

Recebido em: 03/04/09

Aprovado em: 14/05/10

GRASP: a influência da lista restrita de candidatos nas soluções iniciais para otimização do corte unidimensional

Euclides Vieira Neto (ISECENSA – RJ/Brasil) - euclides@censanet.com.br

André Soares Velasco (UENF – RJ/Brasil) - andresv@uenf.br

• R. Professor Reis, 78, Centro, CEP 28013-320, Campos dos Goytacazes-RJ, fone: (55) 22-9832-8443

Geraldo Galdino de Paula Júnior (UENF – RJ/Brasil) - galdino@uenf.br

Resumo

O artigo apresenta um estudo sobre a metaheurística GRASP, aplicada ao problema de corte unidimensional. O objetivo é avaliar a influência do parâmetro de aleatoriedade no processo de construção das soluções iniciais. Utilizou-se o algoritmo GRASP-1D nos testes computacionais, com instâncias geradas aleatoriamente. Os resultados apresentados demonstram que, quanto maior o tamanho da instância, mais guloso deverá ser o parâmetro de aleatoriedade e , também, que a procura de soluções com a variação deste parâmetro é mais importante do que utilizar um maior número de iterações na busca das soluções iniciais.

Palavras-chave: GRASP; Corte Unidimensional; Esquema de Corte.

Abstract

The article presents a study on the GRASP metaheuristics applied to the unidimensional cut problem. The objective is to evaluate the influence of the random parameter α in the process of constructing the initial solutions. The GRASP-1D algorithm was used in the computational experiment with random generated examples. The findings demonstrate that the larger the size of the example instance, the more greedy becomes the random parameter. It also demonstrates the need to look for solutions within variations of this parameter, as it is more important than using a larger number of iterations in the search for the initial solutions.

Keywords: GRASP; Unidimensional Cut; Cut Scheme.

1. INTRODUÇÃO

O corte é uma fase de grande importância dentro da maioria dos processos de fabricação, nos segmentos metal-mecânica, cartongem, construção civil, móveis, entre outros, pois esta fase encontra-se no início do processo e, com um bom planejamento dos cortes, consegue-se reduzir o nível de aparas (perdas de matéria-prima), bem como, em alguns casos, otimizar as próximas fases de produção.

O problema do corte, genericamente, consiste em otimizar o corte de unidades menores, a partir de unidades maiores. No problema de corte unidimensional, a única dimensão relevante para a solução, é o comprimento. A figura 1 mostra um esquema de corte de um objeto, com seção transversal constante, que foi cortado em itens, o que originou uma apara que será tratada de perda.

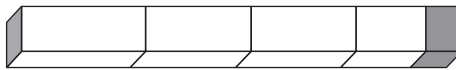


Figura 1 – Esquema de corte unidimensional.

A solução deste problema depende de uma otimização combinatória, que acarreta na intratabilidade do ponto de vista computacional. Esta desarmonia se deve ao grande número de arranjos de cortes possíveis; além disso, o desejo de enumerá-los, em busca da solução ótima, é inviável do ponto de vista prático. Algoritmos exatos, para solução dos problemas de corte, necessitam normalmente de um tempo computacional proibitivo, o que inibe sua aplicação em situações práticas, mesmo para problemas considerados pequenos. Levando em consideração o esforço computacional demandado para o processamento da solução, uma vez que o mercado é muito dinâmico, no que se refere às ofertas de matéria-prima e às alterações de projetos, o uso de heurísticas tem sido a opção preferida.

Entre outras metodologias aplicadas, com êxito, na resolução dos problemas de corte, o trabalho destaca a metaheurística GRASP (VELASCO, PAULA JUNIOR e VIEIRA NETO, 2008; LEITE, PAULA JUNIOR e VELASCO, 2008). Com o objetivo de encontrar boas soluções ou até mesmo a solução ótima, em tempos de execução condizentes com a realidade, este trabalho evidencia a busca do melhor parâmetro de aleatoriedade α , da fase de construção da GRASP, associando esta seleção a determinadas classes do problema de corte unidimensional.

2. O PROBLEMA DE CORTE UNIDIMENSIONAL

O problema de corte unidimensional pertence a uma noção mais geral, denominada Problemas de Corte e Empacotamento, em que Dyckhoff (1990) sugeriu a utilização de uma notação reduzida, para facilitar a identificação dos diversos casos. A classificação de Dyckhoff (1990) para o problema a que daremos ênfase neste artigo é 1/V/I/R, que define ser um problema da classe unidimensional, onde todas as unidades (demanda), de relativos poucos itens (comprimento da barra requisitada), devem ser cortadas de objetos (barras padrão do estoque) com o mesmo tamanho.

É importante definir alguns termos, de forma a não gerar dúvidas sobre os seus significados, no desenvolvimento deste trabalho.

- Barra padrão – barra estoque com comprimento comercial;
- Item – barra demanda com comprimento definido pelo projeto;
- Esquema de corte – arranjo de itens em uma barra padrão;
- Solução para um problema – conjunto de esquemas de corte.

A estrutura do problema de corte unidimensional enfatizado neste trabalho pode ser exemplificada como segue. A figura 2 ilustra um problema de corte unidimensional, onde se tem um número ilimitado de barras padrão em estoque, de comprimento 100 e o objetivo é cortar estas em várias barras menores, com quantidades pré-fixadas de comprimento, que variam de 5 a 46. Estes dois grupos, o de barras em estoque e os de barras demanda, são os dados básicos do problema de corte. Uma solução para o problema busca fornecer esquemas de corte, que venham a atender uma demanda de itens e apresentem perda mínima de material durante o processo.

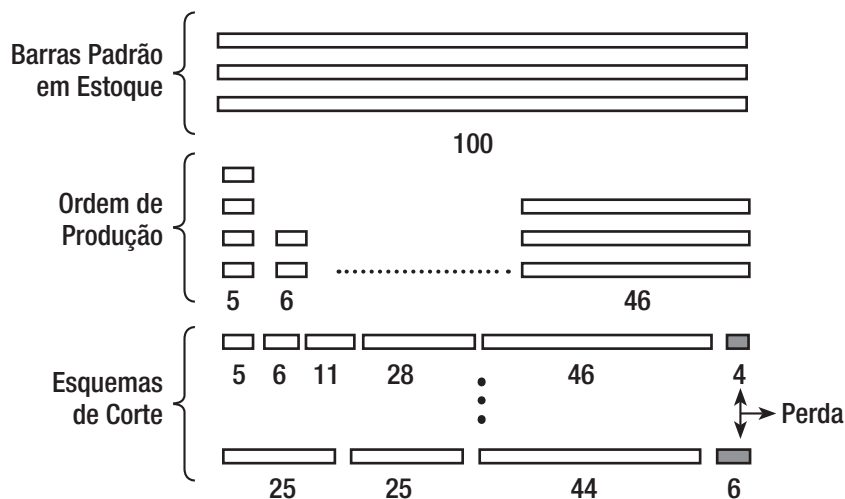


Figura 2 – Exemplo de um problema do corte unidimensional.

2.1. Formulação do problema de corte unidimensional

Com os esquemas de corte definidos e uma quantidade de barras padrão, de comprimento L , em estoque suficiente para atender a uma demanda de d_i itens, este problema de corte pode ser formulado como um problema de Programação Linear. Esta formulação consiste em determinar o número de vezes que cada um destes esquemas de corte será utilizado, de forma a satisfazer a uma demanda de itens pré-estabelecida, consumindo-se o menor número possível de barras estocadas. Sendo assim, a formulação matemática básica do problema de corte apresenta a seguinte estrutura:

$$\begin{aligned} \text{Min } Z &= \sum_{j=1}^n x_j \\ \text{s.a : } &\sum_{j=1}^n a_{ij} x_j = d_i, \quad i = 1, \dots, m \\ &x_j \geq 0 \text{ e Inteiro, } \quad j = 1, \dots, n \end{aligned} \quad (1)$$

Onde:

- x_j é a variável que representa o número de vezes que uma barra padrão é utilizada, segundo o esquema de corte j ;
- a_{ij} é a constante que indica número de itens do tipo i , gerados pelo esquema de corte j ;
- d_i é a constante que informa o número de itens do tipo i que deve ser obtido;
- $[a_{1j}, \dots, a_{mj}]^t$ é a coluna da matriz das restrições, que corresponde ao esquema de corte j ;
- m é a constante que indica o número de itens distintos;
- n é a constante que indica o número total de padrões de corte definidos.

A condição que o modelo impõe às variáveis x_j de não assumir valores contínuos, torna o problema difícil de ser resolvido, até para problemas relativamente pequenos. Se a quantidade de itens diferentes, encontrados em problemas de médio e grande porte, promove um aumento considerável no número n de esquemas viáveis e para encontrar a solução ótima do problema é preciso conhecer todos os esquemas de corte; então, a utilização de métodos exatos para resolução deste problema se torna inviável nestas circunstâncias, visto que o número de colunas $[a_{1j}, \dots, a_{mj}]^t$ pode chegar facilmente à classe dos milhões.

3. GRASP (*GREEDY RANDOMIZED ADAPTIVE SEARCH PROCEDURE*)

A metodologia GRASP, desenvolvida por Feo e Resende (1989), foi idealizada como a combinação de uma heurística construtiva e um procedimento de busca local. Um pseudocódigo genérico GRASP será descrito na figura 3 (FEO e RESENDE, 1995).

```
Procedimento GRASP ( )  
1. Entrada de Dados ( );  
2. Para (critério de parada GRASP não satisfeito) faça  
3.     Construa Solução Gulosa Aleatória (solução);  
4.     Busca Local (solução, Viz(solução));  
5.     Atualiza Solução (solução, melhor solução encontrada);  
6. Fim-para;  
7. Retorna (melhor solução encontrada);  
Fim GRASP
```

Figura 3 – Pseudocódigo GRASP.

É importante destacar que para construir uma solução, em vez de considerar apenas um candidato disponível para a escolha, ela cria um conjunto de candidatos, dentre os quais apenas um será escolhido aleatoriamente; desta maneira, ela pode construir várias soluções diferentes e aplicar um procedimento de busca local, com o objetivo de melhorar cada uma destas soluções iniciais. O algoritmo GRASP é projetado em duas fases a cada iteração: a fase de construção, seguida da fase de melhoria. Cada uma destas fases será apresentada separadamente.

3.1. Procedimento de construção da solução

A fase de construção é responsável pela denominação do método, já que é nela que aparecem as características marcantes da GRASP: o método guloso (*greedy*), a aleatoriedade (*randomized*) e a adaptação da função gulosa (*adaptive*). Esta fase é iterativa, já que, a cada passo, é adicionado à solução um novo elemento, até que se tenha uma solução completa e válida. Cada rodada, dentro de uma iteração, é composta por três subfases:

- Construção do conjunto LRC (Lista Restrita de Candidatos), o qual contém um conjunto reduzido de elementos candidatos a pertencer à solução, definidos de acordo com uma função gulosa e mecanismos de restrição;
- Escolha aleatória do elemento no LRC e inclusão de elemento na solução;
- Adaptação da função gulosa, para os elementos ainda não pertencentes à solução.

O parâmetro α é responsável por tornar a solução construída mais ou menos aleatória, ou seja, controla o grau de miopia e aleatoriedade da fase de construção. Para $\alpha = 0$, são geradas soluções totalmente aleatórias e para $\alpha = 1$, são produzidas soluções totalmente gulosas. A aplicação deste parâmetro ao valor β , chamado de restrição de ordinalidade, determina quais serão os elementos de LRC.

A instrução $K := \text{Randon}(\text{LRC})$ indica que a escolha de um dos candidatos do LRC seja aleatória. A componente probabilística da GRASP é caracterizada pela escolha aleatória de um dos candidatos do LRC, mas não necessariamente os do topo da lista. Essa técnica de escolha permite diferentes soluções a serem obtidas a cada interação GRASP, reduzindo a probabilidade de ótimos locais.

Algumas vezes, a função gulosa é sensível ou é influenciada pelo conjunto solução. Daí, a necessidade de adaptar ou atualizar a função gulosa, a cada vez que o conjunto solução é atualizado. Isto é considerado no procedimento citado na instrução adaptar f_s .

É importante ressaltar que é indispensável um ajuste adequado de α , pois este parâmetro exerce influência direta no que diz respeito à qualidade das soluções encontradas, número de iterações utilizadas e vizinhança explorada.

3.2. Procedimento de melhoria da solução

Como a solução gerada pelo procedimento descrito anteriormente, não garante a otimalidade da solução, é quase sempre vantajoso acrescentar um algoritmo de busca local, na tentativa de melhorar cada solução construída. O sucesso desta fase está associado à qualidade da solução produzida pela fase anterior da GRASP, ou seja, a eficácia da fase de construção reduziria os passos para se alcançar uma solução melhor, diminuindo o tempo de processamento envolvido nesta etapa.

4. ALGORITMO GRASP-1D

A escolha do algoritmo guloso FFD (*First Fit Decreasing*), para o procedimento de construção das soluções iniciais do algoritmo GRASP-1D (VIEIRA NETO, 2004), deve-se ao fato de este ser um algoritmo simples, que apresenta resultados bastante satisfatórios na otimização de cortes unidimensionais, obtendo perdas médias, na ordem de 10% (ROCHA, 1997). O FFD, também, pode ser implementado facilmente, sendo sua complexidade $O(n \log n)$, onde n é o número total de barras a serem cortadas, o que leva a processamento extremamente rápido de uma solução.

Em seguida, será apresentada a relação das variáveis envolvidas no algoritmo GRASP-1D, bem como o seu pseudocódigo, na figura 4. Em relação às variáveis, têm-se:

- m – número de itens;
- L – comprimento da barra padrão do estoque;
- l_i – comprimento do item i ;
- d_i – demanda do item i ;
- α – parâmetro de aleatoriedade;
- Solução – esquemas de corte;
- Perda – comprimento não utilizado da barra-padrão;
- BarrasUsadas – barras padrão utilizadas.

O pseudocódigo GRASP-1D é descrito como segue.

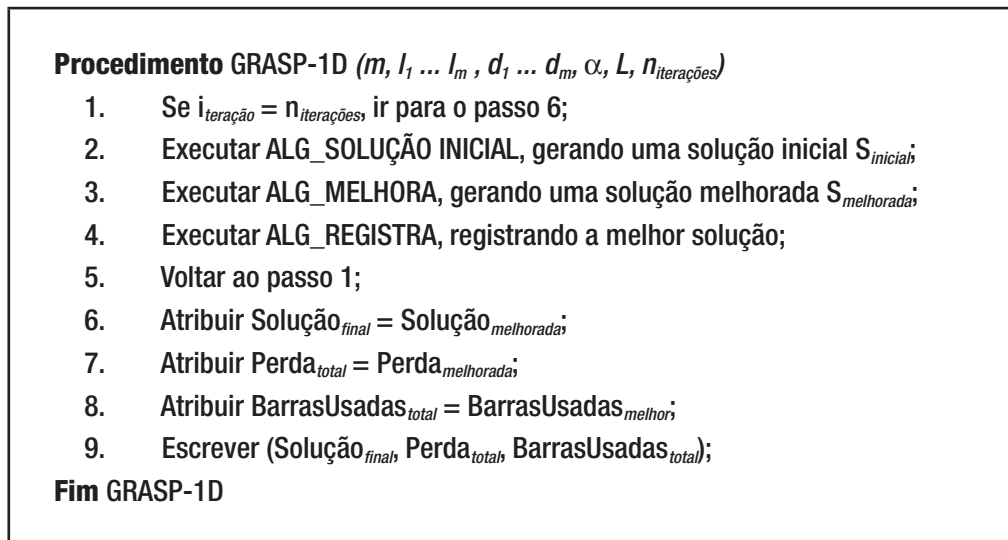


Figura 4 – Pseudocódigo GRASP-1D.

4.1. Procedimento para construir soluções iniciais

O Procedimento para Construir Soluções Iniciais, que será tratado por ALG_SOLUÇÃO INICIAL, é executado a cada iteração do GRASP-1D, que, por sua característica na geração da solução, possibilita criar soluções diferentes e independentes, o que torna esta fase muito importante e interessante para a solução final de um problema do corte unidimensional.

4.2. Parâmetro α na construção da solução inicial

A função gulosa que define a qualidade dos elementos do LRC, no caso desta pesquisa, o maior comprimento, e o parâmetro α que determina a cardinalidade desta lista, são os principais fatores que influenciam no desempenho do método e que podem ser considerados parâmetros para o estudo da metaheurística GRASP.

Como o critério guloso do FFD apresenta bom desempenho nos problemas do corte unidimensional (CINTRA, 1998), torna-se oportuno o estudo do parâmetro α que define a cardinalidade da LRC. Em relação a este parâmetro, têm-se:

- para $\alpha = 1$ tem-se $|LRC| = 1$, construindo uma solução inicial totalmente gulosa;
- para $\alpha = 0$ tem-se $|LRC| = m$, onde m é o número de itens a serem cortados, construindo uma solução totalmente aleatória.

Pela característica probabilística da metaheurística GRASP e pelas particularidades dos problemas de corte, onde não existe um método ideal para resolver todos os tipos de instâncias, procurou-se, então, através de uma pesquisa experimental, avaliar a influência da cardinalidade do LRC, nas soluções de problemas do corte unidimensional.

5. RESULTADOS COMPUTACIONAIS

Foram executados testes em 360 instâncias aleatórias. As instâncias aleatórias foram geradas, utilizando o gerador de instâncias CUTGEN1 (GAU e WASCHER, 1995), usando parâmetros que as aproximassem dos problemas reais. Estas instâncias foram divididas em três categorias. A categoria de instâncias com itens variados, onde l_i (comprimento dos itens) varia de 5% a 70% de L_i (comprimento da barra padrão em estoque); instâncias com itens médios, onde l_i varia de 20% a 50% de L_i e instâncias com itens pequenos, onde l_i varia de 3% a 30% de L_i .

As três categorias foram divididas em subcategorias, onde foram combinados: tamanho do problema m (número de itens de tamanho diferentes a serem cortados) e d_m (demanda média por instância), gerando então 36 subcategorias. Nos testes realizados, m recebeu os valores de 10, 20, 30 e 50. De acordo com Gau e Wäscher (1995) e confirmado por observações dentro dos processos produtivos, quase que a grande maioria das instâncias, com aplicações práticas, possuem tamanho menor que 50, o que justifica limitar o tamanho das instâncias em 50. Para d_m , foram atribuídos os valores 10, 50 e 100. Para cada subcategoria, foram geradas 10 instâncias, totalizando 360 instâncias.

O comprimento da barra padrão adotada foi de 6000 mm, pois é a medida comercial da maioria das barras, tubos e perfis que se encontram no mercado.

A figura 5 mostra como foram agrupadas as instâncias aleatórias, de forma a facilitar o entendimento da sua codificação.

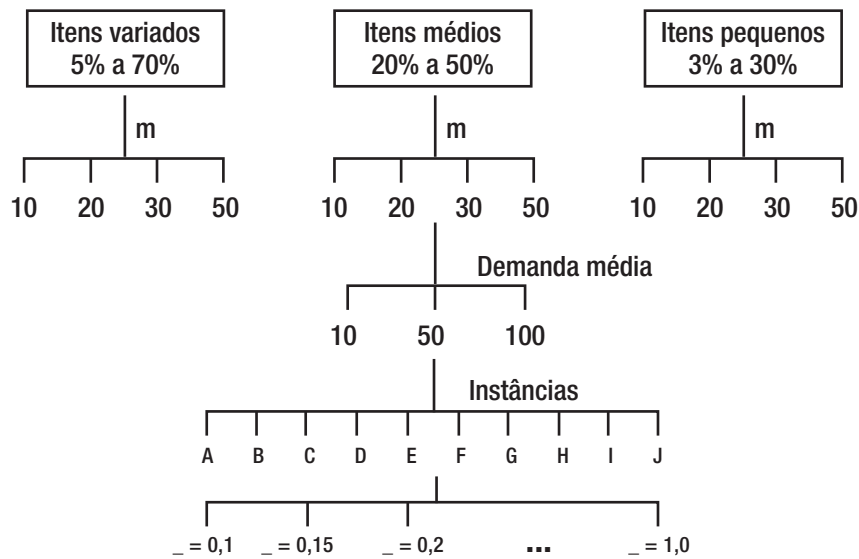


Figura 5 – Agrupamento para codificação das instâncias geradas aleatoriamente.

A codificação das instâncias foi feita da seguinte forma:

- o primeiro dígito é uma letra que indica a instância dentro da subcategoria (A,B, ..., J);
- o segundo e terceiro dígitos indicam o tamanho do problema (10, 20, 30 ou 50);
- do quarto ao sétimo dígitos, indicam-se as categorias (itens variados, itens médios ou itens pequenos);
- os dígitos restantes indicam a demanda média (10, 50 ou 100).

Como exemplo, o código F20033050 deve ser interpretado como a quinta instância das dez geradas, com 20 itens diferentes, com peças pequenas com comprimentos que variam de 3% a 30% do comprimento da barra padrão e têm demanda média dos itens de 50 peças.

Cada instância foi resolvida, utilizando o procedimento de construção das soluções iniciais do GRASP-1D, variando o parâmetro α , de 0,1 a 1, com um incremento de 0,05; resultando, então, em 19 situações diferentes, partindo do $\alpha = 0,1$ praticamente aleatório ao $\alpha = 1$ totalmente guloso (FFD). Em cada instância, foi registrado o menor percentual de perda das 1000 soluções geradas (número de iterações) para cada valor do parâmetro α e, finalmente, das 19000 soluções geradas, é registrada a melhor solução de todas.

A característica combinatoria do problema de corte unidimensional, em função do arranjo dos itens na construção dos esquemas de corte, torna difícil obter um método ideal para todos os problemas. Esta consideração pode ser observada na figura 6, que mostra o comportamento das soluções, segundo a variação do α para instâncias da mesma subcategoria (100330100).

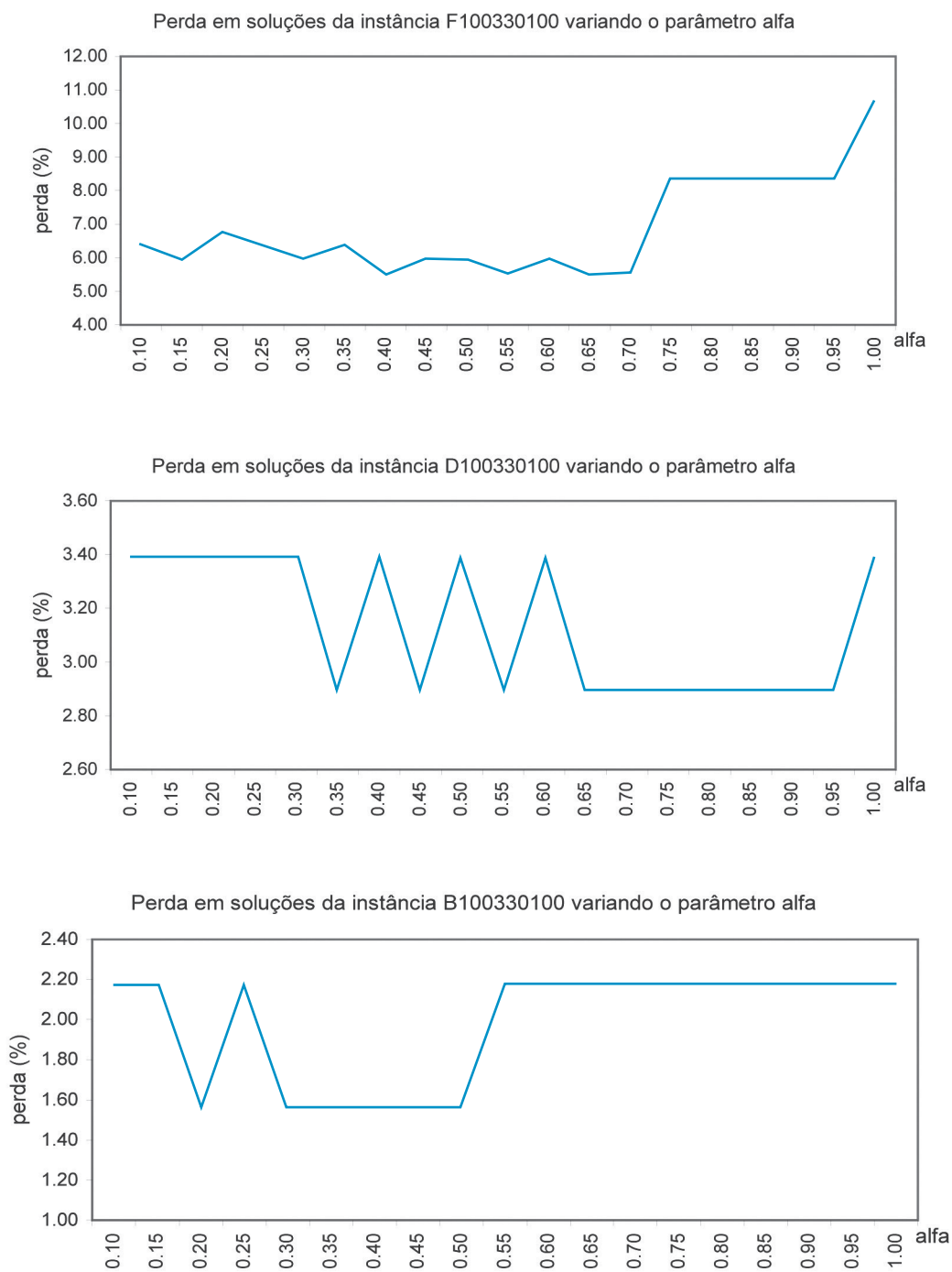


Figura 6 – Desempenho da construção de soluções, usando o GRASP-1D, com a variação do parâmetro α para instâncias com as mesmas características.

A partir da frequência de ocorrência da menor perda para cada valor do parâmetro α das instâncias agrupadas, segundo a combinação dos valores de m e d , construíram-se os histogramas apresentados nas figuras 7 e 8.

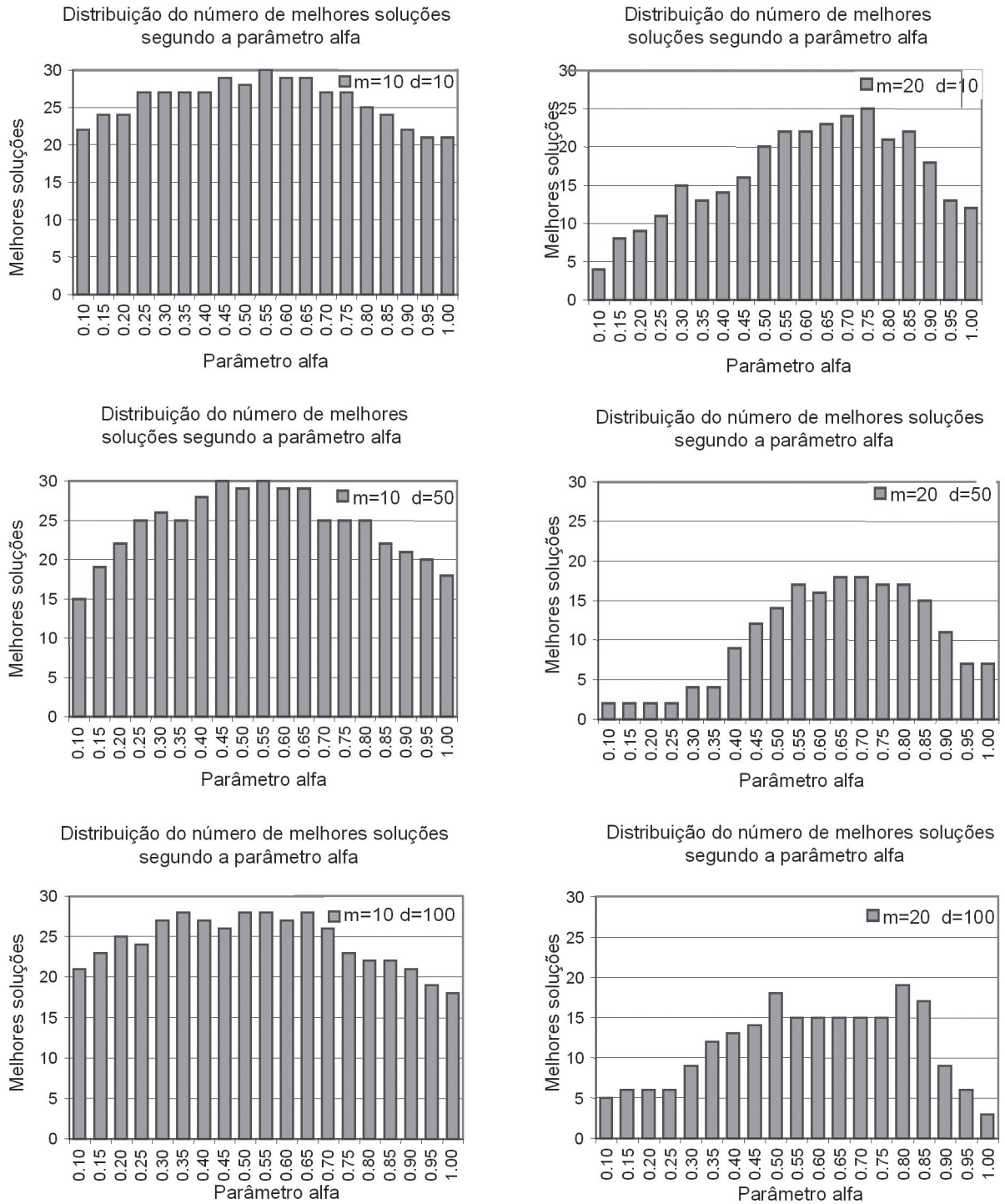


Figura 7 – Instâncias com $m = 10$ e $m = 20$.

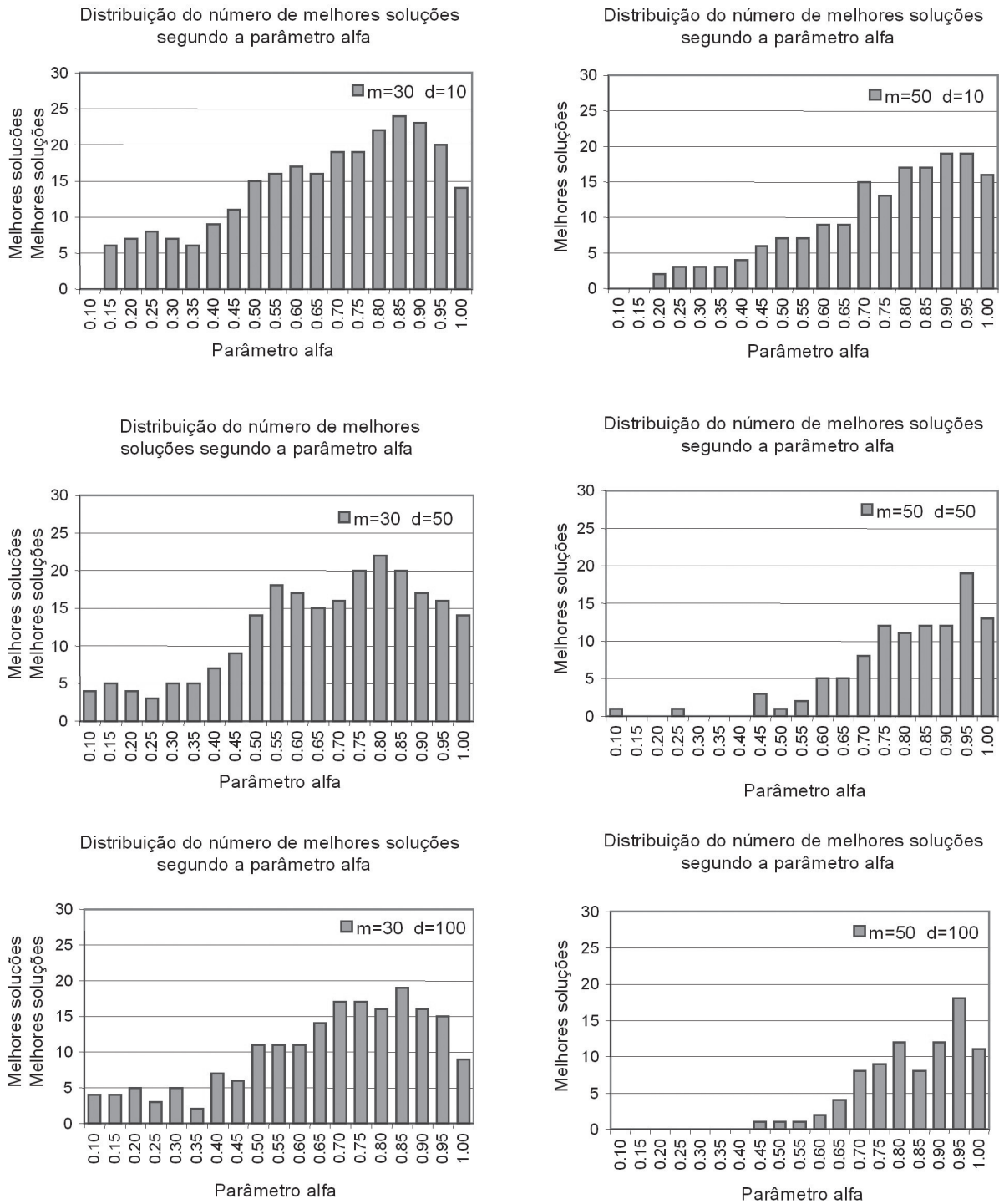


Figura 8 – Instâncias com $m = 30$ e $m = 50$.

Tabela 1 – Percentual de melhores soluções com variação dos valores do parâmetro α segundo as subcategorias das Instâncias.

Variação itens	m	demanda média	Parâmetro α																			
			0.10	0.15	0.20	0.25	0.30	0.35	0.40	0.45	0.50	0.55	0.60	0.65	0.70	0.75	0.80	0.85	0.90	0.95	1.00	
5%-70%	10	10	40	60	60	80	70	80	80	90	90	100	90	80	70	70	60	60	50	50	50	
		50	10	30	30	60	60	70	90	100	90	100	90	90	70	70	70	40	40	40	30	
		100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	
	20	10	0	0	0	0	10	20	20	30	20	50	60	60	80	90	60	50	30	0	0	
		50	0	0	0	0	20	10	20	30	30	30	40	50	60	70	50	40	10	0	0	
		100	0	0	0	0	10	10	20	30	40	30	30	30	30	30	50	60	80	90	80	70
	30	10	0	0	0	0	0	0	0	0	30	30	30	30	50	60	80	90	80	70	40	
		50	0	0	0	0	0	0	0	10	30	30	40	30	40	70	80	70	40	60	50	
		100	0	0	0	0	0	0	0	0	30	30	30	30	50	70	60	50	50	50	40	
	50	10	0	0	0	0	0	0	0	0	0	0	0	10	50	40	50	40	50	50	30	
		50	0	0	0	0	0	0	0	0	0	0	0	10	40	40	40	30	30	50	20	
		100	0	0	0	0	0	0	0	0	0	0	10	20	40	30	40	30	30	60	20	
20%-50%	10	10	100	100	90	90	100	90	90	100	90	100	100	100	100	100	100	90	80	70	70	
		50	80	90	100	90	100	90	90	100	100	100	100	100	100	100	90	90	90	80	80	
		100	90	90	90	80	90	90	90	80	100	100	100	100	90	80	80	80	80	70	70	
	20	10	40	50	40	40	60	30	50	40	90	90	70	90	70	70	60	70	60	40	30	
		50	20	20	20	20	10	10	30	50	70	70	60	70	60	40	60	40	30	10	10	
		100	50	60	60	50	60	60	70	70	80	70	80	70	80	70	80	70	40	20	10	
	30	10	0	20	20	30	20	10	30	40	60	50	70	60	60	40	50	60	50	40	30	
		50	40	50	30	30	40	30	30	40	70	80	70	70	50	70	80	60	60	40	30	
		100	40	30	40	30	50	20	40	50	70	60	70	70	80	70	80	60	60	50	20	
	50	10	0	0	0	0	0	0	0	10	20	30	20	40	30	30	20	40	50	50	40	
		50	10	0	0	10	0	0	0	30	10	20	40	30	40	60	40	50	50	60	40	
		100	0	0	0	0	0	0	0	10	10	0	10	20	30	30	70	50	60	40	20	
3%-30%	10	10	80	80	90	100	100	100	100	100	100	100	100	100	100	100	100	100	100	90	90	
		50	60	80	90	100	100	90	100	100	100	100	100	100	100	90	90	90	80	80	80	
		100	20	40	60	60	80	90	80	80	80	80	80	70	80	70	50	40	40	30	20	10
	20	10	0	30	50	80	90	80	70	90	100	90	90	90	90	90	90	90	90	90	90	
		50	0	0	0	0	10	20	40	40	40	70	60	60	60	60	60	70	70	60	60	
		100	0	0	0	10	20	50	40	40	70	40	50	40	50	50	60	50	40	30	20	
	30	10	0	40	50	50	50	50	60	70	60	80	70	70	80	90	90	90	100	90	70	
		50	0	0	10	0	10	20	40	40	40	70	60	50	70	60	60	70	70	60	60	
		100	0	0	10	0	0	0	20	20	20	30	20	40	40	30	20	40	30	50	30	
	50	10	0	0	20	30	30	30	30	40	40	50	50	60	70	70	80	80	80	90	90	
		50	0	0	0	0	0	0	0	0	0	0	0	10	10	0	20	30	40	40	80	70
		100	0	0	0	0	0	0	0	0	0	0	0	10	10	0	30	10	0	30	80	70

Pode-se observar que quanto menor o número de item dos problemas, maior a frequência de melhores resultados para cada α e a tendência de uma distribuição simétrica. Com o crescimento do número de itens, a distribuição vai definindo uma assimetria negativa, que se acentua de acordo com os acréscimos nos valores de α . Com relação ao valor das frequências, quanto maior o número de itens e o maior valor da demanda média das instâncias, menor a frequência das melhores soluções para cada parâmetro α .

Na tabela 1, são agrupados os percentuais de vezes que o determinado parâmetro α encontrou o menor valor da perda total das dez instâncias de cada subcategoria. Por exemplo, na casa definida pelo cruzamento da primeira linha desta tabela, com a sua primeira coluna, observa-se o valor 40 e este indica que, em 40% das instâncias testadas, com $m = 10$ e $d = 10$, foi alcançada a menor perda total para aquela instância.

É relevante destacar que, conforme os problemas vão crescendo em relação ao número de itens diferentes, a concentração de melhores resultados tende para o α mais guloso. Nesta tabela, confere-se que para os problemas, quando $m = 20$, a maior concentração de melhores soluções ficou para $\alpha \geq 0,5$; quando $m = 30$, a maior concentração de melhores soluções ficou para $\alpha \geq 0,6$ e quando $m = 50$ a maior concentração de melhores soluções, ficou para $\alpha \geq 0,8$. Isto confirma que quanto maior o número de itens tiver o problema, mais importante é a função gulosa para a construção das soluções iniciais.

6. CONCLUSÕES

Analisando o comportamento do parâmetro α , com relação à qualidade das soluções, observou-se que o valor ideal deste parâmetro está mais associado ao tamanho do problema do que à demanda dos itens.

Buscando identificar o melhor parâmetro α para problemas médios e grandes ($m > 20$), pode-se generalizar, em função dos experimentos mostrados neste estudo, que não seria uma má escolha resolverem as instâncias variando o α de 0,75 a 0,95, pois observou-se que quanto maior o tamanho do problema, as melhores soluções foram alcançadas, quando utilizado o critério tendendo ao mais guloso na construção da LRC. Já para problemas pequenos, seria indicado resolvê-los variando o α de 0,4 a 0,75.

Sendo assim, conclui-se que, para a busca da melhor solução final, deve-se utilizar o algoritmo GRASP-1D, com uma certa variação do parâmetro de aleatoriedade, que pode ser definida em função do tamanho do problema. Ao relacionar estes ajustes às diferentes características de cada solução inicial gerada, melhores serão os resultados das combinações dos itens dos esquemas selecionados no procedimento de melhoria.

7. REFERÊNCIAS BIBLIOGRÁFICAS

- CINTRA, G. F. **Algoritmos Híbridos para Problemas de Corte Unidimensional**. Dissertação (Mestrado em Matemática Aplicada) – Instituto de Matemática e Estatística da Universidade de São Paulo – IME – USP, SP, 1998.
- DYCKHOFF, H. A Typology of Cutting and Packing Problems. **European Journal of Operation Research**, v.44, pp.145–159, 1990.
- FEO, T. A.; RESENDE, M. G. C. A Probabilistic Heuristic for a Computationally Dificult Set Covering Problem. **Operation Research Letters**, v.8, pp. 67–71, 1989.
- FEO, T. A.; RESENDE, M. G. C. Greedy Randomized Adaptative Search Procedures. **Journal of Global Optimization**, v.6, pp.109–133, 1995.
- GAU, T.; WASCHER, G. A Problem Generator for the Standard One-Dimensional Cutting Stock Problem. **European Journal of Operation Research**, v.84, pp.572–579, 1995.
- LEITE, E. C. B. R.; PAULA JUNIOR, G. G.; VELASCO, A. S. Empacotamento Tridimensional: Aplicação da Grasp ao Problema de Carregamento de Contêiner. **Anais do XV SIMPEP - Simpósio de Engenharia de Produção**. Bauru, 2008.
- ROCHA, M. N. **Otimização de Cortes de Barras**. Dissertação (Mestrado em Ciências da Computação), Universidade Federal de Minas Gerais – UFMG, MG, 1997.
- VELASCO, A. S.; PAULA JUNIOR, G.G.; VIEIRA NETO, E. Um Algoritmo Heurístico Baseado na GRASP para o Problema de Corte Bidimensional Guilhotinado e Restrito. **Revista GEPROS. Gestão da Produção, Operações e Sistemas**, ed.1, p.129–141, 2008.
- VIEIRA NETO, E. **GRASP: Efeito Da Interdependência das Soluções Iniciais na Otimização do Corte Unidimensional**. Tese (Doutorado em Ciências da Engenharia) – Centro de Ciências e Tecnologia, Laboratório de Engenharia de Produção – Leprod, UENF, Campos dos Goytacazes–RJ, 2004.