



SAPIENZA
UNIVERSITÀ DI ROMA

Ottimizzazione combinatoria

Esercizi 5 — Algoritmi di approssimazione

Esercizi svolti sui capitoli del volume 5, algoritmi di approssimazione: ogni esercizio è seguito dalla sua soluzione completa.

Fabio Furini

fabiofurini.github.io

Dipartimento di Ingegneria informatica, automatica e gestionale — Sapienza Università di Roma

Versione online e materiale scaricabile:
fabiofurini.github.io/ottimizzazione-combinatoria/
github.com/fabiofurini/ottimizzazione-combinatoria

Anno accademico 2026–2027

Indice

34	Principi degli algoritmi di approssimazione	3
	Problemi di zaino	9
35	Algoritmi di approssimazione per lo zaino	10
35.1	Algoritmo greedy	10
35.2	Algoritmo greedy esteso	15
35.3	L'euristica H^2	20
35.4	L'euristica H^ℓ	24
35.5	Lo schema FPTAS	30
	Problemi di packing	35
36	Algoritmi di approssimazione per il bin packing	36
36.1	Next Fit	36
36.2	First Fit	42
36.3	Best Fit	47
36.4	Worst Fit	52
	Problemi di colorazione	58
37	Colorazione dei vertici: limiti dell'approssimazione	59
	Problemi di clique	65
38	Clique massima: limiti dell'approssimazione	66
	Problemi di routing	70
39	TSP: il caso generale e il caso metrico	71

Problemi di scheduling	76
40 Approssimazione per $P\ C_{\max}$	77

Principi degli algoritmi di approssimazione

Definizioni usate in questi esercizi

Le misure del capitolo [Principi degli algoritmi di approssimazione](#), per un algoritmo $\mathcal{A}$ di valore $\mathcal{A}(I)$ su un'istanza I di valore ottimo $z(I)$:

$$\text{rapporto di prestazione: } \frac{\mathcal{A}(I)}{z(I)} \text{ (minimo), } \frac{z(I)}{\mathcal{A}(I)} \text{ (massimo);} \quad (\text{A1})$$

$$\text{errore additivo: } \mathcal{A}(I) - z(I) \text{ (minimo), } z(I) - \mathcal{A}(I) \text{ (massimo);} \quad (\text{A2})$$

$$\text{errore relativo: } \frac{\mathcal{A}(I) - z(I)}{z(I)} \text{ (minimo), } \frac{z(I) - \mathcal{A}(I)}{z(I)} \text{ (massimo).} \quad (\text{A3})$$

Il rapporto di prestazione nel caso peggiore è l'estremo superiore del rapporto di prestazione (A1) su tutte le istanze:

$$r(\mathcal{A}) = \sup_I \frac{\mathcal{A}(I)}{z(I)} \text{ (minimo), } r(\mathcal{A}) = \sup_I \frac{z(I)}{\mathcal{A}(I)} \text{ (massimo).} \quad (\text{A4})$$

Esercizio 34.1 — Rapporti di prestazione ed errori (1)

Un algoritmo euristico $\mathcal{A}$ per un problema di **minimo** e un algoritmo euristico $\mathcal{B}$ per un problema di **massimo** danno i seguenti valori su alcune istanze:

$\mathcal{A}$ (min)	I_1	I_2	I_3	I_4	I_5	$\mathcal{B}$ (max)	I_1	I_2	I_3
$\mathcal{A}(I)$	12	9	15	20	7	$\mathcal{B}(I)$	30	25	16
$z(I)$	10	9	10	16	6	$z(I)$	36	25	24

Domande

1. Calcolare il rapporto di prestazione, l'errore additivo e l'errore relativo di $\mathcal{A}$ sulle sue cinque istanze.
2. Fare lo stesso per $\mathcal{B}$ sulle sue tre istanze.
3. Usando solo questi dati, quale delle costanti $\alpha = \frac{7}{5}$ e $\alpha = \frac{3}{2}$ può essere una garanzia di prestazione relativa di $\mathcal{A}$? Che cosa servirebbe per dimostrare che $\mathcal{A}$ è un algoritmo di $\frac{3}{2}$ -approssimazione?

4. Supporre che sia dimostrato che $\mathcal{A}$ è un algoritmo di $\frac{3}{2}$ -approssimazione. La garanzia è tight? Dare un lower bound su $r(\mathcal{A})$ e su $r(\mathcal{B})$.

Soluzione

Usiamo le definizioni (A1)–(A4) del box all’inizio del capitolo.

1. Per un problema di minimo il rapporto è $\frac{\mathcal{A}(I)}{z(I)}$:

istanza	$\mathcal{A}(I)$	$z(I)$	rapporto	errore additivo	errore relativo
I_1	12	10	$\frac{6}{5}$	2	$\frac{1}{5}$
I_2	9	9	1	0	0
I_3	15	10	$\frac{3}{2}$	5	$\frac{1}{2}$
I_4	20	16	$\frac{5}{4}$	4	$\frac{1}{4}$
I_5	7	6	$\frac{7}{6}$	1	$\frac{1}{6}$

2. Per un problema di massimo il rapporto è $\frac{z(I)}{\mathcal{B}(I)}$:

istanza	$\mathcal{B}(I)$	$z(I)$	rapporto	errore additivo	errore relativo
I_1	30	36	$\frac{6}{5}$	6	$\frac{1}{6}$
I_2	25	25	1	0	0
I_3	16	24	$\frac{3}{2}$	8	$\frac{1}{3}$

Tutti i rapporti sono almeno 1, come deve essere per soluzioni ammissibili.

3. Su I_3 il rapporto è $\frac{3}{2} > \frac{7}{5}$: $\alpha = \frac{7}{5}$ non è una garanzia di prestazione relativa. I dati non escludono $\alpha = \frac{3}{2}$, perché ogni rapporto è al più $\frac{3}{2}$. Una dimostrazione deve mostrare $\mathcal{A}(I) \leq \frac{3}{2} z(I)$ per ogni istanza I : un numero finito di esempi può solo smentire una garanzia.
4. Sì: l’istanza I_3 raggiunge il rapporto $\frac{3}{2}$, quindi nessuna costante minore è una garanzia, e $r(\mathcal{A}) = \frac{3}{2}$. Senza una dimostrazione, i dati danno solo lower bound: $r(\mathcal{A}) \geq \frac{3}{2}$ e $r(\mathcal{B}) \geq \frac{3}{2}$ (istanza I_3 di $\mathcal{B}$).

Esercizio 34.2 — Una garanzia tight per un algoritmo semplice (1)

Per $P||C_{\max}$ consideriamo l’algoritmo ONE, che assegna tutti i lavori alla macchina 1, così che $\text{ONE}(I) = \sum_{j \in \mathcal{J}} t_j$. Ogni istanza soddisfa $z(I) \geq \frac{\sum_j t_j}{m}$, perché la somma degli m carichi è $\sum_j t_j$. Consideriamo le istanze

$$I_1: m = 2 \text{ e } t = (7, 5, 4, 3, 3);$$

$$I_2: m = 3 \text{ e } t = (9, 2, 2, 2, 1);$$

$$I_3: m = 3 \text{ e } t = (5, 5, 4, 4, 3, 3, 3);$$

Domande

1. Calcolare $\text{ONE}(I)$, $z(I)$ (per enumerazione) e il rapporto di prestazione sulle tre istanze.
2. Dimostrare che ONE è un algoritmo di m -approssimazione per ogni numero fissato m di macchine.
3. Costruire, per ogni $m \geq 2$, un'istanza su cui il rapporto è esattamente m , e verificarla con $m = 4$. Che cosa si può concludere?
4. ONE ha una garanzia di prestazione additiva finita? Usare l'istanza I_1 con tutti i tempi di lavorazione moltiplicati per 10.

Soluzione

Usiamo le definizioni (A1)–(A4) del box all'inizio del capitolo.

	istanza	m	$\text{ONE}(I)$	$z(I)$	rapporto	garanzia m
1.	I_1	2	22	11	2	2
	I_2	3	16	9	$\frac{16}{9}$	3
	I_3	3	27	9	3	3

Il rapporto non supera mai m , ed è uguale a m su I_1 e I_3 .

2. Per ogni istanza, $z(I) \geq \frac{\sum_j t_j}{m}$ dà

$$\frac{\text{ONE}(I)}{z(I)} = \frac{\sum_j t_j}{z(I)} \leq \frac{\sum_j t_j}{\frac{\sum_j t_j}{m}} = m.$$

3. Prendiamo $n = 2m$ lavori di tempo 1. Due lavori per macchina danno $z(I) = 2$, mentre $\text{ONE}(I) = 2m$: il rapporto è m . Con $m = 4$: $\text{ONE}(I) = 8$, $z(I) = 2$ (enumerazione) e il rapporto è 4. La garanzia m è tight e $r(\text{ONE}) = m$.
4. Moltiplicando i tempi di I_1 per 10 si ottiene $t = (70, 50, 40, 30, 30)$, con $\text{ONE}(I) = 220$ e $z(I) = 110$: l'errore additivo passa da 11 a 110. Moltiplicando per un qualsiasi K l'errore si moltiplica per K , quindi nessuna costante β lo limita: $e(\text{ONE}) = \infty$, mentre il rapporto resta 2.

Esercizio 34.3 — Un algoritmo senza garanzia (1)

Per il BPP consideriamo l'algoritmo SEP, che mette ogni oggetto in un contenitore separato, così che $\text{SEP}(I) = n$. Consideriamo l'istanza I_0 con $c = 10$ e pesi $(6, 5, 4, 3, 2)$ e, per $k \in \mathbb{Z}_{\geq 2}$, le istanze I_k con $c = k$ e $2k$ oggetti di peso 1.

Domande

1. Calcolare $\text{SEP}(I_0)$, $z(I_0)$, il rapporto di prestazione e l'errore additivo su I_0 .
2. Calcolare il rapporto di prestazione su I_k per $k = 3, 5, 10$ (usare l'enumerazione per $k = 3$) e la sua espressione per ogni k .

3. SEP è un algoritmo di α -approssimazione per qualche costante α ? Dare $r(\text{SEP})$ ed $e(\text{SEP})$.
4. Usando $z(I) \geq 2$, mostrare che $\frac{\text{SEP}(I)}{z(I)} \leq \frac{n}{2}$. Perché $\frac{n}{2}$ non è una garanzia di prestazione relativa?

Soluzione

Usiamo le definizioni (A1)–(A4) del box all’inizio del capitolo.

1. $\text{SEP}(I_0) = 5$ e $z(I_0) = 2$ (schemi $\{1, 3\}$ e $\{2, 4, 5\}$). Il rapporto è $\frac{5}{2}$ e l’errore additivo è $5 - 2 = 3$.
2. Due contenitori di capacità k contengono i $2k$ oggetti unitari, e un contenitore non basta, quindi $z(I_k) = 2$:

k	c	$\text{SEP}(I_k) = 2k$	$z(I_k)$	rapporto
3	3	6	2	3
5	5	10	2	5
10	10	20	2	10

In generale il rapporto è $\frac{2k}{2} = k$.

3. No: per ogni costante α , l’istanza I_k con $k > \alpha$ ha rapporto $k > \alpha$. Quindi $r(\text{SEP}) = \infty$. Anche l’errore additivo su I_k , pari a $2k - 2$, è illimitato: $e(\text{SEP}) = \infty$.
4. Se $z(I) \geq 2$, allora $\frac{\text{SEP}(I)}{z(I)} = \frac{n}{z(I)} \leq \frac{n}{2}$. Il bound dipende dall’istanza attraverso n : una garanzia di prestazione relativa deve essere un’unica costante α valida per tutte le istanze.

Esercizio 34.4 — Rapporto illimitato di un algoritmo greedy (1)

Per il KP consideriamo l’algoritmo GR: ordinare gli oggetti per efficienza $\frac{p_j}{w_j}$ non crescente (pareggi: indice minimo) e inserire ogni oggetto, in questo ordine, se entra nella capacità residua. Consideriamo l’istanza I_0 con $c = 9$, $p = (10, 7, 6, 4)$, $w = (5, 4, 3, 3)$ e, per $M \in \mathbb{Z}_{\geq 3}$, le istanze I_M con $c = M$ e due oggetti: $p_1 = 2$, $w_1 = 1$ e $p_2 = M$, $w_2 = M$.

Domande

1. Applicare GR a I_0 , calcolare $z(I_0)$ per enumerazione, il rapporto di prestazione e l’errore relativo.
2. Applicare GR a I_M e calcolare $z(I_M)$, il rapporto di prestazione e l’errore additivo per $M = 3, 10, 100$.
3. GR è un algoritmo di α -approssimazione per qualche costante α ? È un algoritmo di approssimazione β -additivo?
4. Quale proprietà delle istanze I_M fa fallire GR?

Soluzione

Usiamo le definizioni (A1)–(A4) del box all’inizio del capitolo.

1. Le efficienze sono $2, \frac{7}{4}, 2, \frac{4}{3}$: l’ordine è $(1, 3, 2, 4)$. GR inserisce gli oggetti 1, 3 e ottiene $\text{GR}(I_0) = 16$. L’enumerazione dà $z(I_0) = 17$ (oggetti $\{1, 2\}$). Il rapporto è $\frac{17}{16}$ e l’errore relativo è $\frac{17-16}{17} = \frac{1}{17}$.
2. L’oggetto 1 ha efficienza 2 e l’oggetto 2 ha efficienza 1: GR inserisce l’oggetto 1, e poi l’oggetto 2 non entra nella capacità residua $M - 1$. Quindi $\text{GR}(I_M) = 2$, mentre il solo oggetto 2 dà $z(I_M) = M$:

M	$\text{GR}(I_M)$	$z(I_M)$	rapporto	errore additivo
3	2	3	$\frac{3}{2}$	1
10	2	10	5	8
100	2	100	50	98

3. Il rapporto $\frac{M}{2}$ cresce senza limite, quindi nessuna costante α va bene e $r(\text{GR}) = \infty$. Anche l’errore additivo $M - 2$ è illimitato: $e(\text{GR}) = \infty$.
4. L’oggetto efficiente è piccolo e blocca l’oggetto grande, il cui profitto è quasi tutto il valore ottimo: l’ordine per efficienza ignora quanto profitto porta un oggetto in termini assoluti.

Esercizio 34.5 — Schemi di approssimazione (1)

Uno schema di approssimazione $\{\mathcal{A}_\varepsilon\}_{\varepsilon \in (0,1)}$ viene eseguito su alcune istanze. La tabella riporta il tipo di problema, il parametro di accuratezza ε , il valore ottimo e il valore restituito:

caso	problema	ε	$z(I)$	$\mathcal{A}_\varepsilon(I)$
(1)	minimo	$\frac{1}{10}$	100	108
(2)	minimo	$\frac{1}{10}$	60	67
(3)	massimo	$\frac{1}{5}$	50	41
(4)	massimo	$\frac{1}{5}$	50	38

Domande

1. Per ogni caso, calcolare l’errore relativo e stabilire se il valore è compatibile con la definizione di schema di approssimazione.
2. Tradurre $\varepsilon = \frac{1}{4}$ in una garanzia di prestazione relativa, per un problema di minimo e per un problema di massimo. Qual è l’accuratezza?
3. Uno schema per un problema di massimo deve essere un algoritmo di $\frac{5}{4}$ -approssimazione. Quale valore di ε serve?
4. Classificare i seguenti tempi di esecuzione di uno schema di approssimazione su istanze di dimensione n : $O(n^{1/\varepsilon})$, $O(n^3/\varepsilon)$, $O(n2^{1/\varepsilon})$, $O(n^2/\varepsilon^2)$, $O(2^n/\varepsilon)$.

Soluzione

1. (1) $\frac{108-100}{100} = \frac{2}{25} \leq \frac{1}{10}$: compatibile.
 (2) $\frac{67-60}{60} = \frac{7}{60} > \frac{1}{10}$: non compatibile, lo schema sbaglia su questa istanza.
 (3) $\frac{50-41}{50} = \frac{9}{50} \leq \frac{1}{5}$: compatibile.
 (4) $\frac{50-38}{50} = \frac{6}{25} > \frac{1}{5}$: non compatibile, lo schema sbaglia su questa istanza.
2. Per un problema di minimo $\mathcal{A}_\varepsilon(I) \leq (1 + \varepsilon) z(I)$, quindi la garanzia è $1 + \frac{1}{4} = \frac{5}{4}$.
 Per un problema di massimo $\mathcal{A}_\varepsilon(I) \geq (1 - \varepsilon) z(I)$, quindi la garanzia è $\frac{1}{1-\varepsilon} = \frac{4}{3}$.
 L'accuratezza è $\frac{1}{\varepsilon} = 4$.
3. Serve $\frac{1}{1-\varepsilon} \leq \frac{5}{4}$, cioè $1 - \varepsilon \geq \frac{4}{5}$ e $\varepsilon \leq \frac{1}{5}$: va bene qualunque $\varepsilon \leq \frac{1}{5}$, e $\varepsilon = \frac{1}{5}$ è il più grande.
4.
 - $O(n^{1/\varepsilon})$: PTAS, non FPTAS.
 - $O(n^3/\varepsilon)$: FPTAS.
 - $O(n^{2^{1/\varepsilon}})$: PTAS, non FPTAS.
 - $O(n^2/\varepsilon^2)$: FPTAS.
 - $O(2^n/\varepsilon)$: non è un PTAS.

Un PTAS richiede un polinomio in n per ogni ε fissato; un FPTAS richiede un polinomio sia in n sia in $\frac{1}{\varepsilon}$. I termini $n^{1/\varepsilon}$ e $2^{1/\varepsilon}$ non sono polinomiali in $\frac{1}{\varepsilon}$, e 2^n non è polinomiale in n .

Problemi di zaino



Algoritmi di approssimazione per lo zaino

35.1 Algoritmo greedy

Esercizio 35.1 — Greedy 1

Consideriamo uno zaino di capacità $c = 15$ e un insieme $\mathcal{I} = \{1, 2, 3, 4, 5\}$ di 5 oggetti ($n = 5$), con i seguenti profitti e pesi:

	$i = 1$	$i = 2$	$i = 3$	$i = 4$	$i = 5$
p_i	12	6	6	8	6
w_i	3	2	3	9	7

Domande

1. Verificare che gli oggetti siano ordinati per efficienza non crescente, e determinare l'oggetto critico s e la capacità residua rc .
2. Applicare l'algoritmo greedy (GR) a questa istanza. Mostrare l'esecuzione passo per passo, indicando a ogni iterazione se l'oggetto viene selezionato e la capacità residua che ne risulta. Indicare la soluzione S restituita da GR e $GR(I)$.
3. Calcolare l'upper bound $U = \sum_{i=1}^{s-1} p_i + p_s \frac{rc}{w_s}$ su $z(I)$ dato dal rilassamento lineare, e dedurre un upper bound sul rapporto di prestazione $\frac{z(I)}{GR(I)}$.
4. Stabilire se GR trova una soluzione ottima su questa istanza, mostrando che $GR(I)$ raggiunge l'upper bound $\lfloor U \rfloor$ oppure esibendo una soluzione ammissibile di profitto maggiore.

Soluzione

1. Le efficienze degli oggetti sono

$$\frac{p_1}{w_1} = \frac{12}{3} = 4, \quad \frac{p_2}{w_2} = \frac{6}{2} = 3, \quad \frac{p_3}{w_3} = \frac{6}{3} = 2, \quad \frac{p_4}{w_4} = \frac{8}{9} \approx 0.89, \\ \frac{p_5}{w_5} = \frac{6}{7} \approx 0.86,$$

quindi gli oggetti sono ordinati per efficienza non crescente. Poiché

$$\sum_{i=1}^3 w_i = 3 + 2 + 3 = 8 \leq 15 \quad \text{e} \quad \sum_{i=1}^4 w_i = 3 + 2 + 3 + 9 = 17 > 15,$$

l'oggetto critico è $s = 4$ e la capacità residua è $rc = 15 - 8 = 7$.

2. All'inizio $rc := c = 15$. L'esecuzione di GR procede come segue:

- (a) Oggetto 1 (peso $w_1 = 3$): $w_1 = 3 \leq rc = 15$, quindi l'oggetto viene selezionato e $rc \leftarrow 15 - 3 = 12$.
- (b) Oggetto 2 (peso $w_2 = 2$): $w_2 = 2 \leq rc = 12$, quindi l'oggetto viene selezionato e $rc \leftarrow 12 - 2 = 10$.
- (c) Oggetto 3 (peso $w_3 = 3$): $w_3 = 3 \leq rc = 10$, quindi l'oggetto viene selezionato e $rc \leftarrow 10 - 3 = 7$.
- (d) Oggetto 4 (peso $w_4 = 9$): $w_4 = 9 > rc = 7$, quindi l'oggetto non viene selezionato.
- (e) Oggetto 5 (peso $w_5 = 7$): $w_5 = 7 \leq rc = 7$, quindi l'oggetto viene selezionato e $rc \leftarrow 7 - 7 = 0$.

Quindi GR restituisce la soluzione $S = \{1, 2, 3, 5\}$ con

$$GR(I) = 12 + 6 + 6 + 6 = 30.$$

GR seleziona tutti gli oggetti che precedono l'oggetto critico, non seleziona l'oggetto critico 4, e seleziona anche l'oggetto 5, che lo segue.

3. Il rilassamento lineare dà

$$z(I) \leq U = 12 + 6 + 6 + 8 \cdot \frac{7}{9} = \frac{272}{9} \approx 30.22,$$

e poiché $z(I)$ è intero, $z(I) \leq \lfloor U \rfloor = 30$. Quindi

$$\frac{z(I)}{GR(I)} \leq \frac{30}{30} = 1.$$

4. Poiché $GR(I) = 30 = \lfloor U \rfloor$, nessuna soluzione ammissibile ha profitto maggiore: GR trova una soluzione **ottima** su questa istanza, e $z(I) = 30$.

Esercizio 35.2 — Greedy 2

Consideriamo uno zaino di capacità $c = 13$ e un insieme $\mathcal{I} = \{1, 2, 3, 4, 5, 6\}$ di 6 oggetti ($n = 6$), con i seguenti profitti e pesi:

	$i = 1$	$i = 2$	$i = 3$	$i = 4$	$i = 5$	$i = 6$
p_i	10	6	3	7	11	2
w_i	1	1	1	4	8	3

Domande

1. Verificare che gli oggetti siano ordinati per efficienza non crescente, e determinare l'oggetto critico s e la capacità residua rc .
2. Applicare l'algoritmo greedy (GR) a questa istanza. Mostrare l'esecuzione passo per passo, indicando a ogni iterazione se l'oggetto viene selezionato e la capacità residua che ne risulta. Indicare la soluzione S restituita da GR e $GR(I)$.
3. Calcolare l'upper bound $U = \sum_{i=1}^{s-1} p_i + p_s \frac{rc}{w_s}$ su $z(I)$ dato dal rilassamento lineare, e dedurre un upper bound sul rapporto di prestazione $\frac{z(I)}{GR(I)}$.
4. Stabilire se GR trova una soluzione ottima su questa istanza, mostrando che $GR(I)$ raggiunge l'upper bound $\lfloor U \rfloor$ oppure esibendo una soluzione ammissibile di profitto maggiore.

Soluzione

1. Le efficienze degli oggetti sono

$$\frac{p_1}{w_1} = \frac{10}{1} = 10, \quad \frac{p_2}{w_2} = \frac{6}{1} = 6, \quad \frac{p_3}{w_3} = \frac{3}{1} = 3, \quad \frac{p_4}{w_4} = \frac{7}{4} = 1.75, \\ \frac{p_5}{w_5} = \frac{11}{8} \approx 1.38, \quad \frac{p_6}{w_6} = \frac{2}{3} \approx 0.67,$$

quindi gli oggetti sono ordinati per efficienza non crescente. Poiché

$$\sum_{i=1}^4 w_i = 1 + 1 + 1 + 4 = 7 \leq 13 \quad \text{e} \quad \sum_{i=1}^5 w_i = 1 + 1 + 1 + 4 + 8 = 15 > 13,$$

l'oggetto critico è $s = 5$ e la capacità residua è $rc = 13 - 7 = 6$.

2. All'inizio $rc := c = 13$. L'esecuzione di GR procede come segue:

- (a) Oggetto 1 (peso $w_1 = 1$): $w_1 = 1 \leq rc = 13$, quindi l'oggetto viene selezionato e $rc \leftarrow 13 - 1 = 12$.
- (b) Oggetto 2 (peso $w_2 = 1$): $w_2 = 1 \leq rc = 12$, quindi l'oggetto viene selezionato e $rc \leftarrow 12 - 1 = 11$.

- (c) Oggetto 3 (peso $w_3 = 1$): $w_3 = 1 \leq rc = 11$, quindi l'oggetto viene selezionato e $rc \leftarrow 11 - 1 = 10$.
- (d) Oggetto 4 (peso $w_4 = 4$): $w_4 = 4 \leq rc = 10$, quindi l'oggetto viene selezionato e $rc \leftarrow 10 - 4 = 6$.
- (e) Oggetto 5 (peso $w_5 = 8$): $w_5 = 8 > rc = 6$, quindi l'oggetto non viene selezionato.
- (f) Oggetto 6 (peso $w_6 = 3$): $w_6 = 3 \leq rc = 6$, quindi l'oggetto viene selezionato e $rc \leftarrow 6 - 3 = 3$.

Quindi GR restituisce la soluzione $S = \{1, 2, 3, 4, 6\}$ con

$$GR(I) = 10 + 6 + 3 + 7 + 2 = 28.$$

GR seleziona tutti gli oggetti che precedono l'oggetto critico, non seleziona l'oggetto critico 5, e seleziona anche l'oggetto 6, che lo segue.

3. Il rilassamento lineare dà

$$z(I) \leq U = 10 + 6 + 3 + 7 + 11 \cdot \frac{6}{8} = \frac{137}{4} \approx 34.25,$$

e poiché $z(I)$ è intero, $z(I) \leq \lfloor U \rfloor = 34$. Quindi

$$\frac{z(I)}{GR(I)} \leq \frac{34}{28} = \frac{17}{14} \approx 1.21.$$

4. La soluzione $\tilde{S} = \{1, 2, 3, 5\}$, di peso $1 + 1 + 1 + 8 = 11 \leq 13$ e profitto $z(I) = 10 + 6 + 3 + 11 = 30$ è ammissibile e ha profitto maggiore di $GR(I) = 28$. Quindi GR **non** trova una soluzione ottima su questa istanza; la soluzione $\tilde{S}$ è ottima (lo si verifica enumerando le soluzioni ammissibili), e il rapporto di prestazione è

$$\frac{z(I)}{GR(I)} = \frac{30}{28} = \frac{15}{14} \approx 1.07.$$

L'oggetto di profitto massimo, l'oggetto 5 con $p_5 = 11$, non viene selezionato da GR : la sua efficienza bassa lo mette in fondo all'ordine, quando la capacità residua non basta più.

Esercizio 35.3 — Greedy 3

Consideriamo uno zaino di capacità $c = 12$ e un insieme $\mathcal{J} = \{1, 2, 3, 4, 5\}$ di 5 oggetti ($n = 5$), con i seguenti profitti e pesi:

	$i = 1$	$i = 2$	$i = 3$	$i = 4$	$i = 5$
p_i	8	7	18	14	10
w_i	3	3	9	8	7

Domande

1. Verificare che gli oggetti siano ordinati per efficienza non crescente, e determinare l'oggetto critico s e la capacità residua rc .
2. Applicare l'algoritmo greedy (GR) a questa istanza. Mostrare l'esecuzione passo per passo, indicando a ogni iterazione se l'oggetto viene selezionato e la capacità residua che ne risulta. Indicare la soluzione S restituita da GR e $GR(I)$.
3. Calcolare l'upper bound $U = \sum_{i=1}^{s-1} p_i + p_s \frac{rc}{w_s}$ su $z(I)$ dato dal rilassamento lineare, e dedurre un upper bound sul rapporto di prestazione $\frac{z(I)}{GR(I)}$.
4. Stabilire se GR trova una soluzione ottima su questa istanza, mostrando che $GR(I)$ raggiunge l'upper bound $\lfloor U \rfloor$ oppure esibendo una soluzione ammissibile di profitto maggiore.

Soluzione

1. Le efficienze degli oggetti sono

$$\frac{p_1}{w_1} = \frac{8}{3} \approx 2.67, \quad \frac{p_2}{w_2} = \frac{7}{3} \approx 2.33, \quad \frac{p_3}{w_3} = \frac{18}{9} = 2, \quad \frac{p_4}{w_4} = \frac{14}{8} = 1.75, \\ \frac{p_5}{w_5} = \frac{10}{7} \approx 1.43,$$

quindi gli oggetti sono ordinati per efficienza non crescente. Poiché

$$\sum_{i=1}^2 w_i = 3 + 3 = 6 \leq 12 \quad \text{e} \quad \sum_{i=1}^3 w_i = 3 + 3 + 9 = 15 > 12,$$

l'oggetto critico è $s = 3$ e la capacità residua è $rc = 12 - 6 = 6$.

2. All'inizio $rc := c = 12$. L'esecuzione di GR procede come segue:
 - (a) Oggetto 1 (peso $w_1 = 3$): $w_1 = 3 \leq rc = 12$, quindi l'oggetto viene selezionato e $rc \leftarrow 12 - 3 = 9$.
 - (b) Oggetto 2 (peso $w_2 = 3$): $w_2 = 3 \leq rc = 9$, quindi l'oggetto viene selezionato e $rc \leftarrow 9 - 3 = 6$.
 - (c) Oggetto 3 (peso $w_3 = 9$): $w_3 = 9 > rc = 6$, quindi l'oggetto non viene selezionato.
 - (d) Oggetto 4 (peso $w_4 = 8$): $w_4 = 8 > rc = 6$, quindi l'oggetto non viene selezionato.

(e) Oggetto 5 (peso $w_5 = 7$): $w_5 = 7 > rc = 6$, quindi l'oggetto non viene selezionato.

Quindi GR restituisce la soluzione $S = \{1, 2\}$ con

$$GR(I) = 8 + 7 = 15.$$

Qui GR seleziona esattamente gli oggetti che precedono l'oggetto critico.

3. Il rilassamento lineare dà

$$z(I) \leq U = 8 + 7 + 18 \cdot \frac{6}{9} = 27,$$

e poiché $z(I)$ è intero, $z(I) \leq \lfloor U \rfloor = 27$. Quindi

$$\frac{z(I)}{GR(I)} \leq \frac{27}{15} = \frac{9}{5} = 1.8.$$

4. La soluzione $\tilde{S} = \{1, 3\}$, di peso $3 + 9 = 12 \leq 12$ e profitto $z(I) = 8 + 18 = 26$ è ammissibile e ha profitto maggiore di $GR(I) = 15$. Quindi GR **non** trova una soluzione ottima su questa istanza; la soluzione $\tilde{S}$ è ottima (lo si verifica enumerando le soluzioni ammissibili), e il rapporto di prestazione è

$$\frac{z(I)}{GR(I)} = \frac{26}{15} \approx 1.73.$$

L'oggetto di profitto massimo, l'oggetto 3 con $p_3 = 18$, non viene selezionato da GR : la sua efficienza bassa lo mette in fondo all'ordine, quando la capacità residua non basta più.

35.2 Algoritmo greedy esteso

Esercizio 35.4 — Ext-Greedy 1

Consideriamo uno zaino di capacità $c = 22$ e un insieme $\mathcal{I} = \{1, 2, 3, 4, 5\}$ di 5 oggetti ($n = 5$), con i seguenti profitti e pesi:

	$i = 1$	$i = 2$	$i = 3$	$i = 4$	$i = 5$
p_i	11	5	3	4	3
w_i	8	4	3	9	8

Domande

1. Applicare passo per passo l'algoritmo greedy (GR) a questa istanza, e indicare $GR(I)$.

2. Determinare il profitto massimo $p_{\max}$ e la migliore soluzione ammissibile con un solo oggetto $\bar{S}$. Indicare la soluzione restituita dall'algoritmo Ext-Greedy (GR^e) e $GR^e(I)$.
3. Determinare l'oggetto critico s e verificare le due disuguaglianze $\sum_{i=1}^{s-1} p_i \leq GR^e(I)$ e $p_s \leq GR^e(I)$ usate nella dimostrazione della garanzia di prestazione di GR^e . Dedurre un upper bound su $z(I)$.
4. Stabilire se GR^e trova una soluzione ottima su questa istanza, e calcolare il rapporto di prestazione $\frac{z(I)}{GR^e(I)}$, confrontandolo con la garanzia di prestazione relativa 2 di GR^e .

Soluzione

1. Le efficienze degli oggetti sono

$$\frac{p_1}{w_1} = \frac{11}{8} \approx 1.38, \quad \frac{p_2}{w_2} = \frac{5}{4} = 1.25, \quad \frac{p_3}{w_3} = \frac{3}{3} = 1, \quad \frac{p_4}{w_4} = \frac{4}{9} \approx 0.44, \\ \frac{p_5}{w_5} = \frac{3}{8} \approx 0.38,$$

quindi gli oggetti sono ordinati per efficienza non crescente. All'inizio $rc := c = 22$, e GR procede come segue:

- (a) Oggetto 1 (peso $w_1 = 8$): $w_1 = 8 \leq rc = 22$, quindi l'oggetto viene selezionato e $rc \leftarrow 22 - 8 = 14$.
- (b) Oggetto 2 (peso $w_2 = 4$): $w_2 = 4 \leq rc = 14$, quindi l'oggetto viene selezionato e $rc \leftarrow 14 - 4 = 10$.
- (c) Oggetto 3 (peso $w_3 = 3$): $w_3 = 3 \leq rc = 10$, quindi l'oggetto viene selezionato e $rc \leftarrow 10 - 3 = 7$.
- (d) Oggetto 4 (peso $w_4 = 9$): $w_4 = 9 > rc = 7$, quindi l'oggetto non viene selezionato.
- (e) Oggetto 5 (peso $w_5 = 8$): $w_5 = 8 > rc = 7$, quindi l'oggetto non viene selezionato.

Quindi GR restituisce $S = \{1, 2, 3\}$ e $GR(I) = 11 + 5 + 3 = 19$.

2. Il profitto massimo è $p_{\max} = p_1 = 11$, e l'oggetto 1 entra nello zaino ($w_1 = 8 \leq 22$), quindi $\bar{S} = \{1\}$. Poiché $GR(I) = 19 \geq p_{\max} = 11$, GR^e mantiene la soluzione greedy:

$$GR^e(I) = GR(I) = 19.$$

3. Poiché $\sum_{i=1}^3 w_i = 15 \leq 22$ e $\sum_{i=1}^4 w_i = 24 > 22$, l'oggetto critico è $s = 4$. Allora

$$\sum_{i=1}^{s-1} p_i = 11 + 5 + 3 = 19 \leq GR^e(I) = 19 \quad \text{e} \quad p_s = p_4 = 4 \leq GR^e(I) = 19,$$

e, come nella dimostrazione della garanzia di prestazione,

$$z(I) \leq \sum_{i=1}^{s-1} p_i + p_s = 23 \leq 2 GR^e(I) = 38.$$

4. La soluzione $\tilde{S} = \{1, 2, 4\}$, di peso $8+4+9 = 21 \leq 22$ e profitto $z(I) = 11+5+4 = 20$ è ammissibile e migliore di quella restituita da GR^e , quindi GR^e **non** trova una soluzione ottima; $\tilde{S}$ è ottima (lo si verifica enumerando le soluzioni ammissibili). Il rapporto di prestazione è

$$\frac{z(I)}{GR^e(I)} = \frac{20}{19} \approx 1.05 \leq 2,$$

come garantito.

Esercizio 35.5 — Ext-Greedy 2

Consideriamo uno zaino di capacità $c = 8$ e un insieme $\mathcal{I} = \{1, 2, 3, 4, 5\}$ di 5 oggetti ($n = 5$), con i seguenti profitti e pesi:

	$i = 1$	$i = 2$	$i = 3$	$i = 4$	$i = 5$
p_i	16	19	18	14	8
w_i	5	8	8	7	6

Domande

1. Applicare passo per passo l'algoritmo greedy (GR) a questa istanza, e indicare $GR(I)$.
2. Determinare il profitto massimo $p_{\max}$ e la migliore soluzione ammissibile con un solo oggetto $\tilde{S}$. Indicare la soluzione restituita dall'algoritmo Ext-Greedy (GR^e) e $GR^e(I)$.
3. Determinare l'oggetto critico s e verificare le due disuguaglianze $\sum_{i=1}^{s-1} p_i \leq GR^e(I)$ e $p_s \leq GR^e(I)$ usate nella dimostrazione della garanzia di prestazione di GR^e . Dedurre un upper bound su $z(I)$.
4. Stabilire se GR^e trova una soluzione ottima su questa istanza, e calcolare il rapporto di prestazione $\frac{z(I)}{GR^e(I)}$, confrontandolo con la garanzia di prestazione relativa 2 di GR^e .

Soluzione

1. Le efficienze degli oggetti sono

$$\frac{p_1}{w_1} = \frac{16}{5} = 3.2, \quad \frac{p_2}{w_2} = \frac{19}{8} \approx 2.38, \quad \frac{p_3}{w_3} = \frac{18}{8} = 2.25, \quad \frac{p_4}{w_4} = \frac{14}{7} = 2, \quad \frac{p_5}{w_5} = \frac{8}{6} \approx 1.33,$$

quindi gli oggetti sono ordinati per efficienza non crescente. All'inizio $rc := c = 8$, e GR procede come segue:

- (a) Oggetto 1 (peso $w_1 = 5$): $w_1 = 5 \leq rc = 8$, quindi l'oggetto viene selezionato e $rc \leftarrow 8 - 5 = 3$.

- (b) Oggetto 2 (peso $w_2 = 8$): $w_2 = 8 > rc = 3$, quindi l'oggetto non viene selezionato.
- (c) Oggetto 3 (peso $w_3 = 8$): $w_3 = 8 > rc = 3$, quindi l'oggetto non viene selezionato.
- (d) Oggetto 4 (peso $w_4 = 7$): $w_4 = 7 > rc = 3$, quindi l'oggetto non viene selezionato.
- (e) Oggetto 5 (peso $w_5 = 6$): $w_5 = 6 > rc = 3$, quindi l'oggetto non viene selezionato.

Quindi GR restituisce $S = \{1\}$ e $GR(I) = 16$.

2. Il profitto massimo è $p_{\max} = p_2 = 19$, e l'oggetto 2 entra nello zaino ($w_2 = 8 \leq 8$), quindi $\tilde{S} = \{2\}$. Poiché $GR(I) = 16 < p_{\max} = 19$, GR^e sostituisce la soluzione greedy con $\tilde{S}$:

$$GR^e(I) = p_{\max} = 19.$$

3. Poiché $\sum_{i=1}^1 w_i = 5 \leq 8$ e $\sum_{i=1}^2 w_i = 13 > 8$, l'oggetto critico è $s = 2$. Allora

$$\sum_{i=1}^{s-1} p_i = 16 \leq GR^e(I) = 19 \quad \text{e} \quad p_s = p_2 = 19 \leq GR^e(I) = 19,$$

e, come nella dimostrazione della garanzia di prestazione,

$$z(I) \leq \sum_{i=1}^{s-1} p_i + p_s = 35 \leq 2 GR^e(I) = 38.$$

4. La soluzione restituita da GR^e è **ottima**: $\tilde{S} = \{2\}$, di peso $8 \leq 8$ e profitto $z(I) = 19$. Il rapporto di prestazione è $\frac{z(I)}{GR^e(I)} = 1$.

Esercizio 35.6 — Ext-Greedy 3

Consideriamo uno zaino di capacità $c = 16$ e un insieme $\mathcal{J} = \{1, 2, 3\}$ di 3 oggetti ($n = 3$), con i seguenti profitti e pesi:

	$i = 1$	$i = 2$	$i = 3$
p_i	2	8	8
w_i	1	8	8

Domande

1. Applicare passo per passo l'algoritmo greedy (GR) a questa istanza, e indicare $GR(I)$.
2. Determinare il profitto massimo $p_{\max}$ e la migliore soluzione ammissibile con un solo oggetto $\tilde{S}$. Indicare la soluzione restituita dall'algoritmo Ext-Greedy (GR^e) e $GR^e(I)$.

3. Determinare l'oggetto critico s e verificare le due disuguaglianze $\sum_{i=1}^{s-1} p_i \leq GR^e(I)$ e $p_s \leq GR^e(I)$ usate nella dimostrazione della garanzia di prestazione di GR^e . Dedurre un upper bound su $z(I)$.
4. Stabilire se GR^e trova una soluzione ottima su questa istanza, e calcolare il rapporto di prestazione $\frac{z(I)}{GR^e(I)}$, confrontandolo con la garanzia di prestazione relativa 2 di GR^e .

Soluzione

1. Le efficienze degli oggetti sono

$$\frac{p_1}{w_1} = \frac{2}{1} = 2, \quad \frac{p_2}{w_2} = \frac{8}{8} = 1, \quad \frac{p_3}{w_3} = \frac{8}{8} = 1,$$

quindi gli oggetti sono ordinati per efficienza non crescente. All'inizio $rc := c = 16$, e GR procede come segue:

- (a) Oggetto 1 (peso $w_1 = 1$): $w_1 = 1 \leq rc = 16$, quindi l'oggetto viene selezionato e $rc \leftarrow 16 - 1 = 15$.
- (b) Oggetto 2 (peso $w_2 = 8$): $w_2 = 8 \leq rc = 15$, quindi l'oggetto viene selezionato e $rc \leftarrow 15 - 8 = 7$.
- (c) Oggetto 3 (peso $w_3 = 8$): $w_3 = 8 > rc = 7$, quindi l'oggetto non viene selezionato.

Quindi GR restituisce $S = \{1, 2\}$ e $GR(I) = 2 + 8 = 10$.

2. Il profitto massimo è $p_{\max} = p_2 = 8$, e l'oggetto 2 entra nello zaino ($w_2 = 8 \leq 16$), quindi $\tilde{S} = \{2\}$. Poiché $GR(I) = 10 \geq p_{\max} = 8$, GR^e mantiene la soluzione greedy:

$$GR^e(I) = GR(I) = 10.$$

3. Poiché $\sum_{i=1}^2 w_i = 9 \leq 16$ e $\sum_{i=1}^3 w_i = 17 > 16$, l'oggetto critico è $s = 3$. Allora

$$\sum_{i=1}^{s-1} p_i = 2 + 8 = 10 \leq GR^e(I) = 10 \quad \text{e} \quad p_s = p_3 = 8 \leq GR^e(I) = 10,$$

e, come nella dimostrazione della garanzia di prestazione,

$$z(I) \leq \sum_{i=1}^{s-1} p_i + p_s = 18 \leq 2 GR^e(I) = 20.$$

4. La soluzione $\tilde{S} = \{2, 3\}$, di peso $8 + 8 = 16 \leq 16$ e profitto $z(I) = 8 + 8 = 16$ è ammissibile e migliore di quella restituita da GR^e , quindi GR^e **non** trova una soluzione ottima; $\tilde{S}$ è ottima (lo si verifica enumerando le soluzioni ammissibili). Il rapporto di prestazione è

$$\frac{z(I)}{GR^e(I)} = \frac{16}{10} = \frac{8}{5} = 1.6 \leq 2,$$

come garantito.

Osservazione. Questa istanza appartiene alla famiglia di istanze usata per mostrare che la garanzia di prestazione relativa 2 di GR^e è tight, con $\vartheta = 8$: lì il rapporto è $\frac{2\vartheta}{\vartheta+2}$, e qui $\frac{16}{10} = 1.6$; tende a 2 al crescere di ϑ .

35.3 L'euristica H^2

Esercizio 35.7 — H^2 1

Consideriamo uno zaino di capacità $c = 11$ e un insieme $\mathcal{I} = \{1, 2, 3, 4, 5\}$ di 5 oggetti ($n = 5$), con i seguenti profitti e pesi:

	$i = 1$	$i = 2$	$i = 3$	$i = 4$	$i = 5$
p_i	8	9	2	3	2
w_i	5	8	3	7	9

Domande

1. Verificare che gli oggetti siano ordinati per efficienza non crescente, e determinare l'insieme $\mathcal{P}$ delle coppie ammissibili di oggetti.
2. Applicare l'algoritmo H^2 a questa istanza: per ogni coppia ammissibile (i, k) calcolare la capacità residua rc , l'insieme residuo $\mathcal{I}(i, k)$, il valore di GR^e sull'istanza residua e il valore $lb(i, k)$.
3. Indicare la soluzione restituita da H^2 e $H^2(I)$.
4. Confrontare $H^2(I)$ con $GR^e(I)$ e con $z(I)$, e calcolare il rapporto di prestazione $\frac{z(I)}{H^2(I)}$, confrontandolo con la garanzia di prestazione relativa $\frac{4}{3}$ di H^2 .

Soluzione

1. Le efficienze degli oggetti sono

$$\frac{p_1}{w_1} = \frac{8}{5} = 1.6, \quad \frac{p_2}{w_2} = \frac{9}{8} \approx 1.12, \quad \frac{p_3}{w_3} = \frac{2}{3} \approx 0.67, \quad \frac{p_4}{w_4} = \frac{3}{7} \approx 0.43, \\ \frac{p_5}{w_5} = \frac{2}{9} \approx 0.22,$$

quindi gli oggetti sono ordinati per efficienza non crescente. Una coppia (i, k) con $i < k$ è ammissibile se $w_i + w_k \leq c = 11$, quindi

$$\mathcal{P} = \{(1, 3), (2, 3), (3, 4)\}.$$

2. L'algoritmo parte dal miglior oggetto singolo, $lb = p_{\max} = p_2 = 9$. Per ogni coppia ammissibile, l'insieme residuo mantiene gli oggetti $j \notin \{i, k\}$ con $p_j \leq \min\{p_i, p_k\}$ e $w_j \leq rc$:

coppia	peso	rc	insieme residuo	GR^e sul residuo	valore
(1, 2)	13 > 11		non ammissibile		
(1, 3)	8	3	$\emptyset$	0	10
(1, 4)	12 > 11		non ammissibile		
(1, 5)	14 > 11		non ammissibile		
(2, 3)	11	0	$\emptyset$	0	11
(2, 4)	15 > 11		non ammissibile		
(2, 5)	17 > 11		non ammissibile		
(3, 4)	10	1	$\emptyset$	0	5
(3, 5)	12 > 11		non ammissibile		
(4, 5)	16 > 11		non ammissibile		

3. Il valore massimo è 11, ottenuto con la soluzione $\{2, 3\}$. Quindi

$$H^2(I) = 11.$$

4. Su questa istanza $GR^e(I) = 10$, quindi H^2 migliora GR^e . La soluzione restituita da H^2 è **ottima**: $\tilde{S} = \{2, 3\}$, di peso $8 + 3 = 11 \leq 11$ e profitto $z(I) = 9 + 2 = 11$, e il rapporto di prestazione è 1.

Esercizio 35.8 — H^2 2

Consideriamo uno zaino di capacità $c = 15$ e un insieme $\mathcal{J} = \{1, 2, 3, 4, 5\}$ di 5 oggetti ($n = 5$), con i seguenti profitti e pesi:

	$i = 1$	$i = 2$	$i = 3$	$i = 4$	$i = 5$
p_i	14	9	5	10	15
w_i	1	1	2	5	8

Domande

1. Verificare che gli oggetti siano ordinati per efficienza non crescente, e determinare l'insieme $\mathcal{P}$ delle coppie ammissibili di oggetti.
2. Applicare l'algoritmo H^2 a questa istanza: per ogni coppia ammissibile (i, k) calcolare la capacità residua rc , l'insieme residuo $\mathcal{J}(i, k)$, il valore di GR^e sull'istanza residua e il valore $lb(i, k)$.
3. Indicare la soluzione restituita da H^2 e $H^2(I)$.

4. Confrontare $H^2(I)$ con $GR^e(I)$ e con $z(I)$, e calcolare il rapporto di prestazione $\frac{z(I)}{H^2(I)}$, confrontandolo con la garanzia di prestazione relativa $\frac{4}{3}$ di H^2 .

Soluzione

1. Le efficienze degli oggetti sono

$$\frac{p_1}{w_1} = \frac{14}{1} = 14, \quad \frac{p_2}{w_2} = \frac{9}{1} = 9, \quad \frac{p_3}{w_3} = \frac{5}{2} = 2.5, \quad \frac{p_4}{w_4} = \frac{10}{5} = 2, \\ \frac{p_5}{w_5} = \frac{15}{8} \approx 1.88,$$

quindi gli oggetti sono ordinati per efficienza non crescente. Una coppia (i, k) con $i < k$ è ammissibile se $w_i + w_k \leq c = 15$, quindi

$$\mathcal{P} = \{(1, 2), (1, 3), (1, 4), (1, 5), (2, 3), (2, 4), (2, 5), (3, 4), (3, 5), (4, 5)\}.$$

2. L'algoritmo parte dal miglior oggetto singolo, $lb = p_{\max} = p_5 = 15$. Per ogni coppia ammissibile, l'insieme residuo mantiene gli oggetti $j \notin \{i, k\}$ con $p_j \leq \min\{p_i, p_k\}$ e $w_j \leq rc$:

coppia	peso	rc	insieme residuo	GR^e sul residuo	valore
(1, 2)	2	13	{3}	5	28
(1, 3)	3	12	$\emptyset$	0	19
(1, 4)	6	9	{2, 3}	14	38
(1, 5)	9	6	{2, 3, 4}	14	43
(2, 3)	3	12	$\emptyset$	0	14
(2, 4)	6	9	{3}	5	24
(2, 5)	9	6	{3}	5	29
(3, 4)	7	8	$\emptyset$	0	15
(3, 5)	10	5	$\emptyset$	0	20
(4, 5)	13	2	{2, 3}	9	34

3. Il valore massimo è 43, ottenuto con la soluzione $\{1, 2, 3, 5\}$. Quindi

$$H^2(I) = 43.$$

4. Su questa istanza $GR^e(I) = 38$, quindi H^2 migliora GR^e . La soluzione $\tilde{S} = \{1, 2, 4, 5\}$, di peso $1 + 1 + 5 + 8 = 15 \leq 15$ e profitto $z(I) = 14 + 9 + 10 + 15 = 48$ è migliore, quindi H^2 **non** trova una soluzione ottima; $\tilde{S}$ è ottima (lo si verifica enumerando le soluzioni ammissibili). Il rapporto di prestazione è

$$\frac{z(I)}{H^2(I)} = \frac{48}{43} \approx 1.12 \leq \frac{4}{3} \approx 1.33,$$

come garantito.

Esercizio 35.9 — H^2 3

Consideriamo uno zaino di capacità $c = 10$ e un insieme $\mathcal{I} = \{1, 2, 3, 4, 5\}$ di 5 oggetti ($n = 5$), con i seguenti profitti e pesi:

	$i = 1$	$i = 2$	$i = 3$	$i = 4$	$i = 5$
p_i	13	5	8	5	7
w_i	1	1	3	2	5

Domande

1. Verificare che gli oggetti siano ordinati per efficienza non crescente, e determinare l'insieme $\mathcal{P}$ delle coppie ammissibili di oggetti.
2. Applicare l'algoritmo H^2 a questa istanza: per ogni coppia ammissibile (i, k) calcolare la capacità residua rc , l'insieme residuo $\mathcal{I}(i, k)$, il valore di GR^e sull'istanza residua e il valore $lb(i, k)$.
3. Indicare la soluzione restituita da H^2 e $H^2(I)$.
4. Confrontare $H^2(I)$ con $GR^e(I)$ e con $z(I)$, e calcolare il rapporto di prestazione $\frac{z(I)}{H^2(I)}$, confrontandolo con la garanzia di prestazione relativa $\frac{4}{3}$ di H^2 .

Soluzione

1. Le efficienze degli oggetti sono

$$\frac{p_1}{w_1} = \frac{13}{1} = 13, \quad \frac{p_2}{w_2} = \frac{5}{1} = 5, \quad \frac{p_3}{w_3} = \frac{8}{3} \approx 2.67, \quad \frac{p_4}{w_4} = \frac{5}{2} = 2.5, \quad \frac{p_5}{w_5} = \frac{7}{5} = 1.4,$$

quindi gli oggetti sono ordinati per efficienza non crescente. Una coppia (i, k) con $i < k$ è ammissibile se $w_i + w_k \leq c = 10$, quindi

$$\mathcal{P} = \{(1, 2), (1, 3), (1, 4), (1, 5), (2, 3), (2, 4), (2, 5), (3, 4), (3, 5), (4, 5)\}.$$

2. L'algoritmo parte dal miglior oggetto singolo, $lb = p_{\max} = p_1 = 13$. Per ogni coppia ammissibile, l'insieme residuo mantiene gli oggetti $j \notin \{i, k\}$ con $p_j \leq \min\{p_i, p_k\}$ e $w_j \leq rc$:

coppia	peso	rc	insieme residuo	GR^e sul residuo	valore
(1, 2)	2	8	{4}	5	23
(1, 3)	4	6	{2, 4, 5}	10	31
(1, 4)	3	7	{2}	5	23
(1, 5)	6	4	{2, 4}	10	30
(2, 3)	4	6	{4}	5	18
(2, 4)	3	7	$\emptyset$	0	10
(2, 5)	6	4	{4}	5	17
(3, 4)	5	5	{2}	5	18
(3, 5)	8	2	{2, 4}	5	20
(4, 5)	7	3	{2}	5	17

3. Il valore massimo è 31, ottenuto con la soluzione $\{1, 2, 3, 4\}$. Quindi

$$H^2(I) = 31.$$

4. Su questa istanza $GR^e(I) = 31$, lo stesso valore di H^2 . La soluzione $\tilde{S} = \{1, 2, 3, 5\}$, di peso $1 + 1 + 3 + 5 = 10 \leq 10$ e profitto $z(I) = 13 + 5 + 8 + 7 = 33$ è migliore, quindi H^2 **non** trova una soluzione ottima; $\tilde{S}$ è ottima (lo si verifica enumerando le soluzioni ammissibili). Il rapporto di prestazione è

$$\frac{z(I)}{H^2(I)} = \frac{33}{31} \approx 1.06 \leq \frac{4}{3} \approx 1.33,$$

come garantito.

35.4 L'euristica H^ℓ

Esercizio 35.10 — H^ℓ 1

Consideriamo uno zaino di capacità $c = 16$ e un insieme $\mathcal{S} = \{1, 2, 3, 4, 5\}$ di 5 oggetti ($n = 5$), con i seguenti profitti e pesi:

	$i = 1$	$i = 2$	$i = 3$	$i = 4$	$i = 5$
p_i	14	13	13	10	11
w_i	1	2	3	3	9

Domande

1. Applichiamo l'algoritmo H^ℓ con $\ell = 3$. Calcolare la migliore soluzione ammissibile formata da al più $\ell - 1 = 2$ oggetti.

2. Per ogni insieme ammissibile L di esattamente $\ell = 3$ oggetti, calcolare la capacità residua rc , l'insieme residuo $\mathcal{J}(L)$, il valore di GR^e sull'istanza residua e il valore $lb(L)$.
3. Indicare la soluzione restituita da H^3 e $H^3(I)$.
4. Stabilire se H^3 trova una soluzione ottima su questa istanza, e calcolare il rapporto di prestazione $\frac{z(I)}{H^3(I)}$, confrontandolo con la garanzia di prestazione relativa $\frac{\ell+2}{\ell+1} = \frac{5}{4}$.

Soluzione

1. La migliore soluzione ammissibile con al più 2 oggetti è $\{1, 2\}$, di peso $1 + 2 = 3 \leq 16$ e profitto $14 + 13 = 27$.
2. Per ogni insieme L di 3 oggetti, l'insieme residuo mantiene gli oggetti $j \notin L$ con $p_j \leq \min\{p_i : i \in L\}$ e $w_j \leq rc$:

insieme L	peso	rc	insieme residuo	GR^e sul residuo	valore
$\{1, 2, 3\}$	6	10	$\{4, 5\}$	11	51
$\{1, 2, 4\}$	6	10	$\emptyset$	0	37
$\{1, 2, 5\}$	12	4	$\{4\}$	10	48
$\{1, 3, 4\}$	7	9	$\emptyset$	0	37
$\{1, 3, 5\}$	13	3	$\{4\}$	10	48
$\{1, 4, 5\}$	13	3	$\emptyset$	0	35
$\{2, 3, 4\}$	8	8	$\emptyset$	0	36
$\{2, 3, 5\}$	14	2	$\emptyset$	0	37
$\{2, 4, 5\}$	14	2	$\emptyset$	0	34
$\{3, 4, 5\}$	15	1	$\emptyset$	0	34

3. Il valore massimo è 51, ottenuto con la soluzione $\{1, 2, 3, 5\}$. Quindi

$$H^3(I) = 51.$$

4. La soluzione restituita da H^3 è **ottima**: $\tilde{S} = \{1, 2, 3, 5\}$, di peso $1 + 2 + 3 + 9 = 15 \leq 16$ e profitto $z(I) = 14 + 13 + 13 + 11 = 51$, e il rapporto di prestazione è 1. Per confronto, su questa istanza $H^2(I) = 50$: scommettere su tre oggetti invece che su due basta per trovare la soluzione ottima.

Esercizio 35.11 — H^ℓ 2

Consideriamo uno zaino di capacità $c = 20$ e un insieme $\mathcal{J} = \{1, 2, 3, 4, 5, 6\}$ di 6 oggetti ($n = 6$), con i seguenti profitti e pesi:

	$i = 1$	$i = 2$	$i = 3$	$i = 4$	$i = 5$	$i = 6$
p_i	2	4	4	4	4	4
w_i	1	4	4	4	4	4

Domande

1. Applichiamo l'algoritmo H^ℓ con $\ell = 3$. Calcolare la migliore soluzione ammissibile formata da al più $\ell - 1 = 2$ oggetti.
2. Per ogni insieme ammissibile L di esattamente $\ell = 3$ oggetti, calcolare la capacità residua rc , l'insieme residuo $\mathcal{J}(L)$, il valore di GR^e sull'istanza residua e il valore $lb(L)$.
3. Indicare la soluzione restituita da H^3 e $H^3(I)$.
4. Stabilire se H^3 trova una soluzione ottima su questa istanza, e calcolare il rapporto di prestazione $\frac{z(I)}{H^3(I)}$, confrontandolo con la garanzia di prestazione relativa $\frac{\ell+2}{\ell+1} = \frac{5}{4}$.

Soluzione

1. La migliore soluzione ammissibile con al più 2 oggetti è $\{2, 3\}$, di peso $4 + 4 = 8 \leq 20$ e profitto $4 + 4 = 8$.
2. Per ogni insieme L di 3 oggetti, l'insieme residuo mantiene gli oggetti $j \notin L$ con $p_j \leq \min\{p_i : i \in L\}$ e $w_j \leq rc$:

insieme L	peso	rc	insieme residuo	GR^e sul residuo	valore
$\{1, 2, 3\}$	9	11	$\emptyset$	0	10
$\{1, 2, 4\}$	9	11	$\emptyset$	0	10
$\{1, 2, 5\}$	9	11	$\emptyset$	0	10
$\{1, 2, 6\}$	9	11	$\emptyset$	0	10
$\{1, 3, 4\}$	9	11	$\emptyset$	0	10
$\{1, 3, 5\}$	9	11	$\emptyset$	0	10
$\{1, 3, 6\}$	9	11	$\emptyset$	0	10
$\{1, 4, 5\}$	9	11	$\emptyset$	0	10
$\{1, 4, 6\}$	9	11	$\emptyset$	0	10
$\{1, 5, 6\}$	9	11	$\emptyset$	0	10
$\{2, 3, 4\}$	12	8	$\{1, 5, 6\}$	6	18
$\{2, 3, 5\}$	12	8	$\{1, 4, 6\}$	6	18
$\{2, 3, 6\}$	12	8	$\{1, 4, 5\}$	6	18
$\{2, 4, 5\}$	12	8	$\{1, 3, 6\}$	6	18
$\{2, 4, 6\}$	12	8	$\{1, 3, 5\}$	6	18
$\{2, 5, 6\}$	12	8	$\{1, 3, 4\}$	6	18
$\{3, 4, 5\}$	12	8	$\{1, 2, 6\}$	6	18
$\{3, 4, 6\}$	12	8	$\{1, 2, 5\}$	6	18
$\{3, 5, 6\}$	12	8	$\{1, 2, 4\}$	6	18
$\{4, 5, 6\}$	12	8	$\{1, 2, 3\}$	6	18

3. Il valore massimo è 18, ottenuto con la soluzione $\{1, 2, 3, 4, 5\}$. Quindi

$$H^3(I) = 18.$$

4. La soluzione $\tilde{S} = \{2, 3, 4, 5, 6\}$, di peso $4 + 4 + 4 + 4 + 4 = 20 \leq 20$ e profitto $z(I) = 4 + 4 + 4 + 4 + 4 = 20$ è migliore, quindi H^3 **non** trova una soluzione ottima; $\tilde{S}$ è ottima (lo si verifica enumerando le soluzioni ammissibili). Il rapporto di prestazione è

$$\frac{z(I)}{H^3(I)} = \frac{20}{18} = \frac{10}{9} \approx 1.11 \leq \frac{5}{4} = 1.25,$$

come garantito. Per confronto, su questa istanza $H^2(I) = 18$.

Osservazione. Questa istanza appartiene alla famiglia di istanze usata per mostrare che la garanzia di prestazione relativa $\frac{\ell+2}{\ell+1}$ di H^ℓ è tight, con $\ell = 3$ e $M = 4$: lì $H^\ell(I_M) = (\ell + 1)M + 2$ e $z(I_M) = (\ell + 2)M$, qui 18 e 20.

Esercizio 35.12 — H^ℓ 3

Consideriamo uno zaino di capacità $c = 18$ e un insieme $\mathcal{S} = \{1, 2, 3, 4, 5, 6\}$ di 6 oggetti ($n = 6$), con i seguenti profitti e pesi:

	$i = 1$	$i = 2$	$i = 3$	$i = 4$	$i = 5$	$i = 6$
p_i	15	15	14	8	12	13
w_i	1	2	3	2	6	9

Domande

1. Applichiamo l'algoritmo H^ℓ con $\ell = 3$. Calcolare la migliore soluzione ammissibile formata da al più $\ell - 1 = 2$ oggetti.
2. Per ogni insieme ammissibile L di esattamente $\ell = 3$ oggetti, calcolare la capacità residua rc , l'insieme residuo $\mathcal{J}(L)$, il valore di GR^e sull'istanza residua e il valore $lb(L)$.
3. Indicare la soluzione restituita da H^3 e $H^3(I)$.
4. Stabilire se H^3 trova una soluzione ottima su questa istanza, e calcolare il rapporto di prestazione $\frac{z(I)}{H^3(I)}$, confrontandolo con la garanzia di prestazione relativa $\frac{\ell+2}{\ell+1} = \frac{5}{4}$.

Soluzione

1. La migliore soluzione ammissibile con al più 2 oggetti è $\{1, 2\}$, di peso $1 + 2 = 3 \leq 18$ e profitto $15 + 15 = 30$.
2. Per ogni insieme L di 3 oggetti, l'insieme residuo mantiene gli oggetti $j \notin L$ con $p_j \leq \min\{p_i : i \in L\}$ e $w_j \leq rc$:

insieme L	peso	rc	insieme residuo	GR^e sul residuo	valore
$\{1, 2, 3\}$	6	12	$\{4, 5, 6\}$	20	64
$\{1, 2, 4\}$	5	13	$\emptyset$	0	38
$\{1, 2, 5\}$	9	9	$\{4\}$	8	50
$\{1, 2, 6\}$	12	6	$\{4, 5\}$	12	55
$\{1, 3, 4\}$	6	12	$\emptyset$	0	37
$\{1, 3, 5\}$	10	8	$\{4\}$	8	49
$\{1, 3, 6\}$	13	5	$\{4\}$	8	50
$\{1, 4, 5\}$	9	9	$\emptyset$	0	35
$\{1, 4, 6\}$	12	6	$\emptyset$	0	36
$\{1, 5, 6\}$	16	2	$\{4\}$	8	48
$\{2, 3, 4\}$	7	11	$\emptyset$	0	37
$\{2, 3, 5\}$	11	7	$\{4\}$	8	49
$\{2, 3, 6\}$	14	4	$\{4\}$	8	50
$\{2, 4, 5\}$	10	8	$\emptyset$	0	35
$\{2, 4, 6\}$	13	5	$\emptyset$	0	36
$\{2, 5, 6\}$	17	1	$\emptyset$	0	40
$\{3, 4, 5\}$	11	7	$\emptyset$	0	34
$\{3, 4, 6\}$	14	4	$\emptyset$	0	35
$\{3, 5, 6\}$	18	0	$\emptyset$	0	39
$\{4, 5, 6\}$	17	1	$\emptyset$	0	33

3. Il valore massimo è 64, ottenuto con la soluzione $\{1, 2, 3, 4, 5\}$. Quindi

$$H^3(I) = 64.$$

4. La soluzione $\tilde{S} = \{1, 2, 3, 4, 6\}$, di peso $1 + 2 + 3 + 2 + 9 = 17 \leq 18$ e profitto $z(I) = 15 + 15 + 14 + 8 + 13 = 65$ è migliore, quindi H^3 **non** trova una soluzione ottima; $\tilde{S}$ è ottima (lo si verifica enumerando le soluzioni ammissibili). Il rapporto di prestazione è

$$\frac{z(I)}{H^3(I)} = \frac{65}{64} \approx 1.02 \leq \frac{5}{4} = 1.25,$$

come garantito. Per confronto, su questa istanza $H^2(I) = 64$.

35.5 Lo schema FPTAS

Esercizio 35.13 — Basic-FPTAS 1

Consideriamo uno zaino di capacità $c = 13$ e un insieme $\mathcal{I} = \{1, 2, 3, 4\}$ di 4 oggetti ($n = 4$), con i seguenti profitti e pesi:

	$i = 1$	$i = 2$	$i = 3$	$i = 4$
p_i	24	11	23	5
w_i	2	3	9	3

Applichiamo l'algoritmo Basic-FPTAS con accuratezza $\varepsilon = \frac{1}{2}$.

Domande

1. Calcolare il lower bound $GR^e(I)$ usato dall'algoritmo Basic-FPTAS.
2. Calcolare il fattore di scala e i profitti scalati

$$K = \frac{\varepsilon GR^e(I)}{n} \quad \text{e} \quad p'_j = \left\lfloor \frac{p_j}{K} \right\rfloor \quad \text{per ogni } j \in \mathcal{I}.$$

3. Risolvere l'istanza scalata con la programmazione dinamica sui profitti, cioè calcolare i pesi minimi $u(j, q)$ e trovare il massimo profitto q tale che $u(4, q) \leq c$.
4. Indicare la soluzione restituita da FP^ε e il suo profitto $FP^\varepsilon(I)$ nell'istanza originale, e confrontarlo con $GR^e(I)$ e $z(I)$. Calcolare il rapporto di prestazione e confrontarlo con la garanzia $\frac{1}{1-\varepsilon}$.

Soluzione

1. Gli oggetti sono ordinati per efficienza non crescente. GR seleziona gli oggetti $\{1, 2, 4\}$, con $GR(I) = 24 + 11 + 5 = 40$; il profitto massimo è $p_{\max} = p_1 = 24$. Quindi

$$GR^e(I) = 40.$$

2. Il fattore di scala è

$$K = \frac{\varepsilon GR^e(I)}{n} = \frac{\frac{1}{2} \cdot 40}{4} = 5,$$

e i profitti scalati sono

$$p'_1 = \left\lfloor \frac{24}{5} \right\rfloor = 4, \quad p'_2 = \left\lfloor \frac{11}{5} \right\rfloor = 2, \quad p'_3 = \left\lfloor \frac{23}{5} \right\rfloor = 4, \quad p'_4 = \left\lfloor \frac{5}{5} \right\rfloor = 1.$$

Il profitto scalato totale è $P' = 4 + 2 + 4 + 1 = 11$.

3. I pesi minimi $u(j, q)$ di un sottoinsieme di $\{1, \dots, j\}$ di profitto scalato esattamente q sono (sono riportati solo i valori di q ottenibili; ∞ indica che q non si può ottenere con i primi j oggetti):

q	0	1	2	3	4	5	6	7	8	9	10
$u(0, q)$	0	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
$u(1, q)$	0	∞	∞	∞	2	∞	∞	∞	∞	∞	∞
$u(2, q)$	0	∞	3	∞	2	∞	5	∞	∞	∞	∞
$u(3, q)$	0	∞	3	∞	2	∞	5	∞	11	∞	14
$u(4, q)$	0	3	3	6	2	5	5	8	11	14	14

q	11
$u(0, q)$	∞
$u(1, q)$	∞
$u(2, q)$	∞
$u(3, q)$	∞
$u(4, q)$	17

Il massimo q con $u(4, q) \leq c = 13$ è $q = 8$, di peso $u(4, 8) = 11$, ottenuto con gli oggetti $\{1, 3\}$.

4. FP^ε restituisce $S = \{1, 3\}$, di profitto

$$FP^\varepsilon(I) = 24 + 23 = 47$$

nell'istanza originale, migliore di $GR^e(I) = 40$. La soluzione è **ottima**: $\tilde{S} = \{1, 3\}$, di peso $2 + 9 = 11 \leq 13$ e profitto $z(I) = 24 + 23 = 47$, e il rapporto di prestazione è 1.

Esercizio 35.14 — Basic-FPTAS 2

Consideriamo uno zaino di capacità $c = 16$ e un insieme $\mathcal{I} = \{1, 2, 3, 4\}$ di 4 oggetti ($n = 4$), con i seguenti profitti e pesi:

	$i = 1$	$i = 2$	$i = 3$	$i = 4$
p_i	14	12	14	18
w_i	3	4	5	7

Applichiamo l'algoritmo Basic-FPTAS con accuratezza $\varepsilon = \frac{1}{2}$.

Domande

1. Calcolare il lower bound $GR^e(I)$ usato dall'algoritmo Basic-FPTAS.
2. Calcolare il fattore di scala e i profitti scalati

$$K = \frac{\varepsilon GR^e(I)}{n} \quad \text{e} \quad p'_j = \left\lfloor \frac{p_j}{K} \right\rfloor \quad \text{per ogni } j \in \mathcal{I}.$$

3. Risolvere l'istanza scalata con la programmazione dinamica sui profitti, cioè calcolare i pesi minimi $u(j, q)$ e trovare il massimo profitto q tale che $u(4, q) \leq c$.
4. Indicare la soluzione restituita da FP^ε e il suo profitto $FP^\varepsilon(I)$ nell'istanza originale, e confrontarlo con $GR^e(I)$ e $z(I)$. Calcolare il rapporto di prestazione e confrontarlo con la garanzia $\frac{1}{1-\varepsilon}$.

Soluzione

1. Gli oggetti sono ordinati per efficienza non crescente. GR seleziona gli oggetti $\{1, 2, 3\}$, con $GR(I) = 14 + 12 + 14 = 40$; il profitto massimo è $p_{\max} = p_4 = 18$. Quindi

$$GR^e(I) = 40.$$

2. Il fattore di scala è

$$K = \frac{\varepsilon GR^e(I)}{n} = \frac{\frac{1}{2} \cdot 40}{4} = 5,$$

e i profitti scalati sono

$$p'_1 = \left\lfloor \frac{14}{5} \right\rfloor = 2, \quad p'_2 = \left\lfloor \frac{12}{5} \right\rfloor = 2, \quad p'_3 = \left\lfloor \frac{14}{5} \right\rfloor = 2, \quad p'_4 = \left\lfloor \frac{18}{5} \right\rfloor = 3.$$

Il profitto scalato totale è $P' = 2 + 2 + 2 + 3 = 9$.

3. I pesi minimi $u(j, q)$ di un sottoinsieme di $\{1, \dots, j\}$ di profitto scalato esattamente q sono (sono riportati solo i valori di q ottenibili; ∞ indica che q non si può ottenere con i primi j oggetti):

q	0	2	3	4	5	6	7	9
$u(0, q)$	0	∞	∞	∞	∞	∞	∞	∞
$u(1, q)$	0	3	∞	∞	∞	∞	∞	∞
$u(2, q)$	0	3	∞	7	∞	∞	∞	∞
$u(3, q)$	0	3	∞	7	∞	12	∞	∞
$u(4, q)$	0	3	7	7	10	12	14	19

Il massimo q con $u(4, q) \leq c = 16$ è $q = 7$, di peso $u(4, 7) = 14$, ottenuto con gli oggetti $\{1, 2, 4\}$.

4. FP^ε restituisce $S = \{1, 2, 4\}$, di profitto

$$FP^\varepsilon(I) = 14 + 12 + 18 = 44$$

nell'istanza originale, migliore di $GR^e(I) = 40$. La soluzione $\tilde{S} = \{1, 3, 4\}$, di peso $3 + 5 + 7 = 15 \leq 16$ e profitto $z(I) = 14 + 14 + 18 = 46$ è migliore, quindi FP^ε **non** trova una soluzione ottima; $\tilde{S}$ è ottima (lo si verifica enumerando le soluzioni ammissibili). Il rapporto di prestazione è

$$\frac{z(I)}{FP^\varepsilon(I)} = \frac{46}{44} = \frac{23}{22} \approx 1.05 \leq \frac{1}{1-\varepsilon} = 2,$$

come garantito.

Esercizio 35.15 — Basic-FPTAS 3

Consideriamo uno zaino di capacità $c = 16$ e un insieme $\mathcal{I} = \{1, 2, 3, 4\}$ di 4 oggetti ($n = 4$), con i seguenti profitti e pesi:

	$i = 1$	$i = 2$	$i = 3$	$i = 4$
p_i	14	12	14	18
w_i	3	4	5	7

Applichiamo l'algoritmo Basic-FPTAS con accuratezza $\varepsilon = \frac{1}{4}$.

Domande

1. Calcolare il lower bound $GR^e(I)$ usato dall'algoritmo Basic-FPTAS.
2. Calcolare il fattore di scala e i profitti scalati

$$K = \frac{\varepsilon GR^e(I)}{n} \quad \text{e} \quad p'_j = \left\lfloor \frac{p_j}{K} \right\rfloor \quad \text{per ogni } j \in \mathcal{I}.$$

3. Risolvere l'istanza scalata con la programmazione dinamica sui profitti, cioè calcolare i pesi minimi $u(j, q)$ e trovare il massimo profitto q tale che $u(4, q) \leq c$.
4. Indicare la soluzione restituita da FP^ε e il suo profitto $FP^\varepsilon(I)$ nell'istanza originale, e confrontarlo con $GR^e(I)$ e $z(I)$. Calcolare il rapporto di prestazione e confrontarlo con la garanzia $\frac{1}{1-\varepsilon}$.

Soluzione

1. Gli oggetti sono ordinati per efficienza non crescente. GR seleziona gli oggetti $\{1, 2, 3\}$, con $GR(I) = 14 + 12 + 14 = 40$; il profitto massimo è $p_{\max} = p_4 = 18$. Quindi

$$GR^e(I) = 40.$$

2. Il fattore di scala è

$$K = \frac{\varepsilon GR^e(I)}{n} = \frac{\frac{1}{4} \cdot 40}{4} = \frac{5}{2} = 2.5,$$

e i profitti scalati sono

$$p'_1 = \left\lfloor \frac{14}{\frac{5}{2}} \right\rfloor = 5, \quad p'_2 = \left\lfloor \frac{12}{\frac{5}{2}} \right\rfloor = 4, \quad p'_3 = \left\lfloor \frac{14}{\frac{5}{2}} \right\rfloor = 5, \quad p'_4 = \left\lfloor \frac{18}{\frac{5}{2}} \right\rfloor = 7.$$

Il profitto scalato totale è $P' = 5 + 4 + 5 + 7 = 21$.

3. I pesi minimi $u(j, q)$ di un sottoinsieme di $\{1, \dots, j\}$ di profitto scalato esattamente q sono (sono riportati solo i valori di q ottenibili; ∞ indica che q non si può ottenere con i primi j oggetti):

q	0	4	5	7	9	10	11	12	14	16	17
$u(0, q)$	0	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
$u(1, q)$	0	∞	3	∞	∞	∞	∞	∞	∞	∞	∞
$u(2, q)$	0	4	3	∞	7	∞	∞	∞	∞	∞	∞
$u(3, q)$	0	4	3	∞	7	8	∞	∞	12	∞	∞
$u(4, q)$	0	4	3	7	7	8	11	10	12	14	15

q	21
$u(0, q)$	∞
$u(1, q)$	∞
$u(2, q)$	∞
$u(3, q)$	∞
$u(4, q)$	19

Il massimo q con $u(4, q) \leq c = 16$ è $q = 17$, di peso $u(4, 17) = 15$, ottenuto con gli oggetti $\{1, 3, 4\}$.

4. FP^ε restituisce $S = \{1, 3, 4\}$, di profitto

$$FP^\varepsilon(I) = 14 + 14 + 18 = 46$$

nell'istanza originale, migliore di $GR^e(I) = 40$. La soluzione è **ottima**: $\tilde{S} = \{1, 3, 4\}$, di peso $3 + 5 + 7 = 15 \leq 16$ e profitto $z(I) = 14 + 14 + 18 = 46$, e il rapporto di prestazione è 1.

Osservazione. È l'istanza dell'esercizio Basic-FPTAS 2, risolta lì con $\varepsilon = \frac{1}{2}$. Dimezzare ε dimezza il fattore di scala K : i profitti scalati sono più fini, la tabella della programmazione dinamica è più lunga (P' cresce), e si trova la soluzione ottima.

Problemi di packing



Algoritmi di approssimazione per il bin packing

Proprietà usata in questi esercizi

La **proprietà di riempimento a metà** del capitolo [Algoritmi di approssimazione per il problema del bin packing](#): una soluzione ammissibile con schemi di carico $P_1, \dots, P_k$ la soddisfa quando al più uno schema di carico ha peso totale al più $c/2$, ossia

$$\left| \left\{ b \in \{1, \dots, k\} : \sum_{i \in P_b} w_i \leq \frac{c}{2} \right\} \right| \leq 1. \quad (\text{H})$$

36.1 Next Fit

Esercizio 36.1 — Next Fit 1

Consideriamo contenitori di capacità $c = 10$ e un insieme $\mathcal{J} = \{1, 2, 3, 4\}$ di 4 oggetti ($n = 4$), con i seguenti pesi:

	$i = 1$	$i = 2$	$i = 3$	$i = 4$
w_i	6	6	4	4

Domande

1. Applicare l'algoritmo Next Fit (NF) a questa istanza. Mostrare l'esecuzione passo per passo, indicando a ogni iterazione se l'oggetto è posto nel contenitore aperto o se si apre un nuovo contenitore, e la capacità residua che ne risulta. Riportare l'assegnamento finale ai contenitori e $NF(I)$.
2. Costruire gli schemi di carico $P_1, \dots, P_{NF(I)}$ a partire dal vettore dei contenitori ottenuto nella Domanda 1.

3. Calcolare un lower bound su $z(I)$ e stabilire se NF trova una soluzione ottima su questa istanza.
4. Verificare se gli schemi di carico trovati da NF soddisfano la proprietà di riempimento a metà (H), e stabilire se la soluzione trovata è ottima per questa istanza del BPP, confrontando $NF(I)$ con un lower bound su $z(I)$ oppure esibendo una soluzione ammissibile di valore strettamente minore.

Soluzione

1. L'algoritmo NF produce la seguente soluzione ammissibile:

- (a) L'oggetto 1 (peso 6) è posto nel contenitore 1. La capacità residua del contenitore 1 diventa $10 - 6 = 4$.
- (b) L'oggetto 2 (peso 6) non entra nel contenitore 1 (capacità residua 4). Quindi il contenitore 1 si chiude e l'oggetto 2 è posto in un nuovo contenitore 2. La capacità residua del contenitore 2 diventa $10 - 6 = 4$.
- (c) L'oggetto 3 (peso 4) è posto nel contenitore 2. La capacità residua del contenitore 2 diventa $4 - 4 = 0$.
- (d) L'oggetto 4 (peso 4) non entra nel contenitore 2 (capacità residua 0). Quindi il contenitore 2 si chiude e l'oggetto 4 è posto in un nuovo contenitore 3. La capacità residua del contenitore 3 diventa $10 - 4 = 6$.

Quindi NF restituisce:

$$bin = (1 \quad 2 \quad 2 \quad 3), \quad nb = 3.$$

e pertanto

$$NF(I) = nb = 3.$$

2. Dal vettore $bin = (1, 2, 2, 3)$, gli schemi di carico sono

$$\begin{aligned} P_1 &= \{i \in \{1, 2, 3, 4\} : bin_i = 1\} = \{1\}, \\ P_2 &= \{i \in \{1, 2, 3, 4\} : bin_i = 2\} = \{2, 3\}, \\ P_3 &= \{i \in \{1, 2, 3, 4\} : bin_i = 3\} = \{4\}. \end{aligned}$$

3. Di conseguenza, 3 è un upper bound sul numero ottimo di contenitori per questa istanza:

$$z(I) \leq 3.$$

Inoltre,

$$z(I) \geq \left\lceil \frac{\sum_{i \in \mathcal{I}} w_i}{c} \right\rceil = \left\lceil \frac{6 + 6 + 4 + 4}{10} \right\rceil = \left\lceil \frac{20}{10} \right\rceil = 2.$$

Inoltre, la seguente è una soluzione ammissibile del BPP:

$$\tilde{P}_1 = \{1, 3\}, \quad \tilde{P}_2 = \{2, 4\}$$

che usa 2 contenitori; quindi $z(I) \leq 2$. Pertanto $z(I) = 2$, e NF **non** calcola una soluzione ottima su questa istanza.

4. Dalla Domanda 2, gli schemi di carico trovati da NF hanno pesi totali

$$\sum_{i \in P_1} w_i = 6, \quad \sum_{i \in P_2} w_i = 6 + 4 = 10, \quad \sum_{i \in P_3} w_i = 4.$$

Poiché $c/2 = 5$, solo P_3 ha peso totale al più $c/2$ (gli altri due lo superano); quindi la soluzione **soddisfa** la proprietà di riempimento a metà.

Come mostrato nella Domanda 3, $NF(I) = 3$ mentre la soluzione ammissibile $\tilde{P}_1 = \{1, 3\}$, $\tilde{P}_2 = \{2, 4\}$ usa solo 2 contenitori; poiché questo valore è strettamente minore di $NF(I)$, la soluzione trovata da NF **non è ottima** (infatti $z(I) = 2 < NF(I) = 3$).

Esercizio 36.2 — Next Fit 2

Consideriamo contenitori di capacità $c = 10$ e un insieme $\mathcal{J} = \{1, 2, 3, 4\}$ di 4 oggetti ($n = 4$), con i seguenti pesi:

	$i = 1$	$i = 2$	$i = 3$	$i = 4$
w_i	7	7	3	3

Domande

1. Applicare l'algoritmo Next Fit (NF) a questa istanza. Mostrare l'esecuzione passo per passo, indicando a ogni iterazione se l'oggetto è posto nel contenitore aperto o se si apre un nuovo contenitore, e la capacità residua che ne risulta. Riportare l'assegnamento finale ai contenitori e $NF(I)$.
2. Costruire gli schemi di carico $P_1, \dots, P_{NF(I)}$ a partire dal vettore dei contenitori ottenuto nella Domanda 1.
3. Calcolare un lower bound su $z(I)$ e stabilire se NF trova una soluzione ottima su questa istanza.
4. Verificare se gli schemi di carico trovati da NF soddisfano la proprietà di riempimento a metà (H), e stabilire se la soluzione trovata è ottima per questa istanza del BPP, confrontando $NF(I)$ con un lower bound su $z(I)$ oppure esibendo una soluzione ammissibile di valore strettamente minore.

Soluzione

1. L'algoritmo NF produce la seguente soluzione ammissibile:
 - (a) L'oggetto 1 (peso 7) è posto nel contenitore 1. La capacità residua del contenitore 1 diventa $10 - 7 = 3$.
 - (b) L'oggetto 2 (peso 7) non entra nel contenitore 1 (capacità residua 3). Quindi il contenitore 1 si chiude e l'oggetto 2 è posto in un nuovo contenitore 2. La

capacità residua del contenitore 2 diventa $10 - 7 = 3$.

- (c) L'oggetto 3 (peso 3) è posto nel contenitore 2 (capacità residua 3). La capacità residua del contenitore 2 diventa $3 - 3 = 0$.
- (d) L'oggetto 4 (peso 3) non entra nel contenitore 2 (capacità residua 0). Quindi il contenitore 2 si chiude e l'oggetto 4 è posto in un nuovo contenitore 3. La capacità residua del contenitore 3 diventa $10 - 3 = 7$.

Quindi NF restituisce:

$$bin = (1 \ 2 \ 2 \ 3), \quad nb = 3.$$

e pertanto

$$NF(I) = nb = 3.$$

2. Dal vettore $bin = (1, 2, 2, 3)$, gli schemi di carico sono

$$\begin{aligned} P_1 &= \{i \in \{1, 2, 3, 4\} : bin_i = 1\} = \{1\}, \\ P_2 &= \{i \in \{1, 2, 3, 4\} : bin_i = 2\} = \{2, 3\}, \\ P_3 &= \{i \in \{1, 2, 3, 4\} : bin_i = 3\} = \{4\}. \end{aligned}$$

3. Di conseguenza, 3 è un upper bound sul numero ottimo di contenitori per questa istanza:

$$z(I) \leq 3.$$

Inoltre,

$$z(I) \geq \left\lceil \frac{\sum_{i \in \mathcal{I}} w_i}{c} \right\rceil = \left\lceil \frac{7 + 7 + 3 + 3}{10} \right\rceil = \left\lceil \frac{20}{10} \right\rceil = 2.$$

Inoltre, la seguente è una soluzione ammissibile del BPP:

$$\tilde{P}_1 = \{1, 3\}, \quad \tilde{P}_2 = \{2, 4\}$$

che usa 2 contenitori (poiché $w_1 + w_3 = 7 + 3 = 10$ e $w_2 + w_4 = 7 + 3 = 10$); quindi $z(I) \leq 2$. Pertanto $z(I) = 2$, e NF **non** calcola una soluzione ottima su questa istanza.

4. Dalla Domanda 2, gli schemi di carico trovati da NF hanno pesi totali

$$\sum_{i \in P_1} w_i = 7, \quad \sum_{i \in P_2} w_i = 7 + 3 = 10, \quad \sum_{i \in P_3} w_i = 3.$$

Poiché $c/2 = 5$, solo P_3 ha peso totale al più $c/2$ (gli altri due lo superano); quindi la soluzione **soddisfa** la proprietà di riempimento a metà.

Come mostrato nella Domanda 3, $NF(I) = 3$ mentre la soluzione ammissibile $\tilde{P}_1 = \{1, 3\}$, $\tilde{P}_2 = \{2, 4\}$ usa solo 2 contenitori; poiché questo valore è strettamente minore di $NF(I)$, la soluzione trovata da NF **non è ottima** (infatti $z(I) = 2 < NF(I) = 3$).

Esercizio 36.3 — Next Fit 3

Consideriamo contenitori di capacità $c = 10$ e un insieme $\mathcal{I} = \{1, 2, 3, 4, 5\}$ di 5 oggetti ($n = 5$), con i seguenti pesi:

	$i = 1$	$i = 2$	$i = 3$	$i = 4$	$i = 5$
w_i	4	7	6	3	5

Domande

1. Applicare l'algoritmo Next Fit (NF) a questa istanza. Mostrare l'esecuzione passo per passo, indicando a ogni iterazione se l'oggetto è posto nel contenitore aperto o se si apre un nuovo contenitore, e la capacità residua che ne risulta. Riportare l'assegnamento finale ai contenitori e $NF(I)$.
2. Costruire gli schemi di carico $P_1, \dots, P_{NF(I)}$ a partire dal vettore dei contenitori ottenuto nella Domanda 1.
3. Calcolare un lower bound su $z(I)$ e stabilire se NF trova una soluzione ottima su questa istanza.
4. Verificare se gli schemi di carico trovati da NF soddisfano la proprietà di riempimento a metà (H), e stabilire se la soluzione trovata è ottima per questa istanza del BPP, confrontando $NF(I)$ con un lower bound su $z(I)$ oppure esibendo una soluzione ammissibile di valore strettamente minore.

Soluzione

1. L'algoritmo NF produce la seguente soluzione ammissibile:
 - (a) L'oggetto 1 (peso 4) è posto nel contenitore 1. La capacità residua del contenitore 1 diventa $10 - 4 = 6$.
 - (b) L'oggetto 2 (peso 7) non entra nel contenitore 1 (capacità residua 6). Quindi il contenitore 1 si chiude e l'oggetto 2 è posto in un nuovo contenitore 2. La capacità residua del contenitore 2 diventa $10 - 7 = 3$.
 - (c) L'oggetto 3 (peso 6) non entra nel contenitore 2 (capacità residua 3). Quindi il contenitore 2 si chiude e l'oggetto 3 è posto in un nuovo contenitore 3. La capacità residua del contenitore 3 diventa $10 - 6 = 4$.
 - (d) L'oggetto 4 (peso 3) è posto nel contenitore 3 (capacità residua 4). La capacità residua del contenitore 3 diventa $4 - 3 = 1$.
 - (e) L'oggetto 5 (peso 5) non entra nel contenitore 3 (capacità residua 1). Quindi il contenitore 3 si chiude e l'oggetto 5 è posto in un nuovo contenitore 4. La capacità residua del contenitore 4 diventa $10 - 5 = 5$.

Quindi NF restituisce:

$$bin = (1 \ 2 \ 3 \ 3 \ 4), \quad nb = 4.$$

e pertanto

$$NF(I) = nb = 4.$$

2. Dal vettore $bin = (1, 2, 3, 3, 4)$, gli schemi di carico sono

$$\begin{aligned} P_1 &= \{i : bin_i = 1\} = \{1\}, \\ P_2 &= \{i : bin_i = 2\} = \{2\}, \\ P_3 &= \{i : bin_i = 3\} = \{3, 4\}, \\ P_4 &= \{i : bin_i = 4\} = \{5\}. \end{aligned}$$

3. Di conseguenza, 4 è un upper bound sul numero ottimo di contenitori per questa istanza:

$$z(I) \leq 4.$$

Inoltre,

$$z(I) \geq \left\lceil \frac{\sum_{i \in \mathcal{I}} w_i}{c} \right\rceil = \left\lceil \frac{4 + 7 + 6 + 3 + 5}{10} \right\rceil = \left\lceil \frac{25}{10} \right\rceil = 3.$$

Inoltre, la seguente è una soluzione ammissibile del BPP:

$$\tilde{P}_1 = \{2, 4\}, \quad \tilde{P}_2 = \{1, 3\}, \quad \tilde{P}_3 = \{5\}$$

che usa 3 contenitori (poiché $w_2 + w_4 = 7 + 3 = 10$ e $w_1 + w_3 = 4 + 6 = 10$); quindi $z(I) \leq 3$. Pertanto $z(I) = 3$, e NF **non** calcola una soluzione ottima su questa istanza.

4. Dalla Domanda 2, gli schemi di carico trovati da NF hanno pesi totali

$$\sum_{i \in P_1} w_i = 4, \quad \sum_{i \in P_2} w_i = 7, \quad \sum_{i \in P_3} w_i = 6 + 3 = 9, \quad \sum_{i \in P_4} w_i = 5.$$

Poiché $c/2 = 5$, sia P_1 (peso 4) sia P_4 (peso 5) hanno peso totale al più $c/2$; quindi la soluzione trovata da NF **non soddisfa** la proprietà di riempimento a metà (cioè è possibile proprio perché NF non è un algoritmo Any Fit).

Come mostrato nella Domanda 3, $NF(I) = 4$ mentre la soluzione ammissibile $\tilde{P}_1 = \{2, 4\}$, $\tilde{P}_2 = \{1, 3\}$, $\tilde{P}_3 = \{5\}$ usa solo 3 contenitori; poiché questo valore è strettamente minore di $NF(I)$, la soluzione trovata da NF **non è ottima** (infatti $z(I) = 3 < NF(I) = 4$).

36.2 First Fit

Esercizio 36.4 — First Fit 1

Consideriamo contenitori di capacità $c = 10$ e un insieme $\mathcal{I} = \{1, 2, 3, 4\}$ di 4 oggetti ($n = 4$), con i seguenti pesi:

	$i = 1$	$i = 2$	$i = 3$	$i = 4$
w_i	6	6	4	4

Domande

1. Applicare l'algoritmo First Fit (FF) a questa istanza. Mostrare l'esecuzione passo per passo, indicando a ogni iterazione in quale contenitore è posto l'oggetto (scorrendo i contenitori nell'ordine di apertura) e la capacità residua che ne risulta. Riportare l'assegnamento finale ai contenitori e $FF(I)$.
2. Costruire gli schemi di carico $P_1, \dots, P_{FF(I)}$ a partire dal vettore dei contenitori ottenuto nella Domanda 1.
3. Calcolare un lower bound su $z(I)$ e stabilire se FF trova una soluzione ottima su questa istanza.
4. Verificare se gli schemi di carico trovati da FF soddisfano la proprietà di riempimento a metà (H), e stabilire se la soluzione trovata è ottima per questa istanza del BPP, confrontando $FF(I)$ con un lower bound su $z(I)$ oppure esibendo una soluzione ammissibile di valore strettamente minore.

Soluzione

1. Ricordiamo lo pseudocodice di FF: per ogni oggetto i , $s := -1$ e, per $b := 1$ fino a nb , se $w[i] \leq rc[b]$ allora $s \leftarrow b$ e la scansione si ferma; se $s = -1$ dopo la scansione, si apre un nuovo contenitore.
 - (a) Oggetto 1 (peso 6): il contenitore 1 è aperto all'inizio ($nb = 1$, $rc[1] = c = 10$) e $w[1] \leq rc[1]$, quindi $s = 1$ e FF pone l'oggetto 1 nel contenitore 1, e $rc[1] \leftarrow 10 - 6 = 4$.
 - (b) Oggetto 2 (peso 6): $b = 1$: $w[2] = 6 \leq rc[1] = 4$? No. Nessun contenitore va bene, quindi FF apre il contenitore 2 ($rc[2] \leftarrow 10$). L'oggetto 2 è posto nel contenitore 2, e $rc[2] \leftarrow 10 - 6 = 4$.
 - (c) Oggetto 3 (peso 4): $b = 1$: $w[3] = 4 \leq rc[1] = 4$? Sì, quindi $s \leftarrow 1$ e la scansione si ferma. L'oggetto 3 è posto nel contenitore 1, e $rc[1] \leftarrow 4 - 4 = 0$.
 - (d) Oggetto 4 (peso 4): $b = 1$: $w[4] = 4 \leq rc[1] = 0$? No. $b = 2$: $w[4] = 4 \leq rc[2] = 4$? Sì, quindi $s \leftarrow 2$. L'oggetto 4 è posto nel contenitore 2, e $rc[2] \leftarrow 4 - 4 = 0$.

Quindi FF restituisce:

$$\text{bin} = (1 \ 2 \ 1 \ 2), \quad nb = 2.$$

e pertanto

$$FF(I) = nb = 2.$$

2. Dal vettore $\text{bin} = (1, 2, 1, 2)$, gli schemi di carico sono

$$P_1 = \{i \in \{1, 2, 3, 4\} : \text{bin}_i = 1\} = \{1, 3\},$$

$$P_2 = \{i \in \{1, 2, 3, 4\} : \text{bin}_i = 2\} = \{2, 4\}.$$

3. Di conseguenza,

$$z(I) \leq 2.$$

Inoltre,

$$z(I) \geq \left\lceil \frac{\sum_{i \in \mathcal{I}} w_i}{c} \right\rceil = \left\lceil \frac{6 + 6 + 4 + 4}{10} \right\rceil = \left\lceil \frac{20}{10} \right\rceil = 2.$$

Quindi $z(I) = 2$ e FF calcola una soluzione **ottima** su questa istanza.

4. Dalla Domanda 2, gli schemi di carico trovati da FF hanno pesi totali

$$\sum_{i \in P_1} w_i = 6 + 4 = 10, \quad \sum_{i \in P_2} w_i = 6 + 4 = 10.$$

Poiché $c/2 = 5$, nessuno schema di carico ha peso totale al più $c/2$; in particolare *al più uno* lo ha, quindi la soluzione **soddisfa** la proprietà di riempimento a metà.

Poiché $FF(I) = 2$ coincide con il lower bound $z(I) \geq 2$ trovato nella Domanda 3, la soluzione trovata da FF è **ottima**.

Esercizio 36.5 — First Fit 2

Consideriamo contenitori di capacità $c = 10$ e un insieme $\mathcal{I} = \{1, 2, 3, 4\}$ di 4 oggetti ($n = 4$), con i seguenti pesi:

	$i = 1$	$i = 2$	$i = 3$	$i = 4$
w_i	7	7	3	3

Domande

1. Applicare l'algoritmo First Fit (FF) a questa istanza. Mostrare l'esecuzione passo per passo, indicando a ogni iterazione in quale contenitore è posto l'oggetto (scorrendo i contenitori nell'ordine di apertura) e la capacità residua che ne risulta. Riportare l'assegnamento finale ai contenitori e $FF(I)$.
2. Costruire gli schemi di carico $P_1, \dots, P_{FF(I)}$ a partire dal vettore dei contenitori ottenuto nella Domanda 1.

3. Calcolare un lower bound su $z(I)$ e stabilire se FF trova una soluzione ottima su questa istanza.
4. Verificare se gli schemi di carico trovati da FF soddisfano la proprietà di riempimento a metà (H), e stabilire se la soluzione trovata è ottima per questa istanza del BPP, confrontando $FF(I)$ con un lower bound su $z(I)$ oppure esibendo una soluzione ammissibile di valore strettamente minore.

Soluzione

1. Ricordiamo lo pseudocodice di FF: per ogni oggetto i , $s := -1$ e, per $b := 1$ fino a nb , se $w[i] \leq rc[b]$ allora $s \leftarrow b$ e la scansione si ferma; se $s = -1$ dopo la scansione, si apre un nuovo contenitore.
 - (a) Oggetto 1 (peso 7): il contenitore 1 è aperto all'inizio ($nb = 1$, $rc[1] = c = 10$) e $w[1] \leq rc[1]$, quindi $s = 1$ e FF pone l'oggetto 1 nel contenitore 1, e $rc[1] \leftarrow 10 - 7 = 3$.
 - (b) Oggetto 2 (peso 7): $b = 1$: $w[2] = 7 \leq rc[1] = 3$? No. Nessun contenitore va bene, quindi FF apre il contenitore 2 ($rc[2] \leftarrow 10$). L'oggetto 2 è posto nel contenitore 2, e $rc[2] \leftarrow 10 - 7 = 3$.
 - (c) Oggetto 3 (peso 3): $b = 1$: $w[3] = 3 \leq rc[1] = 3$? Sì, quindi $s \leftarrow 1$ e la scansione si ferma. L'oggetto 3 è posto nel contenitore 1, e $rc[1] \leftarrow 3 - 3 = 0$.
 - (d) Oggetto 4 (peso 3): $b = 1$: $w[4] = 3 \leq rc[1] = 0$? No. $b = 2$: $w[4] = 3 \leq rc[2] = 3$? Sì, quindi $s \leftarrow 2$. L'oggetto 4 è posto nel contenitore 2, e $rc[2] \leftarrow 3 - 3 = 0$.

Quindi FF restituisce:

$$bin = (1 \quad 2 \quad 1 \quad 2), \quad nb = 2.$$

e pertanto

$$FF(I) = nb = 2.$$

2. Dal vettore $bin = (1, 2, 1, 2)$, gli schemi di carico sono

$$P_1 = \{i \in \{1, 2, 3, 4\} : bin_i = 1\} = \{1, 3\},$$

$$P_2 = \{i \in \{1, 2, 3, 4\} : bin_i = 2\} = \{2, 4\}.$$

3. Di conseguenza,

$$z(I) \leq 2.$$

Inoltre,

$$z(I) \geq \left\lceil \frac{\sum_{i \in \mathcal{I}} w_i}{c} \right\rceil = \left\lceil \frac{7 + 7 + 3 + 3}{10} \right\rceil = \left\lceil \frac{20}{10} \right\rceil = 2.$$

Quindi $z(I) = 2$ e FF calcola una soluzione **ottima** su questa istanza.

4. Dalla Domanda 2, gli schemi di carico trovati da FF hanno pesi totali

$$\sum_{i \in P_1} w_i = 7 + 3 = 10, \quad \sum_{i \in P_2} w_i = 7 + 3 = 10.$$

Poiché $c/2 = 5$, nessuno schema di carico ha peso totale al più $c/2$; in particolare *al più uno* lo ha, quindi la soluzione **soddisfa** la proprietà di riempimento a metà.

Poiché $FF(I) = 2$ coincide con il lower bound $z(I) \geq 2$ trovato nella Domanda 3, la soluzione trovata da FF è **ottima**.

Esercizio 36.6 — First Fit 3

Consideriamo contenitori di capacità $c = 10$ e un insieme $\mathcal{I} = \{1, 2, 3, 4, 5\}$ di 5 oggetti ($n = 5$), con i seguenti pesi:

	$i = 1$	$i = 2$	$i = 3$	$i = 4$	$i = 5$
w_i	4	7	6	3	5

Domande

1. Applicare l'algoritmo First Fit (FF) a questa istanza. Mostrare l'esecuzione passo per passo, indicando a ogni iterazione in quale contenitore è posto l'oggetto (scorrendo i contenitori nell'ordine di apertura) e la capacità residua che ne risulta. Riportare l'assegnamento finale ai contenitori e $FF(I)$.
2. Costruire gli schemi di carico $P_1, \dots, P_{FF(I)}$ a partire dal vettore dei contenitori ottenuto nella Domanda 1.
3. Calcolare un lower bound su $z(I)$ e stabilire se FF trova una soluzione ottima su questa istanza.
4. Verificare se gli schemi di carico trovati da FF soddisfano la proprietà di riempimento a metà (H), e stabilire se la soluzione trovata è ottima per questa istanza del BPP, confrontando $FF(I)$ con un lower bound su $z(I)$ oppure esibendo una soluzione ammissibile di valore strettamente minore.

Soluzione

1. Ricordiamo lo pseudocodice di FF: per ogni oggetto i , $s := -1$ e, per $b := 1$ fino a nb , se $w[i] \leq rc[b]$ allora $s \leftarrow b$ e la scansione si ferma; se $s = -1$ dopo la scansione, si apre un nuovo contenitore.
 - (a) Oggetto 1 (peso 4): il contenitore 1 è aperto all'inizio ($nb = 1$, $rc[1] = c = 10$) e $w[1] \leq rc[1]$, quindi $s = 1$ e FF pone l'oggetto 1 nel contenitore 1, e $rc[1] \leftarrow 10 - 4 = 6$.

- (b) Oggetto 2 (peso 7): $b = 1$: $w[2] = 7 \leq rc[1] = 6$? No. Nessun contenitore va bene, quindi FF apre il contenitore 2 ($rc[2] \leftarrow 10$). L'oggetto 2 è posto nel contenitore 2, e $rc[2] \leftarrow 10 - 7 = 3$.
- (c) Oggetto 3 (peso 6): $b = 1$: $w[3] = 6 \leq rc[1] = 6$? Sì, quindi $s \leftarrow 1$ e la scansione si ferma. L'oggetto 3 è posto nel contenitore 1, e $rc[1] \leftarrow 6 - 6 = 0$.
- (d) Oggetto 4 (peso 3): $b = 1$: $w[4] = 3 \leq rc[1] = 0$? No. $b = 2$: $w[4] = 3 \leq rc[2] = 3$? Sì, quindi $s \leftarrow 2$. L'oggetto 4 è posto nel contenitore 2, e $rc[2] \leftarrow 3 - 3 = 0$.
- (e) Oggetto 5 (peso 5): $b = 1$: $w[5] = 5 \leq rc[1] = 0$? No. $b = 2$: $w[5] = 5 \leq rc[2] = 0$? No. Nessun contenitore va bene, quindi FF apre il contenitore 3 ($rc[3] \leftarrow 10$). L'oggetto 5 è posto nel contenitore 3, e $rc[3] \leftarrow 10 - 5 = 5$.

Quindi FF restituisce:

$$bin = (1 \quad 2 \quad 1 \quad 2 \quad 3), \quad nb = 3.$$

e pertanto

$$FF(I) = nb = 3.$$

2. Dal vettore $bin = (1, 2, 1, 2, 3)$, gli schemi di carico sono

$$P_1 = \{i : bin_i = 1\} = \{1, 3\},$$

$$P_2 = \{i : bin_i = 2\} = \{2, 4\},$$

$$P_3 = \{i : bin_i = 3\} = \{5\}.$$

3. Di conseguenza,

$$z(I) \leq 3.$$

Inoltre,

$$z(I) \geq \left\lceil \frac{\sum_{i \in \mathcal{I}} w_i}{c} \right\rceil = \left\lceil \frac{4 + 7 + 6 + 3 + 5}{10} \right\rceil = \left\lceil \frac{25}{10} \right\rceil = 3.$$

Quindi $z(I) = 3$ e FF calcola una soluzione **ottima** su questa istanza.

4. Dalla Domanda 2, gli schemi di carico trovati da FF hanno pesi totali

$$\sum_{i \in P_1} w_i = 4 + 6 = 10, \quad \sum_{i \in P_2} w_i = 7 + 3 = 10, \quad \sum_{i \in P_3} w_i = 5.$$

Poiché $c/2 = 5$, solo P_3 ha peso totale al più $c/2$ (gli altri due lo superano); quindi la soluzione **soddisfa** la proprietà di riempimento a metà.

Poiché $FF(I) = 3$ coincide con il lower bound $z(I) \geq 3$ trovato nella Domanda 3, la soluzione trovata da FF è **ottima**.

36.3 Best Fit

Esercizio 36.7 — Best Fit 1

Consideriamo contenitori di capacità $c = 10$ e un insieme $\mathcal{I} = \{1, 2, 3, 4\}$ di 4 oggetti ($n = 4$), con i seguenti pesi:

	$i = 1$	$i = 2$	$i = 3$	$i = 4$
w_i	6	6	4	4

Domande

1. Applicare l'algoritmo Best Fit (BF) a questa istanza. Mostrare l'esecuzione passo per passo: per ogni oggetto, seguire le variabili $minrc$ e s dello pseudocodice durante la scansione dei contenitori aperti, e indicare in quale contenitore è posto l'oggetto e la capacità residua che ne risulta.
2. Costruire gli schemi di carico $P_1, \dots, P_{BF(I)}$ a partire dal vettore dei contenitori ottenuto nella Domanda 1.
3. Calcolare un lower bound su $z(I)$ e stabilire se BF trova una soluzione ottima su questa istanza.
4. Verificare se gli schemi di carico trovati da BF soddisfano la proprietà di riempimento a metà (H), e stabilire se la soluzione trovata è ottima per questa istanza del BPP, confrontando $BF(I)$ con un lower bound su $z(I)$ oppure esibendo una soluzione ammissibile di valore strettamente minore.

Soluzione

1. Ricordiamo lo pseudocodice di BF: per ogni oggetto, $minrc := c + 1$, $s := -1$, e per $b := 1$ fino a nb , se $w[i] \leq rc[b]$ e $rc[b] < minrc$ allora $s \leftarrow b$, $minrc \leftarrow rc[b]$.
 - (a) Oggetto 1 (peso 6): il contenitore 1 è aperto all'inizio ($nb = 1$, $rc[1] = c$) e l'oggetto 1 vi entra, quindi $s = 1$ e BF pone l'oggetto 1 nel contenitore 1. Capacità residua del contenitore 1: $10 - 6 = 4$.
 - (b) Oggetto 2 (peso 6): $minrc := 11$, $s := -1$. Per $b = 1$: $w[2] = 6 \leq rc[1] = 4$? No. Nessun contenitore va bene, quindi BF apre il contenitore 2 e vi pone l'oggetto 2. Capacità residua del contenitore 2: $10 - 6 = 4$.
 - (c) Oggetto 3 (peso 4): $minrc := 11$, $s := -1$. Per $b = 1$: $w[3] = 4 \leq rc[1] = 4$ e $rc[1] = 4 < 11$, quindi $s \leftarrow 1$, $minrc \leftarrow 4$. Per $b = 2$: $w[3] = 4 \leq rc[2] = 4$, ma $rc[2] = 4$ non è strettamente minore di $minrc = 4$, quindi s non si aggiorna. Quindi $s = 1$: l'oggetto 3 è posto nel contenitore 1 (il primo aperto, perché il pareggio fra il contenitore 1 e il contenitore 2 è risolto dalla disuguaglianza stretta dello pseudocodice). Capacità residua del contenitore 1: $4 - 4 = 0$.

- (d) Oggetto 4 (peso 4): $\text{minrc} := 11$, $s := -1$. Per $b = 1$: $w[4] = 4 \leq \text{rc}[1] = 0$? No. Per $b = 2$: $w[4] = 4 \leq \text{rc}[2] = 4$ e $\text{rc}[2] = 4 < 11$, quindi $s \leftarrow 2$, $\text{minrc} \leftarrow 4$. Quindi $s = 2$: l'oggetto 4 è posto nel contenitore 2. Capacità residua del contenitore 2: $4 - 4 = 0$.

Quindi BF restituisce:

$$\text{bin} = (1 \quad 2 \quad 1 \quad 2), \quad nb = 2.$$

e pertanto

$$BF(I) = nb = 2.$$

2. Dal vettore $\text{bin} = (1, 2, 1, 2)$, gli schemi di carico sono

$$P_1 = \{i \in \{1, 2, 3, 4\} : \text{bin}_i = 1\} = \{1, 3\},$$

$$P_2 = \{i \in \{1, 2, 3, 4\} : \text{bin}_i = 2\} = \{2, 4\}.$$

3. Di conseguenza,

$$z(I) \leq 2.$$

Inoltre,

$$z(I) \geq \left\lceil \frac{\sum_{i \in \mathcal{I}} w_i}{c} \right\rceil = \left\lceil \frac{6 + 6 + 4 + 4}{10} \right\rceil = \left\lceil \frac{20}{10} \right\rceil = 2.$$

Quindi $z(I) = 2$ e BF calcola una soluzione **ottima** su questa istanza.

4. Dalla Domanda 2, gli schemi di carico trovati da BF hanno pesi totali

$$\sum_{i \in P_1} w_i = 6 + 4 = 10, \quad \sum_{i \in P_2} w_i = 6 + 4 = 10.$$

Poiché $c/2 = 5$, nessuno schema di carico ha peso totale al più $c/2$; in particolare *al più uno* lo ha, quindi la soluzione **soddisfa** la proprietà di riempimento a metà.

Poiché $BF(I) = 2$ coincide con il lower bound $z(I) \geq 2$ trovato nella Domanda 3, la soluzione trovata da BF è **ottima**.

Esercizio 36.8 — Best Fit 2

Consideriamo contenitori di capacità $c = 10$ e un insieme $\mathcal{I} = \{1, 2, 3, 4\}$ di 4 oggetti ($n = 4$), con i seguenti pesi:

	$i = 1$	$i = 2$	$i = 3$	$i = 4$
w_i	7	7	3	3

Domande

1. Applicare l'algoritmo Best Fit (BF) a questa istanza. Mostrare l'esecuzione passo per passo: per ogni oggetto, seguire le variabili $minrc$ e s dello pseudocodice durante la scansione dei contenitori aperti, e indicare in quale contenitore è posto l'oggetto e la capacità residua che ne risulta.
2. Costruire gli schemi di carico $P_1, \dots, P_{BF(I)}$ a partire dal vettore dei contenitori ottenuto nella Domanda 1.
3. Calcolare un lower bound su $z(I)$ e stabilire se BF trova una soluzione ottima su questa istanza.
4. Verificare se gli schemi di carico trovati da BF soddisfano la proprietà di riempimento a metà (H), e stabilire se la soluzione trovata è ottima per questa istanza del BPP, confrontando $BF(I)$ con un lower bound su $z(I)$ oppure esibendo una soluzione ammissibile di valore strettamente minore.

Soluzione

1. Ricordiamo lo pseudocodice di BF: per ogni oggetto i , $minrc := c + 1$, $s := -1$, e per $b := 1$ fino a nb , se $w[i] \leq rc[b]$ e $rc[b] < minrc$ allora $s \leftarrow b$, $minrc \leftarrow rc[b]$.
 - (a) Oggetto 1 (peso 7): il contenitore 1 è aperto all'inizio ($nb = 1$, $rc[1] = c$) e l'oggetto 1 vi entra, quindi $s = 1$ e BF pone l'oggetto 1 nel contenitore 1. Capacità residua del contenitore 1: $10 - 7 = 3$.
 - (b) Oggetto 2 (peso 7): $minrc := 11$, $s := -1$. Per $b = 1$: $w[2] = 7 \leq rc[1] = 3$? No. Nessun contenitore va bene, quindi BF apre il contenitore 2 e vi pone l'oggetto 2. Capacità residua del contenitore 2: $10 - 7 = 3$.
 - (c) Oggetto 3 (peso 3): $minrc := 11$, $s := -1$. Per $b = 1$: $w[3] = 3 \leq rc[1] = 3$ e $rc[1] = 3 < 11$, quindi $s \leftarrow 1$, $minrc \leftarrow 3$. Per $b = 2$: $w[3] = 3 \leq rc[2] = 3$, ma $rc[2] = 3$ non è strettamente minore di $minrc = 3$, quindi s non si aggiorna. Quindi $s = 1$: l'oggetto 3 è posto nel contenitore 1 (il primo aperto, perché il pareggio fra il contenitore 1 e il contenitore 2 è risolto dalla disuguaglianza stretta dello pseudocodice). Capacità residua del contenitore 1: $3 - 3 = 0$.
 - (d) Oggetto 4 (peso 3): $minrc := 11$, $s := -1$. Per $b = 1$: $w[4] = 3 \leq rc[1] = 0$? No. Per $b = 2$: $w[4] = 3 \leq rc[2] = 3$ e $rc[2] = 3 < 11$, quindi $s \leftarrow 2$, $minrc \leftarrow 3$. Quindi $s = 2$: l'oggetto 4 è posto nel contenitore 2. Capacità residua del contenitore 2: $3 - 3 = 0$.

Quindi BF restituisce:

$$bin = (1 \quad 2 \quad 1 \quad 2), \quad nb = 2.$$

e pertanto

$$BF(I) = nb = 2.$$

2. Dal vettore $bin = (1, 2, 1, 2)$, gli schemi di carico sono

$$P_1 = \{i \in \{1, 2, 3, 4\} : bin_i = 1\} = \{1, 3\},$$

$$P_2 = \{i \in \{1, 2, 3, 4\} : bin_i = 2\} = \{2, 4\}.$$

3. Di conseguenza,

$$z(I) \leq 2.$$

Inoltre,

$$z(I) \geq \left\lceil \frac{\sum_{i \in \mathcal{I}} w_i}{c} \right\rceil = \left\lceil \frac{7 + 7 + 3 + 3}{10} \right\rceil = \left\lceil \frac{20}{10} \right\rceil = 2.$$

Quindi $z(I) = 2$ e BF calcola una soluzione **ottima** su questa istanza.

4. Dalla Domanda 2, gli schemi di carico trovati da BF hanno pesi totali

$$\sum_{i \in P_1} w_i = 7 + 3 = 10, \quad \sum_{i \in P_2} w_i = 7 + 3 = 10.$$

Poiché $c/2 = 5$, nessuno schema di carico ha peso totale al più $c/2$; in particolare *al più uno* lo ha, quindi la soluzione **soddisfa** la proprietà di riempimento a metà.

Poiché $BF(I) = 2$ coincide con il lower bound $z(I) \geq 2$ trovato nella Domanda 3, la soluzione trovata da BF è **ottima**.

Esercizio 36.9 — Best Fit 3

Consideriamo contenitori di capacità $c = 10$ e un insieme $\mathcal{I} = \{1, 2, 3, 4, 5\}$ di 5 oggetti ($n = 5$), con i seguenti pesi:

	$i = 1$	$i = 2$	$i = 3$	$i = 4$	$i = 5$
w_i	4	7	6	3	5

Domande

1. Applicare l'algoritmo Best Fit (BF) a questa istanza. Mostrare l'esecuzione passo per passo: per ogni oggetto, seguire le variabili $minrc$ e s dello pseudocodice durante la scansione dei contenitori aperti, e indicare in quale contenitore è posto l'oggetto e la capacità residua che ne risulta.
2. Costruire gli schemi di carico $P_1, \dots, P_{BF(I)}$ a partire dal vettore dei contenitori ottenuto nella Domanda 1.
3. Calcolare un lower bound su $z(I)$ e stabilire se BF trova una soluzione ottima su questa istanza.
4. Verificare se gli schemi di carico trovati da BF soddisfano la proprietà di riempimento a metà (H), e stabilire se la soluzione trovata è ottima per questa istanza del BPP, confrontando $BF(I)$ con un lower bound su $z(I)$ oppure esibendo una soluzione ammissibile di valore strettamente minore.

Soluzione

1. Ricordiamo lo pseudocodice di BF: per ogni oggetto i , $minrc := c + 1$, $s := -1$, e per $b := 1$ fino a nb , se $w[i] \leq rc[b]$ e $rc[b] < minrc$ allora $s \leftarrow b$, $minrc \leftarrow rc[b]$.
 - (a) Oggetto 1 (peso 4): il contenitore 1 è aperto all'inizio ($nb = 1$, $rc[1] = c$) e l'oggetto 1 vi entra, quindi $s = 1$ e BF pone l'oggetto 1 nel contenitore 1. Capacità residua del contenitore 1: $10 - 4 = 6$.
 - (b) Oggetto 2 (peso 7): $minrc := 11$, $s := -1$. Per $b = 1$: $w[2] = 7 \leq rc[1] = 6$? No. Nessun contenitore va bene, quindi BF apre il contenitore 2 e vi pone l'oggetto 2. Capacità residua del contenitore 2: $10 - 7 = 3$.
 - (c) Oggetto 3 (peso 6): $minrc := 11$, $s := -1$. Per $b = 1$: $w[3] = 6 \leq rc[1] = 6$ e $rc[1] = 6 < 11$, quindi $s \leftarrow 1$, $minrc \leftarrow 6$. Per $b = 2$: $w[3] = 6 \leq rc[2] = 3$? No. Quindi $s = 1$: l'oggetto 3 è posto nel contenitore 1. Capacità residua del contenitore 1: $6 - 6 = 0$.
 - (d) Oggetto 4 (peso 3): $minrc := 11$, $s := -1$. Per $b = 1$: $w[4] = 3 \leq rc[1] = 0$? No. Per $b = 2$: $w[4] = 3 \leq rc[2] = 3$ e $rc[2] = 3 < 11$, quindi $s \leftarrow 2$, $minrc \leftarrow 3$. Quindi $s = 2$: l'oggetto 4 è posto nel contenitore 2. Capacità residua del contenitore 2: $3 - 3 = 0$.
 - (e) Oggetto 5 (peso 5): $minrc := 11$, $s := -1$. Per $b = 1$: $w[5] = 5 \leq rc[1] = 0$? No. Per $b = 2$: $w[5] = 5 \leq rc[2] = 0$? No. Nessun contenitore va bene, quindi BF apre il contenitore 3 e vi pone l'oggetto 5. Capacità residua del contenitore 3: $10 - 5 = 5$.

Quindi BF restituisce:

$$bin = (1 \quad 2 \quad 1 \quad 2 \quad 3), \quad nb = 3.$$

e pertanto

$$BF(I) = nb = 3.$$

2. Dal vettore $bin = (1, 2, 1, 2, 3)$, gli schemi di carico sono

$$P_1 = \{i : bin_i = 1\} = \{1, 3\},$$

$$P_2 = \{i : bin_i = 2\} = \{2, 4\},$$

$$P_3 = \{i : bin_i = 3\} = \{5\}.$$

3. Di conseguenza,

$$z(I) \leq 3.$$

Inoltre,

$$z(I) \geq \left\lceil \frac{\sum_{i \in \mathcal{I}} w_i}{c} \right\rceil = \left\lceil \frac{4 + 7 + 6 + 3 + 5}{10} \right\rceil = \left\lceil \frac{25}{10} \right\rceil = 3.$$

Quindi $z(I) = 3$ e BF calcola una soluzione **ottima** su questa istanza.

4. Dalla Domanda 2, gli schemi di carico trovati da BF hanno pesi totali

$$\sum_{i \in P_1} w_i = 4 + 6 = 10, \quad \sum_{i \in P_2} w_i = 7 + 3 = 10, \quad \sum_{i \in P_3} w_i = 5.$$

Poiché $c/2 = 5$, solo P_3 ha peso totale al più $c/2$ (gli altri due lo superano); quindi la soluzione **soddisfa** la proprietà di riempimento a metà.

Poiché $BF(I) = 3$ coincide con il lower bound $z(I) \geq 3$ trovato nella Domanda 3, la soluzione trovata da BF è **ottima**.

36.4 Worst Fit

Esercizio 36.10 — Worst Fit 1

Consideriamo contenitori di capacità $c = 10$ e un insieme $\mathcal{J} = \{1, 2, 3, 4\}$ di 4 oggetti ($n = 4$), con i seguenti pesi:

	$i = 1$	$i = 2$	$i = 3$	$i = 4$
w_i	6	6	4	4

Domande

1. Applicare l'algoritmo Worst Fit (WF) a questa istanza. Mostrare l'esecuzione passo per passo: per ogni oggetto, seguire le variabili $maxrc$ e s dello pseudocodice durante la scansione dei contenitori aperti, e indicare in quale contenitore è posto l'oggetto e la capacità residua che ne risulta.
2. Costruire gli schemi di carico $P_1, \dots, P_{WF(I)}$ a partire dal vettore dei contenitori ottenuto nella Domanda 1.
3. Calcolare un lower bound su $z(I)$ e stabilire se WF trova una soluzione ottima su questa istanza.
4. Verificare se gli schemi di carico trovati da WF soddisfano la proprietà di riempimento a metà (H), e stabilire se la soluzione trovata è ottima per questa istanza del BPP, confrontando $WF(I)$ con un lower bound su $z(I)$ oppure esibendo una soluzione ammissibile di valore strettamente minore.

Soluzione

1. Ricordiamo lo pseudocodice di WF: per ogni oggetto i , $maxrc := -1$, $s := -1$, e per $b := 1$ fino a nb , se $w[i] \leq rc[b]$ e $rc[b] > maxrc$ allora $s \leftarrow b$, $maxrc \leftarrow rc[b]$.
 - (a) Oggetto 1 (peso 6): il contenitore 1 è aperto all'inizio ($nb = 1$, $rc[1] = c$) e

l'oggetto 1 vi entra, quindi $s = 1$ e WF pone l'oggetto 1 nel contenitore 1. Capacità residua del contenitore 1: $10 - 6 = 4$.

(b) Oggetto 2 (peso 6): $maxrc := -1$, $s := -1$. Per $b = 1$: $w[2] = 6 \leq rc[1] = 4$? No. Nessun contenitore va bene, quindi WF apre il contenitore 2 e vi pone l'oggetto 2. Capacità residua del contenitore 2: $10 - 6 = 4$.

(c) Oggetto 3 (peso 4): $maxrc := -1$, $s := -1$. Per $b = 1$: $w[3] = 4 \leq rc[1] = 4$ e $rc[1] = 4 > -1$, quindi $s \leftarrow 1$, $maxrc \leftarrow 4$. Per $b = 2$: $w[3] = 4 \leq rc[2] = 4$, ma $rc[2] = 4$ non è strettamente maggiore di $maxrc = 4$, quindi s non si aggiorna. Quindi $s = 1$: l'oggetto 3 è posto nel contenitore 1 (il primo aperto, perché il pareggio fra il contenitore 1 e il contenitore 2 è risolto dalla disuguaglianza stretta dello pseudocodice). Capacità residua del contenitore 1: $4 - 4 = 0$.

(d) Oggetto 4 (peso 4): $maxrc := -1$, $s := -1$. Per $b = 1$: $w[4] = 4 \leq rc[1] = 0$? No. Per $b = 2$: $w[4] = 4 \leq rc[2] = 4$ e $rc[2] = 4 > -1$, quindi $s \leftarrow 2$, $maxrc \leftarrow 4$. Quindi $s = 2$: l'oggetto 4 è posto nel contenitore 2. Capacità residua del contenitore 2: $4 - 4 = 0$.

Quindi WF restituisce:

$$bin = (1 \ 2 \ 1 \ 2), \quad nb = 2.$$

e pertanto

$$WF(I) = nb = 2.$$

2. Dal vettore $bin = (1, 2, 1, 2)$, gli schemi di carico sono

$$\begin{aligned} P_1 &= \{i \in \{1, 2, 3, 4\} : bin_i = 1\} = \{1, 3\}, \\ P_2 &= \{i \in \{1, 2, 3, 4\} : bin_i = 2\} = \{2, 4\}. \end{aligned}$$

3. Di conseguenza,

$$z(I) \leq 2.$$

Inoltre,

$$z(I) \geq \left\lceil \frac{\sum_{i \in \mathcal{I}} w_i}{c} \right\rceil = \left\lceil \frac{6 + 6 + 4 + 4}{10} \right\rceil = \left\lceil \frac{20}{10} \right\rceil = 2.$$

Quindi $z(I) = 2$ e WF calcola una soluzione **ottima** su questa istanza.

4. Dalla Domanda 2, gli schemi di carico trovati da WF hanno pesi totali

$$\sum_{i \in P_1} w_i = 6 + 4 = 10, \quad \sum_{i \in P_2} w_i = 6 + 4 = 10.$$

Poiché $c/2 = 5$, nessuno schema di carico ha peso totale al più $c/2$; in particolare *al più uno* lo ha, quindi la soluzione **soddisfa** la proprietà di riempimento a metà.

Poiché $WF(I) = 2$ coincide con il lower bound $z(I) \geq 2$ trovato nella Domanda 3, la soluzione trovata da WF è **ottima**.

Esercizio 36.11 — Worst Fit 2

Consideriamo contenitori di capacità $c = 10$ e un insieme $\mathcal{I} = \{1, 2, 3, 4\}$ di 4 oggetti ($n = 4$), con i seguenti pesi:

	$i = 1$	$i = 2$	$i = 3$	$i = 4$
w_i	7	7	3	3

Domande

1. Applicare l'algoritmo Worst Fit (WF) a questa istanza. Mostrare l'esecuzione passo per passo: per ogni oggetto, seguire le variabili $maxrc$ e s dello pseudocodice durante la scansione dei contenitori aperti, e indicare in quale contenitore è posto l'oggetto e la capacità residua che ne risulta.
2. Costruire gli schemi di carico $P_1, \dots, P_{WF(I)}$ a partire dal vettore dei contenitori ottenuto nella Domanda 1.
3. Calcolare un lower bound su $z(I)$ e stabilire se WF trova una soluzione ottima su questa istanza.
4. Verificare se gli schemi di carico trovati da WF soddisfano la proprietà di riempimento a metà (H), e stabilire se la soluzione trovata è ottima per questa istanza del BPP, confrontando $WF(I)$ con un lower bound su $z(I)$ oppure esibendo una soluzione ammissibile di valore strettamente minore.

Soluzione

1. Ricordiamo lo pseudocodice di WF: per ogni oggetto i , $maxrc := -1$, $s := -1$, e per $b := 1$ fino a nb , se $w[i] \leq rc[b]$ e $rc[b] > maxrc$ allora $s \leftarrow b$, $maxrc \leftarrow rc[b]$.
 - (a) Oggetto 1 (peso 7): il contenitore 1 è aperto all'inizio ($nb = 1$, $rc[1] = c$) e l'oggetto 1 vi entra, quindi $s = 1$ e WF pone l'oggetto 1 nel contenitore 1. Capacità residua del contenitore 1: $10 - 7 = 3$.
 - (b) Oggetto 2 (peso 7): $maxrc := -1$, $s := -1$. Per $b = 1$: $w[2] = 7 \leq rc[1] = 3$? No. Nessun contenitore va bene, quindi WF apre il contenitore 2 e vi pone l'oggetto 2. Capacità residua del contenitore 2: $10 - 7 = 3$.
 - (c) Oggetto 3 (peso 3): $maxrc := -1$, $s := -1$. Per $b = 1$: $w[3] = 3 \leq rc[1] = 3$ e $rc[1] = 3 > -1$, quindi $s \leftarrow 1$, $maxrc \leftarrow 3$. Per $b = 2$: $w[3] = 3 \leq rc[2] = 3$, ma $rc[2] = 3$ non è strettamente maggiore di $maxrc = 3$, quindi s non si aggiorna. Quindi $s = 1$: l'oggetto 3 è posto nel contenitore 1 (il primo aperto, perché il pareggio fra il contenitore 1 e il contenitore 2 è risolto dalla disuguaglianza stretta dello pseudocodice). Capacità residua del contenitore 1: $3 - 3 = 0$.
 - (d) Oggetto 4 (peso 3): $maxrc := -1$, $s := -1$. Per $b = 1$: $w[4] = 3 \leq rc[1] = 0$? No. Per $b = 2$: $w[4] = 3 \leq rc[2] = 3$ e $rc[2] = 3 > -1$, quindi $s \leftarrow 2$,

$maxrc \leftarrow 3$. Quindi $s = 2$: l'oggetto 4 è posto nel contenitore 2. Capacità residua del contenitore 2: $3 - 3 = 0$.

Quindi WF restituisce:

$$bin = (1 \quad 2 \quad 1 \quad 2), \quad nb = 2.$$

e pertanto

$$WF(I) = nb = 2.$$

2. Dal vettore $bin = (1, 2, 1, 2)$, gli schemi di carico sono

$$P_1 = \{i \in \{1, 2, 3, 4\} : bin_i = 1\} = \{1, 3\},$$

$$P_2 = \{i \in \{1, 2, 3, 4\} : bin_i = 2\} = \{2, 4\}.$$

3. Di conseguenza,

$$z(I) \leq 2.$$

Inoltre,

$$z(I) \geq \left\lceil \frac{\sum_{i \in \mathcal{I}} w_i}{c} \right\rceil = \left\lceil \frac{7 + 7 + 3 + 3}{10} \right\rceil = \left\lceil \frac{20}{10} \right\rceil = 2.$$

Quindi $z(I) = 2$ e WF calcola una soluzione **ottima** su questa istanza.

4. Dalla Domanda 2, gli schemi di carico trovati da WF hanno pesi totali

$$\sum_{i \in P_1} w_i = 7 + 3 = 10, \quad \sum_{i \in P_2} w_i = 7 + 3 = 10.$$

Poiché $c/2 = 5$, nessuno schema di carico ha peso totale al più $c/2$; in particolare *al più uno* lo ha, quindi la soluzione **soddisfa** la proprietà di riempimento a metà.

Poiché $WF(I) = 2$ coincide con il lower bound $z(I) \geq 2$ trovato nella Domanda 3, la soluzione trovata da WF è **ottima**.

Esercizio 36.12 — Worst Fit 3

Consideriamo contenitori di capacità $c = 10$ e un insieme $\mathcal{I} = \{1, 2, 3, 4, 5\}$ di 5 oggetti ($n = 5$), con i seguenti pesi:

	$i = 1$	$i = 2$	$i = 3$	$i = 4$	$i = 5$
w_i	4	7	6	3	5

Domande

1. Applicare l'algoritmo Worst Fit (WF) a questa istanza. Mostrare l'esecuzione passo per passo: per ogni oggetto, seguire le variabili $maxrc$ e s dello pseudocodice durante

la scansione dei contenitori aperti, e indicare in quale contenitore è posto l'oggetto e la capacità residua che ne risulta.

2. Costruire gli schemi di carico $P_1, \dots, P_{WF(I)}$ a partire dal vettore dei contenitori ottenuto nella Domanda 1.
3. Calcolare un lower bound su $z(I)$ e stabilire se WF trova una soluzione ottima su questa istanza.
4. Verificare se gli schemi di carico trovati da WF soddisfano la proprietà di riempimento a metà (H), e stabilire se la soluzione trovata è ottima per questa istanza del BPP, confrontando $WF(I)$ con un lower bound su $z(I)$ oppure esibendo una soluzione ammissibile di valore strettamente minore.

Soluzione

1. Ricordiamo lo pseudocodice di WF: per ogni oggetto i , $maxrc := -1$, $s := -1$, e per $b := 1$ fino a nb , se $w[i] \leq rc[b]$ e $rc[b] > maxrc$ allora $s \leftarrow b$, $maxrc \leftarrow rc[b]$.
 - (a) Oggetto 1 (peso 4): il contenitore 1 è aperto all'inizio ($nb = 1$, $rc[1] = c$) e l'oggetto 1 vi entra, quindi $s = 1$ e WF pone l'oggetto 1 nel contenitore 1. Capacità residua del contenitore 1: $10 - 4 = 6$.
 - (b) Oggetto 2 (peso 7): $maxrc := -1$, $s := -1$. Per $b = 1$: $w[2] = 7 \leq rc[1] = 6$? No. Nessun contenitore va bene, quindi WF apre il contenitore 2 e vi pone l'oggetto 2. Capacità residua del contenitore 2: $10 - 7 = 3$.
 - (c) Oggetto 3 (peso 6): $maxrc := -1$, $s := -1$. Per $b = 1$: $w[3] = 6 \leq rc[1] = 6$ e $rc[1] = 6 > -1$, quindi $s \leftarrow 1$, $maxrc \leftarrow 6$. Per $b = 2$: $w[3] = 6 \leq rc[2] = 3$? No. Quindi $s = 1$: l'oggetto 3 è posto nel contenitore 1. Capacità residua del contenitore 1: $6 - 6 = 0$.
 - (d) Oggetto 4 (peso 3): $maxrc := -1$, $s := -1$. Per $b = 1$: $w[4] = 3 \leq rc[1] = 0$? No. Per $b = 2$: $w[4] = 3 \leq rc[2] = 3$ e $rc[2] = 3 > -1$, quindi $s \leftarrow 2$, $maxrc \leftarrow 3$. Quindi $s = 2$: l'oggetto 4 è posto nel contenitore 2. Capacità residua del contenitore 2: $3 - 3 = 0$.
 - (e) Oggetto 5 (peso 5): $maxrc := -1$, $s := -1$. Per $b = 1$: $w[5] = 5 \leq rc[1] = 0$? No. Per $b = 2$: $w[5] = 5 \leq rc[2] = 0$? No. Nessun contenitore va bene, quindi WF apre il contenitore 3 e vi pone l'oggetto 5. Capacità residua del contenitore 3: $10 - 5 = 5$.

Quindi WF restituisce:

$$bin = (1 \quad 2 \quad 1 \quad 2 \quad 3), \quad nb = 3.$$

e pertanto

$$WF(I) = nb = 3.$$

2. Dal vettore $bin = (1, 2, 1, 2, 3)$, gli schemi di carico sono

$$P_1 = \{i : bin_i = 1\} = \{1, 3\},$$

$$P_2 = \{i : bin_i = 2\} = \{2, 4\},$$

$$P_3 = \{i : bin_i = 3\} = \{5\}.$$

3. Di conseguenza,

$$z(I) \leq 3.$$

Inoltre,

$$z(I) \geq \left\lceil \frac{\sum_{i \in \mathcal{I}} w_i}{c} \right\rceil = \left\lceil \frac{4 + 7 + 6 + 3 + 5}{10} \right\rceil = \left\lceil \frac{25}{10} \right\rceil = 3.$$

Quindi $z(I) = 3$ e WF calcola una soluzione **ottima** su questa istanza.

4. Dalla Domanda 2, gli schemi di carico trovati da WF hanno pesi totali

$$\sum_{i \in P_1} w_i = 4 + 6 = 10, \quad \sum_{i \in P_2} w_i = 7 + 3 = 10, \quad \sum_{i \in P_3} w_i = 5.$$

Poiché $c/2 = 5$, solo P_3 ha peso totale al più $c/2$ (gli altri due lo superano); quindi la soluzione **soddisfa** la proprietà di riempimento a metà.

Poiché $WF(I) = 3$ coincide con il lower bound $z(I) \geq 3$ trovato nella Domanda 3, la soluzione trovata da WF è **ottima**.

Problemi di colorazione



Colorazione dei vertici: limiti dell'approssimazione

Risultati usati in questi esercizi

Dal capitolo [Colorazione dei vertici: limiti dell'approssimazione](#), per un grafo $G = (V, E)$ con n vertici, m spigoli e grado massimo $\Delta(G)$:

nessun rapporto sotto $\frac{4}{3}$: $\frac{\mathcal{A}(I)}{\chi(G)} < \frac{4}{3}$ su ogni grafo deciderebbe $\chi(G) \leq 3$,
a meno che $P = NP$; (C1)

grafi bipartiti: $\chi(G) \leq 2 \iff$ nessun ciclo dispari; visita in ampiezza in $O(n + m)$; (C2)

colorazione greedy: $GC(I) \leq \Delta(G) + 1$ per ogni ordine dei vertici; (C3)

grafi planari: $m \leq 3n - 6$ ($n \geq 3$), $PL(I) - \chi(G) \leq 2$ (≤ 1 con 4 colori); (C4)

grafi di intervalli: GC per estremo sinistro non decrescente usa $\omega(G) = \chi(G)$ colori. (C5)

Esercizio 37.1 — Il rapporto quattro terzi (1)

Consideriamo i due grafi

G_1 : la ruota con 6 vertici: il ciclo $1, 2, 3, 4, 5, 1$ e il vertice 6 adiacente a tutti gli altri,
 $E_1 = \{\{1, 2\}, \{2, 3\}, \{3, 4\}, \{4, 5\}, \{1, 5\}, \{1, 6\}, \{2, 6\}, \{3, 6\}, \{4, 6\}, \{5, 6\}\};$

G_2 : il ciclo con 7 vertici, $E_2 = \{\{1, 2\}, \{2, 3\}, \{3, 4\}, \{4, 5\}, \{5, 6\}, \{6, 7\}, \{1, 7\}\}.$

Supponiamo che un algoritmo polinomiale $\mathcal{A}$ per il VCP abbia la garanzia $\frac{\mathcal{A}(I)}{\chi(G)} < \frac{4}{3}$ su ogni grafo.

Domande

1. Calcolare $\chi(G_1)$ e $\chi(G_2)$, con una colorazione e una dimostrazione che meno colori non bastano.
2. Quali numeri di colori può restituire $\mathcal{A}$ su G_1 e su G_2 ?
3. Su un terzo grafo G_3 , $\mathcal{A}$ restituisce 5 colori. Che cosa si può dire di $\chi(G_3)$? $\mathcal{A}$ deciderebbe se $\chi(G_3) \leq 3$?

4. Lo stesso argomento con $\frac{\mathcal{A}(I)}{\chi(G)} < \frac{3}{2}$ deciderebbe se $\chi(G) \leq 2$. Perché questo non dimostra nulla?

Soluzione

Usiamo i risultati (C1)–(C5) del riquadro all’inizio del capitolo.

1. G_2 è un ciclo di lunghezza dispari, che richiede 3 colori, e $1, 2, 1, 2, 1, 2, 3$ è una colorazione: $\chi(G_2) = 3$. In G_1 il ciclo esterno $1, 2, 3, 4, 5$ è dispari e richiede 3 colori, e il vertice 6 è adiacente a tutti i suoi vertici e richiede un quarto colore; la colorazione $1, 2, 1, 2, 3$ del ciclo e 4 per il vertice 6 dà $\chi(G_1) = 4$. L’enumerazione conferma entrambi i valori.
2. Su G_2 : $\mathcal{A}(I) \geq \chi(G_2) = 3$ e $\mathcal{A}(I) < \frac{4}{3} \cdot 3 = 4$, quindi $\mathcal{A}(I) = 3$. Su G_1 : $4 \leq \mathcal{A}(I) < \frac{4}{3} \cdot 4 = \frac{16}{3}$, quindi $\mathcal{A}(I) \in \{4, 5\}$. Come in (C1), $\mathcal{A}$ restituisce al più 3 colori esattamente sui grafi con $\chi(G) \leq 3$.
3. Da $5 < \frac{4}{3}\chi(G_3)$ otteniamo $\chi(G_3) > \frac{15}{4}$, quindi $\chi(G_3) \geq 4$; l’ammissibilità dà $\chi(G_3) \leq 5$. Sì: $\mathcal{A}$ ha restituito più di 3 colori, quindi $\chi(G_3) \geq 4$ e G_3 non è 3-colorabile.
4. Decidere se $\chi(G) \leq 2$ richiede tempo polinomiale con la visita in ampiezza (C2): un algoritmo che lo decide non dà alcuna contraddizione con $P \neq NP$. La soglia $\frac{4}{3}$ viene dalla 3-colorabilità, che è NP-completa.

Esercizio 37.2 — Grafi bipartiti e visita in ampiezza (1)

Consideriamo i grafi

G_a : $V = \{1, \dots, 8\}$, $E_a = \{\{1, 2\}, \{1, 4\}, \{1, 5\}, \{2, 3\}, \{2, 6\}, \{3, 4\}, \{3, 7\}, \{4, 8\}, \{5, 6\}, \{5, 8\}, \{6, 7\}, \{7, 8\}\}$;

G_b : $V = \{1, \dots, 7\}$, $E_b = \{\{1, 2\}, \{2, 3\}, \{3, 4\}, \{4, 5\}, \{5, 6\}, \{1, 6\}, \{1, 4\}, \{2, 7\}, \{5, 7\}\}$.

Eseguiamo la visita in ampiezza dal vertice 1, visitando i vicini di un vertice in ordine crescente.

Domande

1. Calcolare i livelli dei vertici di G_a e stabilire se G_a è bipartito. Se lo è, dare una colorazione con 2 colori.
2. Calcolare i livelli dei vertici di G_b e trovare uno spigolo fra due livelli della stessa parità (il primo in ordine lessicografico).
3. A partire da quello spigolo, costruire un ciclo di lunghezza dispari di G_b e trarre una conclusione su $\chi(G_b)$.
4. Dare una colorazione di G_b con 3 colori. Quanto vale $\chi(G_b)$?

Soluzione

Usiamo i risultati (C1)–(C5) del riquadro all’inizio del capitolo.

1. I livelli sono: livello 0: $\{1\}$; livello 1: $\{2, 4, 5\}$; livello 2: $\{3, 6, 8\}$; livello 3: $\{7\}$. Nessuno spigolo unisce due livelli della stessa parità, quindi G_a è bipartito (C2): colore 1 per i livelli pari, $\{1, 3, 6, 8\}$, e colore 2 per i livelli dispari, $\{2, 4, 5, 7\}$. Quindi $\chi(G_a) = 2$.
2. I livelli sono: livello 0: $\{1\}$; livello 1: $\{2, 4, 6\}$; livello 2: $\{3, 5, 7\}$. Il primo spigolo con entrambi gli estremi in livelli della stessa parità è $\{5, 7\}$, con livelli 2 e 2.
3. I cammini dell’albero dall’antenato comune a 5 e a 7, con lo spigolo $\{5, 7\}$, danno il ciclo $5, 4, 1, 2, 7, 5$ con 5 spigoli, un numero dispari. Un ciclo dispari richiede 3 colori: $\chi(G_b) \geq 3$.
4. La colorazione greedy nell’ordine $1, \dots, 7$ dà i colori $1, 2, 1, 2, 1, 2, 3$, una colorazione ammissibile con 3 colori. Quindi $\chi(G_b) = 3$.

Esercizio 37.3 — La colorazione greedy e il bound sul grado (1)

Il grafo corona G_k , $k \geq 3$, ha i vertici $a_i = i$ e $b_i = k + i$, $i = 1, \dots, k$, e gli spigoli $\{a_i, b_j\}$ per ogni $i \neq j$. Per $k = 4$: $E = \{\{1, 6\}, \{1, 7\}, \{1, 8\}, \{2, 5\}, \{2, 7\}, \{2, 8\}, \{3, 5\}, \{3, 6\}, \{3, 8\}, \{4, 5\}, \{4, 6\}, \{4, 7\}\}$. La colorazione greedy GC assegna a ogni vertice, in un ordine dato, il colore più piccolo non usato dai suoi vicini già colorati.

Domande

1. Calcolare $\Delta(G_4)$ e $\chi(G_4)$.
2. Applicare GC a G_4 con l’ordine $\lambda_1 = (1, 2, 3, 4, 5, 6, 7, 8)$ e con l’ordine $\lambda_2 = (1, 5, 2, 6, 3, 7, 4, 8)$.
3. Confrontare i due risultati con il bound (C3) e con $\chi(G_4)$. Il bound $\Delta(G) + 1$ è raggiunto?
4. Generalizzare l’ordine λ_2 a G_k e calcolare il rapporto $\frac{GC(I)}{\chi(G_k)}$ per $k = 3, \dots, 6$. GC , con un ordine arbitrario, è un algoritmo di α -approssimazione per qualche costante α ?

Soluzione

Usiamo i risultati (C1)–(C5) del riquadro all’inizio del capitolo.

1. Ogni vertice è adiacente a $k - 1 = 3$ vertici dell’altro lato: $\Delta(G_4) = 3$. Gli insiemi $\{1, 2, 3, 4\}$ e $\{5, 6, 7, 8\}$ sono stabili, e ci sono spigoli: $\chi(G_4) = 2$.
2. Con λ_1 i vertici $1, 2, 3, 4$ non hanno vicini colorati e prendono il colore 1; poi $5, \dots, 8$ prendono il colore 2: $GC(I) = 2$. Con λ_2 , il vertice a_i è adiacente a $b_1, \dots, b_{i-1}$, che hanno i colori $1, \dots, i - 1$, e prende il colore i ; il vertice b_i fa lo stesso. I colori sono $1, 1, 2, 2, 3, 3, 4, 4$ e $GC(I) = 4$.

- Entrambi i risultati soddisfano $GC(I) \leq \Delta(G_4) + 1 = 4$, e λ_2 raggiunge il bound. Con λ_2 la colorazione greedy usa il doppio del numero ottimo di colori, mentre con λ_1 è ottima: l'ordine conta.
- Con l'ordine $a_1, b_1, a_2, b_2, \dots, a_k, b_k$ i vertici a_i e b_i prendono il colore i , quindi $GC(I) = k$:

k	n	$\Delta(G_k)$	$GC(I)$	$\frac{GC(I)}{\chi(G_k)}$
3	6	2	3	$\frac{3}{2}$
4	8	3	4	2
5	10	4	5	$\frac{5}{2}$
6	12	5	6	3

Il rapporto $\frac{k}{2}$ cresce senza limite: su tutti i grafi, GC non ha una garanzia costante. Sui grafi di grado massimo al più Δ il rapporto è al più $\frac{\Delta+1}{2}$, raggiunto qui.

Esercizio 37.4 — Grafi di intervalli (1)

Sette richieste usano una sala negli intervalli di tempo

richiesta v	1	2	3	4	5	6	7
$[a_v, b_v]$	[1, 3]	[2, 6]	[4, 7]	[3, 5]	[6, 10]	[8, 9]	[9, 12]

Due richieste sono in conflitto se i loro intervalli chiusi si intersecano, e un colore è una sala.

Domande

- Costruire il grafo di intervalli G : elencarne gli spigoli.
- Applicare la colorazione greedy con le richieste in ordine di estremo sinistro non decrescente.
- Trovare un punto coperto dal massimo numero di intervalli, e dedurre $\omega(G)$. La colorazione della domanda 2 è ottima?
- Applicare la colorazione greedy con l'ordine (1, 2, 4, 5, 3, 6, 7). Che cosa mostra?

Soluzione

Usiamo i risultati (C1)–(C5) del riquadro all'inizio del capitolo.

- $E = \{\{1, 2\}, \{1, 4\}, \{2, 3\}, \{2, 4\}, \{2, 5\}, \{3, 4\}, \{3, 5\}, \{5, 6\}, \{5, 7\}, \{6, 7\}\}$: 10 spigoli.
- L'ordine è (1, 2, 4, 3, 5, 6, 7), e i colori sono $1 \rightarrow 1, 2 \rightarrow 2, 4 \rightarrow 3, 3 \rightarrow 1, 5 \rightarrow 3, 6 \rightarrow 1, 7 \rightarrow 2$: $GC(I) = 3$.
- Il punto 3 sta negli intervalli $\{1, 2, 4\}$, che si intersecano a due a due e formano una clique: $\omega(G) \geq 3$; nessun punto sta in più di 3 intervalli, quindi $\omega(G) = 3$. Per (C5) e il bound di clique, $GC(I) = \omega(G) = \chi(G) = 3$: la colorazione è ottima, con 3 sale.

4. I colori sono $1 \rightarrow 1, 2 \rightarrow 2, 4 \rightarrow 3, 5 \rightarrow 1, 3 \rightarrow 4, 6 \rightarrow 2, 7 \rightarrow 3$: il vertice 3 vede i colori $\{1, 2, 3\}$ sui suoi vicini e prende il colore 4. $GC(I) = 4 > \chi(G)$: è l'ordine degli estremi sinistri a rendere esatta la colorazione greedy.

Esercizio 37.5 — Grafi planari e una garanzia additiva (1)

Consideriamo i grafi

W : la ruota, $E = \{\{1, 2\}, \{2, 3\}, \{3, 4\}, \{4, 5\}, \{1, 5\}, \{1, 6\}, \{2, 6\}, \{3, 6\}, \{4, 6\}, \{5, 6\}\}$;

O : l'ottaedro, con $V = \{1, \dots, 6\}$ e tutti gli spigoli tranne $\{1, 6\}, \{2, 5\}, \{3, 4\}$;

K_5 : il grafo completo con 5 vertici.

L'algoritmo PL per i grafi planari usa 1 colore se non ci sono spigoli, 2 colori se il grafo è bipartito, e altrimenti i 5 colori del teorema di Heawood.

Domande

1. Calcolare n , m e $3n - 6$ per i tre grafi. Quali di essi possono essere planari?
2. W e O sono planari (disegnarli). Trovare in ciascuno un vertice di grado al più 5, come garantisce il lemma del capitolo.
3. W è bipartito? E O ? Che cosa garantisce PL su di essi, e quanto valgono $\chi(W)$ e $\chi(O)$ (per enumerazione)?
4. Limitare l'errore additivo di PL su W e su O , e l'errore additivo con l'algoritmo dei quattro colori al posto di quello dei cinque colori.

Soluzione

Usiamo i risultati (C1)–(C5) del riquadro all'inizio del capitolo.

	grafo	n	m	$3n - 6$
1.	W	6	10	12
	O	6	12	12
	K_5	5	10	9

Per (C4) un grafo planare ha $m \leq 3n - 6$: K_5 , con $10 > 9$, non è planare. Il bound non esclude W e O .

2. In W i vertici del ciclo hanno grado 3, e in O ogni vertice ha grado 4: entrambi i valori sono al più 5.
3. Entrambi i grafi contengono triangoli ($\{1, 2, 6\}$ in W , $\{1, 2, 3\}$ in O), cicli dispari: nessuno dei due è bipartito, quindi PL usa al più 5 colori. L'enumerazione dà $\chi(W) = 4$ (il ciclo dispari $1, \dots, 5$ richiede 3 colori e il vertice 6 un quarto) e $\chi(O) = 3$ (le classi $\{1, 6\}, \{2, 5\}, \{3, 4\}$).

4. Su W : $PL(I) - \chi(W) \leq 5 - 4 = 1$; su O : $PL(I) - \chi(O) \leq 5 - 3 = 2$, il bound di (C4).
Con 4 colori gli errori sono al più 0 su W , dove l'algoritmo dei quattro colori è esatto, e 1 su O .

Problemi di clique



Clique massima: limiti dell'approssimazione

Risultati usati in questi esercizi

Dal capitolo [Clique massima: limiti dell'approssimazione](#), per i grafi $G = (V, E)$ e $H = (W, F)$:

prodotto forte: $(u, x) \sim (v, y) \iff (u = v \text{ oppure } \{u, v\} \in E) \text{ e } (x = y \text{ oppure } \{x, y\} \in F)$; (Q1)

numero di clique: $\omega(G \boxtimes H) = \omega(G)\omega(H)$, $|K| \leq |p(K)| \cdot \max_u |K_u|$; (Q2)

amplificazione: rapporto α su $G \boxtimes G \Rightarrow$ rapporto $\sqrt{\alpha}$ su G ;
 t prodotti: $\alpha^{1/2^t}$; (Q3)

grado limitato: $\omega(G) \leq \Delta(G) + 1$, enumerazione di $\sum_v 2^{\deg(v)}$ sottoinsiemi; (Q4)

insiemi stabili: $\alpha(G) = \omega(\overline{G})$,
 un insieme stabile massimale S ha $\frac{\alpha(G)}{|S|} \leq \Delta(G) + 1$. (Q5)

Esercizio 38.1 — Il prodotto forte (1)

Sia G il cammino con i vertici $1, 2, 3$ e gli spigoli $\{1, 2\}, \{2, 3\}$, e consideriamo il prodotto forte $G \boxtimes G$, i cui vertici sono le coppie (i, j) con $i, j \in \{1, 2, 3\}$.

Domande

1. Elencare i vicini di $(2, 2)$ e di $(1, 1)$ in $G \boxtimes G$, e contarne gli spigoli.
2. $\{(1, 1), (2, 2), (3, 3)\}$ è una clique di $G \boxtimes G$?
3. Calcolare $\omega(G)$ e $\omega(G \boxtimes G)$, e dare tutte le clique massime di $G \boxtimes G$.
4. Per la clique $K = \{(1, 1), (1, 2), (2, 1), (2, 2)\}$, calcolare la proiezione $p(K)$ e le fibre K_u , e verificare la disuguaglianza di (Q2).

Soluzione

Usiamo i risultati (Q1)–(Q5) del riquadro all'inizio del capitolo.

1. Per (Q1), $(2, 2)$ è adiacente a ogni altra coppia, perché 2 è adiacente a 1 e 3: i suoi 8 vicini sono tutti gli altri vertici. Il vertice $(1, 1)$ ha i vicini $(1, 2), (2, 1), (2, 2)$.

Contando i vicini di tutti i vertici e dividendo per due, $G \boxtimes G$ ha 20 spigoli.

2. No: $(1, 1)$ e $(3, 3)$ non sono adiacenti, perché $1 \neq 3$ e $\{1, 3\}$ non è uno spigolo di G .
3. $\omega(G) = 2$ (gli spigoli), e per (Q2) $\omega(G \boxtimes G) = 2 \cdot 2 = 4$. L'enumerazione lo conferma, con le 4 clique massime $\{i, i+1\} \times \{j, j+1\}$, $i, j \in \{1, 2\}$: $\{(1, 1), (1, 2), (2, 1), (2, 2)\}$, $\{(1, 2), (1, 3), (2, 2), (2, 3)\}$, $\{(2, 1), (2, 2), (3, 1), (3, 2)\}$, $\{(2, 2), (2, 3), (3, 2), (3, 3)\}$.
4. $p(K) = \{1, 2\}$, con le fibre $K_1 = K_2 = \{1, 2\}$: sono tutte clique di G , e $|K| = 4 \leq |p(K)| \cdot \max_u |K_u| = 2 \cdot 2 = 4$.

Esercizio 38.2 — Amplificare un rapporto (1)

Supponiamo che un algoritmo polinomiale $\mathcal{A}$ per il MCP abbia la garanzia $\frac{\omega(G)}{\mathcal{A}(I)} \leq \alpha$ con $\alpha = 4$ su ogni grafo, e applichiamo la costruzione del capitolo a un grafo G con $n = 50$ vertici: t prodotti forti, poi $\mathcal{A}$, poi t estrazioni di una clique di dimensione almeno pari alla radice quadrata.

Domande

1. Calcolare il rapporto garantito e il numero di vertici del grafo dato ad $\mathcal{A}$, per $t = 0, 1, 2, 3, 4$.
2. Qual è il più piccolo t che dà il rapporto $\beta = \frac{3}{2}$? E $\beta = \frac{5}{4}$, $\beta = \frac{11}{10}$?
3. Su $G \boxtimes G$ l'algoritmo restituisce una clique K con 10 vertici. Quanto è grande, almeno, la clique di G estratta da essa? Che cosa dice la garanzia di $\mathcal{A}$ su $\omega(G)$?
4. Perché t non può crescere con n , per esempio $t = \log_2 n$, per ottenere il rapporto 1?

Soluzione

Usiamo i risultati (Q1)–(Q5) del riquadro all'inizio del capitolo.

1. Per (Q3) il rapporto dopo t prodotti è $\alpha^{1/2^t}$, e il grafo ha n^{2^t} vertici:

t	rapporto $4^{1/2^t}$	vertici 50^{2^t}
0	4.000	50
1	2.000	2 500
2	1.414	6 250 000
3	1.189	$3.9 \cdot 10^{13}$
4	1.091	$1.5 \cdot 10^{27}$

2. Serve $4^{1/2^t} \leq \beta$, cioè $2^t \geq \frac{\ln 4}{\ln \beta}$:

- $\beta = \frac{3}{2}$: $\frac{\ln 4}{\ln \beta} \approx 3.42$, quindi $t = 2$ (rapporto 1.414).
- $\beta = \frac{5}{4}$: $\frac{\ln 4}{\ln \beta} \approx 6.21$, quindi $t = 3$ (rapporto 1.189).
- $\beta = \frac{11}{10}$: $\frac{\ln 4}{\ln \beta} \approx 14.55$, quindi $t = 4$ (rapporto 1.091).

3. Per (Q2), $|p(K)| \cdot \max_u |K_u| \geq 10$; entrambi sono interi, e $3 \cdot 3 = 9 < 10$, quindi il più grande ha almeno 4 vertici, una clique di G . La garanzia dà $10 \geq \frac{\omega(G)^2}{4}$, quindi $\omega(G)^2 \leq 40$ e $\omega(G) \leq 6$: la clique estratta rispetta il rapporto $\sqrt{4} = 2$, perché $2 \cdot 4 \geq 6$.
4. Il grafo dato ad $\mathcal{A}$ ha n^{2^t} vertici: per $t = \log_2 n$ ha n^n vertici, e l'algoritmo non è più polinomiale. La costruzione dà ogni rapporto *costante* $\beta > 1$, con t costante.

Esercizio 38.3 — Casi particolari del MCP e del MSSP (1)

Consideriamo il prisma G , con $V = \{1, \dots, 6\}$ e

$$E = \{\{1, 2\}, \{1, 3\}, \{2, 3\}, \{4, 5\}, \{4, 6\}, \{5, 6\}, \{1, 4\}, \{2, 5\}, \{3, 6\}\},$$

e la stella S_4 , con il centro 1 adiacente ai vertici 2, 3, 4, 5 e nessun altro spigolo.

Domande

1. Calcolare $\Delta(G)$ e il numero di sottoinsiemi esaminati dall'enumerazione di (Q4). Quali clique massime trova?
2. Verificare che G soddisfa $m \leq 3n - 6$ (G è planare). Quale bound su $\omega(G)$ dà la planarità, e quale il grado?
3. Costruire un insieme stabile massimale di G aggiungendo i vertici nell'ordine $1, \dots, 6$ finché è possibile, e confrontarlo con $\alpha(G)$ e con il bound (Q5).
4. Fare lo stesso sulla stella S_4 . Quanto è vicino il rapporto al bound $\Delta + 1$?

Soluzione

Usiamo i risultati (Q1)–(Q5) del riquadro all'inizio del capitolo.

1. Ogni vertice ha grado 3: $\Delta(G) = 3$, e l'enumerazione esamina $\sum_v 2^{\deg(v)} = 6 \cdot 2^3 = 48$ sottoinsiemi. Trova le due clique massime $\{1, 2, 3\}$ e $\{4, 5, 6\}$ (ciascuna a partire da uno qualsiasi dei suoi vertici): $\omega(G) = 3$.
2. $m = 9 \leq 3 \cdot 6 - 6 = 12$. La planarità dà $\omega(G) \leq 4$, il grado $\omega(G) \leq \Delta(G) + 1 = 4$; il valore vero è 3.
3. Il vertice 1 si prende; 2, 3, 4 sono adiacenti a 1; 5 si prende; 6 è adiacente a 5: $S = \{1, 5\}$. Un insieme stabile ha al più un vertice per triangolo, quindi $\alpha(G) = 2$ e S è ottimo. Il bound (Q5) garantisce $|S| \geq \frac{n}{\Delta+1} = \frac{3}{2}$, quindi $|S| \geq 2$.
4. Il centro 1 si prende per primo, e ogni altro vertice è adiacente a esso: $S = \{1\}$. Le foglie formano un insieme stabile, $\alpha(S_4) = 4$, e il rapporto è $\frac{4}{1} = 4$, contro il bound $\Delta + 1 = 5$: un insieme stabile massimale può essere lontano da uno massimo.

Esercizio 38.4 — Grafo complementare e insiemi stabili (1)

Sia G il grafo di Petersen, l'istanza di riferimento del VCP, con $V = \{1, \dots, 10\}$ e $E = \{\{1, 2\}, \{1, 5\}, \{1, 6\}, \{2, 3\}, \{2, 7\}, \{3, 4\}, \{3, 8\}, \{4, 5\}, \{4, 9\}, \{5, 10\}, \{6, 8\}, \{6, 9\}, \{7, 9\}, \{7, 10\}, \{8, 10\}\}$.

Domande

1. Calcolare il numero di spigoli e i gradi del grafo complementare $\overline{G}$. $\overline{G}$ è planare?
2. Calcolare $\alpha(G)$ e dedurre $\omega(\overline{G})$. Quanti sottoinsiemi esamina l'enumerazione di (Q4) su $\overline{G}$?
3. Costruire un insieme stabile massimale di G aggiungendo i vertici nell'ordine $1, \dots, 10$ finché è possibile. Che cosa garantisce (Q5)?
4. Un algoritmo per il MCP ha rapporto al più ρ su ogni grafo con 10 vertici. Che cosa dà per il MSSP su G ?

Soluzione

Usiamo i risultati (Q1)–(Q5) del riquadro all'inizio del capitolo.

1. $\overline{G}$ ha $\binom{10}{2} - 15 = 30$ spigoli, e ogni vertice ha grado $9 - 3 = 6$. Poiché $30 > 3 \cdot 10 - 6 = 24$, $\overline{G}$ non è planare: il caso particolare dei grafi planari non si applica al MSSP su G .
2. I due 5-cicli $1, \dots, 5$ e $6, \dots, 10$ contengono ciascuno al più 2 vertici di un insieme stabile, e $\{1, 4, 7, 8\}$ è stabile: $\alpha(G) = 4$, quindi $\omega(\overline{G}) = 4$, come conferma l'enumerazione. Su $\overline{G}$ l'enumerazione esamina $10 \cdot 2^6 = 640$ sottoinsiemi.
3. L'ordine dà $S = \{1, 3, 7\}$: ogni altro vertice è adiacente a uno di essi. Per (Q5), $|S| \geq \frac{n}{\Delta(G)+1} = \frac{5}{2}$, quindi $|S| \geq 3$, e il rapporto è $\frac{\alpha(G)}{|S|} = \frac{4}{3} \leq \Delta(G) + 1 = 4$.
4. Applicato a $\overline{G}$, che ha gli stessi 10 vertici, restituisce una clique di $\overline{G}$, cioè un insieme stabile di G , con almeno $\frac{\omega(\overline{G})}{\rho} = \frac{\alpha(G)}{\rho}$ vertici: lo stesso rapporto ρ per il MSSP.

Problemi di routing



TSP: il caso generale e il caso metrico

Esercizio 39.1 — Approssimazione del TSP metrico 1

Consideriamo un insieme $\mathcal{V} = \{1, \dots, 6\}$ di 6 città di coordinate 1: (3, 4), 2: (6, 1), 3: (3, 8), 4: (8, 8), 5: (0, 4), 6: (8, 0) e distanze pari alle distanze euclidee arrotondate all'intero più vicino (l'istanza è metrica):

d_{ij}	1	2	3	4	5	6
1	—	4	4	6	3	6
2	4	—	8	7	7	2
3	4	8	—	5	5	9
4	6	7	5	—	9	8
5	3	7	5	9	—	9
6	6	2	9	8	9	—

Domande

1. Calcolare un albero ricoprente di costo minimo T con l'algoritmo di Kruskal (spigoli per distanza non decrescente, pareggi per coppia di indici) e il suo peso $d(T)$.
2. Applicare l'algoritmo del doppio albero DT : indicare il circuito euleriano dell'albero raddoppiato dalla città 1 (regola di Fleury), il giro ottenuto con le scorciatoie, e $DT(I)$.
3. Applicare l'algoritmo di Christofides CH : indicare l'insieme O delle città di grado dispari, l'abbinamento perfetto di peso minimo M (confrontando tutti gli abbinamenti perfetti di O), il circuito euleriano di $T \cup M$ dalla città 1, il giro e $CH(I)$.
4. Il giro ottimo ha lunghezza $z(I) = 27$. Calcolare i due rapporti e verificare le disuguaglianze $DT(I) \leq 2d(T) \leq 2z(I)$ e $CH(I) \leq d(T) + d(M) \leq \frac{3}{2}z(I)$.

Soluzione

1. L'algoritmo di Kruskal esamina $\{2, 6\}$ (2, preso), $\{1, 5\}$ (3, preso), $\{1, 2\}$ (4, preso), $\{1, 3\}$ (4, preso), $\{3, 4\}$ (5, preso). Dopo cinque spigoli l'albero copre le sei città:

$$T = \{\{2, 6\}, \{1, 5\}, \{1, 2\}, \{1, 3\}, \{3, 4\}\}, \quad d(T) = 2 + 3 + 4 + 4 + 5 = 18.$$

È l'unico MST dell'istanza.

2. I vicini in T sono 1: $\{2, 3, 5\}$, 2: $\{1, 6\}$, 3: $\{1, 4\}$, 4: $\{3\}$, 5: $\{1\}$, 6: $\{2\}$. Il circuito euleroiano dell'albero raddoppiato dalla città 1 è la visita in profondità con i figli in ordine crescente:

$$W = (1, 2, 6, 2, 1, 3, 4, 3, 1, 5, 1), \quad 4 + 2 + 2 + 4 + 4 + 5 + 5 + 4 + 3 + 3 = 36 = 2d(T).$$

Mantenendo ogni città alla sua prima occorrenza otteniamo $\sigma = (1, 2, 6, 3, 4, 5)$, di lunghezza

$$DT(I) = d_{12} + d_{26} + d_{63} + d_{34} + d_{45} + d_{51} = 4 + 2 + 9 + 5 + 9 + 3 = 32.$$

3. I gradi in T sono $\deg(1) = 3$, $\deg(2) = 2$, $\deg(3) = 2$, $\deg(4) = 1$, $\deg(5) = 1$, $\deg(6) = 1$, quindi $O = \{1, 4, 5, 6\}$. I suoi tre abbinamenti perfetti pesano

$$\{\{1, 4\}, \{5, 6\}\}: 6 + 9 = 15, \quad \{\{1, 5\}, \{4, 6\}\}: 3 + 8 = 11, \quad \{\{1, 6\}, \{4, 5\}\}: 6 + 9 = 15,$$

quindi $M = \{\{1, 5\}, \{4, 6\}\}$, con $d(M) = 11$. Il multigrafo $T \cup M$ ha due copie di $\{1, 5\}$. La regola di Fleury dalla città 1 (vicini 2, 3, 5, 5) va in 2, poi in 6, in 4 (lungo M), in 3, torna in 1, e infine va in 5 e in 1 lungo le due copie di $\{1, 5\}$:

$$W = (1, 2, 6, 4, 3, 1, 5, 1), \quad 4 + 2 + 8 + 5 + 4 + 3 + 3 = 29 = d(T) + d(M).$$

La città 1 si ripete a metà; saltandola con una scorciatoia otteniamo $\sigma = (1, 2, 6, 4, 3, 5)$, di lunghezza

$$CH(I) = d_{12} + d_{26} + d_{64} + d_{43} + d_{35} + d_{51} = 4 + 2 + 8 + 5 + 5 + 3 = 27.$$

La scorciatoia sostituisce $3 \rightarrow 1 \rightarrow 5$ ($4 + 3$) con $3 \rightarrow 5$ (5) e fa risparmiare 2.

4. I rapporti sono

$$\frac{DT(I)}{z(I)} = \frac{32}{27}, \quad \frac{CH(I)}{z(I)} = \frac{27}{27} = 1,$$

quindi CH trova il giro ottimo, che è unico. Le disuguaglianze delle dimostrazioni sono

$$DT(I) = 32 \leq 2d(T) = 36 \leq 2z(I) = 54, \\ CH(I) = 27 \leq d(T) + d(M) = 29 \leq \frac{3}{2}z(I) = \frac{81}{2}.$$

Esercizio 39.2 — Approssimazione del TSP metrico 2

Consideriamo un insieme $\mathcal{V} = \{1, \dots, 6\}$ di 6 città di coordinate 1: (2, 4), 2: (6, 0), 3: (7, 3), 4: (0, 7), 5: (0, 1), 6: (2, 1) e distanze pari alle distanze euclidee arrotondate all'intero più vicino (l'istanza è metrica):

d_{ij}	1	2	3	4	5	6
1	—	6	5	4	4	3
2	6	—	3	9	6	4
3	5	3	—	8	7	5
4	4	9	8	—	6	6
5	4	6	7	6	—	2
6	3	4	5	6	2	—

Domande

1. Calcolare un albero ricoprente di costo minimo T con l'algoritmo di Kruskal e il suo peso $d(T)$.
2. Applicare CH : indicare O , i pesi di tutti gli abbinamenti perfetti di O , l'abbinamento M , il circuito euleriano di $T \cup M$ dalla città 1 (regola di Fleury), il giro e $CH(I)$.
3. Applicare DT e indicare $DT(I)$.
4. L'unico giro ottimo è (1, 3, 2, 6, 5, 4). Calcolare $z(I)$ e i due rapporti. Stabilire se CH è ottimo su questa istanza. Verificare il bound $d(M) \leq \frac{z(I)}{2}$ restringendo con le scorciatoie il giro ottimo alle città di O , come nella dimostrazione delle note.

Soluzione

1. L'algoritmo di Kruskal esamina $\{5, 6\}$ (2, preso), $\{1, 6\}$ (3, preso), $\{2, 3\}$ (3, preso), $\{1, 4\}$ (4, preso), $\{1, 5\}$ (4, chiude il ciclo 1, 5, 6: scartato), $\{2, 6\}$ (4, preso). Quindi

$$T = \{\{5, 6\}, \{1, 6\}, \{2, 3\}, \{1, 4\}, \{2, 6\}\}, \quad d(T) = 2 + 3 + 3 + 4 + 4 = 16,$$

l'unico MST dell'istanza.

2. I gradi in T sono $\deg(1) = 2$, $\deg(2) = 2$, $\deg(3) = 1$, $\deg(4) = 1$, $\deg(5) = 1$, $\deg(6) = 3$, quindi $O = \{3, 4, 5, 6\}$. I suoi abbinamenti perfetti pesano

$$\{\{3, 4\}, \{5, 6\}\}: 8 + 2 = 10, \quad \{\{3, 5\}, \{4, 6\}\}: 7 + 6 = 13, \quad \{\{3, 6\}, \{4, 5\}\}: 5 + 6 = 11,$$

quindi $M = \{\{3, 4\}, \{5, 6\}\}$, con $d(M) = 10$; $T \cup M$ ha due copie di $\{5, 6\}$. La regola di Fleury dalla città 1 (vicini 4, 6) va in 4, poi in 3 lungo M , in 2 e in 6. Nella città 6 gli spigoli non usati sono $\{6, 1\}$ e le due copie di $\{6, 5\}$: la città 1 ha l'indice minimo, ma $\{6, 1\}$ è un ponte degli spigoli non usati (rimuoverlo isola la città 1 dalle due copie di $\{5, 6\}$), quindi il cammino va in 5, torna in 6 e infine va in 1:

$$W = (1, 4, 3, 2, 6, 5, 6, 1), \quad 4 + 8 + 3 + 4 + 2 + 2 + 3 = 26 = d(T) + d(M).$$

Saltando con una scorciatoia la seconda visita della città 6 otteniamo $\sigma = (1, 4, 3, 2, 6, 5)$, di lunghezza

$$CH(I) = d_{14} + d_{43} + d_{32} + d_{26} + d_{65} + d_{51} = 4 + 8 + 3 + 4 + 2 + 4 = 25.$$

3. I vicini in T sono 1: $\{4, 6\}$, 2: $\{3, 6\}$, 3: $\{2\}$, 4: $\{1\}$, 5: $\{6\}$, 6: $\{1, 2, 5\}$. La visita in profondità dalla città 1 è

$$W = (1, 4, 1, 6, 2, 3, 2, 6, 5, 6, 1), \quad \text{di lunghezza } 32 = 2d(T),$$

e le scorciatoie danno $\sigma = (1, 4, 6, 2, 3, 5)$, di lunghezza

$$DT(I) = 4 + 6 + 4 + 3 + 7 + 4 = 28.$$

4. $z(I) = d_{13} + d_{32} + d_{26} + d_{65} + d_{54} + d_{41} = 5 + 3 + 4 + 2 + 6 + 4 = 24$, e

$$\frac{CH(I)}{z(I)} = \frac{25}{24}, \quad \frac{DT(I)}{z(I)} = \frac{28}{24} = \frac{7}{6}.$$

Qui CH non è ottimo, ma supera l'ottimo di una sola unità. Restringendo con le scorciatoie il giro ottimo a $O = \{3, 4, 5, 6\}$ otteniamo il ciclo $(3, 6, 5, 4)$, di lunghezza $d_{36} + d_{65} + d_{54} + d_{43} = 5 + 2 + 6 + 8 = 21 \leq 24$. I suoi spigoli alterni formano gli abbinamenti $\{\{3, 6\}, \{5, 4\}\}$, di peso 11, e $\{\{6, 5\}, \{4, 3\}\}$, di peso 10, e infatti

$$d(M) = 10 \leq \min\{11, 10\} \leq \frac{21}{2} \leq \frac{z(I)}{2} = 12.$$

Esercizio 39.3 — Approssimazione del TSP metrico 3

Consideriamo due grafi con 5 vertici, $V = \{1, 2, 3, 4, 5\}$:

$$G_1 : E_1 = \{\{1, 2\}, \{2, 3\}, \{3, 4\}, \{4, 5\}, \{5, 1\}, \{1, 3\}\},$$

$$G_2 : E_2 = \{\{1, 3\}, \{1, 4\}, \{1, 5\}, \{2, 3\}, \{2, 4\}, \{2, 5\}\}.$$

Supponiamo che un algoritmo polinomiale $\mathcal{A}$ per il TSP abbia la garanzia $\frac{\mathcal{A}(I)}{z(I)} \leq \alpha$ con $\alpha = 2$ su ogni istanza del TSP.

Domande

- Costruire le istanze I_{G_1} e I_{G_2} del TSP della dimostrazione delle note: $d_{ij} = 1$ se $\{i, j\}$ è uno spigolo e $d_{ij} = \lceil \alpha n \rceil + 1$ altrimenti.
- Stabilire se G_1 e G_2 sono hamiltoniani, e calcolare $z(I_{G_1})$ e $z(I_{G_2})$.
- Stabilire che cosa può restituire $\mathcal{A}$ sulle due istanze. Spiegare come $\mathcal{A}$ deciderebbe se un grafo è hamiltoniano.
- Stabilire se I_{G_2} è metrica e se $\mathcal{A}$ potrebbe essere l'algoritmo di Christofides.

Soluzione

1. Con $n = 5$ e $\alpha = 2$, le coppie che non sono spigoli ricevono distanza $\lceil 2 \cdot 5 \rceil + 1 = 11$:

I_{G_1}	1	2	3	4	5
1	—	1	1	11	1
2	1	—	1	11	11
3	1	1	—	1	11
4	11	11	1	—	1
5	1	11	11	1	—

I_{G_2}	1	2	3	4	5
1	—	11	1	1	1
2	11	—	1	1	1
3	1	1	—	11	11
4	1	1	11	—	11
5	1	1	11	11	—

2. G_1 contiene il ciclo $(1, 2, 3, 4, 5)$, che è hamiltoniano, quindi il giro $(1, 2, 3, 4, 5)$ ha lunghezza 5 e $z(I_{G_1}) = 5$.

G_2 è il grafo bipartito completo con lati $\{1, 2\}$ e $\{3, 4, 5\}$. Un ciclo in un grafo bipartito alterna i due lati, quindi ha lo stesso numero di vertici su ciascun lato: nessun ciclo può contenere i tre vertici 3, 4, 5 e solo due vertici dell'altro lato, e G_2 non è hamiltoniano. Ogni giro di I_{G_2} usa almeno una distanza 11. Il giro $(1, 3, 2, 4, 5)$ usa gli spigoli $\{1, 3\}, \{3, 2\}, \{2, 4\}, \{4, 5\}$ e la coppia $\{1, 5\}$, che non è uno spigolo, quindi la sua lunghezza è $4 \cdot 1 + 11 = 15$, e

$$z(I_{G_2}) = 15.$$

(Enumerando i 12 giri, le loro lunghezze sono solo 15 e 35.)

3. Su I_{G_1} : $\mathcal{A}(I_{G_1}) \leq 2 \cdot 5 = 10$. Un giro che usa una distanza 11 ha lunghezza almeno $4 + 11 = 15 > 10$, quindi $\mathcal{A}$ deve restituire un giro di lunghezza 5, cioè un ciclo hamiltoniano di G_1 .

Su I_{G_2} : $\mathcal{A}(I_{G_2}) \geq z(I_{G_2}) = 15 > 10$.

Quindi il test " $\mathcal{A}(I_G) \leq \alpha n = 10$ " risponde "sì" su G_1 e "no" su G_2 . In generale, costruire I_G ed eseguire $\mathcal{A}$ richiede tempo polinomiale e dà la risposta corretta su ogni grafo, quindi il problema NP-completo del ciclo hamiltoniano sarebbe risolto in tempo polinomiale: un tale $\mathcal{A}$ non può esistere a meno che $P = NP$.

4. I_{G_2} non è metrica: per esempio $d_{12} = 11 > d_{13} + d_{32} = 1 + 1 = 2$. La garanzia $\frac{3}{2}$ dell'algoritmo di Christofides vale solo per le istanze metriche, quindi non dice nulla su I_{G_2} , e non c'è contraddizione con la proposizione delle note: è la disuguaglianza triangolare a rendere approssimabile il TSP.

Problemi di scheduling

Approssimazione per $P||C_{\max}$

Esercizio 40.1 — Approssimazione di $P||C_{\max}$ 1

Consideriamo $m = 3$ macchine identiche e un insieme $\mathcal{J} = \{1, \dots, 7\}$ di 7 lavori ($n = 7$), con i seguenti tempi di esecuzione:

	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$	$j = 6$	$j = 7$
t_j	2	2	4	8	8	6	6

Domande

1. Applicare *LIST* con la lista $\lambda = (1, 2, \dots, 7)$. Mostrare, per ogni lavoro, i carichi prima dell'assegnamento e la macchina scelta, e indicare $LIST(I)$.
2. Individuare un lavoro ℓ che termina per ultimo nello schedule della Domanda 1 e verificare la disuguaglianza $S_\ell \leq \frac{1}{m} \sum_{j \neq \ell} t_j$ usata nella dimostrazione della garanzia di *LIST*.
3. Applicare *LPT* e indicare $LPT(I)$.
4. Calcolare il lower bound $\max\{\max_j t_j, \frac{1}{m} \sum_j t_j\}$ su $z(I)$ e determinare $z(I)$. Calcolare i due rapporti $\frac{LIST(I)}{z(I)}$ e $\frac{LPT(I)}{z(I)}$ e confrontarli con le garanzie $2 - \frac{1}{m}$ e $\frac{4}{3} - \frac{1}{3m}$.

Soluzione

1. Ogni lavoro va su una macchina di carico minimo (indice minimo in caso di pareggio):

lavoro j	t_j	carichi (L_1, L_2, L_3)	prima macchina
1	2	(0, 0, 0)	1
2	2	(2, 0, 0)	2
3	4	(2, 2, 0)	3
4	8	(2, 2, 4)	1
5	8	(10, 2, 4)	2
6	6	(10, 10, 4)	3
7	6	(10, 10, 10)	1

Quindi $mach = (1, 2, 3, 1, 2, 3, 1)$, $J_1 = \{1, 4, 7\}$, $J_2 = \{2, 5\}$, $J_3 = \{3, 6\}$, i carichi sono (16, 10, 10) e

$$LIST(I) = 16.$$

2. Il lavoro $\ell = 7$ termina per ultimo: inizia in $S_7 = 10$ sulla macchina 1 e termina in 16. Il bound dà

$$S_7 = 10 \leq \frac{1}{3} \sum_{j \neq 7} t_j = \frac{2 + 2 + 4 + 8 + 8 + 6}{3} = \frac{30}{3} = 10,$$

che vale all'uguaglianza: quando arriva il lavoro 7 i tre carichi sono uguali, quindi il minimo coincide con la media.

3. I lavori ordinati per tempo di esecuzione non crescente (pareggi per indice) sono $\lambda = (4, 5, 6, 7, 3, 1, 2)$:

lavoro j	t_j	carichi (L_1, L_2, L_3)	prima macchina
4	8	(0, 0, 0)	1
5	8	(8, 0, 0)	2
6	6	(8, 8, 0)	3
7	6	(8, 8, 6)	3
3	4	(8, 8, 12)	1
1	2	(12, 8, 12)	2
2	2	(12, 10, 12)	2

Quindi $J_1 = \{3, 4\}$, $J_2 = \{1, 2, 5\}$, $J_3 = \{6, 7\}$, i carichi sono (12, 12, 12) e

$$LPT(I) = 12.$$

4. Il lower bound è

$$z(I) \geq \max \left\{ 8, \frac{2 + 2 + 4 + 8 + 8 + 6 + 6}{3} \right\} = \max \left\{ 8, \frac{36}{3} \right\} = 12.$$

LPT lo raggiunge, quindi $z(I) = 12$ e lo schedule di LPT è ottimo. I rapporti sono

$$\frac{LIST(I)}{z(I)} = \frac{16}{12} = \frac{4}{3} \leq 2 - \frac{1}{3} = \frac{5}{3}, \quad \frac{LPT(I)}{z(I)} = \frac{12}{12} = 1 \leq \frac{4}{3} - \frac{1}{9} = \frac{11}{9}.$$

Entrambe le garanzie valgono. Con la lista nell'ordine dei dati, i due lavori più lunghi arrivano quando le macchine sono già cariche, e $LIST$ supera l'ottimo di 4 unità; ordinare la lista li mette per primi e chiude il divario.

Esercizio 40.2 — Approssimazione di $P||C_{\max}$ 2

Consideriamo $m = 3$ macchine identiche e un insieme $\mathcal{J} = \{1, \dots, 7\}$ di 7 lavori ($n = 7$), con i seguenti tempi di esecuzione:

	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$	$j = 6$	$j = 7$
t_j	1	1	1	1	1	1	3

Domande

1. Applicare *LIST* con la lista $\lambda = (1, 2, \dots, 7)$ e indicare $LIST(I)$.
2. Determinare $z(I)$, esibendo uno schedule ottimo e un lower bound di pari valore.
3. Calcolare $\frac{LIST(I)}{z(I)}$ e confrontarlo con la garanzia di *LIST*. Stabilire a quale istanza delle note corrisponde.
4. Applicare *LPT* e indicare $LPT(I)$. Spiegare perché ordinare la lista elimina il caso sfavorevole.

Soluzione

	lavoro j	t_j	carichi (L_1, L_2, L_3)	prima macchina
1.	1	1	(0, 0, 0)	1
	2	1	(1, 0, 0)	2
	3	1	(1, 1, 0)	3
	4	1	(1, 1, 1)	1
	5	1	(2, 1, 1)	2
	6	1	(2, 2, 1)	3
	7	3	(2, 2, 2)	1

I carichi sono $(5, 2, 2)$, con $J_1 = \{1, 4, 7\}$, $J_2 = \{2, 5\}$, $J_3 = \{3, 6\}$, e

$$LIST(I) = 5.$$

2. Lo schedule $J_1 = \{7\}$, $J_2 = \{1, 2, 3\}$, $J_3 = \{4, 5, 6\}$ ha carichi $(3, 3, 3)$ e makespan 3. Il lower bound dà

$$z(I) \geq \max \left\{ 3, \frac{6 \cdot 1 + 3}{3} \right\} = \max \{3, 3\} = 3.$$

Quindi $z(I) = 3$.

- 3.

$$\frac{LIST(I)}{z(I)} = \frac{5}{3} = 2 - \frac{1}{3},$$

esattamente la garanzia $2 - \frac{1}{m}$ per $m = 3$: il bound di *LIST* è raggiunto. È l'istanza tight I_m delle note per $m = 3$: $m(m-1) = 6$ lavori unitari seguiti da un lavoro di

durata $m = 3$. I lavori unitari riempiono tutte le macchine fino a $m - 1 = 2$, e il lavoro lungo inizia sopra una di esse.

4. La lista ordinata è $\lambda = (7, 1, 2, 3, 4, 5, 6)$. Il lavoro 7 va sulla macchina 1, con carichi $(3, 0, 0)$; i lavori unitari vanno poi sulle macchine 2, 3, 2, 3, 2, 3 (la macchina 1 non è più la meno carica, poiché il suo carico 3 non è mai minore degli altri), con carichi $(3, 3, 3)$. Quindi

$$LPT(I) = 3 = z(I).$$

Il lavoro lungo è quello che fa danno quando arriva per ultimo; LPT lo assegna per primo, e i lavori brevi livellano i carichi intorno a esso.

Esercizio 40.3 — Approssimazione di $P || C_{\max}$ 3

Consideriamo $m = 4$ macchine identiche e un insieme $\mathcal{J} = \{1, \dots, 9\}$ di 9 lavori ($n = 9$), con i seguenti tempi di esecuzione:

	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$	$j = 6$	$j = 7$	$j = 8$	$j = 9$
t_j	7	7	6	6	5	5	4	4	4

Domande

1. Applicare LPT e indicare $LPT(I)$.
2. Determinare $z(I)$.
3. Calcolare $\frac{LPT(I)}{z(I)}$ e confrontarlo con la garanzia $\frac{4}{3} - \frac{1}{3m}$.
4. Stabilire quale dei due casi della dimostrazione della garanzia di LPT si applica al lavoro ℓ che termina per ultimo. Verificare che ogni disuguaglianza di quel caso valga all'uguaglianza su questa istanza.

Soluzione

1. I lavori sono già ordinati, $\lambda = (1, 2, \dots, 9)$:

lavoro j	t_j	carichi (L_1, L_2, L_3, L_4)	prima macchina
1	7	(0, 0, 0, 0)	1
2	7	(7, 0, 0, 0)	2
3	6	(7, 7, 0, 0)	3
4	6	(7, 7, 6, 0)	4
5	5	(7, 7, 6, 6)	3
6	5	(7, 7, 11, 6)	4
7	4	(7, 7, 11, 11)	1
8	4	(11, 7, 11, 11)	2
9	4	(11, 11, 11, 11)	1

Quindi $J_1 = \{1, 7, 9\}$, $J_2 = \{2, 8\}$, $J_3 = \{3, 5\}$, $J_4 = \{4, 6\}$, i carichi sono (15, 11, 11, 11) e

$$LPT(I) = 15.$$

2. Il tempo di esecuzione totale è 48, quindi $z(I) \geq \max\{7, \frac{48}{4}\} = 12$. Lo schedule $J_1 = \{1, 5\}$, $J_2 = \{2, 6\}$, $J_3 = \{3, 4\}$, $J_4 = \{7, 8, 9\}$ ha carichi (12, 12, 12, 12). Quindi $z(I) = 12$.

3.

$$\frac{LPT(I)}{z(I)} = \frac{15}{12} = \frac{5}{4} = \frac{4}{3} - \frac{1}{12} = \frac{4}{3} - \frac{1}{3 \cdot 4}.$$

La garanzia di LPT è raggiunta: è l'istanza tight delle note per $m = 4$, con due lavori per ciascuna durata $2m - 1 = 7$, 6 , $m + 1 = 5$ e tre lavori di durata $m = 4$; dopo $2m = 8$ lavori tutti i carichi valgono $3m - 1 = 11$, e l'ultimo lavoro dà $4m - 1 = 15$, mentre $z(I) = 3m = 12$.

4. L'ultimo lavoro è $\ell = 9$, con $t_9 = 4 = \frac{z(I)}{3}$, quindi si applica il Caso 1 ($t_\ell \leq \frac{z}{3}$), al suo confine. Le sue disuguaglianze sono:

$$\begin{aligned} LPT(I) &= S_9 + t_9 = 11 + 4 = 15, \\ \frac{1}{m} \sum_j t_j + \left(1 - \frac{1}{m}\right) t_9 &= \frac{48}{4} + \frac{3}{4} \cdot 4 = 15, \\ z(I) + \left(1 - \frac{1}{m}\right) \frac{z(I)}{3} &= 12 + \frac{3}{4} \cdot 4 = 15, \\ \left(\frac{4}{3} - \frac{1}{3m}\right) z(I) &= \frac{5}{4} \cdot 12 = 15. \end{aligned}$$

Tutte le quantità sono uguali: l'istante di inizio $S_9 = 11$ è pari alla media $\frac{48-4}{4} = 11$ degli altri carichi (le quattro macchine sono perfettamente bilanciate), $\frac{1}{m} \sum_j t_j = z(I)$, e $t_9 = \frac{z(I)}{3}$.