LOCAL BRANCHING APLICADO AO PROBLEMA DE CORTE DE ESTOQUE COM DATA DE ENTREGA

Elisama de Araújo Silva Oliveira
CEFET-MG
Av. Amazonas, 7675 - Nova Gameleira, Belo Horizonte - MG, 30510-000
elisamamatematica_licenciatura@hotmail.com

Elisângela Martins de Sá
CEFET-MG
Av. Amazonas, 7675 - Nova Gameleira, Belo Horizonte - MG, 30510-000
elisangela.martinss@gmail.com

Sérgio Ricardo de Souza
CEFET-MG
Av. Amazonas, 7675 - Nova Gameleira, Belo Horizonte - MG, 30510-000
sergio@dppg.cefetmg.br

Elizabeth Fialho Wanner
CEFET-MG
Av. Amazonas, 7675 - Nova Gameleira, Belo Horizonte - MG, 30510-000
efwanner@gmail.com

RESUMO

O Problema de Corte de Estoque (PCE) consiste em cortar um conjunto de objetos disponíveis em estoque para produzir um conjunto de itens em quantidades e comprimentos especificados, de modo a otimizar uma função objetivo, que pode ser minimizar a perda do material. É proposto neste artigo uma extensão desse problema, no qual se acrescenta, aos pedidos, uma data de entrega. A abordagem mais utilizada na literatura para a solução do PCE é o método simplex com geração de colunas proposto por Gilmore e Gomory [1961] e Gilmore e Gomory [1963]. Devido à complexidade de resolução desse problema para instâncias de maior dimensão, é necessário aplicar metaheurísticas e/ou matheurísticas para a sua resolução. Nesse sentido, este artigo propõe a aplicação da matheurística *Local Branching* para a resolução desse problema. Os resultados mostram que a técnica de Geração de colunas gera bons padrões de corte, bem com a matheurística proposta consegue encontrar a mesma solução que a determinada pelo *solver* CPLEX.

PALAVRAS CHAVE. Problema de Corte de Estoque, Data de Entrega, *Local Branching*.
Tópicos (OC-Otimização Combinatória, PM-Programação Matemática.)

ABSTRACT

The cutting stock problem (CSP) consists of cutting a set of available objects in stock to produce a set of items in specified quantities and lengths, in order to optimize an objective function, which can be minimizing the loss of the material. An extension of this problem is proposed in this article, in which a due date is added to the orders. The most widely used approach in the literature for the solution of CSP is the simplex method with column generation, as proposed by Gilmore e Gomory [1961] and Gilmore e Gomory [1963]. Due to the complexity of resolution for larger instances, it is necessary to apply meta-heuristics and/or matheuristics for its resolution. In this sense, this article addresses the application of the Local Branching matheuristic for solving CSP with due dates. The results show that the column generation technique yields good cutting patterns and the proposed matheuristic determines the same solution as the one found by the CPLEX solver.

KEYWORDS. Cutting Stock Problem, Due Date, Local Branching.
Paper topics (OC-Combinatorial Optimization, PM-Mathematical Programming.)

https://proceedings.science/p/106945?lang=pt-br

1. Introdução

O Problema de Corte de Estoque (PCE) consiste em cortar um conjunto de objetos em estoque idênticos com dimensões fixas para produzir um conjunto de itens em quantidades e dimensões especificadas, de modo a minimizar as perdas de materiais e atender toda a demanda de itens. O PCE é essencial no planejamento da produção em várias indústrias, como indústrias de papel, vidro, móveis, plástico, chapas de madeira, aço, têxtil, etc. Os objetos a serem cortados podem ser bobinas de aço, bobinas de papel, placas de alumínio, chapas de metal, placas de madeira, dentre outros.

De acordo com Dyckhoff [1990], um PCE pode ser classificado de acordo com a dimensão do objeto a ser cortado. O problema pode ser unidimensional (1D), bidimensional (2D) ou tridimensional (3D), quando uma, duas ou três dimensões dos objetos, respectivamente, são relevantes no processo de corte. O presente trabalho está classificado como um problema de corte unidimensional, pois apenas a dimensão comprimento será considerada. A primeira modelagem matemática para o problema de corte de estoque foi introduzida por Kantorovich [1960].

A tipologia proposta por Dyckhoff [1990] foi um excelente instrumento para a organização e caracterização da literatura. Contudo, ao longo dos anos surgiram algumas deficiências, pois poderiam haver problemas idênticos que receberiam nomes diferentes, o que criou problemas em lidar com os desenvolvimentos recentes. Em 2007, Wäscher et al. [2007] apresentam uma tipologia melhorada da proposta inicial de Dyckhoff [1990], porém é parcialmente baseada em suas ideias originais, mas introduz novos critérios de categorização. Portanto, o resultado final desta nova estrutura tipológica é obtido pela seguinte regra: 1,2,3-D retangular, circular,...,irregular classificação intermediária. De acordo com esta classificação, o presente trabalho aborda uma extensão do seguinte problema: 1D - Single Stock Size Cutting Stock Problem - SSSCSP: Problema de Corte de Estoque com um único tipo (comprimento) de objeto; pois estamos considerando a data de entrega dos itens cortados.

Versões mais recentes de problemas de corte consideram, além das decisões relativas a como cortar os itens, decisões relacionadas às datas de entrega dos itens que serão cortados. Reinertsen e Vossen [2010] abordam o PCE unidimensional com Data de Entrega (PCEDE) aplicado em uma indústria de bobinas de aço. Arbib e Marinelli [2014] estudaram o PCE unidimensional com Data de Entrega (PCEDE), propondo novas formulações e métodos de resolução. Arbib e Marinelli [2017] estudam o problema de *bin packing* unidimensional com data de entrega, em que o objetivo é minimizar uma combinação convexa do *makespan*, *lateness* e *tardiness*.

O presente artigo propõe um modelo matemático de otimização linear inteira para o Problema de Corte de Estoque com Data de Entrega (PCEDE). Além de penalizar o não cumprimento da data dos pedidos, considera-se também o adiantamento na entrega dos itens. Para a resolução do problema, propõe-se um método de resolução em três fases. Na primeira fase, o problema relaxado é resolvido, usando a técnica de geração de colunas. Usando as colunas geradas na primeira fase, a segunda fase consiste na aplicação da heurística *Relax-and-fix* para a resolução do problema em sua versão inteira. A terceira fase consiste na resolução da matheurística *Local Branching*, usando, como solução inicial, a solução vinda da segunda fase.

Este artigo está organizado da seguinte forma. A Seção 2 descreve o problema proposto e apresenta uma formulação matemática. A Seção 3 apresenta os métodos de resolução utilizados. Os resultados computacionais são apresentados na Seção 4. A Seção 5 apresenta as considerações finais e propostas futuras.

2. Problema de Corte de Estoque com Data de Entrega

Uma abordagem para solucionar um PCE se baseia em encontrar padrões de corte e a frequência destes padrões, ou seja, a quantidade de vezes que esses padrões serão executados (cor-

tados). O Problema de Corte de Estoque com Data de Entrega (PCEDE) é uma extensão do PCE em que, além de se considerar o planejamento dos padrões de corte e suas respectivas frequências, considera-se também o atraso e o adiantamento na entrega dos itens [Arbib e Marinelli, 2014], [Arbib e Marinelli, 2017], [Reinertsen e Vossen, 2010]. A Figura 1 exemplifica o objeto que se tem

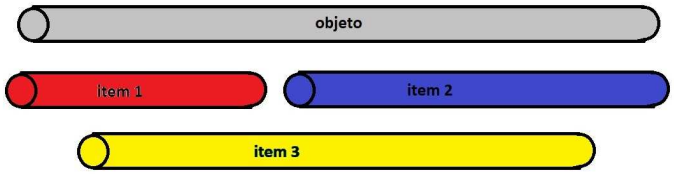


Figura 1: Objeto em estoque e tipos de itens.

em estoque (na cor cinza) e os itens demandados (nas cores vermelho, azul, amarelo).

Padrão de corte é a maneira como um objeto em estoque é cortado para a produção dos itens demandados. A um padrão de corte associa-se um vetor m-dimensional que contabiliza os itens produzidos, na forma:

$$a = (\alpha_1, \alpha_2, \ldots, \alpha_i, \ldots, \alpha_m)^t,$$

em que α_i é o número de vezes que o item tipo i está presente no padrão de corte a.

Um padrão de corte que produza apenas um tipo de item é chamado padrão de corte homogêneo:

$$a = (0, \ldots, 0, \alpha_i, 0, \ldots, 0)^t,$$

com $\alpha_i \neq 0$. Em particular, para o PCE unidimensional, pode-se escrever um padrão de corte homogêneo da seguinte forma:

$$a = \left(0, \ldots, 0, \frac{L}{l_i}, 0, \ldots, 0\right)$$

A Figura 2 representa 5 possíveis padrões de corte que podem ser gerados a partir de um objeto de comprimento $L = 5$ e itens de comprimento $l_i = 2, 3, 4$. O vetor que aparece ao lado de cada objeto é a representação do padrão de corte cortado no objeto. A cor cinza em cada objeto representa a perda ocorrida no objeto após os cortes dos itens.

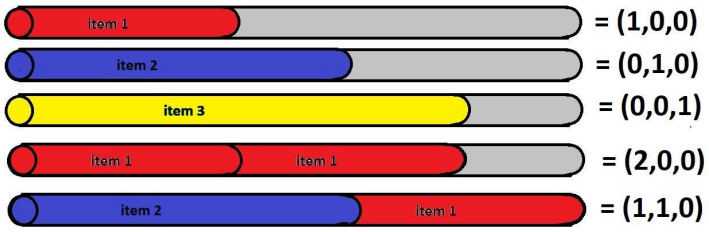


Figura 2: Solução de corte.

Considere que exista em estoque um conjunto de objetos (grandes) idênticos de comprimento L, em uma quantidade suficiente para que seja possível atender a toda a demanda. Seja,

então, um conjunto de n tipos de itens menores de comprimento $\ell_i (i = 1, 2, \ldots, n)$, a serem programados em uma operação de corte para pedidos em um horizonte de planejamento dividido em períodos de tempo $t = 1, 2, \ldots, T$, no qual se deve produzir uma determinada demanda b_{ti} do item i para ser entregue em uma data d_t. Seja tp_t o tempo que se leva para cortar o pedido em um período t. Como, em algumas operações de corte, nem sempre é possível atender a todos os pedidos na sua respectiva data de entrega, há uma penalização θ_t por unidade de término de atraso e, caso no processo de corte haja pedidos entregues antes de sua data de entrega, também há uma penalização β_t por unidade de tempo de término de adiantamento. Considere, além disso, que não seja possível receber pedidos depois que o processo de corte se iniciou. O objetivo do problema será agrupar os n itens a serem processados de acordo com a capacidade do objeto, de forma a minimizar a soma de três termos: as penalidades por adiantamentos, as penalidades por atrasos nas entregas e os custos do uso de matéria-prima.

Os índices, parâmetros e variáveis de decisão para o problema proposto são:

Índices:

(i) i: tipo de item, $i = 1, .., n$;

(ii) t: período de tempo, $t = 1, ..., T$;

(iii) j: padrão de corte, $j = 1, .., N$.

Parâmetros:

(i) L: comprimento dos objetos em estoque;

(ii) ℓ_i: comprimento do item tipo i;

(iii) b_{ti}: demanda no período t do item i;

(iv) d_t: data de entrega do período t;

(v) a_{itj}: quantidade de itens tipo i produzidos no período t pelo padrão de corte j;

(vi) θ_t: penalidade por unidade de atraso na entrega do período t;

(vii) β_t: custo por unidade de adiantamento na entrega do período t.

Variáveis de Decisão:

(i) x_{tj}: número de objetos cortados no período t segundo o padrão de corte j;

(ii) T_t: tempo de atraso na entrega do pedido no período t;

(iii) E_t: tempo de adiantamento na entrega do pedido no período t;

(iv) tp_t: tempo que leva para cortar o pedido no período t.

O modelo matemático é dado por:

$$\min \quad \sum_{t=1}^{T}\sum_{j=1}^{N} x_{tj} + \sum_{t=1}^{T} \beta_t E_t + \sum_{t=1}^{T} \theta_t T_t \tag{1}$$

$$\text{sujeito a:} \quad T_t \geq tp_t + T_{t-1} - E_{t-1} - (d_t - d_{t-1}) \ \forall t, \tag{2}$$

$$E_t \geq d_t - d_{t-1} - T_{t-1} + E_{t-1} - tp_t, \ \forall t, \tag{3}$$

$$tp_t = \sum_{j=1}^{N} x_{jt}, \ \forall t, \tag{4}$$

$$\sum_{j=1}^{N} a_{itj} x_{tj} \geq b_{ti}, \ \forall t \ , \forall i, \tag{5}$$

$$E_0 = 0, \text{ e } T_0 = 0, \tag{6}$$

$$x_{tj} \in \mathbb{Z}, \ \forall j, \ \forall t, \tag{7}$$

$$E_t, T_t, tp_t \in \mathbb{Z}, \ \forall t. \tag{8}$$

O objetivo desse problema é a minimização da função objetivo (1), composta por três parcelas: (i) o número de objetos cortados x_{jt}; (ii) o atraso T_i; e (iii) o adiantamento E_i na entrega dos pedidos. O conjunto de restrições (2)-(3) contabiliza os tempos de atrasos e de adiantamentos. O conjunto de restrições (4) mostra o tempo que será gasto na produção dos pedidos em cada um dos períodos, assumindo que se gasta uma unidade de tempo para cortar um objeto, independente do padrão de corte que será cortado. O conjunto de restrições (5) garante que a demanda dos períodos deve ser atendida. O conjunto de restrições (6) representa as condições de contorno iniciais, que afirmam que não existe atraso ou adiantamento de pedidos no início do processo de corte. As restrições (7)-(8) representam as condições de não-negatividade e integralidade das variáveis de decisão.

3. Método de Solução

3.1. Método de Geração de Colunas

O Método de Geração de Colunas é uma modificação do método simplex, proposta por Gilmore e Gomory [1961] e Gilmore e Gomory [1963], que consiste em resolver o problema relaxado com apenas um subconjunto de padrões de corte, conhecido como problema mestre. Para isso, a cada iteração são gerados padrões de corte que melhoram a solução do problema relaxado até a obtenção da solução ótima.

No presente artigo, o problema mestre é dado pelo modelo matemático do PCDE, formado pelas expressões (1)-(8), tendo apenas alguns padrões de cortes adicionados e considerando, como relaxadas, as restrições de integralidade (7)-(8). Para gerar os padrões de corte, a cada iteração deve ser resolvido um subproblema, dado pelo seguinte problema da mochila, para cada período $t \in T$:

$$\min \quad c_j - \lambda_t - \sum_{i=1}^{n} \pi_{ti} \alpha_i \tag{9}$$

$$\text{sujeito a:} \quad \sum_{i=1}^{n} \ell_i \alpha_i \leq L \tag{10}$$

$$\alpha_i \geq 0 \text{ inteiros} \tag{11}$$

no qual as variáveis π_{ti} e λ_t são variáveis duais do problema mestre, relacionadas às restrições de atendimento da demanda e de tempo de preparo, dadas pelas expressões (5) e (6), respectivamente.

3.2. Heurística *Relax-and-Fix*

A heurística *Relax-and-Fix*, utilizada para construir uma solução para problemas de otimização linear inteira misto [Wolsey, 1998], baseia-se em dividir o problema, com variáveis inteiras, em subproblemas mais simples. Para isso, o conjunto de variáveis inteiras é dividido em P subconjuntos distintos $Q_i, i = 1, \ldots, P$. Em cada iteração n, resolve-se um subproblema em que as variáveis do conjunto Q_n são definidas como inteiras; as variáveis dos conjuntos Q_i, $(i = 1, \ldots, n-1)$, são fixadas; as variáveis dos conjuntos Q_i, $(i = n+1, \ldots, P)$, são relaxadas. Se o subproblema corrente for infactível, o processo é interrompido, pois não é possível encontrar uma solução para o problema com as variáveis pertencentes aos conjuntos Q_i, $i = 1, \ldots, n-1$, fixadas. Caso contrário, as variáveis do conjunto Q_n são fixadas. Então, o processo se repete, até que todas as variáveis tenham solução inteira.

Em um primeiro momento, aplicou-se o método de geração de colunas. Para iniciar o método de geração de colunas, foi utilizado um conjunto de padrões de corte homogêneos. Em seguida, com os padrões de corte determinados, foi aplicada a heurística *Relax-and-Fix*, com o intuito de obter uma solução para as variáveis x_{tj}. Nesta fase, a partição dos conjuntos foi feita sobre os períodos de tempo. Ao final desse passo, obteve-se uma solução para todas as variáveis inteiras x_{tj} do modelo.

A solução obtida pela heurística *Relax-and-Fix* servirá para termos uma solução inicial na fase da aplicação da matheurística *Local Branching*.

3.3. *Local Branching*

Para resolver problemas de maior dimensão, propõe-se a aplicação da matheurística *Local Branching*, proposta inicialmente por [Fischetti e Lodi, 2003].

Considere o problema inteiro (P) a seguir:

$$\begin{aligned} (P)\min \quad & \sum_{t=1}^{T} c^t x \\ \text{sujeito a:} \quad & Ax \geq b \\ & x_j \in \mathbb{Z},\ \forall j \in \mathbb{G} \end{aligned}$$

em que $\mathbb{G}$ denota os índices das variáveis inteiras e positivas.

Dada uma solução factível de referência $x^{'}$ de (P), seja o conjunto $S^{'} = \{j \in \mathbb{G}, x^{'} \geq 0\}$, que denota o suporte de $x^{'}$. Para um dado parâmetro inteiro positivo k, define-se a vizinhança $N^k(x^{'})$ como o conjunto solução factível de (P) satisfazendo uma restrição adicional *Local Branching*.

A técnica *Local Branching* é baseada na suposição de que há um conjunto de variáveis binárias e, além disso, que esse conjunto tem importância para a solução. Todavia, é possível trabalhar com problemas em que as variáveis binárias não desempenham um papel relevante no modelo ou até mesmo pode acontecer que o problema não apresenta nenhuma variável binária. Dessa forma, introduzir restriccão de ramificação local não auxilia o *solver* em sua tarefa de solução. Por isso, surge a necessidade de adicionar restrição de ramificação local modificada, que incluam as variáveis que são inteiras e não binárias na formulação da função distância $\Delta(x, x^{'})$ que será adicionada. Considere que as variáveis inteiras x_j são limitadas, de modo que $l_j \leq x_j \leq u_j$. Então, uma restrição de ramificação local adequada pode ser definida da seguinte maneira:

$$\Delta_1(x, x^{'}) := \sum_{j \in I : x_j^{'} = l_j} \mu_j(x_j - l_j) + \sum_{j \in I : x_j^{'} = u_j} \mu_j(u_j - x_j) + \sum_{j \in I : l_j < x_j^{'} < u_j} \mu_j({x_j}^{+} + {x_j}^{-}) \leq k$$

em que as ponderações μ_j são definidos, por exemplo, como:

$$\mu_j = \frac{1}{u_j - l_j} \quad \forall j \in I,$$

enquanto os termos de variação ${x_j}^+ + {x_j}^-$ requerem a introdução no modelo de Programação Inteira Mista (MIP) de restrições adicionais da forma:

$$\begin{array}{rcl} x_j & = & x_j^{'} + x_j^+ - x_j^- \\ x_j^+ & \geq & 0 \\ x_j^- & \geq & 0 \end{array}$$

$\forall j \in I, l_j < x_j^{'} < u_j$. Alternativamente, pode-se evitar o uso dessas variáveis adicionais através de uma das seguintes definições relaxadas:

$$\Delta_2(x, x^{'}) := \sum_{j \in I : x_j^{'} = l_j} \mu_j (x_j - l_j) + \sum_{j \in I : x_j^{'} = u_j} \mu_j (u_j - x_j) \leq k^{'}$$

ou

$$\Delta_3(x, x^{'}) := \sum_{j \in I : x_j^{'} = l_j} \mu_j (x_j - l_j) \leq k^{''}$$

Dada $x^{'}$, o espaço de solução associado com o nó de *Local Branching* pode ser particionado por meio da disjunção:

$$\Delta_1(x, x^{'}) \leq k \text{ ou } \Delta_1(x, x^{'}) \geq k + 1$$

4. Resultados Computacionais

Para a realização do estudo computacional, o modelo foi resolvido pelo `C++` Concert Tecnology/CPLEX, em um notebook Intel Core i7 com 3.60 GHz e 16 GB de memória.

Encontrar uma base de testes para problemas de maior dimensão foi a principal dificuldade enfrentada para a realização dos testes computacionais de validação dos desenvolvimentos propostos no artigo. Esta base não foi encontrada na literatura. Dessa forma, para a aplicação das técnicas propostas, foram criadas 10 instâncias com as seguintes características:

(i) o objeto tem $L = 10.000u$ de comprimento;

(ii) número de períodos $t = 15$;

(iii) número de tipos de itens $n = 20$;

(iv) o comprimento dos itens variam no intervalo $[1123,\ 8829]$;

(v) a demanda varia no intervalo $[0,\ 7890]$.

Para os testes computacionais, a data de entrega foi calculada usando a seguinte expressão:

$$d_t = 1,4 \times \frac{\sum_{i=1}^{n} l_i}{b}_{ti} L\,, \quad \forall t.$$

Em todos os problemas, assume-se que $\alpha = 1$ (penalizações por atraso) e $\beta = 1$ (penalizações por adiantamento) na entrega dos itens. Na prática, esses valores devem ser ponderados de maneira a ser analisada. As Tabelas 1, 2 e 3 mostram os resultados para as 10 instâncias analisadas. Os resultados estão divididos na fase de Geração de Colunas (Relaxada); na fase de *Relax-and-Fix* (Inteira); e na fase de *Local Branching*. Para avaliar a qualidade da solução inteira obtida, comparamos com a solução obtida pelo *solver* CPLEX.

O valor do parâmetro k foi utilizado como 10 para todos os problemas analisados. A restrição *Local Branching* adicionada foi a seguinte:

$$\Delta_2(x, x^{'}) := \sum_{j \in I : x_j^{'} = l_j} \mu_j(x_j - l_j) + \sum_{j \in I : x_j^{'} = u_j} \mu_j(u_j - x_j) \leq k^{'}$$

Assume-se que os valores $\ell_j = 0$ e que u_j é o maior valor dos x_{tj} encontrado na solução inicial obtida pela fase 2 (*Relax-and-Fix*).

Foi preciso reduzir o limite de tolerância do *gap* usado como padrão pelo CPLEX, visto que, para o valor padrão, os resultados estavam gerando inconsistências no sentido da matheurística encontrar melhores soluções que a obtida pelo CPLEX. Foi usado a tolerância de $10e-8$ com o seguinte parâmetro do Concert (`IloCplex::Param::MIP::Tolerances::MIPGap, 10e-8`). Com isso, as inconsistências foram resolvidas como apresentadas nas tabelas a seguir.

As siglas das Tabelas são: GC é Geração de Colunas, RAF é *Relax-and-Fix*, LB é o *Local Branching*.

Tabela 1: Tempos computacionais obtidos em todas as fases.

Problema	GC	RAF	LB	CPLEX
P1	0,16	0,28	0,36	0,20
P2	0,01	0,13	0,20	0,01
P3	0,21	0,33	0,41	0,25
P4	0,31	0,48	0,58	0,34
P5	0,23	0,37	0,48	0,27
P6	0,14	0,29	0,36	0,17
P7	0,15	0,30	0,36	0,16
P8	0,32	0,45	0,54	0,35
P9	0,28	0,46	0,57	0,29
P10	0,24	0,38	0,42	0,25
Média	0,20	0,34	0,42	0,22

A Tabela 1 apresenta o tempo computacional gasto para as 10 instâncias. Pode-se concluir que a fase de Geração de Colunas, para estas instâncias, resolve o problema relaxado em um tempo computacional pequeno e o mesmo ocorre com a solução obtida pelo CPLEX. Para essas instâncias, o tempo gasto pelas heurísticas *Relax-and-Fix* e *Local Branching* foram maiores, o que permite concluir que as instâncias criadas não são suficientemente desafiadoras de modo a impactar a solução via CPLEX. Um ponto importante para a continuidade do trabalho é, portanto, gerar instâncias mais desafiadoras para que os resultados quanto á validade das técnicas propostas sejam mais conclusivos.

A Tabela 2 apresenta os valores da função objetivo para as 10 instâncias. Com a análise dos dados, pode-se concluir que a heurística *Relax-and-Fix* gera boas soluções aproximadas da

Tabela 2: Valores da função objetivo em todas as fases.

Problema	GC	RAF	LB	CPLEX
P1	122477	122478	122477	122477
P2	100269	100269	100269	100269
P3	112665	112666	112665	112665
P4	120238	120238	120238	120238
P5	109195	109196	109195	109195
P6	125690	125691	125691	125691
P7	107767	107768	107767	107767
P8	106941	106943	106943	106943
P9	116926	116928	116926	116926
P10	112642	112644	112642	112652
Média	113481	113482,1	113481,3	113481,3

solução obtida pela técnica de Geração de Colunas. Pode-se observar também que a proposta da matheurística *Local Branching* encontra a mesma solução que a solução obtida pelo CPLEX, usando como solução inicial a solução obtida pela fase da heurística *Relax-and-Fix*.

Tabela 3: Atraso e Adiantamento obtidos nas fases de Geração de Colunas e *Local Branching*.

Problema	Atraso-GC	Adiantados-GC	Atraso-LB	Adiantados-LB
P1	0,00	0,00	10875	0,00
P2	0,00	1183,51	0,00	0,00
P3	1558,17	669,83	0,00	0,00
P4	1822	705,5	0,00	0,00
P5	343,5	0,00	1037	26
P6	241,5	417	0,00	0,00
P7	0,00	0,00	1229	350
P8	607,5	0,00	0,00	0,00
P9	362,66	574,83	493	493
P10	1496,5	730,5	2360	562
Média	643,18	428,11	1622,5	143,1

A Tabela 3 apresenta o resultado dos atrasos e adiantamentos de cada uma das instâncias pelas técnicas de Geração de Colunas e da matheurística *Local Branching*. Dentre esses problemas, a matheurística *Local Branching* conseguiu resolver sem atraso e adiantamento 5 dos 10 problemas.

5. Conclusões e Propostas futuras

Este artigo apresentou uma proposta de modelagem matemática para o problema de corte de estoque com data de entrega, bem como uma solução inteira usando a heurística *Relax-and-Fix*, baseada nas colunas geradas na fase de geração de colunas. Os resultados mostraram que esta solução gerou tempos computacionais pequenos. Com o uso da Técnica de Geração de Colunas, é possível determinar boas colunas (padrões de corte) para a solução do problema, que podem ser usadas como entrada para a solução do problema inteiro, sem a necessidade de gerar todas as colunas possíveis para o problema.

É possível concluir que o *Relax-and-Fix* gera boas soluções para o problema inteiro para ser usado na fase da matheurística *Local Branching*. A técnica *Local Branching* obteve as mesmas

soluções que as encontradas pelo *solver* CPLEX. É possível concluir que os problemas analisados não são problemas desafiadores para as técnicas propostas.

Uma proposta futura para a continuidade das pesquisas aqui apresentadas é encontrar instâncias mais desafiadoras para poder aplicar a técnica proposta e obter resultados mais conclusivos sobre a qualidade da solução obtida.

Agradecimentos

Os autores agradecem à Fundação Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), à Fundação de Amparo à Pesquisa de Minas Gerais (FAPEMIG) e ao Centro Federal de Educação Tecnológica de Minas Gerais (CEFET-MG) pelo apoio à realização desse trabalho.

Referências

Arbib, C. e Marinelli, F. (2014). On cutting stock with due dates. *Omega*, 46:11–20.

Arbib, C. e Marinelli, F. (2017). Maximum lateness minimization in one-dimensional bin packing. *Omega*, 68:76–84.

Dyckhoff, H. (1990). A typology of cutting and packing problems. *European Journal of Operational Research*, 44(2):145–159.

Fischetti, M. e Lodi, A. (2003). Local branching. *Mathematical programming*, 98(1-3):23–47.

Gilmore, P. C. e Gomory, R. E. (1961). A linear programming approach to the cutting-stock problem. *Operations research*, 9(6):849–859.

Gilmore, P. C. e Gomory, R. E. (1963). A linear programming approach to the cutting stock problemâ – part ii. *Operations Research*, 11(6):863–888.

Kantorovich, L. V. (1960). Mathematical methods of organizing and planning production. *Management science*, 6(4):366–422.

Reinertsen, H. e Vossen, T. W. (2010). The one-dimensional cutting stock problem with due dates. *European Journal of Operational Research*, 201(3):701–711.

Wäscher, G., Haußner, H., e Schumann, H. (2007). An improved typology of cutting and packing problems. *European Journal of Operational Research*, 183(3):1109–1130.

Wolsey, L. A. (1998). *Integer Programming*. John Wiley & Sons, New York, NY, EUA.