

Marching Octahedra

J.C. Torres, F. Soler, F. Velasco, A. León, G. Arroyo

Abstract

La extracción de isosuperficies es un problema esencial en visualización de volúmenes. La mayor parte de los métodos propuestos resuelven el problema usando una estrategia de subdivisión en celdas disjuntas que forman un recubrimiento del volumen a tratar, de tal forma que el problema se resuelva independientemente para cada celda. Las celdas más usadas son las cúbicas. No obstante hay problemas de ambigüedad que pueden originar fisuras en la isosuperficie.

En este trabajo presentamos un nuevo método de teselación del volumen utilizando celdas octaédricas, sobre una rejilla regular, y se describe un algoritmo de extracción para esta estructura. Este método no necesita de información adicional para particionar en celdas el volumen, no genera problemas de ambigüedad y emplea un tiempo muy reducido para la extracción de isosuperficie en comparación con otros métodos.

1. Introducción

La extracción de isosuperficies parte de la idea de determinar el conjunto de puntos incluidos en el volumen cuyo valor de propiedad es igual a un valor umbral fijado a priori. De forma general, el volumen puede estar definido mediante expresiones matemáticas o mediante representaciones discretas de datos volumétricos. Nuestro trabajo se centra en las representaciones del segundo tipo. La técnica de extracción de isosuperficies aplicada a representaciones volumétricas discretas permite aproximar una isosuperficie definida a partir de la ecuación $F(x, y, z) = n$, en donde $F(x, y, z)$ es la función de interpolación de la representación y n es el isovalor que determina dicha isosuperficie.

La técnica proporciona puntos de la isosuperficie, que se unen formando una malla de triángulos que será la aproximación final resultante.

Podemos considerar que una representación volumétrica discreta está formada por un conjunto finito de muestras, que tienen un valor de propiedad asociado, y una función que permite interpolar el valor de la propiedad en el espacio. La distribución de las muestras en el espacio condiciona la forma de la función de interpolación. Una forma simple de definir la función de interpolación es formar regiones poliédricas delimitadas por muestras e interpolar en cada una de ellas los valores de las muestras que delimitan la región. Estas regiones se denominan celdas. Cuando las muestras se distribuyen de forma regular en el espacio es fácil construir celdas cúbicas. En cualquier caso, las celdas deben formar un recubrimiento del espacio, es decir cada punto del espa-

cio debe estar en una y solo una celda. Además es necesario que exista continuidad en la función de interpolación en la frontera de celdas adyacentes.

Se han propuesto diferentes algoritmos de extracción de isosuperficies que procesan el volumen celda a celda. El más conocido de ellos es el Marching Cubes [LC87], propuesto originalmente por Wyvill [WMW86].

Desafortunadamente, Marching Cubes puede presentar inconsistencias topológicas entre celdas vecinas. Varios trabajos han abordado este problema utilizando algún tipo de información adicional para resolver las inconsistencias, manteniendo la celda cúbica como elemento de procesamiento para la extracción [NB93]. Otros investigadores han optado por trabajar con aproximaciones en las que la celda que se procesa es un tetraedro [CP98, TPG99]. Varios trabajos descomponen la celda cúbica inicial en 5, 6 o más tetraedros que son usados para realizar la extracción, y la función trilineal que se usa para la celda se aproxima usando una interpolación baricéntrica en cada tetraedro. Se han utilizado descomposiciones de la celda cúbica en 5 tetraedros [AC97, Blo88, Blo94, MH90, NF97, NS97, NB93, ST90]. Así mismo, se han utilizado descomposiciones en 6 tetraedros [NB93]. Esta descomposición se conoce como "Freudenthal subdivision". Bloomenthal [Blo88] planteó una descomposición en 12 tetraedros y otros trabajos han extendido dicha descomposición a 14, 16, 18, 20, 22 y 24 tetraedros [AC97, CSA00, WB96].

Este trabajo propone un nuevo método de extracción que sigue una estrategia de Marching, utilizando celdas octaé-

dricas. La sección siguiente revisa los métodos de extracción basados en esta estrategia propuestos previamente. En la sección tercera presentamos nuestro método de teselación del volumen. La sección cuarta describe el algoritmo de Marching sobre celdas octaédricas. Por último, la sección 5 presenta resultados de la evaluación del método.

2. Algoritmos de extracción de isosuperficies

Los algoritmos de extracción basados en procesamiento independiente de celdas cúbicas suelen producir superficies con discontinuidades en las fronteras de las celdas debidas a la utilización de datos diferentes para generar la isosuperficie en cada una de las celdas vecinas. Las técnicas propuestas hasta la fecha para resolver este problema han trabajado en tres direcciones diferentes:

- Compartir información entre celdas para forzar una generación consistente.
- Generar información adicional en la frontera común que elimine la ambigüedad.

La primera opción rompe la propia filosofía del proceso de generación basado en el procesamiento independiente de las celdas, degradando su rendimiento. La segunda, requiere realizar cálculos adicionales y supone, conceptualmente, asumir un determinado comportamiento del campo escalar en el interior de la celda.

El origen de estos problemas esta en el hecho de que la frontera común entre celdas sea un cuadrado, ya que existen configuraciones en las que la intersección de la isosuperficie con la cara no se puede determinar de forma unívoca sin información adicional [NY06].

Una forma de resolver el problema es utilizar celdas delimitadas por caras triangulares. Esto se puede conseguir usando tetraedros [TPG99]. El inconveniente de esta opción es que se genera un número mayor de celdas, un mayor número de triángulos, y una triangulación de la isosuperficie menos uniforme.

Otra alternativa propuesta es utilizar celdas octaédricas centradas en las caras de las celdas cúbicas. Cada celda está delimitada por cuatro muestras de la rejilla y dos vértices colocados en el centro de las rejilla original. El valor en estos dos vértices se estima a partir de los valores de la rejilla original [CP98]. Esta estrategia genera un número mayor de celdas que el Marching Cubes.

Por otra parte, el utilizar un valor promediado en el centro del cubo hace un efecto de filtro, que suaviza las aristas del volumen [PV05]. Esta estructura de celdas se puede obtener, sin necesidad de generar valores no muestreados, si los datos en lugar de formar una rejilla regular están desplazados en una distribución en tresbolillo [CTM03, SS07, TY02]. El inconveniente de este método es la necesidad de utilizar un muestreo diferente para los datos. Esto no genera problemas cuando las muestras se obtienen evaluando una función

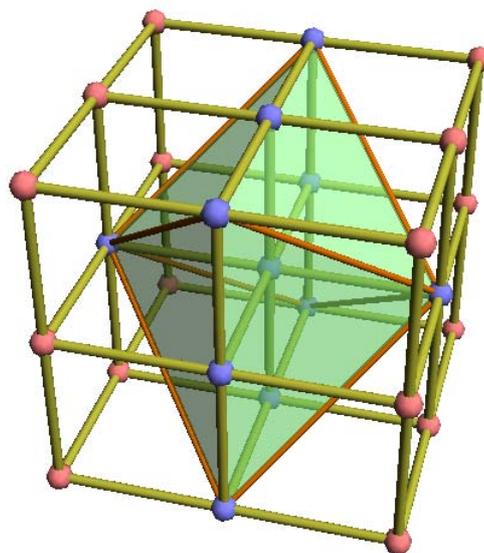


Figura 1: Estructura de la celda. El octaedro tiene sus vértices en seis de las muestras del modelo. Hay otras cuatro muestras en los centros de las aristas de ecuador y otra muestra en su centro. Las muestras usadas para generar la celda están dibujadas en azul.

matemática, pero no es viable para conjuntos de datos ya generados.

Nuestro objetivo es generar isosuperficies libres de fisuras con celdas de mayor tamaño, sobre una rejilla regular, procesadas de forma independiente sin necesidad de asumir, ni calcular, el comportamiento de la isosuperficie en el interior de las mismas.

3. Celdas octaédricas

La idea del método es utilizar celdas del máximo volumen, delimitadas por caras triangulares, con las que se pueda formar un recubrimiento del espacio. Concretamente utilizamos octaedros que integran once muestras del volumen (ver Figura 1).

Puede apreciarse que la celda está delimitada por once vértices (azules en la Figura), seis ubicados en los vértices del octaedro. Las cinco restantes se encuentran en los centros de las aristas del ecuador y en el centro del octaedro. Denominamos ecuador al plano que corta el octaedro cruzando nueve de sus muestras. Con esta terminología podemos decir que el octaedro tiene nueve muestras en el plano ecuatorial y dos en los polos. Igualmente, podemos determinar que la celda está formada por dos hemisferios con forma de pirámide, unidas por sus bases en el plano ecuatorial.

Cada cara exterior del octaedro está delimitada por cua-

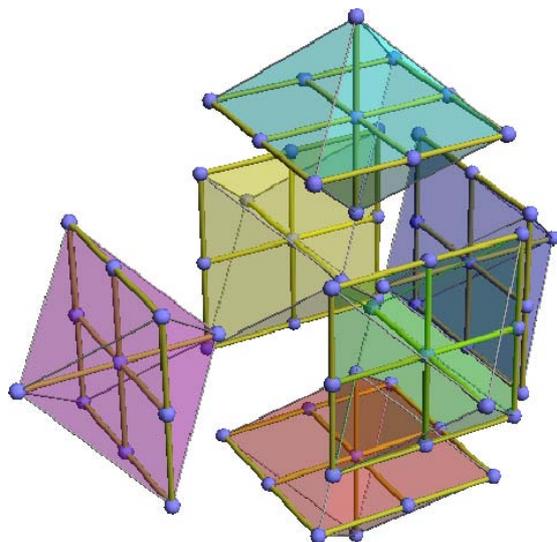


Figura 2: Agrupamiento de celdas. Se muestran las seis celdas que comparten una de sus cúspides. Las celdas se han desplazado para que puedan identificarse.

tro vértices, al igual que con celdas cúbicas. No obstante, al contrario de lo que ocurre con estas, las caras del octaedro admiten una única subdivisión en triángulos de tres muestras, ya que tres de sus muestras son co-lineales. Es decir, las caras exteriores son triángulos.

Para realizar un recubrimiento del espacio con celdas octaédricas se disponen de tal forma que en la muestra de cada cúspide del octaedro coincidan seis octaedros (ver Figura 2). Los octaedros quedan dispuestos según tres direcciones diferentes. Cada muestra pertenece a un número variable de octaedros según su ubicación:

- Uno, cuando está en el centro de un octaedro.
- Cuatro, cuando la muestra está en el centro de una arista ecuatorial del octaedro.
- Seis, cuando la muestra se encuentra en una de las dos cúspides del tetraedro.
- Doce, cuando la muestra está en uno de los cuatro vértices ecuatoriales del octaedro.

A pesar de la diferente orientación de los octaedros y de los diversos grados de pertenencia de una muestra a celdas octaédricas, no es necesario almacenar ninguna información adicional en la estructura de datos del modelo.

Considerando un retículo de muestras, el número medio de celdas por muestra, usando celdas octaédricas, es de siete, ligeramente inferior a las ocho celdas del Marching Cubes. El volumen de la celda octaédrica es de 2,67 veces el de una celda cúbica. Esto hace que los triángulos generados sean de mayor tamaño. El número de celdas generadas para un modelo es del orden de 0,38 por el número de muestras del mode-

lo, mientras que en Marching Cubes es del orden del número de muestras, y con Marching Tetrahedra del orden de seis veces superior. Por tanto, el número de celdas a procesar será también menor. La contrapartida es que el procesamiento de las celdas es algo más complejo. En la sección siguiente se explica el proceso que proponemos para extraer los fragmentos de isosuperficie en el interior de la celda octaédrica.

El esquema propuesto tiene también el inconveniente de recubrir un volumen irregular. La mayor parte de los conjuntos de datos están formados por una distribución regular de muestras contenidas en un espacio cúbico. En los límites de ese espacio, las celdas octaédricas son atravesadas por los planos de la caja englobante del volumen, quedando en el interior de la caja secciones de octaedros. Estas secciones de octaedros pueden subdividirse en tetraedros y considerarse de esta forma a la hora de extraer la isosuperficie de su interior.

4. Generación de isosuperficies con celdas octaédricas

La isosuperficie se genera de forma independiente para cada celda, asegurando continuidad entre celdas vecinas, debido a que las zonas de contacto entre celdas son triángulos. Para generar la isosuperficie dentro de una celda sería posible utilizar una tabla de casos, pero el número de posibles configuraciones sería demasiado alto.

Nuestro método genera la isosuperficie en dos pasos. En primer lugar se analiza la isocurva en el plano ecuatorial de la celda. En segundo lugar, se generan dos mallas de triángulos, correspondientes al hemisferio norte y sur, a partir de esta isocurva y de los valores de las muestras en los polos del octaedro. Esto garantiza la continuidad entre el hemisferio norte y sur. Por otra parte, permite evaluar la cara ecuatorial una única vez. A continuación comentamos más detenidamente el proceso.

4.1. Generación de isocurva

La isocurva en el plano ecuatorial depende solamente de las nueve muestras que se encuentran en dicho plano. Cualquier método de generación de isocurvas 2D serviría para realizar este proceso. No obstante, estamos interesados en realizarlo de tal modo que:

- No haya inconsistencias entre los fragmentos de isosuperficie entre celdas vecinas de arista.
- No haya configuraciones con ambigüedad en las caras ecuatoriales del octaedro que puedan generar agujeros en la isosuperficie.
- Se evalúe cada punto de corte en las aristas una única vez dentro de la celda.
- No se generen triángulos en exceso.
- Se pueda realizar el proceso de extracción sobre celdas cortadas por la caja englobante del modelo.

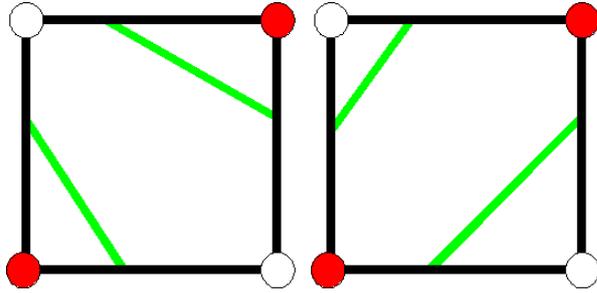


Figura 3: Ambigüedad en una cara del plano ecuatorial del octaedro. Para una misma configuración de valores de propiedad, se puede realizar una extracción de isosuperficie de dos formas diferentes.

Si consideramos caras cuadrangulares en el plano ecuatorial del octaedro, estas caras pueden tener configuraciones de valor de propiedad que generen ambigüedades a la hora de extraer isosuperficies. Esta es la situación que mostramos en la Figura 3.

Como se observa, la isocurva en esta cara mostrada puede presentar dos topologías diferentes. Para evitar esto, procesamos la retícula del plano ecuatorial dividiéndola en triángulos. El hecho de procesar caras triangulares hace que no se generen configuraciones ambiguas en las caras como sucede en Marching Cubes.

Además, dependiendo del número de cortes generados por la isocurva en las aristas ecuatoriales del octaedro, realizaremos agrupaciones con la intención de que el tamaño de los triángulos sea máximo. Este proceso se lleva a cabo teniendo en cuenta el valor umbral para la isosuperficie y no requiere ninguna estructura de datos adicional. La Figura 4 muestra varios ejemplos de agrupaciones.

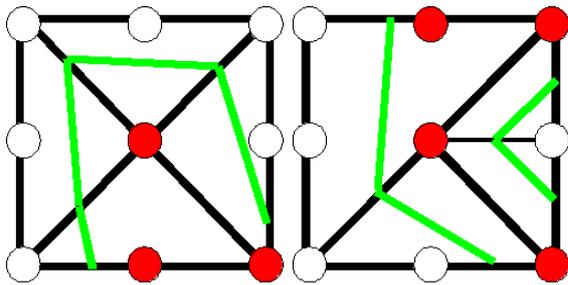


Figura 4: Generación de isocurvas en el plano ecuatorial. A partir del etiquetado de los nueve muestras en el ecuador del octaedro (izquierda) se deciden los triángulos que se han de procesar. El tamaño de los triángulos depende del etiquetado de las muestras.

Partiendo de una situación inicial como la mostrada en la Figura 5.a, se determina en primer lugar el número de cortes

que forma la isocurva en las cuatro aristas exteriores y en las dos diagonales que aparecen en la Figura. Estos puntos de corte se calculan en esta fase usando las tres muestras de cada arista, se almacenan temporalmente y ya no es necesario recalcularlas durante el proceso.

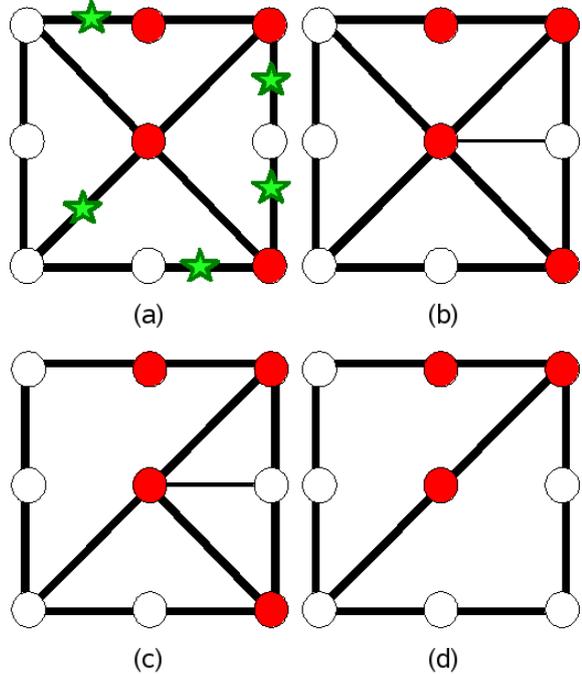


Figura 5: Diferentes formas de agrupación de triángulos en el plano ecuatorial del octaedro. Dependiendo del número de puntos de corte se van realizando agrupaciones en triángulos de mayor tamaño.

Analizando cada arista puede suceder que:

- No haya punto de corte.
- Haya un punto de corte en alguno de los dos segmentos que forman la arista.
- Haya dos puntos de corte, uno en cada segmento de la arista

En el tercer caso, la arista queda subdividida en dos triángulos tal y como se muestra en la Figura 5.b. Si estamos en el primero o en el segundo caso, la arista considerada no se subdivide.

Cuando dos aristas exteriores y adyacentes entre sí tengan un solo punto de corte, es posible efectuar una agrupación a un nivel más alto. Esta teselación es la que mostramos en la Figura 5.c. Esta agrupación es posible cuando se cumplen dos restricciones más:

- La arista diagonal que delimita el triángulo tiene un solo punto de corte.
- No hay ningún punto de corte en la arista interna que une las dos cúspides del octaedro.

El máximo de agrupaciones lo conseguimos en el caso mostrado en la Figura 5.d. Para hacerla posible, hemos cambiado el signo de la muestra que ocasionaba dos puntos de corte en una de las aristas.

A partir de esta teselación del plano ecuatorial, para cada uno de los triángulos podemos obtener un par simétrico de tetraedros formado por una cara compartida en dicho triángulo y las dos cúspides de cada hemisferio. De esta forma, la celda octaédrica queda dividida en una partición adaptativa de tetraedros. Cada uno de ellos generará un fragmento de isosuperficie que podemos obtener mediante un Marching Tetraedra.

Adicionalmente, las celdas octaédricas fragmentadas que obtenemos en las zonas adyacentes a la caja englobante del volumen, pueden ser también subdivididas en tetraedros y procesadas por este método.

Aunque muchos de los tetraedros agrupados no contemplan los vértices intermedios de las aristas, para generar el fragmento de isosuperficie se utiliza el punto de corte que teníamos precalculado y que sí consideraba las tres muestras de la arista.

5. Evaluación

Para evaluar el método hemos implementado un prototipo y realizado varias pruebas en un ordenador con una CPU Core 2 Duo con 3 GB de memoria RAM. Hemos comparado nuestros resultados con los obtenidos usando Marching Cubes y Marching Tetrahedra. La partición en tetraedros del volumen usada para el Marching Tetrahedra es la que se presenta en [VTL508]. Las Figuras 6 muestran varias capturas de pantalla de visualizaciones de nuestro método aplicado sobre tres modelos volumétricos de ejemplo.

Visualmente, las isosuperficies obtenidas con celdas octaédricas son similares a las que pueden generarse mediante otros métodos. Como sucede en otras técnicas que hacen uso de celdas tetraédricas, pueden aparecer zonas con pequeños picos debidos a la interpolación sobre aristas diagonales. En la Figura 7 se muestra la aproximación obtenida a una isosuperficie usando las tres técnicas. Para poder comparar mejor las diferencias entre los tres métodos, el mismo modelo es mostrado con más detalle en la Figura 8.

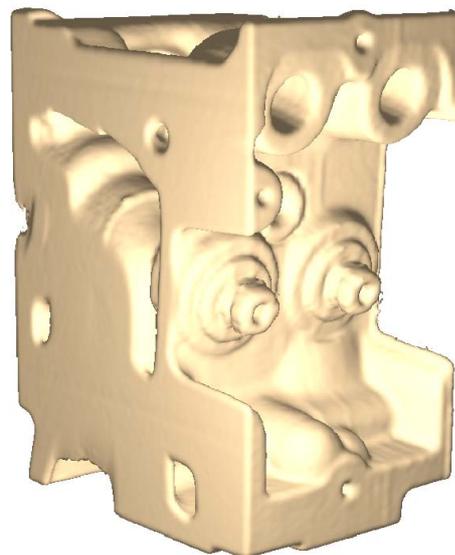
El método propuesto genera isosuperficies que están formadas por un número superior de triángulos comparándolo con Marching Cubes. Sin embargo, si consideramos otros métodos basados en celdas no cúbicas, el número de triángulos es muy inferior. La Tabla 1 muestra el número de triángulos generados en la visualización de los modelos de ejemplo.

En la Tabla 2 se muestra el tiempo de generación de la isosuperficie con Marching Cubes, con Marching Tetrahedra y con nuestra propuesta.

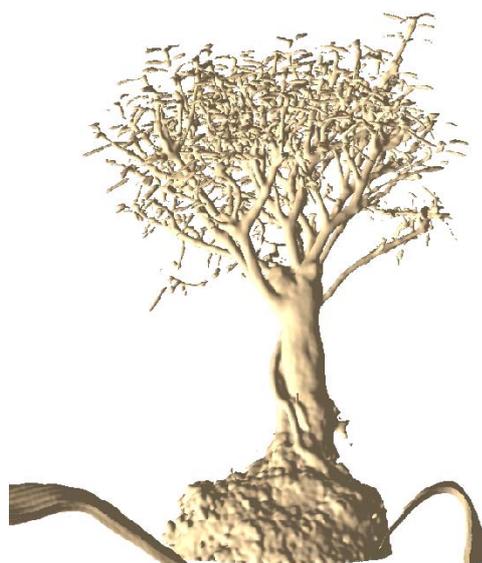
Como se observa, el tiempo empleado por la técnica de



(a) Baby head Model

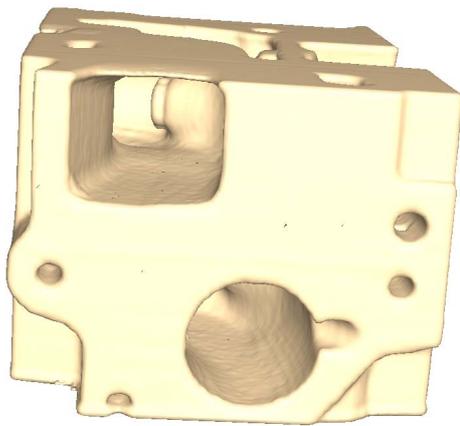


(b) Engine Model

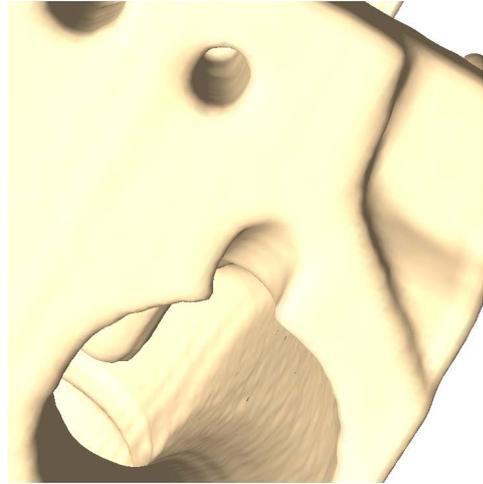


(c) Bonsai Model

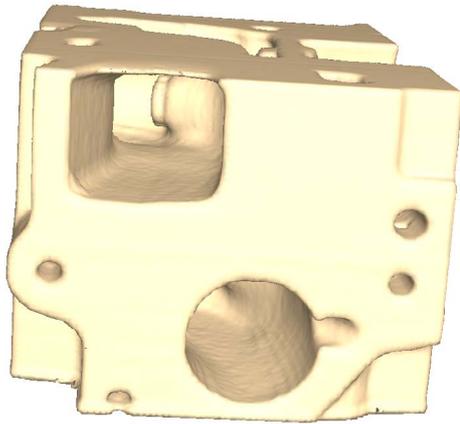
Figura 6: *Isosuperficies de los modelos de ejemplo.*



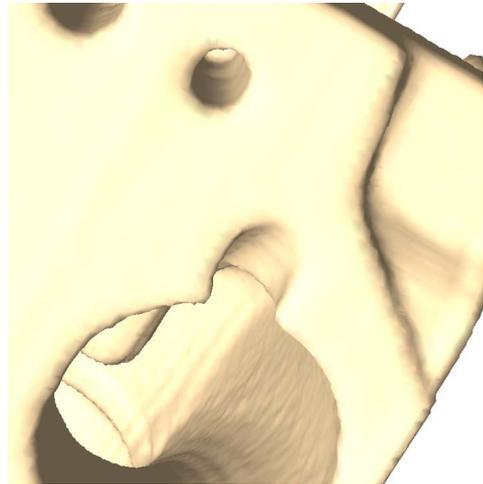
(a) Marching Cubes



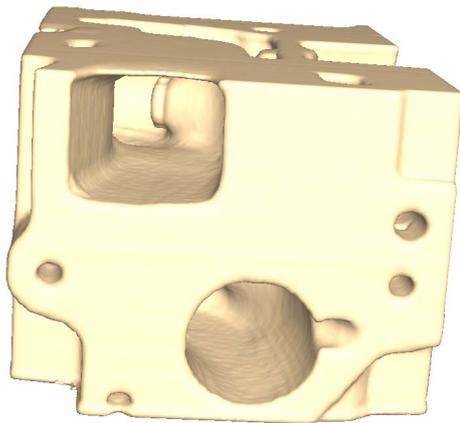
(a) Marching Cubes



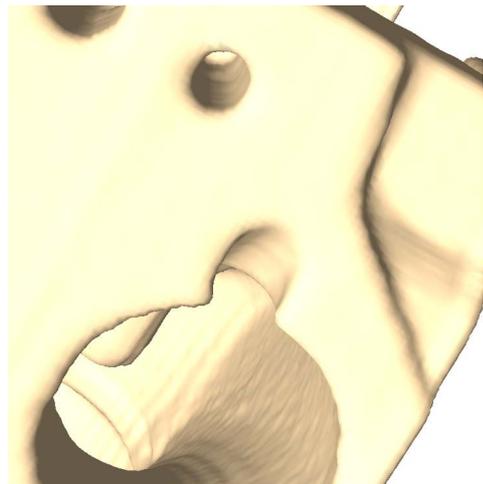
(b) Marching Octahedra



(b) Marching Octahedra



(c) Marching Tetrahedra



(c) Marching Tetrahedra

Figura 7: Isosuperficies obtenidas en el modelo Engine mediante las tres técnicas de extracción de isosuperficies.

Figura 8: Imagen ampliada del modelo Engine mostrado en la Figura 7.

Mod.	Dim.	Nº de Triángulos		
		MC	MT	MO
Baby H.	256x256x98	436.716	1.404.722	565.137
Engine	256x256x256	623.314	2.074.238	851.830
Bonsai	512x512x182	827.572	2.625.209	1.081.814

Tabla 1: Número de triángulos generados con diferentes algoritmos sobre los modelos de ejemplo. Las columnas muestran, de izquierda a derecha, modelo utilizado, dimensiones de la rejilla, triángulos obtenidos con *Marching Cubes*, con *Marching Tetrahedra*, y con nuestro método.

Mod.	Dim.	Tiempo (secs.)		
		MC	MT	MO
Baby H.	256x256x98	1'67	2'59	1'09
Engine	256x256x256	4'21	5'55	2'48
Bonsai	512x512x182	11'37	14'11	6'71

Tabla 2: Tiempos de extracción. Las columnas muestran, de izquierda a derecha, modelo utilizado, dimensiones de la rejilla, tiempo empleado por *Marching Cubes*, por *Marching Tetrahedra* y por nuestro método.

extracción que proponemos en este trabajo es, aproximadamente y en el peor de los casos, un 65 % del tiempo empleado por *Marching Cubes*. El hecho de considerar celdas que abarcan más porción de volumen afecta al tiempo global empleado en obtener la isosuperficie. Incluso aunque el procesamiento de una celda octaédrica sea un poco más complejo que el de una celda cúbica.

6. Conclusiones y Trabajos Futuros

Los algoritmos de *Marching* son eficientes en la generación de isosuperficies, permitiendo extraer la isosuperficie con un orden de complejidad lineal en el número de celdas. No obstante, cuando trabajan con celdas cúbicas, pueden producir fisuras en la isosuperficie. Estas pueden evitarse si se trabaja con celdas delimitadas por caras triangulares. Las propuestas realizadas hasta la fecha en este sentido han trabajado con celdas formadas por un número bajo de muestras (3 en *Marching Tetrahedra* y 4 en *Centered Cubic Lattice*), aumentando el número de celdas y el de triángulos generados.

Es posible generar celdas de mayor tamaño con caras triangulares utilizando celdas octaédricas formadas por 11 muestras. Con estas celdas es posible extraer isosuperficies sin fisuras, con un número de triángulos inferior al generado por otros métodos. Usando esta teselación no se producen casos especiales, el tiempo de procesamiento se reduce significativamente, no se necesita información adicional, ni precalcular nada, y se trabaja directamente con los valores del rejilla de datos.

Debido a la interpolación en las diagonales, se producen irregularidades en la isosuperficie de forma similar a como ocurre en otros métodos de descomposición de celdas en tetraedros. Por esta razón, como trabajo futuro pretendemos nos planteamos la posibilidad de realizar una extracción de isosuperficies dual. Esta técnica basada en el cálculo de isopuntos en el interior de las celdas y no sobre las aristas, nos permitiría solucionar este problema, así como facilitar en un futuro la extracción sobre una partición de celdas octaédricas de distinto nivel de resolución.

Por otra parte, nos proponemos estudiar la posibilidad de combinar el método con técnicas habituales en extracción de isosuperficies como la utilización de índices para la localización de las celdas activas.

Agradecimientos

Baby Head, Engine y Bonsai son modelos volumétricos por gentileza de VolVis distribution of SUNY Stony Brook (NY, USA), General Electric (USA), y Steffan Roettger (University of Erlangen, German) respectivamente.

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

- [AC97] G. Albertelli and R. A. Crawfis: Efficient Subdivision of Finite Element Datasets into Consistent Tetrahedra. *Visualization* 97, 213-219.
- [AG91] E. Allgower and S. Gnatzmann: Simplicial Pivoting for Mesh Generation of Implicitly Defined Surfaces. *Computer Aided Geometric Design*, 8, 305-325, 1991.
- [Blo88] J. Bloomenthal: Polygonization of implicit surfaces. *Computer Aided Geometric Design*, 341-355, 1988.
- [Blo94] J. Bloomenthal: An Implicit Surface Polygonizer. *Graphics Gems IV*, 324-349. Academic Press, 1994.
- [CSA00] H. Carr, J. Snoeyink, U. Axen: Computing Contour Trees in All Dimensions. *ACM-SIAM Symp. on Discrete Algorithms (SODA) 2000*, 918-926.
- [CP98] S. L. Chan and E. O. Purisima: A new Tetrahedral tessellation scheme for isosurface generation. *Computers & Graphics*, 22(1):83-90, 1998.
- [CTM03] H. Carr, T. Theussl, and T. Möller: Isosurfaces on optimal regular samples. In C. D. H. G.-P. Bonneau, S. Hahmann, editor, *Proceedings of the symposium on Data visualisation 2003*, pages 039-048. Eurographics Association, 2003.
- [LC87] W. E. Lorensen and H. E. Cline: *Marching Cubes: A high resolution 3d surface construction algorithm*. In SIGGRAPH '87: *Proceedings of the 14th annual conference on Computer graphics and interactive techniques*, pages 163-169, New York, NY, USA, July 1987. ACM Press
- [MH90] N.L. Max, P. Hanrahan, R.A. Crawfis: Area and Volume Coherence for Efficient Visualization of 3D Scalar Functions. *Computer Graphics*, 24(5), 27-33, 1990.

- [NF97] G.M. Nielson and R. Franke: Computing the Separating Surface for Segmented Data. In *Visualization 97*, 229-233.
- [NS97] G.M. Nielson and J. Sung: Interval Volume Tetrahedrization. In *Visualization 97*, 221-228.
- [NB93] P. Ning and J. Bloomenthal: An Evaluation of Implicit Surface Tilers. *IEEE Comp. Graphics and Appl.*, 13, 33-41, 1993.
- [NY06] T. S. Newman, H. Yi: A survey of the Marching Cubes algorithm. *Computers & Graphics* 30 (2006) 854-879
- [PV05] Jan Patera, Václav Skala: Centered Cubic Lattice Method Comparison. *Proceedings of Algoritmy 2005*, pp. 1-10. 2005.
- [ST90] P. Shirley and A. Tuchman: A Polygonal Approximation to Direct Scalar Volume Rendering. *Computer Graphics*, 24(5), 63-70, 1990.
- [SS07] Robin Strand, Peer Stelldinger: Topology Preserving Marching Cubes-like Algorithms on the Face-Centered Cubic Grid. 14th International Conference on Image Analysis and Processing. *ICIAP 2007*. 2007
- [TY02] T. Takahashi and T. Yonekura: Isosurface construction from a data set sampled on a face-centered-cubic lattice. In *Proceedings of ICCVG 2002*, volume 2, pages 754-763, 2002.
- [TPG99] Treece G.M.; Prager R.W.; Gee A.H.: Regularised Marching Tetrahedra: improved iso-surface extraction. *Computers & Graphics*. 23(4). 1999
- [VTLS08] VELASCO F., TORRES J. C., LEÓN A., SOLER F.: Adaptive cube tessellation for topologically correct isosurfaces. *Journal of Virtual Reality and Broadcasting* 5, 3 (Mar. 2008). , ISSN 1860-2037.
- [WB96] C Weigle and D.C. Banks: Complex-Valued Contour Meshing. *Visualization* 96, 173-181.
- [WMW86] B. Wyvill, C. McPheeters, and G. Wyvill. Animating soft objects. *The Visual Computer*, 2(4):235-242, 1986.