Corso di Perfezionamento

Zaino frazionario e Algoritmi golosi

Maria Rita Di Berardini[1]

[1]Dipartimento di Matematica e Informatica
Università di Camerino

17 febbraio 2009

Il problema dello zaino 0-1

- Un ladro entra in un magazzino e trova n oggetti
- L'i-esimo oggetto vale v_i euro e pesa w_i Kg, dove v_i e w_i sono dei numeri interi
- Il ladro vuole realizzare il furto di maggior valore compatibile con il peso W del suo zaino
- Ciascun oggetto può essere preso per intero o lasciato; il ladro non può prendere frazioni di oggetti
- Formalmente:

$$\max \sum_{i=1}^{n} x_i \cdot v_i$$

con vincoli

$$\sum_{i=1}^{n} x_i \cdot w_i \leq W$$

$$x_i \in \{0,1\} \qquad \text{per ogni } i = 1,\ldots,n$$

Il problema dello zaino frazionario

- Un ladro entra in un magazzino e trova n oggetti
- L'i-esimo oggetto vale v_i euro e pesa w_i Kg, dove v_i e w_i sono dei numeri interi
- Il ladro vuole realizzare il furto di maggior valore compatibile con il peso W del suo zaino
- Il ladro può prendere frazioni di oggetti
- Formalmente:

$$\max \sum_{i=1}^{n} x_i \cdot v_i$$

con vincoli

$$\sum_{i=1}^{n} x_i \cdot w_i \leq W$$

$$0 \leq x_i \leq 1 \qquad \text{per ogni } i = 1, \ldots, n$$

Il problema dello zaino: due versioni

- Abbiamo visto due varianti del problema dello zaino (zaino 0-1, zaino frazionario)
- I due problemi sono molto simili (entrambi hanno una sottostruttura ottima)
- Lo zaino 0-1 viene risolto mediante un algoritmo di PD
- Lo zaino frazionario viene risolto mediate un **algoritmo goloso**

Elementi della strategia golosa

Un algoritmo goloso costruisce una soluzione ottima di un dato problema effettuando una serie di scelte

Ogni decisione viene presa tenendo conto della scelta che, al momento, sembra la migliore

Questa strategia euristica non sempre produce una soluzione ottima

Ma quando lo fa, ci consente di scrivere algoritmi molto efficienti

Strategia golosa per il problema dello zaino frazionario

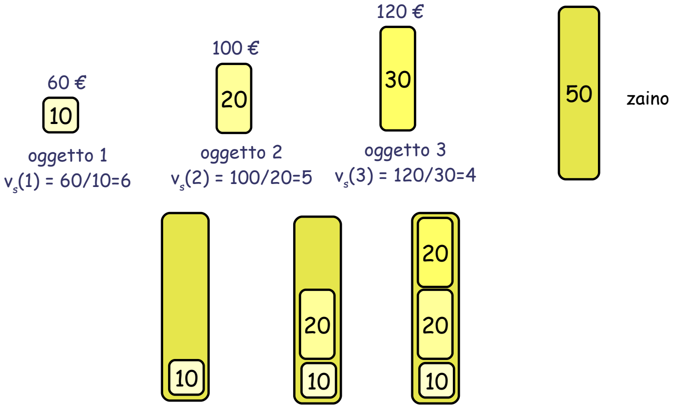
1. calcoliamo il valore per unità peso, o **valore specifico** (pari a v_i/w_i) di ciascun oggetto
2. ordiniamo gli oggetti in maniera decrescente rispetto al loro peso specifico
3. inizialmente, il ladro prende la maggiore quantità possibile del primo oggetto, quello con peso specifico maggiore
4. esaurita la scorta del primo oggetto e se nello zaino c'è ancora spazio, inserisce la maggior quantità possibile del secondo e così via

Strategia golosa per il problema dello zaino frazionario

FractionalKnapsack$(\mathbf{w}, \mathbf{v}, W)$

$n \leftarrow length[\mathbf{v}]$
$\mathbf{x}[1, \ldots, n]$ **of** $[0, 1] \leftarrow 0$
$i \leftarrow 1$
$w \leftarrow W$
while $i \leq n$ **and** $w > 0$
  do
    if $w_i \leq w$
      then $\mathbf{x}[i] \leftarrow 1$
      else $\mathbf{x}[i] \leftarrow w/\mathbf{w}[i]$
    $w \leftarrow w - (\mathbf{x}[i] \cdot \mathbf{w}[i])$
    $i \leftarrow i + 1$
return x

Un algoritmo goloso per lo zaino frazionario



Otteniamo una soluzione composta dali oggetti 1 e 2 (interi), più una parte (2/3) dell'oggetto 3 il cui peso è pari a 50 Kg e con valore

60 + 100 + (2/3) 120 = 60 + 100 + 80 = 240 € (soluzione ottima)

La proprietà della scelta golosa

- Proprietà della scelta golosa: esiste **sempre** una soluzione ottima che contiene la scelta golosa

La proprietà della scelta golosa

- Proprietà della scelta golosa: esiste **sempre** una soluzione ottima che contiene la scelta golosa
- È sempre possibile costruire una soluzione globalmente ottima mediante una serie di scelte localmente ottime
- Ad ogni passo facciamo la scelta che sembra la migliore per il sottoproblema corrente, senza considerare altre soluzioni
- La scelta fatta da un algoritmo goloso può dipendere dalle scelte fatte in precedenza (che determinano il sottoproblema corrente) ma non dalle scelte future
- A differenza della PD (che procede in maniera bottom-up), un alg. goloso procede in maniera top-down: ogni scelta golosa riduce la dimensione del sottoproblema da risolvere

Zaino frazionario e proprietà della scelta golosa

Theorem

È sempre possibile costruire una soluzione dello zaino frazionario che contiene la maggiore quantità possibile del primo oggetto.

Zaino frazionario e proprietà della scelta golosa

Theorem

È sempre possibile costruire una soluzione dello zaino frazionario che contiene la maggiore quantità possibile del primo oggetto.

Sia $X = \langle x_1, \ldots, x_n \rangle$ una soluzione (ottima) per una data istanza dello zaino frazionario

Ogni x_i è tale che $0 \leq x_i \leq 1$ e rappresenta la frazione dell'oggetto i selezionata (se $x_i = 0$, l'i-esimo non contenuto nello zaino)

- il peso di X è $W(X) = \sum_{i=1}^{n} x_i \cdot w_i \leq W$
- il valore di X è $V(X) = \sum_{i=1}^{n} x_i \cdot v_i$

Zaino frazionario e proprietà della scelta golosa

Distinguiamo due casi

1. $x_1 = q$ (dove q rappresenta la maggiore quantità possibile del primo oggetto)
2. $x_1 < q$. In questo caso sia $X' = \langle x'_1, ..., x'_n \rangle$ tale che:

$$x'_i = x_i \qquad \text{per } i = 3, \ldots, n$$

$$x'_1 = q$$

$$x'_2 = x_2 - (q - x_1)\frac{v_1}{v_2}$$

Dimostriamo che X' è ottima, ossia:

1. è ammissibile, i.e. $W(X') \leq W$
2. $V(X') = V(X)$

Valore ottimo

Il valore di x_2' è stato scelto in maniera tale che

$$x_1' \cdot v_1 + x_2' \cdot v_2 = x_1 \cdot v_1 + x_2 \cdot v_2$$

ossia

$$x_2' \cdot v_2 = x_2 \cdot v_2 - (x_1' - x_1) v_1$$

e

$$x_2' = x_2 - (q - x_1) \frac{v_1}{v_2}$$

Allora

$$V(X') = \sum_{i=0}^{n} x_i' \cdot v_i = x_1' \cdot v_1 + x_2' \cdot v_2 + \sum_{i=3}^{n} x_i' \cdot v_i =$$

$$x_1 \cdot v_1 + x_2 \cdot v_2 + \sum_{i=3}^{n} x_i \cdot v_i = V(X)$$

Ammissibilità

$$x_1' \cdot w_1 + x_2' \cdot w_2 = qw_1 + \left(x_2 - (q - x_1)\frac{v_1}{v_2}\right) w_2 =$$

$$qw_1 + x_2 w_2 - (q - x_1) \cdot v_1 \cdot \frac{w_2}{v_2}$$

Inoltre: $\frac{v_2}{w_2} \leq \frac{v_1}{w_1}$ implica $\frac{w_2}{v_2} \geq \frac{w_1}{v_1}$ e quindi

$$qw_1 + x_2 w_2 - (q - x_1) \cdot v_1 \cdot \frac{w_2}{v_2}$$

$$\leq qw_1 + x_2 w_2 - (q - x_1) \cdot v_1 \cdot \frac{w_1}{v_1}$$

$$= qw_1 + x_2 w_2 - (q - x_1) \cdot w_1$$

$$= x_1 \cdot w_1 + x_2 \cdot w_2$$

Ammissibilità

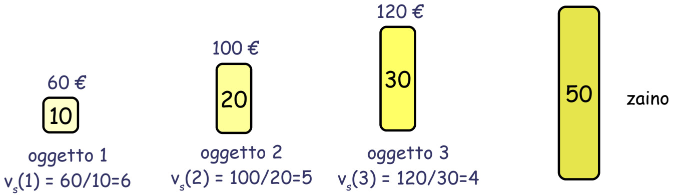
Se $x'_1 \cdot w_1 + x'_2 \cdot w_2 \leq x_1 \cdot w_1 + x_2 \cdot w_2$ allora

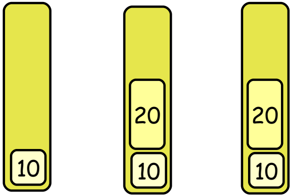
$$W(X') = \sum_{i=0}^{n} x'_i \cdot w_i = x'_1 \cdot w_1 + x'_2 \cdot w_2 + \sum_{i=3}^{n} x'_i \cdot v_i \leq$$

$$x_1 \cdot w_1 + x_2 \cdot w_2 + \sum_{i=3}^{n} x_i \cdot w_i = W(X) \leq W$$

E quindi X' è ammissibile

Zaino 0-1 e scelta golosa





La capacità residua dello zaino non ci consente di inserire il terzo oggetto

otteniamo una soluzione il cui valore è **60+100=160**

La **soluzione ottima** include gli oggetti 2 e 3 ed ha un peso di 50Kg ed un valore pari a **100+120= 220 €**

PD versus Algoritmi Golosi

- PD e algorimi golosi vengono usati per risolvere problemi di ottimizzazione per cui vale la proprietà della sottostruttura ottima.
- Negli algorimi golosi, le scelte ad ogni passo dipendono da un criterio esterno (appetibilità) ogni scelta determina un sottoproblema
- Nella PD le scelte dipendono dalla soluzione dei sottoproblemi
- Gli algorimi golosi agiscono in modo top-down: riducono progressivamente un problema a sottoproblemi di dimensioni decrescenti.
- I metodi basati sulla PD procedono bottom-up risolvendo per primi i problemi pi piccoli.
- I metodi golosi sono molto più efficienti di quelli basati sulla programmazione dinamica che devono provare tutte le alternative per fare una scelta ottima. La PD ha unapplicabilità maggiore rispetto ai metodi greedy.