

# OPTIMIZACIÓN DE LA CALIDAD DE MALLAS DE ELEMENTOS FINITOS MEDIANTE CAMBIOS LOCALIZADOS DE TOPOLOGÍA

MARCELO J. VÉNERE

*Centro Atómico Bariloche - CNEA  
8400 - San Carlos de Bariloche, Argentina  
Tel.: + 54-944-45107 Fax: + 54-944-45299  
E-mail: venere@cab.cnea.edu.ar*

## RESUMEN

Se presenta un método que mediante cambios localizados en la topología de una dada malla tridimensional de elementos finitos, logra mejorar la calidad de sus elementos, y en especial la de aquellos con volumen nulo o negativo. Se describirá el algoritmo general y se mencionarán algunos puntos importantes a tener en cuenta cuando la malla presenta estos elementos de volumen negativo. Por último se muestra el comportamiento del método al utilizarlo como optimizador de mallas ya generadas y se incluyen algunas experiencias al emplearlo como generador de mallas en sí.

## LOCAL TOPOLOGICAL MODIFICATIONS FOR FINITE ELEMENT MESH QUALITY OPTIMIZATION

### SUMMARY

A method for element quality improvement on a given finite element mesh is presented. It is based on local changes on the topology of the mesh and is specially well suited for the removal of null or negative volume elements. The general algorithm is described and some important remarks to deal with negative elements are included. Finally the behavior of the method for mesh optimization is shown and some experiences on its use as a mesh generator are discussed.

## INTRODUCCIÓN

La posibilidad de generar mallas de elementos finitos para geometrías tridimensionales complejas es hoy en día un punto crítico para acceder al modelado computacional de gran parte de los problemas que interesan en la industria. En la literatura pueden encontrarse numerosos trabajos que aseguran la factibilidad de generar estas mallas en forma automática. La mayor parte recurren al método Delaunay<sup>1,2</sup>, y en menor medida también se utiliza el método Frontal (Peraire *et al.*<sup>3</sup>, Dari *et al.*<sup>4</sup>) y el de Octrees (Shephard *et al.*<sup>5</sup>).

Recibido: Marzo 1996

Sin embargo casi sin excepción, los algoritmos propuestos adolecen del mismo problema: la aparición de algunos elementos muy distorsionados o incluso con volumen nulo o negativo. Por ejemplo con una implementación robusta del método Delaunay es posible obtener buenos elementos en el interior del dominio, pero para respetar la frontera del mismo será necesario crear elementos en forma forzada, que normalmente tendrán muy mala calidad.

No hace falta remarcar que estos elementos distorsionados, aunque sean pocos, degradarán o incluso invalidarán los resultados que pueden obtenerse con el código de elementos finitos. Ésto ha llevado a muchos investigadores a buscar soluciones a este problema. Una bastante notable es la que propone abandonar el concepto de elemento y trabajar sólo con vinculaciones entre nodos. Ésta sería una alternativa válida, ya que a pesar de los malos elementos, las distribuciones de nodos que se obtienen son buenas. Existen otros trabajos con propuestas menos drásticas, tal como la que se propone en Dari *et al.*<sup>6,7</sup>. En este caso la idea es mover los nodos libres de la malla de forma de optimizar la calidad de los peores elementos.

En el presente trabajo se describe un método que en nuestra experiencia soluciona este problema en forma muy satisfactoria. El mismo es una variante del método propuesto por T. Coupez<sup>8</sup> y básicamente consiste en cambiar la estructura de la malla manteniendo los nodos en su posición; es decir, redefinir que nodos forman cada elemento. Obviamente es impensable analizar todas las posibilidades de formar elementos con la totalidad de nodos de la malla, pero si es factible tomar un subconjunto pequeño de nodos y analizar las diferentes formas de definir elementos con los mismos.

El algoritmo completo se describe en el punto siguiente y a continuación se muestran las estructuras de datos utilizadas, ya que resultan importantes para comprender las dificultades que pueden presentarse. También se introducen distintas formas de definir la calidad de un elemento tipo tetraedro, magnitud que en definitiva es la que queremos optimizar. Luego se discuten los problemas que aparecen cuando la malla tiene elementos con volumen negativo y se proponen algunas soluciones a los mismos. Se incluyen algunos ejemplos que muestran las prestaciones del método y por último se describen algunas experiencias al utilizar este algoritmo como método de generación en sí y no como simple optimizador.

## DESCRIPCIÓN DEL ALGORITMO

Si bien este trabajo fue realizado exclusivamente para el tratamiento de mallas tridimensionales con elementos tipo tetraedros, el algoritmo se describirá en 2D para facilitar su visualización.

Como se mencionó en la introducción, resulta inabordable analizar todas las formas posibles de formar elementos con un dado conjunto de nodos (al menos para mallas de elementos finitos donde el número de nodos no será inferior a varios cientos). Una alternativa entonces sería realizar un análisis en subconjuntos pequeños o *clusters* de nodos como se muestra en la Figura 1. La propuesta realizada por T. Coupez en su trabajo, es analizar los *clusters* formados por todos los elementos que concurren a un dado nodo y por todos los elementos que concurren a una dada arista.

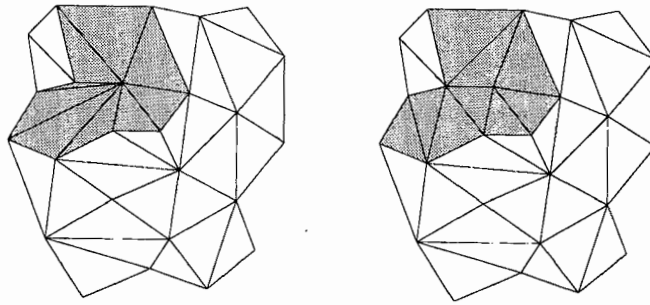


Figura 1. Mejoramiento de la calidad por un cambio local en la topología de la malla

Lamentablemente en 3D, incluso con estos subconjuntos pequeños resulta muy complejo analizar todas las posibilidades de formar elementos. La propuesta entonces es tomar un número más acotado de posibilidades, las cuales se muestran en la Figura 2 y se pueden resumir como:

- Elementos originales
- Elementos formados al unir con el nodo  $i$  a todas las aristas (caras) que no contienen al nodo  $i$ .
- Elementos formados al unir un nuevo nodo  $nc$  en el centro del *cluster* con todas las aristas (caras) de la cavidad.

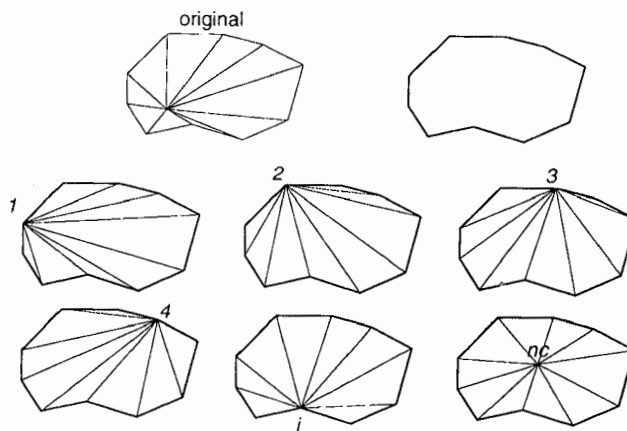


Figura 2. Posibilidades a analizar para formar nuevos elementos

Obsérvese que con este algoritmo, el conjunto de nodos originales se pierde, ya que pueden crearse nuevos y borrarse viejos. Sin embargo, aquellos que están sobre la frontera del dominio se mantendrán en el conjunto, por lo que está garantizado que

se respeta la geometría. Analizando sólo los *clusters* de elementos que concurren a un nodo, el método completo puede describirse de la siguiente forma:

```

mientras ocurran cambios
{
  Para todos los nodos de la malla
  {
    Formar cluster de elementos que concurren al nodo
    Analizar diferentes configuraciones
    Si la mejor configuración no es original, entonces borrar
    viejos elementos, crear nuevos y actualizar estructuras
    de datos
  }
}

```

## ESTRUCTURAS DE DATOS

La implementación eficiente del algoritmo descrito en el punto anterior requerirá una cuidadosa elección de las estructuras de datos a utilizar. Los requerimientos más importantes que deberán cumplir son los siguientes:

- Facilidad para borrar o crear elementos.
- Rápido acceso a los elementos que comparten un nodo dado.
- Rápido acceso a los elementos que comparten una arista dada.

Una estructura que se ajusta bien a los mismos es una lista doblemente encadenada (ver Figura 3), donde cada vértice de la misma representa un elemento de la malla y contiene información de los cuatro nodos que lo forman ( $n_i$ ) y de los cuatro elementos vecinos por las caras ( $v_i$ ).

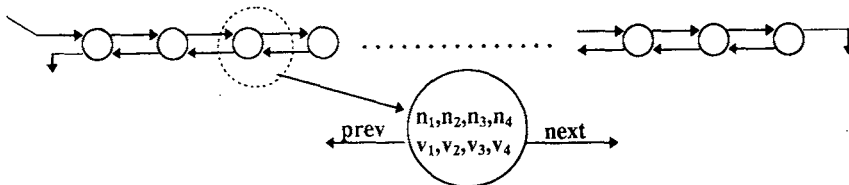


Figura 3. Estructura de datos utilizada

Esta estructura es adecuada siempre que la malla sea *topológicamente válida*. Con ésto se quiere decir que la malla es conexas (es posible visitar todos los elementos de la malla pasando de uno a otro a través de las caras), y que si un elemento  $ei$  tiene como vecino por una cara al elemento  $ej$ , entonces el elemento  $ej$  tiene también como vecino por una cara al elemento  $ei$ . Obsérvese que de esta forma se aceptan

mallas con elementos de volumen negativo (en este caso la malla sería *geoméricamente inválida*), siempre que la misma no contenga caras que sean compartidas por más de dos elementos.

## CRITERIOS DE CALIDAD

Podemos decir que una malla es *geoméricamente válida* para ser utilizada en un análisis por elementos finitos, si no existe superposición de elementos. Es decir, la intersección entre dos elementos cualquiera de la malla es una cara, una arista, un vértice o es nula. Una consecuencia directa de ésto es que en la malla no pueden aparecer elementos con volumen nulo o negativo, condición que es mucho más fácil de verificar que la primera.

En el trabajo de T. Coupez, el algoritmo propuesto no se utiliza como un optimizador de calidad de los elementos, sino como método para generar una malla válida a partir de otra inválida. Por ello para cada *cluster* de elementos se propone minimizar el siguiente indicador  $Q$ :

$$\text{Criterio 1 : minimizar } Q = \sum_{el} |Vol_{el}| \quad (1)$$

El mínimo de este indicador se obtiene cuando los elementos del *cluster* tienen volumen nulo o positivo. Dado que el interés primario de este trabajo es obtener un optimizador que permita remover de una malla a aquellos elementos muy distorsionados (que no necesariamente tienen volumen negativo), se propone optimizar otros criterios de calidad más apropiados.

Es conocido (ver Krizek<sup>9</sup>) que la condición para que una malla sea adecuada para un análisis por elementos finitos, es que los ángulos entre aristas y los ángulos entre caras de los elementos, se mantengan alejados de  $180^\circ$ . Podría entonces tomarse como indicador de calidad de un elemento al mayor de estos ángulos y para cada *cluster* minimizar al mayor de ellos. Es decir:

$$\text{Criterio 2 : minimizar } Q = \max Q_{el} \quad (2)$$

donde  $Q_{el} = \max(\max(\alpha_{aristas}), \max(\alpha_{caras}))$ . Este indicador es relativamente costoso de calcular, y dado que durante el proceso de optimización será necesario hacer esta evaluación un número importante de veces, se propone la utilización de los dos indicadores siguientes, que como se muestra más adelante resultan equivalentes a (2):

$$\text{Criterio 3 : maximizar } Q = \min \left( Q_{el} = 1832.82 \frac{Volumen}{Perimetro^3} \right) \quad (3)$$

$$\text{Criterio 4 : maximizar } Q = \min \left( Q_{el} = \sqrt{72} \frac{Volumen}{(L_{\max})^3} \right) \quad (4)$$

donde  $L_{\max}$  es el tamaño de la mayor de las aristas del tetraedro y por *Perímetro* se entiende la suma de las longitudes de las seis aristas. Las constantes que aparecen en

generar los nuevos elementos para un *cluster* dado, se crea una situación en la que más de dos elementos comparten una arista (situación análoga a que más de dos elementos comparten una cara en 3D). Al tomar como mejor configuración la de unir todas las aristas del *cluster* con el nodo 1, se forma un elemento con volumen negativo superpuesto a otro que ya existía en la malla.

Este problema tiene que ser prevenido y evitado, para lo cual no basta analizar si se está creando un elemento que ya existía en la malla (y eventualmente eliminar el par *elemento-antielemento*). Obsérvese que si la mejor configuración hubiera sido unir todas las aristas al nodo 2, también se habría creado una falla topológica (arista con más de dos elementos), pero no se hubiera creado ningún par *elemento-antielemento*. Por lo tanto la única alternativa es que cada vez que se generan nuevos elementos se verifique que no se esté creando una arista (cara en 3D) que ya está presente en la malla. Esta verificación es bastante más compleja que la de analizar la creación de elementos que ya existían, pero debe ser realizada obligatoriamente para obtener un programa robusto.

Otro factor a tener en cuenta cuando se trabaja con elementos negativos, es que los Criterios 3 y 4 pueden no funcionar correctamente y conducir a situaciones cada vez más negativas.

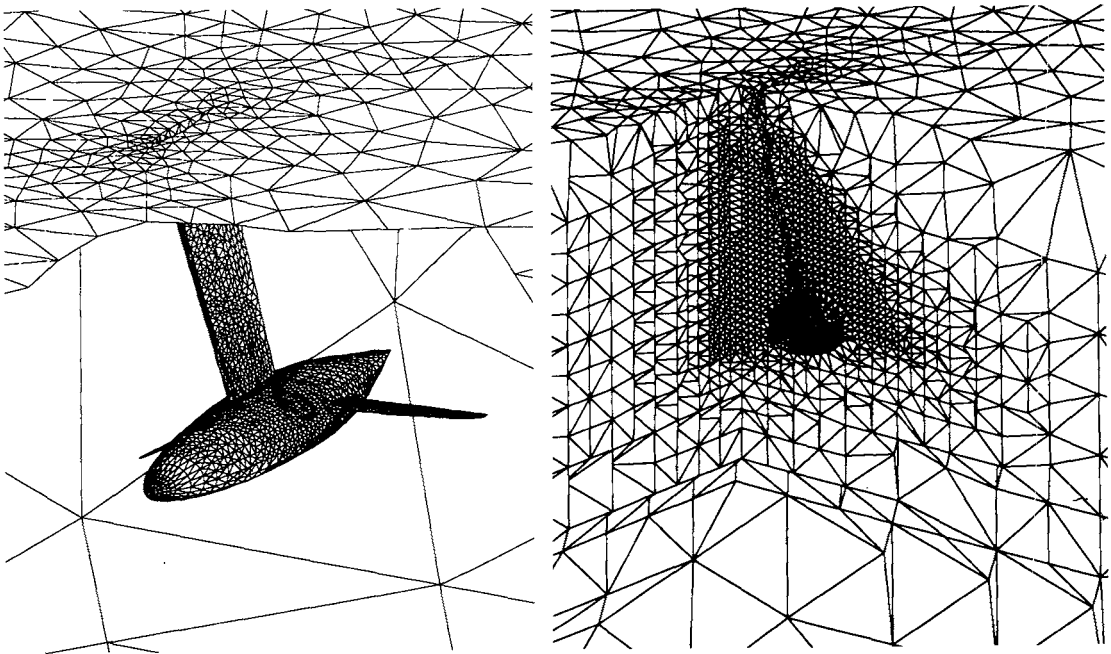


Figura 7. Vista de las mallas de superficie y de volumen correspondiente a la configuración quilla-bulbo-alas de un velero de competición

## GENERACIÓN DE MALLAS

Hemos visto que el método propuesto permite obtener mallas de buena calidad a partir de mallas que incluso pueden presentar varios elementos con volumen negativo. Resulta natural entonces intentar utilizarlo como generador de mallas en sí, partiendo de una malla inicial generada de cualquier forma. La idea más simple para generar esta primera malla consiste en unir a un nodo cualquiera de la superficie del dominio, o eventualmente a uno adicionado a la malla, a todas las caras triangulares que conforman la frontera del mismo. En caso de elegir un nodo de la superficie del dominio, no deben unirse a él aquellas caras de ésta última que lo contuviesen, porque de lo contrario se generarían elementos topológicamente degenerados. Se puede ver un ejemplo de malla inicial en la Figura 8. Si bien este tipo de malla dista de ser válido en el sentido geométrico (casi la mitad de sus elementos son completamente negativos), su validez topológica bastará para poder comenzar el proceso de generación.

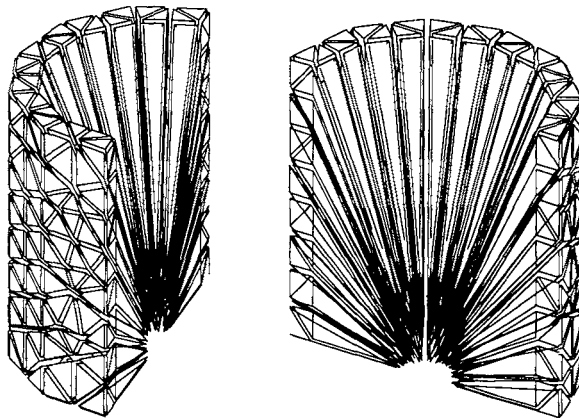


Figura 8. Malla inicial para un semi-tubo

Esta metodología será válida sólo para dominios cuya superficie sea simplemente conexa. Si éste no fuera el caso, es preciso en primer lugar conectar todas las componentes, lo cual puede hacerse como se muestra en la Figura 9, uniendo de a pares de superficies generando un prisma de tres tetraedros (se borran las dos caras de las bases y se agregan las seis caras de los lados).

El proceso de generación debe al menos ser separado en dos partes: una en la que se utiliza el Criterio 1 hasta que no queden elementos con volumen negativo, y otra utilizando el Criterio 3 ó 4 para remover los elementos muy distorsionados. En el análisis realizado, si la superficie de la geometría a discretizar no presentaba grandes partes planas, este algoritmo permitía obtener la malla buscada. Si éste no es el caso, entonces las mallas que se obtienen poseen una gran cantidad de elementos con volumen nulo sobre estas partes planas, los cuales resultan muy difícil de remover.

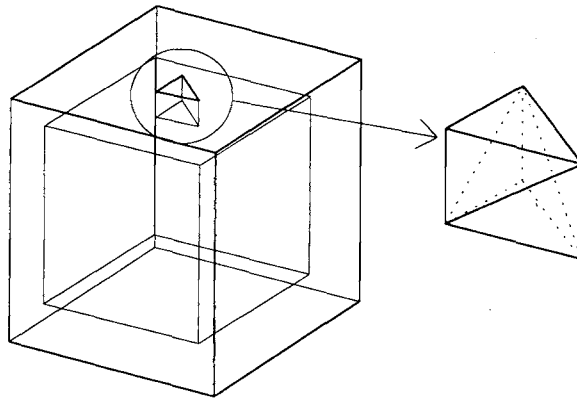


Figura 9. Transformación de una superficie arbitraria en otra simplemente conexa

Para intentar resolver este problema se propuso un nuevo criterio de calidad que consiste en minimizar el número de elementos con volumen nulo o negativo. Observar que tanto el Criterio 1 como el 3 y 4, no ven diferencia entre configuraciones con uno o varios elementos nulos. Alternando este criterio con el 3 o 4, fue posible generar todas las mallas que se intentaron.

## CONCLUSIONES

Se describió un algoritmo que en principio puede funcionar como optimizador o como generador de mallas formadas por elementos tipo simples (triángulos en 2D, tetraedros en 3D). Las experiencias realizadas al utilizarlo como optimizador dieron resultados altamente positivos, lo que resuelve uno de los problemas más serios que presenta la generación automática de mallas tridimensionales.

En el análisis del empleo del algoritmo como generador de mallas no se estudió el problema de cómo controlar la distribución de la discretización, condición necesaria para obtener un verdadero generador de mallas. Ésto se debe a que aún no se consideran satisfactorios los resultados obtenidos en este área. Para considerar el método como apto para generar mallas, el mismo deberá funcionar en forma más automática, tanto en la etapa de obtener la primera malla como en la de obtener elementos positivos y no distorsionados. Especialmente compleja se presenta la automatización de la conexión entre superficies no conexas.

## REFERENCIAS

1. T.J. Baker, "Automatic Mesh Generation for Complex Three Dimensional Regions Using a Constrained Delaunay Triangulation", *Engng. Comp.*, Vol. **5**, pp. 161-175, (1989).
2. P.L. George, F. Hecht y E. Saltel, "Automatic 3D Mesh Generation with Prescribed Meshed Boundaries", *IEEE Trans. Magnetics*, Vol. **26**, pp. 771-774, (1990).



3. J. Peraire, J. Piero, L. Formaggia, K. Morgan y O. Zienkiewicz, "Finite Element Computations in Three Dimensions", *Int. Journal of Numerical Methods in Engineering*, Vol. **26**, pp. 2135-2159, (1988).
4. E.A. Dari, M.J. Vénere y R. Feijoo, "Finite Element Mesh Generation Using the Advancing Front Technique", *Mecánica Computacional*, Vol. **14**, pp. 512-519, (1994).
5. M.S. Shephard, F. Guerioni, J.E. Flaherty, R.A. Ludwig y P.L. Baehmann, "Finite Octree Mesh Generation for Automated Adaptive 3D Flow Analysis", *1st International Conference on Numerical Grid Generation in Computational Fluid Dynamics and Related Fields*, (1988).
6. E.A. Dari y G.C. Buscaglia, "Mesh Optimization: How to Obtain Good Unstructured 3D Finite Element Meshes with Not-So-Good Mesh Generators", *Structural Optimization*, Vol. **8**, pp. 181-184, (1994).
7. P.D. Zavattieri, E.A. Dari y G.C. Buscaglia, "Optimization Strategies in Unstructured Mesh Generation. Submitted to", *International Journal of Numerical Method in Engineering*, (1995).
8. T. Coupez, "A Mesh Improvement Method for 3D Automatic Remeshing", *4th International Conference on Numerical Grid Generation in Computational Fluid Dynamics and Related Fields*, pp. 615-626, (1994).
9. M. Krizek, "On the Maximun Angle Condition for Linear Tetrahedral Elements", *SIAM J. Numer. Anal.*, Vol. **29**, pp. 513-520, (1992).