Algoritmia para problemas difíciles

Elvira Mayordomo

Universidad de Zaragoza

20 de septiembre de 2016

¿Por qué estudiar algoritmia?

- El mejor algoritmo es mucho mejor que el mejor computador (Rico ignorante vs pobre experto en algoritmos)
- Los algoritmos son una poderosa lente para ver la informática:
  - Los grandes cambios en los estándares de Internet se pueden ver como debates sobre deficiencias de un algoritmo de camino más corto y ventajas relativas de otro.
  - Los grandes avances de empresas líder como google son algorítmicos (ranking, maps, self-driving car ...)
- No es exclusivo para informáticos:
  - Muchas definiciones novedosas en biología molecular/genética son algorítmicas: similaridades entre genes o genomas
  - En economía: la viabilidad de subastas combinatorias está ligada a la intratabilidad de problemas relacionados ...
- Es divertido ...
- Las empresas buscan estudiantes que sepan algorítmica

Algoritmia para problemas difíciles

- En *Algoritmia Básica* hemos estudiado las principales técnicas para diseñar algoritmos
- ¿Qué otras técnicas quedan?
- ¿Qué problemas difíciles?

Problemas difíciles

- Algunos ejemplos de problemas tratados en Algoritmia Básica:
  - Coloreado de grafos
  - El viajante de comercio
  - Ciclos hamiltonianos
  - Mochila 0-1
  - ...
- ¿Qué tienen en común estos problemas?
- Los intentos que hicimos (algoritmo voraz, programación dinámica, búsqueda con retroceso, ...) no eran satisfactorios
- O bien los algoritmos **no eran eficientes** o bien encontrar una solución (o una solución óptima) **no estaba garantizado**

Problemas difíciles

- En esta asignatura trataremos sobre todo problemas:
  - NP-completos
  - NP-difíciles
- Son problemas que se consideran intratables porque nadie ha conseguido encontrar algoritmos **eficientes** y que **resuelvan completamente el problema**
- ¿Qué podemos hacer con uno de estos problemas?
- Primero convencernos de que hay pruebas claras para considerarlo **intratable**
- Segundo utilizar técnicas **imaginativas** que nos acerquen a la solución eficiente (aunque no lo consigan completamente)

Técnicas imaginativas para problemas difíciles

- **Algoritmos aproximados:** Nos dan una solución cercana al óptimo con una estimación del error cometido
- **Heurísticas:** Son ideas felices que funcionan bien en algunos casos, sin garantías
- **Algoritmos probabilistas:** Algoritmos que resuelven el problema pero necesitan utilizar el azar
- **Análisis amortizado:** Que el algoritmo sea eficiente cuando lo utilizamos masivamente, no para cada entrada

Ideas felices versus garantías

- ¿Qué diferencia hay entre una heurística y un algoritmo aproximado?
- Es la misma diferencia que entre una cuenta corriente y una cartera de acciones
- Una heurística es como en una cartera de acciones, puedes ganar mucho dinero pero no hay garantías
- Un algoritmo aproximado es como una cuenta corriente, el interés que consigues se conoce a priori
- ¿Qué es mejor? Depende del objetivo, del riesgo ...
- Según otras opiniones, una heurística deja de serlo cuando se convierte en un algoritmo eficiente (conseguimos análisis de eficiencia) o en un algoritmo aproximado (conseguimos análisis de prestaciones)

Contenidos de la asignatura

1. Introducción. Los problemas NP-difíciles.
2. Algoritmos aproximados. Concepto. Diseño de algoritmos. Garantías y límites.
3. Heurísticas. Simulated annealing (templado simulado).
4. Algoritmos genéticos.
5. Algoritmos probabilistas: Las Vegas y Montecarlo. Análisis. Generadores pseudoaleatorios.
6. Análisis amortizado. Estructuras de datos avanzadas.

Bibliografía Básica

- S.S. Skiena. The algorithm Design Manual, Springer, 1997
- J. Hromkovic. Algorithms for Hard Problems, Springer, 2001 *(En la biblioteca)*
- T.H. Cormen, C.E. Leiserson, R.L. Rivest, C. Stein. Introduction to Algorithms (3rd edition), The MIT Press, 2009 *(En la biblioteca)*
- G. Brassard, P. Bratley. Fundamentos de Algoritmia, Prentice Hall, 1997 *(En la biblioteca)*

Bibliografía Complementaria

- J. Kleinberg, E. Tardos. Algorithm Design, Addison-Wesley, 2005 *(En la biblioteca)*
- S. Dasgupta, C. Papadimitriou, U. Vazirani. Algorithms, McGraw-Hill, 2008 *(En la biblioteca)*
- V. V. Vazirani. Approximation Algorithms, Springer, 2001 *En la biblioteca y versión electrónica*

Material adicional

- Información en `http://webdiis.unizar.es/asignaturas/APD`
- Twitter @APDunizar `https://twitter.com/APDunizar`
- Apuntes, ejercicios y guiones de prácticas

Evaluación: opción recomendada

- Prácticas de laboratorio (en parejas) durante el cuatrimestre (2 prácticas): 20 %
- Ejercicios de trabajo individual durante el cuatrimestre (3 ejercicios): 20 %

  Importante: Hay que mandar un mail **antes** *al profesor para pedir la reserva de un ejercicio*

- Prueba escrita intermedia: 20 %
- Examen final (realizar sólo una parte): 40 %

Evaluación exclusivamente por exámenes finales

- Examen práctico: 20 %
- Examen final completo: 80 %

Fecha tentativa de la prueba intermedia

- Viernes 18 de noviembre

Fechas de los exámenes finales

- Jueves 2 de febrero (tarde)
- Viernes 8 de septiembre (tarde)

Nota:

- El jueves 3 de noviembre tiene horario de martes.

Se suspenden las clases los días:

- viernes 7 de cotubre
- martes 15 de noviembre

Las tutorías esos dos días serán electrónicas.

Profesores de la asignatura

- Elvira Mayordomo Cámara
  Despacho 1.06 `elvira@unizar.es`
  `http://webdiis.unizar.es/~elvira`
  Tutorías: Jueves (9-12h), Viernes (9-10h, 12-14h)

Mi investigación: Teoría de la información

Bounded Pushdown dimension vs Lempel Ziv information density. P. Albert, E. Mayordomo, and P. Moser.
[...] We then study the relationships between BPD (Bounded PushDown) compression, and the standard Lempel-Ziv (LZ) compression algorithm, and show that in contrast to the finite-state compressor case, LZ is not universal for bounded pushdown compressors in a strong sense: we construct a sequence that LZ fails to compress significantly, but that is compressed by at least a factor 2 by a BPD compressor.

Mi investigación: Algorítmica

Computing Absolutely Normal Numbers in Nearly Linear Time. Jack H. Lutz, E. Mayordomo.
A real number x is *absolutely normal* if, for every base $b \geq 2$, every two equally long strings of digits appear with equal asymptotic frequency in the base-b expansion of x. This paper presents an explicit algorithm that generates the binary expansion of an absolutely normal number x, with the nth bit of x appearing after $n\mathrm{polylog}(n)$ computation steps. (The fastest previous algorithm takes $n^2\mathrm{polylog}(n)$ steps.) [...]

Mi investigación: Bioinformática

Conservation in Mitochondrial DNA: Parallelized Estimation and Alignment Influence. F.Merino-Casallo, J. Álvarez-Jarreta, and E. Mayordomo. Proceedings of the IEEE International Conference on Bioinformatics and Biomedicine (BIBM 2015), 1434-1440. (2015).
The wide availability of sequenced biological data has challenged the conventional methods and tools used in molecular biology to compute the conservation index. As the size of input datasets increases, the time-cost of current conservation methods is becoming unaffordable.
We propose a new software tool that combines several estimation methods applied to the conservation computation process with parallelization and divide-and-conquer techniques substantially improving its performance without affecting its accuracy. [...]