

# Problemi di packing

# Indice

Perché i problemi di packing?

Panoramica della famiglia

Il problema del bin packing

Il problema del cutting stock

Osservazioni

Errori tipici

In sintesi

# Obiettivi

Al termine di questo capitolo saremo in grado di:

- ▶ enunciare la definizione del problema del bin packing e del problema del cutting stock, con istanze e soluzioni ammissibili;
- ▶ spiegare le ipotesi sui dati;
- ▶ contare le soluzioni candidate di una piccola istanza e riconoscere che sono partizioni;
- ▶ trovare a mano una soluzione ammissibile e calcolare il bound sul peso;
- ▶ usare i due bound per certificare che una soluzione è ottima.

# Prerequisiti

- ▶ Da questo modulo, il capitolo «[Problemi di ottimizzazione combinatoria](#)»: istanze, soluzioni ammissibili, bound primali e duali, il gap e i numeri di Bell che contano le partizioni di un insieme.
- ▶ Da questo modulo, il capitolo «[Problemi di zaino](#)»: la forma delle definizioni e delle ipotesi sui dati.

# Perché i problemi di packing?

# Introduzione: Perché i problemi di packing?

- ▶ Prima delle definizioni formali, vediamo da dove nascono i problemi di packing
- ▶ Una storia dalla logistica e dall'informatica, e una breve storia dei primi risultati
- ▶ Il problema del cutting stock: lo stesso problema con oggetti in più copie identiche
- ▶ Una sfida su una piccola istanza, che risolviamo in questo capitolo

# Una storia

- ▶ Pacchi di pesi diversi devono essere spediti con camion identici di capacità  $c$ .
- ▶ Quanti camion servono, e quali pacchi viaggiano insieme?
- ▶ Lo stesso problema si presenta quando si collocano macchine virtuali su server identici.



# Un po' di storia

- ▶ Il comportamento nel caso peggiore di semplici regole di packing, come **First Fit**, è stato analizzato all'inizio degli anni Settanta:
  - ▶ da Garey, Graham e Ullman (1972);
  - ▶ da Johnson, Demers, Ullman, Garey e Graham (1974).
- ▶ Questi ultimi dimostrarono che First Fit usa al più  $\frac{17}{10}$  del numero ottimo di contenitori, più 2.
- ▶ Studiamo First Fit e la sua garanzia più avanti nel corso.

# Oggetti in più copie

- ▶ Il **problema del cutting stock** è lo stesso problema con oggetti che si presentano in più copie identiche.
- ▶ I contenitori sono rotoli o lastre di materiale.
- ▶ Gli oggetti sono i pezzi da tagliare.

# Una sfida

- ▶ Prima di proseguire, proviamo a rispondere a due domande sull'istanza di riferimento del capitolo: contenitori di capacità  $c = 10$  e cinque oggetti con pesi

| Oggetti |       | $j = 1$ | $j = 2$ | $j = 3$ | $j = 4$ | $j = 5$ |
|---------|-------|---------|---------|---------|---------|---------|
| Pesi    | $w_j$ | 6       | 5       | 4       | 7       | 3       |

- ▶ Quanti contenitori servono?
- ▶ Come possiamo dimostrare che meno contenitori non bastano, senza elencare tutte le partizioni?
- ▶ Qui la seconda domanda ha una risposta semplice.

## Da dove nascono i problemi di packing:

- ▶ pacchi su camion identici, macchine virtuali su server identici
- ▶ pezzi tagliati da rotoli o lastre identici: il problema del cutting stock

## Storia:

- ▶ semplici regole di packing, come First Fit, analizzate dall'inizio degli anni Settanta

## La sfida:

- ▶ quanti contenitori, e come dimostrare che meno contenitori non bastano

# Panoramica della famiglia

# Introduzione: Panoramica della famiglia

- ▶ In questa sezione presentiamo la famiglia dei **problemi di packing**
- ▶ Tutti i suoi membri hanno gli stessi dati e lo stesso obiettivo
- ▶ Differiscono solo nel numero di copie di ogni oggetto da impaccare
- ▶ Questo quadro comune rende facile confrontare i due problemi del capitolo

# I problemi di packing

- ▶ I **problemi di packing** hanno gli stessi dati: un numero infinito di contenitori identici di *capacità*  $c \in \mathbb{Z}_{\geq 1}$ , un insieme  $\mathcal{I} = \{1, 2, \dots, n\}$  di  $n$  oggetti e, per ogni oggetto  $j \in \mathcal{I}$ , un *peso*  $w_j \in \mathbb{Z}_{\geq 1}$ .
- ▶ Una soluzione assegna ogni oggetto, o ogni copia richiesta di un oggetto, a un contenitore: il peso totale degli oggetti in un contenitore non deve superare la capacità  $c$ .
- ▶ L'obiettivo è minimizzare il numero di contenitori usati.

# La famiglia in una frase

Un problema di packing sistema tutti gli oggetti nel minor numero di contenitori di capacità  $c$ .

I membri della famiglia differiscono nel numero di copie di ogni oggetto da impaccare.

# I membri della famiglia

- La tabella elenca i membri della famiglia, con la sezione che definisce ciascuno di essi.

| Problema                   | Sigla | Oggetti da impaccare         | Sezione                       |
|----------------------------|-------|------------------------------|-------------------------------|
| problema del bin packing   | BPP   | una copia di ogni oggetto    | Il problema del bin packing   |
| problema del cutting stock | CSP   | $d_j$ copie dell'oggetto $j$ | Il problema del cutting stock |

# Riepilogo

## Dati comuni:

- ▶ contenitori identici di capacità  $c$ , oggetti  $\mathcal{J} = \{1, 2, \dots, n\}$  con pesi  $w_j$

## Obiettivo comune:

- ▶ impaccare tutto senza superare  $c$  in alcun contenitore, usando il numero minimo di contenitori

## Due membri:

- ▶ BPP: una copia di ogni oggetto
- ▶ CSP:  $d_j$  copie dell'oggetto  $j$

# Il problema del bin packing

# Introduzione: Il problema del bin packing

- ▶ Definiamo il **problema del bin packing** (BPP), le sue istanze e le sue soluzioni ammissibili
- ▶ Una soluzione ammissibile è una partizione degli oggetti in schemi di carico, uno per contenitore
- ▶ Sull'istanza di riferimento del capitolo contiamo le soluzioni, troviamo a mano una prima soluzione e calcoliamo un semplice bound duale
- ▶ I due bound insieme dimostrano che la soluzione trovata a mano è ottima

# Il problema del bin packing (BPP)

## Definizione 1: problema del bin packing

*Dati:*

1. *un numero infinito di contenitori, ciascuno di capacità  $c \in \mathbb{Z}_{\geq 1}$ ,*
2. *un insieme  $\mathcal{J} = \{1, 2, \dots, n\}$  di  $n$  oggetti,*
3. *per ogni oggetto  $j \in \mathcal{J}$ , un peso  $w_j \in \mathbb{Z}_{\geq 1}$ ,*

*il **problema del bin packing** (BPP) chiede di decidere in quale contenitore sistemare ogni oggetto, in modo che il peso totale degli oggetti in ogni contenitore non superi la capacità del contenitore e il numero totale di contenitori usati sia minimo.*

# Istanza del BPP

► Un'istanza del BPP è specificata da:

1. due numeri interi:

la capacità dei contenitori  $c \in \mathbb{Z}_{\geq 1}$ ,  
il numero di oggetti  $n \in \mathbb{Z}_{\geq 1}$ ,

2. un vettore di numeri interi di lunghezza  $n$ :

$w \in \mathbb{Z}_{\geq 1}^n$ , che contiene i pesi degli oggetti.

# Dall'istanza alle soluzioni

Una soluzione dice quali oggetti viaggiano insieme:  
un sottoinsieme di oggetti per ogni contenitore usato.

Questi sottoinsiemi sono gli **schemi di carico**.

# Soluzione ammissibile del BPP (1/2)

- Una **soluzione ammissibile del BPP** è una famiglia di **schemi di carico**, ossia sottoinsiemi di oggetti ciascuno assegnato a un contenitore,

$$\mathcal{P} \subseteq 2^{\mathcal{I}}$$

tale che:

- (i)  $\mathcal{P}$  forma una partizione di  $\mathcal{I}$  in schemi di carico *non vuoti*,  
ossia

$$\begin{aligned} P &\neq \emptyset, \quad \text{per ogni } P \in \mathcal{P}, \\ P \cap P' &= \emptyset, \quad \text{per ogni } P, P' \in \mathcal{P} \text{ con } P \neq P', \\ \bigcup_{P \in \mathcal{P}} P &= \mathcal{I}; \end{aligned}$$

## Soluzione ammissibile del BPP (2/2)

- (ii) il peso totale di ogni schema di carico in  $\mathcal{P}$  rispetta la capacità del contenitore, ossia

$$\sum_{j \in P} w_j \leq c, \quad \text{per ogni } P \in \mathcal{P}.$$

- Di conseguenza, il BPP si può enunciare in modo equivalente come il problema di determinare una partizione dell'insieme degli oggetti in schemi di carico tale che il peso totale degli oggetti in ogni schema di carico non superi la capacità del contenitore e il numero di schemi di carico (contenitori) sia minimo.

# Enunciato formale del BPP

Data un'istanza del BPP, il BPP si può enunciare formalmente come:

$$\tilde{\mathcal{P}} \in \arg \min_{\mathcal{P} \subseteq 2^{\mathcal{I}}} \left\{ |\mathcal{P}| : \begin{array}{l} P \neq \emptyset, \quad \forall P \in \mathcal{P}, \\ P \cap P' = \emptyset, \quad \forall P, P' \in \mathcal{P}, P \neq P', \\ \bigcup_{P \in \mathcal{P}} P = \mathcal{I}, \\ \sum_{j \in P} w_j \leq c, \quad \forall P \in \mathcal{P} \end{array} \right\}.$$

## Esempio (1/2)

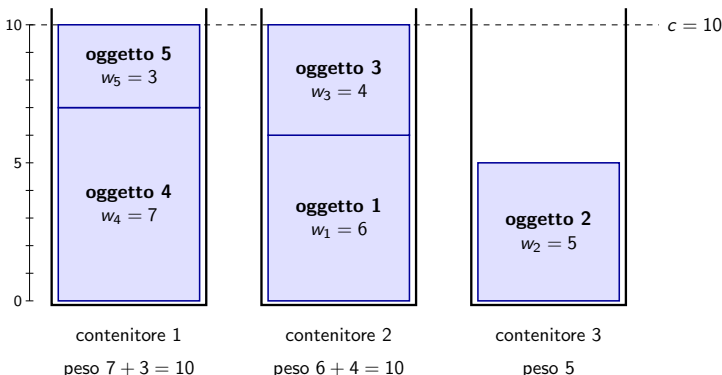
- Consideriamo contenitori di capacità  $c = 10$  e un insieme  $\mathcal{I} = \{1, 2, 3, 4, 5\}$  di 5 oggetti ( $n = 5$ ), con i seguenti pesi:

| Oggetti |       | $j = 1$ | $j = 2$ | $j = 3$ | $j = 4$ | $j = 5$ |
|---------|-------|---------|---------|---------|---------|---------|
| Pesi    | $w_j$ | 6       | 5       | 4       | 7       | 3       |

- Chiamiamo questa istanza del BPP l'**istanza di riferimento** del capitolo.

## Esempio (2/2)

Nel disegno ogni contenitore è una scatola di altezza  $c = 10$  e ogni oggetto un blocco di altezza  $w_j$ . I tre contenitori contengono  $\{4, 5\}$ ,  $\{1, 3\}$  e  $\{2\}$ ; mostriamo più avanti che 3 contenitori sono il minimo.



# Ipotesi

► Facciamo le seguenti ipotesi:

1. Il peso totale di tutti gli oggetti supera la capacità di un contenitore (altrimenti tutti gli oggetti entrerebbero in un solo contenitore). Formalmente:

$$\sum_{j \in \mathcal{J}} w_j > c$$

2. Ogni oggetto, da solo, entra in un contenitore (altrimenti non può essere impaccato). Formalmente:

$$w_j \leq c, \quad \forall j \in \mathcal{J}$$

# Contare le soluzioni

- ▶ **Soluzioni candidate.** Le soluzioni candidate sono le partizioni di  $\mathcal{I}$ : sono contate dai numeri di Bell (capitolo «[Problemi di ottimizzazione combinatoria](#)»). Per l'istanza di riferimento abbiamo  $B(5) = 52$  partizioni.
- ▶ **Soluzioni ammissibili.** Una partizione è ammissibile se ogni contenitore pesa al più 10. L'enumerazione dà 11 partizioni ammissibili:
  - ▶ 1 con 5 contenitori (un oggetto per contenitore),
  - ▶ 6 con 4 contenitori,
  - ▶ 4 con 3 contenitori.

Due contenitori non bastano, poiché il peso totale 25 supera  $2 \cdot 10$ .

# Una prima soluzione, a occhio

- ▶ Accoppiamo gli oggetti pesanti con quelli leggeri:
  - ▶ l'oggetto 4 (peso 7) con l'oggetto 5 (peso 3),
  - ▶ l'oggetto 1 (peso 6) con l'oggetto 3 (peso 4);
  - ▶ l'oggetto 2 (peso 5) resta da solo.

- ▶ La soluzione

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

usa 3 contenitori, di pesi 10, 10 e 5: è un **bound primale**,

$$\text{UB} = 3 \geq z(I).$$

# Da un bound primale a un bound duale

La soluzione trovata a occhio usa 3 contenitori.

È il minimo? Serve un bound che valga  
per **ogni** soluzione ammissibile: un bound duale.

## Il bound sul peso

### Osservazione 1: bound sul peso

*Ogni soluzione ammissibile usa almeno*

$$\left\lceil \frac{\sum_{j \in \mathcal{J}} w_j}{c} \right\rceil$$

*contenitori, quindi questo valore è un bound duale.*

# Dimostrazione

- ▶ Se una soluzione usa  $k$  contenitori, il peso totale che trasporta è al più  $k c$ .
- ▶ Il peso totale degli oggetti è  $\sum_j w_j$ , quindi

$$k c \geq \sum_j w_j.$$

- ▶ Il numero  $k$  è intero, quindi

$$k \geq \left\lceil \frac{\sum_j w_j}{c} \right\rceil.$$



## Il bound duale e il gap

- ▶ Nell'istanza di riferimento abbiamo  $\sum_j w_j = 25$  e  $c = 10$ , quindi

$$LB = \lceil 25/10 \rceil = 3.$$

- ▶ I due bound coincidono: il gap è nullo, la soluzione trovata a occhio è ottima e  $z(I) = 3$ , senza enumerare le partizioni.
- ▶ La soluzione ottima non è unica: ci sono 4 partizioni ottime,

$$\begin{aligned} & \{\{1\}, \{2, 3\}, \{4, 5\}\}, \quad \{\{2\}, \{1, 3\}, \{4, 5\}\}, \\ & \{\{4\}, \{1, 5\}, \{2, 3\}\}, \quad \{\{4\}, \{1, 3\}, \{2, 5\}\}. \end{aligned}$$

# Riepilogo (1/2)

## Il problema del bin packing:

- ▶ un'istanza è data da  $c$ ,  $n$  e dai pesi  $w$
- ▶ una soluzione ammissibile è una partizione di  $\mathcal{I}$  in schemi di carico non vuoti
- ▶ ogni schema di carico pesa al più  $c$ ; il numero di schemi si minimizza

## Ipotesi:

- ▶  $\sum_{j \in \mathcal{I}} w_j > c$  e  $w_j \leq c$  per ogni oggetto  $j$

## Riepilogo (2/2)

### Sull'istanza di riferimento:

- ▶  $B(5) = 52$  soluzioni candidate, di cui 11 ammissibili
- ▶ una soluzione trovata a occhio dà  $UB = 3$

### Il bound sul peso:

- ▶  $\lceil \sum_{j \in \mathcal{J}} w_j / c \rceil$  è un bound duale, qui  $LB = 3$
- ▶  $LB = UB$ : il gap è nullo e  $z(I) = 3$

# Il problema del cutting stock

# Introduzione: Il problema del cutting stock

- ▶ Definiamo il **problema del cutting stock (CSP)**: ogni oggetto va impaccato in un numero dato di copie identiche
- ▶ Le copie diventano gli elementi di un insieme esteso degli oggetti, e una soluzione è di nuovo una partizione
- ▶ Sull'istanza di riferimento del CSP ripetiamo i passi del BPP: contiamo le soluzioni, ne troviamo una a occhio e calcoliamo il bound
- ▶ Il bound sul peso del BPP vale ancora, con i pesi di tutte le copie

# Il problema del cutting stock (CSP)

## Definizione 2: problema del cutting stock

*Dati:*

1. *un numero infinito di contenitori, ciascuno di capacità  $c \in \mathbb{Z}_{\geq 1}$ ,*
2. *un insieme  $\mathcal{J} = \{1, 2, \dots, n\}$  di  $n$  oggetti,*
3. *per ogni oggetto  $j \in \mathcal{J}$ , un peso  $w_j \in \mathbb{Z}_{\geq 1}$  e un numero di copie richieste  $d_j \in \mathbb{Z}_{\geq 1}$ ,*

*il **problema del cutting stock (CSP)** chiede di decidere in quale contenitore sistemare ogni copia di ogni oggetto, in modo che il peso totale delle copie in ogni contenitore non superi la capacità del contenitore, che il numero richiesto di copie di ogni oggetto sia sistemato nei contenitori e che il numero totale di contenitori usati sia minimo.*

# Istanza del CSP

► Un'istanza del CSP è specificata da:

1. due numeri interi:

la capacità dei contenitori  $c \in \mathbb{Z}_{\geq 1}$ ,

il numero di oggetti  $n \in \mathbb{Z}_{\geq 1}$ ,

2. due vettori di numeri interi di lunghezza  $n$ :

$w \in \mathbb{Z}_{\geq 1}^n$ , che contiene i pesi degli oggetti,

$d \in \mathbb{Z}_{\geq 1}^n$ , che contiene il numero di copie richieste di ogni oggetto.

# Perché «cutting stock»?

- ▶ Il problema si chiama *cutting stock* (taglio del materiale) perché, nella sua applicazione classica, i contenitori corrispondono a grandi rotoli di materiale (o lastre).
- ▶ I rotoli devono essere tagliati per produrre le copie richieste degli oggetti.
- ▶ Le copie richieste sono dette anche **domande degli oggetti**.

# L'insieme esteso degli oggetti

- Data un'istanza del CSP, definiamo l'**insieme esteso degli oggetti** come

$$\hat{\mathcal{J}} = \{ (j, k) : j \in \mathcal{J}, k \in \{1, 2, \dots, d_j\} \},$$

dove ogni coppia  $(j, k)$  rappresenta la  $k$ -esima copia richiesta dell'oggetto  $j$ .

- La cardinalità dell'insieme esteso degli oggetti è

$$|\hat{\mathcal{J}}| = \sum_{j \in \mathcal{J}} d_j.$$

# Soluzione ammissibile del CSP (1/2)

- Una **soluzione ammissibile del CSP** è una famiglia di **schemi di carico**, ossia sottoinsiemi di copie ciascuno assegnato a un contenitore,

$$\mathcal{P} \subseteq 2^{\hat{\mathcal{J}}}$$

tale che:

- (i)  $\mathcal{P}$  forma una partizione di  $\hat{\mathcal{J}}$  in schemi di carico *non vuoti*,  
ossia

$$\begin{aligned} P &\neq \emptyset, \quad \text{per ogni } P \in \mathcal{P}, \\ P \cap P' &= \emptyset, \quad \text{per ogni } P, P' \in \mathcal{P} \text{ con } P \neq P', \\ \bigcup_{P \in \mathcal{P}} P &= \hat{\mathcal{J}}; \end{aligned}$$

## Soluzione ammissibile del CSP (2/2)

- (ii) il peso totale delle copie in ogni schema di carico non supera la capacità del contenitore, ossia

$$\sum_{(j,k) \in P} w_j \leq c, \quad \text{per ogni } P \in \mathcal{P}.$$

- Di conseguenza, il CSP si può enunciare in modo equivalente come il problema di determinare una partizione dell'insieme esteso degli oggetti in schemi di carico tale che il peso totale in ogni schema di carico non superi la capacità del contenitore e il numero di schemi di carico (contenitori) sia minimo.

# Enunciato formale del CSP

Data un'istanza del CSP, il CSP si può enunciare formalmente come:

$$\tilde{\mathcal{P}} \in \arg \min_{\mathcal{P} \subseteq 2^{\mathcal{J}}} \left\{ |\mathcal{P}| : \begin{array}{l} P \neq \emptyset, \quad \forall P \in \mathcal{P}, \\ P \cap P' = \emptyset, \quad \forall P, P' \in \mathcal{P}, P \neq P', \\ \bigcup_{P \in \mathcal{P}} P = \hat{\mathcal{J}}, \\ \sum_{(j,k) \in P} w_j \leq c, \quad \forall P \in \mathcal{P} \end{array} \right\}.$$

## Esempio (1/2)

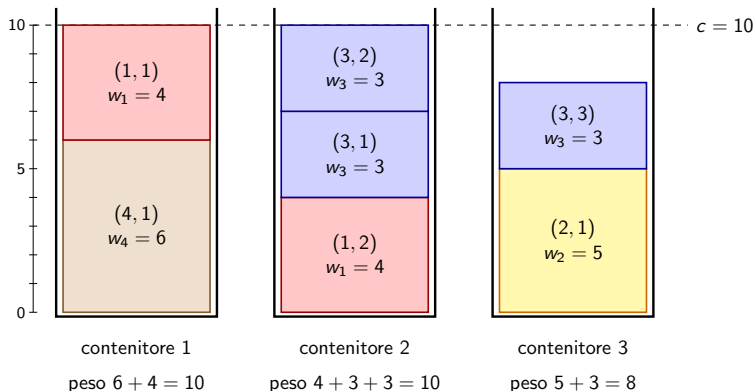
- Consideriamo contenitori di capacità  $c = 10$  e un insieme  $\mathcal{J} = \{1, 2, 3, 4\}$  di 4 oggetti ( $n = 4$ ), con i seguenti pesi e copie richieste:

| Oggetti |       | $j = 1$ | $j = 2$ | $j = 3$ | $j = 4$ |
|---------|-------|---------|---------|---------|---------|
| Pesi    | $w_j$ | 4       | 5       | 3       | 6       |
| Copie   | $d_j$ | 2       | 1       | 3       | 1       |

- Chiamiamo questa istanza del CSP l'**istanza di riferimento del CSP** nel capitolo.

## Esempio (2/2)

Nel disegno le copie dello stesso oggetto hanno lo stesso colore, e la copia  $(j, k)$  è un blocco di altezza  $w_j$ . Le 7 copie entrano in 3 contenitori, di pesi 10, 10 e 8.



► Facciamo le seguenti ipotesi:

1. Il peso totale di tutte le copie richieste supera la capacità di un contenitore (altrimenti tutte le copie entrerebbero in un solo contenitore). Formalmente:

$$\sum_{j \in \mathcal{J}} d_j w_j > c.$$

2. Ogni oggetto, da solo, entra in un contenitore (altrimenti nessuna copia di quell'oggetto può essere impaccata). Formalmente:

$$w_j \leq c, \quad \forall j \in \mathcal{J}.$$

# Contare le soluzioni

- **Soluzioni candidate.** L'istanza di riferimento del CSP ha

$$2 + 1 + 3 + 1 = 7$$

copie.

- Le partizioni dell'insieme esteso degli oggetti sono contate da

$$B(7) = 877,$$

dove copie identiche sono contate come oggetti diversi.

- 262 di esse sono ammissibili.

# Una prima soluzione, a occhio

- ▶ Riempiamo tre contenitori:
  - ▶ accoppiamo l'oggetto 4 (peso 6) con una copia dell'oggetto 1 (peso 4),
  - ▶ mettiamo l'altra copia dell'oggetto 1 con due copie dell'oggetto 3 ( $4 + 3 + 3$ ),
  - ▶ e l'ultima copia dell'oggetto 3 con l'oggetto 2 ( $3 + 5$ ).
- ▶ La soluzione

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

usa 3 contenitori, di pesi 10, 10 e 8, quindi

$$UB = 3.$$

## Il bound duale e il gap

- Il bound sul peso del bin packing vale con i pesi di tutte le copie:

$$\text{LB} = \left\lceil \frac{\sum_j d_j w_j}{c} \right\rceil = \left\lceil \frac{28}{10} \right\rceil = 3.$$

- Il gap è nullo, quindi la soluzione è ottima e  $z(I) = 3$ .

## Il problema del cutting stock:

- ▶ ogni oggetto  $j$  è richiesto in  $d_j$  copie identiche
- ▶ una soluzione ammissibile è una partizione dell'insieme esteso  $\hat{\mathcal{J}}$  in schemi di carico di peso al più  $c$
- ▶ il BPP è il caso  $d_j = 1$  per ogni oggetto

## Sull'istanza di riferimento del CSP:

- ▶ 7 copie,  $B(7) = 877$  soluzioni candidate, 262 ammissibili
- ▶ a occhio  $UB = 3$ , bound sul peso  $LB = \lceil 28/10 \rceil = 3$ :  
 $z(I) = 3$

# Osservazioni

# Introduzione: Osservazioni

- ▶ Alcune osservazioni storiche su semplici regole di packing
- ▶ Regole semplici, come First Fit, danno rapidamente una soluzione: ma quanto lontana dall'ottimo?
- ▶ Le prime risposte risalgono all'inizio degli anni Settanta

# Il caso peggiore di First Fit

- ▶ Il comportamento nel caso peggiore di semplici regole di packing, come First Fit, è stato analizzato da Garey, Graham e Ullman (1972) e da Johnson, Demers, Ullman, Garey e Graham (1974).
- ▶ Johnson et al. (1974) dimostrarono che First Fit usa al più  $\frac{17}{10}$  del numero ottimo di contenitori, più 2.

## Semplici regole di packing:

- ▶ il loro comportamento nel caso peggiore è studiato dall'inizio degli anni Settanta

## First Fit:

- ▶ al più  $\frac{17}{10}$  del numero ottimo di contenitori, più 2 (Johnson et al., 1974)

# Errori tipici

# Introduzione: Errori tipici

- ▶ Raccogliamo gli errori più frequenti sui problemi di packing
- ▶ Due riguardano i bound, due il modo di contare le soluzioni, uno le ipotesi sui dati
- ▶ Riconoscerli evita bound sbagliati e conteggi sbagliati



## Osservazione 2: i bound

- ▶ ***Credere che il bound sul peso sia sempre stretto.***  
*Con tre oggetti di peso 6 e  $c = 10$  il bound dà  $\lceil 18/10 \rceil = 2$ , ma due oggetti non entrano insieme, quindi servono 3 contenitori.*
- ▶ ***Contare i contenitori solo in base ai pesi.*** *Il bound sul peso ignora come gli oggetti si combinano; è un bound duale, e il bound primale richiede comunque una soluzione.*

### Osservazione 3: le soluzioni e i dati

- ▶ ***Elencare due volte lo stesso riempimento.*** Nel problema del cutting stock le copie di un oggetto sono intercambiabili: partizioni che differiscono solo per lo scambio di copie identiche sono lo stesso riempimento.
- ▶ ***Ignorare che l'ordine dei contenitori non conta.*** Una soluzione è una partizione:  $\{\{4, 5\}, \{1, 3\}, \{2\}\}$  e  $\{\{2\}, \{4, 5\}, \{1, 3\}\}$  sono la stessa soluzione.
- ▶ ***Dimenticare le ipotesi.*** Un oggetto con  $w_j > c$  non entra in alcun contenitore, e se il peso totale è al più  $c$  basta un contenitore.

# Riepilogo

## Bound:

- ▶ il bound sul peso è un bound duale, non sempre stretto
- ▶ un bound primale richiede sempre una soluzione ammissibile

## Soluzioni:

- ▶ una soluzione è una partizione: l'ordine dei contenitori non conta
- ▶ nel CSP, scambiare copie identiche dà lo stesso riempimento

## Dati:

- ▶ controllare  $w_j \leq c$  e  $\sum_j w_j > c$

# In sintesi

# Introduzione: In sintesi

- ▶ Raccogliamo le idee chiave del capitolo
- ▶ I due problemi, la struttura delle loro soluzioni e i due bound
- ▶ Un quadro riassuntivo con i bound delle due istanze di riferimento

## Riepilogo (1/2)

- ▶ Il BPP partiziona gli oggetti nel numero minimo di contenitori di capacità  $c$ ; il problema del cutting stock è il BPP con più copie identiche di ogni oggetto, e il BPP è il caso  $d_j = 1$ .
- ▶ Una soluzione è una partizione, quindi le candidate sono contate dai numeri di Bell e crescono molto rapidamente.
- ▶ Una soluzione trovata a occhio è un bound primale; il bound sul peso  $\lceil \sum_j w_j / c \rceil$  è un bound duale.
- ▶ Quando i due bound coincidono, il gap è nullo e la soluzione è certificata ottima.

### Quadro riassuntivo delle istanze di riferimento.

| istanza (min)                       | lower bound        | upper bound  | da              |
|-------------------------------------|--------------------|--------------|-----------------|
| BPP, istanza di riferimento         | 3 (bound sul peso) | 3 (a occhio) | questo capitolo |
| CSP, istanza di riferimento del CSP | 3 (bound sul peso) | 3 (a occhio) | questo capitolo |