Universidade Federal de São Carlos
Departamento de Engenharia de Produção

Otimização Linear Contínua e Discreta (Tópicos Avançados em PCSP)

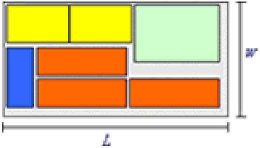
PPGEP, UFSCar - Semestre 01/2022
Prof. Dr. Pedro Munari (munari@dep.ufscar.br)

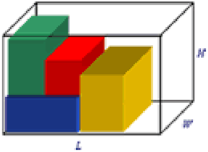
Tópico 13.3: Exemplo de aplicação da Decomposição de Dantzig-Wolfe: Problema de Corte de Estoque

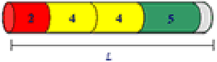
Objetivos deste tópico

- Entender como aplicar a decomposição de Dantzig-Wolfe no Problema de Corte de Estoque, que é um caso clássico da literatura.

O problema de corte de estoque (PCE)







- 1D: Barras de aço, rolos de filmes, bobinas de papel, etc.
- 2D: Placas de madeira, tecido, chapas de aço, etc.
- 3D: Placas com diferentes espessuras, espumas para colchões, etc.

PCE unidimensional

- n barras de tamanho L disponíveis em estoque;
- Produzir m tipos de itens com tamanhos $l_1, l_2, \ldots, l_m$;
- Demandas $d_1, d_2, \ldots, d_m$;
- O objetivo é minimizar o número de barras usadas.

Variáveis de decisão:

- Índices: $I = \{1, \ldots, n\}, \quad J = \{1, \ldots, m\}$;
- x_{ij}: número de vezes que o item j é cortado da barra i;
- y_i: $1 \rightarrow$ barra i é usada; $0 \rightarrow$ caso contrário.

PCE unidimensional

Variáveis de decisão:

- Índices: $I = \{1, \ldots, n\}, \quad J = \{1, \ldots, m\}$;
- x_{ij}: número de vezes que o item j é cortado da barra i;
- y_i: $1 \rightarrow$ barra i é usada; $0 \rightarrow$ caso contrário.

Restrições:

- Atendimento da demanda: $\sum_{i=1}^{n} x_{ij} \geq d_j, \; j = 1, \ldots, m$;
- Tamanho da barra: $\sum_{j=1}^{m} l_j x_{ij} \leq L y_i, \; i = 1, \ldots, n$;
- Domínio das variáveis: $y_i \in \{0, 1\}, \; x_{ij} \in \mathbb{Z}_+$.

PCE unidimensional

Variáveis de decisão:

- Índices: $I = \{1, \ldots, n\}, \quad J = \{1, \ldots, m\}$;
- x_{ij}: número de vezes que o item j é cortado da barra i;
- y_i: $1 \rightarrow$ barra i é usada; $0 \rightarrow$ caso contrário.

Função objetivo:

- Minimizar o número de barras usadas:

$$\min \sum_{i=1}^{n} y_i$$

PCE unidimensional

$$
\begin{aligned}
\text{min} \quad & \sum_{i=1}^{n} y_i \\
\text{s.a} \quad & \sum_{i=1}^{n} x_{ij} \geq d_j, && j = 1, \ldots, m, \\
& \sum_{j=1}^{m} l_j x_{ij} \leq L y_i, && i = 1, \ldots, n, \\
& x_{ij} \in \mathbb{Z}_+, && i = 1, \ldots, n; \quad j = 1, \ldots, m, \\
& y_i \in \{0, 1\}, && i = 1, \ldots, n.
\end{aligned}
$$

PCE unidimensional

▷ Exemplo

Uma fábrica de papel, trabalha com a produção de bobinas para impressoras. As encomendas para a próxima semana correspondem a 32 bobinas de 5 cm de largura, 16 bobinas de 7 cm de largura e 21 bobinas de 7,5 cm de largura. Essas bobinas são cortadas a partir de bobinas grandes disponíveis em estoques, todas com largura de 20 cm. Determine qual a melhor forma de atender às encomendas de modo a minimizar o número de bobinas grandes utilizadas.

PCE unidimensional: Exemplo

$$
\begin{array}{rl}
\text{min} & y_1 + y_2 + \ldots + y_{100} \\
\text{s.a} & x_{11} + x_{21} + \ldots + x_{100,1} \geq 32, \\
& x_{12} + x_{22} + \ldots + x_{100,2} \geq 16, \\
& x_{13} + x_{23} + \ldots + x_{100,3} \geq 21, \\
& 5x_{11} + 7x_{12} + 7{,}5x_{13} \leq 20y_1, \\
& 5x_{21} + 7x_{22} + 7{,}5x_{23} \leq 20y_2, \\
& \vdots \qquad\qquad \vdots \\
& 5x_{100,1} + 7x_{100,2} + 7{,}5x_{100,3} \leq 20y_{100}, \\
& x_{11}, x_{12}, \ldots, x_{100,3} \in \mathbb{Z}_+, \\
& y_1, y_2, \ldots, y_{100} \in \{0, 1\}.
\end{array}
$$

PCE unidimensional

$$
\begin{aligned}
\min \quad & \sum_{i=1}^{n} y_i \\
\text{s.a} \quad & \sum_{i=1}^{n} x_{ij} \geq d_j, && j = 1, \ldots, m, \\
& \sum_{j=1}^{m} l_j x_{ij} \leq L y_i, && i = 1, \ldots, n, \\
& x_{ij} \in \mathbb{Z}_+, && i = 1, \ldots, n; \quad j = 1, \ldots, m, \\
& y_i \in \{0, 1\}, && i = 1, \ldots, n.
\end{aligned}
$$

PCE unidimensional

$$
\begin{aligned}
\text{min} \quad & \sum_{i=1}^{n} y_i \\
\text{s.a} \quad & \sum_{i=1}^{n} x_{ij} \geq d_j, && j = 1, \ldots, m, \\
& \sum_{j=1}^{m} l_j x_{ij} \leq L y_i, && i = 1, \ldots, n, \\
& x_{ij} \in \mathbb{Z}_+, && i = 1, \ldots, n; \;\; j = 1, \ldots, m, \\
& y_i \in \{0, 1\}, && i = 1, \ldots, n.
\end{aligned}
$$

PCE unidimensional

$$\min \quad \sum_{i=1}^{n} y_i$$

$$\text{s.a} \quad \sum_{i=1}^{n} x_{ij} \geq d_j, \quad j = 1, \ldots, m,$$

$$(x_i, y_i) \in \mathcal{X}^i, \quad i = 1, \ldots, n; \;\; j = 1, \ldots, m.$$

PCE unidimensional

$$
\begin{aligned}
\text{min} \quad & \sum_{i=1}^{n} y_i \\
\text{s.a} \quad & \sum_{i=1}^{n} x_{ij} \geq d_j, \quad j = 1, \dots, m, \\
& (x_i, y_i) \in \mathcal{X}^i, \quad i = 1, \dots, n; \;\; j = 1, \dots, m.
\end{aligned}
$$

Para cada $i = 1, \dots, n$:

$$
\mathcal{X}^i = \left\{ (x_i, y_i) \mid \begin{aligned} & \sum_{j=1}^{m} l_j x_{ij} \leq L y_i, \\ & y_i \in \{0, 1\} \\ & x_{ij} \in \mathbb{Z}_+, \qquad j = 1, \dots, m \end{aligned} \right\}
$$

PCE unidimensional

▷ Decomposição de Dantzig-Wolfe (convexificação)

PCE unidimensional

▷ Decomposição de Dantzig-Wolfe (convexificação)

- Reescrevendo (x_i, y_i) usando todos os pontos extremos $(\bar{x}_q^i, \bar{y}_q^i)$ de $conv(\mathcal{X}^i)$:

PCE unidimensional

▷ Decomposição de Dantzig-Wolfe (convexificação)

- Reescrevendo (x_i, y_i) usando todos os pontos extremos $(\bar{x}_q^i, \bar{y}_q^i)$ de $conv(\mathcal{X}^i)$:

$$
\begin{aligned}
\text{min} \quad & \sum_{i=1}^{n} \sum_{q \in Q^i} \bar{y}_q^i \lambda_q^i \\
\text{s.a} \quad & \sum_{i=1}^{n} \sum_{q \in Q^i} \bar{x}_{qj}^i \lambda_q^i \geq d_j, && j = 1, \dots, m, \\
& \sum_{q \in Q^i} \lambda_q^i = 1, && i = 1, \dots, n, \\
& \lambda_q^i \geq 0, && i = 1, \dots, n, \; q \in Q^i. \\
& x_i \in \mathbb{Z}_+^m, y_i \in \{0, 1\} && i = 1, \dots, n, \\
& (x_i, y_i) = \sum_{q \in Q^i} \lambda_q^i (\bar{x}_q^i, \bar{y}_q^i) && i = 1, \dots, n.
\end{aligned}
$$

PCE unidimensional

▷ Decomposição de Dantzig-Wolfe (discretização)

PCE unidimensional

▷ Decomposição de Dantzig-Wolfe (discretização)

- Reescrevendo (x_i, y_i) usando todos os pontos inteiros $(\bar{x}_q^i, \bar{y}_q^i)$ de $\mathcal{X}^i$:

PCE unidimensional

▷ Decomposição de Dantzig-Wolfe (discretização)

- Reescrevendo (x_i, y_i) usando todos os pontos inteiros $(\bar{x}^i_q, \bar{y}^i_q)$ de $\mathcal{X}^i$:

$$\begin{aligned}
\min \quad & \sum_{i=1}^{n} \sum_{q \in Q^i} \bar{y}^i_q \lambda^i_q \\
\text{s.a} \quad & \sum_{i=1}^{n} \sum_{q \in Q^i} \bar{x}^i_{qj} \lambda^i_q \geq d_j, \quad && j = 1, \ldots, m, \\
& \sum_{q \in Q^i} \lambda^i_q = 1, && i = 1, \ldots, n, \\
& \lambda^i_q \in \{0, 1\}, && i = 1, \ldots, n, \ q \in Q^i.
\end{aligned}$$

PCE unidimensional

▷ Decomposição de Dantzig-Wolfe

- Relaxação linear do Problema Mestre (convexificação/discretização):

$$
\begin{aligned}
\text{min} \quad & \sum_{i=1}^{n} \sum_{q \in Q^i} \bar{y}_q^i \lambda_q^i \\
\text{s.a} \quad & \sum_{i=1}^{n} \sum_{q \in Q^i} \bar{x}_{qj}^i \lambda_q^i \geq d_j, \quad && j = 1, \dots, m, \\
& \sum_{q \in Q^i} \lambda_q^k = 1, && i = 1, \dots, n, \\
& \lambda_q^i \geq 0, && i = 1, \dots, n, \; q \in Q^i.
\end{aligned}
$$

PCE unidimensional

▷ Decomposição de Dantzig-Wolfe

- Quando as barras são idênticas temos $\bar{\mathcal{X}} = \mathcal{X}^1 = \ldots = \mathcal{X}^n$ e, portanto, podemos usar agregação, obtendo:

$$
\begin{aligned}
\text{min} \quad & \sum_{q \in Q} \bar{y}_q \lambda_q \\
\text{s.a} \quad & \sum_{q \in Q} \bar{x}_{qj} \lambda_q \geq d_j, \quad j = 1, \ldots, m, \\
& \sum_{q \in Q} \lambda_q^k = n, \\
& \lambda_q \geq 0, \qquad q \in Q.
\end{aligned}
$$

PCE unidimensional

▷ Decomposição de Dantzig-Wolfe

- Quando as barras são idênticas temos $\bar{\mathcal{X}} = \mathcal{X}^1 = \ldots = \mathcal{X}^n$ e, portanto, podemos usar agregação, obtendo:

$$
\begin{aligned}
\text{min} \quad & \sum_{q \in Q} \bar{y}_q \lambda_q \\
\text{s.a} \quad & \sum_{q \in Q} \bar{x}_{qj} \lambda_q \geq d_j, \quad j = 1, \ldots, m, \\
& \sum_{q \in Q} \lambda_q^k \leq n, \\
& \lambda_q \geq 0, \qquad q \in Q.
\end{aligned}
$$

(dado que $\mathbf{0} \in \bar{\mathcal{X}}$)

PCE unidimensional

▷ Decomposição de Dantzig-Wolfe

- Assim, assumindo que há um número suficientemente grande de barras (idênticas) em estoque:

$$\begin{aligned}
\min \quad & \sum_{q \in Q} \bar{y}_q \lambda_q \\
\text{s.a} \quad & \sum_{q \in Q} \bar{x}_{qj} \lambda_q \geq d_j, \quad j = 1, \ldots, m, \\
& \lambda_q \geq 0, \qquad q \in Q.
\end{aligned}$$

PCE unidimensional

▷ Decomposição de Dantzig-Wolfe

- Assim, assumindo que há um número suficientemente grande de barras (idênticas) em estoque:

$$
\begin{aligned}
\text{min} \quad & \sum_{q \in Q} \bar{y}_q \lambda_q \\
\text{s.a} \quad & \sum_{q \in Q} \bar{x}_{qj} \lambda_q \geq d_j, \quad j = 1, \ldots, m, \quad (p_j) \\
& \lambda_q \geq 0, \quad q \in Q.
\end{aligned}
$$

PCE unidimensional

▷ Decomposição de Dantzig-Wolfe

- O subproblema é dado por:

$$\begin{aligned} \text{min} \quad & y - \sum_{j=1}^{m} \overline{p}_j x_j \\ \text{s.a} \quad & (x, y) \in \bar{\mathcal{X}}. \end{aligned}$$

PCE unidimensional

▷ Decomposição de Dantzig-Wolfe

- O subproblema é dado por:

$$\begin{aligned} \text{min} \quad & y - \sum_{j=1}^{m} \overline{p}_j x_j \\ \text{s.a} \quad & \sum_{j=1}^{m} l_j x_j \leq Ly, \\ & x_j \in \mathbb{Z}_+, \qquad j = 1, \ldots, m, \\ & y \in \{0, 1\}. \end{aligned}$$

PCE unidimensional

▷ Decomposição de Dantzig-Wolfe

- O subproblema é dado por:

$$\begin{aligned} \max \quad & \sum_{j=1}^{m} \overline{p}_j x_j \\ \text{s.a} \quad & \sum_{j=1}^{m} l_j x_j \leq L, \\ & x_j \in \mathbb{Z}_+, \qquad j = 1, \ldots, m, \end{aligned}$$

PCE unidimensional

▷ Decomposição de Dantzig-Wolfe

- O subproblema é dado por:

$$\begin{aligned} \max \quad & \sum_{j=1}^{m} \overline{p}_j x_j \\ \text{s.a} \quad & \sum_{j=1}^{m} l_j x_j \leq L, \\ & x_j \in \mathbb{Z}_+, \qquad j = 1, \ldots, m, \end{aligned}$$

- *Problema da mochila*;

PCE unidimensional

▷ Decomposição de Dantzig-Wolfe

- O subproblema é dado por:

$$\begin{aligned} \max \quad & \sum_{j=1}^{m} \overline{p}_j x_j \\ \text{s.a} \quad & \sum_{j=1}^{m} l_j x_j \leq L, \\ & x_j \in \mathbb{Z}_+, \qquad j = 1, \ldots, m, \end{aligned}$$

- *Problema da mochila*;
- Em geral, é resolvido por programação dinâmica ou *branch-and-bound*, com o auxílio de heurísticas.

PCE unidimensional

▷ Decomposição de Dantzig-Wolfe

- O problema mestre agregado resultante é conhecido como formulação de *Gilmore e Gomory*:

$$\begin{aligned} \text{min} \quad & \sum_{q \in Q} \bar{y}_q \lambda_q \\ \text{s.a} \quad & \sum_{q \in Q} \bar{x}_{qj} \lambda_q \geq d_j, \quad j = 1, \ldots, m, \\ & \lambda_q \geq 0, \qquad q \in Q. \end{aligned}$$

PCE unidimensional

▷ Decomposição de Dantzig-Wolfe

- O problema mestre agregado resultante é conhecido como formulação de *Gilmore e Gomory*:

$$
\begin{aligned}
\text{min} \quad & \sum_{q \in Q} \lambda_q \\
\text{s.a} \quad & \sum_{q \in Q} a_{qj} \lambda_q \geq d_j, \quad j = 1, \ldots, m, \\
& \lambda_q \geq 0, \qquad q \in Q.
\end{aligned}
$$

PCE unidimensional

▷ Gilmore e Gomory, Operations Research 9, 1961

A LINEAR PROGRAMMING APPROACH TO THE CUTTING-STOCK PROBLEM

P. C. Gilmore and R. E. Gomory

International Business Machines Corporation, Research Center, Yorktown, New York

(Received May 8, 1961)

The cutting-stock problem is the problem of filling an order at minimum cost for specified numbers of lengths of material to be cut from given stock lengths of given cost. When expressed as an integer programming problem the large number of variables involved generally makes computation infeasible. This same difficulty persists when only an approximate solution is being sought by linear programming. In this paper, a technique is described for overcoming the difficulty in the linear programming formulation of the problem. The technique enables one to compute always with a matrix which has no more columns than it has rows.

SOME linear programming problems arising from combinatorial problems become intractable because of the large number of variables involved. Usually each variable represents some activity, and the difficulty is that there are too many possible competing activities satisfying the combinatorial restrictions of the problem. An example of this is the cutting-stock problem described below in a form similar to that used by EISEMANN.[1]

PCE unidimensional

▷ Decomposição de Dantzig-Wolfe

- Colunas são padrões de corte:

$$\begin{bmatrix} 3 \\ 0 \\ 1 \\ \vdots \\ 0 \end{bmatrix} \quad \begin{matrix} \rightarrow \text{número de vezes que o item 1 está no padrão} \\ \\ \\ \rightarrow \text{número de vezes que o item } i \text{ está no padrão} \\ \\ \end{matrix}$$

PCE unidimensional

▷ Exemplo

Uma fábrica de papel, trabalha com a produção de bobinas para impressoras. As encomendas para a próxima semana correspondem a 32 bobinas de 5 cm de largura, 16 bobinas de 7 cm de largura e 21 bobinas de 7,5 cm de largura. Essas bobinas são cortadas a partir de bobinas grandes disponíveis em estoques, todas com largura de 20 cm. Determine qual a melhor forma de atender às encomendas de modo a minimizar o número de bobinas grandes utilizadas.

PCE unidimensional

▷ Exemplo: Alguns padrões de corte

Padrão	5 cm	7 cm	7,5 cm	Total
1	4	0	0	20
2	0	2	0	14
3	0	0	2	15
4	1	1	1	19,5
5	2	1	0	17
⋮	⋮	⋮	⋮	⋮

PCE unidimensional

▷ Exemplo: Problema Mestre Restrito

$$
\begin{aligned}
\text{[PMR]} \quad \min \quad & \lambda_1 + \lambda_2 + \lambda_3 + \lambda_4 + \lambda_5 + \dots \\
\text{s.a} \quad & 4\lambda_1 + 0\lambda_2 + 0\lambda_3 + 1\lambda_4 + 2\lambda_5 + \dots \geq 32, \\
& 0\lambda_1 + 2\lambda_2 + 0\lambda_3 + 1\lambda_4 + 1\lambda_5 + \dots \geq 16, \\
& 0\lambda_1 + 0\lambda_2 + 2\lambda_3 + 1\lambda_4 + 0\lambda_5 + \dots \geq 21, \\
& \lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5, \dots \in \mathbb{Z}_+.
\end{aligned}
$$

PCE unidimensional

▷ Exemplo: Problema Mestre Restrito inicial e Subproblema

$$
\begin{aligned}
\text{[PMR]} \quad \min \quad & \lambda_1 + \lambda_2 + \lambda_3 + \lambda_4 + \lambda_5 \\
\text{s.a} \quad & 4\lambda_1 + 0\lambda_2 + 0\lambda_3 + 1\lambda_4 + 2\lambda_5 \geq 32, \\
& 0\lambda_1 + 2\lambda_2 + 0\lambda_3 + 1\lambda_4 + 1\lambda_5 \geq 16, \\
& 0\lambda_1 + 0\lambda_2 + 2\lambda_3 + 1\lambda_4 + 0\lambda_5 \geq 21, \\
& \lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5 \in \mathbb{Z}_+.
\end{aligned}
$$

PCE unidimensional

▷ Exemplo: Problema Mestre Restrito inicial e Subproblema

$$
\begin{aligned}
\text{[PMR]} \quad \min \quad & \lambda_1 + \lambda_2 + \lambda_3 + \lambda_4 + \lambda_5 \\
\text{s.a} \quad & 4\lambda_1 + 0\lambda_2 + 0\lambda_3 + 1\lambda_4 + 2\lambda_5 \geq 32, \qquad (\bar{p}_1) \\
& 0\lambda_1 + 2\lambda_2 + 0\lambda_3 + 1\lambda_4 + 1\lambda_5 \geq 16, \qquad (\bar{p}_2) \\
& 0\lambda_1 + 0\lambda_2 + 2\lambda_3 + 1\lambda_4 + 0\lambda_5 \geq 21, \qquad (\bar{p}_3) \\
& \lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5 \in \mathbb{Z}_+.
\end{aligned}
$$

PCE unidimensional

▷ Exemplo: Problema Mestre Restrito inicial e Subproblema

$$
\begin{array}{lll}
\text{[PMR]} & \min & \lambda_1 + \lambda_2 + \lambda_3 + \lambda_4 + \lambda_5 \\
& \text{s.a} & 4\lambda_1 + 0\lambda_2 + 0\lambda_3 + 1\lambda_4 + 2\lambda_5 \geq 32, \quad (\bar{p}_1) \\
& & 0\lambda_1 + 2\lambda_2 + 0\lambda_3 + 1\lambda_4 + 1\lambda_5 \geq 16, \quad (\bar{p}_2) \\
& & 0\lambda_1 + 0\lambda_2 + 2\lambda_3 + 1\lambda_4 + 0\lambda_5 \geq 21, \quad (\bar{p}_3) \\
& & \lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5 \in \mathbb{Z}_+.
\end{array}
$$

$$
\begin{array}{lll}
\text{[SP]} & \max & \bar{p}_1 x_1 + \bar{p}_2 x_2 + \bar{p}_3 x_3 \\
& \text{s.a} & 5x_1 + 7x_2 + 7{,}5x_3 \leq 20, \\
& & x_1, x_2, x_3 \in \mathbb{Z}_+.
\end{array}
$$

Obs.: Dado um ponto extremo $(\bar{x}_1, \bar{x}_2, \bar{x}_3)$, o custo relativo é dado por: $1 - (\bar{p}_1\bar{x}_1 + \bar{p}_2\bar{x}_2 + \bar{p}_3\bar{x}_3)$

PCE unidimensional

▷ Decomposição de Dantzig-Wolfe

Vantagens:

- Relaxação linear apertada (*tight*);

PCE unidimensional

▷ Decomposição de Dantzig-Wolfe

Vantagens:

- Relaxação linear apertada (*tight*);
- O gap de otimalidade é em geral menor do que 1; (Conjectura: gap < 2)

PCE unidimensional

▷ Decomposição de Dantzig-Wolfe

Vantagens:

- Relaxação linear apertada (*tight*);
- O gap de otimalidade é em geral menor do que 1; (Conjectura: gap < 2)
- Heurísticas de arredondamento são efetivas!

PCE unidimensional

▷ Decomposição de Dantzig-Wolfe

Vantagens:

- Relaxação linear apertada (*tight*);
- O gap de otimalidade é em geral menor do que 1; (Conjectura: gap < 2)
- Heurísticas de arredondamento são efetivas!
- O que muda para os casos 2D e 3D?

PCE unidimensional: heurísticas

▷ Poldi e Arenales, Computers & Operations Research 36, 2009

ELSEVIER

Contents lists available at ScienceDirect

Computers & Operations Research

journal homepage: www.elsevier.com/locate/cor

Heuristics for the one-dimensional cutting stock problem with limited multiple stock lengths

Kelly Cristina Poldi*, Marcos Nereu Arenales

Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, CP 668, CEP 13560-970 São Carlos-SP, Brazil

A R T I C L E I N F O

Available online 10 July 2008

Keywords:
Cutting stock
Column generation
Heuristics

A B S T R A C T

This paper deals with the classical one-dimensional integer cutting stock problem, which consists of cutting a set of available stock lengths in order to produce smaller ordered items. This process is carried out in order to optimize a given objective function (e.g., minimizing waste). Our study deals with a case in which there are several stock lengths available in limited quantities. Moreover, we have focused on problems of low demand. Some heuristic methods are proposed in order to obtain an integer solution and compared with others. The heuristic methods are empirically analyzed by solving a set of randomly generated instances and a set of instances from the literature. Concerning the latter, most of the optimal solutions of these instances are known, therefore it was possible to compare the solutions. The proposed methods presented very small objective function value gaps.

PCE unidimensional: branch-price-and-cut

▷ Wei et al., INFORMS JoC, 2020

informs
http://pubsonline.informs.org/journal/ijoc

INFORMS JOURNAL ON COMPUTING
Vol. 32, No. 2, Spring 2020, pp. 428–443
ISSN 1091-9856 (print), ISSN 1526-5528 (online)

A New Branch-and-Price-and-Cut Algorithm for One-Dimensional Bin-Packing Problems

Lijun Wei,[a] Zhixing Luo,[b,*] Roberto Baldacci,[d] Andrew Lim[c]

[a] Key Laboratory of Computer Integrated Manufacturing System, School of Electromechanical Engineering, Guangdong University of Technology, 510006 Guangzhou, People's Republic of China; [b] School of Management and Engineering, Nanjing University, 210093 Nanjing, People's Republic of China; [c] Department of Industrial Systems Engineering and Management, National University of Singapore, Singapore 119077; [d] Department of Electrical, Electronic, and Information Engineering "Guglielmo Marconi," University of Bologna, 47521 Cesena, Italy

*Corresponding author

Contact: villagerwei@gmail.com (LW); luozx.hkphd@gmail.com, http://orcid.org/0000-0002-0491-1501 (ZL); r.baldacci@unibo.it (RB); alim.china@gmail.com (AL)

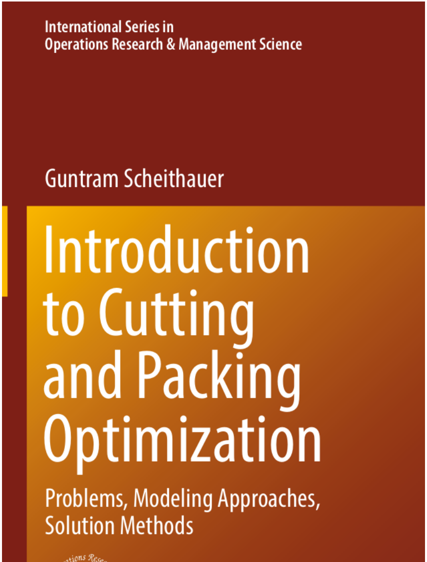
Received: August 24, 2017
Revised: February 27, 2018; September 12, 2018
Accepted: September 20, 2018
Published Online in Articles in Advance: November 1, 2019

https://doi.org/10.1287/ijoc.2018.0867

Abstract. In this paper, a new branch-and-price-and-cut algorithm is proposed to solve the one-dimensional bin-packing problem (1D-BPP). The 1D-BPP is one of the most fundamental problems in combinatorial optimization and has been extensively studied for decades. Recently, a set of new 500 test instances were proposed for the 1D-BPP, and the best exact algorithm proposed in the literature can optimally solve 167 of these new instances, with a time limit of 1 hour imposed on each execution of the algorithm. The exact algorithm proposed in this paper is based on the classical set-partitioning model for the 1DBPPs and the subset row inequalities. We describe an ad hoc label-setting algorithm to solve the pricing problem, dominance, and fathoming rules to speed up its computation and a new primal heuristic. The exact algorithm can easily handle some practical constraints, such as the incompatibility between the items, and therefore, we also apply it to solve the one-dimensional bin-packing problem with conflicts (1D-BPPC). The proposed method is tested on a large family of 1D-BPP and 1D-BPPC classes of instances. For the 1D-BPP, the proposed method can optimally solve 237 instances of the new set of difficult instances; the largest instance involves 1,003 items and bins of capacity 80,000. For the 1D-BPPC, the experiments show that the method is highly competitive with state-of-the-art methods and that it successfully closed several open 1D-BPPC instances.

Problemas de corte e empacotamento

▷ Scheithauer, 2018



- Obrigado pela atenção!
- Dúvidas?