

Introducción a la Geometría Computacional

Dr. Eduardo A. RODRÍGUEZ TELLO

CINVESTAV-Tamaulipas

8 de enero del 2013



Cinvestav

- 1 Introducción a la Geometría Computacional
 - ¿Qué es la Geometría Computacional?
 - Ejemplo de un problema típico de GC
 - Análisis de algoritmos, notación estándar
 - Áreas de aplicación de la GC
 - Fortalezas de la GC
 - Limitantes de la GC



Definición

- El término Geometría Computacional (GC) fue quizás utilizado por primera vez por Marvin Minsky en su libro *Perceptrons* (reconocimiento de patrones) y también ha sido usado frecuentemente para describir algoritmos para manipular curvas y superficies en modelado de sólidos
- Sin embargo, su uso más reconocido es para describir la rama de las ciencias computacionales encargada del estudio sistemático de algoritmos y estructuras de datos necesarios para la solución eficiente de problemas que implican como entrada y salida objetos geométricos



Orígenes

- La GC se originó a finales de los años 70 y ha continuado su rápido desarrollo hasta la fecha
- Históricamente, la GC se desarrolló como una generalización del estudio de algoritmos para búsqueda y ordenamiento en un espacio 1-dimensional, a problemas que involucran entradas multi-dimensionales
- Debido a su historia, el campo de la GC se ha enfocado principalmente en problemas en un espacio 2-dimensional y en un grado menor en el espacio 3-dimensional



Orígenes ...

- En GC cuando se consideran problemas en espacios multi-dimensionales, se asume generalmente que la dimensión del espacio es una constante pequeña (≤ 10)
- No obstante, trabajos recientes en esta área han considerado un conjunto limitado de problemas en espacios altamente dimensionales
- Debido a que la GC ha sido desarrollada por investigadores cuya formación fue en algoritmos discretos (opuesto al análisis numérico) el área también se ha enfocado más en la naturaleza discreta de los problemas geométricos



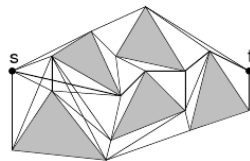
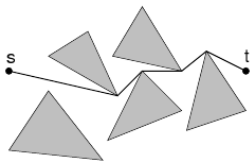
Orígenes ...

- La GC se ocupa sobre todo del estudio de objetos rectos o planos (líneas, segmentos de línea, polígonos, planos, y poliedros) y objetos curvos simples como círculos
- Al contrario del modelado de sólidos que se enfoca en el estudio de curvas y superficies y sus representaciones



Problema del camino más corto

- Dado un conjunto de obstáculos poligonales en el plano, encontrar el camino más corto evitando los obstáculos desde un punto origen s hasta un punto destino t dados



Problema del camino más corto ...

- Es posible reducir este problema al del camino más corto en un grafo y resolverlo utilizando un algoritmo no geométrico como el de Dijkstra
- Sin embargo, el resolver este problema en su dominio geométrico permite encontrar soluciones más eficientes
- Esto sucede con muchos otros problemas, de ahí el gran interés en desarrollar algoritmos geométricos (geometría computacional)



Análisis de algoritmos, notación estándar

- En GC, al igual que otras áreas de las ciencias computacionales, ha sido tradicionalmente utilizada como medida de la calidad de un algoritmo el tiempo asintótico de ejecución del peor caso $O(f(n))$
- $O(f(n))$, la clase C_1 de funciones tales que para cualquier $g \in C_1$ existe una constante c_g tal que $f(n) \geq c_g g(n)$ para todas excepto un número finito de n 's. En otros términos $O(f(n))$ es la clase de funciones que son a lo más tan grandes como $f(n)$
- Por lo tanto, un algoritmo corriendo en tiempo $O(n)$ es más rápido que uno que corre en $O(n \log n)$ que a su vez es mejor que uno ejecutándose en tiempo $O(n^2)$



Análisis de algoritmos, notación estándar

- Algunas otras notaciones también empleadas son:
- $o(f(n))$, la clase C_2 de funciones tales que para cualquier $g \in C_2$, $\lim_{n \rightarrow \infty} g(n)/f(n) = 0$, i.e. $o(f(n))$ es la clase de funciones que son menores que $f(n)$
- $\Omega(f(n))$, la clase C_3 de funciones tales que para cualquier $g \in C_3$ existe una constante c_g tal que $f(n) < c_g g(n)$ para todas excepto un número finito de n 's, es decir, $\Omega(f(n))$ es la clase de funciones que son al menos tan grandes como $f(n)$



Análisis de algoritmos, notación estándar

- El problema del camino más corto en GC puede resolverse en tiempo $O(n^2 \log n)$ mediante un algoritmo relativamente simple y en $O(n \log n)$ con un algoritmo un poco más complejo
- En algunas ocasiones también es utilizada como medida de la calidad de un algoritmo en GC el tiempo de ejecución del caso promedio
- Sin embargo, para muchos tipos de problemas (éste por ejemplo) es difícil definir distribuciones de probabilidad que sean tanto fáciles de analizar como representativas de los datos de entrada típicos.



Ejemplos de aplicaciones de la GC

- Existen diversos campos de las ciencias computacionales que están relacionados con la solución de problemas de naturaleza geométrica
- Éstos incluyen:
 - Gráficas computacionales
 - Visión por computadora y procesamiento de imágenes
 - Robótica
 - Diseño y manufactura asistidos por computadora
 - Sistemas de información geográficos, ...



Ejemplos de aplicaciones de la GC ...

- Una de las metas de la GC es proveer las herramientas geométricas básicas que esas áreas requieren para la construcción de aplicaciones prácticas
- Aun cuando se ha tenido un progreso significativo para cumplir con esta meta, todavía existen un gran número de problemas geométricos por resolver



Fortalezas de la GC

- Desarrollo de herramientas geométricas: Antes de que la GC existiera, se desarrollaron muchas soluciones *ad hoc* a problemas geométricos computacionales, algunas eficientes, otras no tanto e incluso algunas incorrectas. Debido al rigor matemático de la GC se han hecho grandes progresos en el establecimiento de soluciones algorítmicas correctas y de probada eficiencia para muchos de estos problemas
- Énfasis la eficiencia comprobable: Previo al desarrollo de la GC se tenía poco conocimiento acerca de la complejidad computacional de muchos cálculos geométricos. La GC puso las bases firmes para el análisis de la complejidad asintótica, y en algunos casos ha hecho posible probar que los algoritmos descubiertos en esta área son óptimos



Fortalezas de la GC

- Énfasis en la corrección y robustez: Antes del desarrollo de la GC muchos de los sistemas de software que se desarrollaron tenían errores que surgían de la confluencia de la naturaleza continua de la geometría y de la naturaleza discreta de la computación. La investigación en GC ha fomentado el cálculo correcto y robusto de primitivas geométricas basado en fundamentos matemáticos sólidos
- Relación con la Geometría Discreta Combinatoria: El estudio de nuevas soluciones a problemas computacionales ha dado origen a muchos nuevos problemas en el campo de la Geometría Discreta Combinatoria. Por ejemplo, el problema de la galería de arte que consiste en encontrar cuántos guardias (en función de n) son necesarios para vigilar una sala representada por un polígono de n lados.



Limitantes de la GC

- Énfasis en geometría discreta: Existen muchas áreas de aplicación en las cuales los objetos son de naturaleza continua, por ejemplo en la física computacional, la planificación de movimientos, el cómputo de la dinámica de fluidos, etc. Debido a su naturaleza discreta, la GC no puede resolver completamente las necesidades de todas estas áreas de aplicación
- Énfasis en objetos planos: Otra limitante es el hecho de que la GC trabaja principalmente con objetos rectos o planos. En gran parte, ésta es una consecuencia del interés de los investigadores en la complejidad geométrica discreta, en oposición a las matemáticas continuas. Por otra parte, probar la correctez y eficiencia de un algoritmo es posible solamente cuando todos los cálculos están bien definidos



Limitantes de la GC

- Énfasis en espacios dimensionales pequeños: Una limitante más es que la GC se ha enfocado principalmente en problemas de 2 dimensiones y relativamente menos a problemas de 3 dimensiones debido a que son más fáciles de visualizar y entender. Pero muchos problemas reales residen en espacios de 3 dimensiones o más

