

UM ALGORITMO GENÉTICO MULTI OBJETIVO APLICADO AO PROBLEMA DE CORTE COM DIMENSÃO ABERTA GUILHOTINADO

Dayanne Gouveia Coelho¹, Marcelus Xavier Oliveira¹, Elizabeth Fialho Wanner¹,
Sergio Ricardo de Souza¹, Marcone Jamilson Freitas Souza²

1. Centro Federal de Educação Tecnológica de Minas Gerais (CEFET-MG)
Av. Amazonas, 7675, CEP 30510-000, Belo Horizonte – MG, Brasil.

E-mails: dayagc@gmail.com, marcelusxavier@gmail.com, efwanner@gmail.com,
Sergio@dppg.cefetmg.br

2. Universidade Federal de Ouro Preto – UFOP
Campus Universitário – ICEB
CEP 30400-000, Ouro Preto – MG, Brasil.

E-mails: marcone@iceb.ufop.br

RESUMO: Este artigo apresenta uma aplicação do algoritmo genético multiobjetivo SPEA 2 (*Strength Pareto Evolutionary Algorithm 2*), em conjunto com duas heurísticas de encaixe, *Best-Fit* (BF) e *Best-Fit Decreasing Height* (BFDH), para resolver o Problema de Corte com Dimensão Aberta (PCDA) guilhotinado. O PCDA multiobjetivo consiste em alocar um conjunto de peças menores (itens) em uma peça maior (objeto), de forma que sejam minimizados a altura utilizada deste objeto e o número de cortes realizados. Para resolver o problema, incorpora-se ao SPEA 2 a fase de construção do algoritmo GRASP (*Greedy Randomized Adaptive Search Procedure*), para gerar parte da população inicial. A metodologia proposta foi testada em um conjunto de problemas-testes da literatura, verificando-se que os resultados obtidos são satisfatórios, à medida que foi possível gerar, para todos os problemas teste, um conjunto representativo de soluções.

PALAVRAS CHAVE: Problema de Corte com Dimensão Aberta, SPEA 2, Heurísticas de Encaixe.

ABSTRACT: This paper presents an application of genetic multiobjective algorithm SPEA 2, together with two fit heuristics, Best-Fit (BF) and Best-Fit Decreasing Height (BFDH), to solve the guillotined Open Dimensional Cutting Problem (ODP). The multiobjective ODP consists of allocating a set of smaller parts (items) in a larger piece (object) in order to minimize the height of this object used and the number of cuts. To solve the problem, the construction phase of GRASP (*Greedy Randomized Adaptive Search Procedure*) is incorporated to SPEA 2 to generate part of the initial population. The proposed methodology was tested on a set of instances from the literature. The obtained results are satisfactory, since it was possible to generate, for all test problems, a representative set of solutions.

KEYWORDS: Open Dimensional Problem, SPEA 2, Fit Heuristics.

1. INTRODUÇÃO

Em muitas indústrias, o Problema de Corte (PC) é essencial para o planejamento da produção. A matéria-prima utilizada nas indústrias de vidro, de papel e metalúrgicas, por exemplo, é produzida no primeiro momento em tamanhos grandes e padronizados. Conforme a necessidade, para atender às demandas internas ou externas das indústrias, essa matéria-prima será reduzida a itens de tamanhos menores, variados, e geralmente não padronizados. Planejar como será o corte não é uma tarefa fácil, visto que esta operação gera perdas da matéria-prima. Porém, um bom planejamento evita a necessidade de constantes preparações das máquinas para os tamanhos dos produtos requisitados, o que irá minimizar os efeitos negativos gerados pelo desperdício sobre os custos de produção.

Neste trabalho, será estudado o Problema de Corte com Dimensão Aberta (PCDA), um caso particular dos PC. Para definir o problema, foi utilizada a tipologia de (WÄSCHER, HAUBNER e SCHUMANN, 2007), que denomina o problema como *Open Dimensional Problem* (ODP). O PCDA consiste em determinar o melhor arranjo de um conjunto de itens retangulares, sobre um objeto maior, que possui largura fixa e altura variável, com o objetivo de minimizar a altura utilizada e o número de cortes realizados. Esta variação do problema pode ser encontrada, principalmente, em indústrias que realizam o corte de bobinas ou rolos.

Alguns trabalhos, como (HIFI, 1998), (LIU e TENG, 1999) e (ANDRADE, 2009), abordam metodologias baseadas em algoritmos evolutivos para resolver o PCDA em uma versão mono-objetivo. Tal escolha se deve ao fato desses métodos serem flexíveis e capazes de gerar soluções de boa qualidade em problemas de grande porte, em um tempo computacional viável.

Recentemente, trabalhos apresentando uma abordagem multiobjetivo para o PCDA têm sido propostas. Em (TIWARI e CHAKRABORTI, 2006) é feito um estudo sobre o problema guilhotinado e sobre o problema não-guilhotinado, avaliando os resultados obtidos para cada uma dessas versões. Além disso, é feita uma comparação entre esses dois problemas, verificando-se que, com o acréscimo da restrição da guilhotina, o número de cortes para uma determinada instância aumenta. Em (ILLICH, WHILE e BARONE 2007) estuda-se o PCDA não-guilhotinado admitindo-se a rotação dos itens. São comparados dois algoritmos multiobjetivo, um puramente determinístico e outro de natureza evolucionária. Segundo os autores ambos os algoritmos retornam um conjunto de soluções razoável, mas verifica-se para algumas instâncias os métodos retornam apenas um ponto para representar a curva de compromisso entre os objetivos. Nos trabalhos de (ARMAS, *et al.*, 2009) e (MIRANDA, *et al.*, 2010), o PCDA possui a restrição do corte guilhotinado e admiti-se, além disso, a rotação dos itens durante o processo. Em todos esses trabalhos, as funções objetivo tratadas são o desperdício de matéria-prima e a velocidade de operação do equipamento que realiza o corte. O segundo objetivo é alcançado minimizando-se o número de cortes realizados durante o processo. Além disso, para resolver o PCDA multiobjetivo, são utilizadas estratégias evolutivas.

Este trabalho apresenta uma metodologia de solução para o PCDA multiobjetivo, considerando o corte guilhotinado e não admitindo-se a rotação dos itens. É utilizado o algoritmo genético multiobjetivo SPEA 2 (*Strength Pareto Evolutionary Algorithm 2*), em conjunto com as heurísticas de encaixe *Best Fit* (BF) e *Best-Fit Decreasing Height* (BFDH), para a solução do problema tratado.

O artigo está estruturado como segue: a seção 2 descreve o PCDA multiobjetivo; a seção 3 apresenta a metodologia proposta; a seção 4 apresenta e faz uma análise dos resultados computacionais gerados, aplicando-se a metodologia proposta para instâncias da literatura; a seção 5 apresenta as conclusões obtidas a respeito do trabalho realizado.

2. DESCRIÇÃO DO PROBLEMA ESTUDADO

O Problema de Corte com Dimensão Aberta (PCDA) é caracterizado por tratar do corte de apenas um objeto e por possuir dimensões variáveis, sendo utilizado em itens e objetos com mais de uma dimensão. Este problema é caracterizado, segundo (ANDRADE, 2009), por:

- i) Alocar os itens de forma ortogonal no objeto;
- ii) Os itens apresentarem orientação fixa;
- iii) O corte ser do tipo guilhotinado;
- iv) O corte ser feito em 2-estágios.

O problema consiste em um objeto S , de largura W e altura grande o suficiente (“altura infinita”) para alocar todos os itens, e uma lista de itens retangulares $I = \{r_1, \dots, r_n\}$, em que cada item $r_i = (w_i, h_i)$, tal que $w_i \leq W$, para $i = 1, \dots, n$, possui largura w_i e altura h_i , como mostra a Figura 1.

A formulação matemática do PCDA mono-objetivo, segundo (LODI, MARTELLO e VIGO, 2004), apresenta as seguintes proposições para uma dada solução:

- i) O primeiro item alocado em cada faixa mais a esquerda possui a maior altura;
- ii) A primeira faixa do objeto (a faixa mais baixa) é a mais alta;
- iii) Os itens são ordenados de forma decrescente em relação à altura, de modo que $(h_1 \geq h_2 \geq \dots \geq h_n)$.

A altura de cada faixa corresponde à altura h_i do primeiro item i a ser alocado. Dessa forma, considerando n o número de faixas formadas para alocar os itens, a variável de decisão é definida como:

$$y_i = \begin{cases} 1, & \text{se o item } i \text{ inicia a faixa } i, \quad (i = 1, \dots, n); \\ 0, & \text{caso contrário.} \end{cases}$$

Além disso, apenas os itens j , tal que $j > i$, podem ser colocados na faixa. Isso se justifica pois, dado um item j , tal que $j = i$ inicia a faixa i , ele não poderá ser atribuído novamente a nenhuma outra faixa. Dessa forma, a variável de decisão x_{ij} é definida como:

$$x_{ij} = \begin{cases} 1, & \text{se o item } j \text{ está alocado na faixa } i, \quad (i = 1, \dots, n-1); j > i \\ 0, & \text{caso contrário.} \end{cases}$$

Com base nessas considerações, a formulação matemática do problema mono-objetivo é dada por:

Minimizar:

$$\sum_{i=1}^n h_i \cdot y_i \quad (1)$$

Sujeito a:

$$\sum_{i=1}^{j-1} x_{ij} + y_j = 1, \quad j = 1, \dots, n; \quad (2)$$

$$\sum_{j=i+1}^n w_j x_{ij} \leq (W - w_i) y_i, \quad i = 1, \dots, (n-1); \quad (3)$$

$$\begin{aligned} x_{ij} &\in \{1,0\} & i = 1, \dots, (n-1); & j > i \\ y_i &\in \{0,1\} & i = 1, \dots, n; \end{aligned} \quad (4)$$

A função objetivo mostrada na expressão (1) tem, como critério, minimizar a altura utilizada do objeto. A restrição (2) garante que o item só será alocado uma vez no objeto; a restrição (3) garante que a largura do objeto não será ultrapassada em nenhuma das faixas; e, por fim, as restrições (4) definem o tipo de variável do problema.

O PDCA multiobjetivo possui as mesmas características do PCDA mono-objetivo, com o acréscimo de uma segunda função objetivo, que busca minimizar o número de cortes realizado durante o processo. Esta segunda função calcula o número de cortes necessários durante o processo de corte. Reduzindo este número, o tempo gasto com o processo de corte dentro da indústria também será menor. O número de cortes é definido pelo número distinto de arestas (ou linhas) dentro do objeto a ser cortado, excluindo-se as margens no exterior do objeto.

3. METODOLOGIA PROPOSTA

Nesta seção, é mostrado como é representada uma solução para o PCDA; o algoritmo de encaixe utilizado para alocar os itens dentro do objeto; a função de avaliação de uma solução; os operadores genéticos utilizados; bem como apresentado o Algoritmo Genético multiobjetivo utilizado para resolver o problema.

3.1. REPRESENTAÇÃO DE UMA SOLUÇÃO

Cada solução do PCDA é representada por uma sequência de números inteiros, em que cada posição da sequência representa a ordem na qual o item é encaixado. Para um problema com seis itens, por exemplo, uma possível solução é $s = (2\ 1\ 3\ 6\ 5\ 4)$. A solução s indica que o primeiro item a ser encaixado é o item 2 e o último é o item 4. Assim, os indivíduos de uma determinada população serão representados por permutações deste conjunto de itens.

3.3. TÉCNICAS DE ENCAIXE

Técnicas de encaixe (ou algoritmos de níveis) são algoritmos aproximados, desenvolvidos para tratar do encaixe de itens em problemas de corte (ORTMANN, 2010). Nesses algoritmos, os itens são alocados em faixas, da esquerda para a direita, sendo que o início de uma nova faixa coincide com o topo do item mais alto da faixa anterior.

A utilização destas técnicas se deve ao fato delas serem mais rápidas e gerarem padrões guilhotinados. Neste trabalho, são utilizadas as técnicas de encaixe *Best-Fit* (BF) e *Best-Fit Decreasing Height* (BFDH) (para maiores detalhes a respeito dessas técnicas, ver (ORTMANN, 2010)). Em ambas as técnicas, o encaixe dos itens é feito na ordem em que eles aparecem na solução. Os itens são inseridos no canto inferior esquerdo do objeto até que a largura do objeto não seja mais suficiente.

3.3.1 BEST FIT (BF)

Na técnica BF, após inserir o primeiro item, pesquisa-se a largura disponível da faixa. Se o item seguinte não puder ser inserido nesta faixa, verifica-se entre os demais, na ordem em que aparecem na solução, se é possível inseri-lo ou não. Uma nova faixa só é formada quando nenhum outro item puder ser inserido na faixa atual. A Figura 1 (b) mostra o resultado da aplicação desta técnica, para o encaixe da sequência de cinco itens mostrado na Figura 1 (a). O pseudocódigo da técnica BF é apresentado na Figura 2.

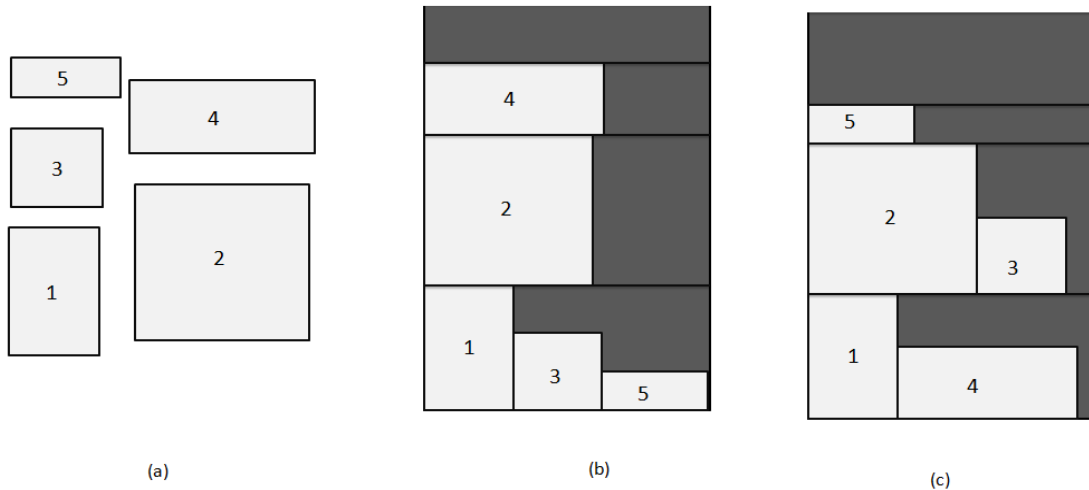


Figura 1: A figura apresenta uma solução para PCDA guilhotinado envolvendo 5 itens. Estes itens são encaixados em uma bobina de acordo com o Algoritmo de Nível (a) BF e (b) BFDH.

Algoritmo Best-Fit ($I_i = (w_i, h_i), W$):

Início

- 1 $n \leftarrow$ quantidade de itens
- 2 $Faixa = 1, i = 1$;
- 3 Insira o primeiro item na primeira faixa;
- 4 $Altura(faixa) = h(I_i)$;
- 5 $Largura(faixa) = W - w(I_i)$;
- 6 **Para** $i = 2$ até n **faça**
- 7 **Se** o item i puder ser inserido na faixa;
- 8 Insere o item e atualiza largura da faixa;
- 9 **Senão**
- 10 Insere os próximos itens que couberem;
- 11 $faixa = faixa + 1$;
- 12 $Largura(faixa) = W - w(I_i)$;
- 13 $Altura(faixa) = h(I_i)$;
- 14 **Fim Se**
- 15 **Fim Para**
- 16 Retorne a sequência do encaixe

Fim Best-Fit

Figura 2: Algoritmo Best-Fit

Algoritmo Best-Fit Decreasing Height ($I_i = (w_i, h_i), W$):

Início

- 1 $n \leftarrow$ quantidade de itens
- 2 $Faixa = 1, i = 1$;
- 3 Insira o primeiro item na primeira faixa;
- 4 $Altura(faixa) = h(I_i)$;
- 5 $Largura(faixa) = W - w(I_i)$;
- 6 **Para** $i = 2$ até n **faça**
- 7 Insira o item na faixa que resultar o melhor encaixe;
- 8 **Se** o item não puder ser inserido em nenhuma das faixas então
- 9 Forme uma nova faixa;
- 10 $faixa = faixa + 1$;
- 11 $Altura(faixa) = h(I_i)$;
- 12 **Fim Se**
- 13 **Fim Para**
- 14 Retorne a sequência do encaixe

Fim Best-Fit Decreasing Height

Figura 3: Algoritmo Best-Fit Decreasing Height

3.3.2 BEST-FIT DECREASING HEIGHT (BFDH)

Na técnica BFDH, após inserir o primeiro item, pesquisa-se a largura disponível das faixas existentes. O próximo item é inserido na faixa que resultar no menor espaço residual em relação à largura. Uma nova faixa só é formada se o item não puder ser inserido em nenhuma das faixas existentes. Sendo assim, é permitido que as faixas formadas fiquem abertas, possibilitando que o item seja inserido apenas na faixa que resultar no melhor encaixe, como pode ser observado em (ORTMANN, 2010). A Figura 1(b) mostra o resultado

da aplicação desta técnica à sequência de cinco itens apresentados na Figura 1(a) . O pseudocódigo da técnica BFDH é apresentado na Figura 3.

3.4 STRENGTH PARETO EVOLUTIONARY ALGORITHM (SPEA2)

Os Algoritmos Genéticos (AGs) são métodos flexíveis e robustos que conseguem produzir soluções de boa qualidade em problemas complexos com um tempo computacional viável (HOLLAND, 1975). Devido a essa flexibilidade e a eficácia na realização de uma busca global em diversos ambientes, os AGs vêm sendo aplicados com sucesso na resolução de problemas de otimização combinatória.

Os problemas multiobjetivo admitem diferentes funções objetivo que, em geral, são conflitantes entre si. Resolver um problema como este não é uma tarefa trivial, uma vez que a melhora em um objetivo ocorre em detrimento do outro objetivo. A solução de um problema multiobjetivo não é representada por apenas uma solução, mas por um conjunto de soluções eficientes, que representam relações de compromisso entre os objetivos considerados no problema.

Para resolver estes problemas, os métodos baseados em metaheurísticas têm sido uma alternativa muito utilizada (ZITZLER, 2002). A preferência pelos AGs se deve ao fato desses algoritmos trabalharem com uma população de soluções que podem conter informações sobre diferentes espaços de buscas, além de poder gerar, em apenas uma execução, um conjunto de soluções eficientes, denominado por conjunto *Pareto-ótimo*. Esse conjunto contém as soluções das diferentes funções objetivo do problema. As soluções presentes neste conjunto caracterizam a relação conflitante existente neste tipo de problema, ou seja, neste conjunto não é possível melhorar o valor de um critério sem tornar o outro pior.

Dentre as formulações de AGs multiobjetivo encontradas na literatura, as que mais se destacam são o *Multi-objective Genetic Algorithm* (MOGA), o *Nondominated Sorting Genetic Algorithm* (NSGA) e o *Strength Pareto Evolutionary Algorithm* (SPEA) (ZITZLER, 2002).

SPEA 2 (N, \bar{N}, T)

Entrada:

N (Tamanho da população);

\bar{N} (tamanho do arquivo);

T (Número máximo de gerações).

Saída:

A (conjunto das soluções não dominadas)

Início: Gera uma população inicial P_0 e cria um arquivo vazio (conjunto externo) $\bar{P}_0 = \emptyset$. Faça $t = 0$.

Atribuição de aptidão: Calcula os valores de aptidão dos indivíduos de P_t e \bar{P}_t . (A aptidão de cada indivíduo da população atual é calculada de acordo com as forças de todas as soluções não dominadas do conjunto \bar{P}_t que dominam esse determinado indivíduo e por todas as soluções dominadas por esse indivíduo.).

Seleção Ambiental: Copia todos os indivíduos não dominados de P_t e \bar{P}_t para \bar{P}_{t+1} . Se o tamanho de $\bar{P}_{t+1} < \bar{N}$, então reduz \bar{P}_{t+1} por meio do operador de truncamento; caso contrário, se o tamanho de $\bar{P}_{t+1} > \bar{N}$, então preenche \bar{P}_{t+1} com os indivíduos dominados de P_t e \bar{P}_t .

Término: Se $t \geq T$ ou outro critério de parada satisfeito, então se defini um conjunto de vetores de decisão, representado pelos indivíduos não dominados de \bar{P}_t . Parar.

Seleção por Reprodução: Fazer a seleção usando o método do torneio binário, com a substituição de \bar{P}_{t+1} , a fim de preencher o conjunto para a reprodução.

Variações: Aplica os operadores de recombinação e mutação após o processo de reprodução e define \bar{P}_{t+1} para a população resultante. Incrementar o contador de geração ($t = t + 1$). Voltar ao passo 2.

Figura 4: Pseudocódigo do Algoritmo SPEA 2

Neste artigo, para resolver o PCDA multiobjetivo, é utilizado o *Strength Pareto Evolutionary Algorithm 2* (SPEA2) (ZITZLER et al., 2002). O SPEA2 possui a estrutura básica de um algoritmo genético multiobjetivo, utilizando uma população regular e um arquivo externo. Este arquivo externo contém as soluções não dominadas encontradas pelo algoritmo, de modo a garantir a preservação das melhores soluções encontradas. Este algoritmo se destaca por apresentar uma boa convergência e garantir diversidade no conjunto de soluções retornado. Ele apresenta um esquema para atribuir melhor aptidão, o que determina, para cada indivíduo, quantos outros indivíduos ele domina e por quantos ele é dominado.

Em geral, o SPEA 2 apresenta a estrutura descrita na Figura 4. O algoritmo utiliza uma estratégia para atribuir aptidão que incorpora informações de densidade. O tamanho do arquivo externo é fixo. Assim, quando o número de indivíduos não dominados é menor que o tamanho pré-definido para este arquivo, ele será preenchido por indivíduos não dominados.

3.4.1. CONSTRUÇÃO DA POPULAÇÃO INICIAL

A população inicial do SPEA 2 é constituída por 30% de indivíduos gerados de forma aleatória e 70% utilizando a fase de construção do Algoritmo GRASP (*Greedy Randomized Adaptive Search Procedure*), proposto por (FEO e RESENDE, 1995).

Fase de construção do GRASP ($g(\cdot), k, s$).

Início

1. $s^* \leftarrow \emptyset$;
2. Inicia o conjunto C de itens candidatos;
3. Ordena o conjunto C de acordo com $g(\cdot)$;
4. **Enquanto** $C \neq \emptyset$ faça
5. $LRC = \{\text{conjunto dos } k \text{ melhores itens de } C\}$;
6. Selecione aleatoriamente, um item $t \in LRC$;
7. $s^* \leftarrow s^* \cup \{t\}$;
8. Atualize o conjunto C de itens candidatos;
9. **fim-enquanto**;
10. $s \leftarrow \text{Best-Fit Decreasing Height de } s^*$;
11. Retorne s ;

fim

Figura 5: Pseudocódigo do método da fase de construção do GRASP

Na fase de construção, cada solução é construída pela inserção dos itens, sendo um de cada vez. O pseudocódigo dessa fase está representado na Figura 5.

Este procedimento inicia-se na linha 2 do Algoritmo descrito na Figura 5, com a construção de uma lista de itens candidatos C . Os elementos (itens) desta lista são ordenados de forma decrescente, de acordo com a função $g(t)$ que avalia o benefício do candidato ser incluído na solução. A função $g(\cdot)$ retorna, para cada item t da lista C , uma nota, que representa a altura deste item. A cada iteração da fase de construção, os k melhores itens da lista C são colocados em uma Lista Restrita de Candidatos (LRC). Para cada inserção na solução, é escolhido aleatoriamente um item dentre os k itens candidatos da LRC . Em seguida, as listas C e LRC são atualizadas e o processo é repetido, até que todos os itens tenham sido incluídos na solução.

O encaixe dos itens da solução inicial gerada é feito utilizando-se uma técnica de encaixe (*Best-Fit* ou *Best-Fit Decreasing Height*), descritas na seção 3.3.

Os 70% da população inicial SPEA 2, construídos pela fase de construção do GRASP, são gerados de forma que 20% são construídos com $k = 2$, 25% com $k = 3$ e 25% com $k = 5$.

3.4.2 FUNÇÃO DE AVALIAÇÃO

Neste trabalho, são utilizadas, como função de avaliação, as duas funções objetivo do problema. Dessa forma, cada indivíduo recebe duas notas, que correspondem à altura utilizada do objeto e ao número de cortes realizado.

3.4.3 OPERADORES GENÉTICOS

Para descrever os operadores genéticos utilizados neste trabalho será usada, como exemplo, uma solução com 6 itens.

CROSSOVER OX

Este operador constrói os cromossomos filhos escolhendo, de um pai, uma sequência de corte, e preservando, do outro pai, a ordem de corte dos demais itens. Será usada uma definição deste operador de acordo com (SOUZA, 2009).

Considere os pais P_1 e P_2 abaixo e a escolha aleatória de dois pontos desses pais para fazer o corte. Para este exemplo, o corte será feito nas posições 3 e 4:

$$P_1 = [6 \ 1 \ |3 \ 4| 5 \ 2] \ e \ P_2 = [1 \ 2|4 \ 6| 5 \ 3]$$

Os filhos F_1 e F_2 herdam a sequência de itens dos pais P_1 e P_2 , respectivamente, nas posições que foram feitos os cortes:

$$F_1 = [X \ X|3 \ 4| X \ X] \ e \ F_2 = [X \ X| 4 \ 6| X \ X]$$

Partindo-se do segundo corte de P_1 , é criada uma lista com os n itens deste pai:

$$L_1 = 5 - 2 - 6 - 1 - 3 - 4.$$

De forma análoga, cria-se uma lista para P_2 , com a ordem de itens a partir do segundo corte:

$$L_2 = 5 - 3 - 1 - 2 - 4 - 6.$$

Os itens contidos nas listas L_1 e L_2 são inseridos nos filhos F_2 e F_1 , respectivamente, a partir do segundo corte, seguindo a sequência em que os itens se encontram nas listas. Os itens que já estão contidos nos filhos não são inseridos. Deste modo, os filhos formados são:

$$F_1 = [2 \ 6 \ 3 \ 4 \ 5 \ 1] \ e \ F_2 = [1 \ 3 \ 4 \ 6 \ 5 \ 2].$$

MUTAÇÃO

O operador de mutação nos problemas que utilizam permutação deve garantir que o indivíduo obtido seja factível. Para fazer a mutação, são escolhidos p pontos de corte no cromossomo e, então, realiza-se a troca dos genes entre esses pontos. Para a mutação usada neste trabalho, foi escolhido $p = 2$.

Considere um indivíduo representado por uma sequência com seis números inteiros $S = [6 \ 1 \ 4 \ 3 \ 5 \ 2]$. Deste indivíduo, são escolhidas duas posições aleatórias, 2 e 5, por exemplo. O operador age trocando os genes dessas posições, criando-se, assim, um novo indivíduo $B = [6 \ 5 \ 4 \ 3 \ 1 \ 2]$.

SELEÇÃO

O operador de seleção é utilizado para selecionar os indivíduos que irão para a próxima geração. Neste trabalho, é utilizado o Método da Roleta. Este método é usado para selecionar X indivíduos para a próxima geração a partir de Y pais. Cada indivíduo recebe uma probabilidade de seleção, que pode ser calculada usando-se a seguinte fórmula:

$$Probabilidade_{\{seleção\}} = \frac{aptidão (indivíduo_i)}{\sum aptidão (indivíduo_i)}$$

Com o valor gerado pela função, é criada uma roleta, em que seus espaços são proporcionais à aptidão relativa de cada indivíduo. Assim, os indivíduos com alta aptidão terão maiores chances de participar do processo de criação da nova população.

4. APRESENTAÇÃO E ANÁLISE DOS RESULTADOS

O algoritmo SPEA2 foi desenvolvido utilizando-se MATLAB versão R2009b. Os testes foram feitos em um PC Intel Core 2 Duo 2.2 GHz, com 3 GB de memória RAM, em ambiente Windows 7.

Para realizar os testes computacionais, foram utilizados cinco dos 35 problemas-teste de (Hopper e Turton, 2002), gerados para problemas de corte guilhotinado. Este conjunto de problemas é dividido em sete categorias, sendo que cada categoria é subdividida em cinco problemas como mostra a Tabela 1. Neste trabalho, foram testadas cinco instâncias da categoria $T7$. Tal escolha se deve ao fato desta categoria apresentar um maior número de itens.

Tabela 1: Problemas-teste de Hopper e Turton (2002)

Guilhotinado	Nº de itens	Guilhotinado	Nº de itens
$T_1(a, b, c, d, e)$	17		
$T_2(a, b, c, d, e)$	25	$T_5(a, b, c, d, e)$	73
$T_3(a, b, c, d, e)$	29	$T_6(a, b, c, d, e)$	97
$T_4(a, b, c, d, e)$	49	$T_7(a, b, c, d, e)$	199

Para realizar os testes computacionais, o algoritmo proposto foi executado 3 vezes para cada problema-teste. Os parâmetros utilizados foram o número de indivíduos igual a 100, o número de gerações igual a 100, a probabilidade de mutação igual a 5% e a probabilidade do *Crossover* igual a 80%. Além disso, durante a fase de construção, foram utilizados três valores diferentes para k ($k = 2, k = 3, e k = 5$).

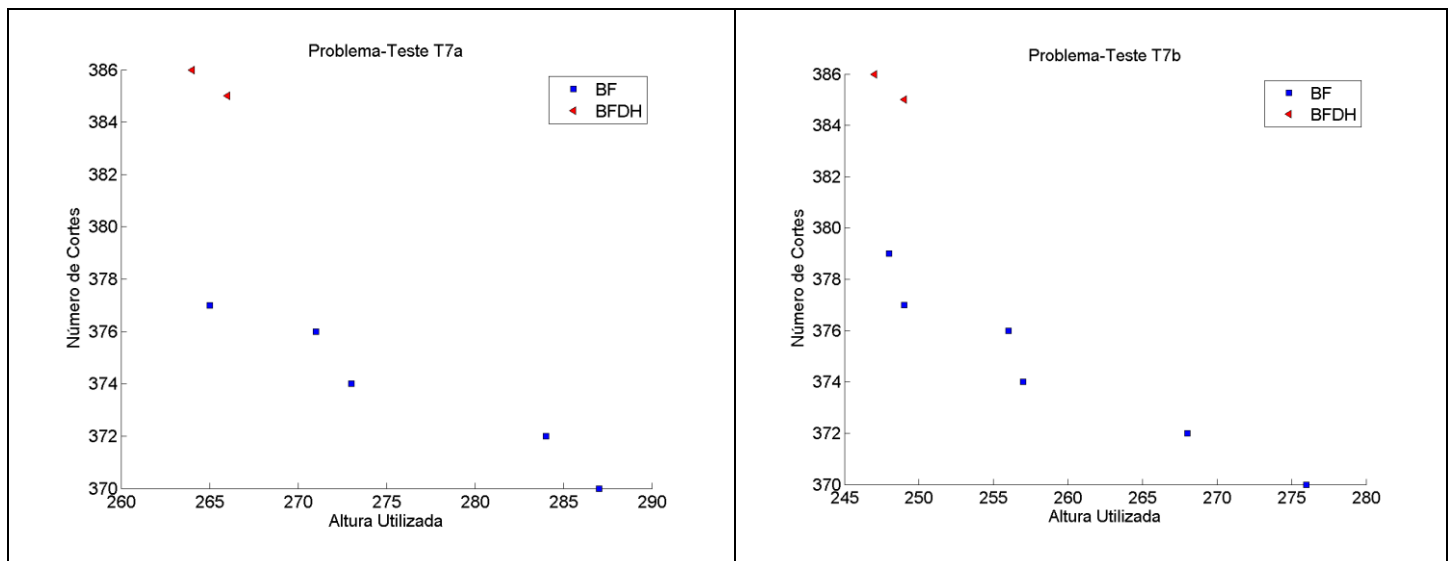


Figura 6: Conjunto de Pareto para os problemas T7a e T7b

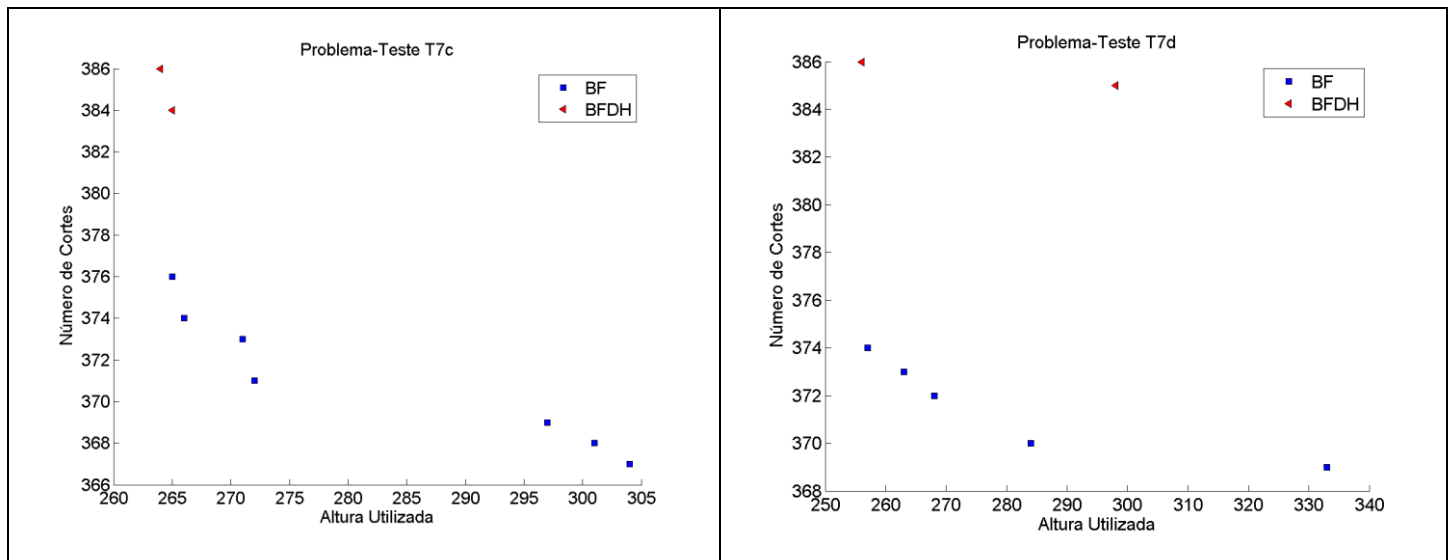


Figura 7: Conjunto de Pareto para os problemas T7c e T7d

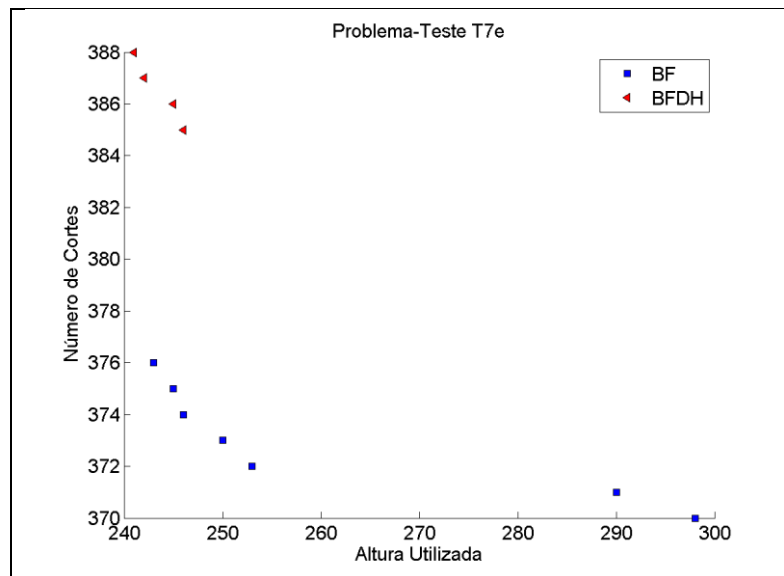


Figura 8: Conjunto de Pareto para o Problema T7e

Para as cinco instâncias consideradas ($T7_a$, $T7_b$, $T7_c$, $T7_d$ e $T7_e$), foram encontrados conjuntos de Pareto, que representam as soluções para as instâncias tratadas do problema. Estes resultados podem ser vistos na Figura 6 (para os problemas $T7_a$ e $T7_b$), na Figura 7 (para os problemas $T7_c$ e $T7_d$) e na Figura 8 (para o problema $T7_e$). Observa-se que o número de soluções obtidas no conjunto de Pareto pelo SPEA 2 com a técnica BF (por simplicidade, SPEA2-BF) é maior para todas as instâncias que o obtido pelo SPEA 2 com a técnica BFDH (para simplificar, SPEA2-BFDH). Para quatro dos problemas, o conjunto de Pareto obtido pelo SPEA2-BFDH possui apenas duas soluções, e, para o problema $T7_e$ o conjunto de Pareto obtido por este algoritmo possui quatro soluções. Por outro lado, o algoritmo SPEA2-BF retorna uma curva de Pareto variando de 5 a 7 pontos para as instâncias testadas. Assim, o algoritmo SPEA2-BF apresenta uma maior diversidade de soluções geradas possibilitando a realização de melhores análises a respeito dos resultados encontrados. Deve-se ressaltar que, em ambos os casos evidencia-se o caráter multiobjetivo da formulação apresentada para o PCDA, uma vez que, conforme as figuras 6, 7 e 8, a uma diminuição da

altura utilizada corresponde um aumento do número de cortes realizados, evidenciando-se, assim, a relação de compromisso entre as funções objetivo tratadas no problema.

É importante salientar que, apesar das soluções encontradas serem distintas, considerando-se as duas técnicas de encaixe, as soluções obtidas pelo SPEA2-BF dominam quase a totalidade das outras soluções obtidas pelo SPEA 2-BFDH.

Uma vez que não existem outros trabalhos publicados nos quais o problema em questão é resolvido, com as mesmas restrições aqui abordadas, não é possível fazer uma comparação de resultados.

O tempo médio de processamento para o SPEA2-BF foi de 303,90 segundos e, para o SPEA2-BFDH, foi de 381,73 segundos.

5. CONCLUSÃO

Neste trabalho, o Problema de Corte com Dimensão Aberta guilhotinado e sem rotação dos itens, foi tratado, utilizando-se uma abordagem multiobjetivo. O problema proposto consiste em minimizar a altura do objeto e minimizar o número de cortes realizados. O problema foi solucionado utilizando-se uma aplicação do algoritmo genético multiobjetivo SPEA2 combinado com as heurísticas de encaixe *Best-Fit* e *Best-Fit Decreasing Height*. Parte da população inicial dos algoritmos foi obtida através utilização da fase de construção do Algoritmo GRASP (*Greedy Randomized Adaptive Search Procedure*).

A metodologia proposta foi testada em cinco dos 35 problemas-teste de (HOPPER e TURTON, 2002), gerados para problemas de corte guilhotinado. Os problemas escolhidos pertencem à categoria *T7* e tal escolha se deve ao fato desta categoria apresentar um maior número de itens. Em todos os cinco problemas, o algoritmo multiobjetivo SPEA2-BF foi capaz de gerar um conjunto de Pareto representativo com boas soluções para o problema. O algoritmo multiobjetivo SPEA2-BFDH, apesar de encontrar soluções de boa qualidade, não apresenta um conjunto de Pareto representativo, gerando uma pequena quantidade de soluções para a escolha.

Uma vez que não existem trabalhos disponíveis na literatura que tratam do problema abordado, não é possível efetuar uma comparação com os resultados obtidos. Entretanto, por se tratar de um trabalho inicial, novos testes estão sendo realizados com outras instâncias do problema.

AGRADECIMENTOS

Os autores agradecem ao CAPES, CEFET-MG, FAPEMIG e CNPq pelo apoio ao desenvolvimento deste trabalho.

REFERÊNCIAS

- ANDRADE, M. S. F. de. **Algoritmos Evolutivos Mono e Multiobjetivo para Problemas Bidimensionais de Corte**. 2009. 105 f. Dissertação (Mestrado) - Curso de Mestrado em Modelagem Matemática e Computacional, Departamento de Pós-Graduação, Centro Federal de Educação Tecnológica de Minas Gerais, Belo Horizonte, 2009.
- ARMAS, Jesica de et al. Optimisation of a Multi-Objective Two-Dimensional Strip Packing Problem based on Evolutionary Algorithms. **International Journal of Production Research**, v.48, p. 2011-2028. 2009.
- FEO, Thomas A.; RESENDE, Maurício G. C.. Greedy Randomized Adaptive Search Procedures. **Journal of Global Optimization**, v.9, p. 109-133. 1995. Disponível em: <<http://dx.doi.org/10.1007/BF01096763>>.
- HIFI, Mhand. Exact algorithms for the guillotine strip cutting/packing problem. **Computers**

& Operations Research, v.25, p. 925-940. 11 nov. 1998.

HOLLAND, J. H. **Adaptation in natural and artificial systems**. Ann Arbor, MI: University of Michigan Press, 1975.

HOPPER, Eva; TURTON, B. C. H. Problem Generators for Rectangular packing problems. **Studia Informatica Universalis**, p. 123-136. 2002.

ILLICH, S.; WHILE, L.; BARONE, L.. Multi-objective strip packing using an evolutionary algorithm. **IEEE Congress On Evolutionary Computation: CEC**, p. 4207-4214. 1 set. 2007

LIU, D.; TENG, H.. An improved BL-algorithm for genetic algorithm of the orthogonal packing rectangles. **European Journal Of Operational Research**, v.112,p. 413-420. 16 jan. 1999.

LODI, A.; MARTELLO, S.; VIGO, D.. Models and Bound for two-dimensional level packing problems. **Journal Of Combinatorial Optimization**, v.8, p. 383-379. 2004.

MIRANDA, G. et al. Hyper heuristic codification for the multi-objective 2D guillotine strip packing problem. **IEEE Congress On Evolutionary Computation: CEC**, Barcelona, p. 1-8. 27 set. 2010.

ORTMANN, Frank Gerald. **Heuristics for Offline Rectangular Packing Problems**. 2010. 280 f. Tese (Doutorado) - Departamento de Department Of Logistics, Stelbosch University, África do Sul, 2010.

SOUZA, M. J. F. **Inteligência Computacional Para Otimização: Notas de Aula**. Disponível em:

<<http://www.decom.ufop.br/prof/marccone/Disciplinas/InteligenciaComputacional/InteligenciaComputacional.pdf>>. Acesso em: 10 maio 2011.

TIWARI, S.; CHAKRABORTI, N.. Multi-objective optimization of a two-dimensional cutting problem using genetic algorithms. **Journal of Materials Processing Technology**, v.173, p. 384-393. 2006.

WASCHER, G.; HAUBNER, H.; SCHUMANN, H.. An improved typology of cutting and packing problems. **European Journal Of Operations Research**, v. 183, p. 1109-1130. 2007.

Zitzler, E.; Laumanns, M.; Thiele, L. SPEA2: Improving the strength pareto evolutionary algorithm for multiobjective optimization. **Evolutionary Methods for Design, Optimization and Control with Application to Industrial Problems (EUROGEN 2001)**, 2002.