Análisis de Algoritmos y Estructuras de Datos

Tema 3: Algunos Algoritmos Clásicos y su Análisis

Mª Teresa García Horcajadas José Fidel Argudo Argudo
Antonio García Domínguez Francisco Palomo Lozano

Universidad de Cádiz

Versión 1.0

Índice

Búsqueda secuencial: definición

Problema

- Sea V un conjunto, $x \in V$ el elem. a buscar y $v \in V^n$ el vector
- Se desea la posición de la primera aparición de x en v, si éste aparece, o bien $n+1$ en caso contrario

Nota sobre esta definición

Requiere que $\wedge$ funcione en cortocircuito.

$$
\begin{array}{l}
\textit{búsqueda}(x, v, n) \rightarrow p \\
p \leftarrow 1 \\
\text{mientras } p \leqslant n \wedge x \neq v[p] \\
\qquad p \leftarrow p + 1
\end{array}
$$

Operación crítica

- Comparaciones entre elem. del vector
- Varían según el valor final de p, que depende del contenido de v: hay diferencias entre caso peor y mejor

Búsqueda secuencial: análisis (I)

Mejor caso: $p = 1$

Se encuentra x en la primera comparación:

$$t_{\text{mín}}(n) = 1 \in \Theta(1) \qquad\qquad t(n) \in \Omega(1)$$

Peor caso: $p = n$ o $p = n + 1$

Hay que recorrer todo el vector:

$$t_{\text{máx}}(n) = n \in \Theta(n) \qquad\qquad t(n) \in O(n)$$

Caso promedio

- x está en v con probabilidad α (no está con prob. $\beta = 1 - \alpha$)
- Si x está en v, a falta de más información, todas las posiciones son equiprobables

Búsqueda secuencial: análisis (II)

Si x está en el vector, entonces ocupa una de sus n posiciones; así, bajo hipótesis de equiprobabilidad:

$$\mathcal{P}(p = 1) = \cdots = \mathcal{P}(p = n) = \frac{\alpha}{n}$$

El número de comparaciones que realiza en el promedio es:

$$\begin{aligned}\bar{t}(n) &= \beta n + \sum_{i=1}^{n} \frac{\alpha}{n} i \\ &= (1 - \alpha)n + \frac{\alpha}{n} \sum_{i=1}^{n} i \\ &= (1 - \alpha)n + \frac{\alpha}{n} \left(\frac{n(n+1)}{2} \right) \\ &= \frac{(2 - \alpha)n + \alpha}{2} \in \Theta(n)\end{aligned}$$

ya que $0 \leqslant \alpha \leqslant 1$.

Métodos directos de ordenación

Problema

- Sea $\langle V, \leqslant \rangle$ un orden total y $v \in V^n$ un vector
- Se desea obtener v con sus elementos ordenados según $\leqslant$

Algoritmos a considerar

- Nos centraremos en los que emplean como operación básica la comparación entre elementos: ordenación por comparación
- Entre ellos, consideraremos los métodos directos:
  - Ordenación por intercambio directo
  - Ordenación por selección directa
  - Ordenación por inserción directa

Ordenación por intercambio directo

$ord\text{-}intercambio(v, n) \rightarrow v$
desde $i \leftarrow 1$ hasta $n-1$
    $burbujeo(v, i, n)$

$burbujeo(v, i, j) \rightarrow v$
desde $k \leftarrow j$ hasta $i+1$ con decremento 1
    si $v[k] < v[k-1]$
        $v[k] \leftrightarrow v[k-1]$

Casos peor, mejor y promedio

- El número de comparaciones no depende de v, sólo de n.
- No hay diferencia entre casos mejor, peor y promedio.

Análisis del bucle externo

Varía i desde 1 hasta $n-1$ realizando $n-i$ comparaciones:

$$t(n) = \sum_{i=1}^{n-1}(n-i) = \frac{n(n-1)}{2} \in \Theta(n^2)$$

Ordenación por selección directa

$$
\begin{array}{l}
\textit{ord-selección}(v, n) \rightarrow v \\
\text{desde } i \leftarrow 1 \text{ hasta } n-1 \\
\qquad p \leftarrow \textit{posición-mínimo}(v, i, n) \\
\qquad v[p] \leftrightarrow v[i]
\end{array}
$$

$$
\begin{array}{l}
\textit{posición-mínimo}(v, i, j) \rightarrow p \\
(p, m) \leftarrow (i, v[i]) \\
\text{desde } k \leftarrow i+1 \text{ hasta } j \\
\qquad \text{si } v[k] < m \qquad \leftarrow j-i \text{ veces} \\
\qquad \quad (p, m) \leftarrow (k, v[k])
\end{array}
$$

posición-mínimo es una versión de *mínimo* que devuelve la posición del mínimo en vez de su valor.

El análisis es idéntico al del algoritmo de intercambio directo.

Ordenación por inserción directa: casos mejor y peor

ord-inserción$(v, n) \to v$
desde $i \leftarrow 2$ hasta n
    inserción(v, i)

inserción$(v, j) \to v$
$x \leftarrow v[j]$
mientras $j > 1 \wedge \overbrace{x < v[j-1]}^{\text{entre } 1 \text{ y } j-1 \text{ veces, si } j > 1}$
    $v[j] \leftarrow v[j-1]$
    $j \leftarrow j - 1$
$v[j] \leftarrow x$

Mejor caso: el vector está ordenado

$$t\_ins_{\text{mín}}(j) = 1, \text{ si } j > 1;$$
$$t_{\text{mín}}(n) = (n-1) \cdot t\_ins_{\text{mín}}(i) = n - 1 \in \Theta(n)$$

Peor caso: el vector está ordenado inversamente

$$t\_ins_{\text{máx}}(j) = j - 1, \text{ si } j > 1;$$
$$t_{\text{máx}}(n) = \sum_{i=2}^{n} t\_ins_{\text{máx}}(i) = \sum_{i=2}^{n} (i-1) = \frac{n(n-1)}{2} \in \Theta(n^2)$$

Ordenación por inserción directa: caso promedio

Procedimiento general

1. Supondremos que los n elementos son distintos y sus $n!$ permutaciones equiprobables
2. Por tanto, al insertar $v[i]$, la probabilidad de que quede en cualquiera de las posiciones $[1, i]$ es la misma
3. Podemos utilizar el análisis anterior de la *inserción* ordenada

$$\overline{t\_ins}(i) = \frac{i+1}{2} - \frac{1}{i}$$

$$\bar{t}(n) = \sum_{i=2}^{n} \overline{t\_ins}(i) = \sum_{i=2}^{n} \left(\frac{i+1}{2} - \frac{1}{i} \right) = \frac{1}{2} \sum_{i=2}^{n} (i+1) - \sum_{i=2}^{n} \frac{1}{i}$$

$$= \frac{(n+4)(n-1)}{4} - \sum_{i=2}^{n} \frac{1}{i} \in \Theta(n^2)$$

ya que $\sum_{i=2}^{n} \frac{1}{i} = \sum_{i=1}^{n} \frac{1}{i} - 1 = H_n - 1 \in \Theta(\log n)$

Limitaciones teóricas y algoritmos no directos

Consideraciones formales

- Un vector a ordenar es una permutación del vector ordenado
- El número de inversiones de esa permutación mide su desorden
- Los algoritmos directos eliminan, a lo sumo, una inversión por comparación: mejor rendimiento requiere métodos no directos

Teorema

Todo algoritmo directo ha de realizar, al menos, $n(n-1)/2$ comparaciones en el peor caso y $n(n-1)/4$ en el promedio.

Algoritmos de ordenación no directos

- Ordenación rápida $\rightarrow$ intercambio
- Ordenación por montículo $\rightarrow$ selección
- Ordenación de «Shell» $\rightarrow$ inserción

Referencias

- Brassard, Gilles & Bratley, Paul.
  Fundamentos de Algoritmia.
  Prentice-Hall. 1997.
- Cormen, Thomas H.; Leiserson, Charles E.; Rivest, Ronald L. & Stein, Clifford.
  Introduction to Algorithms.
  MIT Press. 2001. 2ª ed.
- Levitin, Anany V.
  Introduction to the design and analysis of algorithms.
  Addison-Wesley. 2003.
- Manber, Udi.
  Introduction to Algorithms. A Creative Approach.
  Addison-Wesley. 1989.