

MÉTODOS EXATOS BASEADOS EM RELAXAÇÕES LAGRANGIANA E SURROGATE PARA O PROBLEMA DE CARREGAMENTO DE PALETES DO PRODUTOR

Lilian Kátia de Oliveira

Reinaldo Morabito *

Departamento de Engenharia de Produção
Universidade Federal de São Carlos (UFSCar)
São Carlos – SP

lilian.oliveira@fsc.br

morabito@power.ufscar.br

* *Corresponding author* / autor para quem as correspondências devem ser encaminhadas

Recebido em 07/2005; aceito em 03/2006

Received July 2005; accepted March 2006

Resumo

Neste artigo apresentamos métodos exatos, baseados em relaxações Lagrangiana e *surrogate*, com bom desempenho para resolver o problema de carregamento de paletes do produtor. Este problema consiste em arranjar ortogonalmente e sem sobreposição o máximo número de retângulos de dimensões (l, w) ou (w, l) sobre um retângulo maior (L, W) . Os métodos propostos são procedimentos de busca em árvore do tipo *branch and bound* que, em cada nó, utilizam limitantes derivados de relaxações Lagrangiana e/ou *surrogate* de uma formulação de programação linear 0–1. Algoritmos de otimização do subgradiente são usados para otimizar estes limitantes. São aplicados ainda testes de redução do problema e heurísticas Lagrangiana e *surrogate* para se obter boas soluções factíveis na otimização do subgradiente. Testes computacionais foram realizados utilizando exemplos da literatura e exemplos reais de uma transportadora. Os resultados mostram que um dos métodos propostos é competitivo com outros métodos da literatura, incluindo o *software* GAMS/CPLEX.

Palavras-chave: problema de carregamento de paletes do produtor; relaxação Lagrangiana e/ou *surrogate*; método *branch and bound*; otimização do subgradiente.

Abstract

In this paper we present exact methods, based on Lagrangean and surrogate relaxations, with good performance to solve the manufacturer's pallet loading problem. This problem consists of orthogonally arranging the maximum number of rectangles of sizes (l, w) or (w, l) into a larger rectangle (L, W) without overlapping. The proposed methods involve a branch and bound based tree search procedure for which the bounds, in each node of the tree, are derived from Lagrangean and/or surrogate relaxations of a 0–1 linear programming formulation. Subgradient optimization algorithms are used to optimize such bounds. Problem reduction tests and Lagrangean and surrogate heuristics are also applied to obtain good feasible solutions in the subgradient optimization. Computational experiments were performed with instances of the literature and actual instances of a carrier. The results show that one of the proposed methods is competitive with methods found in the literature, including the software GAMS/CPLEX.

Keywords: manufacturer's pallet loading problem; Lagrangean and/or surrogate relaxation; branch and bound method; subgradient optimization.

1. Introdução

Este trabalho trata de um caso particular dos problemas de corte e empacotamento, denominado problema de carregamento de paletes do produtor (PCP do produtor), o qual é classificado como 2/B/O/C (bidimensional, seleção de itens, único objeto, itens iguais), de acordo com a tipologia de Dyckhoff (1990). Basicamente, este problema consiste em arranjar sem sobreposição o máximo número de retângulos de dimensões (l,w) ou (w,l) (faces inferiores das caixas) sobre um retângulo maior (L,W) (superfície do palete). Admitimos que as caixas estão disponíveis em grandes quantidades e devem ser arranjadas ortogonalmente, isto é, com seus lados paralelos aos lados do palete. Estes arranjos de caixas formam camadas que são então empilhadas sobre o palete. Devido à escala e extensão de certos sistemas logísticos, um pequeno aumento do número de produtos carregados sobre cada palete pode resultar em economias substanciais.

Discussões sobre as vantagens da paletização de carga podem ser encontradas, por exemplo, em Sobral (1996), Novamad Pallets (2004) e Interlogis (2004). O PCP do produtor é importante nas atividades logísticas de armazenagem e transporte de produtos embalados em caixas e carregados sobre paletes e caminhões (Morabito *et al.*, 2000). Ele pode ser aplicado para apoiar decisões tanto num nível mais estratégico (por exemplo, o projeto ou a escolha de paletes mais adequados, o desenvolvimento de embalagens para produtos), quanto num nível mais operacional (por exemplo, a geração de padrões de carregamento para armazenagem e distribuição). Além disso, o PCP do produtor é NP-difícil do ponto de vista de complexidade computacional, o que implica que provavelmente não deve existir um algoritmo polinomial capaz de resolvê-lo (Tarnowski *et al.*, 1994; Nelissen, 1994; Letchford & Amaral, 2001; Martins, 2002).

Poucos autores propuseram métodos exatos para resolver o PCP do produtor, entre eles, Dowsland (1987), Tsai *et al.* (1993) e Bhattacharya *et al.* (1998) (este último método nem sempre é válido, conforme mostrado em Martins, 2002). Dowsland (1987) apresentou um algoritmo que consiste em encontrar o máximo conjunto estável de um grafo, representando o carregamento do palete. O algoritmo funciona bem para problemas em que o número de caixas por camada não é grande, por exemplo, menor que 30, o que nem sempre corresponde aos casos práticos. Tsai *et al.* (1993) formularam o PCP do produtor como um modelo 0–1 com restrições disjuntivas, e sugeriram resolvê-lo por algoritmos do tipo *branch and bound* explorando a estrutura particular do modelo. Porém, o número destas restrições cresce exponencialmente com o número de caixas disponíveis. Nenhum destes trabalhos apresentou extensivos experimentos computacionais com seus métodos, o que dificulta a comparação com outros métodos.

Devido à dificuldade com os métodos exatos, diversas heurísticas têm sido propostas na literatura para resolver o PCP do produtor. Exemplos podem ser encontrados em Steudel (1979), Smith & De Cani (1980), Bischoff & Dowsland (1982), Dowsland (1993), Nelissen (1994, 1995), Arenales & Morabito (1995), Scheithauer & Terno (1996), Herbert & Dowsland (1996), Scheithauer & Sommerweiss (1998), Morabito & Morales (1998, 1999), Farago & Morabito (2000), G & Kang (2001), Martins (2002), Lins *et al.* (2003), Pureza & Morabito (2003, 2006), Alvarez-Valdes *et al.* (2005a) e Birgin *et al.* (2005). Lins *et al.* (2003) apresentaram uma abordagem explorando partições na forma de um L, e conjecturaram que ela é exata para o PCP do produtor. A desvantagem desta abordagem é o relativamente grande requisito de tempo computacional, o que foi atenuado no estudo recente em Birgin *et al.* (2005). Limitantes superiores para o PCP do produtor foram

estudados em Nelissen (1995), Letchford & Amaral (2001), Morabito & Farago (2002) e Martins (2002).

Neste trabalho, modelamos o PCP do produtor como um problema inteiro 0–1, similarmente a Farago & Morabito (2000). O modelo pode ser visto como um caso particular do modelo proposto em Beasley (1985) para o problema mais geral de corte 2/B/O/M (bidimensional, seleção de itens, único objeto, vários itens diferentes). Apresentamos métodos exatos baseados na aplicação de relaxações Lagrangiana e/ou *surrogate* e, para otimizar os limitantes superiores, utilizamos procedimentos de otimização do subgradiente. Testes de redução do problema e heurísticas Lagrangiana e *surrogate* são aplicados na otimização do subgradiente para otimizar os limitantes inferiores (soluções factíveis). Um procedimento enumerativo de busca em árvore é usado para a obtenção da solução ótima do problema. Heurísticas construtivas também são exploradas para melhorar os limitantes inferiores, além do uso de outros limitantes superiores disponíveis na literatura para o PCP do produtor. Os resultados computacionais foram realizados utilizando exemplos da literatura e exemplos reais de uma empresa transportadora.

É importante observar que o trabalho de Beasley (1985) analisa somente o uso de relaxação Lagrangiana combinada com relaxação *surrogate* (LAGSUR), e não reporta resultados utilizando apenas relaxação Lagrangiana (LAG) ou apenas relaxação *surrogate* (SUR). Além disso, o trabalho trata do problema de corte mais geral, os limitantes obtidos para tal problema não são apertados e os resultados mostram que a abordagem produz bons resultados apenas para problemas de tamanhos moderados (similarmente para o trabalho de Hadjiconstantinou & Christofides, 1995). O trabalho de Farago & Morabito (2000) aplica LAG e LAGSUR no PCP do produtor e utiliza uma heurística Lagrangiana para encontrar boas soluções factíveis na otimização do subgradiente. No entanto, não incorpora a otimização do subgradiente em uma árvore de busca e por isso o método não tem garantia de otimalidade. O presente trabalho pode ser visto como uma extensão do trabalho de Farago & Morabito (2000). Nossa motivação é avaliar o desempenho de métodos exatos baseados em LAG, SUR e LAGSUR para tratar o PCP do produtor.

O artigo está organizado da seguinte maneira: na seção 2 apresentamos uma formulação do PCP do produtor adaptada de Beasley (1985), e na seção 3 discutimos limitantes superiores e apresentamos as aplicações de LAG, SUR e LAGSUR para o problema. Na seção 4 discutimos técnicas de redução do problema e apresentamos heurísticas Lagrangiana e *surrogate* para melhorar os limitantes inferiores. Na seção 5 detalhamos o procedimento de busca em árvore do tipo *branch and bound*. Na seção 6 os desempenhos dos métodos LAG, SUR e LAGSUR são analisados e comparados em diversos experimentos com exemplos da literatura e exemplos reais de uma transportadora. Os resultados mostram que o método LAG é competitivo com outros métodos da literatura, incluindo o *software* GAMS/CPLEX. Finalmente, a seção 7 apresenta as conclusões e perspectivas para pesquisa futura.

2. Formulação do Problema

O PCP do produtor pode ser modelado como um programa inteiro 0–1 (Farago & Morabito, 2000), o qual é um caso particular do modelo proposto em Beasley (1985) para o problema de corte bidimensional não-guilhotinado. Admitimos que L , W , l e w são números inteiros, e que $l \geq w$ e $l \leq \min\{L, W\}$. Por simplicidade, definimos $(l_1, w_1) = (l, w)$ e $(l_2, w_2) = (w, l)$, assim (l_i, w_i) , $i=1,2$, correspondem ao comprimento e largura da face de uma caixa com

orientação i . O problema consiste em encontrar um arranjo de caixas em camadas horizontais (padrão de carregamento bidimensional) tal que a utilização da superfície do palete seja a máxima possível.

Sejam P e Q o número mínimo e o máximo de caixas por camada, respectivamente, a serem colocadas sobre o palete ($0 \leq P \leq Q$). Adotamos um sistema de coordenadas cartesianas com origem no canto inferior esquerdo do palete. Sabemos que as caixas podem ser colocadas em várias posições ao longo do comprimento L e largura W do palete. Sejam $X = \{p \mid 0 \leq p \leq L - w, \text{ inteiro}\}$ e $Y = \{q \mid 0 \leq q \leq W - w, \text{ inteiro}\}$, os conjuntos das coordenadas (p, q) , respectivamente, para se colocar o canto inferior esquerdo da face de uma caixa dentro do palete. Sem perda de generalidade, podemos restringir os conjuntos X e Y para os conjuntos normais (Christofides & Whitlock, 1977):

$$X = \{p \mid p = \sum_{i=1}^2 l_i b_i, \quad 0 \leq p \leq L - w, \quad b_i \geq 0 \text{ e inteiro}, i = 1, 2\}$$

$$Y = \{q \mid q = \sum_{i=1}^2 w_i b_i, \quad 0 \leq q \leq W - w, \quad b_i \geq 0 \text{ e inteiro}, i = 1, 2\}.$$

Definimos a função a para descrever restrições de sobreposição de caixas sobre o palete, e que pode ser computada com antecedência para cada posição (p, q) , cada posição (r, s) e cada orientação i , com $p \in X$ tal que $p \leq L - l_i$, $q \in Y$ tal que $q \leq W - w_i$, $r \in X$, $s \in Y$ e $i = 1, 2$ (Figura 1):

$$a_{ipqrs} = \begin{cases} 1, & \text{se } 0 \leq p \leq r \leq p + l_i - 1 \leq L - 1 \text{ e } 0 \leq q \leq s \leq q + w_i - 1 \leq W - 1 \\ 0, & \text{caso contrário} \end{cases}$$

Note que $a_{ipqrs} = 1$ indica que a coordenada (r, s) fica ocupada, ao se colocar uma caixa com orientação i na coordenada (p, q) do palete. Para todo $p \in X$ tal que $p \leq L - l_i$, $q \in Y$ tal que $q \leq W - w_i$ e $i = 1, 2$, definimos a variável de decisão x_{ipq} :

$$x_{ipq} = \begin{cases} 1, & \text{se uma caixa é colocada na posição } (p, q) \text{ do palete com orientação } i \\ 0, & \text{caso contrário} \end{cases}$$

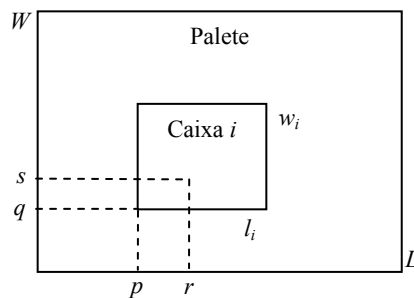


Figura 1 – Posição (r, s) ocupada em função da colocação de uma caixa na posição (p, q) com orientação i .

O PCP do produtor pode então ser formulado pelo seguinte problema linear 0–1 :

$$\text{Max} \quad \sum_{i=1}^2 \sum_{\{p \in X | p \leq L - l_i\}} \sum_{\{q \in Y | q \leq W - w_i\}} x_{ipq} \quad (1)$$

sujeito a:

$$\sum_{i=1}^2 \sum_{\{p \in X | p \leq L - l_i\}} \sum_{\{q \in Y | q \leq W - w_i\}} a_{ipqrs} x_{ipq} \leq 1, \text{ para todo } r \in X, s \in Y \quad (2)$$

$$P \leq \sum_{i=1}^2 \sum_{\{p \in X | p \leq L - l_i\}} \sum_{\{q \in Y | q \leq W - w_i\}} x_{ipq} \leq Q \quad (3)$$

$$x_{ipq} \in \{0, 1\}, \quad i = 1, 2, \quad p \in X \text{ tal que } p \leq L - l_i, \quad q \in Y \text{ tal que } q \leq W - w_i \quad (4)$$

A função objetivo (1) maximiza o número de caixas colocadas sobre o paleta. Note que a restrição (2) garante que cada posição (r,s) seja ocupada por no máximo uma caixa e, desta maneira, evita sobreposição no arranjo de caixas (Figura 1). A restrição (3) garante que o número de caixas arranjadas esteja dentro do intervalo $[P,Q]$. Em geral temos $P=0$ e Q é um limitante superior para o problema. Em certos casos a restrição (3) pode ser reescrita como:

$$P_i \leq \sum_{i=1}^2 \sum_{\{p \in X | p \leq L - l_i\}} \sum_{\{q \in Y | q \leq W - w_i\}} x_{ipq} \leq Q_i,$$

para todo i , para impor limitantes para o número de caixas arranjadas em cada orientação i . Isto pode ser útil para gerar padrões mais estáveis do ponto de vista da amarração da carga. O problema (1)-(4) envolve $O(|X||Y|)$ variáveis e restrições, o que o torna em geral difícil de ser resolvido otimamente nos casos práticos, dependendo do tamanho dos conjuntos X e Y . Estes conjuntos podem ser ainda mais reduzidos, usando o conceito de *raster points* (para mais detalhes, veja Scheithauer & Terno, 1996; Morabito & Morales, 1998):

$$X' = \{(L - p)_X \mid p \in X\} \cup \{0\}, \quad (p')_X = \max\{p \mid p \in X, p \geq p'\}$$

$$Y' = \{(W - q)_Y \mid q \in Y\} \cup \{0\}, \quad (q')_Y = \max\{q \mid q \in Y, q \geq q'\}.$$

Outras formulações para o PCP do produtor podem ser encontradas em Tsai *et al.* (1993), Hadjiconstantinou & Christofides (1995) e Martins (2002) (o modelo de Hadjiconstantinou & Christofides (1995) nem sempre é válido, conforme mostrado em Amaral & Letchford, 1999). No presente trabalho, a escolha do modelo (1)-(4) deveu-se às experiências anteriores da literatura com LAGSUR para o problema de corte mais geral (Beasley, 1985) e com LAG e LAGSUR para o PCP do produtor (Frago & Morabito, 2000), e o fato dos limitantes superiores obtidos com as relaxações linear, LAG e LAGSUR deste modelo para o PCP do produtor serem bem apertados (Letchford & Amaral, 2001; Morabito & Frago, 2002).

3. Limitantes Superiores e Relaxações Lagrangiana e *Surrogate*

Diversos limitantes superiores para o problema (1)-(4) são encontrados na literatura (Nelissen, 1995; Letchford & Amaral, 2001; Morabito & Frago, 2002; Martins, 2002). O mais simples e intuitivo é o limitante da razão da área, definido por $\lfloor LW/lw \rfloor$. Outros limitantes são o de Barnes (1979) e de exemplos equivalentes (usando a noção de partições

eficientes), dado por: $\lfloor L^*W^*/lw \rfloor$, onde $L^* = \max \{x \mid x = rl + sw \leq L, r, s \geq 0, \text{ inteiro}\}$ (similarmente para W^*). Limitantes mais apertados para o PCP do produtor são obtidos com a relaxação linear do problema (1)-(4) e as relaxações discutidas abaixo.

3.1 Relaxação Lagrangiana (LAG)

A relaxação Lagrangiana (LAG) tal como é conhecida hoje foi introduzida por Held & Karp (1970, 1971). Para mais detalhes de LAG, os leitores podem consultar, por exemplo, Nemhauser & Wolsey (1988) e Reeves (1993). Introduzindo os multiplicadores de Lagrange $\lambda_{rs} (\geq 0)$ para todo $r \in X$ e todo $s \in Y$ da expressão (2), obtemos o seguinte problema Lagrangiano do limitante superior:

$$\text{Max} \quad \sum_{i=1}^2 \sum_{\{p \in X \mid p \leq L - l_i\}} \sum_{\{q \in Y \mid q \leq W - w_i\}} V_{ipq} x_{ipq} + \sum_{r \in X} \sum_{s \in Y} \lambda_{rs} \quad (5)$$

sujeito a:

$$P \leq \sum_{i=1}^2 \sum_{\{p \in X \mid p \leq L - l_i\}} \sum_{\{q \in Y \mid q \leq W - w_i\}} x_{ipq} \leq Q \quad (6)$$

$$x_{ipq} \in \{0, 1\}, \quad i = 1, 2, \quad p \in X \text{ tal que } p \leq L - l_i, \quad q \in Y \text{ tal que } q \leq W - w_i \quad (7)$$

onde $V_{ipq} = 1 - \sum_{r \in X} \sum_{s \in Y} \lambda_{rs} a_{ipqrs}$. Sejam $\{X_{ipq}\}$ representando os valores das variáveis $\{x_{ipq}\}$ na solução do problema Lagrangiano (5)-(7), com valor dado por:

$$Z_{UB} = \sum_{i=1}^2 \sum_{\{p \in X \mid p \leq L - l_i\}} \sum_{\{q \in Y \mid q \leq W - w_i\}} V_{ipq} X_{ipq} + \sum_{r \in X} \sum_{s \in Y} \lambda_{rs} .$$

Note que Z_{UB} corresponde a um limitante superior para o valor ótimo do problema original (1)-(4), para quaisquer $\{\lambda_{rs}\}$ não negativos. Dados $\{\lambda_{rs}\}$, o problema (5)-(7) pode ser facilmente resolvido por inspeção. Seja n o número de variáveis x_{ipq} com valores V_{ipq} estritamente positivos:

- Se $n \geq Q$, então escolha as Q variáveis com maiores valores V_{ipq} , e fixe-as em 1.
- Se $P \leq n < Q$, então escolha as n variáveis com maiores valores V_{ipq} , e fixe-as em 1.
- Se $n < P$, então escolha as P variáveis com maiores valores V_{ipq} , e fixe-as em 1.

As demais variáveis são fixadas em 0.

O problema Lagrangiano (5)-(7) possui a propriedade de integralidade, uma vez que, dados $\{\lambda_{rs}\}$, sua solução não se altera ao trocarmos a restrição de integralidade $x_{ipq} \in \{0, 1\}$ por sua relaxação linear $0 \leq x_{ipq} \leq 1$ (note que todos os coeficientes de (6) são iguais a 1, e P e Q são números inteiros). Isto implica que o melhor limitante superior produzido pelo problema dual Lagrangiano:

$$\min_{\lambda_{rs} \geq 0} \{\text{problema Lagrangiano (5)-(7)}\}$$

para o problema (1)-(4) não é melhor do que o limitante superior obtido pela relaxação linear de (1)-(4). Para resolver o problema dual Lagrangiano é utilizado o método de otimização do subgradiente (Camerini *et al.*, 1975; Crowder, 1976; Beasley, 1985; Farago & Morabito, 2000), resumido a seguir (o algoritmo em pseudo-código está apresentado em Oliveira, 2004):

Passo 1: Gere uma solução factível, por exemplo, a solução homogênea com todas as caixas arranjadas sob a mesma orientação. Faça Z_{LB} igual ao valor desta solução homogênea. Seja $ITMAX$ o máximo número de iterações e faça $it = 0$. Fixe $\lambda_{rs} = 0$ para todo $s \in Y$, e todo $r \in X$, como valores iniciais para os multiplicadores.

Passo 2: Resolva o problema Lagrangiano (5)-(7) com o conjunto de multiplicadores atual, obtendo a solução $\{X_{ipq}\}$ e valor Z_{UB} . Faça $it = it + 1$.

Passo 3: Teste de factibilidade: se a solução Lagrangiana é factível para o problema original (1)-(4), então atualize Z_{LB} , o limitante inferior para o problema, correspondendo a uma solução factível. Atualize o menor limitante superior Z_{min} com Z_{UB} .

Passo 4: Teste de otimalidade: pare se $Z_{min} = Z_{LB}$; caso contrário, vá para o *Passo 5*.

Passo 5: Calcule os subgradientes:

$$G_{rs} = -1 + \sum_{i=1}^2 \sum_{\{p \in X | p \leq L-l_i\}} \sum_{\{q \in Y | q \leq W-w_i\}} a_{ipqs} X_{ipq}, \text{ para todo } (r, s) \in (X, Y).$$

Passo 6: Defina o tamanho do passo t por:

$$t = \frac{f(Z_{UB} - Z_{LB})}{\sum_{r \in X} \sum_{s \in Y} G_{rs}^2},$$

onde $0 < f \leq 2$, e atualize os multiplicadores de Lagrange por:

$$\lambda_{rs} = \max(0, \lambda_{rs} + tG_{rs}), \text{ para todo } (r, s) \in (X, Y).$$

Passo 7: Se $it = ITMAX$, pare; caso contrário, vá para o *Passo 2*.

Algumas variações deste algoritmo podem ser encontradas na literatura. Por exemplo, Camerini *et al.* (1975) sugeriram modificações na atualização do tamanho do passo em cada iteração $m(t^m)$. Na primeira iteração o tamanho é dado por um valor fixo ($t^1 = 10^{-5}$) e nas demais iterações, (t^m) é definido pela expressão a seguir (note que os subgradientes utilizados dependem dos subgradientes das iterações anteriores):

$$t^m = \frac{f(Z_{UB} - Z_{LB})}{\|s^m\|^2}$$

$$\text{onde: } s^m = G_m + \beta_m s^{m-1}, \quad s^{m-1} = 0 \text{ para } m = 0, \quad \beta_m = \begin{cases} -\gamma \frac{s^{m-1} G_{rs}}{\|s^{m-1}\|^2}, & \text{se } s^{m-1} G_{rs} < 0 \\ 0, & \text{caso contrário} \end{cases} \quad \text{e } \gamma = 1.4142.$$

Em Reeves (1993) é sugerido que uma boa prática para a convergência do procedimento de otimização do subgradiente é ajustar os subgradientes antes do cálculo do tamanho do passo (t) (durante o *Passo 5*), usando $G_{rs} = 0$ sempre que $\tilde{G}_{rs} < 0$ e $\lambda_{rs} = 0$ (veja também Lucena, 2004). Ou seja, se a restrição (2) não está violada (isto é, $G_{rs} < 0$) na iteração corrente, e não possui multiplicador associado não nulo (isto é, $\lambda_{rs} > 0$), seu gradiente é desconsiderado, fixando-o em zero (*Passo 5*). Com isso, G_{rs} não contribui no cálculo do tamanho do passo t (*Passo 6*).

3.2 Relaxação *surrogate* (SUR)

A relaxação *surrogate* (SUR) foi introduzida por Glover (1965, 1968). Para mais detalhes de SUR, os leitores podem consultar, por exemplo, Greenberg & Pierskalla (1970), Glover (1975) e Karvan & Rardin (1979). Introduzindo os multiplicadores *surrogate* $\mu_{rs} > 0$ para todo $r \in X$ e todo $s \in Y$ da expressão (2) e aplicando SUR nesta expressão, obtemos o seguinte problema *surrogate* (relaxado):

$$\text{Max} \quad \sum_{i=1}^2 \sum_{\{p \in X | p \leq L - l_i\}} \sum_{\{q \in Y | q \leq W - w_i\}} x_{ipq} \quad (8)$$

sujeito a:

$$\sum_{i=1}^2 \sum_{\{p \in X | p \leq L - l_i\}} \sum_{\{q \in Y | q \leq W - w_i\}} V_{ipq} x_{ipq} \leq \sum_{r \in X} \sum_{s \in Y} \mu_{rs} \quad (9)$$

$$P \leq \sum_{i=1}^2 \sum_{\{p \in X | p \leq L - l_i\}} \sum_{\{q \in Y | q \leq W - w_i\}} x_{ipq} \leq Q \quad (10)$$

$$x_{ipq} \in \{0, 1\}, \quad i = 1, 2, \quad p \in X \text{ tal que } p \leq L - l_i, \quad q \in Y \text{ tal que } q \leq W - w_i \quad (11)$$

onde $V_{ipq} = \sum_{r \in X} \sum_{s \in Y} \mu_{rs} a_{ipqrs}$ (por conveniência, usamos a mesma notação anterior V_{ipq}).

Note que V_{ipq} sempre é não negativo, dado que $\mu_{rs} > 0$. Dados $\{\mu_{rs}\}$, o modelo (8)-(11) resulta em um problema 0-1 com coeficientes reais fácil de se resolver por inspeção, pois os coeficientes da função objetivo são todos iguais a 1. Seja n o número de variáveis x_{ipq} :

- Se $n \geq Q$, então escolha as Q variáveis com menores valores V_{ipq} , não ultrapassando o valor de $\sum_{r \in X} \sum_{s \in Y} \mu_{rs}$, e fixe-as em 1. Se ultrapassar este limite, escolha o maior número possível de variáveis com menores valores V_{ipq} não ultrapassando o limite, e fixe-as em 1.
- Se $P \leq n \leq Q$, então escolha as n variáveis com menores valores V_{ipq} , não ultrapassando o valor de $\sum_{r \in X} \sum_{s \in Y} \mu_{rs}$, e fixe-as em 1. Se ultrapassar este limite, escolha o maior número possível de variáveis com menores valores V_{ipq} não ultrapassando o limite, e fixe-as em 1.

- Se $n < P$, então escolha as P variáveis com menores valores V_{ipq} , não ultrapassando o valor de $\sum_{r \in X} \sum_{s \in Y} \mu_{rs}$, e fixe-as em 1. Se ultrapassar este limite o problema é infactível (isto nunca ocorre se P for definido como um limitante inferior para o problema).

As demais variáveis são fixadas em 0.

Não encontramos um contra-exemplo do PCP do produtor (usando $P=0$ e Q como um limitante de Barnes ou de exemplos equivalentes, definidos no início da seção 3), mostrando que o problema *surrogate* (8)-(11) não satisfaz a propriedade de integralidade. Também não encontramos, ao longo dos experimentos computacionais apresentados na seção 6, um contra-exemplo cujo limitante gerado pela relaxação *surrogate* fosse melhor do que o limitante gerado pela relaxação Lagrangiana. Isto sugere que o melhor limitante superior produzido pelo problema dual *surrogate*:

$$\min_{\mu > 0} \{\text{problema } surrogate (8)-(11)\}$$

para o problema (1)-(4) não deve ser melhor do que o limitante superior obtido pela relaxação linear de (1)-(4). Para resolver o problema dual *surrogate* pode ser utilizado o método de otimização do subgradiente, porém, não há garantia teórica de convergência (Galvão *et al.*, 2000). Métodos com garantia de convergência estão discutidos em Banerjee (1971), Dyer (1980) e Sarin *et al.* (1987), mas, devido às suas complexidades e dificuldades de implementação, estão além do escopo deste trabalho.

O procedimento de otimização do subgradiente para a resolução do problema (8)-(11) é similar ao da seção 3.1, trocando-se o problema Lagrangiano (5)-(7) pelo problema *surrogate* (8)-(11), com algumas adaptações nos *Passos 1 e 6* descritos a seguir:

Passo 1: Gere uma solução factível, por exemplo, a solução homogênea com todas as caixas arranjadas sob a mesma orientação. Faça Z_{LB} igual ao valor desta solução homogênea. Seja $ITMAX$ o máximo número de iterações e faça $it = 0$. Fixe $\mu_{rs} = 1$ para todo $r \in X$ e $s \in Y$, como valores iniciais para os multiplicadores.

Passo 6: Defina o tamanho do passo t por:

$$t = \frac{f(Z_{UB} - Z_{LB})}{\sum_{r \in X} \sum_{s \in Y} G_{rs}^2},$$

onde $0 < f \leq 2$, e atualize os multiplicadores *surrogate* por:

$$\mu_{rs} = \max(0, \mu_{rs} + tG_{rs}), \text{ para todo } (r, s) \in (X, Y).$$

3.3 Relaxação Lagrangiana *surrogate* (LAGSUR)

Beasley (1985) utilizou as duas restrições *surrogate* apresentadas abaixo, obtidas pela soma de todos os comprimentos possíveis $r \in X$ (usando $\mu_r = 1$) da restrição (2), e todas as larguras possíveis $s \in Y$ (usando $\mu_s = 1$) da restrição (2):

$$\sum_{r \in X} \left(\sum_{i=1}^2 \sum_{\{p \in X \mid p \leq L-l_i\}} \sum_{\{q \in Y \mid q \leq W-w_i\}} a_{ipqrs} x_{ipq} \right) \leq |X|, \text{ para todo } s \in Y \quad (12)$$

$$\sum_{s \in Y} \left(\sum_{i=1}^2 \sum_{\{p \in X \mid p \leq L-l_i\}} \sum_{\{q \in Y \mid q \leq W-w_i\}} a_{ipqrs} x_{ipq} \right) \leq |Y|, \text{ para todo } r \in X \quad (13)$$

Substituindo no problema (1)-(4) a restrição (2) pelas restrições *surrogate* (12) e (13), e introduzindo os multiplicadores de Lagrange $g_s (\geq 0)$ para todo $s \in Y$ em (12) e $h_r (\geq 0)$ para todo $r \in X$ em (13), obtemos o seguinte problema Lagrangiano do limitante superior (com relaxação *surrogate*). Note agora que temos apenas $|X|+|Y|$ multiplicadores, ao invés de $|X||Y|$ multiplicadores como no problema (5)-(7).

$$\text{Max} \quad \sum_{i=1}^2 \sum_{\{p \in X \mid p \leq L-l_i\}} \sum_{\{q \in Y \mid q \leq W-w_i\}} V_{ipq} x_{ipq} + \sum_{s \in Y} g_s |X| + \sum_{r \in X} h_r |Y| \quad (14)$$

sujeito a:

$$P \leq \sum_{i=1}^2 \sum_{\{p \in X \mid p \leq L-l_i\}} \sum_{\{q \in Y \mid q \leq W-w_i\}} x_{ipq} \leq Q \quad (15)$$

$$x_{ipq} \in \{0,1\}, \quad i=1,2, \quad p \in X \text{ tal que } p \leq L-l_i, \quad q \in Y \text{ tal que } q \leq W-w_i \quad (16)$$

onde $V_{ipq} = 1 - \left(\sum_{r \in X} \sum_{s \in Y} (g_s + h_r) a_{ipqrs} \right)$ (por conveniência, novamente usamos a mesma notação anterior V_{ipq}). Sejam $\{x_{ipq}\}$ representando os valores das variáveis $\{x_{ipq}\}$ na solução do problema Lagrangiano (14)-(16), com valor dado por:

$$Z_{UB} = \sum_{i=1}^2 \sum_{\{p \in X \mid p \leq L-l_i\}} \sum_{\{q \in Y \mid q \leq W-w_i\}} V_{ipq} x_{ipq} + \sum_{s \in Y} g_s |X| + \sum_{r \in X} h_r |Y|.$$

Note que Z_{UB} corresponde a um limitante superior para o valor ótimo do problema (1)-(4), para quaisquer $\{g_s\}$ e $\{h_r\}$ não negativos. Este limitante não é melhor do que o limitante produzido pelo problema (5)-(7), uma vez que deriva de um problema Lagrangiano construído a partir de restrições *surrogate* (com $\mu_r = 1$ e $\mu_s = 1$). Por outro lado, é mais fácil de ser computado, dado que envolve um menor número de multiplicadores de Lagrange. Convém observar que esta LAGSUR é diferente da relaxação Lagrangiana e *surrogate* utilizada em outros estudos da literatura como, por exemplo, em Lorena & Senne (1996) e Narciso & Lorena (1999).

Assim como no problema (5)-(7), dados $\{g_s\}$ e $\{h_r\}$, o problema (14)-(16) também pode ser facilmente resolvido por inspeção. O mesmo procedimento anteriormente descrito para resolver o problema (5)-(7) também pode ser aqui aplicado. Para resolver o problema dual Lagrangiano (com relaxação *surrogate*):

$$\min_{g_s \geq 0, h_r \geq 0} \{ \text{problema Lagrangiano (14)-(16)} \}$$

podemos utilizar o método de otimização do subgradiente, o qual é similar ao da seção 3.1, trocando-se o problema Lagrangiano (5)-(7) pelo problema (14)-(16), com pequenas modificações nos Passos 1, 5 e 6, conforme Farago & Morabito (2000):

Passo 1: Gere uma solução factível, por exemplo, a solução homogênea com todas as caixas arranjadas sob a mesma orientação. Faça Z_{LB} igual ao valor desta solução homogênea. Seja $ITMAX$ o máximo número de iterações e faça $it = 0$. Fixe $g_s = 0$ para todo $s \in Y$, e $h_r = 0$ para todo $r \in X$, como valores iniciais para os multiplicadores.

Passo 5: Calcule os subgradientes:

$$G_s = -|X| + \sum_{r \in X} \left(\sum_{i=1}^2 \sum_{\{p \in X | p \leq L - l_i\}} \sum_{\{q \in Y | q \leq W - w_i\}} a_{ipqrs} X_{ipq} \right), \text{ para todo } s \in Y,$$

$$H_r = -|Y| + \sum_{s \in Y} \left(\sum_{i=1}^2 \sum_{\{p \in X | p \leq L - l_i\}} \sum_{\{q \in Y | q \leq W - w_i\}} a_{ipqrs} X_{ipq} \right), \text{ para todo } r \in X.$$

Passo 6: Defina o tamanho do passo t por:

$$t = \frac{f(Z_{UB} - Z_{LB})}{\sum_{s \in Y} G_s^2 + \sum_{r \in X} H_r^2}$$

onde $0 < f \leq 2$, e atualize os multiplicadores de Lagrange por:

$$g_s = \max(0, g_s + tG_s), \text{ para todo } s \in Y,$$

$$h_r = \max(0, h_r + tH_r), \text{ para todo } r \in X.$$

4. Limitantes Inferiores e Heurísticas Lagrangiana e *Surrogate*

As soluções dos problemas Lagrangiano e *surrogate* do limitante superior podem ser utilizadas na obtenção de soluções factíveis (limitantes inferiores) para o problema (1)-(4) por meio de heurísticas Lagrangiana e *surrogate*. É possível também reduzir o tamanho do problema por meio de técnicas de redução, tentando fixar, sem perda de generalidade, certas variáveis em 0 ou 1. Tanto as heurísticas Lagrangiana e *surrogate*, quanto às técnicas de redução, podem ser aplicadas no procedimento de otimização do subgradiente.

4.1 Redução do problema

Em Farago & Morabito (2000) foram discutidas técnicas de fixação de variáveis (isto é, $x_{ipq} = 0$ ou $x_{ipq} = 1$) em cada iteração do procedimento de otimização do subgradiente, para reduzir o tamanho do problema. Estas técnicas foram definidas para LAG e LAGSUR, mas podem ser estendidas também para SUR. Naquele trabalho foram realizados testes quanto à inclusão ou não destas técnicas de redução para LAG e LAGSUR. Os autores concluem que é melhor utilizar a redução, dado que o esforço computacional de tentar fixar variáveis ao longo das iterações é compensatório. Desta forma, no presente trabalho estas técnicas de redução foram incluídas no procedimento de otimização do subgradiente para LAG, SUR e LAGSUR. Para detalhes destas técnicas, o leitor pode consultar Farago & Morabito (2000) e Oliveira (2004).

4.2 Heurística Lagrangiana

O desempenho dos testes de redução depende da qualidade do limitante inferior. Dessa forma, utilizamos o procedimento heurístico em Farago & Morabito (2000), que é simples e relativamente eficaz para encontrar uma solução factível (limitante inferior) a partir de uma solução do problema Lagrangiano. Por conveniência, descrevemos a seguir os passos desta heurística Lagrangiana:

- Passo 1:** Coloque todas as variáveis fixadas em 1 pela redução do problema na solução da heurística e retire-as da lista dos $\{V_{ipq}\}$. Ordene os $\{V_{ipq}\}$ restantes de forma decrescente e forme uma lista F com as variáveis x_{ipq} correspondentes aos V_{ipq} .
- Passo 2:** Pare se a lista F estiver vazia, caso contrário, coloque na solução heurística a primeira variável da lista F , e retire-a dessa lista.
- Passo 3:** Teste a factibilidade da solução heurística em questão. Se for factível, mantenha a variável correspondente na solução; caso contrário, desconsidere-a e volte para o *Passo 2*.

Ao término deste procedimento, podemos estar com uma solução factível melhor que as das iterações anteriores. Neste caso, atualizamos o limitante inferior Z_{LB} do problema (1)-(4). A otimalidade desta solução é testada no decorrer do método de otimização do subgradiente.

Consideramos também um procedimento de melhoria na heurística Lagrangiana apresentada acima. A idéia deste procedimento consiste em modificar a solução atual obtida com o objetivo de se encontrar uma melhor solução. Inicialmente, ordena-se a solução corrente de forma decrescente (ou seja, consideramos primeiramente todas as variáveis fixadas em 1). Em seguida, desconsideramos uma variável da solução ordenada e aplicamos novamente a heurística Lagrangiana, na tentativa de se obter uma solução melhor que a anterior. Este procedimento é repetido para cada variável fixada em 1 na solução corrente. No caso de termos um grande número de variáveis presentes na solução, estipula-se um número máximo de repetições deste procedimento (por exemplo, apenas para as primeiras m variáveis fixadas em 1).

4.3 Heurística *surrogate*

Utilizamos um simples procedimento heurístico, muito similar ao da seção 4.2, para encontrar uma solução factível (limitante inferior) a partir de uma solução do problema *surrogate*. A única diferença em relação à heurística Lagrangiana está no *Passo 1*, descrito abaixo:

- Passo 1:** Coloque todas as variáveis fixadas em 1 pela redução do problema na solução da heurística e retire-as da lista dos $\{V_{ipq}\}$. Ordene os $\{V_{ipq}\}$ restantes de forma crescente e forme uma lista F com as variáveis x_{ipq} correspondentes aos V_{ipq} .

Ao término deste procedimento, podemos estar com uma solução factível melhor que as das iterações anteriores. Neste caso, atualizamos o limitante inferior Z_{LB} . A otimalidade desta solução é testada no decorrer do método de otimização do subgradiente. Assim como a heurística Lagrangiana, a heurística *surrogate* é aplicada em cada iteração do procedimento de otimização do subgradiente.

5. Busca em Árvore

No final do procedimento de otimização do subgradiente, o menor limitante superior (Z_{min}) e o máximo limitante inferior (Z_{LB}) correspondente a uma solução factível podem não coincidir e, para resolver este problema, aplicamos o método *branch and bound* (Nemhauser & Wolsey, 1988). O problema é resolvido utilizando uma busca em árvore binária em que, em cada nó da árvore, é computado um limitante superior das relaxações Lagrangiana e/ou *surrogate*. Em cada nó da árvore é aplicado o procedimento de otimização do subgradiente e, se em qualquer iteração do procedimento, obtivermos $Z_{min} = Z_{LB}$, então o problema é resolvido otimamente.

O conjunto inicial de multiplicadores de cada nó é o conjunto associado com o melhor limitante superior encontrado no nó predecessor da árvore. Durante cada iteração do método de otimização do subgradiente, é aplicada a heurística Lagrangiana ou *surrogate* (seções 4.2 e 4.3) para gerar soluções factíveis para o problema original (1)-(4), a partir da solução obtida para o problema Lagrangiano e/ou *surrogate* (modelos LAG (5)-(7), SUR (8)-(11) e LAGSUR (14)-(16)). Também é aplicada a técnica de redução do problema mencionada na seção 4.1 para tentar fixar, sem perda de generalidade, alguma variável em 0 ou 1, e, desta maneira, reduzir o tamanho do problema (1)-(4).

Os critérios para sondagem do método *branch and bound* para os nós da árvore são: otimalidade, infactibilidade e valor de dominância. No caso de encontrarmos uma solução factível com $Z_{min} > Z_{LB}$ devemos fazer a divisão (*branch*) do nó e teremos então dois novos subproblemas a serem resolvidos, ou seja, um novo nó à esquerda e outro à direita. Neste caso, selecionamos uma variável e esta será fixada com valor igual a 1 no nó da esquerda e valor igual a 0 no nó da direita. O nó da esquerda é considerado primeiro do que o da direita; assim, ao fazermos a sondagem de um nó, exploramos primeiro o nó da esquerda. Intuitivamente, estamos interessados em colocar o máximo número de caixas em um palete e, portanto, devemos explorar primeiro a possibilidade de se colocar tais variáveis na solução do problema e avaliar se esta solução é viável ou não.

A seguir é apresentado resumidamente o algoritmo do procedimento de busca em árvore implementado neste trabalho (Nemhauser & Wolsey, 1988). Este algoritmo está detalhado em pseudo-código em Oliveira (2004). Sejam:

- L lista de problemas inteiros (P^i) que devem ser investigados ou lista de nós abertos;
- S^i conjunto das soluções factíveis do problema (P^i) pertencente a L ;
- Z_{LB} limitante inferior para o valor objetivo de P (solução factível);
- Z_{UB} limitante superior para o valor objetivo de P (relaxação);
- Z valor da função objetivo do problema P .

Passo 1: Inicialização: forme uma lista $L = \{P\}$ (sendo P o problema (1)-(4)) com os problemas ainda não resolvidos, $S = S^0$, $Z_{UB}^0 = \infty$ e $Z_{LB} = -\infty$.

Passo 2: Teste de parada: se $L = \emptyset$ então a solução x^i que assegura $Z_{LB} = Z$ é a solução ótima.

Passo 3: Seleção do problema e relaxação: selecione e elimine um problema P^i de L . Resolva a relaxação do problema selecionado (RP^i). Seja Z_R^i o valor ótimo da relaxação e seja x_R^i a solução ótima se existir.

Passo 4: Sondagem:

- a) se $Z_R^i \leq Z_{LB}$, vá para o *Passo 2*;
- b) se $x_R^i \notin S^i$, vá para o *Passo 5*;
- c) se $x_R^i \in S^i$, faça $Z = Z_R^i$ e se $Z > Z_{LB}$, então faça $Z_{LB} = Z$, elimine de L todos os problemas com $Z_{UB}^i \leq Z_{LB}$ e vá para o *Passo 2*.

Passo 5: Divisão (*branch*): Seja $\{S^{ij}\}$, $j=1,2$, a divisão de S^i . Assim teremos dois novos subproblemas e devemos selecionar uma variável e esta será fixada com valor igual a 1 no nó da esquerda e valor igual a 0 no nó da direita. Esses novos subproblemas (P^{ij}) , $j=1,2$, são incluídos na lista L onde $Z_{UB}^{ij} = Z_R^i$. Em seguida, vá para o *Passo 2*.

5.1 Regras para escolha da variável a ser fixada

No *Passo 5* do algoritmo de busca em árvore acima, devemos fazer a divisão dos nós (*branching*), ou seja, temos dois novos subproblemas e devemos selecionar uma variável para ser fixada com valor igual a 1 no nó da esquerda e valor igual a 0 no nó da direita. As três regras definidas abaixo foram testadas para escolher a variável a ser fixada na divisão dos nós. Nos experimentos realizados (seção 6), estas regras se comportaram melhor do que simplesmente escolher a variável a ser fixada na divisão dos nós de forma aleatória, sem considerar nenhum critério pré-definido, ou simplesmente escolher a variável com maior valor Lagrangiano, ou seja, com maior V_{ipq} para ser fixada em 0 ou em 1.

- Regra 1 (Beasley, 1985)

Em qualquer nó da árvore de busca, seja $G = \{(i, p, q) \notin F \mid x_{ipq} \neq 0\}$ (onde F é a lista com as variáveis x_{ipq} correspondentes aos V_{ipq}), isto é, G é o conjunto de posições onde ainda é possível colocar uma caixa, pois este conjunto não considera as posições (i, p, q) com $x_{ipq} = 1$ ($(i, p, q) \notin F$) e ainda, as posições (i, p, q) com $x_{ipq} = 0$. Então definimos,

$$R = \min(r \mid (i, r, s) \in G), \quad S = \min(s \mid (i, r, s) \in G) \quad \text{e} \quad V_{jRS} = \max(V_{iRS} \mid (i, R, S) \in G).$$

A escolha da variável a ser fixada em 0 e em 1 corresponde a colocar a caixa j no palete (L, W) com o seu canto inferior esquerdo em (R, S) . Intuitivamente esta escolha corresponde a colocar no palete (L, W) a caixa j com máximo valor Lagrangiano, que então será colocada na posição mais abaixo e mais a esquerda, ou seja, no ponto (R, S) formando um padrão normalizado na busca em árvore. Podemos dizer que a escolha dessas variáveis nos dá padrões de empacotamento em camadas, pois as caixas são colocadas sempre de forma a encontrar os pontos (R, S) com coordenadas mais abaixo e mais a esquerda e, desta maneira, as caixas são colocadas lado a lado sem espaços entre elas (ou com o mínimo espaço).

Ainda, ao ser considerado que $x_{jRS} = 1$ significa que devemos ter $x_{ipq} = 0$ se existe um ponto (r, s) tal que $a_{jRSrs} + a_{ipqrs} = 2$. Isto porque, ao fixar a variável $x_{jRS} = 1$, não podemos ter

outra caixa no ponto (r, s) , pois, neste caso, haverá sobreposição das peças e assim devemos retirar tais variáveis do problema, isto é, considerar tais variáveis como iguais a zero ($x_{ipq} = 0$).

- Regra 2 (Reeves, 1993)

No procedimento de busca em árvore baseado no limitante de relaxação Lagrangiana (modelo (5)-(7)), a regra de seleção da variável a ser fixada é a seguinte:

1. Seja (λ, X) representando os multiplicadores e os valores das variáveis associados com o melhor limitante inferior encontrado no nó que está sendo ramificado;
2. Considere o vetor de valor $\sum_{r \in X} \sum_{s \in Y} \lambda_{rs} \left(1 - \sum_{i=1}^2 \sum_{\{p \in X / p \leq L - l_i\}} \sum_{\{q \in X / q \leq W - w_i\}} a_{ipqrs} X_{ipq} \right) = \lambda G$ e escolha a restrição com maior valor absoluto neste vetor;
3. Selecione a seguir a variável com maior valor Lagrangiano, ou seja, com maior V_{ipq} .

- Regra 3 – Estratégia utilizando o conceito de memória de longo prazo de busca tabu

Utilizamos uma estratégia para a escolha da variável que aplica o conceito de memória de longo prazo da meta-heurística busca tabu. Nesta estratégia, consideramos a frequência com que as variáveis aparecem na solução do problema. A cada determinado número de iterações (por exemplo, 5 ou 10 iterações), escolhe-se uma variável que tenha baixa frequência (por exemplo, frequência menor que 0,35, ou seja, se a variável escolhida apareceu em 35% ou menos das soluções nas iterações anteriores) e então esta variável é escolhida para ser fixada na divisão dos nós. A idéia é diversificar a solução do problema, fazendo com que as variáveis que tenham baixa frequência façam parte da solução. No restante das iterações, utilizamos a regra 1 definida acima.

Alguns autores recomendam aplicar um maior esforço computacional (maior número de iterações na otimização do subgradiente) no nó inicial da árvore. Com isto a heurística Lagrangiana tem maior chance de encontrar boas soluções factíveis, o procedimento de redução tem maior chance de reduzir o tamanho do problema e, ainda, a otimização do subgradiente tem maior chance de encontrar melhores limitantes. Com respeito a inicialização dos multiplicadores de Lagrange e ao número de iterações na otimização do subgradiente, autores têm sugerido (Reeves, 1993): (a) iniciar os multiplicadores de Lagrange em cada nó da árvore com o conjunto associado com o melhor limitante superior encontrado no nó predecessor da árvore; (b) executar 30 iterações do subgradiente, com $f (0 < f \leq 2)$ sendo dividido a cada 5 (ou 10) iterações nos nós da árvore das ramificações seguintes e, (c) duplicar ambos os parâmetros (para 60 sendo dividido a cada 10 (ou 20)) nos nós da árvore associados com a sondagem.

No presente trabalho, nos métodos LAG, SUR e LAGSUR os multiplicadores de Lagrange e/ou *surrogate* são iniciados em cada nó da árvore com o conjunto associado com o melhor (máximo) limitante superior encontrado no nó predecessor da árvore, segundo a estratégia apresentada acima. Ainda, na árvore de busca, para o cálculo do limitante superior foram utilizados os limitantes da Razão da Área e o de Barnes. O limitante inferior pode ser iniciado com a solução homogênea com todas as caixas arranjadas na mesma direção, ou ainda com a heurística de Morabito & Morales (1998), que é relativamente simples e efetiva.

6. Resultados Computacionais

Nesta seção são descritos os resultados dos testes computacionais realizados com o objetivo de avaliar o desempenho dos métodos exatos LAG, SUR e LAGSUR. As implementações foram codificadas em linguagem Pascal (compilador Delphi 5), e os testes foram realizados em um Pentium III 650 MHz. Nestes experimentos, admitimos no modelo (1)-(4) que $P=0$ e Q é igual ao mínimo entre os limitantes de Barnes e de exemplos equivalentes (seção 3).

6.1 Resultados com Grupos 1 e 2

Inicialmente foram realizados diversos testes computacionais preliminares para se verificar o desempenho de possíveis variações nos procedimentos de otimização do subgradiente e na busca em árvore dos métodos LAG, SUR e LAGSUR. Em relação à otimização do subgradiente consideramos: (i) utilizar ou não o método de redução do problema (seção 4.1), (ii) utilizar ou não as heurísticas Lagrangiana e *surrogate* (seções 4.2 e 4.3), (iii) utilizar ou não a melhoria da heurística Lagrangiana (seção 4.2), (iv) utilizar ou não modificações na atualização dos multiplicadores baseados na discussão da seção 3.1 (v) utilizar ou não modificações propostas por Camerini *et al.* (1975) para cálculo do tamanho do passo t (seção 3.1), e (vi) qual variação escolher para iniciar e reduzir o valor do parâmetro f do tamanho do passo t (seção 3.1).

Com respeito a este último item, investigamos iniciar este parâmetro com $f=2$ ou $f=1$, sendo reduzido após um determinado número de iterações. Sete variações foram exploradas para reduzir o valor de f ao longo das iterações: (i) dividir f a cada 60 iterações, (ii) dividir f a cada 30 iterações, (iii) dividir f a cada 15 iterações, (iv) dividir f a cada 60 iterações se Z_{min} não diminuir nas últimas 60 iterações, (v) dividir f a cada 30 iterações se Z_{min} não diminuir nas últimas 30 iterações, e (vi) dividir f a cada 15 iterações se Z_{min} não diminuir nas últimas 15 iterações, (vii) dividir f a cada $|X||Y|$ iterações se Z_{min} não diminuir nas últimas $|X||Y|$ iterações até $Z_{min} < Q$, então dividir f a cada 30 iterações se Z_{min} não diminuir nas últimas 30 iterações (conforme Farago & Morabito, 2000).

Em relação ao procedimento de busca em árvore consideramos: (i) qual regra escolher para a seleção da variável a ser fixada na divisão dos nós (seção 5.1), (ii) executar ou não um número maior de iterações no nó inicial da árvore (seção 5), (iii) utilizar ou não valores de multiplicadores associados com o melhor limitante superior obtido no nó predecessor da árvore (seção 5), (iv) utilizar ou não os conjuntos de *raster points* ao invés dos conjuntos normais (seção 2), e (v) utilizar ou não a heurística construtiva de Morabito & Morales (1998) para gerar bons limitantes inferiores iniciais (seção 5).

Para viabilizar a realização destes diversos testes preliminares, optamos por utilizar apenas dois grupos de exemplos com características diferentes. No primeiro grupo (subdivido nos Grupos 1A e 1B conforme Tabelas 1 e 2), escolhemos aleatoriamente 20 exemplos do conjunto Cover I da literatura (Nelissen, 1993, 1994; Scheithauer & Terno, 1996), que contém exemplos do PCP do produtor com solução ótima conhecida (composta de até 50 caixas). O Grupo 1A contém 10 exemplos (L1-L10) que não tem *gap* entre o valor da solução ótima e o valor da solução da relaxação linear de (1)-(4), enquanto o Grupo 1B contém 10 exemplos (L11-L20) que tem *gap* e, portanto, são mais desfavoráveis para os métodos aqui propostos. Nas tabelas abaixo, as colunas $2|X||Y|$ e $2|X'||Y'|$ representam o

número (aproximado) de variáveis 0–1 envolvidas no modelo (1)-(4) usando os conjuntos normais X e Y e os conjuntos de *raster points* X' e Y' , respectivamente.

O segundo grupo de exemplos foi subdividido nos Grupos 2A e 2B conforme Tabelas 3 e 4. No Grupo 2A consideramos 9 exemplos (L21-L29) analisados em diversos artigos, como Dowsland (1993), Nelissen (1994), Scheithauer & Terno (1996), Morabito & Morales (1998), Farago & Morabito (2000) e Pureza & Morabito (2006). Assim como o Grupo 1A, tais exemplos não tem *gap* entre o valor da solução ótima e o valor da relaxação linear de (1)-(4). Os 10 exemplos do Grupo 2B (L30-L39) foram analisados em Letchford & Amaral (2001) e, assim como o Grupo 1B, tem *gap* e, portanto são mais desfavoráveis para os métodos aqui propostos. Todos os exemplos dos Grupos 1A, 1B, 2A e 2B têm solução ótima não-guilhotinada de primeira ordem (esta classificação foi definida em Arenales & Morabito (1995) e refere-se a padrões não-guilhotinados mais simples).

Tabela 1 – Exemplos do Grupo 1A – sem *gap* entre solução ótima e relaxada.

GRUPO 1A								
Exemplo	(L,W)	(l,w)	$ X $	$ Y $	$2(X Y)$	$ X' $	$ Y' $	$2(X' Y')$
L1	(11,5)	(4,1)	11	5	110	11	5	110
L2	(16,11)	(4,3)	11	6	132	10	5	100
L3	(24,18)	(8,3)	15	9	270	10	6	120
L4	(28,27)	(8,3)	19	23	874	14	13	364
L5	(14,14)	(3,2)	12	12	288	12	12	288
L6	(20,16)	(5,2)	17	13	442	16	12	384
L7	(48,39)	(11,4)	30	21	1260	18	15	540
L8	(31,20)	(7,2)	27	16	864	25	14	700
L9	(11,8)	(2,1)	11	8	176	11	8	176
L10	(33,30)	(5,4)	24	21	1008	21	18	756
Média			17,70	13,40	542,40	14,80	10,80	353,80

Tabela 2 – Exemplos do Grupo 1B – com *gap* entre solução ótima e relaxada.

GRUPO 1B								
Exemplo	(L,W)	(l,w)	$ X $	$ Y $	$2(X Y)$	$ X' $	$ Y' $	$2(X' Y')$
L11	(36,28)	(8,3)	27	19	1026	22	14	616
L12	(36,36)	(10,3)	25	25	1250	18	18	648
L13	(60,33)	(9,5)	40	48	3840	28	11	616
L14	(37,25)	(7,3)	29	17	986	25	13	650
L15	(30,30)	(5,4)	21	21	882	18	18	648
L16	(50,36)	(8,5)	32	18	1152	22	14	616
L17	(39,39)	(11,3)	27	27	1458	19	19	722
L18	(33,30)	(7,3)	25	22	1100	21	18	756
L19	(37,26)	(5,4)	28	17	952	25	14	700
L20	(45,32)	(10,3)	34	21	1428	27	14	756
Média			28,80	23,50	1407,40	22,50	15,30	672,80

Tabela 3 – Exemplos do Grupo 2A – sem *gap* entre solução ótima e relaxada.

GRUPO 2A								
Exemplo	(L,W)	(l,w)	$ X $	$ Y $	$2(X Y)$	$ X^* $	$ Y^* $	$2(X^* Y^*)$
L21	(22,16)	(5,3)	16	10	320	14	8	224
L22	(86,82)	(15,11)	24	22	1056	17	16	544
L23	(57,44)	(12,5)	31	19	1178	19	14	532
L24	(42,39)	(9,4)	27	24	1296	20	15	600
L23	(124,81)	(21,10)	41	18	1476	24	13	624
L26	(40,25)	(7,3)	32	17	1088	28	13	728
L27	(52,33)	(9,4)	37	18	1332	28	13	728
L28	(56,52)	(12,5)	30	26	1560	21	17	714
L29	(87,47)	(7,6)	67	27	3618	57	17	1938
Média			33,89	20,11	1436,00	25,33	14,00	736,89

Tabela 4 – Exemplos do Grupo 2B – com *gap* entre solução ótima e relaxada.

GRUPO 2B								
Exemplo	(L,W)	(l,w)	$ X $	$ Y $	$2(X Y)$	$ X^* $	$ Y^* $	$2(X^* Y^*)$
L30	(32,22)	(5,4)	23	13	598	20	11	440
L31	(40,26)	(7,4)	28	14	784	22	11	484
L32	(100,64)	(17,10)	32	14	896	22	11	484
L33	(32,27)	(5,4)	23	18	828	20	15	600
L34	(37,30)	(8,3)	28	21	1176	23	16	736
L35	(100,82)	(22,8)	32	23	1472	23	16	736
L36	(100,83)	(22,8)	32	23	1472	23	16	736
L37	(40,33)	(7,4)	28	21	1176	22	16	704
L38	(53,26)	(7,4)	41	14	1148	35	11	770
L39	(81,39)	(9,7)	51	13	1326	33	11	726
Média			31,80	17,40	1087,60	24,30	13,40	641,60

Em relação à otimização do subgradiente, os resultados destes testes com os exemplos dos Grupos 1 e 2 apontaram para: (i) utilizar o método de redução do problema, (ii) utilizar as heurísticas Lagrangiana e *surrogate*, (iii) não utilizar a melhoria da heurística Lagrangiana, (iv) não utilizar as modificações na atualização dos multiplicadores (seção 3.1), (v) não utilizar as modificações propostas por Camerini *et al.* (1975) para cálculo do tamanho do passo t , e (vi) iniciar o parâmetro com $f=1$ e dividi-lo a cada 30 iterações.

Em relação ao procedimento de busca em árvore, os resultados dos testes apontaram para: (i) utilizar a regra 1 para a seleção da variável a ser fixada na divisão dos nós, (ii) não executar um número maior de iterações no nó inicial da árvore, (iii) utilizar os valores de multiplicadores associados com o melhor limitante superior obtido no nó predecessor da árvore, (iv) utilizar os conjuntos de *raster points* ao invés dos conjuntos normais, e (v) utilizar a heurística construtiva de Morabito & Morales (1998) para gerar bons limitantes inferiores iniciais.

Por motivo de espaço, a seguir apresentamos apenas alguns resultados destes testes. Os demais resultados encontram-se detalhados em Oliveira (2004). Convém salientar que em todos estes testes o método LAG em média superou os métodos SUR e LAGSUR. Por exemplo, a Tabela 5 compara os resultados obtidos com os métodos LAG, SUR e LAGSUR

para os exemplos L1-L10 do Grupo 1A, iniciando com parâmetro $f = 2$ e dividindo-o a cada 60 iterações, utilizando os conjuntos normais e não utilizando a heurística de Morabito & Morales (1998). A tabela apresenta as médias do limitante superior (Z_{min}) e do limitante inferior (Z_{LB}) no nó inicial da árvore para cada um dos métodos. Ainda são apresentados os valores médios da solução ótima (Z^*), da solução obtida em um tempo computacional limitado em 3 horas de processamento, do número de nós envolvidos na busca em árvore, e do tempo total (em minutos) gasto para se resolver o conjunto de exemplos (computado apenas para os exemplos resolvidos em menos de 3 horas).

Tabela 5 – Resultados computacionais dos métodos LAG, SUR e LAGSUR para o Grupo 1A (utilizando $f = 2$ e divisão de f a cada 60 iterações, utilizando os conjuntos normais e sem a heurística de Morabito & Morales, 1998).

Método exato	Nó inicial		Busca em árvore			
	Z_{min}	Z_{LB}	Z^*	Solução obtida	# nós	Tempo (min)
LAG	31,90	29,60	31,90	31,90*	4,00	0,24
SUR	31,90	29,60	31,90	31,50	3,67	0,06
LAGSUR	31,90	29,60	31,90	31,70	9,25	1,30

Podemos observar na Tabela 5 que entre os três métodos, o que apresentou melhores resultados quanto ao valor da solução ótima e ao tempo computacional foi o método LAG (lembre-se que os tempos correspondem à média apenas dos exemplos resolvidos em menos de 3 horas). O método LAG conseguiu resolver todos os exemplos do Grupo 1A (indicado na tabela com um asterisco), diferente dos métodos SUR e LAGSUR. Outros experimentos foram realizados com os outros grupos, variando-se os parâmetros da otimização do subgradiente e da busca em árvore, e o método LAG continuou apresentando desempenho superior aos métodos SUR e LAGSUR. Uma vantagem do método LAG sobre o método LAGSUR é o maior número de multiplicadores ($|X||Y|$ ao invés de $|X| + |Y|$), o que implica em mais informações sobre as restrições do problema ao longo das iterações da otimização do subgradiente. Com relação ao método SUR, aparentemente ele não produz limitantes (do dual *surrogate*) melhores do que os limitantes do método LAG (do dual Lagrangiano). Além disso, a otimização do subgradiente para SUR não tem garantia de convergência, conforme discussão na seção 3.2.

A seguir apresentamos alguns resultados obtidos para os Grupos 1 e 2 comparando algumas das variações descritas acima para a otimização do subgradiente e a busca em árvore do método LAG. Outros resultados estão apresentados em Oliveira (2004). As tabelas apresentam a solução obtida num tempo computacional máximo de 3 horas, iniciando com parâmetro $f = 1$ e dividindo-o a cada 30 iterações. Por ilustração, as Tabelas 6 a 8 comparam o impacto de: (i) utilizar conjuntos normais ou conjuntos de *raster points*, e (ii) utilizar ou não a heurística construtiva de Morabito & Morales (1998) para gerar bons limitantes inferiores iniciais. A Tabela 7 mostra que, utilizando os conjuntos de *raster points*, o método LAG foi capaz de resolver todos os exemplos em tempos computacionais menores do que os apresentados na Tabela 6, especialmente para os Grupos 2A e 2B. Note também que o número de nós envolvidos na busca em árvore foi em média um pouco menor considerando os *raster points*. A Tabela 8 mostra que, utilizando a heurística de Morabito & Morales (1998), os tempos computacionais foram reduzidos nos problemas que não apresentam *gap*

entre a solução ótima e a solução de relaxação linear (Grupos 1A e 2A). Já para os Grupos 1B e 2B, os tempos computacionais foram aproximadamente mantidos, em relação aos mostrados na Tabela 7.

Tabela 6 – Resultados computacionais do método LAG para Grupos 1 e 2 utilizando:
(i) conjuntos normais, (ii) sem heurística de Morabito & Morales (1998).

Grupo	Nó inicial		Busca em árvore			
	Z_{min}	Z_{LB}	Z^*	Solução obtida	# nós	Tempo (min)
1A	31,90	29,60	31,90	31,90*	10,0	0,26
1B	45,20	40,50	44,20	44,20*	29,20	1,27
2A	48,56	41,89	48,56	48,56*	95,00	15,72
2B	43,60	37,90	42,60	42,60*	131,60	6,30

Tabela 7 – Resultados computacionais do método LAG para Grupos 1 e 2 utilizando:
(i) conjuntos de *raster points*, (ii) sem heurística de Morabito & Morales (1998).

Grupo	Nó inicial		Busca em árvore			
	Z_{min}	Z_{LB}	Z^*	Solução obtida	# nós	Tempo (min)
1A	31,90	29,60	31,90	31,90*	10,0	0,18
1B	45,20	40,50	44,20	44,20*	25,20	0,72
2A	48,56	41,89	48,56	48,56*	76,00	4,55
2B	43,60	37,90	42,60	42,60*	105,80	2,67

Tabela 8 – Resultados computacionais do método LAG para Grupos 1 e 2 utilizando:
(i) conjuntos de *raster points*, (ii) com heurística de Morabito & Morales (1998).

Grupo	Nó inicial		Busca em árvore			
	Z_{min}	Z_{LB}	Z^*	Solução obtida	# nós	Tempo (min)
1A	31,90	31,90	31,90	31,90*	1,00	< 0,01
1B	45,20	44,20	44,20	44,20*	18,00	0,70
2A	48,56	48,44	48,56	48,56*	1,00	0,02
2B	43,60	42,60	42,60	42,60*	104,00	3,31

6.2 Resultados com Grupos 3 e 4

Também consideramos outros grupos de exemplos além dos Grupos 1 e 2. O terceiro grupo de exemplos foi subdividido nos Grupos 3A e 3B conforme Tabelas 9 e 10. Os 16 exemplos do Grupo 3A (L40-L55) foram escolhidos aleatoriamente do conjunto Cover II da literatura (Nelissen, 1993, 1994; Scheithauer & Terno, 1996), que contém exemplos do PCP do produtor com solução ótima conhecida (composta de 51 a 100 caixas). Todos os exemplos do Grupo 3A têm solução ótima não-guilhotinada de primeira ordem. Os 16 exemplos do Grupo 3B (L56-L71) são exemplos do Cover II difíceis de se resolver com as heurísticas de literatura (Morabito & Morales, 1998; Lins *et al.*, 2003), pois nenhum deles tem solução ótima não-guilhotinada de primeira ordem. O quarto grupo de exemplos (Grupo 4) refere-se

a 30 exemplos reais (R1-R30 na Tabela 11) de uma transportadora em Campinas, SP. Estes exemplos utilizam o palete Brasil (PBR: 120x100 cm), adotado na maioria das empresas brasileiras. Note nas tabelas que o número médio de variáveis 0–1 dos Grupos 3A e 3B é bem maior, se comparado aos demais grupos, enquanto que o número médio de variáveis 0–1 no Grupo 4 é bem menor.

Tabela 9 – Exemplos do Grupo 3A – com solução ótima de primeira ordem.

GRUPO 3A								
Exemplo	(L,W)	(l,w)	$ X $	$ Y $	$2(X Y)$	$ X^* $	$ Y^* $	$2(X^* Y^*)$
L40	(77,47)	(10,7)	44	17	1496	28	13	728
L41	(41,36)	(7,4)	29	24	1392	23	18	828
L42	(55,55)	(11,5)	31	31	1922	18	18	648
L43	(44,29)	(5,4)	35	20	1400	32	17	1088
L44	(85,46)	(12,5)	59	21	2478	42	15	1260
L45	(51,47)	(7,5)	35	31	2170	37	23	1702
L44	(70,43)	(7,6)	50	23	2300	40	16	1280
L47	(68,56)	(13,4)	47	35	3290	32	22	1408
L48	(52,52)	(9,4)	37	37	2738	28	28	1568
L49	(52,36)	(8,3)	43	27	2322	38	22	1672
L50	(78,40)	(8,5)	60	22	2640	50	16	1600
L51	(70,63)	(11,5)	46	39	3588	30	27	1620
L52	(120,64)	(19,5)	80	27	4320	48	19	1824
L53	(60,56)	(9,4)	45	41	3690	36	32	2304
L54	(100,82)	(17,5)	64	46	5888	42	27	2268
L55	(113,57)	(11,6)	83	27	4482	63	19	2394
Média			49,25	29,25	2882,25	36,69	20,75	1512,00

Tabela 10 – Exemplos do Grupo 3B – sem solução ótima de primeira ordem.

GRUPO 3B								
Exemplo	(L,W)	(l,w)	$ X $	$ Y $	$2(X Y)$	$ X^* $	$ Y^* $	$2(X^* Y^*)$
L56	(43,26)	(7,3)	35	18	1260	31	14	868
L57	(49,28)	(8,3)	40	19	1520	35	14	980
L58	(57,34)	(7,4)	45	22	1980	39	17	1326
L59	(63,44)	(8,5)	45	26	2340	35	19	1330
L60	(61,35)	(10,3)	50	24	2400	43	17	1462
L61	(67,37)	(11,3)	55	25	2750	47	18	1692
L62	(61,38)	(10,3)	50	27	2700	43	20	1720
L63	(61,38)	(6,5)	47	24	2256	41	19	1558
L64	(67,40)	(11,3)	55	28	3080	47	20	1880
L65	(74,49)	(11,4)	56	31	3472	44	22	1936
L66	(93,46)	(13,4)	72	25	3600	57	17	1938
L67	(106,59)	(13,5)	78	31	4836	58	19	2204
L68	(141,71)	(13,8)	92	26	4784	61	19	2318
L69	(74,46)	(7,5)	58	30	3480	50	23	2300
L70	(86,52)	(9,5)	66	32	4224	54	23	2484
L71	(108,65)	(10,7)	75	32	4800	54	23	2484
Média			57,44	26,25	3092,63	46,19	19,00	1780,00

Tabela 11 – Exemplos do Grupo 4 – exemplos reais de uma transportadora com $(L, W)=(120,100)$ cm.

GRUPO 4							
Exemplo	(l,w)	$ X $	$ Y $	$2(X Y)$	$ X^* $	$ Y^* $	$2(X^* Y^*)$
R1	(31,22)	12	8	192	9	7	126
R2	(50,20)	9	7	126	8	6	96
R3	(33,23)	10	7	140	8	6	96
R4	(34,26)	9	6	108	8	5	80
R5	(36,15)	16	11	352	12	9	216
R6	(28,21)	12	9	216	11	8	176
R7	(32,18)	14	10	280	11	8	176
R8	(38,26)	8	5	80	7	5	70
R9	(25,15)	18	14	504	16	12	384
R10	(46,30)	6	4	48	4	4	32
R11	(39,25)	8	6	96	7	4	56
R12	(38,20)	12	9	216	6	5	60
R13	(49,20)	10	7	140	8	6	96
R14	(28,17)	17	11	374	13	9	234
R15	(40,29)	7	5	70	6	5	60
R16	(35,12)	22	15	660	10	8	160
R17	(27,22)	13	9	234	11	8	176
R18	(21,12)	28	21	1176	22	16	704
R19	(24,19)	17	12	408	13	10	260
R20	(32,24)	10	7	140	9	6	108
R21	(26,20)	15	11	330	12	9	216
R22	(19,14)	28	20	1120	19	15	570
R23	(44,29)	7	4	56	6	4	48
R24	(52,33)	5	4	40	5	3	30
R25	(36,21)	11	8	176	9	7	126
R26	(35,20)	12	9	216	6	5	60
R27	(20,14)	27	19	1026	18	14	504
R28	(22,17)	20	14	560	15	11	330
R29	(37,20)	12	9	216	6	5	60
R30	(24,13)	25	18	900	13	14	364
Média		14,00	9,97	340,00	10,27	7,80	189,13

A Tabela 12 apresenta os resultados do método LAG obtidos para os exemplos dos Grupos 3A e 3B, usando as melhores variações para a otimização do subgradiente e a busca em árvore descritas na seção 6.1. Observe na tabela que o método não encontrou a solução ótima de todos os exemplos do Grupo 3B dentro do limite de tempo de 3 horas. Nos exemplos L68, L69 e L70, a melhor solução obtida pelo método LAG empacota uma caixa a menos do que a solução ótima.

Tabela 12 – Resultados computacionais do método LAG para Grupo 3.

Grupo	Nó inicial		Busca em árvore			
	Z_{min}	Z_{LB}	Z^*	Solução obtida	# nós	Tempo (min)
3A	73,44	73,31	73,31	73,31*	12,63	1,34
3B	80,06	79,06	80,06	79,88	249,31	22,76

Tabela 13 – Resultados computacionais do método LAG para Grupo 4.

Exemplo	Sol. Transp. Americana	Nó inicial		Busca em árvore			Z^*-Z_{TA}
	Z_{TA}	Z_{min}	Z_{LB}	Z^*	Nº de nós	Tempo (min)	
R1	14	16	16	16	1	< 0,01	2
R2	10	12	12	12	1	< 0,01	2
R3	13	15	15	15	1	< 0,01	2
R4	12	12	12	12	1	< 0,01	0
R5	20	21	20	20	1	0,02	0
R6	18	19	19	19	1	< 0,01	1
R7	18	20	20	20	1	< 0,01	2
R8	10	10	10	10	1	< 0,01	0
R9	28	32	32	32	1	< 0,01	4
R10	7	8	8	8	1	< 0,01	1
R11	12	12	12	12	1	0,01	0
R12	15	15	15	15	1	< 0,01	0
R13	12	12	12	12	1	< 0,01	0
R14	23	24	23	23	1	0,01	0
R15	10	10	10	10	1	< 0,01	0
R16	25	27	26	26	1	< 0,01	1
R17	18	19	19	19	1	< 0,01	1
R18	46	47	46	46	75	2,93	0
R19	26	26	26	26	1	< 0,01	0
R20	15	15	15	15	1	< 0,01	0
R21	20	23	22	22	1	0,02	2
R22	41	43	43	43	1	< 0,01	2
R23	8	8	8	8	1	< 0,01	0
R24	6	6	6	6	1	< 0,01	0
R25	14	15	15	15	1	< 0,01	1
R26	14	17	15	15	1	0,01	1
R27	40	42	42	42	1	< 0,01	2
R28	27	31	31	31	1	< 0,01	4
R29	15	16	15	15	1	0,01	0
R30	37	38	38	38	1	< 0,01	1
Média				20,10	3,47	0,10	0,97

Lins *et al.* (2003) também realizaram testes computacionais com o Grupo 3B, e os resultados apresentados mostraram que o método proposto pelos autores (com conjectura de ser exato) resolve a maioria dos exemplos em menos de 1 hora de tempo computacional. Em outros exemplos, como os L67, L68 e L71, o tempo computacional foi bem maior, em torno de 8 horas. Convém observar, no entanto, que o método de Lins *et al.* (2003), ao resolver cada problema, também resolve muitos subproblemas deste problema, dado que utiliza programação dinâmica. O tempo médio dos 13 exemplos resolvidos em Lins *et al.* (2003) foi de 60,50 minutos, ou seja, maior que o tempo médio dos 13 exemplos resolvidos pelo método LAG (22,76 minutos, conforme a Tabela 12). O microcomputador utilizado foi um Pentium III 700 MHz, ou seja, comparável ao microcomputador utilizado no presente trabalho (Pentium III 650 MHz). Recentemente Birgin *et al.* (2005) realizaram experimentos adicionais com a abordagem em Lins *et al.* (2003), explorando estruturas de dados diferentes

daquelas utilizadas em Lins *et al.* (2003), mostrando que os tempos computacionais da abordagem podem ser bem reduzidos. No entanto, os requisitos de memória computacional são bem maiores do que os anteriores. Uma comparação mais efetiva entre os dois métodos está além do escopo deste trabalho.

A Tabela 13 compara as soluções do método LAG com as soluções utilizadas pela transportadora, para os 30 exemplos reais do Grupo 4. Note que o método LAG foi capaz de resolver otimamente todos os exemplos em um tempo médio relativamente pequeno (0,10 minutos), o que sugere que ele pode ser uma boa alternativa para resolver exemplos reais. Observamos ainda que em 16 exemplos, a solução da transportadora não é ótima. Podemos dizer que, em média, o método LAG é capaz de melhorar em 0,97 caixas (por camada do palete) a solução da transportadora. Observe nos exemplos R9 e R28 que a solução do método coloca 4 caixas a mais que a solução da transportadora.

6.3 Comparação com GAMS/CPLEX e resultados com Cover I e II

Implementamos o modelo (1)-(4) na linguagem de modelagem GAMS (Brooke *et al.*, 1992) e resolvemos todos os exemplos dos Grupos 1-4 pelo *software* CPLEX (versão 7 – que utiliza um método *branch and cut*). A Tabela 14 compara as soluções obtidas pelo GAMS/CPLEX e pelo método LAG. Note que, assim como o método LAG, o pacote GAMS/CPLEX foi capaz de resolver todos os exemplos em um tempo computacional máximo de 3 horas, exceto para o Grupo 3B. Nos Grupos 1 e 2, o método LAG é mais rápido do que o GAMS/CPLEX, principalmente nos Grupos 1A e 2A (que correspondem a exemplos sem *gap*). No Grupo 3, ele é mais rápido no Grupo 3A (1,34 minutos contra 9,23 minutos), mas mais lento no Grupo 3B (22,76 minutos contra 17,55 minutos). E no Grupo 4, ele é bem mais rápido (0,10 minutos contra 18,52 minutos). Assim, o método LAG parece ser uma boa alternativa para resolver exemplos reais.

Tabela 14 – Resultados computacionais do método LAG e do *software* GAMS/CPLEX para Grupos 1-4.

Grupo	Z^*	LAG	Tempo (min)	GAMS/CPLEX	Tempo (min)
1A	31,90	31,90*	<0,01	31,90*	0,19
1B	44,20	44,20*	0,70	44,20*	0,81
2A	48,56	48,56*	0,02	48,56*	4,05
2B	42,60	42,60*	3,31	42,60*	4,87
3A	73,31	73,31*	1,34	73,31*	9,23
3B	80,06	79,88	22,76	79,94	17,55
4	20,10	20,10*	0,10	20,10*	18,52

Finalmente, realizamos experimentos limitados com os 3179 exemplos do conjunto Cover I com $L, W \leq 1000$, com padrões ótimos de até 50 caixas. Dentre esses exemplos, 116 tem *gap*. O número médio de elementos dos conjuntos normais e *raster points* são $|X| = 25,9$, $|Y| = 14,2$, $|X'| = 19,8$ e $|Y'| = 11,0$. O número médio de variáveis 0-1 envolvidas no modelo (1)-(4) é $2|X||Y| = 735,56$ e $2|X'||Y'| = 435,60$, respectivamente. Estes exemplos são de interesse nas aplicações práticas do PCP do produtor. Também realizamos experimentos limitados com os 16938 exemplos do conjunto Cover II com $L, W \leq 1000$,

com padrões ótimos de 51 a 100 caixas. O número médio de elementos dos conjuntos normais e *raster points* são $|X|=54,6$, $|Y|=29,7$, $|X'|=38,4$ e $|Y'|=20,8$. O número médio de variáveis 0–1 envolvidas no modelo (1)-(4) é $2|X||Y|=3243,24$ e $2|X'||Y'|=1597,44$, respectivamente.

A Tabela 15 apresenta resumidamente os resultados obtidos com o método LAG para os conjuntos Cover I e II. Os tempos médios foram computados apenas para os exemplos resolvidos dentro do limite de tempo de 3 horas. Note que o método resolveu todos os exemplos de Cover I em menos de 3 horas (indicado na tabela com um asterisco), mas não foi capaz de resolver todos os exemplos de Cover II (82 dos 16938 exemplos não foram resolvidos; em todos eles o padrão obtido tem uma caixa a menos do que o padrão ótimo). Os tempos médios são bem aceitáveis para apoiar os tipos de decisões envolvidas nas situações práticas, tipicamente com problemas envolvendo menos de 50 caixas no carregamento (Cover I).

Tabela 15 – Resultados computacionais do método LAG para Cover I e II.

Cover I e II	Z^*	Solução obtida	# nós	Tempo (min)
Cover I – 3063 exemplos sem <i>gap</i>	35,98	35,98*	1,00	< 0,01
Cover I – 116 exemplos com <i>gap</i>	41,82	41,82*	23,79	0,79
Cover I – 3179 exemplos	36,19	36,19*	1,83	0,03
Cover II – 16938 exemplos	80,22	80,16	18,32	3,27

7. Conclusões

Neste trabalho estudamos métodos exatos baseados nas relaxações Lagrangiana (LAG), *surrogate* (SUR) e Lagrangiana com restrições *surrogate* (LAGSUR) para tratar o PCP do produtor. Diversos experimentos computacionais foram realizados e o desempenho do método LAG foi em média sempre superior aos dos métodos SUR e LAGSUR. Com relação ao método SUR, aparentemente ele não produz limitantes (do dual *surrogate*) melhores do que os limitantes do método LAG (do dual Lagrangiano). Além disso, a otimização do subgradiente do método SUR não tem garantia de convergência, conforme discussão na seção 3.2.

Nos diversos experimentos, o método LAG resolveu a maioria dos exemplos em um tempo computacional aceitável, levando-se em conta as decisões envolvidas na prática. Comparado com o *software* GAMS/CPLEX (que utiliza um método *branch and cut*), o método LAG foi em média mais rápido na maioria dos experimentos, principalmente nos exemplos reais de uma transportadora, o que sugere que ele seja eficiente para aplicações práticas. Apesar do desempenho do método LAG ser competitivo com outros métodos, capaz de gerar soluções ótimas e provar suas otimalidades em tempos computacionais relativamente pequenos (considerando as decisões envolvidas no PCP do produtor), nossa expectativa inicial era de

que ele teria desempenho computacional melhor do que o obtido ao longo dos experimentos, devido aos resultados obtidos em Morabito & Farago (2000).

Uma das perspectivas para pesquisas futuras é tentar reduzir os tempos computacionais dos métodos LAG, SUR e LAGSUR. Em particular pretende-se considerar as idéias apresentadas no trabalho de Alvarez-Valdes *et al.* (2005b) para reduzir o problema, considerando a redução de variáveis e restrições nos modelos com os conjuntos de *raster points* e considerações de dominância de posições nestes conjuntos. Pesquisas futuras podem ainda ser realizadas para melhorar o desempenho dos métodos LAG, SUR e LAGSUR, em particular, desenvolvendo heurísticas Lagrangianas e *surrogates* mais poderosas, com o objetivo de obter limitantes de melhor qualidade. Também, pode-se estudar a aplicação de técnicas de *relax and cut* (Lucena, 2004) no PCP do produtor, incorporando algoritmos de planos de corte aos métodos Lagrangianos.

Outra perspectiva importante para pesquisa futura é aplicar as técnicas de relaxação Lagrangiana e/ou *surrogate* em outros modelos 0–1 como, por exemplo, nos modelos de Tsai *et al.* (1993), Amaral & Letchford (1999) e Martins (2002) comparando, dessa forma, o desempenho de métodos LAG, SUR e LAGSUR com os diferentes modelos para o PCP do produtor. Uma outra contribuição seria desenvolver um procedimento para a busca dos multiplicadores do dual *surrogate* utilizando uma sucessão de buscas no dual Lagrangiano, como apresentado no trabalho de Sarin *et al.* (1987). Finalmente, seria importante mostrar se a relaxação *surrogate* produz ou não melhores limitantes do que a relaxação Lagrangiana para o PCP do produtor.

Agradecimentos

Os autores agradecem aos três revisores anônimos pelos úteis comentários e sugestões. Esta pesquisa contou com apoio do CNPq (processos 141337/2000-1 e 522973/95-7) e FAPESP (processo 9522-0).

Referências Bibliográficas

- (1) Alvarez-Valdes, R.; Parreño, F. & Tamarit, J.M. (2005a). A Tabu Search Algorithm for the Pallet Loading Problem. *OR Spectrum*, **27**(1), 43-61.
- (2) Alvarez-Valdes, R.; Parreño, F. & Tamarit, J.M. (2005b). A Branch-and-Cut Algorithm for the Pallet Loading Problem. *Computers & Operations Research*, **32**(11), 3007-3029.
- (3) Amaral, A. & Letchford, A. (1999). An Exact Algorithm for General, Orthogonal, Two-dimensional Knapsack Problems. Working paper, Department of Management Science, Lancaster University.
- (4) Arenales, M. & Morabito, R. (1995). An AND/OR-Graph to the Solution of Two-Dimensional Non-guillotine Cutting Problems. *European Journal of Operational Research*, **84**, 599-617.
- (5) Banerjee, K. (1971). Generalized Lagrange Multipliers in Dynamic Programming. Research Report No. OCR 71-12, Operations Research Center, University of California, Berkeley, California.

- (6) Barnes, F.W. (1979). Packing the Maximum Number of $m \times n$ Tiles in a Large $p \times q$ Rectangle. *Discrete Mathematics*, **26**, 93-100.
- (7) Beasley, J. (1985). An Exact Two-Dimensional Non Guillotine Cutting Tree Search Procedure. *Operations Research*, **33**, 49-64.
- (8) Bhattacharya, S.; Roy, R. & Bhattacharya, S. (1998). An Exact Depth-First Algorithm for the Pallet Loading Problem. *European Journal of Operational Research*, **110**, 610-625.
- (9) Birgin, E.G.; Morabito, R. & Nishihara, F.H. (2005). A Note on a L-Approach for Solving the Manufacturer's Pallet Loading Problem. *Journal of the Operational Research Society*, **56**, 1448-1451.
- (10) Bischoff, E. & Dowsland, W. (1982). An Application of the Micro to Product Design and Distribution. *Journal of the Operational Research Society*, **33**, 271-280.
- (11) Brooke, A.; Kendrick, D.; Meeraus, A. & Raman, R. (1998). *GAMS: A User's Guide*. GAMS Development Corporation, Washington.
- (12) Camerini, P.M.; Fratta, L. & Maffioli, F. (1975). On Improving Relaxation Methods by Gradient Techniques. *Mathematical Programming Study*, **3**, 26-34.
- (13) Christofides, N. & Whitlock, C. (1977). An Algorithm for Two-Dimensional Cutting Problems. *Operations Research*, **25**, 30-45.
- (14) Crowder, H. (1976). Computational Improvements for Subgradient Optimization. *Symposio Matematica*, Rome, **17**, 357-372.
- (15) Dowsland, K. (1987). An Exact Algorithm for the Pallet Loading Problem. *European Journal of Operational Research*, **31**, 78-84.
- (16) Dowsland, K. (1993). Some Experiments With Simulated Annealing Techniques for Packing Problems. *European Journal of Operational Research*, **68**, 389-399.
- (17) Dyckhoff, H. (1990). A Typology of Cutting and Packing Problems. *European Journal of Operational Research*, **44**, 145-159.
- (18) Dyer, M.E. (1980). Calculating Surrogate Constraints. *Mathematical Programming*, **19**, 255-278.
- (19) Farago, R. & Morabito, R. (2000). Um Método Heurístico Baseado em Relaxação Lagrangiana para o Problema de Carregamento de Paletes do Produtor. *Pesquisa Operacional*, **2(20)**, 197-212.
- (20) G, Y. & Kang, M. (2001). A Fast Algorithm for Two-dimensional Pallet Loading Problems of Large Size. *European Journal of Operational Research*, **134**, 193-200.
- (21) Galvão, R.D.; Espejo, L.G.A. & Boffey, B. (2000). A Comparison of Lagrangean and Surrogate Relaxations for the Maximal Covering Location Problem. *European Journal of Operational Research*, **124**, 377-389.
- (22) Glover, F. (1965). A Multiphase Dual Algorithm for the Zero-One Integer Programming Problem. *Operations Research*, **13**, 879-919.
- (23) Glover, F. (1968). Surrogate Constraints. *Operations Research*, **16(4)**, 741-749.

- (24) Glover, F. (1975). Surrogate Constraints Duality in Mathematical Programming. *Operations Research*, **23**, 434-451.
- (25) Greenberg, H.J. & Pierskalla, W.P. (1970). Surrogate Mathematical Programming. *Operations Research*, **18**, 924-939.
- (26) Hadjiconstantinou, E. & Christofides, N. (1995). An Exact Algorithm for General, Orthogonal, Two-Dimensional Knapsack Problems. *European Journal of Operational Research*, **83**, 39-56.
- (27) Held, M. & Karp, R.M. (1970). The Traveling Salesman Problem and Minimum Spanning Trees. *Operations Research*, **18**, 1138-1162.
- (28) Held, M. & Karp, R.M. (1971). The Traveling Salesman Problem and Minimum Spanning Trees: Part II. *Mathematical Programming*, **1**, 6-25.
- (29) Herbert, A. & Dowsland, K. (1996). A Family of Genetic Algorithms for the Pallet Loading Problem. *Annals of Operations Research*, **63**, 415-436.
- (30) Interlogis (2004). Disponível em: <www.interlogis.com.br/artigos>. Acesso em: 02/08/2004.
- (31) Karvan, M.H. & Rardin, R.L. (1979). Some Relationship between Lagrangian and Surrogate Duality in Integer Programming. *Mathematical Programming*, **17**, 320-334.
- (32) Letchford, A.N. & Amaral, A. (2001). Analysis of Upper Bounds for the Pallet Loading Problem. *European Journal of Operational Research*, **132**(3), 582-593.
- (33) Lins, L.; Lins, S. & Morabito, R. (2003). An *L*-Approach for Packing (*l,w*)-Rectangles into Rectangular and *L*-Shaped Pieces. *Journal of the Operational Research Society*, **54**, 777-789.
- (34) Lorena, L.A.N. & Senne, E.L.F. (1996). A Lagrangean/Surrogate Heuristic for Uncapacitated Facility Location Problems. In: *Anais dos resumos estendidos VIII CLAIO e XXVIII SBPO*, Rio de Janeiro, 854-859.
- (35) Lucena, A. (2004). Relax and Cut Algorithms. Submetido para publicação.
- (36) Martins, G.H.A. (2002). Packing in Two and Three Dimensions. Tese de Doutorado, Naval Postgraduate School, Monterey, California.
- (37) Morabito, R. & Morales, S.R. (1998). A Simple and Effective Recursive Procedure for the Manufacturer's Pallet Loading Problem. *Journal of the Operational Research Society*, **49**, 819-828.
- (38) Morabito, R. & Morales, S.R. (1999). Errata: A Simple and Effective Recursive Procedure for the Manufacturer's Pallet Loading Problem. *Journal of the Operational Research Society*, **50**, 876.
- (39) Morabito, R.; Morales, S.R. & Widmer, J.A. (2000). Loading Optimization of Palletized Products on Trucks. *Transportation Research*, **36**, 285-296.
- (40) Morabito, R. & Farago, R. (2002). A Tight Lagrangean Relaxation Bound for the Manufacturer's Pallet Loading Problem. *Studia Informatica Universalis*, **2**(1), 57-76.

- (41) Narciso, M.G. & Lorena, L.A.N. (1999). Lagrangean/surrogate Relaxation for Generalized Assignment Problems. *European Journal of Operational Research*, **114**(1), 165-177.
- (42) Nelissen, J. (1993). New Approaches to the Pallet Loading Problem. Working paper, RWTH Aachen. <ftp://ftp.informatik.rwth-aachen.de/pub/reports/pallet.ps.Z>.
- (43) Nelissen, J. (1994). Solving the Pallet Loading Problem More Efficiently. Working Paper, Graduiertenkolleg Informatik und Technik, Aachen.
- (44) Nelissen, J. (1995). How to Use Constraints to Compute an Upper Bound for the Pallet Loading Problem. *European Journal of Operational Research*, **84**, 662-680.
- (45) Nemhauser, G.L. & Wolsey, L.A. (1988). *Integer and Combinatorial Optimization*. John Wiley & Sons, Inc.
- (46) Novamad Pallets (2004). Disponível em: <www.lpnet.com.br/novamad/palletscom.htm>. Acesso em: 02/08/2004.
- (47) Oliveira, L.K. (2004). Métodos Exatos Baseados em Relaxação Lagrangiana e *Surrogate* para o Problema de Carregamento de Paletes do Produtor. Tese de Doutorado, Departamento de Engenharia de Produção, Universidade Federal de São Carlos.
- (48) Pureza, V. & Morabito, R. (2003). Uma Heurística de Busca Tabu Simples para o Problema de Carregamento de Paletes do Produtor. *Pesquisa Operacional*, **23**(2), 259-378.
- (49) Pureza, V. & Morabito R. (2006). Some Experiments with a Simple Tabu Search Algorithm for the Manufacturer's Pallet Loading Problem. *Computers & Operations Research*, **33**, 804-819.
- (50) Reeves, C.R. (1993). *Modern Heuristic Techniques for Combinatorial Problem*. Department of Statistics and Operational Research, School of Mathematical and Information Sciences, Coventry University.
- (51) Sarin, S.; Karvan, M.H. & Rardin, R.L. (1987). A New Surrogate Dual Multiplier Search Procedure. *Naval Research Procedure*, **34**, 431-450.
- (52) Scheithauer, G. & Terno, J. (1996). The G4-Heuristic for the Pallet Loading Problem. *Journal of the Operational Research Society*, **47**, 511-522.
- (53) Scheithauer, G. & Sommerweiss, G. (1998). 4-block Heuristic for the Rectangle Packing Problem. *European Journal of Operational Research*, **108**(3), 509-526.
- (54) Smith, A. & De Cani, P. (1980). An Algorithm to Optimize the Layout of Boxes in Pallets. *Journal of the Operational Research Society*, **31**, 573-578.
- (55) Sobral, F.H.A. (1996). *Técnicas de Unitização de Cargas em Paletes*. Associação Brasileira de Logística (ASLOG) – Comitê de Unitização de Cargas.
- (56) Steudel, H. (1979). Generating Pallet Loading Patterns: a Special Case of the Two-Dimensional Cutting Stock Problem. *Management Science*, **10**, 997-1004.

- (57) Tarnowski, A.G.; Terno, J. & Scheithauer, G. (1994). A Polynomial Time Algorithm for the Guillotine Pallet Loading Problem. *Information Systems and Operational Research*, **32**, 275-287.
- (58) Tsai, R.D.; Malstrom, E.M. & Kuo, W. (1993). Three Dimensional Palletization of Mixed Box Sizes. *IIE Transactions*, **25**(4), 64-75.