

Sistemas de Información. Representación de Superficies de Terrenos para su Visualización en Tres Dimensiones

Yusnier Valle Martínez
José Ortiz Rojas

Durante los últimos años la interdisciplinariedad ha encontrado un punto de encuentro entre tres especialidades, las Ciencia de la Información, la cartografía y la ciencia de la computación con el objetivo de representar y extraer información sobre conceptos geográficos, creando nuevos marcos teóricos y aplicaciones. En particular, el modelado jerárquico y la visualización interactiva y en tiempo real en tres dimensiones de grandes volúmenes de información relativa a superficies de terrenos, requieren de estructuras de datos y algoritmos muy eficientes para mantener la precisión y el realismo proporcionado por los datos. Partiendo de las limitaciones en cuanto a memoria y desempeño que poseen en la actualidad tanto los sistemas computacionales como los gráficos, la principal contribución del presente trabajo está enfocada en la representación eficiente de información relativa a Modelos Digitales de Terrenos que puedan ser cargados completamente en memoria principal. Se describe una estructura de datos similar a un Quadtree para conjuntos de datos de tamaño arbitrario, así como una variante de codificación de una triangulación basada en abanicos de triángulos que constituye la base para la extracción de triangulaciones multi-resolución para su visualización en tiempo real. Se utiliza la memoria de la Unidad de Procesamiento Gráfico para almacenar la información de relieve de las superficies, reduciendo significativamente los costos de almacenamiento de información de los modelos en memoria principal utilizando solamente algunos bytes adicionales para la topología de las mallas multi-resolución.

Palabras clave: Sistema de Información geográfica, representación de datos, almacenamiento, sistema de recuperación de información, Quadtree, visualización de terrenos, modelado jerárquico.

RESUMEN

ABSTRACT

In recent years interdisciplinarity has found a meeting point between three specialties, information science, cartography and computer science in order to represent and extract information about geographical concepts, creating new theoretical frameworks and applications. Hierarchical modeling and interactive and real-time 3D visualization of large volumes of information dealing with land surfaces require very efficient algorithms and data structures in order to maintain the accuracy of data and their realism. Taking into account the limitations of memory and performance of today's graphics platforms, the main contribution of this paper is focused on the efficient representation of terrain models that can be completely loaded into the main storage. We describe a novel hierarchical Quadtree-like data structure for variable size data sets, as well as a codification variant based on the triangle-fan which is the base for multi-resolution triangulations extraction for their visualization in real time. By taking advantage of the Graphics Processing Unit memory to store the surface contour data, our approach significantly reduces the space requirements in the main store by using only some additional bytes for the topology of multi-resolution mesh.

Keywords: Geographic Information System, data representation, storage, retrieval system, quadtree, viewing areas, hierarchical modeling.

Introducción

La dimensión geográfica es determinante en esta época para la vida humana; la gestión de datos en el contexto de las Tecnologías de la Información y las Comunicaciones (TICs) proporcionan eficiencia a los sistemas de información, disciplina propia de las Ciencias de la Información que a definición de investigadores como (Andreu; Ricart; y Valor, 1996), lo conceptualizan como un conjunto de procesos que maniobrando sobre una colección de datos estructurada de acuerdo a las necesidades de un proyecto o empresa, propicia recopilar, elaborar, y distribuir la información necesaria para la acción, dirección y control de las actividades en función de las estrategias del proyecto o empresa a crear.

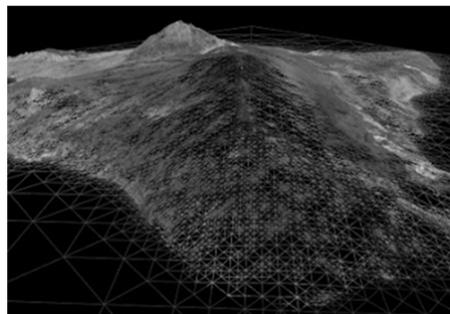
Las aplicaciones técnicas dependientes del proyecto o empresa son las que determinan la tecnología a aplicar en sus procesos, y el uso de una metodología. Las Bases de Datos (BD) y las herramientas construidas a partir de ellas constituyen los usos más generalizados al concebir un sistema de información. Los Sistemas de Información Geográfica (SIG) se derivan del sistema de información y sus aplicaciones; y se concibe a partir del acelerado desarrollo de las tecnologías como un sistema de hardware y software, elaborado para facilitar la obtención, gestión, manipulación, análisis, modelado y representación de datos referenciados para la resolución de problemas complejos la toma de decisiones.

Actualmente se integran los SIG a las utilidades de la tecnología de punta, en este caso a la representación en tercera dimensión (3D), que trascienden su simple uso como visores 3D permitiendo un mejor modelado de la realidad. De esta forma facilitan su utilización en un mayor rango de aplicaciones, como es el caso en los propios SIG en el manejo de la cartografía, la fotogrametría, los sistemas geo-referenciales (GPS), radares de penetración terrestre (GPR), magnetometría, imágenes de radar de alta resolución, entre otras.

Durante los últimos años han sido realizadas varias investigaciones en el área de las Ciencias de la Información Geográfica (GISciences), con el objetivo de representar y extraer información sobre conceptos

geográficos de diversa índole (Chen et al., 1997; Galton, 2009). En tal sentido se han creado metodologías tanto para la interpretación de dicha información, como para la representación efectiva del conocimiento adquirido por el procesamiento de ésta (Karalopoulos et al., 2005). El incremento constante de las capacidades de almacenamiento en los sistemas de cómputo, así como el perfeccionamiento de las técnicas de percepción remota y de visualización científica, ha permitido crear grandes volúmenes de datos de superficies de terrenos que requieren de gran eficiencia tanto en la gestión de los mismos, como en las técnicas a utilizar para su visualización en tiempo real en Tres Dimensiones (3D)¹, Figura 1 (izquierda). Ésta representación tridimensional favorece el análisis de los datos con el objetivo de adquirir información acerca de la influencia de las variaciones locales de las propiedades del terreno en la biosfera, microclima, ciclo hidrológico y la distribución de la actividad humana en el planeta, entre otras (Siska and Hung, 2004).

Las representaciones que combinan datos, información semántica y conocimiento en Sistemas de Información Geográfica (SIG) constituyen la base para el descubrimiento de nuevo conocimiento geográfico (Mennis and Peuquet, 2003). Un SIG consiste en una agrupación de hardware, software, datos y aplicaciones utilizadas por usuarios para gestionar, analizar y desplegar información geográficamente referenciada (Harmon and Anderson, 2003). Los Modelos Digitales de Terrenos (MDT) devienen en importante fuente de información en SIG, y consisten en una representación digital de las propiedades de la topografía de una superficie.



Los Modelos Digitales de Elevación (MDE)¹ se encuentran entre los MDT más conocidos, y comúnmente se almacenan en forma de Malla Regular (RG, en inglés), Redes Irregulares Trianguladas (TIN, en inglés) o curvas de nivel o de contorno, Figura 1 (derecha). A los efectos de este trabajo, solo son de interés los MDE almacenados en forma de RG, que pueden representarse por una cuaterna

$$(m, n, \delta_x, \delta_y), \quad m, n \in \mathbb{Z}^+, \\ \delta_x, \delta_y \in \mathbb{R}^+ : \text{un conjunto de } m \times n \\ \text{puntos en el plano } P_{ij}, \quad 0 \leq i < m, \\ 0 \leq j < n, \text{ cuyas coordenadas vienen} \\ \text{dadas por la función } g: G \rightarrow \mathbb{R}^2 \text{ de la} \\ \text{forma } g(P_{ij}) = (i \cdot \delta_x, j \cdot \delta_y).$$

En el presente trabajo se aborda el problema de la representación eficiente de grandes volúmenes de información relativa a superficies de terrenos en SIG, tanto en memoria externa como en memoria principal. Como novedad principal se proponen dos estructuras de datos basadas en quadtrees y Curvas de Recorrido del Espacio para codificar de manera eficiente y compacta los MDE a manejar, reduciendo significativamente la información a gestionar y por consiguiente los costos en memoria para su representación. Adicionalmente, se combinan las ventajas de esta codificación con los avances tecnológicos experimentados por las unidades de procesamiento gráfico o GPU su acrónimo del inglés, en la realización computacional del modelo propuesto, concretamente para la visualización interactiva y en tiempo real de MDE en 3D.

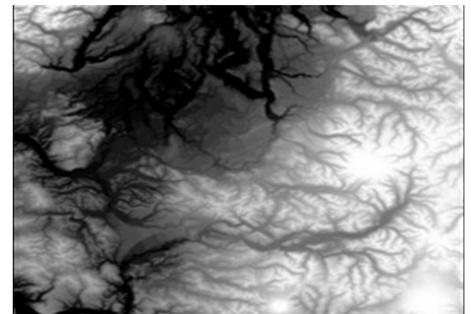


Figura 1. Ejemplo de triangulación correspondiente a la mayor isla del archipiélago de Hawaii (izquierda), donde se observa con claridad la variabilidad en su resolución dependiendo de las características del relieve de la superficie. Vista de un MDE correspondiente al área de Puget Sound (derecha), USA, disponible libremente para su uso.

¹ En general los acrónimos se utilizarán como se definen tanto para el plural como para el singular.

² En lo adelante se utilizará el término MDE indistintamente con el de superficie de terrenos.

Referentes Teóricos

En la representación de información y conocimiento en SIG se han utilizado una gran variedad de técnicas como los modelos de nubes (Li et al, 2000), reglas (Ragia and Berenzon, 2007) y ontologías (Fonseca et al, 2002; Garea and Oliva, 2007). En particular, la extensión de estructuras como el diagrama de Voronoi para incluir información semántica sobre sus regiones respectivas, constituye un ejemplo del uso de la representación del conocimiento basada en grafos en SIG (Palacio et al, 2003). El uso de grafos que codifican las relaciones de dependencia entre vértices para evitar la aparición de grietas en las superficies multi-resolución, constituye una de las principales contribuciones en lo que se conoce como Triangulaciones Quadtree Restringidas (RQT) (Lindstrom et al, 1996). En (Pajarola, 1998) se usan estas relaciones de dependencia para generar triangulaciones mínimas a partir de quadtrees de regiones construidos sobre grandes mallas regulares.

Röttger y colaboradores en (Röttger, 1998) también proponen un enfoque basado en quadtrees, centrado fundamentalmente en la representación eficiente de los datos en una estructura Quadtree codificada por una matriz de ceros y unos. La información relativa a las dependencias entre los vértices se representa de manera implícita. El costo total en memoria de esta variante de solución es muy reducido debido a que, además de los valores de altura del MDE, datos de textura y valores de error, solo se necesita un byte de información adicional por cada punto en el MDE.

El desarrollo creciente en cuanto a potencia de cálculo y velocidades de comunicación que experimentan en la actualidad las GPU, tiene un impacto significativo en el diseño y desempeño de las técnicas de representación y visualización de información relativa a MDE en 3D en tiempo real en nuestros días. Las propuestas de solución que hacen uso de la GPU actualmente están enfocadas principalmente en el *rendering* de fragmentos triangulares de superficies (denominados parches³) calculados en etapas de pre-procesamiento y almacenados en su memoria caché. Entre las propuestas más representativas en este sentido se encuentran las presentadas en (Cignoni et al, 2003; Bösch et al, 2009), quienes en sentido

general proponen esquemas de comunicación eficientes entre la CPU y la GPU para actualizar incrementalmente conjuntos de vértices e índices almacenados en esta última, disminuyendo significativamente los requerimientos de ancho de banda.

Modelo de representación de mde en 3D

El Quadtree de regiones es ampliamente utilizado para representar información relativa a MDE puesto que divide recursiva y jerárquicamente las superficies en regiones regulares. Cada nodo interior contiene exactamente 4 nodos hijos, donde cada uno de ellos representa uno de los 4 cuadrantes en que es dividida recursivamente la región. Dada una RG G_N de orden N , $N = 2^n$, $n \geq 0$, en (Asano et al, 1997) se define Curva de Recorrido del Espacio (SFC, en inglés) como una numeración de las celdas de con valores desde hasta, para algún. Una numeración de se define como una correspondencia uno-uno de la forma .

Una SFC numerada desde, para algún, es considerada recursiva (RSFC, en inglés) si o puede ser dividida en cuatro RSFCs siguiendo una condición dada (ej. la curva de Hilbert y el Orden Z son ejemplos de RSFC bien conocidos) (Asano et al, 1997). Tomando como base estas definiciones y a partir de una, en (Valle, 2008) se propone una implementación multi-arreglo de un Quadtree — denominado HQuadtree — basado en la SFC de Hilbert. En la estructura de datos propuesta, Figura 2, no es necesario almacenar información correspondiente a nodos ancestros, hijos o vecinos puesto que la jerarquía se manifiesta de manera implícita y su localización respectiva puede calcularse mediante simples operaciones matemáticas.

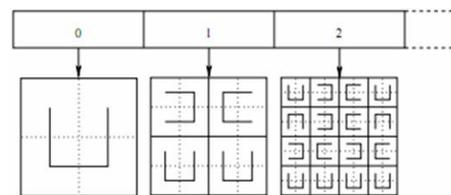


Figura 2. Primeros tres niveles en una representación multi-arreglo de un HQuadtree

Una ventaja importante de esta representación consiste en que cada patrón de Hilbert creado en cada paso de la subdivisión representa 4 cuadrantes en la estructura, razón por la cual la información correspondiente a los cuadrantes de máxima resolución no necesita almacenarse explícitamente, representando esto una disminución significativa en los requerimientos de memoria de la propuesta.

A. FTTree

La estructura de datos Quadtree propuesta por (de Berg et al., 2000, cap. 14)⁴ es creada a partir de un cuadrado inicial con extremos en $(0,0)$, $(0,U)$, $(U,0)$ y (U,U) , con $U = 2^j$ para algún entero positivo j . De forma similar a un Quadtree, un FTTree (esto es *Four-Two Tree: Árbol 4-2*) se define como sigue:

Definición 3.1. Un FTTree es un árbol en el que cada nodo interno tiene dos o cuatro hijos

- Cada nodo en el FTTree corresponde a un cuadrante.
- Si un nodo v tiene hijos, entonces sus correspondientes cuadrantes son dos o cuatro cuadrantes del correspondiente de v .

El proceso de creación de un FTTree es llevado a cabo en forma similar a un Quadtree. Tres tipos de cuadrantes pueden resultar en el proceso de división:

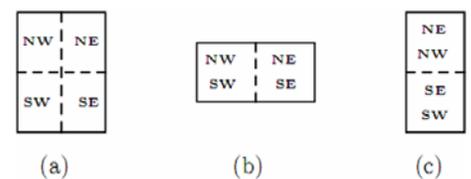


Figura 3. Tipos de cuadrantes con diferentes orientaciones en un FTTree.

I. Con cuatro hijos, Figura 3 (a): cada nodo hijo puede ser identificado como Suroeste (SW), Noroeste (NW), Noreste (NE) y Sureste (SE).

II. Orientado horizontalmente con dos hijos, Figura 3 (b): el nodo hijo que almacena el cuadrante a la izquierda (derecha) es identificado como NW y SW (NE y SE).

III. Orientado verticalmente con dos hijos, Figura 3 (c): el nodo hijo que almacena el

³ Traducido del término en inglés *parches*.

⁴ En lo adelante el término *Quadtree* refiere esta propuesta

cuadrante superior (inferior) es identificado como NW y NE (SW y SE).

Notación 3.1. Sea G_n^m que denota una RG de orden $m \times n$, $m, n > 1$.

El punto de partida para la creación de un FTTTree lo constituye. Sea el cuadrante inicial con extremos en $(x_i, y_i), (x_f, y_i), (x_f, y_f), (x_i, y_f)$. Notar que si entonces el FTTTree consiste en solo una hoja donde el cuadrante es almacenado. Sean w y h los 4 cuadrantes de σ , y sea (x_p, y_p) el punto por el cual un cuadrante dado será dividido. El pseudocódigo del Algoritmo 1 como es llevado a cabo el proceso.

El procedimiento DIVIDECUADRANTE comienza calculando la relación entre el ancho y la altura de los nuevos cuadrantes que serán creados, y es llamado recursivamente mientras un cuadrante sea divisible. Si esta relación proporcional (r) es mayor que un umbral, de tolerancia especificado, debe ser dividido horizontal o verticalmente en dos cuadrantes dependiendo de sus medidas. En caso contrario, el valor de la relación r es cercano a 1 (los cuadrantes a ser creados son prácticamente cuadrados), σ debe ser dividido en cuatro cuadrantes.

El procedimiento DIVIDECUADRANTE en el Algoritmo 1 es llamado recursivamente por cada nodo siempre que estos puedan dividirse, o sea, sus medidas sean distintas de $\delta_x \times \delta_y$. El criterio de división usado en el proceso de construcción de un FTTTree garantiza que si $m = n = 2^n + 1$, $n \geq 1$, la estructura resultante es un Quadtree. El mecanismo de búsqueda de vecinos es similar al de un Quadtree, solo se verifica que si v es un HijoSE (HijoSW) de su nodo padre P_v , entonces para los nodos de tipo I y III — Figura 3 — su vecino norte correspondiente es el HijoNE (HijoNW) de. En otro caso, si es de tipo II, o es de tipo II y es un HijoNE (HijoNW) de, el algoritmo se comporta exactamente

como en un Quadtree. De forma análoga ocurre para los vecinos en las demás direcciones.

Visualización

La primitiva básica usada en este trabajo para el *rendering* de superficies es la conocida en OpenGL como TRIANGLE-FAN (en lo adelante «abanico de triángulos»). Cada patrón en un HQuadtree construido a partir de una como en la sección III, se almacena un abanico de triángulos codificado con solo un byte de memoria. La Figura 4 ilustra algunas de las configuraciones que pueden resultar en el proceso de codificación jerárquico.

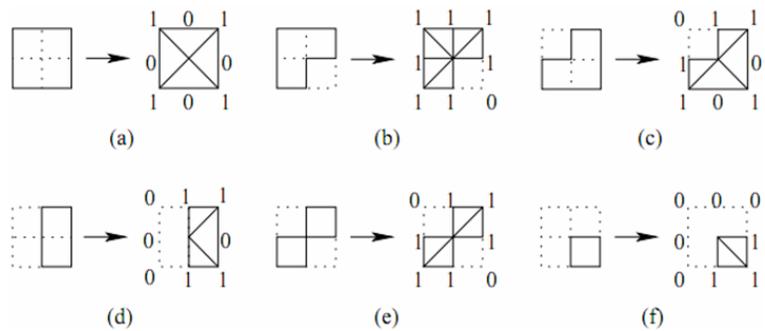


Figura 4. Seis tipos diferentes de abanicos de triángulos. Cada bit marcado con 1 significa que el correspondiente punto en G_N está activo y debe incluirse en la triangulación final, 0 en otro caso.

Algoritmo 1: División de un cuadrante dado σ .

procedure DIVIDECUADRANTE(σ : **quadrant**, τ : **float**)

```

1:  $w = (\text{ANCHO}(\sigma) \gg 1)$ 
2:  $h = (\text{ALTO}(\sigma) \gg 1)$ 
3:  $r = \frac{\text{MAX}(w,h)}{\text{MIN}(w,h)}$ 
4: if  $r > \mu$  then
5:   if  $\text{ANCHO}(\sigma) > \text{ALTO}(\sigma)$  then
6:      $\sigma_{sw}^{nw} \leftarrow [(x_f, y_i), (x_i, y_i), (x_i, y_p), (x_f, y_p)]$ 
7:      $\sigma_{se}^{ne} \leftarrow [(x_f, y_p), (x_i, y_p), (x_i, y_f), (x_f, y_f)]$ 
8:   else
9:      $\sigma_{sw}^{se} \leftarrow [(x_f, y_i), (x_p, y_i), (x_p, y_f), (x_f, y_f)]$ 
10:     $\sigma_{nw}^{ne} \leftarrow [(x_p, y_i), (x_i, y_i), (x_i, y_f), (x_p, y_f)]$ 
11:   end if
12: else
13:    $\sigma_{sw} \leftarrow [(x_f, y_i), (x_p, y_i), (x_p, y_p), (x_f, y_p)]$ 
14:    $\sigma_{nw} \leftarrow [(x_p, y_i), (x_i, y_i), (x_i, y_p), (x_p, y_p)]$ 
15:    $\sigma_{ne} \leftarrow [(x_p, y_p), (x_i, y_p), (x_i, y_f), (x_p, y_f)]$ 
16:    $\sigma_{se} \leftarrow [(x_f, y_p), (x_p, y_p), (x_p, y_f), (x_f, y_f)]$ 
17: end if
end procedure

```

Cada bit de información es asociado a un punto en G_N (8 en total por cada byte), determinando cuando estos deben ser incluidos o no en la triangulación requerida para ser visualizada. Comenzando con la esquina superior-izquierda, se define una correspondencia uno-uno de izquierda a derecha en la secuencia de bits de cada byte (ej. la secuencia de bits en la Figura 4 a) es 10101010, indicando que los puntos superior-izquierdo, inferior-izquierdo, inferior-derecho y superior-derecho deben ser incluidos en la triangulación).

A. Triangulación

Adoptando una métrica de error según (Pajarola, 1998), quiénes codifican un grafo de dependencias entre vértices en la jerarquía para garantizar la continuidad en las triangulaciones generadas en lo que se conoce como «saturación del error», en el presente trabajo proponemos un enfoque clásico *top-down* para extraer triangulaciones multi-resolución a partir de un HQuadtree H . El Algoritmo 2 muestra como se lleva a cabo el proceso.

El algoritmo comienza calculando el tamaño

s (inicialmente $N - 1$) del cuadrante en la posición h dada (inicialmente $[1][0,0]$), seguido de los valores de x e y , las coordenadas del valor e_v correspondiente en el arreglo bidimensional e que almacena los errores por vértices calculados previamente. Si un cuadrante en la posición h debe ser triangulado, en la línea 5 el byte correspondiente es iniciado con sus esquinas en 1, Figura 4.

a). El procedimiento

CHEQUEAR_PUNTOS_MEDIOS

toma un tiempo constante en chequear si los puntos medios de las aristas del cuadrante deben ser activados (ej. sea α el abanico en la Figura 4 b), el punto medio-izquierdo es activado mediante un simple OR lógico: $\alpha \mid 64$). Las líneas de la 7 a la 13 garantizan el proceso recursivo chequeando si los cuadrantes hijos deben ser incluidos en la triangulación. Este proceso tiene dos partes:

1. Línea 9: el procedimiento

ACTIVA_VERTICE_NW(de forma similar para los hijos SW, NE y SE) toma tiempo constante en desactivar el punto superior-

izquierdo del abanico padre (ej. el abanico en la Figura 4 c)). Esto se lleva a cabo mediante las siguientes operaciones:

a.: almacena el estado del punto superior-derecho.

b.: almacena el estado del punto inferior-izquierdo.

c.: desactiva los puntos medio-izquierdo, superior-izquierdo y medio-superior, y restablece los estados almacenados en m_τ y m_τ .

2. Línea 10: realización de un llamado recursivo en el correspondiente cuadrante hijo.

Al final de la línea 13 la jerarquía de abanicos de triángulos ha sido actualizada dependiendo del valor de τ especificado. Esta representación compacta de las triangulaciones trae consigo que el tratamiento de los grandes volúmenes de información relativa a los MDE a manejar, se realice de manera eficiente en términos de costos en memoria principal. La Figura 5 muestra una posible jerarquía resultante sobre un HQuadtree construido a partir de una RG de orden 9×9 , y su correspondiente triangulación.

Algoritmo 2: Extracción de una triangulación multi-resolución a partir de H dadas una posición h y un umbral τ determinado.

procedure TRIANGULAR (h, τ)

```

1:  $s \leftarrow \text{POW}(H.\text{niveles} - h.\text{nivel})$ 
2:  $x \leftarrow h.f * N + (N \gg 1)$ 
3:  $y \leftarrow h.c * N + (N \gg 1)$ 
4: if  $e[x * N + y] > \tau$  then
5:    $H[h] \leftarrow 170$  //esto es 10101010
6:   CHEQUEAR_PUNTOS_MEDIOS( $h$ )
7:   if  $h.\text{nivel} < H.\text{niveles} - 1$  then
8:     if  $e\left[\left(x - \left\lfloor \frac{s}{4} \right\rfloor\right) * N + \left(y - \left\lfloor \frac{s}{4} \right\rfloor\right)\right] > \tau$  then
9:       ACTIVA_VERTICE_NW( $h$ )
10:      TRIANGULAR(HIJO_NW( $h$ ),  $\tau$ )
11:    end
12:    ...
13:  end
14:  if  $H[h] \neq 0$  then
15:    ADICIONAR_ABANICO( $H[h]$ )
16:  end
17: end
end procedure

```

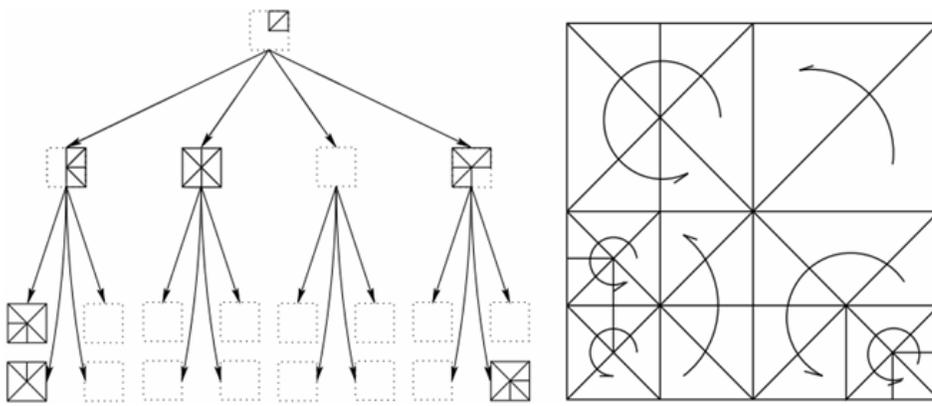


Figura 5. Jerarquía de abanicos de triángulos (izquierda) y su correspondiente triangulación representada mediante abanicos de triángulos (derecha). Los cuadrantes representados completamente con líneas discontinuas son marcados con 0 y se descartan en la línea 14 del Algoritmo 2.

B. Modelado de superficies de terrenos en RAM

La eficiencia con que pueda ser manejada la información relativa a las superficies trianguladas deviene en elemento crítico en varias aplicaciones en el campo de la visualización científica, realidad virtual, entre otras. El proceso de representación de MDE en memoria principal consta de tres momentos fundamentales:

I. El MDE fuente se divide en bloques de orden $2^k + 1$, $k \geq 1$ por medio de un FTTree F que constituye la base para manejar dinámicamente la información. A partir de F puede determinarse en tiempo real que bloques se encuentran en el rango del observador en una escena.

II. Por cada nodo hoja de F se crea un HQuadtree H que manejará las distintas triangulaciones multi-resolución, donde los bloques más cercanos al punto de observación brindarán un mayor nivel de detalle que las alejados o fuera de ese rango.

III. Se crea una estructura en la memoria de la GPU para almacenar cada conjunto de valores de alturas perteneciente a cada bloque, así como índices a estos que permitirán su procesamiento en tiempo real. Esta estructura es actualizada dinámicamente según cambien los niveles de aproximación requeridos por cada triangulación.

Resultados

Con el objetivo de ilustrar la efectividad del modelo propuesto en la representación de grandes volúmenes de información relativa a MDE en SIG, para su visualización interactiva y en tiempo real en 3D, algunas pantallas fueron tomadas en tiempo de ejecución de una aplicación prototipo, en un sistema con un Core 2 Duo E7200 a 2.53 GHz y una tarjeta gráfica NVidia GeForce 9400 GT de 512 Mb de RAM. Los datos usados corresponden a determinadas áreas de la superficie terrestre con los valores de alturas escalados arbitrariamente para resaltar las características del relieve, Figura 6.

En la Tabla 1 se presentan datos preliminares de análisis del desempeño de la técnica propuesta para los DEM de la Figura 6. Como puede apreciarse, para DEM de tamaño mediano ($4K \times 4K$, alrededor de 16 millones de valores de altura) se obtiene un desempeño promedio de más de 100 marcos por segundo, comparable con las demás propuestas existentes en la literatura.

La principal contribución del presente trabajo está enfocada en la representación eficiente de grandes volúmenes de información relativa a MDE que pueden ser cargados completamente en memoria principal. Los resultados se resumen en dos puntos fundamentales:

1. FTTree: una estructura de datos similar a un Quadtree que elimina la restricción sobre el orden de las RG representativas de los MDE base para su creación, significando un ahorro significativo en memoria puesto que no es necesaria la inclusión de puntos adicionales para la construcción de la estructura.

2. Una jerarquía de abanicos de triángulos modelada por un HQuadtree para reducir el consumo total de memoria requerida por los modelos multi-resolución de terrenos. Debido al hecho de que cada abanico de triángulos codifica la información necesaria para nueve puntos en el MDE, y cuatro cuadrantes en el modelo jerárquico, el número total de bytes requeridos por la jerarquía completa para una malla regular como en la sección III es:

$$M_b = \sum_{i=0}^{n-1} 4^i = \frac{(4^{(n-1)+1} - 1)}{4 - 1} \quad (1)$$

que es menor que los N^2 bytes — aproximadamente solo el 25% — necesarios por el enfoque RQT de (Röttger et al, 1998), una propuesta muy bien conocida por su consumo de memoria extremadamente bajo.

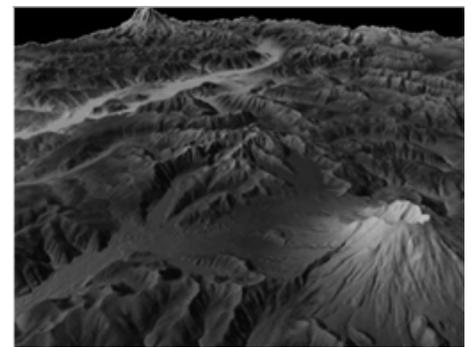
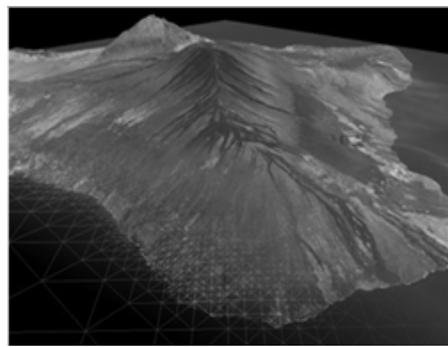
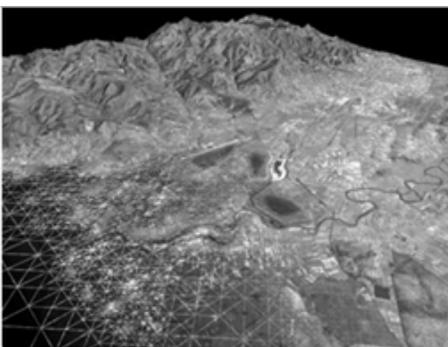


Figura 6. Vista texturizada de tres conjuntos de datos utilizados en el trabajo. Área alrededor de la ciudad de Baia Mare, Norte de Transilvania, Rumanía (izquierda), isla del archipiélago de Hawaii (centro) y área de Puget Sound, Washington, USA (derecha).

Tabla 1: Resultados que muestran el comportamiento en tiempo real de nuestra propuesta. Se ilustra el valor promedio de millones de triángulos manejados en tiempo real con su respectivo desempeño en marcos por segundo, para un número determinado de pantallas durante un recorrido sobre cada conjunto de datos. Todas las pruebas fueron realizadas para un tamaño de imagen de 1024 x 768 pixels.

Datos	Resolució	Textura	# de pantallas	Prom. de mts	Prom. fps
Hawaii	4K × 4K	2K × 2K	7694	88	159
Puget Sound	4K × 4K	2K × 2K	4991	124	109
Costa Rica	2K × 2K	2K × 2K	9861	48	157
Baia Mare	2K × 2K	2K × 2K	10829	35	178

Conclusiones y trabajo futuro

- El desarrollo experimentado por las Ciencias de la Información y sus manifestaciones interdisciplinarias, como los sistemas computacionales, así como por las técnicas de adquisición de datos de la topografía terrestre, ha posibilitado la disposición por parte de los investigadores de grandes volúmenes de datos de superficies de terrenos. El manejo y la representación de la información asociada a éstos, requiere de técnicas eficientes en cuanto a velocidad de procesamiento y almacenamiento se refieren.

- La representación y visualización 3D de la información geográfica en el software SIG proporciona una perspectiva realista de fenómenos o situaciones críticas que ayudan a una gran variedad de especialistas en la planificación y preparación para, de forma proactiva, predecir y/o evaluar posibles resultados a obtener en sus proyectos.

- Las estructuras de datos propuestas, en conjunto con el esquema de codificación de las triangulaciones, constituyen el soporte para la representación de la información relativa a los MDE modelados. Además de los datos de altura de las superficies, se codifican las relaciones de dependencia tanto horizontales (vecindades) como verticales (padres e hijos) entre vértices de manera implícita, significando una reducción significativa en los costos en memoria. Además de ser visualizada en tiempo real en 3D, la estructuración de la información es adecuada para su interpretación en la solución de problemas como el cálculo de acumulación de precipitaciones, la construcción y visualización de modelos geológicos tridimensionales o la creación de mapas de visibilidad de superficies, entre otros.

- En la bibliografía consultada y referenciada no se evidencian desarrollos de esta clase

de aplicaciones de los SIG en nuestro país, lo que representa un reto actual para la comunidad de desarrolladores SIG por el impacto que tienen estas tecnologías en los procesos de toma de decisiones en organismos cuyo campo de actuación reviste vital importancia para nuestra sociedad.

- El trabajo en un futuro inmediato involucra la extensión de la propuesta para llevar a cabo el *rendering* interactivo y eficiente en 3D de MDE, que por su extensión no puedan ser representados completamente en RAM, contemplando la incorporación de técnicas de compresión de datos en memoria externa para su representación eficiente en ese aspecto. En términos generales, para lograr un efectivo aprovechamiento de las potencialidades que ofrecen los actuales sistemas computacionales, se hace necesario que el volumen de información que se visualice en un instante determinado constituya un mínimo imprescindible para cumplir con parámetros de calidad previamente fijados.

Referencias Bibliograficas

Andreu, R., Ricart, J. E. y Valor, J. (1996): Estrategia y Sistemas de Información, 2ª Edición, McGraw-Hill, pag. 13

Chen, H., Smith, T. R., Larsgaard, M. L., Hill, L. L. and Ramsey, M. (1997). A geographic knowledge representation system for multimedia geospatial retrieval and analysis [versión electrónica]. International Journal on Digital Libraries 1(2), 132-152. Disponible en <http://ai.arizona.edu/intranet/papers/AGeographicKnowledge.pdf>

Galton, A. (2009). Spatial and temporal knowledge representation [versión electrónica]. Earth Science Informatics 2 (3), 169-

187. Disponible en <http://ali.cmi.ac.in/isla2010/slides/ag3.pdf>

Karalopoulos, A., Kokla, M., and Kavouras, M. (2005). Comparing Representations of Geographic Knowledge Expressed as Conceptual Graphs [versión electrónica]. Proceedings of the 1st International Conference on GeoSpatial Semantics (GeoS 2005), Lecture Notes in Computer Science 3799, Springer. Disponible en http://www.ntua.gr/ontogeo/publications/Karalopoulos_Kokla_Kavouras_FinalVersion.PDF

Siska, Peter P. and Hung, I-Kuai (2004). ADVANCED DIGITAL TERRAIN ANALYSIS USING ROUGHNESS-DISSECTIVITY PARAMETERS IN GIS [version electrónica]. Proceedings of the 24th Annual ESRI International User Conference, San Diego, California. Disponible en http://www.faculty.sfasu.edu/hungikua/pdf/research/ESRI_CONF2004.pdf

Mennis, J. and Peuquet, D. J. (2003). The Role of Knowledge Representation in Geographic Knowledge Discovery: A Case Study [version electrónica]. Transactions in GIS 7(3), 371-391. Disponible en <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.92.9590&rep=rep1&type=pdf>

Harmon, J. E. and Anderson, S. J. (2003). The Design and Implementation of Geographic Information Systems (Wiley).

Li, D.R., Di, K.C. and Li, D.Y., 2000. Knowledge representation and uncertainty reasoning in GIS based on cloud models [versión electrónica].

Referencias

- The 9th International Symposium on Spatial Data Handling, Beijing, (8), 1-12. Disponible en http://shoreline.eng.ohio-state.edu/dkc/SDH2000_Li_Di.pdf
- Ragia L. and Berenzon V. (2007). Rules Extraction and Representation for Geographic Information Systems [version electrónica]. In 10th AGILE International Conference on Geographic Information Science. 8-11 May, Aalborg, Denmark. Disponible en <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.94.2462&rep=rep1&type=pdf>
- Fonseca, F. T., Egenhofer, M. J., Agouris, P. and Câmara, G. (2002). Using Ontologies for Integrated Geographic Information Systems [version electrónica]. Transactions in GIS, 6(3), 231-257. Disponible en http://www.personal.psu.edu/faculty/f/u/fuf1/publications/Fonseca_TGIS_2002.pdf
- Garea E, Oliva R. (2007). Geo-ontologies as integration structures of knowledge data and metadata [version electrónica], CD Proceedings of the 2nd International Conference of Geospatial Semantics (GEOS 2007), Ciudad de Mexico, Noviembre, ISBN 978-970-36-0461-6. Disponible en http://uh-cu.academia.edu/RafaelOlivaSantos/Papers/536719/Geo-ontologies_as_Integration_Structures_of_Knowledge_Data_and_Metadata
- Palacio, M. P., Sol, D. and González, J. (2003). Graph-Based Knowledge Representation for GIS Data [version electrónica]. Proceedings of the 4th Mexican International Conference on Computer Science, 117-125. Disponible en <http://biblioteca.coqcyt.gob.mx/bvic/Captura/upload/GRAPH-BASED-KNOWLEDGE-ANACON0001.pdf>
- Berg, M., van Kreveld, M., Overmars, M., and Schwarzkopf, O. (2000). Computational Geometry: Algorithms and Applications (Springer, 2 edition).
- Lindstrom, P., Koller, D., Ribarsky, W., Hodges, L. F., Faust, N., and Turner, G. A. (1996). Real-Time, Continuous Level of Detail Rendering of Height Fields [version electrónica]. In SIGGRAPH '96: Proceedings of the 23rd annual conference on computer graphics and interactive techniques, New York, NY, USA, 109-118. Disponible en <http://citeseer.ist.psu.edu/viewdoc/summary?doi=10.1.1.15.7224>
- Pajarola, R. (1998). Large Scale Terrain Visualization Using The Restricted Quadtree Triangulation. In VIS '98: Proceedings of the Conference on Visualization '98 [version electrónica], Los Alamitos, CA, USA, 19-26. Disponible en <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.45.1402>
- Röttger, S., Heidrich, W., Slussallek, P., and Seidel, H.-P. (1998). Real-Time Generation of Continuous Levels of Detail for Height Fields [version electrónica]. In Proceedings of the 6th International Conference in Central Europe on Computer Graphics and Visualization, 315-322. Disponible en <http://citeseer.ist.psu.edu/viewdoc/summary?doi=10.1.1.116.8235>
- Asano, T., Ranjan, D., Roos, T., Welzl, E., and Widmayer, P. (1997). Space Filling Curves and Their Use in the Design of Spatial Data Structures [version electrónica]. Theoretical Computer Science, 181(1), 3-15. Disponible en <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.56.3068>
- Cignoni, P., Ganovelli, F., Gobbetti, E., Marton, F., Ponchio, F., and Scopigno, R. (2003b). Planet-Sized Batched Dynamic Adaptive Meshes (P-BDAM) [version electrónica]. In Proceedings of IEEE Visualization, Seattle, WA, USA, 147-155. Disponible en <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.10.9754>
- Bösch, J, Goswami, P, Pajarola, R (2009). RASTeR : Simple and Efficient Terrain Rendering on the GPU [version electrónica]. In Proceedings of the 30th annual conference of the European Association for Computer Graphics (EUROGRAPHICS 2009), Munich, Germany, 35-42. Disponible en <http://vmml.ifi.uzh.ch/files/pdf/publications/RASTeR.pdf>
- Valle, Y. (2008). An Efficient Restricted Quadtree Triangulation Based on the Hilbert Space-filling Curve (Eurographics Association). In CEIG08: Congreso Español de Informática Gráfica, Barcelona, Spain, 241-244.

Recibido: 18 de julio de 2011.
Aprobado en su forma definitiva:
22 de octubre de 2011

Ms.C. Yusnier Valle Martínez

Universidad de las Ciencias Informáticas (UCI)

País: Cuba

Correo electrónico: <yvm@uci.cu>

Dr.C. José Ortiz Rojas

Universidad de las Ciencias Informáticas (UCI)

País: Cuba

Correo electrónico: <jortiz@uci.cu>
