

## UM PROBLEMA DE CORTE DE ESTOQUE MULTIOBJETIVO

Luiz Leduino Salles-Neto<sup>a\*</sup>, Silvio Araujo<sup>b</sup>, Rodrigo Golfet<sup>c</sup>

<sup>a</sup>Universidade Federal de São Paulo – USP, São Paulo-SP, Brasil

<sup>b</sup>Universidade Estadual Paulista – UNESP, São José do Rio Preto-SP, Brasil

<sup>c</sup>Equus Group, LLC., São Paulo, Brasil

### Resumo

Neste trabalho, estuda-se o problema de corte de estoque unidimensional que consiste em cortar um conjunto de objetos disponíveis em estoques, para produzir uma demanda de itens menores de forma a otimizar uma determinada função objetivo que, neste trabalho é composta por três objetivos distintos: minimização do número de objetos processados (matéria-prima), minimização do número de diferentes padrões de corte (tempo de preparação da máquina), minimização do número de ciclos de serra (otimização da produtividade da serra). Para a resolução deste problema complexo foi adaptado um algoritmo genético simbiótico proposto na literatura. Alguns resultados teóricos e computacionais são apresentados.

**PALAVRAS-CHAVE:** Problema de Corte de Estoque, Otimização Multiobjetivo, Algoritmo Genético Simbiótico

### Abstract

In this paper we deal with the one-dimensional integer cutting stock problem, which consists of cutting a set of available objects in stock in order to produce ordered smaller items in such a way as to optimize a given objective function, which in this paper is composed of three different objectives: minimization of the number of objects to be cut (raw material), minimization of the number of different cutting patterns (setup time), minimization of the number of saw cycles (optimization of the saw productivity). For solving this complex problem we adopt a multiobjective approach in which we adapt, for the problem studied, a symbiotic genetic algorithm proposed in the literature. Some theoretical and computational results are presented.

**KEY-WORDS:** Cutting Stock Problem, Multiobjective Optimization; Genetic Symbiotic Algorithm.

\*Autor para correspondência: e-mail: [luiz.leduino@unifesp.br](mailto:luiz.leduino@unifesp.br)

### 1. Introdução

De acordo com a tipologia proposta em Wäscher *et al.* (2007) o problema estudado neste trabalho é classificado como Problema de Corte de Estoque Unidimensional com Único Tamanho em Estoque (*One Dimensional Single Stock Size Cutting Stock Problem (1D-SSSCSP)*), que requer que itens pequenos fracamente heterogêneos sejam completamente alocados a uma seleção de objetos grandes de dimensões iguais e fixas. Este problema surge em vários contextos industriais, tais como, indústrias de papel, aço, plástico, madeira, dentre outras.

Em geral nos problemas de corte de estoque, utiliza-se como critério de otimização a minimização da perda de material que, quando se considera o problema com apenas um tipo de objeto em estoque e a demanda deve ser atendida exatamente, é equivalente a minimização do número de objetos cortados.

No entanto, com base em alguns problemas práticos, observou-se que em alguns casos outros critérios de otimização também são importantes e devem ser levados em consideração. Por exemplo, nos casos em que se tem escassez de capacidade, o tempo necessário para a troca de padrões de corte ganha significativa importância e, conseqüentemente, a redução do número de padrões de corte diferentes passa a ser um critério relevante (Moretti e Salles-Neto 2008).

Outro critério que também tem sido considerado está relacionado a otimização da produtividade da serra considerando-se a capacidade da serra em termos de espessura, as espessuras dos objetos a serem cortados e a possibilidade de cortar vários objetos simultaneamente usando o mesmo padrão de corte. A otimização da capacidade da serra está relacionada a minimização dos ciclos da serra (conjunto de todas as operações necessárias para cortar um, ou mais objetos juntos, de acordo com um determinado padrão de corte, até que todos os itens nele (ou neles) contidos sejam produzidos (Yanasse *et al.* 1993)).

No presente trabalho os três critérios supracitados são considerados em uma única função multiobjetivo. Alguns modelos matemáticos são discutidos na Seção 2. Na Seção 3 são desenvolvidos alguns conceitos teóricos relacionados a problemas multiobjetivo, bem como, aos problemas estudados neste trabalho. Na Seção 4 apresenta-se o algoritmo genético simbiótico adaptado para o problema em questão. Os resultados computacionais são apresentados na Seção 5 e, na Seção 6, têm-se algumas conclusões e propostas de trabalhos futuros.

## 2. Modelos Matemáticos

Considere inicialmente o problema 1D-SSSCSP com o objetivo de minimizar o número de objetos a serem cortados. Para tanto considere os seguintes dados e variáveis:

Dados:

- $m$ : número de tipos de itens;
- $l_i$ : comprimento do item do tipo  $i$ ,  $i=1, \dots, m$ ;
- $d_i$ : demanda para o item do tipo  $i$ ,  $i=1, \dots, m$  ( $d$ );
- $L$ : comprimento do objeto em estoque;
- $a_j$ : vetor que corresponde ao  $j$ -ésimo padrão de corte,  $j=1, \dots, N$ . Existem  $N$  possíveis

$$\text{padrões de corte: } a_1 = \begin{pmatrix} \alpha_{11} \\ \alpha_{21} \\ \vdots \\ \alpha_{m1} \end{pmatrix}, \quad a_2 = \begin{pmatrix} \alpha_{12} \\ \alpha_{22} \\ \vdots \\ \alpha_{m2} \end{pmatrix}, \quad \dots, \quad a_N = \begin{pmatrix} \alpha_{1N} \\ \alpha_{2N} \\ \vdots \\ \alpha_{mN} \end{pmatrix},$$

em que  $\alpha_{ij}$  é a quantidade de itens do tipo  $i$  ( $i=1, \dots, m$ ) no  $j$ -ésimo padrão de corte. Além

disso, um vetor  $a_j$  é um padrão de corte se, e somente se,  $\sum_{i=1}^m l_i \alpha_{ij} \leq L$ , com  $\alpha_{ij} \geq 0$  e inteiro.

Variáveis:

- $x_j$ : número de objetos cortados de acordo com o padrão de corte  $j$ ,  $j=1, \dots, N$ .

Modelo Matemático:

$$\text{Minimizar } f(x_j) = \sum_{j=1}^N x_j \quad (1)$$

$$\text{Sujeito a: } \sum_{j=1}^N \alpha_{ij} x_j \geq d_i \quad i = 1, \dots, m \quad (2)$$

$$x_j \geq 0, \text{ e inteiro, } j=1, \dots, N. \quad (3)$$

As restrições (2) asseguram que a quantidade total produzida de itens atente a demanda e, as restrições (3) asseguram que o número de objetos a serem cortados é não negativo e inteiro.

Caso as restrições de demanda (2) sejam de igualdade a minimização do número de objetos (1) é equivalente à minimização da perda de material, ou seja, a função objetivo fica:

$$\text{Minimizar } f(x_j) = \sum_{j=1}^N (L - \sum_{i=1}^m l_i \alpha_{ij}) x_j \quad (4)$$

Vários trabalhos da literatura consideram o problema 1D-SSSCSP e a minimização da perda total de material, dentre eles tem-se: Degraeve e Peeters (2003), Scheithauer e Terno (1995) e Vance (1998), Mobasher e Ekici (2013) que desenvolveram métodos exatos e Hinxman (1980), Stadtler (1990), Wäscher e Gau (1996), Malaguti *et al.* (2013) Powar *et al.* (2013) que apresentam métodos heurísticos. De acordo com Umetani *et al.* (2003), outros objetivos não relacionados a perda de material também são importantes. Considerando o problema de minimização do número de diferentes padrões de corte, tem-se o seguinte modelo matemático:

Variáveis Adicionais:

- $y_j$ : igual 1 se o padrão de corte  $j$  é cortado e igual a 0 caso contrário  $j, j=1, \dots, N$ .

Modelo Matemático:

$$\text{Minimizar } f(y_j) = \sum_{j=1}^N y_j \quad (5)$$

$$\text{Sujeito a: } \sum_{j=1}^N \alpha_{ij} x_j = d_i \quad i = 1, \dots, m \quad (6)$$

$$x_j \leq M y_j \quad j = 1, \dots, N \quad (7)$$

$$x_j \geq 0, \text{ e inteiro, } y_j \in \{0,1\} \quad j=1, \dots, N. \quad (8)$$

A função objetivo (5) minimiza o número de diferentes padrões de corte. As restrições (6) asseguram que a quantidade total produzida de itens atende a demanda; as restrições (7) asseguram que um objeto só é cortado, de acordo com um padrão de corte, se houver preparação para este padrão e, nas restrições (8) tem-se o domínio das variáveis.

Observa-se que, caso a restrição (6) seja modificada para uma desigualdade ( $\geq$ ) no lugar da igualdade, o problema passa a ter solução trivial, ou seja, a solução ótima será o número de padrões necessários para se colocar apenas uma unidade de cada item demandado nestes padrões e repeti-los a quantidade de vezes necessárias para que as demandas sejam atendidas.

Equivalentemente o modelo (5)-(8) poderia ser escrito suprimindo a restrição (7) e substituindo a função objetivo (5) por:

$$\text{Minimizar } f(x_j) = \sum_{j=1}^N \delta(x_j) \quad (9)$$

em que:

$$\delta(x_j) = \begin{cases} 1, & \text{se } x_j > 0 \\ 0, & \text{caso contrário} \end{cases} \quad j=1, \dots, N$$

Yanasse e Limeira (2006) destacam a importância de problemas de decisão multiobjetivo quando os custos da máquina são significantes quando comparados aos custos de material. Assim como Moretti e Sales Neto (2008), Yanasse e Limeira (2006) consideraram um problema multiobjetivo em que a função objetivo considera a minimização do número de objetos a serem cortados e a minimização do número de diferentes padrões de corte, ou seja, tem-se a função objetivo (10) a seguir e, as restrições (6) (substituindo a igualdade pelo desigualdade ( $\geq$ )) e (8).

$$\text{Minimizar } (f^1(x_j), f^2(x_j)) = \left( \sum_{j=1}^N x_j, \sum_{j=1}^N \delta(x_j) \right) \quad (10)$$

Esta função multiobjetivo torna o problema mais difícil de ser resolvido. Alguns artigos da literatura propõem heurísticas para resolver o problema de corte de estoque e consideram relevante a minimização da perda de material, bem como, do número de diferentes padrões de corte: Haessler (1975), Farley e Richardson (1984), Haessler e Sweeney (1991), Diegel *et al.* (1993), Foerster e Wäscher (2000), Vanderbeck (2000), Umetani *et al.* (2003), Yanasse e Limeira (2006), Moretti e Sales Neto (2008), Alves e Valerio de Carvalho (2008), Alves *et al.* (2009), Golfeto *et al.* (2009a), Golfeto *et al.* (2009b), Brandão *et al.* (2011) e Mobasher e Ekici (2013). Henn e Wäscher (2013) apresentam um detalhado survey do problema e dos métodos propostos na literatura.

Embora considerem os dois objetivos relevantes, a maior parte destes artigos não consideram o problema multiobjetivo, mas sim utilizam diferentes estratégias para tornar o problema mono-objetivo, como por exemplo, fixam o número de objetos no valor ótimo, obtido ao se resolver o problema considerando apenas a minimização do número de objetos e, posteriormente, resolvem o problema mono-objetivo de minimização do número de diferentes padrões.

Considerando o problema de minimização do número de ciclos da serra, a quantidade de trabalhos encontrados na literatura é bastante reduzida. Ver, por exemplo, Vanzela (2013). Considere o seguinte modelo matemático:

Dados Adicionais:

- *CAP*: capacidade da serra, ou seja, é quantidade máxima de objetos que a serra pode cortar de forma simultânea. É dada por  $CAP = \lfloor H/w \rfloor$  em que  $H$  é a espessura da serra e  $w$  é a espessura do objeto.

Variáveis Adicionais:

- $z_j$ : número de ciclos da serra cortados de acordo com o padrão  $j, j=1, \dots, N$ .

Modelo Matemático:

$$\text{Minimizar } f(z_j) = \sum_{j=1}^N z_j \quad (11)$$

$$\text{Sujeito a: } \sum_{j=1}^N \alpha_{ij} x_j \geq d_i \quad i = 1, \dots, m \quad (12)$$

$$x_j \leq CAP z_j \quad j = 1, \dots, N \quad (13)$$

$$x_j \text{ e } z_j \geq 0, \text{ e inteiro, } j=1, \dots, N. \quad (14)$$

A função objetivo (11) minimiza o número de ciclos da serra. As restrições (12) asseguram que a quantidade total produzida de itens atente a demanda; as restrições (13) asseguram que o número de objetos cortados respeita a capacidade da serra; nas restrições (14) tem-se o domínio das variáveis.

Yanasse (2008) mostrou que o modelo (11)-(14) poderia ser escrito suprimindo a restrição (13) e substituindo a função objetivo (11) por:

$$\text{Minimizar } f(x_j) = \sum_{j=1}^N \lceil x_j / CAP \rceil \quad (15)$$

Dentre os poucos trabalhos que consideram o problema de minimização dos ciclos da serra tem-se Yanasse *et al.* (1993) que propuseram uma heurística para redução de ciclos da serra para o problema bidimensional. Mosquera e Rangel (2007) propuseram a inclusão de uma restrição de multiplicidade que garante que se um padrão de corte for usado, sua frequência será um múltiplo da capacidade da máquina de corte (*CAP*).

Ranck Jr. (2008) permitiram a utilização de ciclos incompletos por meio de um modelo e de heurísticas para minimização de ciclos e objetos. Chu e Antonio (1999) consideraram uma empresa de corte de tubos de metais em que se tem um problema unidimensional em que um conjunto de objetos que é cortado ao mesmo tempo (um ciclo da serra). Os autores desenvolveram dois algoritmos de programação dinâmica e utilizam uma função que faz a soma ponderada dos objetivos pretendidos. Santos *et al.* (2011) trataram a

minimização de ciclos da serra num problema integrado de dimensionamento de lotes e corte de estoque.

Yanasse (2008) mostrou que considerando um ambiente com altas demandas, resolver o problema de corte de estoque considerando a redução dos ciclos da serra é equivalente a resolver o problema de corte de estoque com as demandas escalonadas por *CAP*. O autor também apresentou um modelo para minimização do número de objetos e ciclos simultaneamente, que consiste no modelo (11)-(14) substituindo a função objetivo (11) por (16):

$$\text{Minimizar } (f^1(x_j), f^2(z_j)) = \left( \sum_{j=1}^N x_j, \sum_{j=1}^N z_j \right) \quad (16)$$

Para tratar o problema de corte com os três objetivos supracitados, propõe-se o seguinte modelo matemático.

Modelo Proposto:

$$\text{Minimizar } (f^1(x_j), f^2(x_j), f^3(x_j)) = \left( \sum_{j=1}^N x_j, \sum_{j=1}^N \delta(x_j), \sum_{j=1}^N \lceil x_j / CAP \rceil \right) \quad (17)$$

$$\text{Sujeito a: } \sum_{j=1}^N \alpha_{ij} x_j \geq d_i \quad i = 1, \dots, m \quad (18)$$

$$x_j \geq 0, \text{ e inteiro, } j=1, \dots, N. \quad (19)$$

A função objetivo (17) minimiza o número de objetos, número de diferentes padrões de corte e número de ciclos da serra. As restrições (18) asseguram o atendimento da demanda e, nas restrições (19) tem-se o domínio das variáveis.

Ao invés de se trabalhar com uma soma dos custos associados a cada um desses objetivos, que são difíceis de mensurar (Moretti e Salles Neto 2008) adotou-se uma abordagem multiobjetivo. Para resolver o problema multiobjetivo, neste artigo o algoritmo genético simbiótico proposto em Golfeto *et al.* (2009a), Golfeto *et al.* (2009b) e Brandão *et al.* (2011) é estendido para o caso em que se consideram os três objetivos conflitantes. Vale ressaltar que também pode ser incorporado outros objetivos, como minimização do número de pilhas abertas. Contudo, não há trabalhos na literatura que trabalham com três, quatro ou mais objetivos no contexto do Problema de Corte Unidimensional.

### 3. Análise do Problema de Corte de Estoque Multiobjetivo (17)-(19)

Antes de se analisar o Modelo proposto (17)-(19) deve-se considerar alguns conceitos básicos sobre problemas de otimização vetorial.

De forma geral o problema de otimização vetorial (POV) pode ser descrito da seguinte forma:

$$\begin{aligned} (POV) \quad & \text{Min } f(x) \\ & \text{s. a: } g(x) \leq 0 \\ & x \in X \end{aligned}$$

onde  $X$  é um conjunto aberto de  $R^n$ ,  $f = (f_1, \dots, f_p): X \subset R^n \rightarrow R^p$  e  $g = (g_1, \dots, g_m): X \subset R^n \rightarrow R^m$  são diferenciáveis.

Para uma precisa definição de uma solução eficiente (Pareto-otimal) para o (POV), foram assumidas as seguintes convenções para igualdades e desigualdades: se  $x = (x_1, \dots, x_n), y = (y_1, \dots, y_n) \in R^n$ , então

$$\begin{aligned} x = y &\Leftrightarrow x_i = y_i, \quad \forall i = 1, \dots, n, \\ x < y &\Leftrightarrow x_i < y_i, \quad \forall i = 1, \dots, n, \\ x \leq y &\Leftrightarrow x_i \leq y_i, \quad \forall i = 1, \dots, n, \\ x \leq y &\Leftrightarrow x_i \leq y_i, \quad \forall i = 1, \dots, n, \text{ e existe } j \text{ tal que } x_j < y_j. \end{aligned}$$

Pode-se agora definir o que seria uma solução eficiente e uma solução fracamente eficiente.

- Definição 1: Um ponto viável  $x^*$  é chamado de solução eficiente para o (POV) se não existe outro ponto viável  $x$ , tal que  $f(x) \leq f(x^*)$ .
- Definição 2: Um ponto viável  $x^*$  é chamado de solução fracamente eficiente para o (POV) se não existe outro ponto viável  $x$ , tal que  $f(x) < f(x^*)$ .

Portanto algumas perguntas que são colocadas quando se estuda um problema de otimização vetorial são: os objetivos são conflitantes? Ou, em que condições eles são equivalentes?

Para buscar responder essa questão foram analisados nesse trabalho os objetivos dois a dois, ou seja: i) número de objetos e número de *setups*; iii) número de objetos e número de ciclos de serra; iii) número de *setups* e número de ciclos de serra.

A literatura possui diversos trabalhos que mostram que, de forma geral, é conflitante minimizar o número de objetos utilizados e o número de *setups* (Alves e Valério de Carvalho, 2009, Golfeto *et al.*, 2009a,b, Henn 2013), bem como é conflitante minimizar número de



objetos e número de ciclos de serra (Yanasse, 2008). Contudo, em casos particulares esses objetivos podem ser equivalentes (Yanasse, 2008).

Não foram encontrados na literatura resultados que determinem, em função dos parâmetros do problema, quando minimizar o número de setups é equivalente a minimizar o número de ciclos de serra. Para essa busca utilizou-se o buscador [www.scholar.google.com](http://www.scholar.google.com) e as palavras-chave: ciclo de serra, problema de corte unidimensional, *setup*, número de diferentes padrões de corte, e seus equivalentes na língua inglesa.

A equivalência entre os problemas significa que a solução eficiente de um problema também é solução eficiente do outro e as funções-objetivo possuem o mesmo valor.

Considerem-se os seguintes problemas de otimização multiobjetivo:

Problema de Corte Biobjetivo 1 – PCB1:

$$\text{Minimizar } (f^1(x_j), f^2(z_j)) = \left( \sum_{j=1}^N x_j, \sum_{j=1}^N z_j \right) \quad (20)$$

$$\text{Sujeito a: } \sum_{j=1}^N \alpha_{ij} x_j \geq d_i \quad i = 1, \dots, m \quad (21)$$

$$x_j \leq CAP z_j \quad j = 1, \dots, N \quad (22)$$

$$x_j \text{ e } z_j \geq 0, \text{ e inteiro, } j = 1, \dots, N. \quad (23)$$

Problema de Corte Biobjetivo 2 – PCB2:

$$\text{Minimizar } (f^1(x_j), f^2(z_j)) = \left( \sum_{j=1}^N c_j x_j, \sum_{j=1}^N y_j \right) \quad (24)$$

$$\text{Sujeito a: } \sum_{j=1}^N \alpha_{ij} x_j \geq d_i \quad i = 1, \dots, m \quad (25)$$

$$x_j \leq M y_j \quad j = 1, \dots, N \quad (26)$$

$$x_j \geq 0, \text{ e inteiro, } y_j \in \{0,1\} \quad j = 1, \dots, N. \quad (27)$$

Vale observar que esses problemas não estão formulados como problemas multiobjetivos na literatura. As proposições abaixo estabelecem duas condições suficientes para a equivalência entre o PCB1 e o PCB2:

**Proposição 1.** *Se  $CAP$  é maior ou igual a  $d_i$  para todo  $i$ , então os problemas PCB1 e o PCB2 são equivalentes.*

Demonstração: Seja  $(x^*, z^*)$  uma solução eficiente do PCB1. Provaremos que, se as hipóteses forem satisfeitas  $(x^*, z^*)$  é também solução eficiente para o PCB2 e os valores das funções objetivos são iguais. Seja  $d = \max\{d_i\}$ . Se  $CAP \geq d_i$ , para todo  $i=1, \dots, n$ , então  $CAP \geq d$ . Como  $d \geq x_j^*$ , para todo  $j=1, \dots, n$ , temos  $CAP \geq d \geq x_j^*$ , para todo  $j=1, \dots, n$ . Tendo em vista que as afirmações abaixo são equivalentes, a proposição está demonstrada:

- I.  $CAP \geq x_j^*$ , para todo  $j=1, \dots, n$ .
- II.  $z_j^* = 0$  se  $x_j^* = 0$  e  $z_j^* = 1$  se  $x_j^* > 0$ .
- III.  $(x^*, z^*)$  é solução eficiente para o PCB2.

Além disso, é fácil ver que os valores das funções objetivos em (20) e (24) serão iguais.

**Proposição 2.** *Sejam  $h_i = \lfloor L/l_i \rfloor$ ,  $s_i = \lfloor d_i/h_i \rfloor$  e  $s = \sum_{i=1}^m s_i$ . Se  $CAP \geq s$ , os problemas PCB1 e o PCB2 são equivalentes.*

Demonstração: Seja  $(x^*, z^*)$  uma solução eficiente do PCB1. Provaremos que, se as hipóteses forem satisfeitas  $(x^*, z^*)$  é também solução eficiente para o PCB2 e os valores das funções objetivos são iguais. Temos que se  $CAP \geq s$  então  $CAP \geq s \geq x_j^*$ , para todo  $j=1, \dots, n$ . Tendo em vista que as afirmações abaixo são equivalentes, a proposição está demonstrada:

- I. Se  $CAP \geq s$  então  $CAP \geq s \geq x_j^*$  para todo  $j=1, \dots, n$ .
- II.  $z_j^* = 0$  se  $x_j^* = 0$  e  $z_j^* = 1$  se  $x_j^* > 0$ .
- III.  $(x^*, z^*)$  é solução eficiente para o PCB2.

Além disso, é fácil ver que os valores das funções objetivos em (20) e (24) serão iguais.

Vale observar que na Proposição 2  $s$  é o número de objetos utilizados na solução homogênea do problema (1)-(3). A proposição também é válida para qualquer outra solução factível deste problema. Por exemplo, poderíamos usar a propriedade MIRUP (*Modified Integer Round-Up Property*) proposta em Scheithauer e Terno (1995).

Nos exemplos a seguir, estão listadas cada uma das soluções eficientes, obtidas após geração de todas as soluções viáveis.

**Exemplo 1:** Seja  $w=20$ ;  $m=4$ ;  $l_1=10$ ;  $l_2=6$ ;  $l_3=5$ ;  $l_4=4$ ;  $d_1=600$ ;  $d_2=153$ ;  $d_3=300$ ;  $d_4=15$ ,  $CAP=3$ .

Tem-se as seguintes soluções eficientes:

$f^1$	$f^2$	$f^3$
429	4	143
451	3	151
453	2	151
456	3	152

**Exemplo 2:** Seja  $w=20$ ;  $m=4$ ;  $l_1=10$ ;  $l_2=6$ ;  $l_3=5$ ;  $l_4=4$ ;  $d_1=600$ ;  $d_2=153$ ;  $d_3=300$ ;  $d_4=15$ ,  $CAP=20$ .

Tem-se as seguintes soluções eficientes:

$f^1$	$f^2$	$f^3$
429	4	23
451	3	24
453	2	22
456	3	24

**Exemplo 3:** Seja  $w=20$ ;  $m=4$ ;  $l_1=10$ ;  $l_2=6$ ;  $l_3=5$ ;  $l_4=4$ ;  $d_1=600$ ;  $d_2=153$ ;  $d_3=300$ ;  $d_4=15$ ,  $CAP=300$ . Tem-se as seguintes soluções eficientes:

$f^1$	$f^2$	$f^3$
429	4	4
451	3	3
453	2	2

No exemplo 3, onde  $CAP$  é consideravelmente elevado, minimizar o número de diferentes padrões é equivalente a minimizar o número de ciclos de serra. Já nos exemplos 1 e 2 é possível notar que os objetivos são, dois a dois, conflitantes.

#### 4. Algoritmo Genético Simbiótico

Para obter uma aproximação das soluções eficientes deste problema foi adaptado, para o problema (17)-(19) com três objetivos, o algoritmo genético simbiótico desenvolvido para o problema de corte com dois objetivos – número de objetos e *setup* - (Golfeto *et al.* 2009a, 2009b).

O Algoritmo Genético Simbiótico (AGS), também chamado de algoritmo cooperativo

(Potter, 1997; Kim et al., 2000), basicamente tem como objetivo dividir o problema estudado em k subproblemas utilizando k diferentes espécies para sua representação computacional. Dividindo o problema em k populações distintas, pode-se obter boas soluções para o problema utilizando estruturas simples, que trabalhando em conjunto, tem-se mostrado mais eficiente que estruturas complexas.

Kim *et al.* (2001, 2006) propõe um algoritmo evolucionário endossimbiótico em que a ideia básica é incorporar a evolução de células eucarióticas (Margullis, 1981) nos já existentes algoritmos simbióticos. Dessa forma, quando um indivíduo encontra um parceiro com alta aptidão em relação a ele, a combinação inteira passa por um processo de evolução sem haver a troca de parceiros.

O método *Symbio* foi introduzido no trabalho de Golfeto *et al.* (2009a) através da aplicação de um algoritmo genético simbiótico para o problema de corte de estoque unidimensional, paralelamente um estudo para o caso multiobjetivo do mesmo problema também foi desenvolvido por Golfeto *et al.* (2009b).

Além disso, uma aplicação do método *Symbio* visando apenas a redução do número de *setups*, ou seja, partindo de uma solução inicial otimizada em relação ao número de objetos processados, também foi realizada com grande sucesso (Golfeto *et al.*, 2009b).

A estrutura computacional do método *Symbio* se baseia no trabalho de Khalifa *et al.* (2006) que apresenta uma solução para o problema de corte utilizando algoritmos genéticos. Neste trabalho os genes são processados aos pares, sendo que o primeiro gene representa a quantidade de vezes que o padrão representado pelo segundo gene será processado. No caso aqui relatado, este segundo gene representará um indivíduo na população de padrões que, como pode ser visto na Figura 1, serve apenas como um ponteiro para população de padrões.

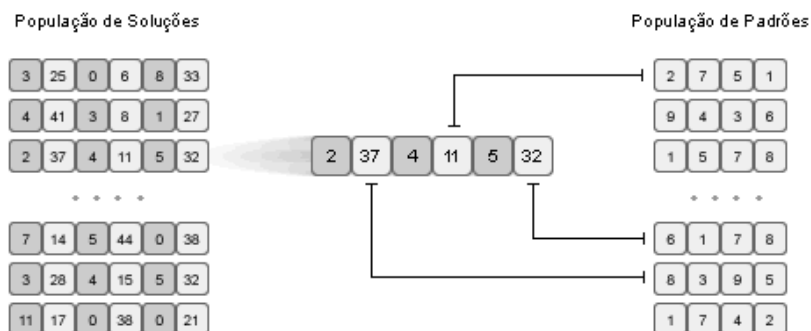


Figura 1: Estrutura dos genes.

Assim, no caso do indivíduo em destaque da Figura 1, tem-se três padrões distintos (37, 11 e 32) sendo executados, respectivamente, duas, quatro e cinco vezes cada, perfazendo um total de onze objetos processados.

Vale ressaltar que a estrutura simbiótica utilizada é idêntica a encontrada no trabalho de Golfeto *et al.* (2009a), já citado anteriormente.

Na construção aqui relatada considera-se um único critério de parada, sendo ele a convergência do algoritmo durante 1.000 gerações consecutivas.

#### 4.1 Indivíduos Soluções

Estão listadas abaixo as configurações gerais da população de soluções:

- I. Tamanho da população: 5.000 indivíduos, apesar do grande custo computacional em lidar-se com uma população deste tamanho, ela apresenta uma grande diversidade, o que é importantíssimo para evitar convergências prematuras do algoritmo;
- II. Tipo de seleção: elitismo - são selecionados os 10% melhores indivíduos da população; este valor foi ajustado de forma experimental, e representa um bom balanceamento entre preservação de indivíduos e geração de novos;
- III. Taxa de recombinação: 90%;
- IV. Tipo de recombinação: uniforme, 70% de chances para o melhor *fitness* (Figura 2);
- V. Taxa de mutação: calcula-se a probabilidade para que dois genes sofram mutação e o substituimos de forma aleatória; caso seja um gene de frequência, será sorteado um valor entre os limites superior e inferior (descritos abaixo), caso contrário será sorteado um indivíduo da população de padrões e um ponteiro será criado (Figura 3);

Pai 1 - 70%	1	8	7	2	5	0	5
Pai 2 - 30%	4	7	5	1	8	1	4
Nova Solução	1	7	7	2	5	0	4

Figura 2: Recombinação Uniforme.

Antes da Mutação	4	7	5	1	7	1	3
Depois da Mutação	2	7	5	3	7	1	3

Figura 3: Mutação de um Indivíduo-Solução.

Deve ser estimado o número máximo de padrões de corte diferentes que um problema poderá ter, para com isso determinar o tamanho da cadeia de DNA dos indivíduos. Adota-se  $m * 2$ , ou seja, o número de genes de nosso indivíduo será o dobro do número máximo de padrões de corte diferentes.

Para os genes ímpares, a frequência do padrão, deve-se definir o limite superior e inferior para restringir a região de soluções do problema. Para o limite inferior atribui-se o valor zero e para o limite superior define-se o valor da maior demanda para um único item, ou seja, o valor do limite superior será exatamente igual a maior demanda possível do pedido. Com isso, é possível que um item apareça em apenas um padrão e tenha toda sua demanda atendida.

O cálculo do *fitness* ( $F_s$ ) do indivíduo-solução é feito da seguinte forma: calcula-se o valor de cada função-objetivo ( $f^1, f^2, f^3$ ) de cada indivíduo, e determina-se, então, para cada indivíduo  $i$ ,  $F_s(i)$  como sendo o número de soluções  $j$  que dominam a solução  $i$ .

### 4.2 *Indivíduos Padrões*

Estão listadas abaixo as configurações gerais da população de padrões, que são significativamente diferentes em comparação a de soluções:

- I. Tamanho da população: 2.000 indivíduos; valores pequenos desta população se mostraram ineficazes para diversas classes de problemas, já que não conseguem obter uma grande diversidade, populações muito grandes ampliaram demais o espaço de busca prejudicando o desempenho do método proposto;
- II. Tipo de seleção: elitismo - são selecionados os 66% melhores indivíduos; este valor também foi obtido de forma experimental;
- III. Taxa de recombinação: 34%;
- IV. Tipo de recombinação: 2 pontos, são sorteados, aleatoriamente, dois pontos de corte para que seja feita a recombinação dos indivíduos;
- V. Taxa de mutação: 90% de um único gene sofrer mutação, que é feita sorteando-se um item para entrar no lugar do gene que sofreu a mutação;

A Figura 4 mostra como a recombinação é realizada na população de padrões.

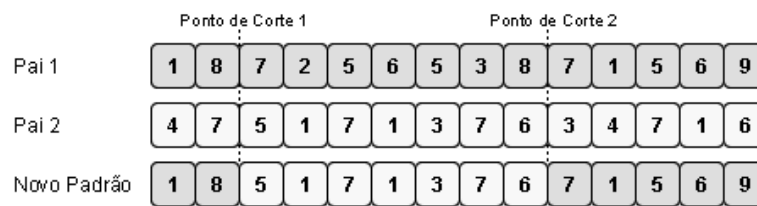


Figura 4: Recombinação Dois-Pontos.

O dimensionamento da cadeia de DNA dos padrões será igual ao maior inteiro menor ou igual ao quociente entre o comprimento da bobina padrão pelo comprimento da menor peça encontrada no pedido. Assim garante-se que é possível formar um padrão utilizando apenas o item de menor comprimento. Isto se faz necessário para que o algoritmo possa cobrir todas as maneiras para resolução do problema.

Porém, com isso, nem todos os padrões poderão utilizar todos os genes disponíveis, pois o comprimento total das peças contidas no padrão excederá o comprimento da peça mestre. Para solucionar este problema, adiciona-se itens ao padrão da esquerda para direita, tendo como referencial o vetor que dará origem ao padrão, se e somente se o padrão tiver comprimento livre suficiente para acomodar o item.

A Figura 5 apresenta como a mutação é realizada na população de padrões. Um gene é desativado quando seu respectivo item excede o tamanho do objeto e, caso contrário, é ativado. Por exemplo, o quinto gene, que corresponde ao item 5, estava desativado antes da mutação, pois a soma dos itens já presentes no padrão impossibilitava sua entrada. Entretanto, após a mutação do quarto gene, que passou a corresponder ao item 3 ao invés do item 2, a entrada do quinto item foi possível, modificando seu estado para ativado.

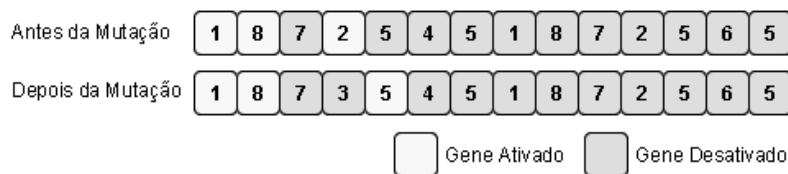


Figura 5: Mutação de um Indivíduo-Padrão.

Para definir o *fitness* de cada padrão ( $F_p$ ) utiliza-se a elite da população de soluções: para cada indivíduo na elite adiciona-se pontos ao *fitness* desse padrão. Quanto melhor a solução maior será a quantidade de pontos, segundo a seguinte regra:

**FAÇA**  $i = 1$  **ATÉ** Elite Soluções  
**PARA CADA** padrão  $j$  contido na solução  $i$  **FAÇA**  
 $F_p(j) \leftarrow F_p(j) + 1 + (1/i)$   
**FIM-PARA**  
**FIM-FAÇA**

### 5. Resultados Computacionais

Nesta seção utilizou-se como base de dados cinco exemplares gerados pelo gerador CUTGEN1 (Gau e Wäscher, 1995). Utilizamos os primeiros exemplares de cada classe indicada na Tabela 1. Nosso algoritmo foi implementado na linguagem Fortran.

Na Tabela 1 listam-se os resultados obtidos pelo método symbio com três funções-objetivo e os comparamos com os resultados descritos em (Golfeto *et al.* 2009b-Método Symbio2009b) e (Umetami *et al.* 2006-Método ILS). Na coluna Setup tem-se o número de setups, nas colunas relativa a cada método tem-se o número de objetos cortados e, na coluna ciclos da serra tem-se o número de ciclos da serra obtido pelo método proposto neste trabalho.

Pode-se notar que, apesar de se trabalhar com mais um objetivo, o método continua sendo competitivo em termos de número de setups e número de objetos. Infelizmente não foram encontrados na literatura outros artigos que trabalham com os três objetivos por meio de uma abordagem multiobjetiva, de forma que não foi possível comparar o número de clicos da serra com outros trabalhos da literatura.

Tabela 1: Resultados Computacionais

Exemplar	Setup	ILS	Symbio2009b	Symbio2012	Ciclos de Serra
Cutgen02	8		128		
	7		129		
	5	123	130		
	4	126	134	128	28
	3	129	150	129	27
	2	139		144	29
Cutgen03	6		25	26	7
	5			29	9
	5		26	28	8
	4	24	27	30	8
	3	25	30	36	8
Cutgen07	7		80		
	5	66		71	17
Cutgen09	15	91	96		
	14	92			
	13	93	97		
	12			97	26
	11	94	98	100	25
Cutgen15	10	97	99	103	24
	13	127	127	127	30



## 6. Conclusão e Propostas Futuras

Neste trabalho estudou-se um problema de corte de estoque multiobjetivo. Alguns resultados teóricos foram demonstrados. Foi proposto uma adaptação do algoritmo genético simbiótico para o problema em questão. Os resultados computacionais mostram que a abordagem é competitiva se compararmos com outros métodos da literatura que consideram apenas dois objetivos, número de objetos processados e setup. Desta forma, é possível afirmar que esse trabalho será uma referência por ser o primeiro a tratar dos três objetivos simultaneamente numa abordagem multiobjetiva.

Como propostas futuras, além de possíveis melhorias no método proposto, pretende-se fazer testes com outras bases de dados, como por exemplo, a base de dados proposta em Umetani *et al.* (2003).

## Agradecimentos

Os autores agradecem aos revisores anônimos da revista pelos relevantes comentários, críticas e sugestões.

## Referências

Alves, C., Macedo, R., Valério de Carvalho, J. (2009), New Lower Bounds Based on Column Generation and Constraint Programming for the Pattern Minimization Problem, *Computers and Operations Research*, 36, 2944-2954.

Alves, C., Valério de Carvalho, J. (2008), A branch-and-price-and-cut algorithm for the pattern minimization problem, *RAIRO Operations Research*, 42, 435-453.

Brandão, J.S., Coelho, A.M., Vasconcellos, J.F.V., Salles Neto, L.L., Pinto, A.V. (2011), Application of genetic algorithm to minimize the number of objects processed and setup in a one-dimensional cutting stock problem, *International Journal of Applied Evolutionary Computation*, 2, 34-48.

Chu, C., Antonio, J. (1999), Approximation algorithms to solve real-life multicriteria cutting stock problems, *Operations Research*, 47, 495-508.

Degraeve, Z., Peeters, M. (2003), Optimal integer solution to industrial cutting stock problems: part 2. benchmark results, *INFORMS Journal on Computing*, 15, 58-81

Diegel, A., Chetty, M. Van Schalkwyk, S., Naidoo, S. (1993), Setup combining in the trim loss problem—3to2 & 2to1, Working paper, Business Administration. University of Natal. Durban. First Draft.

Farley, A. A. e Richardson, K. V. (1984), Fixed charge problems with identical fixed charges, *European Journal of Operational Research*, 18, 245-249.

Foerster, H., Wäscher, G. (2000), Pattern reduction in one-dimensional cutting stock problem, *International Journal of Production Research*, 38, 1657-1676.

Gau, T., Wäscher, G., 1995. CUTGEN1: A problem generator for the one-dimensional cutting stock problem. *European Journal of Operational Research*, 84(3), 572-579.

Golfeto, R.R., Moretti, A.C., Salles Neto, L.L. (2009a), A genetic symbiotic algorithm applied to the one-dimensional cutting stock problem, *Pesquisa Operacional*, 29(3), 365-382.

Golfeto, R.R., Moretti, A.C., Salles Neto, L.L. (2009b), A genetic symbiotic algorithm applied to the cutting stock problem with multiple objectives, *Advanced Modeling and Optimization*, 11, 473-501.

Haessler, R.W. (1975), Controlling cutting pattern changes in one-dimensional trim problems, *Operations Research*, 23, 483-493.

Haessler, R. W. e Sweeney, P. E. (1991), Cutting stock problem and solution procedures, *European Journal of Operational Research*, 54, 141-150.

Henn S., Wäscher,G. (2013), Extensions of cutting problems: setups, *Pesquisa Operacional*. 33.,2, 133-162.

Hinxman, A. (1980), The trim-loss and assortment problems: a survey, *European Journal of Operational Research*, 5, 8-18.

Khalifa, Y., Salem, O. e Shahin, A. (2006), Cutting Stock Waste Reduction Using Genetic Algorithms, *Proceedings of the 8th Conference on Genetic and evolutionary computation*, pp. 1675-1680.

Kim, Y. K., Kim, J. Y. and Kim, Y. (2000), A coevolutionary algorithm for balancing and sequencing in mixed model assembly lines, *Applied Intelligence*, 13, pp: 247-258.

Kim, J. Y., Kim Y. and Kim Y. K. (2001), An endosymbiotic evolutionary algorithm for optimization, *Applied Intelligence*, 15, pp: 117-130.

Kim, Y. K., Kim, J. Y. and Kim, Y. (2006), An endosymbiotic evolutionary algorithm for the integration of balancing and sequencing in mixed-model U-lines, *European Journal of Operational Research*, 168, 838-852.

Solution approaches for the cutting stock problem with setup cost

Malaguti, E., Medina Durán, R., & Toth, P. (2013). Approaches to real world two-dimensional cutting problems. *Omega*.

Mobasher, A., Ekici A. (2013), Solution approaches for the cutting stock problem with setup cost, *Computers & Operations Research*, 40, 225–235.

Moretti, A.C., Salles Neto, L.L. (2008), Nonlinear cutting stock problem model to minimize the number of different patterns and objects, *Computational and Applied Mathematics*, 27, 61-78.

Mosquera, G. P., Rangel, S. (2007), Redução de ciclos da serra no problema de corte de estoque bidimensional na indústria de móveis, *Anais do XXX CNMAC*.

Potter, M. A. (1997), The design and analysis of a computational model of cooperative coevolution, Ph.D. Dissertation, George Mason University.

Powar, P. L., Jain, V., Saraf, M., & Vishwakarama, R. (2013). One-Dimensional Cutting Stock Problem (1D-CSP): A New Approach. *International Journal of Advanced Computing, Recent Science Publication*, 46(2), 1223-1228.

- Ranck Jr. R. (2008), Desenvolvimento de alguns métodos de solução para o problema de redução de ciclos da serra, Dissertação de Mestrado, INPE.
- Santos, S. G., Araujo, S. A., Rangel, M. S. N. (2011). Integrated cutting machine programming and lot sizing in furniture industry. *Pesquisa Operacional para o Desenvolvimento*, 3 (1), 249-266.
- Scheithauer, G. and Terno, J. (1995), The modified integer round-up property of the one-dimensional cutting stock problem, *European Journal of Operational Research*, 84, 562-571.
- Stadtler, H. (1990), A one-dimensional cutting stock problem in the aluminium industry and its solution, *European Journal of Operational Research*, 44, 209-223
- Umetani, S., Yagiura, M., Ibaraki, T. (2003), One-dimensional cutting stock problem to minimize the number of different patterns, *European Journal of Operational Research*, 146, 388-402.
- Umetani, S., Yagiura, M., Ibaraki, T. (2006), One-dimensional cutting stock problem with a given number of setups: a hybrid approach of metaheuristics and linear programming, *Journal of Mathematical Modelling and Algorithms*, 5, 43-64.
- Vance, P. (1998), Branch-and-price algorithms for the one-dimensional cuttings stock problem, *Computational Optimization and Applications*, 9, 212-228.
- Vanderbeck, F. (2000), Exact algorithm for minimizing the number of setups in the one-dimensional cutting stock problem, *Operations Research*, 48, 915-926.
- Vanzela, M., Rangel, S., & de Araujo, S. (2013, May). The Integrated Lot Sizing and Cutting Stock Problem in a Furniture Factory. In *Intelligent Manufacturing Systems*, 11, 390-395.
- Wäscher, G., Gau, T. (1996), Heuristics for the integer one-dimensional cutting stock problem: a computational study, *OR Spektrum*, 18, 131-144.
- Wäscher, G., Haußner, H. e Schumann, H. (2007), An improved typology of cutting and packing problems, *European Journal of Operational Research*, 183, 1109-1130.
- Yanasse, H. H., Limeira, M. S. (2006), A hybrid heuristic to reduce the number of different patterns in cutting stock problems, *Computers and Operations Research*, 33, 2744-2756.
- Yanasse, H. H. (2008), A note on the minimization of the number of cutting cycles problem, *Livro de resumos do XI SPOLM*. v. 1, p. 24-24.
- Yanasse, H. H., Harris, R. G. e Zinober, A. S. I. (1993), Two-dimensional cutting stock with multiple stock sizes, *Journal Operations Research Society*, 42, 673-683