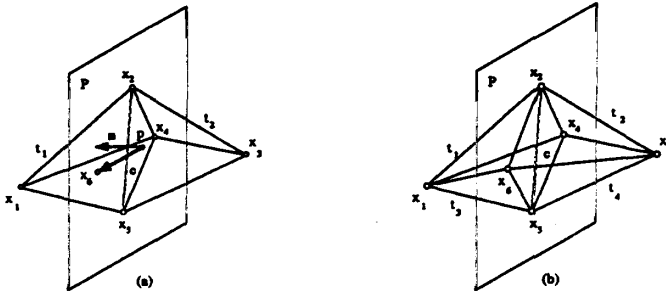


Figura 5. Control sobre tetraedros planos.

Supongamos que el tetraedro $t_1 = \{x_1, x_2, x_4, x_3\}$ no pertenece a $T1$, pero el

$t_2 = \{x_2, x_3, x_4, x_5\}$ si pertenece. Si la cantidad $\bar{n} \cdot \overline{px_5} / \|\overline{px_5}\| < \varepsilon$, el tetraedro que se construyera con el punto x_5 sería muy plano, no sería admisible en la triangulación, Figura 5 (a); en este caso hacemos nulo el radio de la esfera circunscrita al tetraedro t_2 para evitar que éste pertenezca a $T1$, comenzando de nuevo la determinación de $T1_1$. Se forman los nuevos tetraedros $t_3 = \{x_1, x_2, x_3, x_5\}$

y $t_4 = \{x_2, x_3, x_4, x_5\}$; Figura 5 (b). Una vez concluida ésta, de forma que todos los tetraedros sean aceptables, restablecemos los valores originales de los radios. Mediante el procedimiento expuesto en este apartado se consigue construir, en cualquier caso, un conjunto $T1$, admisible para algún valor del parámetro ε .

4. DEFINICIÓN DEL OBJETO. GENERACIÓN DE PUNTOS.

El objeto a mellar se define mediante un conjunto mínimo de puntos que permitan descomponerlo en superficies planas poligonales y regiones volumétricas poliédricas, ambas convexas. Se definen todas las líneas por los puntos extremos que las forman, las superficies poligonales se determinan por sus líneas y las regiones volumétricas por sus superficies. En función de unas densidades previamente especificadas, se generan automáticamente puntos sobre las líneas,

superficies y regiones volumétricas. Los puntos interiores muy próximos al contorno no son insertados a fin de que se respete la geometría del objeto.

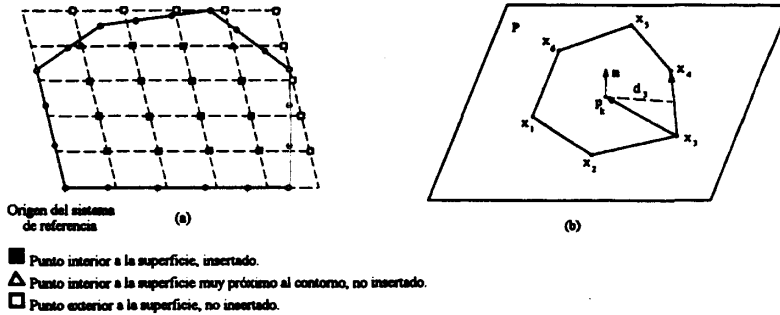


Figura 6. Generación de puntos en una superficie (a). Control en la generación (b).

En la generación de los puntos hay que considerar un control para decidir si los puntos generados se incluyen en el conjunto X . Si llamamos \bar{n} al vector unitario normal a la superficie $\bar{n} = \overline{x_1 x_2} \times \overline{x_1 x_l} / \|\overline{x_1 x_2} \times \overline{x_1 x_l}\|$, el punto p_k está dentro de la

poligonal si y sólo si: $\forall i = 1, 2, \dots, l$ resulta $d_i = [(\bar{n} \times \overline{x_i x_{i+1}}) \cdot \overline{x_i p_k} / \|\overline{x_i x_{i+1}}\|] > 0$,

donde $x_{i+1} = x_1$; la cantidad $|d_i|$ representa la distancia a cada lado de la poligonal, Figura 6 (b). Una idea similar se utiliza para determinar si un punto está dentro de una de las regiones poliédricas en las que se descompone el objeto. Los tetraedros de la triangulación cuyos centros geométricos estén fuera del objeto, son eliminados al final del proceso.

5. APLICACIONES.

En la Figura 7 se muestra la triangulación tridimensional de un dominio no convexo en el que se han fijado diferentes densidades de discretización sobre algunas de sus líneas, superficies y regiones. Para la definición del objeto se han considerado como datos 20 puntos, 36 líneas, 19 superficies y 3 regiones.

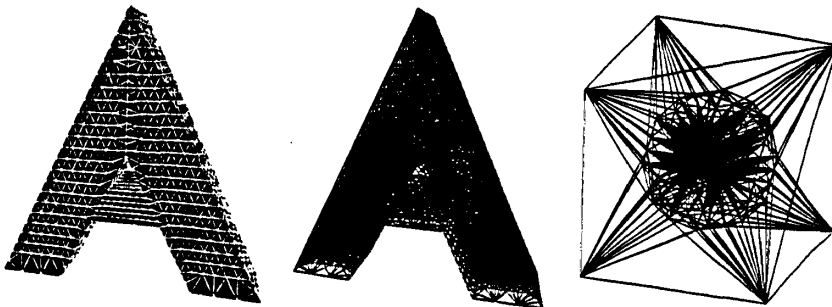


Figura 7. Aplicación del generador a un objeto no convexo.

En la figura de la izquierda se muestra la triangulación de la superficie del objeto, en la central puede verse la malla formada por 1878 nodos y 8126 tetraedros, y a la derecha se observa la malla global donde el objeto se encuentra en el interior de una esfera sobre la que se han definido automáticamente una serie de puntos. Dicha esfera se define con el fin de que los tetraedros transitorios, que se producen en el proceso de inserción de puntos, sean menos degenerados que los que se producirían si solo se considerara el cubo inicial. La malla global está formado por 1948 nodos y 11123 tetraedros. El tiempo de CPU para la creación de la malla fue del orden de 5 segundos en una estación de trabajo HP-730.

AGRADECIMIENTOS

Este trabajo ha sido financiado parcialmente a través de la Fundación Universitaria de Las Palmas por la empresa AHEMON, S.A.

REFERENCIAS

- CAVENDISH, J.C.; FIELD, D.; FREY, W. [1985]: An Approach to Automatic Three Dimensional Finite Element Mesh Generation, *Int. J. Num. Meth. Eng.* **21**, 329-347.
- GEORGE, P.L.; HERMELINE, F. [1992]: Delaunay's Mesh of a Convex Polyhedron in Dimension d. Application to Arbitrary Polyhedra, *Int. J. Num. Meth. Eng.* **33**, 975-995.
- GEORGE, P.L.; HECHT, F.; SALTEL, E. [1991]: Automatic mesh generator with specified boundary, *Comp. Meth. Appl. Mech. Eng.* **92**, 269-288.
- PERRONNET, A. [1988]: Un algorithme de tétraédration d'un objet multi-matériaux ou de l'extérieur d'un objet, *Publications du Laboratoire D'Analyse Numérique*.
- REBAY, S. [1993]: Efficient Unstructured Mesh Generator by Means of Delaunay Triangulation and Bowyer-Watson Algorithm, *J. Comp. Ph.* **106**, 125-138.
- SCHROEDER, W.J. and SHEPARD, M.S. [1990]: Combined Octree/Delaunay Method for Fully Automatic 3-D Mesh Generation, *Int. J. Num. Meth. Eng.* **29**, 37-55.

J.M. Escobar

Departamento de Electrónica y Telecomunicación
 Universidad de Las Palmas de Gran Canaria
 Campus Universitario de Tafira
 35017-Las Palmas de G.C., ESPAÑA

R. Montenegro

Departamento de Matemáticas
 Universidad de Las Palmas de Gran Canaria
 Campus Universitario de Tafira
 35017-Las Palmas de G.C., ESPAÑA