Problemi di ottimizzazione

I istanze

S soluzioni

$\Pi \subseteq I \times S$ problema

$\forall i \in I$ definisco $S(i) = \{ s \in S : (i,s) \in \Pi \}$

↳ "insieme delle soluzioni ammissibili"

$c : S \to \mathbb{R}$ funzione di costo

"soluzione ottima"

obiettivo: determinare, data $i \in I$, $s^* \in S(i)$ tale che

$$c(s^*) = \underset{\text{o max}}{\min} \{ c(s) : s \in S(i) \}$$

Paradigma generale nello sviluppo di un algoritmo di prog. dinamica

Caratteristiche dei problemi:

1) Struttura ricorsiva: la soluzione ottima si ottiene come funzione di soluzioni ottime di sottoistanze.
("Proprietà di sottostruttura ottima")

2) Esistenza di sottoistanze ripetute ("Overlapping subproblems")

→ sennò applico il D&C

3) Spazio dei sottoproblemi piccolo: poche sottoistanze che possono contribuire a creare la soluzione al livello superiore

Ricetta:

1) Caratterizzare la struttura di una soluzione ottima s^* in funzione di soluzioni ottime $s_1^*, \dots, s_k^*$ di sottoistanze di taglia inferiore.

2) Determinare una relazione di ricorrenza del tipo
$c(s^*) = f\left(c(s_1^*), \ldots, c(s_k^*)\right)$.

3) Calcola $c(s^*)$ utilizzando la ricorrenza ma impostando il calcolo in maniera bottom-up (iterativo) oppure memoizzando il codice D&C basato sulla definizione di $c(s^*)$.

4) (opzionale) Mantieni informazioni strutturali aggiuntive che permettano di ricostruire s^*.

Applicazioni della programmazione dinamica:

Problemi su stringhe

Dato un alfabeto finito Σ, una stringa $X = \langle x_1, x_2, \ldots, x_m \rangle$, $x_i \in \Sigma$ per $1 \le i \le m$, è una concatenazione finita di simboli in Σ

lunghezza di X $\quad |X| = m$

Σ^* è l'insieme di tutte le stringhe di lunghezza finita costruibili su Σ

stringa vuota: ε, per convenzione $\in \Sigma^*$

data $X = \langle x_1, x_2, \ldots, x_m \rangle$ il prefisso di X è $X_i = \langle x_1, x_2, \ldots, x_i \rangle$ $1 \le i \le m$, e il suffisso di X è $X^i = \langle x_i, \ldots, x_m \rangle$

$X_0 = X^{m+1} = \varepsilon$

sottostringa di X: $X_{i \ldots j} = \langle x_i, x_{i+1}, \ldots, x_j \rangle \quad 1 \le i \le j \le m$

$X_{i \ldots j} = \varepsilon$ se $i > j$

Quante sono le sottostringhe di una stringa?

$$1 + m + \binom{m}{2} = 1 + m + \frac{m(m-1)}{2} = \Theta(m^2)$$

$1 \to \varepsilon$; $m \to$ stringhe con 1 carattere; $\binom{m}{2} \to$ sottostringhe con ≥ 2 caratteri

$$1 + \sum_{i=1}^{m} \sum_{j=i}^{m} 1 = 1 + \sum_{i=1}^{m} (m-i+1) = 1 + \sum_{k=1}^{m} k =$$

$$= 1 + \frac{m(m+1)}{2} = \Theta(m^2)$$

→ spazio di sottostringhe non è troppo grande

Data $X = \langle x_1, x_2, \dots, x_m \rangle$ e $Z = \langle z_1, z_2, \dots, z_k \rangle \in \Sigma^*$ si dice che Z è <u>sottosequenza</u> di X se esiste una successione crescente di indici $1 \leq i_1 < i_2 < \dots < i_k \leq m$ tale che $z_j = x_{i_j}$ $\quad 1 \leq j \leq k$

↳ "successione che realizza Z in X"

sottostringa è caso particolare di sottosequenza in cui la successione degli indici è particolare: sono indici che crescono sempre di 1

Esempi:

$X = \langle A, B, C, B, B, D \rangle$

$Z_1 = \langle A, B, C \rangle = X_{1 \dots 3}$ sottostringa

$Z_2 = \langle A, C, B \rangle$ no sottostringa; sottosequenza associata agli indici $i_1 = 1 \quad i_2 = 3 \quad i_3 = 4 \text{ o } 5$

$Z_3 = \langle B, B \rangle = X_{4 \dots 5}$ sottostringa, e sottosequenza rispetto agli indici $i_1 = 2$ $i_2 = 5$

Quante sono le sottosequenze di una stringa?

$$\sum_{k=0}^{m} \binom{m}{k} = 2^m$$

$\rightarrow$ spazio delle sottosequenze è enorme

Determinazione della sottosequenza comune di massima lunghezza di due stringhe ("Longest Common Subsequence" LCS):
date X, Y determina Z tale che
1) Z è sottosequenza di X e di Y (cioè sottoseq. comune)
2) Z è la più lunga tra tutte le sottosequenze comuni