

Una Triangulación Jerárquica para Modelos Topográficos Multiresolución

M. J. Abásolo

Becaria de CONICET, UNICEN. E-mail: abasolo@ipc4.uib.es

A. De Giusti

LIDI, UNLP. E-mail: degiusti@lidi.info.unlp.edu.ar

J. Blat

Universidad de las Islas Baleares.(07071)Palma,España

Resumen

La visualización en tiempo real de modelos de terreno triangulados sigue considerándose un problema en los sistemas de realidad virtual. Los modelos poligonales de grandes volúmenes de datos de terreno requieren un gran número de polígonos. Problemas fundamentales son la eficiencia del *rendering* o visualización de la representación, así como también la transmisión del modelo por red. Los modelos multiresolución representan múltiples niveles de detalle de un objeto. De este modo un algoritmo de *rendering* puede elegir el nivel de detalle a visualizar de acuerdo a la posición del observador, y un algoritmo de transmisión puede realizar la transmisión progresiva del mismo. En el presente trabajo se propone una jerarquía de los puntos y un esquema de triangulación multiresolución que no necesita estructuras de jerarquía auxiliares con la que se obtienen mejores resultados que con otros esquemas similares como la triangulación quadtree restringida.

1. INTRODUCCIÓN

Los modelos topográficos 3D pueden ser usados como herramienta de interpretación de los datos de terreno como por ejemplo análisis topográficos, planeamiento territorial, sistemas de información geográficos (*GIS*), o como herramienta para sistemas de simulación o realidad virtual [Casillas99], [Hernández99]. En este último caso el tiempo real ha de considerarse, y el tamaño del modelo a visualizar o transmitir cobra la mayor importancia.

Los triángulos son la primitiva más popular en computación gráfica. Pueden representar cualquier modelo aproximadamente y además existe hardware gráfico especializado que puede visualizarlos a gran velocidad. Desafortunadamente, para representar un modelo 3D con exactitud se requiere un gran número de polígonos. La simplificación de polígonos es el acto de transformar un modelo poligonal 3D en una versión más simple [Erikson96]. Se reduce el número de polígonos que se necesitan para representar un modelo mientras se trata de retener una buena aproximación a la forma y apariencia originales. Esto no solo provee el beneficio de incrementar la eficiencia del *rendering* sino que reduce el espacio necesario para almacenamiento y agiliza la transmisión por red.

Los modelos multiresolución proveen diferentes niveles de detalle o *LOD* (*level-of-detail*) del objeto modelado. Se puede elegir la resolución apropiada para visualizar el modelo dependiendo de parámetros de visión como el tamaño del objeto con respecto al tamaño de la pantalla, o la distancia al punto y dirección de visión. El modelo apropiado es aquel de resolución más simple que presenta igual apariencia que la representación más detallada.

En la sección 2 se presenta un modelo de triangulación multiresolución para bases de datos de terreno, que:

- Es adaptativo a la estructura del terreno, es decir que las regiones con cambios de elevación de alta frecuencia se modelan con más triángulos por unidad de área que las regiones de baja frecuencia;
- Provee medios para extraer una representación de la superficie a precisión variable. Este hecho permite la visualización usando múltiples *LODs*, o transmisión progresiva.

El modelo de triangulación trata de usar la mínima cantidad de triángulos necesarios, eliminando las restricciones en la selección de puntos de la triangulación quadtree la cual construye triángulos extras para prevenir *cracks* [Pajarola98].

En la sección 3 se describen los pasos necesarios para tratar un modelo multiresolución: determinación del nivel de detalle, selección de puntos, actualización de l modelo. En la sección 4 se describe una implementación eficiente del modelo mediante el uso de una estructura del tipo "*triangle strip*". Finalmente en la sección 5 se presentan los resultados obtenidos comparándolos con una representación similar.

2. TRIANGULACIÓN JERÁRQUICA SIN RESTRICCIONES

2.1. Jerarquía y Triangulación

Un modelo de terreno es generalmente un campo de alturas, que es una matriz de puntos distribuidos regularmente en una grilla bidimensional. El esquema de triangulación presentado realiza una triangulación jerárquica adaptativa para un campo de alturas.

Otros esquemas de triangulación de mallas regulares están basados en triangulación quadtree (figura 1). El cuadrante básico consiste de cuatro puntos y dos triángulos (figura 1-a). El primer paso de refinamiento agrega el punto medio del bloque. Los dos triángulos iniciales se dividen en dos (figura 1-b). El siguiente paso consiste en agregar los puntos medios de los lados del cuadrante y dividir los triángulos de acuerdo a ello (figura 1-c). Cada subcuadrante obtenido puede se dividido recursivamente como el bloque inicial.

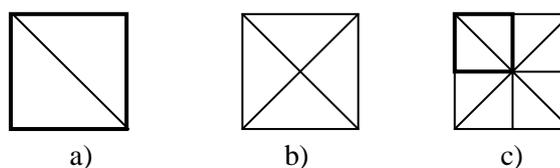


FIGURA 1. Subdivisión Quadtree

La jerarquía de triángulos presentada tiene solo un paso de división recursiva de un triángulo patrón (figura 2). El patrón básico consiste de un triángulo (figura 2-a), que es recursivamente dividido al agregar el punto medio de su diagonal (figura 2-b). Inicialmente dos patrones triangulares dispuestos como en la figura 2-a son equivalentes al bloque quadtree inicial de la figura 1-a. La figure 2-c muestra como dividiendo recursivamente cada patrón triangular puede obtenerse una división equivalente a la división quadtree (figura 1-c).

La subdivisión "bitree" difiere de la subdivisión quadtree en que la primera obtiene una jerarquía de triángulos que pueden dividirse independientemente. La figura 3 muestra algunas posibles subdivisiones con este esquema que no son posibles usando una subdivisión quadtree.

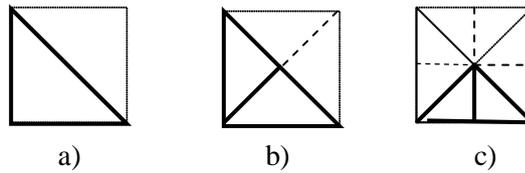


FIGURA 2. Subdivisión "Bitree"

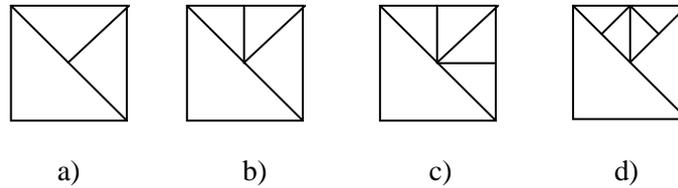


FIGURA 3. Variantes de la subdivisión "Bitree"

La figura 4 muestra la asignación de niveles jerárquicos a los puntos de la grilla. N_0 es el nivel más alto en la jerarquía, y denota la doble raíz del "bitree" formado inicialmente por dos triángulos de nivel N_0 . Un triángulo de nivel N_i se subdivide en dos triángulos de nivel N_{i+1} al agregarse el punto medio de la diagonal que tiene nivel N_{i+1} .

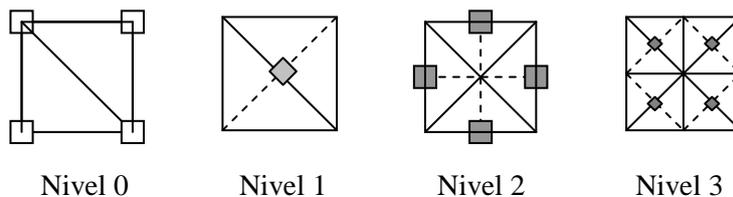


FIGURA 4. Niveles Jerárquicos

Dado que la triangulación está jerárquicamente definida, cada triángulo se divide recursivamente. La triangulación es computacionalmente eficiente porque está dada de forma implícita, y no hay necesidad de cálculos. [Erikson96] define una medida de proporción de apariencia de un triángulo como la proporción del lado más largo sobre el más corto del mínimo rectángulo que encierra el triángulo. Es deseable tener triángulos con proporción cercana a la unidad, dado que los triángulos alargados son más propensos a errores numéricos o de *rendering*. La subdivisión "bitree" produce triángulos que tienen buena forma dado que tienen proporción de apariencia igual a uno.

2.2. Prevención de "cracks"

La presencia de "cracks" o agujeros negros es indeseable en cualquier esquema de triangulación. Cuando el modelo se visualiza desde cierto ángulo, aún se pueden notar agujeros muy pequeños provocando un efecto negativo. Una solución propuesta en [DeHaemer91] es la construcción de un polígono extra para llenar el agujero. En nuestro esquema esto produce una ruptura en la jerarquía, además de agregar polígonos mientras se está tratando de reducir su número. La figura 5 muestra la presencia de *cracks* en una triangulación quadtree sin restricciones.

[Herzen87] presenta una triangulación quadtree restringida con el requerimiento de que cuadrantes adyacentes pueden diferir a lo sumo en un nivel de detalle en la jerarquía quadtree y que cada bloque quadtree se triangula mediante dos triángulos por lado a menos que el lado sea adyacente a un bloque mayor.

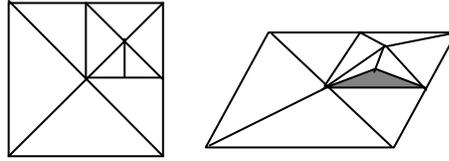


FIGURA 5. Presencia de *cracks* en una triangulación quadtree sin restricciones

[Pajarola98] presenta un modelo basado en triangulación quadtree que evita *cracks* con una restricción en la selección de puntos de acuerdo a un grafo de dependencias definidas entre los vértices. Los vértices centrales de un bloque dependen de los dos vértices opuestos por la diagonal del bloque (figura 6-a). Los vértices no centrales dependen de los dos vértices centrales de bloques adyacentes horizontal o verticalmente (figura 6-b). Los cuatro vértices del nivel 0 son mutuamente dependientes.

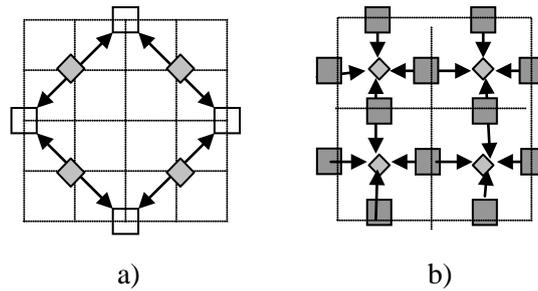


FIGURA 6. Dependencias de vértices en la subdivisión Quadtree restringida

La regla es tal que si un vértice se selecciona para la triangulación de acuerdo a algún criterio de aproximación, deben ser seleccionados los vértices de los que depende también. Al mismo tiempo, estos vértices relacionados tienen otras dependencias que deben ser seleccionadas. Finalmente esta regla restringe al quadtree de manera que dos bloques adyacentes pueden diferir al menos en un nivel en la jerarquía.

La presencia de *cracks* del ejemplo de la figura 5 puede ser evitada mediante el agregado de puntos siguiendo las dependencias. Este modelo garantiza una triangulación "matching" pero se producen triángulos extras para evitar *cracks* como pueden verse en la figura 7 en línea cortada.

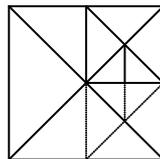


FIGURA 7. Evitar *cracks* con Quadtree restringido

En la triangulación jerárquica presentada cada ocurrencia de un triángulo puede ser recursivamente reemplazada por dos triángulos. En los ejemplos de la figura 3, puede verse que existe el riesgo de *cracks*. Para evitar, la presencia de los mismos, se agregan puntos llamados "ficticios", los cuales tienen un valor de altura interpolado entre los dos puntos más cercanos pertenecientes a la diagonal de la cual es punto medio. La selección de puntos para triangular es realizada siguiendo un criterio para satisfacer una cierta tolerancia de aproximación al modelo original (sección 2.3). En cambio, los puntos ficticios son necesarios solo para completar la triangulación determinada por los puntos seleccionados de manera que no haya *cracks* entre triángulos adyacentes de diferentes niveles de simplificación.

De manera análoga al esquema quadtree restringido se definen dependencias entre vértices, pero se verá como estas resultan reducidas, y en consecuencia se agregarán menos puntos relacionados. Para cada punto p_i de nivel N_i ($i > 0$), punto medio de la diagonal de dos triángulos T_{i-1}^1 y T_{i-1}^2 de nivel N_{i-1} adyacentes por la diagonal (si la diagonal pertenece a un borde del bloque inicial solo se habla de un triángulo), se redefinen las dependencias como sigue:

- Si p_i está seleccionado, depende de los puntos p_{i-1}^1 y p_{i-1}^2 de nivel N_{i-1} que son los vértices de los triángulos T_{i-1}^1 y T_{i-1}^2 que no pertenecen a la diagonal. En la figura 8 se muestran las dependencias para todas las orientaciones posibles del patrón triangular. Estas dependencias son equivalentes a las definidas en el quadtree restringido, pero se simplificó su definición dado que no existe diferencia entre vértices centro y no centro de un bloque quadtree.

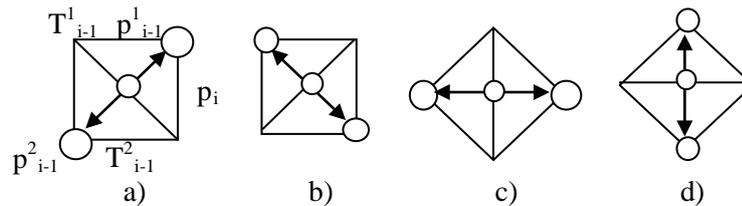


FIGURA 8. Dependencias de un vértice seleccionado

- Si p_i es "ficticio", solo depende de un punto p_{i-1}^k de nivel N_{i-1} que vértice del triángulo T_{i-1}^k ($k=1$ o $k=2$) siendo T_{i-1}^k el triángulo que necesita p_i . Esto significa que existe al menos un triángulo hijo de T_{i-1}^k , llamado T_i^{kj} cuyo punto medio de su diagonal p_{i-1}^k depende de p_i . La figura 9 muestra la dependencia de p_i y el triángulo que depende de p_i está sombreado.

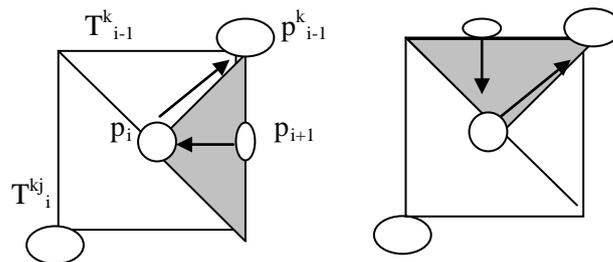


FIGURA 9. Dependencias de un vértice ficticio

Se modifica la regla de triangulación de quadtree restringido de manera que si un vértice es seleccionado para satisfacer un criterio de aproximación, los vértices de los que depende no necesitan ser seleccionados sino que pueden ser ficticios. Como un punto ficticio solo depende de un vértice, que a su vez puede ser ficticio, se reduce la cadena de dependencias y en consecuencia se agregan menos triángulos. No hay restricciones de niveles entre triángulos adyacentes, y los *cracks* causados por estas diferencias se evitan mediante los puntos ficticios.

No es necesario contar con una estructura auxiliar pues la jerarquía está dada implícitamente y las dependencias de los puntos se deducen según posición.

3. MODELO MULTIRESOLUCIÓN

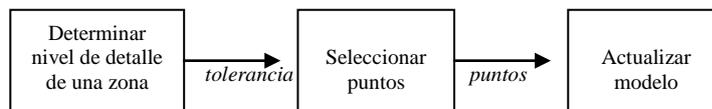
En la sección anterior se ha descrito una triangulación jerárquica de una grilla regular de datos de terreno. La estructura provee medios para extraer una representación de la superficie a una precisión variable según lo requerido por los modelos multiresolución. En lugar de tener una serie de diferentes niveles de detalle a priori, la estructura tiene la información básica para generar los diferentes niveles de detalle según se necesiten. Diferentes regiones del modelo pueden ser extraídas a diferentes

resoluciones. Por esto puede decirse que hay un número finito de niveles de detalle limitados por el número de puntos pero que pueden combinarse casi sin límite de acuerdo a las diferentes regiones variables.

Además, se soporta un modelo adaptativo a la estructura del terreno.

El modelo de terreno puede ser usado tanto para visualizar una representación con cierto nivel de resolución como para la transmisión del mismo. En ambos casos, podemos generalizar tres pasos a realizar:

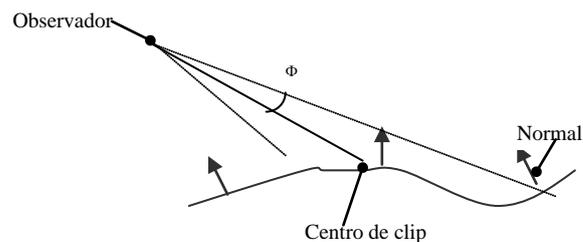
- Determinar el nivel de detalle con que se visualizará o transmitirá el modelo o una cierta región;
- Seleccionar los puntos de acuerdo a una cierta tolerancia de error determinada en el paso anterior;
- Actualizar el modelo, agregando o quitando puntos según sea un refinamiento o una simplificación del modelo anterior.



3.1. Determinación del nivel de detalle

Al visualizar un modelo, dinámicamente se puede determinar que algunas partes de la escena se muestren a una baja resolución pero con un resultado visual equivalente al obtenido con el modelo de máxima resolución. Se trata entonces de maximizar una función de beneficio visual minimizando al mismo tiempo la resolución del modelo basándonos en la posición del observador relativa a la zona analizada. La posición de la zona relativa al observador debe considerar:

- distancia de la zona al punto de vista;
- ángulo Φ con la dirección de visión;
- ángulo entre la normal del terreno y la dirección de visión.



Así, las zonas que se hallen lejos del observador, o que se alejen de la dirección de visión, o que tengan una gran inclinación con respecto a la dirección de visión se mostrarán con una baja resolución, pues una resolución mayor nos daría un beneficio o resultado visual equivalente.

Si en cambio lo que se quiere es transmitir el modelo por red simplemente se elegirá la resolución a transmitir .

Según un criterio de selección de nivel de detalle se determinara una tolerancia de error o parámetro similar que controlará la selección de puntos (sección 3.2).

3.2. Selección de puntos

El propósito de un proceso de simplificación de una superficie es obtener un modelo reducido que use menos polígonos y aproxime al modelo original en cierto grado. Es deseable que la selección de puntos sea adaptativa a la estructura del terreno, de manera que se elijan más puntos en las regiones donde sean más necesarios para aproximar la superficie. La curvatura local de una región es una medida aproximada de cuan rápido cambia la geometría en dicha área. Una región de alta curvatura cambia de dirección abruptamente, y tiene que ser modelada con más triángulos por unidad de área que una región de baja curvatura la cual permanece relativamente estable.

La selección de puntos que contribuyan a una triangulación puede hacerse mediante un criterio basado en un error de aproximación en el espacio del objeto, o en cambio puede basarse un criterio sobre una representación paramétrica de la superficie

Algunas características en el espacio del objeto que pueden usarse para la selección son:

- Curvatura local.
- Distancia Euclídeana del punto original a su correspondiente proyección sobre el modelo simplificado.
- Normales de los triángulos que tienen un vértice en común
- [Hoppe93] selecciona puntos minimizando una función de energía compuesta de tres términos, uno de ellos basado en la distancia entre la malla original y la simplificada, y otro basado en el número de puntos seleccionados.

Una alternativa es tener una representación paramétrica de la superficie y triangularla seleccionando puntos en el espacio de los parámetros. En [Lounsbery97] y [Gross95] la superficie inicial se descompone por medio de una transformada Wavelet. La selección de puntos se hace de acuerdo a los coeficientes resultantes. [Gross95] computa la energía parcial de los coeficientes en las regiones de la superficie. Si la energía es baja, la aproximación puede hacerse con triángulos grandes, y eso significa menor número de puntos.

Para controlar la exactitud de la aproximación puede usarse una característica en el espacio del objeto, como por ejemplo:

- Tolerancia de error en término de distancia Euclídeana del modelo simplificado al modelo original. Es la manera más común de controlar la aproximación y hay numerosos algoritmos que la usan como [DeHaemer91] [Schroeder92] [Varshney94] [Pajarola98].
- Tolerancia de error en términos de diferencia angular entre las normales de los triángulos relacionados con un vértice y el triángulo que los reemplaza en el modelo simplificado. [Hinker93] usa este criterio para eliminar polígonos de superficies coplanares.
- Número de elementos geométricos como vértices o triángulos. [Hamann94] remueve un porcentaje de triángulos del modelo, y usa la curvatura local como criterio de selección para remover primero los triángulos de menor curvatura.

A diferencia del modelo quadtree restringido, en el modelo presentado el criterio de selección de puntos puede ser independizado de la triangulación. La selección de puntos puede hacerse directamente basándose en los datos de altura en el espacio del objeto, o puede tenerse una representación paramétrica seleccionando los puntos en el espacio paramétrico. Una vez seleccionados los puntos que determina el criterio, se completa el modelo con puntos ficticios para realizar la triangulación. Debido a que no se exige que la jerarquía se halle balanceada, el modelo permite una triangulación adaptativa, y se puede controlar el error de la aproximación localmente.

En particular, se realizaron pruebas usando un criterio de selección basado en la distancia Euclídeana. Para cada triángulo T_i de nivel N_i se computa una medida de error $e(T_i)$ como la distancia Euclídeana

de todos los puntos cuya proyección xy pertenece al dominio del triángulo en su proyección xy . El cómputo de este error considera los puntos hijos p_k de todos los niveles desde N_i al nivel de máxima resolución N_n .

3.3. Actualización del modelo

Luego de seleccionar los puntos de acuerdo al criterio y tolerancia, como resultado deberán añadirse o suprimirse puntos del modelo según se trate de un refinamiento o de una simplificación. Una vez que se conocen el o los puntos seleccionados se determinan los puntos ficticios necesarios para completar la estructura bitree. En la sección 4 se describe una implementación del modelo y un método sistemático para agregar o suprimir un punto dado.

Cuando se usa un modelo con diferentes niveles de detalle, se tiene que considerar la continuidad de diferentes regiones con diferentes niveles de detalle (continuidad espacial), y la continuidad al cambiar de niveles de detalle una misma región (continuidad temporal). Los puntos ficticios evitan *cracks* (sección 2.2) y esto soluciona la discontinuidad entre diferentes regiones. La actualización del modelo puede hacerse de a un punto, o se puede actualizar el modelo completo según la nueva aproximación deseada. Es fácil ver que la primera solución dará un resultado visual con transiciones más suaves, y la segunda solución nos dará un mejor tiempo de actualización global. Al agregar o quitar un punto se puede interpolar entre su correspondiente ficticio (es decir la proyección del punto sobre el modelo) y el punto, esto produce un cambio suave entre los diferentes LODs.

Transmisión Progresiva

La transmisión progresiva [Hoppe96] significa mostrar mejores aproximaciones del modelo, por ejemplo, a medida que un modelo es transmitido por red o simplemente visualizarlo progresivamente. Como la mayoría de las triangulaciones jerárquicas, la estructura bitree soporta triangulación progresiva. La manera más simple de hacerlo es transmitir los puntos por niveles comenzando por el nivel de menos detalle hasta completar la máxima resolución. Sin embargo, resulta un mejor método seleccionar los puntos a transmitir considerando tolerancias de error cada vez mayores. Comenzando por los puntos de nivel N_0 , el receptor tiene el modelo más simple. Una vez que se tiene la estructura inicial de menor detalle, se seleccionan puntos en base a una tolerancia dada y se transmiten los puntos agregados con respecto al modelo anterior. El receptor completa la estructura con los puntos ficticios necesarios para formar el bitree. Luego se seleccionará una tolerancia de error mayor, y se repetirá el proceso, hasta que se hallan transmitido todos los puntos. Notar que los puntos ficticios no necesitan ser transmitidos pues su altura es deducible.

La transmisión puede ser por niveles (mayor nivel a menor) si el receptor realiza una actualización por cada punto recibido.

4. IMPLEMENTACIÓN: ESTRUCTURA "TRIANGLE STRIP"

Los *triangle strips* son una construcción eficientemente soportadas por el hardware gráfico, y ocupan menos espacio que la implementación con triángulos separados.

En vez de necesitar definir 3 puntos por triángulo, una estructura *triangle strip* construye triángulos a partir de una lista ordenada de puntos a, b, c, d, \dots tomando de a grupos de tres puntos secuenciales, haciendo los triángulos abc, bcd, cde, \dots El tercer lado del triángulo está implícito en la secuencia. Los triángulos sucesivos comparten un lado explícitamente generado en la secuencia.

A veces es necesario cambiar la orientación de un triángulo, para esto se agregan triángulos invisibles o triángulos línea. Si por ejemplo se quiere generar el triángulo *cbd* en lugar de *bcd*, la secuencia de puntos debe ser *a, b, c, b, d, e...* así se generan los triángulos *abc, bcb, cbd, bde...* donde *bcb* es un triángulo línea. En pruebas realizadas se comprueba que esta clase de triángulos no provoca errores e insumen un tiempo de rendering que es inferior a un triángulo común.

En la figura 10 se muestra un ejemplo donde se quieren generar los triángulos *abc, bcd* y *dae* en algún orden de sus vértices. En la figura 10-a se generan correctamente, agregándose el triángulo línea *cda*. En la figura 10-b se muestra una secuencia incorrecta dado que genera el triángulo ***cde*** indeseable.

Este error se produce debido a que el lado *cd* no es compartido totalmente por los triángulos vecinos *bcd* y *dae*. Solo en el caso de los triángulos vecinos compartan el lado pueden ser encolados uno luego del otro, pero si no se deberá generar explícitamente cada lado adyacente e insertar un triángulo línea entre ambos.

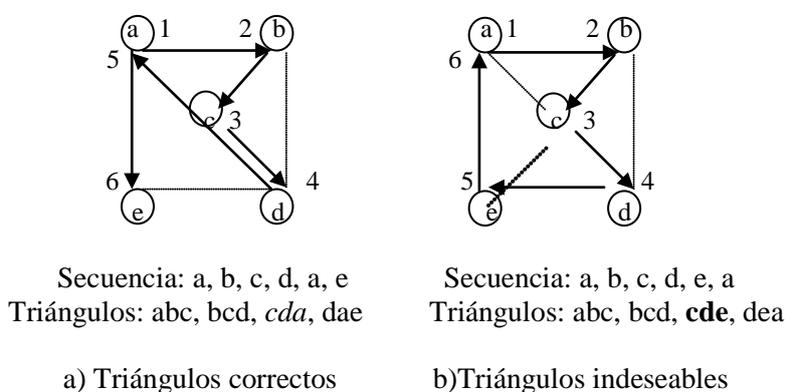


FIGURA 10. Secuencias de *triangle strips*

Para generar sistemáticamente un *triangle strip* a partir de recorrer recursivamente una jerarquía bitree, definiremos dos reglas a tener en cuenta para prevenir la generación de triángulos erróneos, que dicen:

1. Dos triángulos vecinos que comparten un lado por completo serán encolados consecutivamente en la secuencia y el lado compartido es explícito en la misma.
2. Dos triángulos vecinos que no comparten un lado por completo serán encolados consecutivamente en la secuencia, pero los lados vecinos deben ser ambos explícitos en la misma.

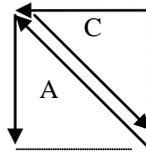
En el caso 2 se generará un triángulo línea entre ambos triángulos, pero esto no representa un problema.

La figura 11-a muestra los puntos de nivel N0 y dos posibles secuencias de comienzo. Se clasifican las diferentes posibles secuencias según los puntos inicial y final sean vértice entre catetos o entre cateto-diagonal (figura 11-b, 11-c, 11-d), y se realiza su división sistemática. Cuando un triángulo se divide sus lados deben ser mantenidos siempre y cuando estuvieran compartidos con sus vecinos. En la figura 11 se muestran con línea gris punteada.

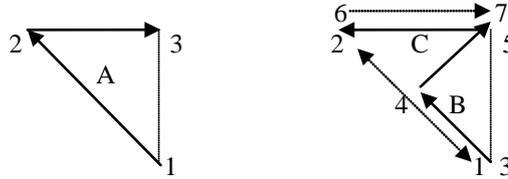
La figura 12 muestra la generación del *triangle strip* en cada paso de la subdivisión de una grilla de 3 x 3. La figura 12-a muestra las secuencias iniciales C y A; en la figura 12-b se subdivide la secuencia C en A y B; en la figura 12-c se subdivide la subsecuencia A en B y C; y así sucesivamente.

Además de la eficiencia del *rendering* de una malla triangular implementada con *triangle strip*, esta estructura ofrece un uso económico del espacio comparada con la implementación de cada triángulo por separado. En el ejemplo de la figura 12, en una grilla de 3x3 completamente triangulada hay 8 triángulos. El número de referencias usadas si se implementa cada triángulo por separado es 24,

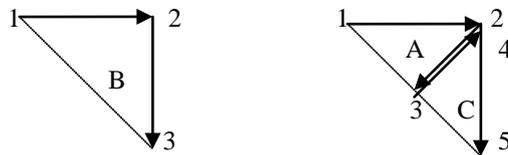
mientras que en la figura 12-f se usan solo 15. En el siguiente capítulo se expondrán algunos resultados obtenidos en un caso real.



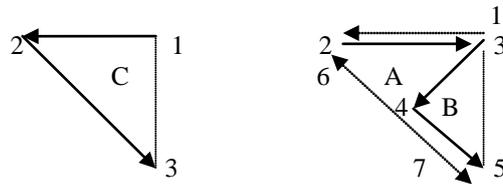
a) Secuencias iniciales



b) Secuencia A - Secuencia A dividida en B y C

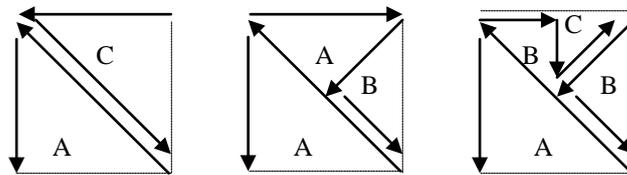


c) Secuencia B - Secuencia B dividida en A y C



d) Secuencia C- Secuencia C dividida en A y B

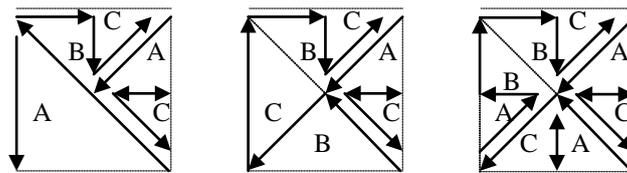
FIGURE 11. Clasificación de secuencias y su división



a)

b)

c)



d)

e)

f)

FIGURA 12. Generación del *triangle strip* en la subdivisión de una grilla de 3x3

5. RESULTADOS OBTENIDOS

Se realizaron pruebas con una grilla de 65 x 65 con datos topográficos reales de una zona que posee áreas planas así como también áreas montañosas, con alturas que van de 0 a 1445 metros. La figura 13-a muestra una parte del mapa con el total de la geometría y una textura mapeada. La figura 13-b muestra la malla triangulada con la totalidad de los puntos sin textura mapeada. En la figura 13-c puede verse como se simplifica el modelo eliminando los puntos totalmente redundantes que pueden ser descartados. Puesto que se elige una tolerancia de error cero el modelo es equivalente al original con tan sólo el 65% de los triángulos, el 60% de los puntos seleccionados y un 5% de los puntos agregados como ficticios para evitar *cracks*. En las figuras 13-d y 13-e se muestra el modelo simplificado con diferentes tolerancias de error.

La triangulación quadtree restringida agrega triángulos extras para evitar cracks y lo hace de una manera no local. La figura 14 muestra como partiendo del mismo conjunto de puntos seleccionados la triangulación bitree (figura 14-a) agrega menos puntos y en consecuencia triángulos que la triangulación quadtree restringida.

En la tabla 1 puede verse una comparación entre la triangulación bitree con el esquema de triangulación quadtree restringido presentado en [Pajarola98]. Para las distintas tolerancias de error se muestran los puntos agregados por cada modelo a fin de evitar *cracks*, los puntos que se transmitirían por red considerando que los puntos ficticios no se transmiten pues son deducibles, el número de triángulos y el numero de referencias del *triangle strip*.

Tol. Error	Puntos Agregados			Puntos a Transmitir			Triángulos			Referencias		
	BT	QT		BT	QT		BT	QT		BT	QT	
Orig.	0	0		4225	4225	0%	8192	8192	0%	13655	12972	-5%
0%	215	272	21%	2533	2805	10%	5296	5843	3%	8685	8467	-2%
1%	344	407	15%	1235	1642	25%	2899	3184	9%	4643	4788	3%
2%	281	360	22%	704	1064	34%	1736	2041	15%	2803	3049	8%
7%	161	241	33%	229	470	51%	647	884	27%	1037	1307	20%
10%	144	202	29%	182	384	53%	527	715	26%	851	1061	20%
14%	89	142	37%	88	230	62%	268	418	36%	443	601	26%
50%	10	19	47%	9	28	68%	23	41	44%	37	62	40%

TABLA 1. Comparación de resultados obtenidos con Bitree y Quadtree

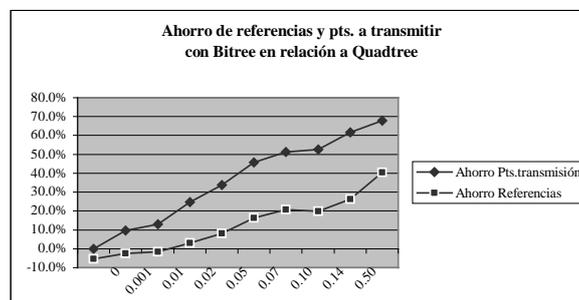
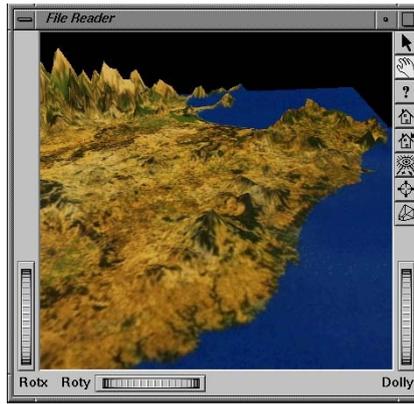
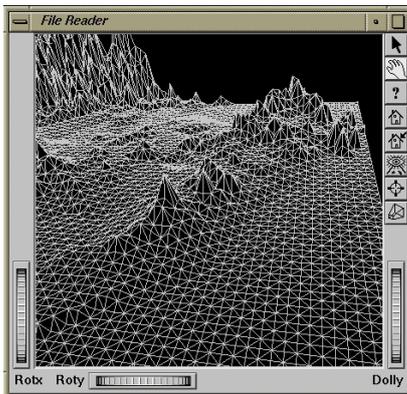


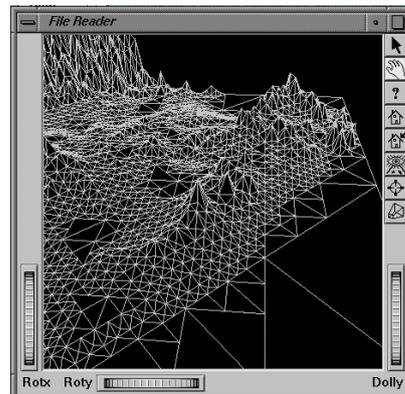
GRÁFICO 1. Ahorro obtenidos con Bitree



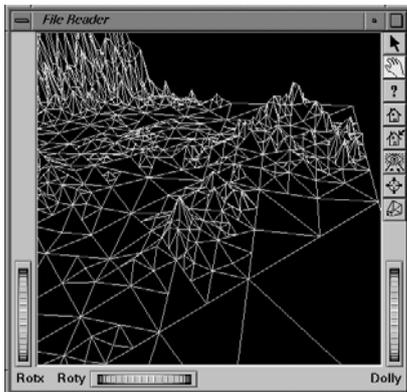
a) Malla original con textura mapeada



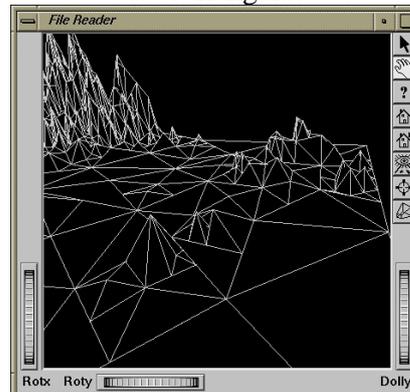
b) Malla original:
100% pts.seleccionados
100% triángulos



c) Tolerancia: 0%
Modelo equivalente al modelo original.
60% pts. Seleccionados + 5% pts. Ficticios
65% triángulos

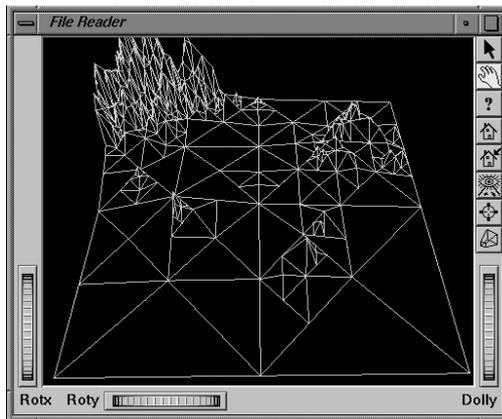


d) Tolerancia: 2%
17% pts. Seleccionados + 7% pts.ficticios
21% triángulos

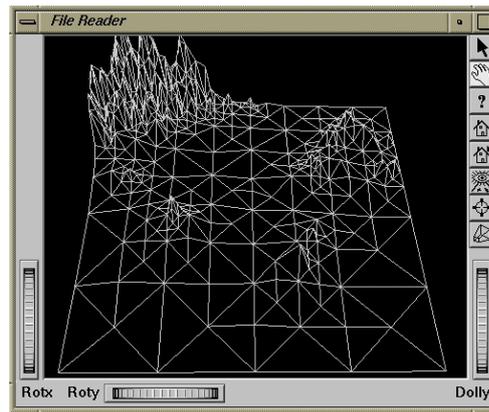


e) Tolerancia: 7%
5% pts.seleccionados + 4% pts.ficticios
8% triángulos

FIGURA 13. Triangulación según tolerancia de error



a) Bitree



b) Quadtree restringido

FIGURA 14. Puntos agregados para evitar *cracks*

En el gráfico 1 se observa que el ahorro obtenido con la estructura Bitree en los puntos a transmitir (considerando que los ficticios no se transmiten) y en las referencias de la estructura *triangle strip* que es la que se visualizará. Se puede observar que el ahorro aumenta a medida que la tolerancia de error es mayor, esto significa que cuanto más simplificado es el modelo mejor funciona el bitree con respecto al quadtree. Solo en el caso de tener un modelo de máxima resolución se observa que el bitree presenta una pequeña sobrecarga de referencias con respecto al quadtree, y esto es debido a que dado que no hay restricción de niveles entre triángulos vecinos el método de formación de la secuencia del *triangle strip* debe insertar triángulos línea que preparen la estructura para un posible salto de niveles.

En la tabla 2 puede compararse la implementación con *strip triangle* y con triángulos separados. Con la primera se obtiene en todos los modelos simplificados un promedio de 44% de ahorro con respecto a la implementación de triángulos separados. Esta mejoría se obtiene pese a que la implementación con *triangle strips* agrega triángulos línea, los cuales son aproximadamente un 40% del total de los triángulos del modelo.

Tolerancia Error	Triángulos	Referencias Con tr.sep.	Triángulos línea	Referencias con <i>tr.strip</i>
Malla original	8192	24576	5461	13655
0%	5296	15888	3446	8685
2%	1736	5208	1172	2803
7%	647	1941	441	1037

TABLA 2. Comparación de la implementación con *triangle strips* y triángulos separados

6. CONCLUSIONES

En este trabajo se presenta un modelo multiresolución basado en una jerarquía de triángulos. La selección de puntos del modelo puede hacerse según un criterio de aproximación independiente de la jerarquía y sin imponer restricciones. Para evitar *cracks* se agrega la mínima cantidad de puntos ficticios de manera que se minimicen los triángulos extras. Los resultados obtenidos muestran que se simplifica en un grado mayor que otros esquemas restringidos basados en quadtree como el presentado en [Pajarola98].

El modelo provee una estructura de triangulación que soporta triangulación adaptativa y multiresolución. Por esto puede ser integrado como núcleo de un sistema de visualización. Además, se adapta bien a problemas típicos de computación gráfica como son simplificación de mallas, refinamiento selectivo, compresión de mallas, LOD continuos y transmisión progresiva [Hoppe96].

Al igual que la geometría del terreno, es deseable contar con un modelo multiresolución de texturas. Como trabajo futuro se plantea la realización de un *framework* de un modelo multiresolución que considere tanto geometría como texturas. Asimismo, podrían obtenerse otros datos de un sistema de información geográfica (*GIS*) para posicionar sobre la escena carreteras, puentes, etc. de forma automática.

Agradecimientos

Los autores agradecen al Laboratorio de Sistemas de Información Geográfica de la Universidad de las Islas Baleares por suministrar los datos topográficos de las Islas Baleares.

BIBLIOGRAFÍA

[Casillas99] J.Casillas, J.Sevilla, J.Blasco, C.Pardo, C.Romero, J.Fernandez. "Implementación de un sistema visual para un simulador de helicóptero". Congreso Español de Informatica Grafica, CEIG 99, Jaen, España, junio de 1999

[DeHaemer91] M.DeHaemer, M.Zyda. "Simplification of Objects Rendered by Polygonal Approximations". *Computer & Graphics*, 15(2): pages 175-184, 1991.

[Erikson96] C.Erikson. "Polygonal Simplification: An Overview". TR96-016, *Department of Computer Science, University of North Carolina - Chapel Hill*. USA.

[Gross95] M. Gross, R. Gatti, O. Staadt. "Fast Multiresolution Surface Meshing". *Internal report no. 230. Institute for Information Systems. ETH Swiss Federal Institute of Technology*. Zürich, Switzerland.

[Hamann94] B.Hamann. "A Data Reduction Scheme for Triangulated Surfaces". *Computer Aided Geometric Design*, 11(2): pages 197-214, 1994.

[Heckbert94] P.Heckbert, M.Garland. "Multiresolution Modeling for Fast Rendering". *Proceedings of Graphics Interface '94, Banff, Alberta, Canada, May 1994*.

[Hernández99] L.Hernández, J.Taibo, A.Seoane. "Una aplicación para la navegación en tiempo real sobre grandes modelos topográficos". Congreso Español de Informatica Grafica, CEIG 99, Jaen, España, junio de 1999

[Herzen87] B.Von Herzen and A.Barr. "Accurate triangulations of deformed, intersecting surfaces". *Computer Graphics, (SIGGRAPH '87 Proceedings)*, pages 103-110, 1987.

[Hinker93] P.Hinker, C.Hansen. "Geometric Optimization". *Proceedings of Visualisation*, pages 189-195, 1993.

[Hoppe93] H.Hoppe, T.DeRose, T.Duchamp, J.McDonald, W.Stuetzle. "Mesh Optimization". *Computer Graphics, (SIGGRAPH '93 Proceedings)*, pages 19-26, 1993.

[Hoppe96] H.Hoppe. "Progressive Meshes". *Computer Graphics, (SIGGRAPH '96 Proceedings)*, pages 99-108, 1996.

[Hoppe97] H.Hoppe. "View-Dependent Refinement of Progressive Meshes". *Computer Graphics, (SIGGRAPH'97 Proceedings)*, pages 189-198, 1997.

[Lounsbery97] M.Lounsbery, T.DeRose, J.Warren. "Multiresolution Analysis for Surfaces of Arbitrary Topological Type". *ACM Transactions on Graphics*, v.16, no.1, 1997.

[Pajarola98] R.Pajarola. "Large scale Terrain Visualisation using the Restricted Quadtree Triangulation". Internal report.292 *Institute of Theoretical Computer Science. ETH Swiss Federal Institute of Technology*. Zürich, Switzerland, 1998.

[Schroeder92] W.Schroeder, J.Zarge, W.Lorensen. "Decimation of Triangle Meshes". *Computer Graphics, (SIGGRAPH'92 Proceedings)*, 26(2): pages 65-70, 1992.

[Varshney94] A.Varshney. "Hierarchical Geometric Approximations". PhD Thesis. *Department of Computer Science, University of North Carolina-Chapell Hill, USA*, 1994.