

# Representación de Volúmenes mediante Agrupación de Tetraedros para la Extracción de Isosuperficies

F. Soler, F. Velasco y A. León

Grupo de Investigación en Informática Gráfica (<http://giig.ugr.es>), Dpto. de Lenguajes y Sistemas Informáticos,  
E.T.S. de Ingenierías Informática y de Telecomunicación, Periodista Daniel Saucedo Aranda, s/n 18071 Granada (Spain)

---

## Abstract

*La extracción de isosuperficies mediante el uso del algoritmo de Marching Cubes es uno de los métodos habituales en la visualización de volúmenes representados a través de un retículo o grid regular de datos. Sin embargo, este algoritmo produce algunas ambigüedades que han sido resueltas mediante diferentes aproximaciones implicando normalmente un proceso complejo. Velasco et al. proponen una tetraedrización adaptativa basada en el comportamiento de la función trilineal dentro de las celdas del modelo que resuelve estas ambigüedades. Dicha tetraedrización es independiente del valor de umbral utilizado y permite preservar la correcta topología de la función trilineal en la isosuperficie aproximada. Además, la transición entre las diferentes configuraciones con ambigüedad ocurre de forma suave frente a cambios en el valor umbral.*

*En este trabajo se propone una partición en poliedros basada en dicha tetraedrización y un diseño para la representación de datos de la malla obtenida. Esta estructura permite reducir considerablemente el número de poliedros obtenidos, interrelacionar los elementos geométricos de la malla, así como poder visualizar isosuperficies y efectuar operaciones de reducción de aristas y poliedros de forma eficiente.*

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling: *Curve, surface, solid, and object representations*

---

## 1. Introducción

Un volumen u objeto volumétrico puede venir determinado mediante un conjunto de muestras en el espacio tridimensional junto con una función para estimar valores en puntos no muestreados, representada por la expresión:

$$( \{ (x, y, z, I) \}, F(x, y, z) : \mathbb{R}^3 \rightarrow \Gamma ) \quad (1)$$

En tales circunstancias, una forma de representación del volumen consiste en una partición en celdas cúbicas de igual tamaño de tal manera que en los vértices de dichas celdas tenemos los valores de propiedad que han sido muestreados, estimando los valores de propiedad en el interior de la celda mediante alguna función de interpolación, normalmente la interpolación trilineal.

Una forma de visualizar un volumen así representado es mediante la extracción y visualizado de una isosuperficie: una superficie contenida en el espacio de volumen que cum-

ple con la propiedad de que todos sus puntos presentan un valor de propiedad fijo que ha sido establecido a priori. Usualmente, este valor de propiedad fijo recibe el nombre de umbral o isovalor. Formalmente:

$$F(x, y, z) = \Gamma_0 \quad (2)$$

Para la extracción de isosuperficies mediante interpolación del valor de propiedad en las celdas del volumen, se recurre comúnmente al popular algoritmo de *Marching cubes* [LC87]. Este método realiza una clasificación de las celdas del volumen comparando el valor de propiedad en los vértices con el valor umbral o isovalor. Así, un vértice se etiqueta como positivo o negativo según su valor de propiedad se encuentre por encima o por debajo del valor umbral. De esta forma, todas las posibles configuraciones de las celdas se clasifican en quince clases de equivalencia. Cada clase (o caso) posee un fragmento característico de isosuperficie representado por una malla de triángulos. Para visualizar la

isosuperficie de un valor umbral determinado, se unen todos los fragmentos que aportan cada una de las celdas. Comúnmente se denominan como *activas* a aquellas celdas o aristas que interconectan vértices positivos y negativos para un determinado valor de umbral y que, por tanto, van a ser atravesadas por la isosuperficie.

Sin embargo, este método presenta ambigüedades en ciertas configuraciones topológicas. Estas ambigüedades han sido estudiadas ampliamente en [Che95], [LB03], [Nie03], [Vel02]. En algunos trabajos como [LB03] y [Nie03] se resuelven las ambigüedades estudiando el valor de propiedad en el interior y en las caras de las celdas usando interpolación trilineal. Como consecuencia, el número de casos de *Marching cubes* aumenta a 31 y 57 respectivamente, si bien, las isosuperficies obtenidas son topológicamente correctas, es decir, topológicamente equivalentes a la isosuperficie que se está aproximando mediante el interpolador trilineal.



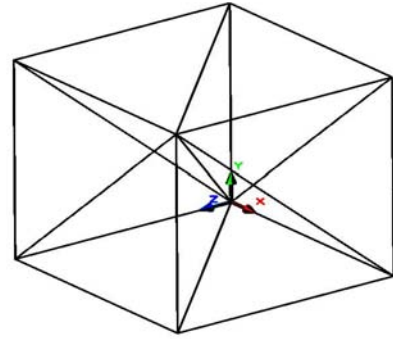
**Figure 1:** Casos de *Marching Tetraedra*.

Otras aproximaciones encaminadas a resolver las ambigüedades de *Marching cubes*, proponen una solución basada en una descomposición de las celdas en tetraedros para, posteriormente, extraer la isosuperficie mediante *marching tetraedra* [PT90]. Este método es similar a *marching cubes* pero considerando tetraedros en vez de celdas o cubos; además posee dos ventajas importantes: sólo existen tres clases de casos y no existen ambigüedades (figura 1).

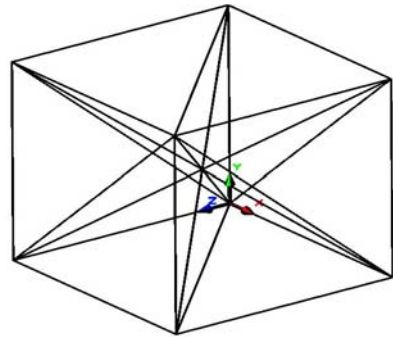
Sin embargo, no todas las topologías correctas de isosuperficie son obtenidas mediante las tetraedrizaciones de otros trabajos como [ZCT95], [GH95], [CP98] o [GP00]. Velasco et al. [VTLS07] proponen un método de tetraedrización basado en el análisis de los puntos de silla de la función volumétrica  $F$  tanto en las caras como en el interior de las celdas. La tetraedrización obtenida es independientemente del umbral de visualización, por lo que no es necesario realizar cambios en la estructura de tetraedros que representa el modelo volumétrico al modificar dicho umbral. Además, la isosuperficie generada cambia de forma suave ante cambios progresivos en el valor de umbral, especialmente entre configuraciones de celda con ambigüedades. Esto permite obtener isosuperficies robustas frente a alteraciones bruscas en el valor umbral.

Este procedimiento hace uso de esos puntos de inflexión para tetraedrización de la celda. Según existan o no puntos de silla se aplica una combinación de una serie de casos base (figura 2) obteniendo un determinado conjunto de tetraedros. Para más detalles acerca de la tetraedrización puede consultarse el trabajo referenciado.

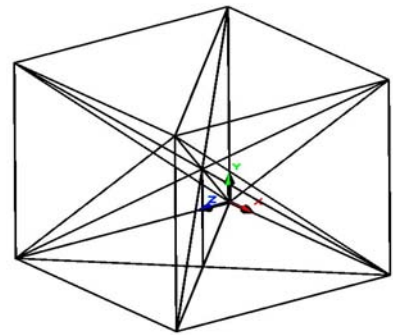
Una vez obtenida una estructura de tetraedros a partir de la



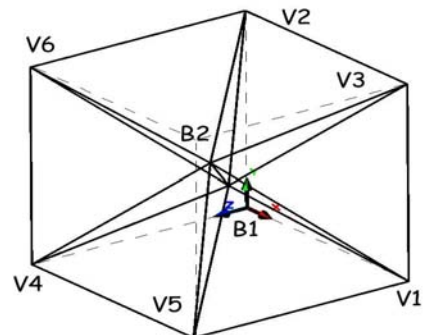
(a) Caso base de una celda sin puntos de silla.



(b) Caso base con punto de silla interno.



(c) Caso base con punto de silla en cara inferior.



(d) Cuboide interior del caso base con dos puntos de silla internos (B1 y B2).

**Figure 2:** Tetraedrización de celdas.

información volumétrica, se puede recurrir a métodos de reducción de la malla, como el propuesto por Chiang [CL03]; dicho trabajo expone una técnica de simplificación progresiva usando operaciones de colapsamiento de aristas. El sistema garantiza la correcta topología de todas las isosuperficies contenidas en el volumen.

Sin embargo, el principal problema de los métodos de tetraedrización, es que se obtienen una alta cantidad de tetraedros. En este trabajo proponemos la agrupación de los mismos en pirámides o cubos donde sea posible, así como el diseño de una estructura de datos que permita representar mallas formadas por este tipo de poliedros. Con esta representación perseguimos varios objetivos: por un lado, reducir en lo posible el número de tetraedros obtenidos en el proceso de tetraedrización inicial; por otro lado facilitar el acceso a los elementos de la malla que estén relacionados entre sí con el fin de poder realizar operaciones de colapsamiento de aristas y/o poliedros de forma eficiente. Por último, se pretende obtener una representación eficiente en cuanto al uso de memoria.

El trabajo que presentamos se organiza de la forma siguiente. En la sección 2 se revisan algunos trabajos previos relacionados con estructuras de datos para la representación de objetos geométricos. En la sección 3 se describe el proceso de partición del volumen en poliedros a partir de la tetraedrización expuesta anteriormente, así como el proceso de visualización de isosuperficies a partir de los poliedros. Posteriormente, en la sección 4 se presenta el diseño de la estructura de datos propuesta, junto con la justificación de la representación de los distintos elementos que la componen. Se finaliza presentando conclusiones y trabajos futuros.

## 2. Trabajos relacionados

Desde la representación de aristas aladas de Baumgart [Bau75] se han ido proponiendo modificaciones que tratan de hacer extensible esta estructura de datos a otros objetos geométricos. Fundamentalmente estos trabajos se pueden clasificar en cuanto a la dimensión de la variedad de los objetos que tratan de representar.

En el ámbito del modelado de objetos 3-variedad, los trabajos de Doblin y Laszlo [DL87] y Lopes y Tavares [LT97] proponen las estructuras *Facet-Edge* y *Handle-Face* respectivamente. Estas estructuras pueden usarse para la representación de mallas de tetraedros describiendo de forma implícita los tetraedros a partir de la codificación del resto de elementos 2-variedad. Lage y Lopes [LLL05] utilizando estos mismos conceptos, proponen la estructura *CHF*. El diseño contempla varios niveles de implementación según los requisitos de rapidez, almacenamiento,...

Por otro lado, en el ámbito de los objetos no-variedad, Floriani y Hui [FH03] presentan la estructura de datos denominada *non-manifold indexed data structure with adja-*

*cencies* que permite representar y manipular objetos no-variedad así como mallas de tetraedros.

Una propuesta más encontramos en el trabajo [CFM\*04] que describen una estructura de datos compacta para representar mallas multiresolución de tetraedros generadas a partir de conjuntos irregulares de datos volumétricos. La estructura que proponen contiene un gran número de referencias para representar las relaciones de adyacencia entre elementos.

Desde la perspectiva de los modelos volumétricos, estos trabajos contemplan en sus representaciones elementos y relaciones topológicas que no son necesarias cuando tratamos con volúmenes de los que se pretende visualizar isosuperficies. Además de esto, la gran cantidad de datos que contiene un modelo volumétrico hace necesario buscar formas de representación que observen especialmente este problema, tratando de reducir en la medida de lo posible el número de referencias entre elementos geométricos.

## 3. Uso de Poliedros para la Simplificación de Tetraedros

El principal problema de los métodos de tetraedrización, es que se obtienen una alta cantidad de tetraedros. En la tabla 1 se muestra una estimación del número de tetraedros generados para algunos tamaños de rejilla con muestras de volúmenes. Se ha supuesto que todas las celdas se clasifican como caso 0-0 de Velasco [VTLS07] (6 tetraedros):

<i>Samples</i>	<i>Cells</i>	<i>Tetrahedra</i>
256 <sup>3</sup>	16.581.375	99.488.250
128 <sup>3</sup>	2.048.383	12.290.298
64 <sup>3</sup>	250.047	1.500.282

**Table 1:** Estadísticas para algunos elementos en volúmenes con 256, 128 y 64 muestras al cubo. Las columnas representan (de izq. a dcha.): tamaño de cada dimensión de la rejilla del volumen, número de celdas resultante de la rejilla y número de tetraedros obtenidos solamente realizando una tetraedrización con el caso 0-0.

Por otro lado, sucede que existen clases de equivalencia de nuestra partición inicial que no pueden ocurrir a causa de la configuración de valores de propiedad; se pueden consultar los trabajos de Chernyaev [Che95] y Nielson [Nie03] para un estudio sobre la imposibilidad de ocurrencia de dichos casos. Además unas clases de equivalencia se dan con una mayor frecuencia que otras. Estas conclusiones se obtienen empíricamente tras realizar una variación con repetición sobre los ocho vértices de una celda en un rango de valores de propiedad.

En la práctica, a la hora de visualizar isosuperficies de modelos procedentes de TACs y, posiblemente debido a su convexidad, el caso que más aparece es el caso 0-0. En algunos modelos, geoméricamente similares a una esfera, la proporción alcanza el 100 % de los casos. En la tabla 2 se

presentan los datos obtenidos en cuatro modelos volumétricos bien conocidos cuyas imágenes mostramos en la figura 8 del Anexo.

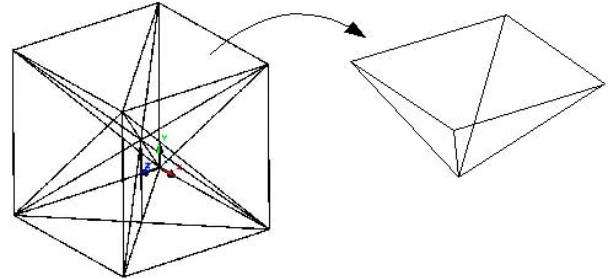
Model:	SPHERE		BABY HEAD	
G.Size:	256x256x256		256x256x98	
C. 0-0:	16.581.299	100,00 %	5.711.996	90,56 %
C. 0-1:	88	0,00 %	41.853	0,66 %
C. 1-1:	0	0,00 %	379.014	6,01 %
C. 2-1:	0	0,00 %	134.875	2,14 %
C. 3-1:	0	0,00 %	38.800	0,62 %
C. 4-1:	0	0,00 %	310	0,00 %
C. 5-1:	0	0,00 %	0	0,00 %
C. 6-1:	0	0,00 %	552	0,01 %
C. 0-2:	0	0,00 %	0	0,00 %
C. 1-2:	0	0,00 %	0	0,00 %
C. 2-2:	0	0,00 %	0	0,00 %
C. 3-2:	0	0,00 %	0	0,00 %
C. 4-2:	0	0,00 %	0	0,00 %
C. 5-2:	0	0,00 %	0	0,00 %
C. 6-2:	0	0,00 %	25	0,00 %
Total:	16.581.375		6.307.425	
Model:	ENGINE		BONSAI	
G.Size:	256x256x256		256x256x91	
C. 0-0:	15.589.604	94,02 %	4.946.950	84,53 %
C. 0-1:	50.658	0,31 %	78.800	1,35 %
C. 1-1:	699.964	4,22 %	645.220	11,03 %
C. 2-1:	217.012	1,31 %	132.103	2,26 %
C. 3-1:	23.647	0,14 %	44.666	0,76 %
C. 4-1:	247	0,00 %	1.682	0,03 %
C. 5-1:	0	0,00 %	0	0,00 %
C. 6-1:	237	0,00 %	2.281	0,04 %
C. 0-2:	0	0,00 %	0	0,00 %
C. 1-2:	0	0,00 %	0	0,00 %
C. 2-2:	0	0,00 %	0	0,00 %
C. 3-2:	0	0,00 %	0	0,00 %
C. 4-2:	0	0,00 %	0	0,00 %
C. 5-2:	0	0,00 %	0	0,00 %
C. 6-2:	6	0,00 %	548	0,01 %
Total:	16.581.375		5.852.250	

**Table 2:** Ocurrencia de los distintos casos de la tetraedrización propuesta por Velasco en algunos modelos volumétricos. Para cada modelo, la primera fila muestra el tamaño de la rejilla, la primera columna muestra el número de celdas clasificadas en cada caso y la segunda columna presenta el porcentaje del modelo que representa la primera columna.

Con todo esto, la propuesta que hacemos es no particionar las celdas cuando no sea necesario. Así, una celda cúbica que no presenta puntos de silla no va a ser ambigua para ningún valor umbral y por tanto se puede dejar como cubo y visualizarla mediante el *Marching Cubes* de Lorensen ya que siempre va a ser clasificada como uno de los casos sin ambigüedad. Así todos los cubos de nuestra propuesta inicial pertenecientes al caso 0-0 y que generan 6 poliedros tipo

tetraedro, se dejan tal cual, generando un solo poliedro tipo cubo.

De igual modo, como se observa en la cara superior de la figura 3, si una cara no es ambigua, no es necesario particionarla en dos tetraedros, sino que se puede representar mediante un poliedro tipo pirámide, obteniendo una reducción de un poliedro por cara no ambigua.



**Figure 3:** Agrupación de los dos tetraedros de la cara superior de la celda en un único objeto con topología de pirámide.

Realizando esta agrupación de tetraedros podemos obtener una reducción en el número de poliedros muy significativa, tal y como mostramos en la tabla 3.

	Vel. T.	T. Poly.	Tetra.	Pyram.	Cube	Red.
Case 0-0:	6	1	0	0	1	83,33 %
Case 0-1:	12	6	0	6	0	50,00 %
Case 1-1:	14	9	4	5	0	35,71 %
Case 2-1:	16	12	8	4	0	25,00 %
Case 3-1:	18	15	12	3	0	16,67 %
Case 4-1:	20	18	16	2	0	10,00 %
Case 5-1:	22	21	20	1	0	4,55 %
Case 6-1:	24	24	24	0	0	0,00 %
Case 0-2:	18	12	6	6	0	33,33 %
Case 1-2:	20	15	10	5	0	25,00 %
Case 2-2:	22	18	14	4	0	18,18 %
Case 3-2:	24	21	18	3	0	12,50 %
Case 4-2:	26	24	22	2	0	7,69 %
Case 5-2:	28	27	26	1	0	3,57 %
Case 6-2:	30	30	30	0	0	0,00 %

**Table 3:** Número de poliedros para cada caso de la partición propuesta por Velasco. En cada columna mostramos (de izquierda a derecha): caso de la partición, número de tetraedros obtenidos por el método de Velasco, el número total de poliedros, pirámides, tetraedros y cubos usando nuestra propuesta, y el índice de reducción obtenido.

Como se observa, el mayor índice de reducción se obtiene en los casos observados más frecuentemente según la tabla 2. En la tabla 4 se muestra la reducción obtenida para los modelos usados como ejemplo. Hay que advertir cómo la convexidad característica de los modelos volumétricos de la



esfera, la cabeza del bebé o el motor nos proporciona un mayor ratio de simplificación. Incluso con modelos no tan convexos como el Bonsai, se obtiene un buen resultado.

	<i>Tetrahedra</i>	<i>Polyhedra</i>	<i>% Reduction</i>
<i>SPHERE</i>	99.488.778	16.581.815	83,33 %
<i>BABY HEAD</i>	42.957.006	11.594.318	73,01 %
<i>ENGINE</i>	107.853.662	25.162.391	76,67 %
<i>BONSAI</i>	42.682.840	13.583.416	68,18 %

**Table 4:** Reducción obtenida en los modelos de ejemplo pasando agrupación de poliedros. Para cada modelo, la primera columna muestra la cantidad de tetraedros obtenida por el método de Velasco, la segunda muestra el número de poliedros obtenidos mediante nuestra propuesta y la tercera presenta el ratio de reducción.

Además de estas ventajas, este tipo de agrupamiento ocasiona una reducción simultánea en el número de triángulos de las isosuperficies generadas. En la tabla 5 mostramos la reducción obtenida en número de triángulos usando agrupación de poliedros frente a la tetraedrización original.

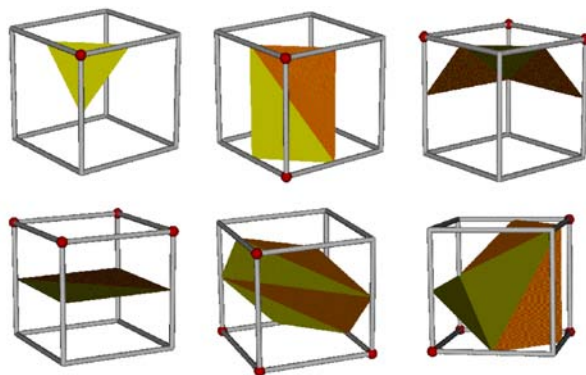
	<i>Thr.</i>	<i>MC</i>	<i>T. Polyh.</i>	<i>T. Velasco</i>	<i>% Red.</i>
SPHERE	30,0	540.536	540.536	1.617.672	66,59 %
BABY H.	30,0	434.080	464.939	1.386.363	66,46 %
ENGINE	30,0	622.482	670.550	2.052.563	67,33 %
BONSAI	4,0	324.898	487.342	1.069.075	54,41 %

**Table 5:** Reducción de triángulos obtenida en los modelos de ejemplo. De izquierda a derecha mostramos: modelo, umbral de la isosuperficie generada, número de triángulos obtenidos mediante el método original de Marching Cubes [LC87], mediante nuestra propuesta, y mediante el método de Velasco, y finalmente el ratio de reducción comparando agrupación de poliedros frente a tetraedrización original.

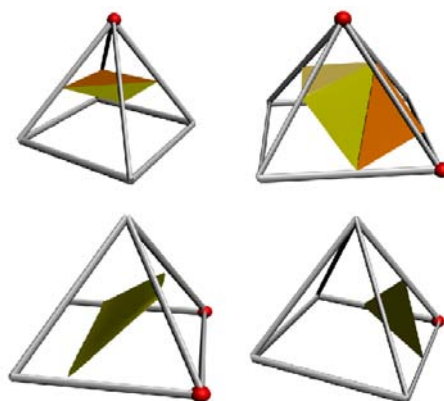
Para extraer los fragmentos de isosuperficie de cada poliedro tendremos que:

- Para tetraedros: uso de *marching tetrahedra* comentado anteriormente (Figura 1).
- Para cubos: usamos el método de *marching cubes* pero en una versión simplificada, ya que ahora podemos obviar todos los casos que presentan ambigüedades. De esta forma sólo contemplamos un subconjunto de las clases de equivalencia del algoritmo original de Lorensen [LC87] (Figura 4).
- Para pirámides: usamos un *marching* con las 5 clases de equivalencia que aparecen en la Figura 5.

La isosuperficie que obtenemos es topológicamente correcta obteniendo los mismos casos que contempla la propuesta de Lopes [LB03]. De forma similar a como ocurre en



**Figure 4:** Subconjunto de casos de Marching Cubes que no presentan caras ambiguas.



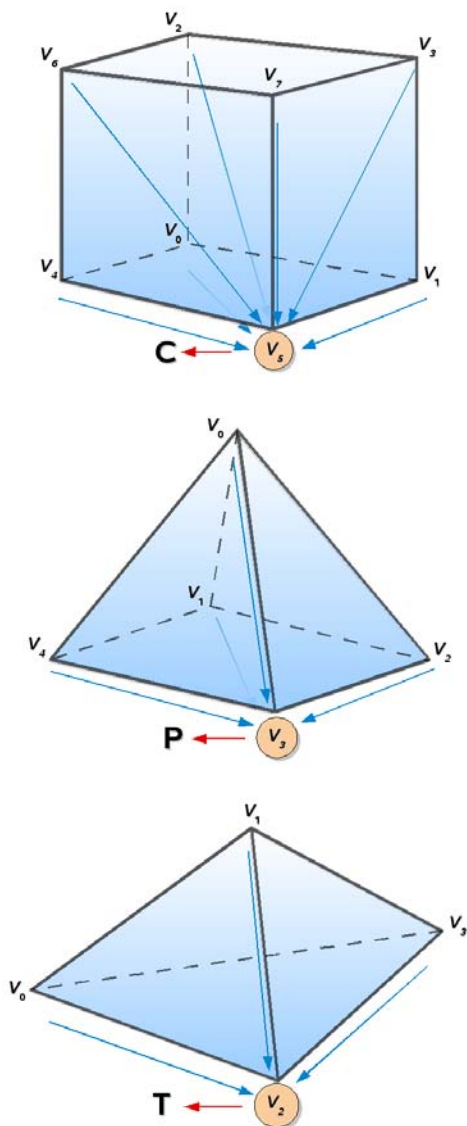
**Figure 5:** Casos de equivalencia de marching Pirámide.

dicho proceso, con variaciones suaves del umbral, la isosuperficie cambia de forma suave también, especialmente en valores cercanos a los que existen en los puntos de silla. La principal ventaja de nuestro método frente al de Lopes es que no es necesario estudiar de nuevo la ambigüedad en las celdas o poliedros cuando el valor de umbral cambia. En nuestra propuesta se realiza un preprocesamiento previo donde se determinan los puntos de silla y la partición en poliedros. Esta información no varía aún cuando el valor de umbral cambie.

#### 4. Representación de la Malla de Poliedros mediante Ordenación por Valor de Propiedad de Vértices

Con el fin de poder realizar operaciones de reducción en la estructura de poliedros o bien integrar estos elementos en una estructura jerárquica para conseguir más adaptabilidad o elementos multiresolución, se hace necesario encontrar una representación para el conjunto de poliedros obtenidos de un determinado volumen. La representación que buscamos debe también de ser capaz de expresar (explícita o implícitamente) relaciones de adyacencia entre los distintos ele-





**Figure 7:** Estructura de referencias en los distintos poliedros. Las flechas azules indican referencias entre vértices y las flechas rojas indican referencias desde los vértices representativos hacia las tablas de registros de poliedros.

De este modo, la estructura de los registros de la tabla de vértices queda como sigue:

- Coordenadas de la posición del vértice.
- Coordenadas del vector normal en dicho punto.
- Valor de propiedad en el vértice.
- Referencias a otros elementos geométricos relacionados con él. En concreto estas referencias pueden ser, bien a otro vértice: en el caso de que aquel vértice sea representativo de un poliedro en común a ambos elementos. O bien

a un poliedro: en el caso de que el vértice en cuestión sea representativo para dicho poliedro.

Con esta representación podemos encontrar cualquier relación de adyacencia entre elementos que necesitemos:

- A partir de un poliedro, obtenemos todos los vértices que lo forman que están directamente indexados en el registro correspondiente.
- A partir de un vértice  $v$ , podemos obtener todos los poliedros adyacentes. Para esto, obtenemos, los vértices representativos de poliedros a los cuales se tiene acceso desde  $v$ . De los poliedros a los que referenciamos estos vértices representativos se seleccionan aquellos que posean una referencia a  $v$ .
- A partir del conjunto de poliedros adyacentes a un vértice  $v$ , podemos encontrar los vértices adyacentes a dicho vértice.
- A partir de un poliedro  $p$ , podemos obtener los poliedros adyacentes mediante el conjunto de poliedros adyacentes a los vértices que lo delimitan.

Para analizar el ahorro de espacio que hacemos con este sistema podemos tomar como referencia una estructura trivial que referencie cada objeto con sus vértices y cada vértice con los objetos de los que forma parte. Hay que tener en cuenta que, en la práctica, en una representación de un modelo volumétrico mediante una malla de tetraedros se cumple que (Cignoni, [CFM\*04]): (3):

$$n_t \approx 6n_v \tag{3}$$

siendo  $n_t$  el número de tetraedros y  $n_v$  el número de vértices de la malla. En nuestra propuesta, las referencias a vértices desde cada uno de los registros de poliedros son las mismas, esto es, no se realiza ninguna reducción en el tamaño que se necesita en la estructura trivial.

Sin embargo, cuando tratamos las referencias desde los vértices a los poliedros adyacentes nuestro método realiza una reducción considerable. La explicación viene del hecho de que cada referencia entre dos vértices  $v_1$  y  $v_2$  (siendo por tanto  $v_2$  un vértice representativo) es reutilizada para expresar una relación de adyacencia en todos los poliedros que contienen la referencia que une  $v_1$  y  $v_2$ . Esto es especialmente importante cuando se considera realizar simplificaciones mediante reducción de aristas en la malla, ya que este tipo de operaciones incrementa enormemente el grado de adyacencias en las aristas de los poliedros. Por tanto, cuanto mayor sea el número de adyacencias entre elementos, esta representación será más eficiente.

Comparándolo con otras representaciones citadas en la sección se trabajos relacionados, se obtiene una mejora sustancial. El más cercano en cuanto a eficiencia de uso de memoria es la representación que proponen Floriani y Hui [FH03]. En esta propuesta, cuando se trata de representar una malla de (sólo) tetraedros, se contempla un coste, en referencias, de  $48n_v$ , siendo  $n_v$  el número de vértices

de la malla. En nuestra estructura, tendremos un coste estimado de  $4n_t + n_t + kn_t$ , siendo  $n_t$  el número de tetraedros. En esta expresión el primer término de la suma proporciona el coste de las referencias de cada registro de la tabla de tetraedros (Figura 6(a)). El segundo término representa la referencia al poliedro desde el vértice representativo. El tercer término proporciona el coste de las referencias entre vértices.  $k$  un parámetro que dependerá del grado de adyacencias entre poliedros y que tendrá valores en el intervalo  $]0, 3[$ . En la tabla 6 mostramos el valor de este parámetro medido en los modelos de ejemplo que hemos utilizado.

	Tamaño	T. Velasco.	Referencias	k
SPHERE	256x256x256	99.488.778	115.803.903	1,16
BABY H.	256x256x98	42.957.006	47.066.802	1,10
ENGINE	256x256x256	107.853.663	120.908.997	1,12
BONSAI	256x256x91	42.682.840	45.979.612	1,08

**Table 6:** Valores para la constante  $k$  obtenidos en los modelos de ejemplo. En las columnas se muestra, de izquierda a derecha, tamaño del retículo del modelo, tetraedros obtenidos mediante el método de Velasco, número de referencias desde vértices a los vértices representativos de cada poliedro adyacente, y valor de la constante  $k$  obtenido.

Como se aprecia, los valores más comunes son los próximos a  $k = 1, 10$ . Así, usando la expresión 3, obtenemos un coste total, en el teórico peor de los casos de  $48n_v$ , similar al de Floriani y Hui. En la práctica, para valores de  $k$  como los obtenidos, el coste se aproximará a  $36n_v$ , lo que supone una reducción de un 25% del coste de almacenamiento. Si además tenemos en cuenta que, como ya hemos indicado anteriormente, estos costes se refieren a mallas de tetraedros y no poliedros más grandes como los que hemos propuesto, este coste se reduce aún más.

## 5. Conclusiones y Trabajos Futuros

A lo largo de este trabajo hemos presentado un método de representación de modelos volumétricos mediante mallas de poliedros que hace posible la visualización de isosuperficies. Estas isosuperficies reflejan correctamente el comportamiento que presenta la función de interpolación trilineal dentro de las celdas del retículo, garantizando la correcta topología de isosuperficie aproximada mediante dicho interpolador. Además, estas aproximaciones sufren alteraciones sin brusquedades frente a cambios suaves en el umbral incluso en valores cercanos a los de los puntos de silla.

Junto con esta forma de agrupamiento de poliedros, hemos presentado una estructura de datos que permite adaptar la representación y búsqueda de elementos de la malla a los problemas concretos que se puedan plantear. Dicha búsqueda es capaz de obtener de forma local, elementos adyacentes a otros con el fin de poder facilitar en un futuro operaciones

de reducción de elementos. Además, no es necesario efectuar cambios en la estructura cuando cambia el valor de umbral.

Según el análisis realizado, el problema fundamental va a ser el tamaño en memoria de la estructura final. Por este motivo, las líneas de trabajo futuro se encaminan a optimizar el uso de memoria y a buscar modelos jerárquicos que permitan simplificar los elementos de la estructura, así como obtener modelos multiresolución a partir de la misma. Entre los mecanismos de simplificación que contemplamos está la simplificación de aristas, que ha sido ampliamente descrita en muchos trabajos ([CFM\*04], [CL03]...). Otra alternativa posible se encuentra en el uso de estructuras jerárquicas que permitan salvaguardar la topología de la isosuperficie [CL03].

## Agradecimientos

Queremos agradecer a Carlos Ureña sus útiles sugerencias durante el desarrollo de esta línea de investigación. Este trabajo ha sido parcialmente financiado por el Ministerio de Ciencia y Tecnología y los fondos FEDER, a través del proyecto TIN2007-67474-C03-02 y por la Consejería de innovación Ciencia y Empresa de la Junta de Andalucía a través del proyecto de excelencia TIC-401.

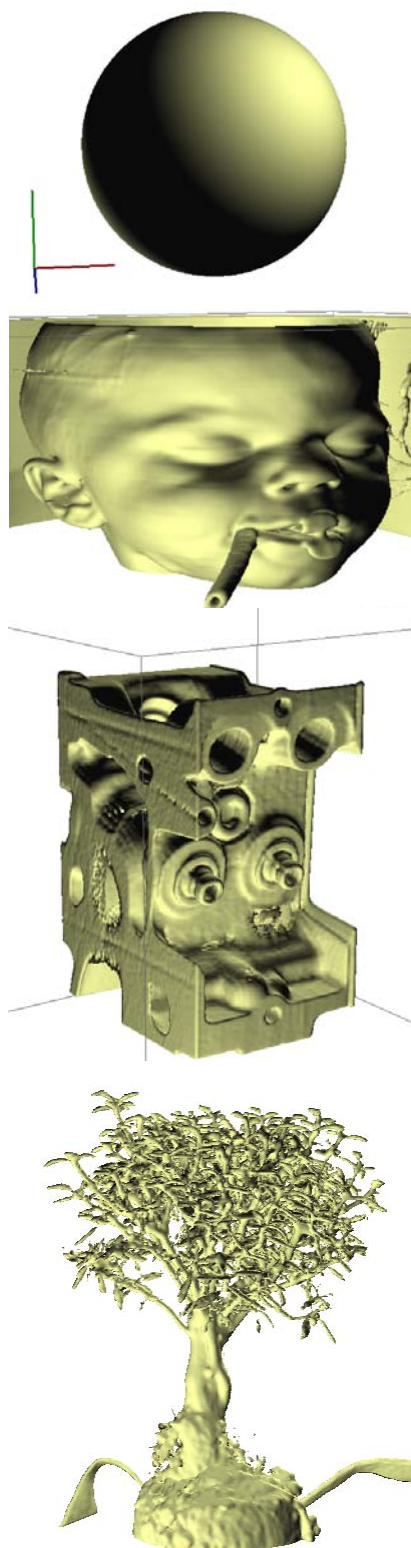
## References

- [Bau75] BAUMGART B. G.: Winged-edge polyhedron representation for computer vision. In *National Computer Conference* (May 1975).
- [CFM\*04] CIGNONI P., FLORIANI L. D., MAGILLO P., PUPPO E., SCOPIGNO R.: Selective refinement queries for volume visualization of unstructured tetrahedral meshes. *IEEE Transactions on Visualization and Computer Graphics* 10, 1 (2004), 29–45.
- [Che95] CHERNYAEV E.: Marching cubes 33: Construction of topologically correct isosurfaces, 1995.
- [CL03] CHIANG Y., LU X.: Progressive simplification of tetrahedral meshes preserving all isosurface topologies, 2003.
- [CP98] CHAN S. L., PURISIMA E. O.: A new tetrahedral tessellation scheme for isosurface generation. *Computers* 22, 1 (1998), 83–90.
- [DL87] DOBKIN D. P., LASZLO M. J.: Primitives for the manipulation of three-dimensional subdivisions. In *SCG '87: Proceedings of the third annual symposium on Computational geometry* (New York, NY, USA, 1987), ACM Press, pp. 86–99.
- [FH03] FLORIANI L. D., HUI A.: A scalable data structure for three-dimensional non-manifold objects. In *SGP '03: Proceedings of the 2003 Eurographics/ACM SIGGRAPH symposium on Geometry processing* (Aire-la-Ville, Switzerland, Switzerland, 2003), Eurographics Association, pp. 72–82.



- [GH95] GUEZIEC A., HUMMEL R.: Exploiting triangulated surface extraction using tetrahedral decomposition. *IEEE Transactions on Visualization and Computer Graphics* 1, 4 (December 1995), 328–342.
- [GP00] GERSTNER T., PAJAROLA R.: Topology preserving and controlled topology simplifying multiresolution isosurface extraction. In *Proceedings Visualization 2000* (2000), Ertl T., Hamann B., Varshney A., (Eds.), pp. 259–266.
- [LB03] LOPES A., BRODLIE K.: Improving the robustness and accuracy of the marching cubes algorithm for isosurfacing. *IEEE Transactions on Visualization and Computer Graphics* 09, 1 (2003), 16–29.
- [LC87] LORENSEN W. E., CLINE H. E.: Marching cubes: A high resolution 3d surface construction algorithm. In *SIGGRAPH '87: Proceedings of the 14th annual conference on Computer graphics and interactive techniques* (New York, NY, USA, July 1987), vol. 21, ACM Press, pp. 163–169.
- [LLV05] LAGE M., LEWINER T., LOPES H., VELHO L.: Chf: A scalable topological data structure for tetrahedral meshes. In *SIBGRAPI* (2005), pp. 349–356.
- [LT97] LOPES H., TAVARES G.: Structural operators for modeling 3-manifolds. In *SMA '97: Proceedings of the fourth ACM symposium on Solid modeling and applications* (New York, NY, USA, 1997), ACM Press, pp. 10–18.
- [Nie03] NIELSON G. M.: On marching cubes. *IEEE Trans. Vis. Comput. Graph.* 9, 3 (2003), 283–297.
- [PT90] PAYNE B. A., TOGA A. W.: Medical imaging: Surface mapping brain function on 3d models. *IEEE Comput. Graph. Appl.* 10, 5 (1990), 33–41.
- [Vel02] VELASCO F.: *Representación y Visualización de Datos Volumétricos*. PhD thesis, Universidad de Granada, Spain, 2002.
- [VTLS07] VELASCO F., TORRES J. C., LEÓN A., SOLER F.: Adaptive cube tessellation for topologically correct isosurfaces. In *Grapp 2007: Proceedings of the Second International Conference on Computer Graphics Theory and Applications* (2007), Braz J., Vázquez P.-P., Pereira J. M., (Eds.), INSTICC - Institute for Systems and Technologies of Information, Control and Communication, pp. 210–219.
- [ZCT95] ZHOU Y., CHEN W., TANG Z.: An elaborate ambiguity detection method for constructing isosurfaces within tetrahedral meshes. *j-COMPUTERS-AND-GRAPHICS* 19, 3 (May–June 1995), 355–364.

## Anexo



**Figure 8:** Isosuperficies de los modelos utilizados como ejemplo.