Elementi e analisi della programmazione dinamica

Due proprietà sono necessarie per applicare la programmazione dinamica:

- Sottostruttura ottima
- Sottoproblemi sovrapposti

Un problema esibisce la proprietà di sottostruttura ottima se una sua soluzione ottima contiene soluzioni ottime ai sottoproblemi

Generalmente per dimostrare che possiede la sottostruttura ottima si suppone che le restrizioni ai sottoproblemi non siano ottime e ne deriva una contraddizione

ESEMPIO

Avendo l'algoritmo ROD-CUTTING
Esprimiamo il problema come:

$$r_n = \max_{\substack{i_1 \dots i_k \\ i_1 + \dots + i_k = n}} p_{i_1} + \dots + p_{i_k} = \max \{ p_n, r_1 + r_{n-1}, r_2 + r_{n-2} \dots \}$$

$$= \max \{ p_i + r_{n-i} \mid 1 \leq i \leq n \}$$

Dimostriamo la proprietà di sottostruttura ottima

$r_n = r_{n-i} + p_i$ se r_n non è ottimo vuol dire che esiste una soluzione di taglio tale che tale che

$$r'_{n-i} > r_{n-i}$$

$$\Rightarrow r'_{n-i} + p_i > r_{n-i} + p_i$$

$$\Rightarrow r'_n > r_n$$

Ne deriva un assurdo poiché r_n è un max per ipotesi

ESEMPIO 2

Proviamo a dimostrarlo per il MATRIX-CHAIN

Sia $1 \le i < j \le n$

Supponiamo che per parentesizzare $A_i \cdot A_{i+1} \cdots \cdot A_j$ lo spezziamo in posizione K avremo:

$$(A_i \cdot \ldots \cdot A_K)(A_{K+1} \cdot \ldots \cdot A_j)$$

Il costo totale è dato da P

$$P_{min} = P_1(A_i \ldots A_K)\, P_2(A_{K+1} \ldots A_j)$$

$$costo(P_{min}) = costo(P_1) + costo(P_2) + costo(P_1 \cdot P_2)$$

Supponiamo che P_1 non sia ottimo

$\exists\; P_1'(A_i \ldots A_K)$ tale che $costo(P_1') < costo(P_1)$

così potrò definire:

$P^* = P_1'(\ldots)\, P_2(\ldots)$

$P = P_1(A_i \dots A_K)\, P_2(A_{K+1} \dots A_J)$

$P^* = costo(P_1') + costo(P_2) + costo(P_1 \cdot P_2)$

$\Rightarrow costo(P^*) < costo(P_{min})$

Da questo ne deriva un assurdo poiché P_{min} era una parentesi~~zza~~zzazione con costo minimo per ipotesi

ANALISI DELLA PROGRAMMAZIONE DINAMICA

Informalmente il tempo computazionale di un algoritmo di programmazione dinamica dipende dal prodotto di questi due fattori:

- # sottoproblemi usati
- # scelte per i sottoproblemi da usare in una soluzione ottima

Es: ROD-CUTTING

Abbiamo:

$\Theta(n)$ sottoproblemi totali
$O(n)$ scelte

$\rightarrow$ ROD-CUTTING $O(n^2)$

Es 2: MATRIX - MULTIPLICATION

$O(n^2)$ sottoproblemi totali
$O(n)$ numero di scelte per sottoproblema

$\rightarrow$ MATRIX - MULTIPLICATION $O(n^3)$

→ MATRIX-MULTIPLICATION $O(M$ $)$