

UNIVERSIDADE FEDERAL DE SÃO CARLOS
CENTRO DE CIÊNCIAS EXATAS E DE TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE PRODUÇÃO

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

LILIAN KÁTIA DE OLIVEIRA

TESE DE DOUTORADO

UNIVERSIDADE FEDERAL DE SÃO CARLOS
CENTRO DE CIÊNCIAS EXATAS E DE TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE PRODUÇÃO

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

Lílian Kátia de Oliveira

Tese de Doutorado apresentada ao Programa de Pós-Graduação em Engenharia de Produção da Universidade Federal de São Carlos, como parte dos requisitos para a obtenção do título de Doutora em Engenharia de Produção.

Orientador: Prof. Dr. Reinaldo Morabito

Agência Financiadora: CNPq

SÃO CARLOS - SP

2004

**Ficha catalográfica elaborada pelo DePT da
Biblioteca Comunitária/UFSCar**

O48me

Oliveira, Lilian Kátia de.

Métodos exatos baseados em relaxação lagrangiana e surrogate para o problema de carregamento de paletes do produtor / Lilian Kátia de Oliveira. -- São Carlos : UFSCar, 2005.

184 p.

Tese (Doutorado) -- Universidade Federal de São Carlos, 2004.

1. Paletes. 2. Relaxação lagrangiana. 3. Relaxação surrogate. 4. Método Branch and Bound. I. Título.

CDD: 674.82 (20^a)

*A meu pai, que se estivesse entre nós,
teria muito orgulho dessa minha conquista.*

AGRADECIMENTOS

A Deus, primeiramente, por ter me guiado e dado forças em mais uma etapa de minha vida.

Ao Prof. Dr. Reinaldo Morabito pela orientação, confiança, dedicação e amizade construída durante o desenvolvimento desta pesquisa.

À minha família pelo amor, incentivo e presença em todos os anos de minha vida.

Ao Josemar que é meu companheiro, amigo e maior incentivador.

Aos amigos que fizeram e fazem parte de minha vida, mesmo os mais distantes e, em especial, aos amigos feitos em São Carlos.

Aos professores e funcionários do DEP-UFSCar pelo apoio.

Ao CNPq pelo apoio financeiro, sem o qual este trabalho não teria sido realizado.

E finalmente a todos que colaboraram indiretamente na realização deste trabalho.

SUMÁRIO

Lista de Figuras	iv
Lista de Tabelas	vi
Resumo.....	ix
Abstract	x
Capítulo 1. Introdução.....	1
Capítulo 2. Paletização de Carga.....	8
2.1. Introdução	8
2.2. Atividades da Logística.....	11
2.3. Unitização de Carga	13
2.4. Palete	14
2.4.1. Vantagens da Paletização	15
2.4.2. Tipos e Modelos de Paletes Existentes	16
2.5. O Problema de Carregamento de Paletes	21
2.5.1. Problema de Corte e Empacotamento.....	21
2.5.2. Problema de Carregamento de Paletes.....	21
2.5.2.1. Definição do Problema do Produtor	23
2.5.2.2. Definição do Problema do Distribuidor.....	28

Capítulo 3. Formulação Matemática.....	29
3.1. Introdução	29
3.2. Formulação de Beasley	30
3.2.1. Exemplo Numérico	32
3.3. Formulação de Tsai <i>et al.</i>	36
3.4. Formulação de Hadjiconstantinou e Christofides	40
3.5. Formulação de Chen <i>et al.</i>	44
3.6. Formulação não Linear de Beasley	47
3.7. Relação entre os Modelos de TSAI <i>et al.</i> e Chen <i>et al.</i>	50
Capítulo 4. Métodos de Solução.....	53
4.1. Introdução	53
4.2. Revisão da Literatura	54
4.3. Partições Eficientes e Classes de Equivalência.....	56
4.4. Alguns Limitantes Superiores para a Solução Ótima	59
4.4.1. Limitantes Simples	60
4.4.2. Limitantes de Barnes.....	60
4.4.3. Limitantes de Exemplos Equivalentes	62
4.4.4. Limitantes de Exemplos Relaxados.....	62
4.4.5. Limitante de Empacotamento	63
4.5. Relaxação de Programação Linear, Lagrangiana, <i>Surrogate</i>	64
4.5.1. Relaxação de Programação Linear.....	65
4.5.1.1. Aplicação de Relaxação por Programação Linear no PCP do Produtor ...	66
4.5.2. Relaxação Lagrangiana (LAG)	66
4.5.2.1. Aplicação de Relaxação Lagrangiana no PCP do Produtor.....	69
4.5.2.2. Otimização do Subgradiente para Relaxação Lagrangiana.....	72
4.5.3. Relaxação <i>Surrogate</i> (SUR).....	75
4.5.3.1. Aplicação de Relaxação <i>Surrogate</i> no PCP do Produtor	77
4.5.3.2. Otimização do Subgradiente para Relaxação <i>Surrogate</i>	81
4.5.4. Relaxação Lagrangiana <i>Surrogate</i> (LAGSUR)	82
4.5.4.1. Aplicação de Relaxação Lagrangiana <i>Surrogate</i> no PCP do Produtor	83
4.5.4.2. Otimização do Subgradiente para Relaxação Lagrangiana <i>Surrogate</i>	89

Capítulo 5. Busca em Árvore	90
5.1. Introdução	90
5.2. Heurística Lagrangiana e Redução do Problema	91
5.2.1. Heurística Lagrangiana	91
5.2.2. Redução do Problema	93
5.2.3. Procedimento de Otimização do Subgradiente para o Método LAG e LAGSUR	96
5.3. Heurística <i>Surrogate</i> e Redução do Problema	100
5.3.1. Procedimento de Otimização do Subgradiente para o Método SUR	101
5.4. Busca em Árvore.....	103
5.4.1. Regra para Seleção da Variável a ser Fixada.....	107
5.4.2. Pseudo-código do Método de Busca em Árvore	109
 Capítulo 6. Resultados Computacionais	 114
6.1. Introdução	114
6.2. Exemplos da Literatura e Reais.....	118
6.3. Resultados Utilizando o GAMS/CPLEX	125
6.4. Resultados Utilizando os Grupos 1 e 2	125
6.5. Resultados Utilizando o Grupo 3	133
6.6. Resultados Obtidos com os Exemplos Reais	138
6.7. Considerações Finais.....	141
 Capítulo 7. Conclusões e Propostas Futuras	 143
7.1. Conclusões	143
7.1. Propostas Futuras	148
 Referências Bibliográficas.....	 149
 Apêndices	 160

LISTA DE FIGURAS

Figura 2.1. Implantação de uma central de distribuição para diminuir os custos logísticos (FARAGO, 1999).....	10
Figura 2.2. Palete descartável.....	17
Figura 2.3. Palete de uso repetitivo.....	17
Figura 2.4. Palete de duas entradas.....	18
Figura 2.5. Palete de quatro entradas.....	18
Figura 2.6. Palete de face simples.....	19
Figura 2.7. Palete de face duplas.....	19
Figura 2.8. Palete reversível.....	19
Figura 2.9. Palete com abas.....	20
Figura 2.10. Caixa palete.....	20
Figura 2.11. Padrão de empacotamento para o problema de carregamento de paletes do (a) produtor e (b) do distribuidor.....	22
Figura 2.12. Orientações horizontais das caixas dentro do palete.....	23
Figura 2.13. (a) cortes ortogonais e (b) cortes não-ortogonais.....	24
Figura 2.14. (a) cortes guilhotinados e (b) cortes não-guilhotinados.....	25
Figura 2.15. Exemplo de padrão não-guilhotinado de 1ª ordem.....	25
Figura 2.16. Padrão de corte não-guilhotinado de ordem superior.....	26

Figura 2.17. Exemplo de padrão de empacotamento não-ortogonal.....	26
Figura 3.1. Posição (r,s) não permitida em função da colocação de uma caixa na posição (p,q) com orientação i	31
Figura 3.2. Exemplo de um palete $(L,W)=(5,4)$ e caixa $(l,w)=(3,2)$	33
Figura 3.3. Padrão de carregamento ótimo para o exemplo utilizando o modelo GAMS/CPLEX de BEASLEY (1985b).....	36
Figura 3.4. Localização do palete no eixo xy	38
Figura 3.5. Localização de uma caixa em um palete	41
Figura 3.6. Sobreposição de peças	48
Figura 4.1. <i>Layout</i> ótimo para o exemplo $(22,16,5,3)$, $(30,22,7,4)$, $(50,36,11,7)$ e todos os exemplos com a mesma classe de equivalência (MARTINS, 2002)	57
Figura 4.2. Exemplo da identificação de segmentos em um palete carregado (MARTINS, 2002)	58
Figura 5.1. Diagrama ilustrativo de uma árvore binária de busca	106
Figura 5.2. Divisão do palete em 5 blocos com orientação das caixas	112
Figura B.1. Padrão de carregamento ótimo para o exemplo utilizando o modelo GAMS/CPLEX de TSAI <i>et al.</i> (1993)	167
Figura C.1. Padrão de carregamento ótimo para o exemplo utilizando o modelo GAMS/CPLEX de HADJICONSTANTINOU e CHRISTOFIDES (1995)	173
Figura C.2. Padrão de carregamento ótimo para o exemplo utilizando o <i>software</i> CPLEX no modelo de AMARAL e LETCHFORD (2001)	174

LISTA DE TABELAS

Tabela 3.1. Valores binários e equações de restrições associadas	39
Tabela 6.1. Grupo 1A de exemplos para teste	120
Tabela 6.2. Grupo 1B de exemplos para teste.....	120
Tabela 6.3. Grupo 2A de exemplos para teste	121
Tabela 6.4. Grupo 2B de exemplos para teste.....	121
Tabela 6.5. Grupo 3A de exemplos para teste	122
Tabela 6.6. Grupo 3B de exemplos para teste.....	123
Tabela 6.7. Exemplos reais coletados na Transportadora Americana com (L, W)=(120,100) cm.....	124
Tabela 6.8. Soluções obtidas utilizando o pacote GAMS/CPLEX (versão 7).....	125
Tabela 6.9. Resultados computacionais obtidos utilizando $f=2$ e divisão de f a cada 60 iterações com o Grupo 1A.....	126
Tabela 6.10. Resultados computacionais obtidos com o método LAG utilizando $f=2$ e divisão de f a cada 60 iterações	127
Tabela 6.11. Resultados computacionais obtidos com o método LAG utilizando $f=2$ e divisão de f a cada 30 iterações	127
Tabela 6.12. Resultados computacionais obtidos com o método LAG utilizando $f=2$ e divisão de f a cada 15 iterações	128

Tabela 6.13. Resultados computacionais obtidos com o método LAG utilizando $f=2$ e divisão de f a cada 60 iterações se Z_{min} não diminuir nas últimas 60 iterações.....	128
Tabela 6.14. Resultados computacionais obtidos com o método LAG utilizando $f=2$ e divisão de f a cada 30 iterações se Z_{min} não diminuir nas últimas 30 iterações.....	128
Tabela 6.15. Resultados computacionais obtidos com o método LAG utilizando $f=2$ e divisão de f a cada 15 iterações se Z_{min} não diminuir nas últimas 15 iterações.....	128
Tabela 6.16. Resultados computacionais obtidos com o método LAG utilizando $f=1$ e divisão de f a cada 30 iterações	129
Tabela 6.17. Resultados computacionais obtidos com o método LAG utilizando CAMERINI <i>et al.</i> (1975).....	130
Tabela 6.18. Resultados computacionais obtidos com o método LAG utilizando os <i>raster points</i>	130
Tabela 6.19. Resultados computacionais obtidos com o método LAG utilizando os <i>raster points</i> com a ramificação dos nós usando a idéia de memória de longo prazo de Busca Tabu	131
Tabela 6.20. Resultados computacionais obtidos com o método LAG utilizando os <i>raster points</i> com a heurística de MORABITO e MORALES (1998).....	132
Tabela 6.21. Resultados computacionais obtidos com o método LAG utilizando os <i>raster points</i> com a heurística de MORABITO e MORALES (1998) para o conjunto Cover I.....	133
Tabela 6.22. Resultados computacionais obtidos com o método LAG utilizando $f=1$ e a divisão de f a cada 30 iterações	133
Tabela 6.23. Resultados computacionais obtidos com o método LAG utilizando CAMERINI <i>et al.</i> (1975).....	134
Tabela 6.24. Resultados computacionais obtidos com o método LAG utilizando os <i>raster points</i>	134
Tabela 6.25. Resultados computacionais obtidos com o método LAG utilizando os <i>raster points</i> com a ramificação dos nós usando a idéia de memória de longo prazo de Busca Tabu	135
Tabela 6.26. Resultados computacionais obtidos com o método LAG utilizando os <i>raster points</i> com a heurística de MORABITO e MORALES (1998).....	135

Tabela 6.27. Resultados computacionais obtidos em LINS <i>et al.</i> (2003) e com o método LAG.....	136
Tabela 6.28. Resultados computacionais obtidos com o método LAG utilizando os <i>raster points</i> com a heurística de MORABITO e MORALES (1998) para o conjunto Cover II.....	137
Tabela 6.29. Resultados computacionais do Grupo 4 obtidos com o método LAG com a divisão de f a cada 30 iterações com os <i>raster points</i>	139
Tabela 6.30. Resultados computacionais do Grupo 4 obtidos com o método LAG com a divisão de f a cada 30 iterações com os <i>raster points</i> e a heurística de MORABITO e MORALES (1998)	140
Tabela 6.31. Resultados computacionais obtidos com o método LAG utilizando os <i>raster points</i> com a heurística de MORABITO e MORALES (1998).....	142

RESUMO

O objetivo deste trabalho é desenvolver métodos exatos, baseados em relaxação Lagrangiana e *Surrogate*, com bom desempenho para resolver o problema de carregamento de paletes do produtor. Tal 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) . Tais métodos exatos 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* na otimização do subgradiente para obter boas soluções factíveis. Testes computacionais foram realizados utilizando exemplos da literatura e exemplos reais, obtidos de uma transportadora. Os resultados mostram que os métodos são capazes de resolvê-los, em média, mais rapidamente do que outros métodos exatos, incluindo o *software* GAMS/CPLEX.

Palavras-chave: Problema de Carregamento de Paletes do Produtor, Relaxação Lagrangiana, Relaxação *Surrogate*, Método *Branch and Bound*.

ABSTRACT

The purpose of this work is to develop exact methods, based on Lagrangean and Surrogate relaxation, 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) and (w,l) into a larger rectangle (L,W) without overlapping. Such methods involve a tree search procedure of branch and bound type and they use, in each node of the branch and bound tree, bounds 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 in the subgradient optimization to obtain good feasible solution. Computational experiments were performed with instances from the literature and also real instances obtained from a carrier. The results show that the methods are able to solve these instances, on average, more quickly than other exact methods, including the software GAMS/CPLEX.

Key-words: Manufacturer's Pallet Loading Problem, Lagrangean Relaxation, Surrogate Relaxation, Branch and Bound Method.

CAPÍTULO 1

INTRODUÇÃO

Basicamente a logística tenta resolver o problema da distância e do tempo entre a produção e a demanda, de modo a atender os clientes de maneira satisfatória no local e ocasião desejados. Envolve uma série de atividades que podemos dividir naquelas de importância primária e de apoio, quando tratamos de nível de serviço e custo (BALLOU, 1993). São relevantes para este trabalho, as atividades de transporte (atividade primária); manuseio de materiais e embalagem de proteção (atividades de apoio).

SULE (1988) define carga unitizada como vários itens arranjados que podem ser manuseados em um único volume. A paletização de carga tem como característica reunir vários itens em uma plataforma, em geral de madeira, disposta horizontalmente, a fim de que a carga possa ser empilhada e estabilizada (MORALES *et al.*, 1997). Dessa forma há um melhor controle de estoque, a carga fica protegida contra roubos e danos físicos e pode ser transportada de uma só vez por meio de empilhadeiras, guindastes, caminhões, diminuindo o tempo de carga e descarga (UELZE, 1974).

Segundo um estudo feito pelo Grupo Palete de Distribuição, criado pela Associação Brasileira de Supermercados (ABRAS), em 1987 havia mais de mil modelos de paletes diferentes no mercado brasileiro. Após este estudo, definiu-se o

palete padrão brasileiro, o PBR (Palete Brasil) cujas especificações são: palete face dupla não-reversível, 4 entradas, de madeira e com dimensões 1200x1000 milímetros (MORALES, 1995).

Diversos autores têm observado que a unitização de carga está sendo cada vez mais utilizada na distribuição e armazenagem de produtos. No Brasil, assim como em outros países, há uma crescente tendência a paletização de carga ao longo de toda a cadeia logística, principalmente em empresas atacadistas e do setor de alimentos. Em particular, com as perspectivas de padronização de paletes e ainda a consolidação do Mercosul, um problema que recebe especial atenção por parte de produtores e transportadores é o problema da escolha e do carregamento de paletes.

Basicamente, o problema do carregamento de paletes consiste em carregar produtos (embalados em caixas) sobre um palete retangular, de maneira a otimizar o aproveitamento do palete. Admite-se que as caixas, em geral disponíveis em grandes quantidades, devem ser arranjadas ortogonalmente, isto é, com seus lados paralelos aos lados do 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.

O problema de carregamento de paletes pode ser visto como um caso especial dos problemas de corte e empacotamento. Empacotar unidades (por exemplo, caixas) sobre unidades maiores (paletes) é similar a cortar unidades (por exemplo, chapas) para produzir unidades menores (itens). Centenas de artigos foram publicados na literatura de pesquisa operacional e gestão da produção tratando os problemas de corte e empacotamento, conforme indicam os artigos de exames e edições especiais em DOWSLAND e DOWSLAND (1992), DYCKHOFF e FINKE (1992), SWEENEY e PATERNOSTER (1992), MARTELLO (1994a, 1995b), BISCHOFF e WAESCHER (1995), MUKHACHEVA (1997), DYCKHOFF *et al.* (1997), ARENALES *et al.* (1999), WANG e WAESCHER (2002), HIFI (2002) e SICUP (2004). Curiosamente, até o final da década de 80, poucos autores haviam de fato evidenciado (e explorado) a similaridade entre os problemas de corte e empacotamento. A partir de DYCKHOFF (1990), tais problemas passaram a ser tratados como uma mesma classe, com estrutura lógica similar a de outros problemas da literatura (MORABITO e ARENALES, 1992).

HODGSON (1982), ao estudar o problema de carregamento de paletes, dividiu-o em dois possíveis casos: o problema de carregamento de paletes do produtor (*manufacturer's pallet loading problem*), onde carregamos o palete somente com caixas idênticas; e o problema de carregamento de paletes do distribuidor (*distributor's pallet loading problem*), onde acomodamos sobre o palete caixas de diversas dimensões.

Neste trabalho estudamos o problema de carregamento de paletes do produtor (PCP do produtor). DYCKHOFF (1990) apresentou uma tipologia fundamentada na estrutura lógica dos problemas de corte e empacotamento. Assim, o PCP do produtor é classificado com a seguinte notação: 2/B/O/C que representa um problema bidimensional (2), tendo como forma de alocação a seleção de unidades pequenas (B) (do alemão *Beladeproblem*), com uma única unidade grande (palete) (O) e unidades pequenas (caixas) de tamanhos iguais (C). Essa notação pode ser utilizada, na literatura de corte e empacotamento, para buscar métodos de solução que sejam específicos para resolver a classe de problemas 2/B/O/C.

O PCP do produtor é importante do ponto de vista da complexidade computacional, pois aparentemente simples de ser resolvido otimamente, o problema não é resolvido em tempo polinomial, embora ainda não seja provado pertencer à classe NP-difícil. Enquanto que o PCP guilhotinado é resolvido em tempo polinomial $O(\log_2 L(\log_2 l)^2)$, como mostrado por TARNOWSKI *et al.* (1994), nenhum resultado similar está disponível para o caso não-guilhotinado (LETCHFORD e AMARAL, 2001). Este problema pode ser modelado como um programa inteiro 0–1 (MORALES e MORABITO, 1997; FARAGO e MORABITO, 2000), o qual é um caso particular do modelo proposto em BEASLEY (1985b), ou como em HADJICONSTANTINOU e CHRISTOFIDES (1995) para o problema de corte bidimensional não-guilhotinado da classe 2/B/O/M (bidimensional, seleção de itens, único objeto, vários itens diferentes). AMARAL e LETCHFORD (2001) mostraram que o modelo em HADJICONSTANTINOU e CHRISTOFIDES (1995) nem sempre é válido. O PCP do produtor também pode ser modelado com outros problemas inteiros conforme TSAI *et al.* (1993), CHEN *et al.* (1995) e MARTINS (2002).

Poucos autores propuseram métodos exatos para resolver o PCP do produtor, entre eles, DOWSLAND (1987), TSAI *et al.* (1993) e BATTACHARYA *et al.* (1998). Tais métodos foram testados em limitados experimentos computacionais.

MARTINS (2002) mostrou que o método em BATTACHARYA *et al.* (1998) não é exato. 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 e DE CANI (1980), BISCHOFF e DOWSLAND (1982), DOWSLAND (1993), NELISSEN (1994, 1995), ARENALES e MORABITO (1995), SCHEITHAUER e TERNO (1996), HERBERT e DOWSLAND (1996), SCHEITHAUER e SOMMERWEISS (1998), MORABITO e MORALES (1998, 1999), G e KANG (2001), MARTINS (2002), LINS *et al.* (2003), PUREZA e MORABITO (2004), ALVAREZ-VALDES *et al.* (2004) e BIRGIN *et al.* (2004). Limitantes superiores para o problema são estudados em NELISSEN (1995), LETCHFORD e AMARAL (2001) e MORABITO e FARAGO (2002).

Os principais objetivos deste trabalho são revisar o estado da arte do PCP do produtor, desenvolver métodos exatos, baseados em relaxação Lagrangiana e *Surrogate*, com bom desempenho para resolver este problema (similarmente ao que foi feito em BEASLEY (1985b) e HADJICONSTANTINOU e CHRISTOFIDES (1995) para o problema mais geral de corte 2/B/O/M) e, avaliar em uma situação prática, o desempenho dos métodos desenvolvidos em relação a outros métodos exatos. É importante observar que o trabalho de BEASLEY (1985b) analisa somente o uso de relaxação Lagrangiana combinada com relaxação *Surrogate* (LAGSUR) e não reporta resultados utilizando apenas a relaxação Lagrangiana (LAG) ou apenas relaxação *Surrogate* (SUR). Além disso, o trabalho de BEASLEY (1985b) trata do problema de corte mais geral, os limitantes obtidos para este 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 e CHRISTOFIDES, 1995). Uma motivação para o presente trabalho é avaliar o desempenho de métodos de relaxação Lagrangiana e/ou *Surrogate* (LAG, SUR, LAGSUR) aplicados ao problema do PCP do produtor.

De acordo com os objetivos propostos, para o desenvolvimento do trabalho foram revisados alguns modelos encontrados na literatura para descrever o PCP do produtor, em seguida, foi escolhido o modelo a ser usado e a partir daí, deu-se início a aplicação das relaxações Lagrangiana e/ou *Surrogate* em um procedimento de otimização do subgradiente para otimizar estes limitantes superiores. 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 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 obtidos em uma empresa transportadora.

Uma das justificativas para o desenvolvimento deste projeto é a importância técnica e econômica das aplicações do PCP do produtor nos processos industriais e logísticos. No nível estratégico, uma análise estatística baseada na solução do PCP do produtor pode ser útil para escolher o palete mais adequado para uma empresa (dentro um conjunto de paletes padronizados disponíveis), assim como para projetar novos paletes (MORALES *et al.*, 1997). Com um programa computacional eficaz, a empresa pode, por exemplo, simular padrões de empacotamento para um conjunto de produtos e paletes, e a partir dos resultados obtidos, escolher o palete mais adequado. Outra aplicação do PCP do produtor no nível estratégico, refere-se ao desenvolvimento de novas embalagens de produtos, que depende de fatores logísticos e mercadológicos. Neste caso, a metodologia pode ser útil para analisar o arranjo físico das embalagens dentro de uma unidade de carga intermediária (DOWSLAND, 1995).

No nível operacional, métodos mais efetivos para resolver o PCP do produtor permitem encontrar melhores padrões de empacotamento para a armazenagem e distribuição de um produto. Além disso, com um programa computacional eficaz, a geração de padrões para cada produto, pode ser desempenhada com maior rapidez e flexibilidade em cada centro de distribuição, junto com o pessoal encarregado da montagem da carga paletizada. Ao melhorar a iteração entre o processo de geração de padrões de empacotamento e a montagem física da carga, melhoramos o desempenho de atividades como testes de estabilidade da carga e análise de padrões alternativos.

O PCP do produtor também tem outras aplicações ao longo da cadeia logística. É crescente o número de empresas que buscam, cada vez mais, inserir o palete no segmento de transporte rodoviário, com o objetivo de reduzir custos e tempos de carga e descarga nos terminais. Assim, além do problema de carregar caixas sobre o

palete, o PCP do produtor também pode ser aplicado para carregar paletes sobre a carroceria de caminhões ou vagões ferroviários (MORABITO *et al.*, 2000).

A organização do trabalho está descrita a seguir. No Capítulo 2, discutimos a evolução da logística desde a era do artesanato até a era da sistemofatura, assim como a importância da unitização de carga dentro do contexto da logística moderna. Conceituamos unitização de carga, apresentamos os tipos e modelos de paletes assim como as vantagens de se paletizar cargas. Ainda, é discutido o problema de carregamento de paletes tanto do produtor como do distribuidor.

No Capítulo 3 apresentamos modelos matemáticos para o PCP do produtor, adaptados dos modelos em BEASLEY (1985b), TSAI *et al.* (1993), HADJICONSTANTINOU e CHRISTOFIDES (1995), CHEN *et al.* (1995) e BEASLEY (2004). Em particular, mostramos que os modelos de TSAI *et al.* (1993) e CHEN *et al.* (1995) estão bem relacionados e observamos que o modelo de HADJICONSTANTINOU e CHRISTOFIDES (1995) nem sempre é válido (por meio de um contra exemplo apresentado em LETCHFORD e AMARAL, 2001).

No Capítulo 4 apresentamos uma revisão da literatura apresentando os trabalhos que tratam do PCP do produtor. Ainda, revisamos alguns limitantes superiores para a solução ótima encontrados na literatura para este problema, pois tais limitantes são úteis em heurísticas para verificar a otimalidade, e em algoritmos exatos, pois permitem o uso de limitantes na redução do espaço de busca da solução. São apresentadas também as relaxações Linear, Lagrangiana, *Surrogate* e Lagrangiana *Surrogate*, aplicadas a uma das formulações do Capítulo 3 do PCP do produtor.

No Capítulo 5 propomos métodos de solução exatos para a resolução do PCP do produtor baseado na aplicação das técnicas de relaxação Lagrangiana e/ou *Surrogate* discutidas no Capítulo 4: os métodos LAG, SUR e LAGSUR. Tais métodos utilizam uma árvore binária com busca em profundidade e, em cada nó da árvore, é computado um limitante superior dependendo da relaxação (LAG, SUR, LAGSUR). São aplicados ainda testes de redução do problema e heurísticas Lagrangiana e *Surrogate* na otimização do subgradiente. Apresentamos também algumas regras de seleção da variável a ser fixada na divisão dos nós, para a inicialização dos multiplicadores de Lagrange em cada nó da árvore, e para a redução de parâmetros utilizados no procedimento de otimização do subgradiente. Apresentamos ainda, os

pseudo-códigos dos procedimentos de otimização do subgradiente com os métodos propostos e da busca em árvore.

No Capítulo 6 são apresentados os resultados computacionais obtidos com os métodos propostos. Comparamos o valor das soluções e os tempos computacionais encontrados pelos métodos com as soluções encontradas por outros métodos e pelo *software* GAMS/CPLEX (BROOKE *et al.*, 1992). Mostramos que os métodos propostos são competitivos e capazes de gerar resultados médios melhores que outros métodos. E, finalmente, no Capítulo 7 são apresentadas as conclusões e sugestões para pesquisas futuras.

CAPÍTULO 2

PALETIZAÇÃO DE CARGA

2.1. INTRODUÇÃO

Dentro da logística empresarial temos diversas atividades que geram custos e que são extremamente necessárias na empresa. Assim, quando falamos dos modos de organização da produção observamos mudanças nessas atividades no decorrer do tempo, desde a era do artesanato até a era da sistemofatura, esta última marcada por uma maior flexibilidade e diminuição de estoques (HOFFMAN e KAPLINSKY, 1988). Por vários séculos, os instrumentos logísticos vêm evoluindo no sentido de tornarem possíveis as mudanças necessárias nas economias. Se analisarmos a partir das formas primitivas e agrárias de se produzir, observamos que as economias modernas são bem mais flexíveis no que diz respeito às mudanças na tecnologia. Com todas estas transformações tecnológicas, foi necessário um aumento na capacidade de distribuição física e comercialização de produtos, visando o progresso do país. Um país que não tivesse investido o mínimo nos instrumentos logísticos passaria a viver isolado do mundo, tendo seu progresso fortemente comprometido (FARAGO, 1999).

Portanto, é interessante tratar da evolução de algumas dessas atividades logísticas, apresentando seus conceitos e importâncias. Em particular, damos uma maior atenção àquelas relacionadas ao problema de carregamento de paletes, tais como

atividades de transporte, manuseio de materiais e embalagens de proteção. A área de logística é um dos setores da indústria que mais sofre modificações. No decorrer dessas mudanças, percebemos uma crítica intensa travada contra o desconhecimento, os métodos ultrapassados e a ineficiência. Isso faz com que os homens da administração procurem outros meios para não perderem a competitividade, uma vez que este passou a ser um dos principais objetivos das empresas de hoje (MAGEE, 1977; BALLOU, 1999).

Avanços nos transportes e na tecnologia de manipulação de materiais propiciaram uma maior flexibilidade à empresa, conseguindo administrar de maneira satisfatória, o fluxo de materiais e produtos da fonte ao cliente. Poucas décadas depois da Revolução Industrial, homens que tinham grande poder de administração e criatividade deram início a uma revolução na produção. Deste modo, foi possível produzir em massa produtos com preços bem mais baixos e que precisavam anteriormente de muitas horas/homem de trabalho pesado (FARAGO, 1999). Produzindo mais houve então a necessidade de estocar em maior quantidade esses produtos. Como aumentou a demanda por esses produtos e estes não estavam no mesmo local onde eram confeccionados, estava estabelecido então o problema de como transportá-los desde a fábrica até esses clientes potenciais, a um custo que fosse o mínimo para a empresa, com a máxima satisfação do cliente (HOFFMAN e KAPLINSKY, 1988).

Com o objetivo de diminuir os custos logísticos, a mercadoria pode ser estocada uma única vez, pois os custos de estocagem comumente representam de um a dois terços dos custos logísticos totais da empresa (BALLOU, 1993, 1999). Uma maneira que se pensou para estocar uma única vez foi a implantação de armazéns de distribuição, intermediários à empresa e ao cliente, como mostra a Figura 2.1:

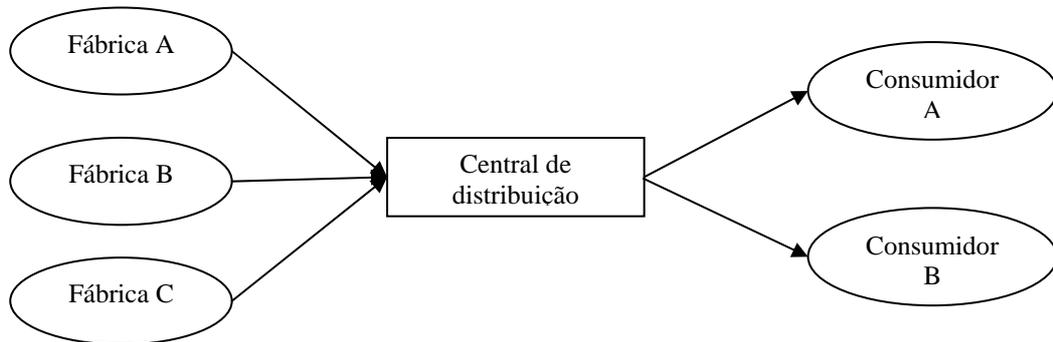


Figura 2.1. Implantação de uma central de distribuição para diminuir os custos logísticos (FARAGO, 1999).

É importante ter conhecimento que os custos logísticos correspondem à soma dos custos de transporte, de manuseio de materiais, de armazenagem, de obsolescência e do capital empregado (BALLOU, 1999; SHAPIRO, 2001).

O problema é mais complexo do que aparenta, ou seja, deve-se encontrar uma forma de combinar estes custos de maneira a minimizar os custos logísticos. Não podemos simplesmente pensar numa única forma de transporte, devemos pensar no tempo em que a mercadoria será transportada, as disponibilidades de espaço, embalagem apropriada. Ilustra-se assim a complexidade do problema (BALLOU, 1993). Uma vez que o custo de manter os produtos na cadeia logística ou em trânsito, pode ser bem alto, quanto mais rápido estas mercadorias se deslocarem de um ponto a outro, menor será o capital investido nessa cadeia. Uma das principais características que observamos na maioria das empresas na era da sistemofatura é que as indústrias estão com um nível de estoque cada vez mais baixo em seus depósitos, enquanto os níveis de produção são os mais altos. Isso se deve ao fato de que o custo de manter a mercadoria em estoque é em geral muito alto, frente aos outros da cadeia logística, já que abrange impostos associados às mercadorias guardadas em armazéns, seguros contra incêndios e roubo, custos de capital, pagamentos de depósitos públicos, riscos e deterioração, obsolescência e danos (UELZE, 1974).

A logística empresarial, em suma, estuda a melhor forma dos administradores fornecerem melhor nível de rentabilidade nos serviços de distribuição a todos os seus clientes. Tal nível é conseguido por intermédio de planejamentos, organização e controle efetivos para as diversas atividades de movimentação e

armazenagem que visam facilitar o fluxo dos produtos ao longo da cadeia logística. Procura resolver o problema da distância entre a produção e a demanda, de modo que possa atender os clientes de maneira satisfatória, quando e onde quiserem.

Na escala mundial da economia, o emprego de um bom sistema logístico tem outro papel fundamental: permite que empresas de certas regiões que dominam uma determinada produção possam exportar seus produtos a outras regiões que não são tão especializadas nesse mesmo ramo de negócio, seja por falta de infra-estrutura, clima adequado, entre outros. Com isso, a empresa aumenta significativamente seus lucros e abastece todo o território carente daquela mercadoria. Em outras palavras, se o sistema logístico é eficaz, então em meio a toda essa exportação, é possível se diminuir alguns custos logísticos e de produção, para que sejam competitivos com o mercado externo. O que levou os administradores a refletirem sobre os custos logísticos foram o aumento da competitividade mundial, a falta de recursos em determinadas regiões e a atratividade da mão-de-obra no Terceiro Mundo.

2.2. ATIVIDADES DA LOGÍSTICA

Quando falamos em transporte, estamos englobando não só vários tipos de equipamentos com os quais pessoas, bens e serviços são movimentados de um local para o outro, mas também a rede física onde esses movimentos acontecem. O transporte é um componente vital na administração do sistema logístico (BALLOU, 1999). Conhecemos até hoje cinco tipos de transportes de bens: ferroviário, rodoviário, aeroviário, hidroviário e dutoviário. O transporte é avaliado por medidas de custo e desempenho: custo direto dos serviços, tempo médio para entrega, variação no tempo de entrega e perdas e danos. Todo profissional que atua na área de logística tem um interesse crescente nos transportes: a empresa pode ter seu próprio transporte, como também contrata esse serviço de terceiros.

A manipulação de materiais é um dos principais geradores de custo de cadeia logística e não podemos evitá-lo. Sabemos que nem sempre é viável produzir nos locais onde os clientes estão. A manipulação de materiais nada mais é que um transporte de poucas mercadorias a uma distância relativamente pequena. A princípio pensamos somente em gastos quando falamos de manuseio de materiais e embalagens de proteção.

Porém, a diminuição dos gastos com perdas e danos na empresa, pode significar economias que, no balanço final da firma, diminuem consideravelmente o custo total do transporte de mercadoria. Além disso, a entrega de mercadoria em mal estado ou em volumes de difícil manuseio pode levar o cliente a perder a atração e motivação para comprar dessa empresa, que não se preocupa em facilitar o manuseio da carga, nem entregá-la sem danos físicos. Métodos e equipamentos para manuseio de materiais evoluíram muito nos últimos tempos se comparados com os de algumas décadas atrás. Os equipamentos de movimentação abrangem empilhadeiras e tratores, transportadores, esteiras e guinchos. São usados ainda equipamentos auxiliares como estantes e caixas reutilizáveis. Todos eles são combinados para propiciarem um melhor aproveitamento do volume do depósito.

A embalagem de proteção serve tanto para divulgação do produto como também para o seu próprio uso. Serve para proteção dos produtos contra danos, deterioração e aumenta a eficiência do transporte da mercadoria. Um profissional da área de embalagens de proteção deve desenvolver um determinado tipo de embalagem, considerando o intervalo de tempo em que será mantido e onde será exposto o produto. Faz isso através de simulações, testes em laboratórios, tais como testes de queda, vibrações, compressões e impacto.

A maneira de facilitar o manuseio e dificultar roubos, especialmente com mercadorias pequenas e de grande valor, é colocar diversas embalagens menores dentro de uma maior. As características de resistência, tamanho e configuração ditam o tipo de equipamento de movimentação e armazenagem que será usado. Para efeitos práticos, um pacote deve ser movimentado uma única vez. É possível conseguir economias quando utilizamos a unitização de carga, e esta é obtida com a utilização de contêineres, paletes, entre outros. Neste trabalho temos especial interesse nos paletes.

Anteriormente se pensava em unitizar carga com a ajuda do palete para estabilizar a carga. Contudo, hoje em dia, percebemos que o interesse está além da amarração da carga. Empresas continuam usando os paletes mesmo tendo resolvido o problema da estabilidade, já que usam o chamado filme ou *stretch* passado por um robô que circunda o palete. Com a perspectiva de consolidação do Mercosul, iniciou-se a tentativa de padronizar uma medida para os paletes, resultando em um interesse crescente nesse assunto. O trabalho de DOWSLAND (1990) trata do problema de

minimizar o número de movimentos que um robô realiza para carregar ou descarregar o palete e, dessa forma, o equipamento de paletização automático se torna mais difundido.

O problema tratado neste trabalho consiste em arranjar o maior número possível de caixas sobre a superfície do palete, de modo a otimizar a sua capacidade. Ao aumentarmos as quantidades de caixas em uma camada, o incremento pode ser bem substancial se considerarmos todas as camadas do palete, e em última escala, se considerarmos o transporte e armazenagem ao longo da cadeia logística. Aumentando o número de caixas sobre o palete, temos uma diminuição no volume ocupado do armazém e uma diminuição relativa dos custos de transporte e manuseio de materiais.

2.3. UNITIZAÇÃO DE CARGA

O trabalho de carregar, descarregar e movimentar materiais é um esforço que toda empresa deve realizar durante suas atividades. Porém, tratam-se de operações que, além de não agregarem qualquer valor ao produto, possuem custos elevados, sendo um fator decisivo na competitividade das empresas. Dentro deste enfoque, surge uma alternativa para buscar a minimização destes custos: a unitização de cargas. Este tipo de abordagem vem sendo empregado por várias empresas, que além de obterem maior lucro, ganham em segurança e em facilidade de manuseio de produtos.

Podemos dizer então que o conceito de unitizar carga resulta do fato de que é mais fácil e econômico movimentar itens e materiais em grupos do que individualmente. A unitização de carga para o transporte envolve os conceitos de logística e de embalagens de transporte. A logística estuda formas de suprir e distribuir mercadorias. Tem na unitização de cargas aliada a movimentação mecanizada um forte potencial de produtividade e eficácia. SULE (1988) define uma carga unitizada como sendo vários itens arranjados que podem ser manuseados em um único volume. Isto é possível através da paletização e da containerização. Com a unitização de carga o palete se torna uma carga compacta, já que além de reunir os produtos, utiliza materiais auxiliares para empacotá-los e envolvê-los. A carga unitizada pode ser movimentada pelos mesmos dispositivos de transporte de paletização, dependendo apenas dos pesos e tamanhos dos produtos.

A containerização é a reunião de itens em uma grande caixa, em geral de dimensão e formatos padronizados. Diferentemente da paletização, devemos considerar as três dimensões para otimizar o espaço do contêiner. Em geral, o transporte dos contêineres se dá com o uso de caminhões, trens, navios ou aviões. O tipo adequado de unitização de carga depende da situação em questão. Por exemplo, usamos mais os paletes quando estamos querendo empilhar itens similares. Isso não impede carregar um palete com itens diferentes, entretanto em geral é aconselhável, neste caso, transportar estes itens usando um contêiner.

Podemos dizer que entre as formas de unitização de cargas para o transporte no mundo, o sistema paletizado é aquele que tem demonstrado o maior crescimento. O conhecimento da logística aplicada à distribuição física dos produtos reconhece no palete uma alternativa eficiente de unitização, a qual facilita a movimentação, o transporte e a armazenagem das cargas. Apesar da obviedade da vantagem do sistema paletizado sobre a movimentação manual, os dois sistemas convivem simultaneamente no Brasil. O motivo é o custo da implantação do sistema versus o custo de manutenção do convencional. Não existem estudos sobre quando a movimentação paletizada é possível, ou quanto de carga a ser movimentada é necessária para a implantação de um sistema unitizado (INTERLOGIS, 2004).

2.4. PALETE

Um palete é uma plataforma, geralmente feita de madeira, disposta horizontalmente, onde a carga é empilhada e estabilizada e cuja movimentação é feita por meio de garfos. É construído de maneira que os garfos das empilhadeiras possam entrar por dois ou quatro de seus lados. É usado pela maioria das empresas para armazenar e transportar bens e tem como função a movimentação de grande escala. A palavra palete é uma adaptação do termo em inglês *pallet* (BUARQUE DE HOLANDA, 1986). Várias empresas distribuem os produtos fisicamente como, por exemplo, a automotiva, a química, a de produtos de consumo, supermercados, a de produção de alimentos, a de produtos de papel e agropecuários, varejo, a de aço e metais industriais, empresas transportadoras, etc..

Os paletes passaram a ser muito utilizados após a II Guerra Mundial e hoje se apresentam em vários modelos e tamanhos. Embora os primeiros paletes tenham sido confeccionados em madeira, podemos encontrar materiais como aço, plástico, papelão, fibra de madeira moldada e outras que satisfazem um nicho de mercado menor. Entretanto, os paletes de madeira têm sido tradicionalmente a base para o projeto de prateleiras de armazéns, de áreas de estocagem em armazéns, de empilhadeiras, docas e contêineres usados para embarcar bens (SOBRAL, 1996). Os paletes são selecionados de acordo com o peso e o tamanho da carga unitizada e do equipamento utilizado para movimentá-los. Além disso, a seleção de paletes também depende do seu custo, do tamanho de portas, passagens e outros espaços.

A seguir, serão apresentadas algumas vantagens da paletização de cargas e, em seguida, serão apresentados os tipos e modelos de paletes existentes.

2.4.1. VANTAGENS DA PALETIZAÇÃO

A implantação de um Programa de Paletização adequado pode fornecer aos produtores e distribuidores várias vantagens. Ou seja, de uma forma geral os atacadistas e as empresas de armazenagem que recebem produtos e mercadorias, durante a operação normal ou de movimentação de materiais, estão notando que métodos de movimentação mecânica adotados, repetidamente, confirmam a validade de potenciais ganhos prontamente disponíveis para melhores negócios da empresa.

A NWPMA (*National Wooden Pallet & Container Association*) norte-americana aponta dezesseis vantagens de um programa de paletização (SOBRAL, 1996; NOVAMAD PALLETS, 2004; INTERLOGIS, 2004):

1. A melhor utilização dos espaços verticais da empresa, pois a paletização possibilita uma quantidade maior de armazenagem em uma dada área, ou seja, uma maior cubagem de carga armazenada por área;
2. Redução de acidentes pessoais na substituição da movimentação manual pela movimentação mecânica;
3. Economia de 40% a 45% no custo da movimentação;
4. O furto é reduzido quando itens individuais são unitizados por cintas ou filmes (*stretch*);
5. Ventilação entre as mercadorias tanto nos depósitos como durante o transporte;

6. Simplificação no controle de inventário;
7. Eliminação quase total de danos aos produtos;
8. Redução do tempo de rotulagem e de despesas operacionais dessa tarefa. Um ou dois rótulos por unidade de carga elimina a necessidade de identificar cada embalagem do produto;
9. Diminuição do tempo de movimentação. Economia por homens/horas na eliminação dos sobretempos ociosos;
10. Uniformização do local de estocagem, resultando em áreas com aproveitamento otimizado;
11. Os paletes são a forma natural de subpisos para o qual cintas de aço podem ser usadas facilmente na ancoragem segura das mercadorias;
12. Eliminação freqüente de custos de transporte; possibilidade de distribuição, cargas e descargas dentro de qualquer ponto acessível por equipamento de movimentação;
13. Utilização na linha de produção elimina interrupções e gargalos e proporciona maior produtividade;
14. Redução de mão-de-obra para movimentação de carga;
15. Redução pela metade do tempo de carga e descarga de caminhões acarretando na eliminação de taxas de estadias; grande incremento na utilização dos equipamentos de embarque e desembarque; incrementos nos ganhos de frete para os transportadores, tendo em vista o maior tempo de unitização dos veículos rodando;
16. Com a redução no tempo de carga e descarga, verifica-se um rápido carregamento e descarregamento de um navio e, conseqüentemente, redução de custos.

A seguir, serão apresentados os tipos e modelos de paletes existentes.

2.4.2. TIPOS E MODELOS DE PALETES EXISTENTES

De acordo com a Norma de Terminologia para paletes, NBR 8254 editada em novembro de 1983, os tipos de paletes são classificados da seguinte forma (FARAGO, 1999; NOVAMAD PALLETS, 2004; INTERLOGIS, 2004):

- Quanto ao tipo de operação:
 - Paletes descartáveis: são aqueles destinados a uma única operação de transporte e/ou armazenamento (Figura 2.2);

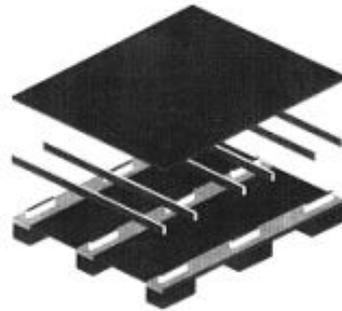


Figura 2.2. Paleta descartável.

- Paletes de uso repetitivo: são aqueles destinados a várias operações de transporte e/ou armazenamento (Figura 2.3);



Figura 2.3. Paletes de uso repetitivo.

- Paletes sem retorno (*One-way*): são aqueles descartáveis ou de usos repetitivo, porém que não retornam necessariamente à origem da operação de transporte.
- Quanto ao modelo:
 - Paletes de duas entradas: são aqueles que permitem a introdução de garfo somente por dois lados opostos (Figura 2.4). Geralmente são utilizados quando a movimentação ou a armazenagem da carga é restrita pelo equipamento utilizado ou pelo tipo de operação pretendida. Por exemplo, quando for utilizado somente um tipo de equipamento de movimentação na operação; ou o paleta for utilizado apenas na armazenagem interna ou quando o paleta tiver uma única viagem ou um único manuseio.

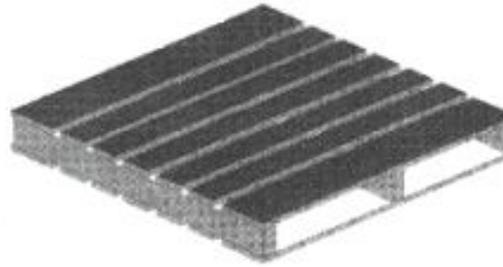


Figura 2.4. Palete de duas entradas.

- Paletes de quatro entradas: são os que permitem a introdução de garfos pelos quatro lados (Figura 2.5). Uma das vantagens do uso deste tipo de palete é o fato de possibilitar dois tipos de equipamentos de movimentação: a paleteira e a empilhadeira. Muitas vezes a carga que é descarregada do caminhão com uma empilhadeira precisa ser movimentada no interior da fábrica com a paleteira. Desta forma, o palete de quatro entradas apresenta uma maior versatilidade. Outra vantagem está na versatilidade para movimentação em cargas blocadas, além de possibilitar arranjos de forma alternada nos meios de transportes.



Figura 2.5. Palete de quatro entradas.

- Paletes de faces simples: são aqueles com apenas uma face destinada a receber a carga (Figura 2.6). São geralmente utilizados para movimentações internas. Oferecem facilidade na movimentação por paleteiras e necessitam de dispositivo porta-paletes para seu acondicionamento verticalizado. Não são indicados para cargas pesadas, pois seu desenho possibilita menor resistência do palete à flexão.

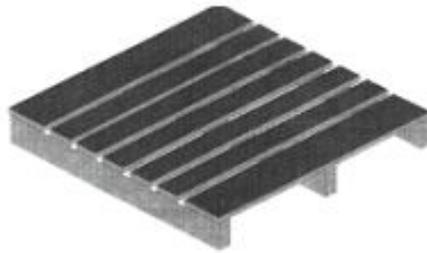


Figura 2.6. Paleta de face simples.

- Paletes faces duplas: são aqueles que têm uma face superior para receber carga e outra inferior para apoio (Figura 2.7). São os paletes mais utilizados pelo mercado. A resistência à flexão é maior quanto mais próximo de face dupla estiver reversível. No empilhamento de forma blocada as cargas que sofreram remonte estão sujeitas a menor possibilidade de danos ou avarias.



Figura 2.7. Paleta de faces duplas.

- Paletes reversíveis: são aqueles com duas faces iguais (Figura 2.8). São utilizados principalmente para acondicionamento de cargas blocadas. Usuários de sacarias dão preferência aos paletes de faces reversíveis, pois são os que menos danificam a carga de suporte e permitem a melhor estabilidade de pilha.

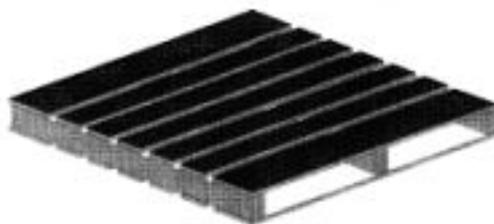


Figura 2.8. Paleta reversível.

- Paletes com abas: são aqueles cujas faces superiores ou inferiores (ou ambas) se projetam além dos apoios, em lados opostos (Figura 2.9). São utilizados de forma a permitir a inserção de barras ou cabos de içamento, além de serem ideais nos casos de carga paletizada através do sistema de capas termoretráteis.

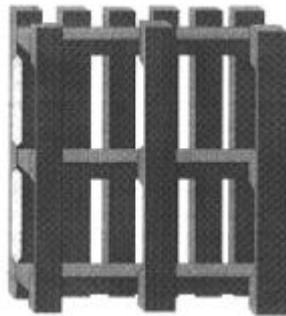


Figura 2.9. Paleta com abas.

- Paletes desquidados: são aqueles com os cantos cortados ou arredondados. São utilizados para evitar esforços perfurantes nas cargas, principalmente as blocadas.
- Caixas-Paletes: são paletes com paredes que podem ser removíveis, com ou sem tampo superior (Figura 2.10). Tem como função conter determinadas mercadorias, suportar a compressão no empilhamento e proporcionar maior proteção mecânica ao conjunto. Geralmente utilizados para produtos que não oferecem regularidade na pilha ou com dimensões muito diminutas.

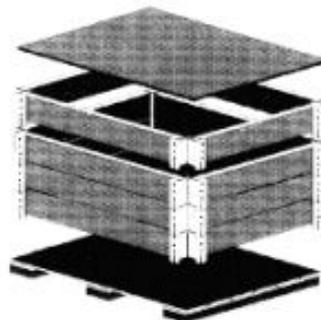


Figura 2.10. Caixa paleta.

2.5. O PROBLEMA DE CARREGAMENTO DE PALETES

O problema de carregamento de paletes pode ser visto como um caso particular do problema de empacotamento. A seguir, é apresentado o problema de corte e empacotamento e, em seguida, apresentamos o problema de carregamento de paletes.

2.5.1. PROBLEMA DE CORTE E EMPACOTAMENTO

O problema de corte, genericamente, consiste em cortar unidades maiores (objetos) em unidades menores (itens), otimizando uma determinada função (por exemplo, minimização da perda de material). Esse tipo de problema aparece em diversos processos industriais de corte, tais como no corte de bobinas de papel e alumínio, barras de aço, chapas metálicas e de madeira, caixas de papelão, entre outros.

Análogo ao problema de corte, o problema de empacotamento consiste em empacotar unidades menores em unidades maiores, otimizando uma determinada função (por exemplo, minimização do espaço ocioso) e satisfazendo um conjunto de restrições adicionais (por exemplo, estabilidade do carregamento). Exemplos de problema de empacotamento ocorrem quando se carregam produtos embalados sobre paletes, ou dentro de contêineres e caminhões.

Essas duas classes de problemas de otimização estão intimamente relacionadas e são tratadas na literatura como problemas de corte e empacotamento (DYCKHOFF, 1990).

2.5.2. PROBLEMA DE CARREGAMENTO DE PALETES

Conforme o Capítulo 1, o problema de carregamento de paletes consiste em carregar produtos (embalados em caixas) sobre um palete retangular, de maneira a otimizar o aproveitamento do palete. Admite-se que as caixas, em geral disponíveis em grandes quantidades, devem ser arranjadas ortogonalmente, isto é, com seus lados paralelos aos lados do 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.

O problema de carregamento de paletes pode ser visto como um caso especial dos problemas de corte e empacotamento. Empacotar unidades (por exemplo,

caixas) sobre unidades maiores (paletes) é similar a cortar unidades (por exemplo, chapas) para produzir unidades menores (itens). HODGSON (1982), ao estudar o problema de carregamento de paletes, dividiu-o em dois possíveis casos: o problema de carregamento de paletes do produtor (*manufacturer's pallet loading problem*) ou PCP do produtor, e o problema de carregamento de paletes do distribuidor (*distributor's pallet loading problem*).

No primeiro caso, o produtor fabrica um bem que é embalado em caixas iguais. Estas caixas devem ser então carregadas em camadas horizontais sobre paletes (Figura 2.11(a)). Em cada camada, uma orientação vertical das caixas em geral é fixada. Se nenhuma orientação for fixada (isto é, as caixas puderem ser carregadas sob qualquer uma de suas faces), o problema é decomposto em dois subproblemas:

- i) para cada face das caixas, o problema bidimensional de arranjar ortogonalmente o máximo número de retângulos (isto é, faces das caixas) dentro de um retângulo maior (superfície do palete);
- ii) o problema unidimensional de arranjar o conjunto mais valioso de camadas ao longo da altura máxima permitida sobre o palete (onde o valor de cada camada corresponde ao número de caixas nela contidas). Note que este último é um problema da mochila (*knapsack problem*) (MARTELLO e TOTH, 1990).

No segundo caso, o distribuidor recebe vários bens de diversos fornecedores, que estão embalados em caixas de diferentes tamanhos. Essas caixas são então carregadas sobre paletes (Figura 2.11(b)). O problema de carregamento de paletes do distribuidor pode ser visto como um problema tridimensional de carregamento de contêineres (BISCHOFF e RATCLIFF, 1995; MORABITO e ARENALES, 1994).

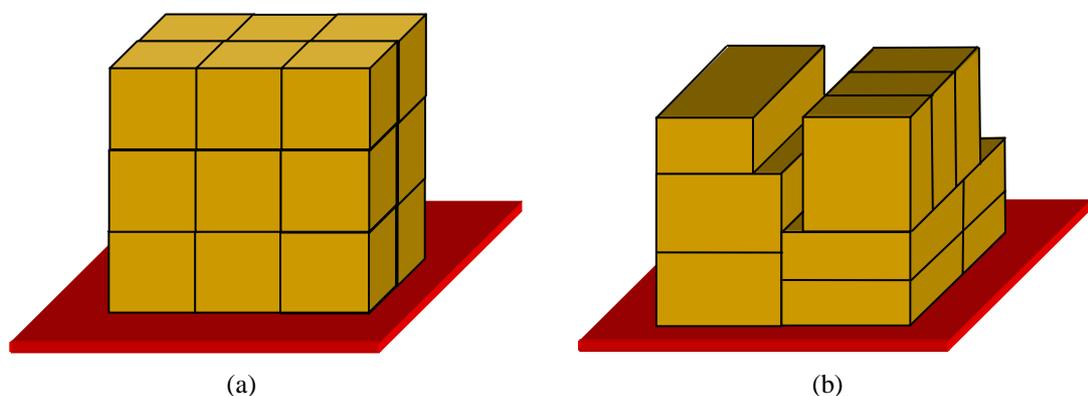


Figura 2.11. Padrão de empacotamento para o problema de carregamento de paletes (a) do produtor e (b) do distribuidor.

2.5.2.1. DEFINIÇÃO DO PROBLEMA DO PRODUTOR

Como dito anteriormente, no PCP do produtor, os paletes devem ser carregados com caixas de dimensões idênticas, ou seja, caixas de dimensões (l,w) devem ser arranjadas sobre a superfície (L,W) do plete resultando em uma camada de altura h . Supõe-se que as condições $l \leq L$, $w \leq W$, $l \leq W$ e $w \leq L$ sejam satisfeitas. Em geral, apenas a orientação vertical é fixada, isto é, as caixas são colocadas sobre o plete sempre com a face (l,w) voltada para cima. Considerações práticas também impõem que as caixas devem ser arranjadas sempre paralelas às bordas do plete. Dessa forma, as caixas podem ter duas orientações horizontais, como pode ser visto na Figura 2.12:

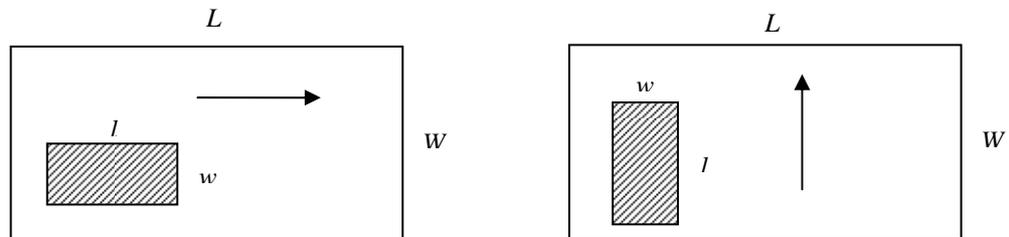


Figura 2.12. Orientações horizontais das caixas dentro do plete.

Um plete carregado sob essas condições consiste de camadas horizontais iguais de altura h , por sua vez empilhadas, umas sobre as outras, ao longo de uma altura H fixada para o plete. A maneira como as caixas estão dispostas nas camadas é denominado padrão de empacotamento. Essas considerações práticas fazem com que o problema original de gerar um arranjo ótimo de três dimensões seja reduzido a um problema de determinar um padrão ótimo para cada camada, tornando-se assim um arranjo bidimensional de retângulos menores dentro de um retângulo maior, sem sobreposição. Sabemos que, ao contrário de muitos equipamentos de corte, as operações com os equipamentos de empacotamento não impõem muitas restrições nos padrões produzidos.

Podemos dizer que a maioria dos métodos de solução disponíveis na literatura foi desenvolvida originalmente para os problemas de corte e, assim, a partir de agora serão discutidas resumidamente as principais restrições que podem aparecer nesses problemas, conforme MORABITO (1992). Nos problemas de corte, a tecnologia dos equipamentos cortantes restringe os possíveis padrões de corte a serem produzidos.

Este tipo de restrição afeta diretamente os métodos de solução, tornando-os mais ou menos complexos.

Uma primeira classificação refere-se aos cortes retos ou curvos. Problemas com cortes curvos são pouco abordados na literatura. Alguns exemplos são corte de discos a partir de placas circulares e o corte de peças de roupas. No entanto, a maioria dos trabalhos publicados na literatura refere-se aos cortes retos. Estes podem ser divididos em cortes ortogonais e não-ortogonais. Os cortes ortogonais são aqueles feitos sempre paralelos às bordas da peça a ser cortada e os não-ortogonais são aqueles produzidos em qualquer direção (Figura 2.13).

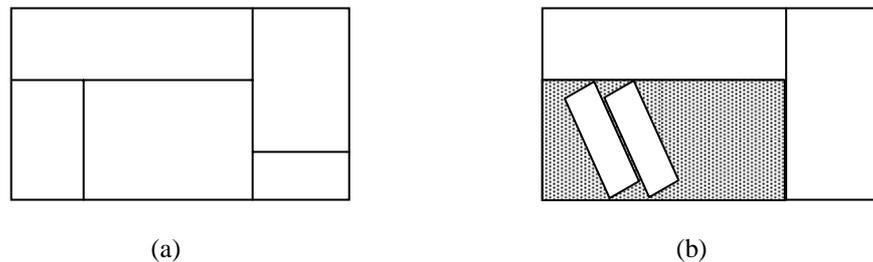


Figura 2.13. (a) cortes ortogonais e (b) cortes não-ortogonais.

Os cortes ortogonais podem ser classificados em cortes guilhotinados e não-guilhotinados. São denominados cortes guilhotinados aqueles que vão de uma borda até a outra borda oposta de uma folha de papel, produzindo dois retângulos menores e adjacentes. Assim, um padrão do tipo guilhotinado é obtido por sucessivos cortes guilhotinados, como pode ser visto na Figura 2.14(a). Exemplos práticos de cortes guilhotinados podem ser observados em indústrias de papel, vidro, móveis e recebem esse nome por causa da semelhança com uma guilhotina de papel.

Cortes não-guilhotinados são aqueles produzidos por equipamentos que permitem interromper o corte antes de chegar à borda oposta da folha e a partir deste ponto, prosseguir com o corte perpendicular na direção oposta do primeiro. Este tipo de corte, comum em chapas metálicas, é ilustrado na Figura 2.14(b).

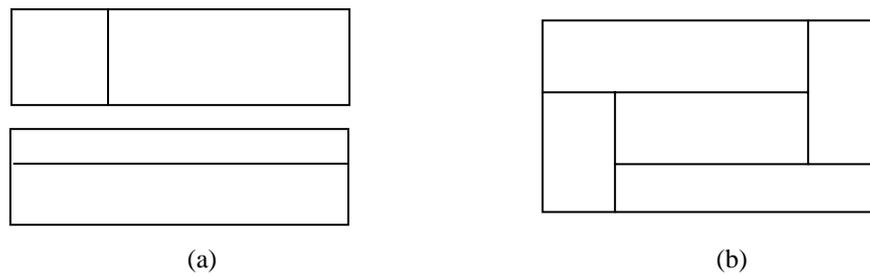


Figura 2.14. (a) cortes guilhotinados e (b) cortes não-guilhotinados.

Os cortes não-guilhotinados podem ainda ser classificados em cortes de 1ª ordem e cortes de ordem superior (ARENALES e MORABITO, 1995). Pela definição, um corte é do tipo não-guilhotinado de 1ª ordem se ele produz cinco retângulos, de maneira que o arranjo desses retângulos não forme um padrão guilhotinado (Figura 2.14(b)). A partir dessa definição, tem-se que um padrão é do tipo não-guilhotinado de 1ª ordem se é obtido através de sucessivos cortes guilhotinados e/ou não-guilhotinados de 1ª ordem (Figura 2.15). Estes padrões foram chamados de 1ª ordem devido ao fato de que os padrões não-guilhotinados mais simples necessitam de cinco retângulos.

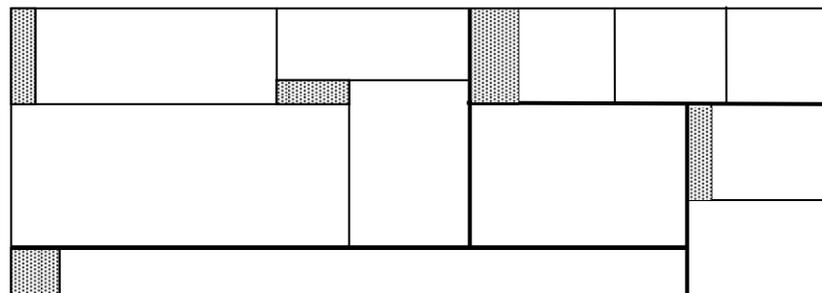


Figura 2.15. Exemplo de padrão não-guilhotinado de 1ª ordem.

Observe pela Figura 2.15 que este padrão é composto por um corte não-guilhotinado de 1ª ordem (indicados pelas linhas em negrito) e cada retângulo resultante é cortado novamente por diferentes cortes guilhotinados e não-guilhotinados de 1ª ordem. Os padrões de corte não-guilhotinados que não são de 1ª ordem são chamados padrões de corte de ordem superior, como mostra a Figura 2.16.

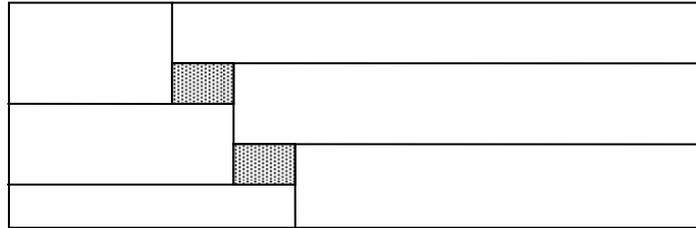


Figura 2.16. Padrão de corte não-guilhotinado de ordem superior.

O PCP do produtor tem como restrições de processo que as caixas estejam arranjadas nas direções ortogonais do palete, produzindo os chamados arranjos ortogonais, que correspondem aos cortes ortogonais já discutidos. Assim, os padrões de empacotamento factíveis são todos aqueles restritos apenas aos empacotamentos ortogonais. ERDÖS e GRAHAM (1975) mostraram que quando empacotamos unidades menores em unidades maiores, é possível melhorar a solução se permitirmos a rotação. A Figura 2.17 ilustra esta situação onde 10 unidades são empacotadas em um retângulo com lado igual a $3 + 1/\sqrt{2}$. Se não permitirmos a rotação, então somente 9 caixas poderiam ser empacotadas.

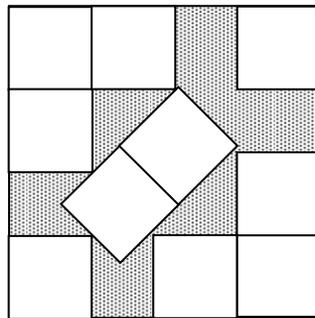


Figura 2.17. Exemplo de padrão de empacotamento não-ortogonal.

Uma consideração a ser feita refere-se à forma de alocação de unidades. Nos processos de corte, unidades são cortadas para produzir unidades menores. Por outro lado, nos problemas de empacotamento, unidades (por exemplo, caixas) são empacotadas para formar unidades maiores (por exemplo, palete). Observe que em ambos os problemas podemos pensar em unidades pequenas sendo alocadas em unidades grandes. Pelo menos duas formas de alocação são possíveis: seleção de unidades pequenas e seleção de unidades grandes. No problema de seleção de unidades grandes, supomos que todas as unidades grandes sejam suficientes para alocar todas as

unidades pequenas. O objetivo do problema é escolher o conjunto ótimo de unidades grandes necessário para alocar todas as unidades pequenas. No problema de seleção de unidades pequenas, supomos que todas as unidades pequenas sejam suficientes para utilizar todas as unidades grandes. O objetivo do problema é escolher o conjunto ótimo de unidades pequenas utilizando todas as unidades grandes.

Neste trabalho estamos interessados no problema de seleção de unidades pequenas (caixas) quando temos apenas uma unidade grande disponível (palete). Dessa forma, o objetivo é carregar da melhor maneira possível um único palete e depois repetimos este padrão muitas vezes ao longo da cadeia logística. Convém lembrar que não consideramos restrições referentes às propriedades dos materiais que compõem os produtos e os paletes, tais como: peso, densidade, fragilidade, etc..

Conforme mencionado no Capítulo 1, DYCKHOFF (1990) apresentou uma tipologia fundamentada na estrutura lógica dos problemas de corte e empacotamento. Assim, o problema de carregamento de paletes do produtor é classificado com a seguinte notação: 2/B/O/C que representa um problema bidimensional (2), tendo como forma de alocação a seleção de unidades pequenas (B) (do alemão *Beladeproblem*), com uma única unidade grande (palete) (O) e unidades pequenas (caixas) de tamanhos iguais (C). Essa notação pode ser utilizada, na literatura de corte e empacotamento, para buscar métodos de solução que sejam específicos para resolver a classe de problemas 2/B/O/C.

O problema de carregamento de paletes do produtor tem importância estratégica e operacional para a empresa. No nível estratégico, uma análise estatística baseada na solução do problema de carregamento de paletes do produtor pode ser útil para escolher o palete mais adequado para uma empresa (dentro um conjunto de paletes padronizados disponíveis), assim como para projetar novos paletes (MORALES *et al.*, 1997). No nível operacional, métodos mais efetivos para resolver o problema de carregamento de paletes do produtor permitem encontrar melhores padrões de empacotamento para a armazenagem e distribuição de um produto.

2.5.2.2. DEFINIÇÃO DO PROBLEMA DO DISTRIBUIDOR

No problema do distribuidor, caixas de dimensões (l_i, w_i, h_i) são arranjadas em camadas ou em colunas sobre a superfície (L, W) do palete. No caso mais geral também podemos arranjar as caixas sobre o palete considerando as três dimensões espaciais. Nesse caso, o problema coincide com o problema de carregamento de contêineres (MORABITO e ARENALES, 1994). Na prática, o problema de carregamento de palete do distribuidor consiste em selecionar caixas com a mesma altura h e arranjá-las inicialmente em camadas e depois, empilhar essas camadas umas sobre as outras, ao longo de uma altura H fixada para o palete. Podemos considerar este problema como uma generalização do problema do produtor, porém mais complexo e envolvendo maiores dificuldades de solução. Quanto à tipologia de DYCKHOFF (1990), este problema pode ser classificado como 2/B/O/M, onde M indica que podemos ter unidades pequenas (caixas) de tamanhos diferentes.

Neste trabalho abordamos somente o PCP do produtor, em particular o problema bidimensional 2/B/O/C descrito na seção 2.5.2.1, de arranjar, sem sobreposição, o maior número possível de retângulos (l, w) e (w, l) dentro do retângulo (L, W) . Conforme NELISSEN (1994), LETCHFORD e AMARAL (2001) e MARTINS (2002), quanto a complexidade computacional do PCP do produtor, podemos dizer que não se sabe se este (simples) problema pode ser resolvido em tempo polinomial e também ainda não foi provado se ele pertence a classe NP-difícil. Como sabemos, todo exemplo do PCP pode ser definido usando apenas 4 valores inteiros e assim um código “razoável” exigiria $O(\log_2 L)$. Se toda solução certificada, produzida pelo algoritmo, isto é, o arranjo de caixas no palete, é representado pela designação de coordenadas planares para cada caixa, o qual pode exigir $O((L/w)^2)$ valores, então este não é limitado por um polinômio no tamanho da entrada e o problema não estaria em NP. Enquanto que o PCP guilhotinado é resolvido em $O(\log_2 L(\log_2 l)^2)$, como mostrado por TARNOWSKI *et al.* (1994), não temos conhecimento de nenhum resultado similar disponível para o caso não-guilhotinado.

CAPÍTULO 3

FORMULAÇÃO MATEMÁTICA

3.1. INTRODUÇÃO

Conforme mencionado no capítulo anterior, o PCP do produtor é classificado como 2/B/O/C (bidimensional, seleção de itens, único objeto, itens iguais), de acordo com a tipologia de DYCKHOFF (1990). Basicamente, o PCP do produtor 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). Admite-se que as caixas estão disponíveis em grandes quantidades e devam 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.

Neste capítulo são apresentadas diferentes formulações matemáticas que podem ser utilizadas para o problema de carregamento de paletes encontradas na literatura. A seguir, revisamos os trabalhos de BEASLEY (1985b), TSAI *et al.* (1993), HADJICONSTANTINOU e CHRISTOFIDES (1995), CHEN *et al.* (1995) e BEASLEY (2004) bem como as formulações utilizadas pelos autores adaptadas para a

resolução do PCP do produtor. Outras formulações relacionadas com as de TSAI *et al.* (1993) e CHEN *et al.* (1995) podem ser encontradas em MARTINS (2002).

3.2. FORMULAÇÃO DE BEASLEY

De forma geral, o trabalho de BEASLEY (1985b) considera o problema bidimensional de cortar um número de peças retangulares de um único retângulo maior de forma a maximizar o valor das peças cortadas. É aplicada a técnica de relaxação Lagrangiana (sobre restrições *surrogate*) na formulação de programação inteira 0–1 e o limitante obtido é usado no procedimento de busca em árvore (*branch and bound*). O procedimento de otimização do subgradiente é utilizado para otimizar o limitante obtido por relaxação Lagrangiana. São usados testes de redução do problema derivados do problema original e de relaxação Lagrangiana. A incorporação de limitantes e testes de redução no procedimento de busca em árvore são capazes de reduzir moderadamente o tamanho dos problemas a serem resolvidos.

O PCP do produtor pode ser modelado como um programa inteiro 0–1 (FARAGO e MORABITO, 2000), o qual é um caso particular do modelo proposto em BEASLEY (1985b) para o problema de corte bidimensional não-guilhotinado. Consideremos então que um palete (L, W) , de comprimento L e largura W , deve ser carregado com caixas iguais com faces inferiores de dimensões l e w . 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:

$$X = \{p \mid 0 \leq p \leq L - w, \text{inteiro}\} \text{ e } Y = \{q \mid 0 \leq q \leq W - w, \text{inteiro}\},$$

os conjuntos das coordenadas (p,q) , respectivamente, para se colocar o canto inferior esquerdo da face (l,w) de uma caixa dentro do palete. Sem perda de generalidade, podemos restringir os conjuntos X e Y para os conjuntos normais (CHRISTOFIDES e 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 que é útil 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 3.1):

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

Note que $a_{ipqrs} = 1$ indica que a coordenada (r,s) fica coberta, ao se colocar uma caixa com orientação i na coordenada (p,q) do palete.

Finalmente, 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} :

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

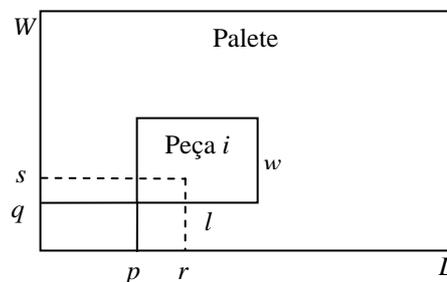


Figura 3.1. Posição (r,s) não permitida em função da colocação de uma caixa na posição (p,q) com orientação i .

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

$$\text{Max} \quad \sum_{i=1}^2 \sum_{\{p \in X | p \leq L-l_i\}} \sum_{\{q \in Y | q \leq W-w_i\}} x_{ipq} \quad (3.1)$$

sujeito a:

$$\sum_{i=1}^2 \sum_{\{p \in X | p \leq L-l_i\}} \sum_{\{q \in Y | q \leq W-w_i\}} a_{ipqrs} x_{ipq} \leq 1, \text{ para todo } r \in X, s \in Y \quad (3.2)$$

$$P \leq \sum_{i=1}^2 \sum_{\{p \in X | p \leq L-l_i\}} \sum_{\{q \in Y | q \leq W-w_i\}} x_{ipq} \leq Q \quad (3.3)$$

$$x_{ipq} \in \{0,1\}, i=1,2, p \in X \text{ tal que } p \leq L-l_i, q \in Y \text{ tal que } q \leq W-w_i \quad (3.4)$$

A função objetivo (3.1) maximiza o número de caixas colocadas sobre o palete. Note que a restrição (3.2) garante que cada par (r,s) seja ocupado por no máximo uma caixa e, desta maneira, evita sobreposição no arranjo de caixas (Figura 3.1). A restrição (3.3) 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. No PCP do produtor, esta restrição (3.3), em geral, é redundante para o problema, mas será aqui mantida por ser útil na relaxação Lagrangiana e *Surrogate*. Em certos casos a restrição (3.3) 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 (3.1)-(3.4) 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 $|Y|$.

3.2.1. EXEMPLO NUMÉRICO

Nesta seção, apresentamos um simples exemplo numérico com o objetivo de ilustrar a aplicação do modelo proposto por BEASLEY (1985b) para o PCP do

produtor. Consideremos um palete com dimensões $(L,W)=(5,4)$ e caixas $(l,w)=(3,2)$. As caixas podem ser colocadas em duas posições, ou seja, na horizontal e na vertical, com seus lados paralelos aos lados do palete. Assim, temos $(l_1,w_1)=(3,2)$ e $(l_2,w_2)=(2,3)$.

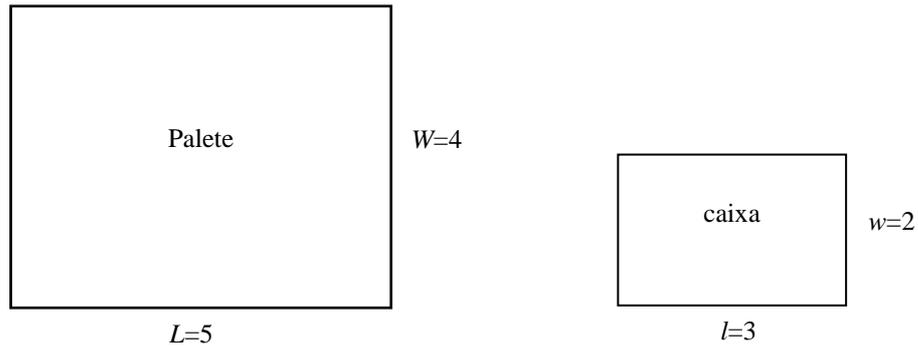


Figura 3.2. Exemplo de um paleta $(L,W)=(5,4)$ e caixa $(l,w)=(3,2)$.

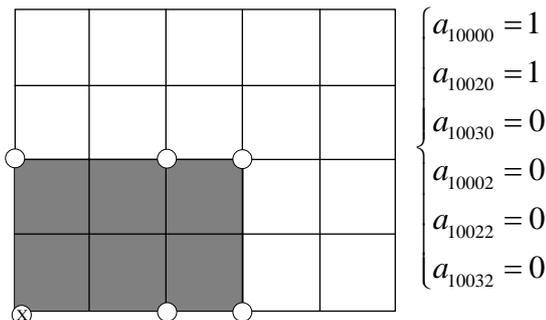
As posições onde é possível colocar o canto inferior esquerdo das caixas são dadas pelos conjuntos X e Y :

$$X = \{0,1,2,3\} \text{ e } Y = \{0,1,2\}.$$

Consideramos (p,q) a posição onde colocamos o canto inferior esquerdo da caixa, dentro do paleta (L,W) , com $p \in X$ e $q \in Y$. No entanto, estes conjuntos podem ser discretizados (conjuntos normais), ou seja, só é considerado no conjunto os pontos que realmente devemos analisar. Logo, os conjuntos discretizados são: $X = \{0,2,3\}$ e $Y = \{0,2\}$.

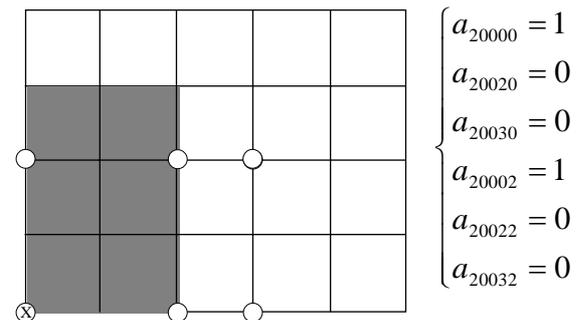
Definimos os valores da função a por:

- para $(p,q) = (0,0)$ e $i=1$:



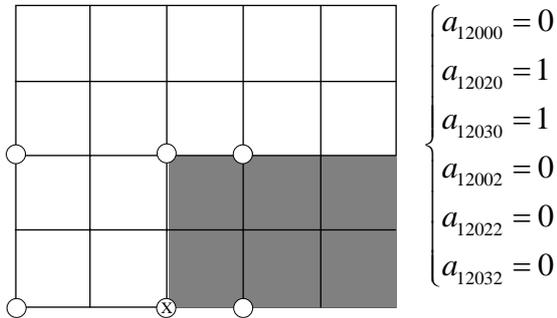
$$\begin{cases} a_{10000} = 1 \\ a_{10020} = 1 \\ a_{10030} = 0 \\ a_{10002} = 0 \\ a_{10022} = 0 \\ a_{10032} = 0 \end{cases}$$

- para $(p,q) = (0,0)$ e $i=2$:

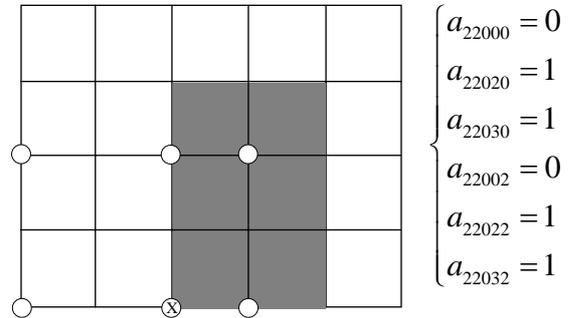


$$\begin{cases} a_{20000} = 1 \\ a_{20020} = 0 \\ a_{20030} = 0 \\ a_{20002} = 1 \\ a_{20022} = 0 \\ a_{20032} = 0 \end{cases}$$

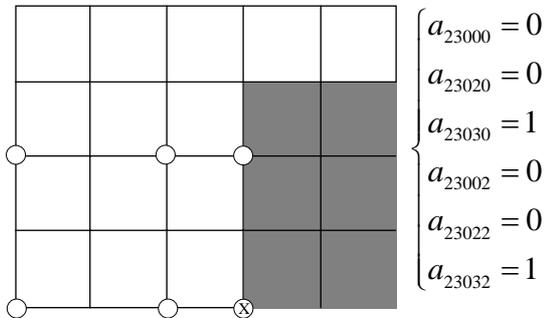
- para $(p, q) = (2, 0)$ e $i=1$:



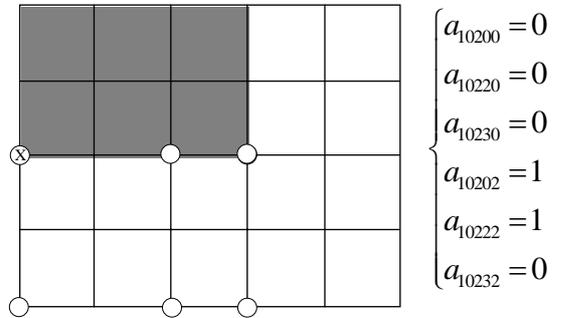
- para $(p, q) = (2, 0)$ e $i=2$:



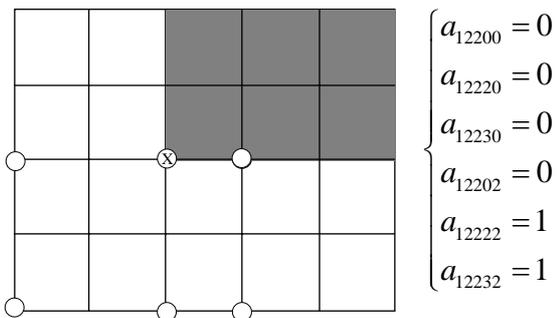
- para $(p, q) = (3, 0)$ e $i=2$:



- para $(p, q) = (0, 2)$ e $i=1$:



- para $(p, q) = (2, 2)$ e $i=1$:



Temos as seguintes inequações para este exemplo:

- para $(r, s) = (0, 0)$: $x_{100} + x_{200} \leq 1$;
- para $(r, s) = (0, 2)$: $x_{102} + x_{200} \leq 1$;
- para $(r, s) = (2, 0)$: $x_{100} + x_{120} + x_{220} \leq 1$;
- para $(r, s) = (2, 2)$: $x_{102} + x_{122} + x_{220} \leq 1$;

- para $(r, s) = (3, 0)$: $x_{120} + x_{220} + x_{230} \leq 1$;
- para $(r, s) = (3, 2)$: $x_{122} + x_{220} + x_{230} \leq 1$.

Note que temos seis restrições para o modelo original, já que o número de restrições é dado por $|X||Y|$. Assim, o problema é dado por:

$$\text{Max } x_{100} + x_{102} + x_{120} + x_{122} + x_{200} + x_{220} + x_{230}$$

sujeito a:

$$x_{100} + x_{200} \leq 1$$

$$x_{102} + x_{200} \leq 1$$

$$x_{100} + x_{120} + x_{220} \leq 1$$

$$x_{102} + x_{122} + x_{220} \leq 1$$

$$x_{120} + x_{220} + x_{230} \leq 1$$

$$x_{122} + x_{220} + x_{230} \leq 1$$

$$x_{100} + x_{102} + x_{120} + x_{122} + x_{200} + x_{230} \leq 3$$

$$x_{ipq} \in \{0, 1\}$$

Estamos considerando neste modelo o número mínimo e máximo de caixas com ambas as orientações, onde:

$$Q = \left\lfloor \frac{L * W}{l * w} \right\rfloor = \left\lfloor \frac{5 * 4}{3 * 2} \right\rfloor = \left\lfloor \frac{20}{6} \right\rfloor = 3; P = 0,$$

que resulta na restrição $x_{100} + x_{102} + x_{120} + x_{122} + x_{200} + x_{230} \leq 3$. Este problema foi resolvido utilizando o pacote GAMS/CPLEX. A solução ótima obtida foi $x_{100} = x_{122} = x_{230} = 1$ e $x_{102} = x_{120} = x_{200} = x_{220} = 0$, com o valor da função objetivo igual a 3. A Figura 3.3 ilustra tal solução ótima. No Apêndice A encontra-se o respectivo programa GAMS do modelo de BEASLEY (1985b) que produziu o padrão ótimo da Figura 3.3.

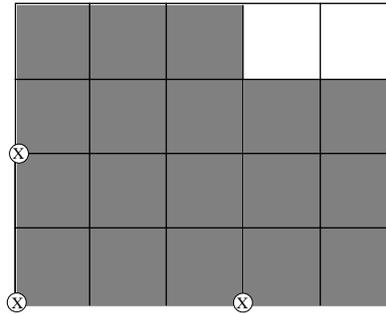


Figura 3.3. Padrão de carregamento ótimo para o exemplo utilizando o modelo GAMS/CPLEX de BEASLEY(1985b).

3.3. FORMULAÇÃO DE TSAI *ET AL.*

TSAI *et al.* (1993) apresentam uma versão do problema de carregamento de paletes tridimensional com caixas de vários tamanhos. O método de carregamento permite que várias caixas de diversos tamanhos sejam colocadas no mesmo palete. Uma restrição é imposta no número de caixas de cada tamanho que pode ser carregada e o problema é formulado como um problema de programação inteira mista 0–1. Consideram, também, uma proporção pré-determinada entre o número de caixas de cada tipo e o total de caixas carregadas no palete. Essa restrição de proporção não será considerada neste trabalho. O modelo original de TSAI *et al.* (1993) pode ser adaptado para modelar o PCP do produtor. As variáveis e os parâmetros do modelo são:

- S : conjunto de n caixas a serem consideradas;
- (l_i, w_i) : dimensões da caixa i no conjunto S (comprimento, largura);
- (L, W) : dimensões do palete (comprimento, largura). Observe que $(l_i, w_i) = (l, w)$ ou ainda, $(l_i, w_i) = (w, l)$;
- (X^o, Y^o) : canto inferior esquerdo do palete no plano cartesiano ao longo dos eixos x e y ;
- (x_i, y_i) : variáveis de decisão (coordenadas x e y do canto inferior esquerdo da caixa i);
- P_k : variável de decisão binária associada à k -ésima caixa do conjunto S . A caixa k é colocada no palete se $P_k = 1$ e é descartada do conjunto S se $P_k = 0$.

Note que no conjunto S devemos ter um número suficientemente grande de caixas com dimensões (l, w) e (w, l) . Assim, a formulação matemática para o

problema é dada por:

$$\text{Max } \sum_{k=1}^n P_k \quad (3.5)$$

sujeito a:

$$x_i - x_j \geq l_j \quad \text{para todo } i \text{ e } j \quad (3.6)$$

ou

$$x_j - x_i \geq l_i \quad \text{para todo } i \text{ e } j \quad (3.7)$$

ou

$$y_i - y_j \geq w_j \quad \text{para todo } i \text{ e } j \quad (3.8)$$

ou

$$y_j - y_i \geq w_i \quad \text{para todo } i \text{ e } j \quad (3.9)$$

$$x_k \geq X^0 P_k \quad \text{para todo } k \quad (3.10)$$

$$y_k \geq Y^0 P_k \quad \text{para todo } k \quad (3.11)$$

$$x_k \leq (X^0 + L) - l_k \quad \text{para todo } k \quad (3.12)$$

$$y_k \leq (Y^0 + W) - w_k \quad \text{para todo } k \quad (3.13)$$

$$P_k \in \{0,1\} \quad (3.14)$$

$$x_k, y_k \geq 0 \quad (3.15)$$

$$i = 1, 2, \dots, n-1 \quad (3.16)$$

$$j = i+1, i+2, \dots, n \quad (3.17)$$

$$k = 1, 2, \dots, n \quad (3.18)$$

O canto inferior esquerdo do palete é localizado na coordenada (X^0, Y^0) do plano cartesiano. (X^0, Y^0) é selecionada no eixo xy tal que todas as caixas do conjunto S possam ser alocadas no retângulo maior de comprimento (X^0+L) e largura (Y^0+W) . A Figura 3.4 mostra a localização do palete $(ABCD)$ no eixo xy . Essa alocação garante que as variáveis contínuas x_i e y_i resultem em uma solução positiva, condição necessária para a formulação de (3.15)-(3.18).

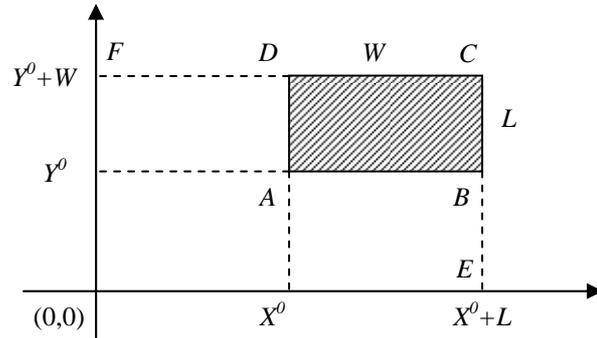


Figura 3.4. Localização do palete no eixo xy .

No modelo (3.5)-(3.18), a função objetivo maximiza o total da área coberta no palete pelas caixas. As restrições disjuntivas (3.6)-(3.9) garantem que não haverá sobreposição das peças. As restrições (3.10)-(3.13) delimitam a colocação da caixa no palete. Observe que a solução fornece não somente o número de caixas de cada tipo, mas também a localização de cada caixa no palete.

Na formulação matemática acima, cada conjunto de restrições de posição consiste em quatro restrições de múltipla escolha (restrições (3.6)-(3.9)). Considerando as seguintes variáveis binárias u_{ij}^1 e u_{ij}^2 , o modelo (3.5)-(3.18) pode ser transformado em um modelo de programação inteira mista 0-1. As quatro possíveis combinações de u_{ij}^1 e u_{ij}^2 são apresentadas na Tabela 3.1. Estas quatro combinações podem ser usadas para compor o lado direito das restrições de múltipla escolha ((3.6)-(3.9)). Assim, as equações (3.6)-(3.9) são equivalentes a:

$$x_j - x_i \leq -l_j + M(u_{ij}^1 + u_{ij}^2) \quad (3.19)$$

$$x_i - x_j \leq -l_i + M[1 - (u_{ij}^2 - u_{ij}^1)] \quad (3.20)$$

$$y_j - y_i \leq -w_j + M[1 - (u_{ij}^1 - u_{ij}^2)] \quad (3.21)$$

$$y_i - y_j \leq -w_i + M[2 - (u_{ij}^1 + u_{ij}^2)] \quad (3.22)$$

$$u_{ij}^1, u_{ij}^2 \in \{0, 1\} \quad (3.23)$$

Nas equações (3.19)-(3.22), M é um número suficientemente grande. As possíveis combinações de u_{ij}^1 e u_{ij}^2 e suas possíveis equações de restrições são apresentadas na Tabela 3.1.

Tabela 3.1. Valores binários e equações de restrições associadas.

Variáveis binárias		Valor das equações (lado direito)				Equações de restrições aplicáveis
u_{ij}^1	u_{ij}^2	(3.19)	(3.20)	(3.21)	(3.22)	
0	0	$-l_j$	M	M	$2M$	(3.19)
0	1	M	$-l_i$	$2M$	M	(3.20)
1	0	M	$2M$	$-w_j$	M	(3.21)
1	1	$2M$	M	M	$-w_i$	(3.22)

Desde que M seja suficientemente grande, o valor resultante do lado direito das equações de $-l_i + M$, $-w_i + M$, etc. está simplesmente indicado como M na Tabela 3.1. Nesta tabela, uma das quatro restrições originais não deverá mudar e o restante não terá efeito por causa do valor suficientemente grande do lado direito das equações. Assim, resolvendo a formulação do modelo apresentado ((3.5), (3.10)-(3.18) e (3.19)-(3.23)) a solução fornece o número necessário de caixas ($\sum_{k=1}^n P_k$) e sua localização no palete (x_k, y_k) . O tamanho do modelo está relacionado aproximadamente ao número de restrições e variáveis. Assim, temos que o número de variáveis é $O(3n + 2n(n-1)/2)$ e o número de restrições é $O(4n(n-1)/2 + 4n)$, onde n é o número de diferentes caixas no conjunto S .

Note que o modelo cresce com o aumento do número n de caixas disponíveis. Isso não ocorre no modelo de BEASLEY (1985b), onde o modelo cresce com o aumento de $|X|$ e $|Y|$. No Apêndice B encontra-se um exemplo numérico e o modelo GAMS de TSAI *et al.* (1993).

3.4. FORMULAÇÃO DE HADJICONSTANTINOU E CHRISTOFIDES

No trabalho de HADJICONSTANTINOU e CHRISTOFIDES (1995) é apresentado um procedimento exato de busca em árvore para a resolução do problema de corte bidimensional, no qual um número de peças retangulares pequenas, cada uma com um dado tamanho e valor, são cortados de uma peça retangular maior. O objetivo é minimizar a sobra. É considerado o número máximo de vezes que uma peça pode ser usada em um padrão de corte. O algoritmo utilizado limita o tamanho da busca em árvore usando um limitante obtido através da aplicação de relaxação Lagrangiana de uma formulação de programação inteira 0–1 e, para otimizar este limitante, é utilizado o procedimento de otimização do subgradiente. Testes de redução derivados do problema original e do problema com a aplicação da técnica de relaxação Lagrangiana produzem um ganho computacional. O problema de corte bidimensional segundo HADJICONSTANTINOU e CHRISTOFIDES (1995) é definido como segue.

Considera-se uma peça retangular maior de comprimento L e largura W e o conjunto de m pequenas peças retangulares com dimensões $(l_1, w_1), (l_2, w_2), \dots, (l_m, w_m)$ com correspondentes valores (v_1, v_2, \dots, v_m) que podem ser cortadas (ortogonalmente) da peça retangular maior. O objetivo é construir um padrão de corte (ortogonal, mas não necessariamente do tipo guilhotinado) de valor máximo usando não mais que Q_i réplicas de cada peça do tipo i . Note que ao se considerar apenas duas peças (isto é, $m = 2$) com $(l_1, w_1) = (l, w)$, $(l_2, w_2) = (w, l)$ e $v_1 = v_2 = 1$, temos o caso especial do PCP do produtor. Neste caso, admitimos Q_1 e Q_2 suficientemente grandes.

As posições na peça retangular maior são referenciadas através de coordenadas. Supõe-se que a origem é no canto inferior esquerdo e o topo do lado direito da peça tem coordenadas (L, W) .

Para formular o problema como um problema de programação inteira 0–1, seja:

$$- x_{ijp} = \begin{cases} 1, & \text{se a } j\text{-ésima réplica da peça } i \text{ é cortada com o seu canto inferior esquerdo} \\ & \text{na posição } p \text{ do eixo } x \text{ onde } j = 1, \dots, Q_i \text{ e } p = 0, \dots, L - l_i \\ 0, & \text{caso contrário} \end{cases} ;$$

$$\begin{aligned}
 - \quad y_{ijq} &= \begin{cases} 1, & \text{se a } j\text{-ésima réplica da peça } i \text{ é cortada com o seu canto inferior esquerdo} \\ & \text{na posição } q \text{ do eixo } y \text{ onde } j = 1, \dots, Q_i \text{ e } q = 0, \dots, W - w_i \\ 0, & \text{caso contrário} \end{cases} \quad ; \\
 - \quad z_{rs} &= \begin{cases} 1, & \text{se o ponto } (r, s) \text{ em } (L, W) \text{ não é cortado por qualquer peça, onde} \\ & r = 0, \dots, L - 1, s = 0, \dots, W - 1 \\ 0, & \text{caso contrário} \end{cases} \quad .
 \end{aligned}$$

A Figura 3.5 mostra a localização de uma caixa (l_i, w_i) em um palete (L, W) , onde (r, s) é uma posição não permitida em função da colocação de uma caixa em (p, q) .

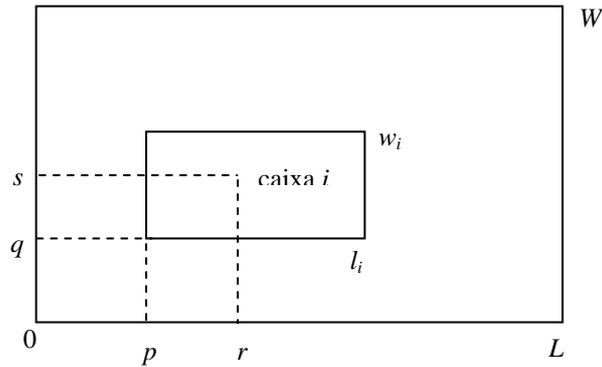


Figura 3.5. Localização de uma caixa em um palete.

Assim, a formulação matemática inteira 0–1 é dada por:

$$\text{Max} \sum_{i=1}^m v_i \sum_{j=1}^{Q_i} \sum_{p=0}^{L-l_i} x_{ijp} \quad (3.24)$$

sujeito a:

$$\sum_{s=q}^{q+w_i-1} \sum_{r=p}^{p+l_i-1} z_{rs} \leq (2 - x_{ijp} - y_{ijq}) l_i w_i \quad i = 1, \dots, m; j = 1, \dots, Q_i; p = 0, \dots, L - l_i$$

$$q = 0, \dots, W - w_i \quad (3.25)$$

$$\sum_{p=0}^{L-l_i} x_{ijp} \leq 1 \quad i = 1, \dots, m; j = 1, \dots, Q_i \quad (3.26)$$

$$\sum_{p=0}^{L-l_i} x_{ijp} = \sum_{q=0}^{W-w_i} y_{ijq} \quad i = 1, \dots, m; j = 1, \dots, Q_i \quad (3.27)$$

$$\sum_{i=1}^m w_i \sum_{j=1}^{Q_i} \sum_{p=\max\{0, r-l_i+1\}}^{\min\{r, L-l_i\}} x_{ijp} + \sum_{s=0}^{W-1} z_{rs} = W \quad r=0, \dots, L-1 \quad (3.28)$$

$$\sum_{i=1}^m l_i \sum_{j=1}^{Q_i} \sum_{q=\max\{0, s-w_i+1\}}^{\min\{s, W-w_i\}} y_{ijq} + \sum_{r=0}^{L-1} z_{rs} = L \quad s=0, \dots, W-1 \quad (3.29)$$

$$x_{ijp} \in \{0,1\} \quad i=1, \dots, m; j=1, \dots, Q_i; p=0, \dots, L-l_i \quad (3.30)$$

$$y_{ijq} \in \{0,1\} \quad i=1, \dots, m; j=1, \dots, Q_i; q=0, \dots, W-w_i \quad (3.31)$$

$$z_{rs} \in \{0,1\} \quad r=0, \dots, L-1; s=0, \dots, W-1 \quad (3.32)$$

A função objetivo (3.24) maximiza o número de caixas colocadas sobre o palete. O conjunto de restrições (3.25) garante que qualquer ponto em (L,W) é cortado por no máximo uma peça. As restrições (3.26) e (3.27) expressam que, de fato, qualquer peça é cortada no máximo uma vez de (L,W) . As restrições (3.28) garantem que nem todas as peças, cuja soma das larguras excedem W , podem ser cortadas de (L,W) com seu canto esquerdo inferior no mesmo comprimento. Similarmente, as restrições (3.29) garantem que se a soma dos comprimentos de algumas peças excedem L , nem todas essas peças podem ser cortadas de (L,W) com seu canto inferior esquerdo na mesma largura.

Note que se $x_{ijp}=1$ e $y_{ijq}=1$, significa que a j -ésima peça do tipo i é colocada no palete com seu canto inferior esquerdo em (p,q) . Se a j -ésima não é colocada no palete, então x_{ijp} deve ser zero para todo p e y_{ijq} será zero para todo q . Sem perda de generalidade, os índices p e q podem ser limitados a $p \in X$ e $q \in Y$ (conjuntos normais) ao invés de $p=0, \dots, L-l_i$ e $q=0, \dots, W-w_i$. Este modelo envolve aproximadamente N_c restrições e N_v variáveis inteiras, onde $N_c = 2 \sum_{i=1}^m Q_i + \sum_{i=1}^m Q_i (|X| + |Y|) + L + W$ e $N_v = \sum_{i=1}^m Q_i (|X| + |Y|) + LW$, onde X e Y são os conjuntos normais. O tamanho desta formulação depende do número total de caixas e do número máximo Q_i de réplicas de cada caixa. Note que, no caso do PCP do produtor, temos $m=2$ e Q_1, Q_2 iguais ao número máximo de caixas carregadas no palete.

De acordo com AMARAL e LETCHFORD (2001) esta formulação nem sempre é válida, pois há soluções inteiras que não representam padrões de corte factíveis. Isto é comprovado pelo seguinte exemplo. Seja $m=2$, $(L,W)=(3,3)$,

$(l_1, w_1, Q_1, v_1) = (2, 1, 2, 1)$ e $(l_2, w_2, Q_2, v_2) = (1, 2, 2, 1)$. Uma solução factível inteira para (3.24)-(3.32) é obtida considerando $x_{110}, x_{121}, x_{210}, x_{222}, y_{110}, y_{122}, y_{210}, y_{221}$ e z_{11} iguais a 1 e todas as outras variáveis iguais a zero. No entanto, esta solução não representa um padrão de corte factível, pois a primeira peça do tipo 1 e a primeira peça do tipo 2 se sobrepõem e, o mesmo acontece para a segunda peça do tipo 1 e a segunda peça do tipo 2. Este exemplo é para o caso restrito, ou seja, a solução envolve apenas 3 caixas, enquanto que para o caso irrestrito envolveria 4 caixas. Assim, não sabemos se para o caso irrestrito existe um exemplo que também invalide a formulação matemática apresentada em HADJICONSTANTINOU e CHRISTOFIDES (1995).

Ainda segundo AMARAL e LETCHFORD (2001), uma formulação válida pode ser construída eliminando as variáveis z e substituindo as restrições (3.25), (3.28), (3.29) e (3.32) por restrições da forma $x_{ijp} + y_{ijq} + x_{klr} + y_{kls} \leq 3$, para todo i, p, q, k, r, s tal que a peça do tipo i colocada com seu canto inferior esquerdo em (p, q) sobrepõe a peça do tipo k com seu canto inferior esquerdo em (r, s) ; e para $j = 1, \dots, Q_i$ e $l = 1, \dots, Q_k$ fornece que ou $i \neq k$ ou $j \neq l$. No entanto, a formulação resultante envolve um número de variáveis e restrições intratável computacionalmente nos problemas práticos. A formulação segundo AMARAL e LETCHFORD (2001) é dada por:

$$\text{Max} \sum_{i=1}^m v_i \sum_{j=1}^{Q_i} \sum_{p=0}^{L-l_i} x_{ijp} \quad (3.33)$$

sujeito a:

$$\sum_{p=0}^{L-l_i} x_{ijp} \leq 1 \quad i = 1, \dots, m; j = 1, \dots, Q_i \quad (3.34)$$

$$\sum_{p=0}^{L-l_i} x_{ijp} = \sum_{q=0}^{W-w_i} y_{ijq} \quad i = 1, \dots, m; j = 1, \dots, Q_i \quad (3.35)$$

$$\begin{aligned} x_{ijp} + y_{ijq} + x_{klr} + y_{kls} &\leq 3 & i = 1, \dots, m \\ & & p = 1, \dots, L-l_i; q = 0, \dots, W-w_i \\ & & r = 0, \dots, L-1; s = 0, \dots, W-1 \\ & & j = 1, \dots, Q_i; l = 1, \dots, Q_k \\ & & k = 1, \dots, m, i \neq k \text{ ou } j \neq l \end{aligned} \quad (3.36)$$

$$x_{ijp} \in \{0,1\} \quad i = 1, \dots, m; j = 1, \dots, Q_i; p = 0, \dots, L - l_i \quad (3.37)$$

$$y_{ijq} \in \{0,1\} \quad i = 1, \dots, m; j = 1, \dots, Q_i; q = 0, \dots, W - w_i \quad (3.38)$$

No Apêndice C encontra-se um exemplo numérico, o modelo GAMS de HADJICONSTANTINOU e CHRISTOFIDES (1995) e o modelo CPLEX de AMARAL e LETCHFORD (2001).

3.5. FORMULAÇÃO DE CHEN *ET AL.*

CHEN *et al.* (1995) consideram o problema de carregamento de contêineres com caixas de tamanho não uniforme e o formulam como um problema de programação inteira mista 0–1, considerando múltiplos contêineres, vários tamanhos de caixas, a orientação das caixas e, ainda, a sobreposição das caixas. Baseado em um modelo geral, casos especiais do problema de carregamento de contêineres são discutidos. As soluções não têm garantia de serem carregamentos estáveis. Um caso especial do problema de carregamento de contêineres é selecionar um contêiner para um dado conjunto de caixas a fim de minimizar a perda de espaço. Baseado neste caso especial, a seguir é apresentado a formulação matemática para o PCP do produtor baseada em CHEN *et al.* (1995).

Os parâmetros e variáveis do modelo são:

- M : número suficientemente grande;
- m : número total de paletes disponíveis;
- N : número total de caixas a serem empacotadas;
- (l_i, w_i) : parâmetros que indicam o comprimento e largura da caixa i ;
- (L, W) : parâmetros que indicam o comprimento e largura do palete;
- n_j : variável binária igual a 1 se o palete j é usado e igual a zero, caso contrário.
- (x_i, y_i) : variáveis contínuas (para localização) que indicam as coordenadas do canto inferior frontal esquerdo da caixa i ;
- (l_x, l_y) : variáveis binárias que indicam se o comprimento da caixa i é paralelo ao eixo x ou y . Por exemplo, o valor de l_x é igual a 1 se o comprimento da caixa i é paralelo ao eixo x ; caso contrário, ele é igual a zero;

- (wx_i, wy_i) : variáveis binárias que indicam se a largura da caixa i é paralela ao eixo x ou y . Por exemplo, o valor de wx_i é igual a 1 se a largura da caixa i é paralela ao eixo x ; caso contrário, ele é igual a zero;
- $a_{ik}, b_{ik}, c_{ik}, d_{ik}$: variáveis binárias definidas para indicar a localização das caixas em relação umas às outras. A variável a_{ik} é igual a 1 se a caixa i está no lado esquerdo da caixa k . Similarmente, as variáveis b_{ik}, c_{ik} e d_{ik} definem se a caixa i está à direita, em cima, abaixo da caixa k , respectivamente. Essas variáveis são necessárias e definidas apenas quando $i < k$.

O problema 0–1 é formulado como segue.

$$\text{Min } \sum_{j=1}^m L_j W_j n_j - \sum_{i=1}^N l_i w_i \quad (3.39)$$

sujeito a:

$$x_i + l_i l x_i + w_i w x_i \leq x_k + M(1 - a_{ik}) \quad \text{para todo } i, k, i < k \quad (3.40)$$

$$x_k + l_k l x_k + w_k w x_k \leq x_i + M(1 - b_{ik}) \quad \text{para todo } i, k, i < k \quad (3.41)$$

$$y_i + w_i w y_i + l_i l y_i \leq y_k + M(1 - c_{ik}) \quad \text{para todo } i, k, i < k \quad (3.42)$$

$$y_k + w_k w y_k + l_k l y_k \leq y_i + M(1 - d_{ik}) \quad \text{para todo } i, k, i < k \quad (3.43)$$

$$a_{ik} + b_{ik} + c_{ik} + d_{ik} \geq 1 \quad \text{para todo } i, k, i < k \quad (3.44)$$

$$x_i + l_i l x_i + w_i w x_i \leq \sum_{j=1}^m L_j n_j \quad \text{para todo } i \quad (3.45)$$

$$y_i + w_i w y_i + l_i l y_i \leq \sum_{j=1}^m W_j n_j \quad \text{para todo } i \quad (3.46)$$

$$\sum_{j=1}^m n_j = 1 \quad (3.47)$$

$$l x_i, l y_i, w x_i, w y_i \in \{0, 1\} \quad (3.48)$$

$$a_{ik}, b_{ik}, c_{ik}, d_{ik} \in \{0, 1\} \quad (3.49)$$

$$x_i, y_i \geq 0 \quad (3.50)$$

A função objetivo (3.39) minimiza o espaço total não utilizado do palete. As restrições (3.40)-(3.43) garantem que não haja sobreposição de caixas, o que é

verificado pela restrição (3.44). As restrições (3.45) e (3.46) garantem que todas as caixas colocadas no palete se ajustam dentro de suas dimensões físicas. A restrição (3.47) garante que apenas um palete é selecionado para um dado conjunto de caixas.

Para o caso do PCP do produtor é necessário que se faça algumas modificações no modelo (3.39)-(3.50), pois é preciso que se selecione apenas um palete onde são empacotadas um determinado número de caixas. Assim, consideremos:

- m : limitante superior para o número de caixas;
- (X^0, Y^0) : canto inferior esquerdo do palete no plano cartesiano ao longo dos eixos x e y ;
- P_i : variável de decisão binária onde $P_i = 1$ se a caixa i é colocada no palete e $P_i = 0$, caso contrário.

Logo, o problema 0–1 é formulado como segue.

$$\text{Max } \sum_{i=1}^m P_i \quad (3.51)$$

sujeito a:

$$x_i + l_i l x_i + w_i w x_i \leq x_k + M(1 - a_{ik}) \quad \text{para todo } i, k, i < k \quad (3.52)$$

$$x_k + l_k l x_k + w_k w x_k \leq x_i + M(1 - b_{ik}) \quad \text{para todo } i, k, i < k \quad (3.53)$$

$$y_i + w_i w y_i + l_i l y_i \leq y_k + M(1 - c_{ik}) \quad \text{para todo } i, k, i < k \quad (3.54)$$

$$y_k + w_k w y_k + l_k l y_k \leq y_i + M(1 - d_{ik}) \quad \text{para todo } i, k, i < k \quad (3.55)$$

$$a_{ik} + b_{ik} + c_{ik} + d_{ik} \geq 1 \quad \text{para todo } i, k, i < k \quad (3.56)$$

$$x_i \geq X^0 P_i \quad \text{para todo } i \quad (3.57)$$

$$y_i \geq Y^0 P_i \quad \text{para todo } i \quad (3.58)$$

$$x_i + l_i l x_i + w_i w x_i \leq (X^0 + L) \quad \text{para todo } i \quad (3.59)$$

$$y_i + w_i w y_i + l_i l y_i \leq (Y^0 + W) \quad \text{para todo } i \quad (3.60)$$

$$l x_i, l y_i, w x_i, w y_i \in \{0, 1\} \quad (3.61)$$

$$P_i, a_{ik}, b_{ik}, c_{ik}, d_{ik} \in \{0, 1\} \quad (3.62)$$

$$x_i, y_i \geq 0 \quad (3.63)$$

A função objetivo (3.51) maximiza o total da área coberta no palete pelas caixas. As restrições (3.52)-(3.55) garantem que não haja sobreposição de caixas, o que é verificado pela restrição (3.56). As restrições (3.57)-(3.60) garantem que todas as caixas colocadas no palete se ajustam dentro de suas dimensões físicas. Na seção 3.7 mostra-se que este modelo está diretamente relacionado com o modelo de TSAI *et al.* (1993).

3.6. FORMULAÇÃO NÃO LINEAR DE BEASLEY

Recentemente BEASLEY (2004) apresentou um modelo não linear com variáveis inteiras para o problema de corte não-guilhotinado bidimensional restrito. Esta formulação pode ser estendida para tratar o problema com múltiplos retângulos grandes e com defeitos nos mesmos. Baseado nessa formulação, o autor apresenta uma heurística populacional que, computacionalmente, é capaz de encontrar a solução ótima para alguns problemas da literatura com baixo tempo computacional.

O modelo não linear 0–1 de BEASLEY (2004) também pode ser adaptado para tratar o PCP do produtor. Considerando disponível um número Q_i de cópias idênticas de cada caixa i , definimos:

$$z_{ip} = \begin{cases} 1, & \text{se a } p\text{-ésima cópia } (p = 1, \dots, Q_i) \text{ da caixa } i \text{ é colocada no palete } (L, W) \\ 0, & \text{caso contrário.} \end{cases},$$

onde a posição de qualquer caixa é considerada com referência ao seu centro, ou seja:

x_{ip} é a coordenada x do centro da p -ésima cópia da caixa i ;

y_{ip} é a coordenada y do centro da p -ésima cópia da caixa i .

A Figura 3.6 ilustra a situação onde ocorre a sobreposição das peças. Temos duas caixas, ou seja, i e j , com centros de coordenadas em (x_{ip}, y_{ip}) e (x_{jp}, y_{jp}) , respectivamente.

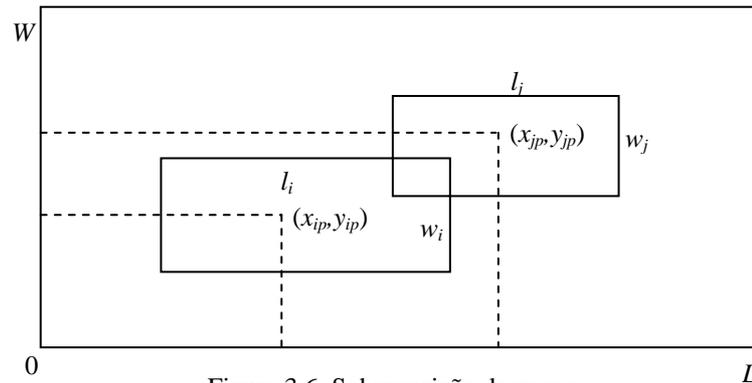


Figura 3.6. Sobreposição de peças.

Esses centros de coordenadas são limitados por:

$$l_i/2 \leq x_{ip} \leq L - l_i/2 \quad i = 1, \dots, m; p = 1, \dots, Q_i,$$

$$w_i/2 \leq y_{ip} \leq W - w_i/2 \quad i = 1, \dots, m; p = 1, \dots, Q_i.$$

Para assegurar um número mínimo de caixas temos:

$$z_{ip} = 1 \quad i = 1, \dots, m; P_i > 0; p = 1, \dots, P_i.$$

Esta restrição nos diz que devemos escolher arbitrariamente as primeiras P_i cópias da peça i . Naturalmente, caixas que não podem ser consideradas no palete (L, W) não podem se sobrepor e considerando a Figura 3.6, na qual mostramos a sobreposição de duas caixas (caixas i e j), podemos expressar esta condição por:

$$|x_{ip} - x_{jp}| \geq (l_i + l_j)/2 \text{ ou } |y_{ip} - y_{jp}| \geq (w_i + w_j)/2, \quad (3.64)$$

ou seja, a diferença entre o centro das coordenadas x é suficiente $(\geq (l_i + l_j)/2)$ para assegurar que não haja sobreposição entre as caixas ou a diferença entre as coordenadas y é suficiente $(\geq (w_i + w_j)/2)$ e, da mesma forma, assegurar que não haja sobreposição entre as caixas.

Para facilitar a notação, consideremos $\alpha_{ij} = (l_i + l_j)/2$ e $\beta_{ij} = (w_i + w_j)/2$ e, assim, a restrição (3.64) pode ser reescrita como:

$$|x_{ip} - x_{jq}| - \alpha_{ij} \geq 0 \text{ ou } |y_{ip} - y_{jq}| - \beta_{ij} \geq 0.$$

Portanto, a restrição de sobreposição entre as caixas é dada por:

$$\begin{aligned} \max \quad & [|x_{ip} - x_{jq}| - \alpha_{ij}, |y_{ip} - y_{jq}| - \beta_{ij}] z_{ip} z_{jq} \geq 0 \quad i = 1, \dots, m; j = 1, \dots, m \\ & p = 1, \dots, Q_i; q = 1, \dots, Q_j \\ & (i \neq j \text{ ou } p \neq q), \end{aligned}$$

onde os termos $z_{ip}z_{jq}$ tornam a restrição inativa a menos que $z_{ip} = z_{jq} = 1$ (ou seja, a menos que p -ésima cópia da caixa i e a q -ésima cópia da caixa j sejam colocadas no palete).

Assim como nos modelos anteriores, admitimos apenas dois tipos de caixas com dimensões $(l_1, w_1) = (l, w)$, $(l_2, w_2) = (w, l)$. Logo, temos a seguinte formulação:

$$\text{Max} \sum_{i=1}^m \sum_{p=1}^{Q_i} z_{ip} \quad (3.65)$$

sujeito a:

$$\begin{aligned} \max \quad & [|x_{ip} - x_{jq}| - \alpha_{ij}, |y_{ip} - y_{jq}| - \beta_{ij}] z_{ip} z_{jq} \geq 0 \quad i = 1, \dots, m; j = 1, \dots, m \\ & p = 1, \dots, Q_i; q = 1, \dots, Q_j \\ & (i \neq j \text{ ou } p \neq q) \end{aligned} \quad (3.66)$$

$$z_{ip} = 1 \quad i = 1, \dots, m; P_i > 0; p = 1, \dots, P_i \quad (3.67)$$

$$l_i/2 \leq x_{ip} \leq L - l_i/2 \quad i = 1, \dots, m; p = 1, \dots, Q_i \quad (3.68)$$

$$w_i/2 \leq y_{ip} \leq W - w_i/2 \quad i = 1, \dots, m; p = 1, \dots, Q_i \quad (3.69)$$

$$z_{ip} \in \{0, 1\} \quad i = 1, \dots, m; p = 1, \dots, Q_i \quad (3.70)$$

A função objetivo (3.65) maximiza o valor total das caixas empacotadas. A restrição (3.66) é a restrição de sobreposição, onde (l_i, w_i) são as dimensões da caixa i . Essa restrição se torna inativa, como dito anteriormente, se a p -ésima cópia da caixa i ou a q -ésima cópia da caixa j não são colocadas no palete. A restrição (3.67) garante que será empacotado o número mínimo (P_i) de caixas do tipo i , a restrição (3.68) define os limites da variável x_{ip} , a restrição (3.69) define os limites da variável y_{ip} e a restrição (3.70) é a restrição de integralidade.

Note que, diferentemente do modelo de BEASLEY (1985b) (seção 3.2), as dimensões das caixas não precisam necessariamente ser inteiras. Outro ponto que deve ser notado é que a restrição (3.66) contém redundância pois, para qualquer exemplo particular, a mesma equação matemática aparece duas vezes (por exemplo, para $i=3, p=1, j=7$ e $q=2$ obtemos a mesma equação que para $i=7, p=2, j=3$ e $q=1$).

Entretanto, o autor preserva essa formulação para uso posterior no algoritmo heurístico.

Essa formulação tem $3 \cdot \sum_{i=1}^m Q_i$ variáveis e $O\left(\sum_{i=1}^m Q_i\right)^2$ restrições e, ao contrário de

BEASLEY (1985b), ela não depende do tamanho do palete. O autor não comenta como adaptar a formulação para tratar a orientação das caixas no caso do problema de corte mais geral.

Podemos dizer que essa nova formulação é mais compacta que a formulação anterior de BEASLEY (1985b), principalmente devido à representação da restrição de sobreposição (3.66). Essa característica tem uma desvantagem: a não linearidade da equação. Muitas vezes, dentro da Pesquisa Operacional, termos não lineares devem ser evitados para que os algoritmos tenham sucesso computacional. Neste caso, entretanto, a combinação da representação compacta da restrição de sobreposição e a abordagem de solução (heurística populacional) adotada conduzem a certo sucesso computacional do algoritmo heurístico.

3.7. RELAÇÃO ENTRE OS MODELOS DE TSAI *ET AL.* E CHEN *ET AL.*

Podemos mostrar que a formulação matemática apresentada no trabalho de CHEN *et al.* (1995) (seção 3.5), particularizada para um contêiner ou um palete, está relacionada com a apresentada no trabalho de TSAI *et al.* (1993) (seção 3.3). Consideremos então o conjunto de restrições (3.40)-(3.43) do trabalho de CHEN *et al.* (1995), as quais por conveniência reescrevemos a seguir:

$$x_i + p_i l_{xi} + q_i w_{xi} \leq x_k + M(1 - a_{ik})$$

$$x_k + p_k l_{xk} + q_k w_{xk} \leq x_i + M(1 - b_{ik})$$

$$y_i + q_i w_{yi} + p_i l_{yi} \leq y_k + M(1 - c_{ik})$$

$$y_k + q_k w_{yk} + p_k l_{yk} \leq y_i + M(1 - d_{ik})$$

Inicialmente, consideremos a restrição (3.40). Reescrevendo esta restrição temos:

$$x_i - x_k \leq -p_i l_{xi} - q_i w_{xi} + M(1 - a_{ik})$$

Sabemos que l_{xi} e w_{xi} são variáveis binárias que indicam se a largura ou o comprimento respectivamente da caixa i é paralelo ao eixo x . Dessa forma, temos dois possíveis casos:

- $l_{xi}=1$ e $w_{xi}=0$

Substituindo $l_{xi}=1$ e $w_{xi}=0$ na restrição (3.40) temos:

$$x_i - x_k \leq -p_i + M(1 - a_{ik}).$$

Ainda, sabemos que a variável a_{ik} é igual a 1 se a caixa i está no lado esquerdo da caixa k e é igual a 0 caso contrário. Assim, temos:

$$x_i - x_k \leq -p_i + M \quad \text{se } a_{ik}=1$$

$$x_i - x_k \leq -p_i \quad \text{se } a_{ik}=0.$$

- $l_{xi}=0$ e $w_{xi}=1$

Substituindo $l_{xi}=0$ e $w_{xi}=1$ na restrição (3.40) temos:

$$x_i - x_k \leq -q_i + M(1 - a_{ik}).$$

Ainda, sabemos que a variável a_{ik} é igual a 1 se a caixa i está no lado esquerdo da caixa k e é igual a 0 caso contrário. Assim, temos:

$$x_i - x_k \leq -q_i + M \quad \text{se } a_{ik}=1$$

$$x_i - x_k \leq -q_i \quad \text{se } a_{ik}=0.$$

Analogamente, utilizamos o mesmo raciocínio nas restrições (3.41)-(3.43). Consideremos agora as restrições (3.19)-(3.22) apresentadas na formulação matemática de TSAI *et al.* (1993). Por conveniência, reescrevemos as restrições a seguir:

$$x_j - x_i \leq -l_j + M(u_{ij}^1 + u_{ij}^2)$$

$$x_i - x_j \leq -l_i + M[1 - (u_{ij}^2 - u_{ij}^1)]$$

$$y_j - y_i \leq -w_j + M[1 - (u_{ij}^1 - u_{ij}^2)]$$

$$y_i - y_j \leq -w_i + M[2 - (u_{ij}^1 + u_{ij}^2)]$$

Nas equações (3.19)-(3.22), M é um número suficientemente grande. As possíveis combinações de u_{ij}^1 e u_{ij}^2 e suas possíveis equações de restrições são apresentadas na Tabela 3.1. Analisando esta tabela vemos que para os casos em que $u_{ij}^1 = u_{ij}^2 = 0$; $u_{ij}^1 = u_{ij}^2 = 1$; $u_{ij}^1 = 1$ e $u_{ij}^2 = 0$ o conjunto de restrições resultantes de (3.20)

da formulação de TSAI *et al.* (1993) está relacionado com as restrições (3.40) do trabalho de CHEN *et al.* (1995) no caso de $l_{xi}=1$ e $a_{ik}=1$. No caso de $u_{ij}^1 = 0$ e $u_{ij}^2 = 1$, temos a relação quando consideramos $l_{xi}=1$ e $a_{ik}=0$. Como em CHEN *et al.* (1995) é considerado que $(l, w) = (w, l)$ e, em TSAI *et al.* (1993) que $(l, w) \neq (w, l)$, os mesmos resultados são obtidos, ou seja, podemos dizer que há relação entre as restrições. O mesmo se verifica para as restrições (3.41)-(3.43) do trabalho de CHEN *et al.* (1995) em relação às restrições (3.19), (3.21) e (3.22) do trabalho de TSAI *et al.* (1993). Note também que a função objetivo (3.5) do trabalho de TSAI *et al.* (1993) está relacionada com a função objetivo (3.51) do modelo de CHEN *et al.* (1995). Portanto, podemos dizer que os dois modelos estão relacionados.

CAPÍTULO 4

MÉTODOS DE SOLUÇÃO

4.1. INTRODUÇÃO

Neste capítulo inicialmente fazemos uma revisão da literatura apresentando alguns trabalhos relacionados ao PCP do produtor. São apresentadas também as idéias de partições eficientes e classes de equivalência, pois se dois exemplos possuem o mesmo conjunto de partições eficientes, então ambos possuem o mesmo conjunto de soluções ótimas e, assim, o conjunto de exemplos para o PCP pode ser dividido em classes de equivalência, baseado no conjunto de partições eficientes.

Em seguida, revisamos alguns limitantes superiores para a solução ótima, encontrados na literatura para este problema, pois tais limitantes são úteis em heurísticas, para verificar o *gap* de otimalidade, e em algoritmos exatos, para reduzir o espaço de busca. São apresentadas também as relaxações Linear, Lagrangiana, *Surrogate* e Lagrangiana *Surrogate*, bem como suas aplicações no PCP do produtor utilizando o modelo de BEASLEY (1985b) (modelo (3.1)-(3.4)). A escolha deste modelo do PCP do produtor deveu-se às experiências anteriores da literatura com estas relaxações, com o problema de corte mais geral (BEASLEY, 1985b) e com o PCP do produtor (FARAGO e MORABITO, 2000). Neste último foi estudado um procedimento heurístico baseado em uma heurística Lagrangiana junto com a otimização do

subgradiente. Também consideramos para a escolha do modelo (3.1)-(3.4) nossa experiência anterior resolvendo alguns exemplos ilustrativos com os modelos do Capítulo 3 por meio do pacote GAMS/CPLEX (veja Apêndices). Apesar da limitação destes experimentos, o modelo (3.1)-(3.4) teve bom comportamento em relação aos demais. Além disso, conforme LETCHFORD e AMARAL (2001) e MORABITO e FARAGO (2002), os limitantes superiores obtidos com relaxação Linear e relaxação Lagrangiana e *Surrogate* com este modelo para o PCP do produtor são bem apertados, o que estimula a escolha deste modelo para o presente estudo. O modelo de HADJICONSTANTINOU e CHRISTOFIDES (1995) foi desconsiderado devido a nem sempre ser válido, conforme discutido no Capítulo 3. A aplicação destas técnicas nos outros modelos inteiros apresentados no Capítulo 3 é um tópico para pesquisa futura.

4.2. REVISÃO DA LITERATURA

Na literatura podemos encontrar diversos artigos e edições especiais de revisão do problema do carregamento de paletes, entre eles, HODGSON (1982), RAM (1992), SWEENEY e PARTERNOSTER (1992), DYCKHOFF e FINKE (1992), MARTELLO (1994), CHENG *et al.* (1994), BISCHOFF e WAESCHER (1995), DYCKHOFF *et al.* (1997), ARENALES *et al.* (1999), WANG e WAESCHER (2002), HIFI (2002) e SICUP (2004).

Poucos autores propuseram métodos exatos para resolver o PCP do produtor, entre eles, DOWSLAND (1987), TSAI *et al.* (1993) e BATTACHARYA *et al.* (1998). DOWSLAND (1987) apresentou um algoritmo exato que consiste em encontrar o máximo conjunto estável de um grafo, representando o carregamento do palete (um conjunto estável de um grafo G é definido como um subconjunto de nós de G com a condição de que não haja dois nós ligados por um arco). O algoritmo funciona bem para problemas em que o número de caixas por camada não é grande, por exemplo, é menor do que 30, o que nem sempre corresponde aos casos práticos. TSAI *et al.* (1993) formularam o PCP do produtor como um problema 0–1 com restrições disjuntivas (múltipla escolha, seção 3.3) e sugeriram resolvê-lo por algoritmos do tipo *branch and bound* explorando a estrutura particular do problema inteiro; porém, o número dessas restrições cresce exponencialmente com o número de caixas disponíveis.

BATTACHARYA *et al.* (1998) propuseram um algoritmo de busca em árvore baseado em empacotar caixas não individuais, mas em partições eficientes do problema parcial, em cada passo da busca. Recentemente, MARTINS (2002) mostrou que o método proposto por BATTACHARYA *et al.* (1998) não é exato. Nenhum destes estudos apresentou extensivos experimentos computacionais com seus métodos propostos, o que dificulta sua 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 e DE CANI (1980), BISCHOFF e DOWSLAND (1982), DOWSLAND (1993), NELISSEN (1994, 1995), ARENALES e MORABITO (1995), SCHEITHAUER e TERNO (1996), SCHEITHAUER e SOMMERWEISS (1998), MORABITO e MORALES (1998, 1999), FARAGO e MORABITO (2000), G e KANG (2001), MARTINS (2002), LINS *et al.* (2003), PUREZA e MORABITO (2003, 2004), ALVAREZ-VALDES *et al.* (2004) e BIRGIN *et al.* (2004). LINS *et al.* (2003) apresentam uma abordagem explorando partições na forma de L , e conjecturam que esta abordagem é exata para o PCP do produtor. A desvantagem desta abordagem são os relativamente grandes requisitos de tempo computacional, o que foi atenuado no estudo recente em BIRGIN *et al.* (2004), porém com maiores requisitos de memória computacional. Limitantes superiores para o problema são estudados em NELISSEN (1995), LETCHFORD e AMARAL (2001) e MORABITO e FARAGO (2002). Em particular, o trabalho de FARAGO e MORABITO (2000) apresenta um método heurístico baseado em relaxação Lagrangiana e *Surrogate* no modelo de BEASLEY (1985b) para resolver o PCP do produtor. Também é aplicado um método de redução do problema e uma heurística Lagrangiana no procedimento de otimização do subgradiente, similarmente ao presente trabalho. MORABITO e FARAGO (2002) analisaram o desempenho de limitantes superiores obtidos por relaxação Lagrangiana no PCP do produtor. Os limitantes obtidos são bem apertados e foram capazes de garantir um certificado de otimalidade para quase todos os exemplos analisados, o que estimula a pesquisa com aplicação de relaxações sobre o modelo de BEASLEY (1985b).

Diversos autores têm tratado o problema do PCP produtor por heurísticas de construção conhecidas como heurísticas de blocos, onde são gerados padrões de

carregamento compostos por um ou mais blocos, cujas peças possuem a mesma orientação. Exemplos de heurísticas de blocos podem ser encontrados em STEUDEL (1979), SMITH e DE CANI (1980), BISCHOFF e DOWSLAND (1982), NELISSEN (1994, 1995), SCHEITHAUER e TERNO (1996) e MORABITO e MORALES (1998, 1999). Uma característica desses métodos é que os padrões gerados estão limitados aos padrões não-guilhotinados de 1ª ordem.

Apesar do sucesso que as meta-heurísticas (técnicas que, quando aplicadas a métodos de busca local, permitem a superação da otimalidade local com vistas à obtenção de soluções de qualidade superior) têm alcançado em uma variedade de problemas, em relação ao problema do produtor, poucas aplicações são encontradas na literatura; por exemplo, DOWSLAND (1993) (*simulated annealing*), HERBERT e DOWSLAND (1996) (algoritmos genéticos), DOWSLAND (1996), YAMASSAKI e PUREZA (2003) e PUREZA e MORABITO (2003, 2004) (busca tabu). Em particular, os trabalhos de DOWSLAND (1996), YAMASSAKI e PUREZA (2003) utilizam uma versão sem memória de busca tabu que parte de uma solução com o número ótimo de caixas e ineficaz com respeito à sobreposição, e realiza movimentos destas caixas para novas posições com vistas à sobreposição zero (padrão ótimo). O trabalho de PUREZA e MORABITO (2003, 2004) propõe uma abordagem baseada na incorporação de busca tabu simples (ou seja, sem estruturas de memória de longo prazo) em heurísticas de bloco capaz de produzir bons padrões não-guilhotinados de ordem superior.

Na literatura também encontramos trabalhos que tratam outros problemas não-guilhotinados mais gerais como, por exemplo, corte bidimensional, carregamento tridimensional, etc., mas que podem ser úteis para o carregamento de paletes. Podemos citar, entre eles, BEASLEY (1985a, 1985b), ARENALES e MORABITO (1995), HADJICONSTANTINOU e CHRISTOFIDES (1995), CHEN *et al.* (1995) e BEASLEY (2004); alguns destes baseados em modelos inteiros revisados no Capítulo 3.

4.3. PARTIÇÕES EFICIENTES E CLASSES DE EQUIVALÊNCIA

Esta seção baseia-se nos estudos apresentados em LETCHFORD e AMARAL (2001) e MARTINS (2002). A idéia de partições eficientes (BISCHOFF e

DOWSLAND, 1982) vem da observação de que alguns problemas com dimensões diferentes possuem o mesmo *layout* de caixas na solução ótima. Assim, consideremos a seguir para identificar um exemplo do PCP, a quádrupla (L, W, l, w) que representa as dimensões do palete (L, W) e da caixa (l, w) . Então, como exemplo, temos que o *layout* da solução ótima do exemplo $(22, 16, 5, 3)$ é o mesmo para os exemplos $(30, 22, 7, 4)$, $(50, 36, 11, 7)$ e muitos outros exemplos. A Figura 4.1 mostra o *layout* do exemplo $(22, 16, 5, 3)$ e a área hachurada indica o espaço não utilizado no palete.

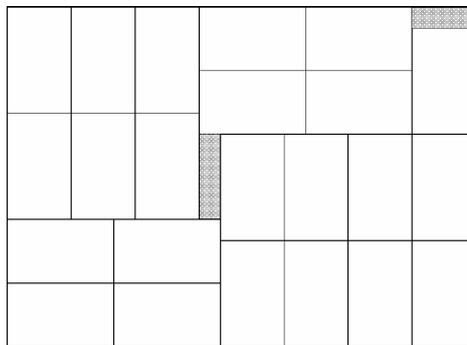


Figura 4.1. *Layout* ótimo para o exemplo $(22, 16, 5, 3)$, $(30, 22, 7, 4)$, $(50, 36, 11, 7)$ e todos os exemplos com a mesma classe de equivalência (MARTINS, 2002).

Se desenharmos uma linha horizontal ou vertical pelo palete carregado, como na Figura 4.2, este cruza um número de caixas. Se a dimensão das caixas, cruzadas pela linha é l , chamamos o segmento da linha contido na caixa de segmento l . Caso contrário, será chamado de segmento w . A soma dos segmentos cruzados pela linha vertical ou horizontal deve ser menor que a dimensão do palete. Na Figura 4.2, a linha horizontal H-H' cruza 4 caixas, 2 na dimensão l e 2 na dimensão w . A linha vertical V-V' cruza 5 caixas sendo 1 na dimensão l e 4 na dimensão w .

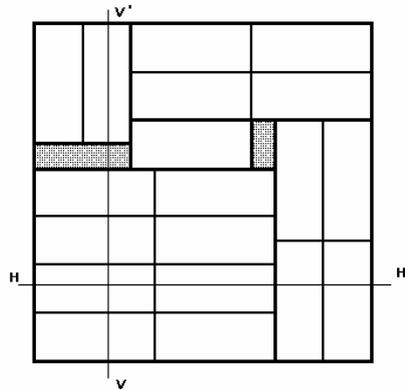


Figura 4.2. Exemplo da identificação de segmentos em um palete carregado (MARTINS, 2002).

Se considerarmos as possíveis combinações dos segmentos l e w obtidos para cada dimensão independentemente, podemos definir um conjunto de desigualdades que satisfeitas produzem um *layout* factível, ou seja, um conjunto de desigualdades que assegure uma combinação de caixas na horizontal e vertical não excedendo o comprimento ou largura do palete. Essas desigualdades são chamadas de partições factíveis. Seja (n, m) um par ordenado de inteiros não-negativos satisfazendo

$$n * l + m * w \leq S$$

para uma dimensão de palete S , o qual poderia ser L ou W . Então, o par ordenado (n, m) é chamado de partição factível de S , ou seja, uma combinação de segmentos de comprimentos l e w que não excedem a dimensão S é uma partição factível de S . Se n e m também satisfazem

$$0 \leq S - n * l - m * w < w$$

então (n, m) é chamado de partição eficiente de S (BISCHOFF e DOWSLAND, 1982). Uma partição eficiente não pode ser aumentada pela adição de uma outra caixa sem tornar-se infactível. Podemos definir uma partição eficiente para cada n satisfazendo

$$0 \leq n \leq \lfloor S/l \rfloor.$$

Como mostrado por DOWSLAND (1984), se dois exemplos para o PCP possuem o mesmo conjunto de partições eficientes, para o comprimento e largura do palete, então ambos os exemplos possuem o mesmo conjunto de soluções ótimas. Assim, o conjunto de exemplos para o PCP pode ser dividido em classes de equivalência, baseado no conjunto de partições eficientes. Se a solução é conhecida para uma classe representativa, então esta solução pode ser usada em qualquer outro exemplo da classe. Isto porque a multiplicação de todas as dimensões por um valor inteiro pode

produzir um novo exemplo, na mesma classe, e cada classe possui uma infinidade de exemplos. Se, em adição, n e m satisfazem

$$n * l + m * w = S$$

então (n, m) é chamado de partição perfeita de S (DOWSLAND, 1984). Um exemplo em uma classe de equivalência, com dimensões mínimas, contém pelo menos uma partição perfeita para cada dimensão, L e W (DOWSLAND, 1984). Isto pode ser facilmente observado se considerarmos um exemplo arbitrário sem partição perfeita, para uma dada dimensão. Neste caso, podemos reduzir a correspondente dimensão do palete, sem alterar o conjunto de partições eficientes. Isto implica que um novo exemplo, com dimensões menores, também pertence a mesma classe.

O conceito de classe de equivalência está relacionado com os conjuntos de exemplos testes usados para o PCP do produtor. Para avaliar o desempenho dos métodos exatos com exemplos conhecidos da literatura, consideramos os conjuntos Cover I e Cover II, cujos resultados são apresentados no Capítulo 6. O conjunto Cover I contém 8274 exemplos de tamanho moderado satisfazendo $1 \leq \frac{LW}{lw} \leq 50$ (NELISSEN, 1994; SCHEITHAUER e TERNO, 1996; MORABITO e MORALES, 1998) e, este conjunto de dados contém um representante de cada uma das 8274 classes de equivalência para o qual o valor da solução ótima está no intervalo $[1, 50]$. Por simplicidade, neste trabalho consideramos somente exemplos com $L, W \leq 1000$, em um total de 3179 exemplos. O conjunto Cover II contém 41831 exemplos satisfazendo $50 \leq \frac{LW}{lw} \leq 100$ (NELISSEN, 1994; SCHEITHAUER e TERNO, 1996; MORABITO e MORALES, 1998). Similarmente ao Cover I, também por simplicidade, consideramos somente exemplos em Cover II com $L, W \leq 1000$, resultando em um total de 16938 exemplos.

4.4. ALGUNS LIMITANTES SUPERIORES PARA A SOLUÇÃO ÓTIMA

A existência de um limitante superior apertado para o número máximo de caixas que podem ser empacotadas em um exemplo do PCP é útil em heurísticas para verificar a otimalidade, e em algoritmos exatos, pois permitem o uso de limitantes na

redução do espaço de solução. Nesta seção, será apresentada uma revisão de alguns limitantes encontrados na literatura, baseada nos trabalhos de NELISSEN (1995), LETCHFORD e AMARAL (2001), MORABITO e FARAGO (2002) e MARTINS (2002).

4.4.1. LIMITANTES SIMPLES

O mais simples e mais intuitivo limitante é obtido considerando a parte inteira da razão entre as áreas do palete e das caixas, ou seja, $\lfloor (L*W)/(l*w) \rfloor$, o qual é chamado de limitante da Razão da Área. Este limitante é muito simples de ser computado, mas SMITH e DE CANI (1980) e DOWSLAND (1985) reportaram que este é igual a solução ótima em menos de 15% dos problemas testes gerados aleatoriamente.

Outro limitante intuitivo é obtido multiplicando o número máximo de caixas que podem ser colocadas em qualquer linha horizontal, e o máximo número de caixas em uma coluna vertical. Se w é relativamente próximo de l , este limitante, chamado de Produto Máximo, pode ser melhor que o limitante da Razão da Área. O exemplo (23,23,5,4) (MARTINS, 2002) é um exemplo desta situação pois o limitante da Razão da Área é $\lfloor (23*23)/(5*4) \rfloor = 26$ e o limitante de Produto Máximo é $\lfloor 23/4 \rfloor * \lfloor 23/4 \rfloor = 25$.

4.4.2. LIMITANTES DE BARNES

BARNES (1979) propôs um limitante diferente. Ele observou que padrões com $l \times w$ caixas podem ser representados por padrões de caixas $l \times 1$ ou $w \times 1$. Por causa da solução para problemas com largura unitária ser obtida facilmente, como mostrado por BARNETT e KYNCH (1967), Barnes computou a área perdida ao empacotar caixas $l \times 1$ ou $w \times 1$ e usou esta área como um limitante inferior na área perdida do problema original, produzindo um limitante superior no número de caixas empacotadas. Barnes também provou que se as dimensões do palete são “suficientemente grandes”, então a área perdida no melhor empacotamento possível é

igual ao limitante. Assim, o máximo número $n_{l \times 1}$ de $l \times 1$ faixas que podem ser ortogonalmente colocadas em um palete $L \times W$ é dado por:

$$n_{l \times 1} = \begin{cases} (LW - (L \bmod l)(W \bmod l))/l & \text{se } L \bmod l + W \bmod l \leq l \\ (LW - (l - L \bmod l)(l - W \bmod l))/l & \text{c.c.} \end{cases},$$

onde *mod* é a abreviação de módulo, e $L \bmod l$ significa o resto inteiro obtido quando L

é dividido por l . Como cada caixa consiste de w faixas, a quantidade $UB(l) = \left\lfloor \frac{n_{l \times 1}}{w} \right\rfloor$ é

um limitante superior válido para o PCP. Analogamente, o máximo número $n_{w \times 1}$ de $w \times 1$ faixas que podem ser ortogonalmente colocadas em um palete $L \times W$ é dado por:

$$n_{w \times 1} = \begin{cases} (LW - (L \bmod w)(W \bmod w))/w & \text{se } L \bmod w + W \bmod w \leq w \\ (LW - (w - L \bmod w)(w - W \bmod w))/w & \text{c.c.} \end{cases},$$

e, portanto, a quantidade $UB(w) = \left\lfloor \frac{n_{w \times 1}}{l} \right\rfloor$ é também um limitante superior válido para o

problema de carregamento de palete. O mínimo entre $UB(l)$ e $UB(w)$ nós dá o limitante de Barnes.

Consideremos o exemplo (22,18,4,3) como uma aplicação deste limitante. O limitante da Razão da Área é dado por $\lfloor (22*18)/(4*3) \rfloor = 33$, o qual não produz área perdida. Os cálculos exigidos para a aplicação do limitante de Barnes são descritos a seguir. Inicialmente precisamos calcular o número $n_{l \times 1}$ de $l \times 1$ faixas que podem ser ortogonalmente colocadas em um palete $L \times W$, então como $22 \bmod 4 + 18 \bmod 3 = 2 \leq 4$ temos que $n_{l \times 1} = (22*18 - (22 \bmod 4)*(18 \bmod 3))/4 = 98$, o

que nos dá $UB(l) = \left\lfloor \frac{98}{3} \right\rfloor = 32$. Analogamente, precisamos calcular o máximo número

$n_{w \times 1}$ de $w \times 1$ faixas que podem ser ortogonalmente colocadas em um palete $L \times W$,

assim como $22 \bmod 3 + 18 \bmod 3 = 1 \leq 3$ temos que

$n_{w \times 1} = (22*18 - (22 \bmod 3)*(18 \bmod 3))/3 = 131,67$ e temos o seguinte limitante

superior, $UB(w) = \left\lfloor \frac{131,67}{4} \right\rfloor = 32$. O valor mínimo entre $UB(l)$ e $UB(w)$ nós dá o

limitante de Barnes, o qual é 32.

4.4.3. LIMITANTES DE EXEMPLOS EQUIVALENTES

O limitante de Área e o de Barnes podem ser melhorados usando a noção de partições factíveis e exemplos equivalentes (BISCHOFF e DOWSLAND, 1982). Uma partição factível de L é um par ordenado (n,m) de inteiros não negativos tal que $n * l + m * w \leq L$; partições factíveis de W são definidas similarmente. Dois exemplos do PCP são equivalentes se o conjunto de partições factíveis de L e W são os mesmos para ambos os problemas. Ainda, exemplos equivalentes possuem a mesma solução ótima.

Mesmo que dois exemplos para o PCP sejam equivalentes, eles podem ter diferentes limitantes de Área, ou de Barnes ou ainda de ambos. Por exemplo, o limitante de Área e de Barnes para o exemplo $(9,9,5,5)$ (MARTINS, 2002) são ambos 3, mas para o exemplo equivalente $(1,1,1,1)$ são ambos 1. Este fato é usado por DOWSLAND (1985) para melhorar o limitante de Área e de Barnes e é dado a seguir. Dado o exemplo (L,W,l,w) , definimos:

$$L' = \max \{n * l + m * w \mid (n,m) \text{ é uma partição factível de } L\};$$

$$W' = \max \{n * l + m * w \mid (n,m) \text{ é uma partição factível de } W\}.$$

Assim, (L,W,l,w) é equivalente a (L',W',l,w) e $L' \leq L$ e $W' \leq W$. Esta redução é conhecida como redução de partição perfeita. Para o exemplo considerado anteriormente, $(9,9,5,5)$, a redução de partição perfeita nos dá $(5,5,5,5)$, e como efeito temos que o limitante de Área e de Barnes reduz de 3 para 1, o qual é o ótimo.

4.4.4. LIMITANTES DE EXEMPLOS RELAXADOS

Podemos dizer que não é necessário transformar um exemplo em um exemplo equivalente para obter um limitante válido. Um exemplo que é uma relaxação do exemplo original também pode ser usado. Um exemplo do PCP I_1 é uma relaxação do exemplo I_2 se o conjunto de partições factíveis de I_1 é um subconjunto do conjunto de partições factíveis de I_2 . Ainda, um limitante de um exemplo relaxado pode ser mais forte do que um limitante equivalente de um exemplo. Por exemplo, o limitante de Barnes para o exemplo $(19,19,7,2)$ (LETCHFORD e AMARAL, 2001) é 25, mas para o exemplo relaxado $(10,10,4,1)$ o limitante é 24.

4.4.5. LIMITANTE DE EMPACOTAMENTO

Este limitante superior é baseado na formulação do PCP do produtor apresentado no Capítulo 3 em (3.1)-(3.4). Podemos associar um grafo com a formulação (3.1)-(3.4) definindo um vértice para cada variável, e um arco entre dois vértices se e somente se duas variáveis aparecem juntas na restrição (3.2). Assim, o PCP é um problema de conjunto estável de máxima cardinalidade e cada restrição individual (3.2) corresponde a um clique no grafo (DOWSLAND, 1987). Substituindo as restrições de integralidade (3.4) por

$$x_{ipq} \geq 0, \quad i=1,2, \quad p \in X \text{ tal que } p \leq L-l_i, \quad q \in Y \text{ tal que } q \leq W-w_i \quad (4.1)$$

obtemos uma relaxação de Programação Linear do problema inteiro. Então, considerando o maior inteiro menor que o valor ótimo da relaxação por Programação Linear temos um limitante superior para o PCP, o qual é chamado de limitante de empacotamento devido a formulação ser do tipo padrão do conjunto de empacotamento e foi estudado por LETCHFORD e AMARAL (2001).

O problema linear (3.1)-(3.3) e (4.1) pode ser bastante grande na prática e isto levou alguns autores a computar apenas uma aproximação para o limitante de empacotamento usando relaxação Lagrangiana e otimização do subgradiente (BEASLEY, 1985b; HADJICONSTANTINOU e CHRISTOFIDES, 1995; MORABITO e FARAGO, 2002). No entanto, o problema linear pode ser bem reduzido, sem perder a solução ótima, eliminando algumas variáveis usando o conceito de *raster points* (SCHEITHAUER e TERNIO, 1996; MORABITO e MORALES, 1998; ALVAREZ-VALDES *et al.* 2004).

Segundo MORABITO e MORALES (1998) tem-se que os pontos (p, q) , sem perda de generalidade, devem fazer parte dos conjuntos normais X e Y (conforme Capítulo 3) ou ainda, fazer parte dos conjuntos *raster points* X' e Y' dados por:

$$X' = \{(L-p)_x \mid p \in X\} \cup \{0\}$$

$$(p')_x = \max\{p \mid p \in X, p \leq p'\}$$

$$Y' = \{(W-q)_y \mid q \in Y\} \cup \{0\}$$

$$(q')_y = \max\{q \mid q \in Y, q \leq q'\}.$$

Para ilustrar o conjunto dos *raster points* consideremos o exemplo (11,10,4,3). Temos que os conjuntos normais X e Y para este exemplo são dados por $X = \{0,3,4,6,7,8\}$ e $Y = \{0,3,4,6,7\}$. Assim, os conjuntos dos *raster points* (X' e Y') para estes exemplos são $X' = \{0,3,4,7,8\}$ e $Y' = \{0,3,4,6,7\}$.

Recentemente, o trabalho de ALVAREZ-VALDES *et al.* (2004) propõe um algoritmo *branch and cut* para o PCP e utiliza a formulação de BEASLEY (1985b) adaptada ao problema, adicionando novas restrições e novos procedimentos para a redução de variáveis e restrições. O procedimento de redução de variáveis é capaz de reduzir ainda mais o conjunto dos *raster points* e tal procedimento se baseia na idéia de posições nos conjuntos X' e Y' não dominadas pelas variáveis. Para a redução de restrições é considerado a restrição correspondente a um quadrado unitário (r,s) . Se todas as variáveis envolvidas na restrição aparecem em uma restrição prévia (r',s') (para a esquerda e/ou para o canto de (r,s)), ou seja, se para cima e/ou a direita de (r',s') para (r,s) nenhuma nova variável aparecer, então a restrição (r,s) é redundante. Neste caso, a restrição evitando a sobreposição de (r',s') é também responsável por evitar a sobreposição de (r,s) .

4.5. RELAXAÇÃO DE PROGRAMAÇÃO LINEAR, LAGRANGIANA, SURROGATE

A relaxação de um problema tem como finalidade tornar o problema mais fácil de se resolver. No entanto, a solução fornecida pela relaxação geralmente não satisfaz todas as restrições do problema original. Mas, através do problema relaxado, obtemos um limitante para a solução ótima. Se o problema for de maximização (minimização), o valor da função objetivo fornecerá um limitante superior (inferior) para a solução ótima do problema.

A seguir, são apresentadas as relaxações Linear, Lagrangiana e *Surrogate*, as quais são bastante conhecidas e amplamente divulgadas na literatura. São apresentados com mais detalhes as relaxações Lagrangiana e *Surrogate* por serem de maior interesse neste trabalho. Todas as relaxações são apresentadas para o modelo do PCP do produtor apresentado no Capítulo 3 em (3-1)-(3.4). Estas relaxações também

podem ser aplicadas para os outros modelos 0–1 apresentados no Capítulo 3, mas isto está além do escopo do presente trabalho.

4.5.1. RELAXAÇÃO DE PROGRAMAÇÃO LINEAR

Consideremos o seguinte problema 0–1 (escrito em notação matricial) o qual pode ser referenciado como problema (P):

$$\text{Max } cx$$

sujeito a:

$$Ax \leq b$$

$$Bx \leq d$$

$$x \in \{0,1\}$$

Uma maneira de gerar um limitante superior (Z_{UB}) é relaxar o problema através de uma relaxação por Programação Linear (PL), isto é, ao invés de considerarmos a restrição de integralidade ($x \in \{0,1\}$), consideramos um conjunto maior para os valores das variáveis, trocando-a por sua relaxação Linear ($0 \leq x \leq 1$). A grande vantagem desta relaxação é a existência de algoritmos eficientes para a solução de problemas de Programação Linear, como por exemplo, o método simplex ou o método de pontos interiores, e a solução nos dá um limitante superior para o problema original acima (P) (maximização).

Em muitos casos, no entanto, resolver uma relaxação por Programação Linear é inviável na prática, pois o problema (P) pode envolver um número grande (às vezes, extremamente grande) de variáveis e/ou restrições. Por exemplo, no caso do PCP do produtor, este número chega facilmente a ordem de milhares, ou dezenas de milhares, dependendo do tamanho dos conjuntos X e Y , o que tornaria o cálculo do limitante muito custoso.

4.5.1.1. APLICAÇÃO DE RELAXAÇÃO POR PROGRAMAÇÃO LINEAR NO PCP DO PRODUTOR

A relaxação Linear do modelo (3.1)-(3.4) descrito na seção anterior, como dito anteriormente, consiste em relaxar a condição de integralidade de x , ou seja, o domínio de x será alterado de $\{0,1\}$ para $0 \leq x \leq 1$. Assim, a relaxação Linear do modelo (3.1)-(3.4) é dado a seguir:

$$\text{Max} \quad \sum_{i=1}^2 \sum_{\{p \in X | p \leq L-l_i\}} \sum_{\{q \in Y | q \leq W-w_i\}} x_{ipq} \quad (4.2)$$

sujeito a:

$$\sum_{i=1}^2 \sum_{\{p \in X | p \leq L-l_i\}} \sum_{\{q \in Y | q \leq W-w_i\}} a_{ipqrs} x_{ipq} \leq 1, \text{ para todo } r \in X, s \in Y \quad (4.3)$$

$$P \leq \sum_{i=1}^2 \sum_{\{p \in X | p \leq L-l_i\}} \sum_{\{q \in Y | q \leq W-w_i\}} x_{ipq} \leq Q \quad (4.4)$$

$$0 \leq x_{ipq} \leq 1, i=1,2, p \in X \text{ tal que } p \leq L-l_i; q \in Y \text{ tal que } q \leq W-w_i \quad (4.5)$$

No PCP do produtor, em geral, a restrição (4.4) é redundante, mas também pode ser útil para produzir padrões mais estáveis se for escrita para cada orientação i da caixa (veja discussão na seção 3.2).

4.5.2. RELAXAÇÃO LAGRANGIANA (LAG)

O “nascimento” da relaxação Lagrangiana (LAG) tal como é conhecido hoje ocorreu em 1970 quando HELD e KARP (1970, 1971) usaram a relaxação Lagrangiana para fornecer um limite inferior para o problema do caixeiro viajante. Motivados pelo sucesso da relaxação Lagrangiana, esse método foi aplicado no início da década de 70 em problemas de Programação Inteira em geral (SHAPIRO, 1971). Segundo Beasley (REEVES, 1993), encontrar boas soluções para problemas difíceis em otimização combinatória exige a consideração do cálculo do limitante superior, que deve ser tão próximo quanto possível do ótimo, e do cálculo do limitante inferior, que também deve ser tão próximo quanto possível do ótimo.

Técnicas para gerar bons limitantes são essencialmente métodos heurísticos. Além disso, para qualquer problema particular, podemos ter técnicas as quais são específicas para o problema sendo resolvido. Uma técnica bem conhecida e usada para o cálculo do limitante é a relaxação Lagrangiana a qual envolve:

- considerar uma formulação de Programação Inteira ou Inteira-mista do problema;
- anexar os multiplicadores de Lagrange para algumas restrições nesta formulação, relaxando-as e inserindo-as na função objetivo;
- resolver o problema inteiro ou inteiro-misto resultante.

Após a resolução do problema inteiro ou inteiro-misto obtém-se um limitante inferior ou superior na solução ótima para o problema original, dependendo se o problema em questão é de minimização ou maximização, respectivamente. Ainda segundo Beasley (REEVES, 1993), existem algumas razões que justificam o uso da técnica de relaxação Lagrangiana. Muitos problemas de otimização combinatória consistem em um problema "fácil", complicado pela adição de restrições extras. A absorção destas restrições complicadas dentro da função objetivo, faz com que possa ser obtido um problema fácil de resolver e a atenção pode então ser voltada para a escolha de valores numéricos para os multiplicadores de Lagrange. Além disso, experiências práticas com relaxação Lagrangiana têm indicado que esta fornece muitos limitantes bons e de razoável custo computacional.

A escolha de valores para os multiplicadores de Lagrange é a chave importante na qualidade dos limitantes gerados e, para isto, pode ser utilizado o método de otimização do subgradiente. Podemos dizer ainda que o limitante obtido por relaxação Lagrangiana é melhor ou igual ao obtido por relaxação de Programação Linear. Se há a propriedade de integralidade, ou seja, se a solução do problema relaxado não varia com as restrições de integralidade relaxadas, podemos dizer que teoricamente o limitante obtido por relaxação Lagrangiana é igual ao obtido por Programação Linear. É muito extensa a lista de aplicações de relaxação Lagrangiana, assim como a amplitude do tipo de problemas que são abordados por esta técnica. Neste trabalho vemos a aplicação desta relaxação ao modelo (3.1)-(3.4) apresentado no capítulo anterior.

Voltemos ao problema (P). Obtemos a relação Lagrangiana do problema (P), com respeito ao conjunto de restrições $Ax \leq b$, com a introdução de um vetor

multiplicador de Lagrange ($\lambda \geq 0$), o qual é incorporado a este conjunto de restrições e inserido na função objetivo, resultando em:

$$\text{Max } cx + \lambda(b - Ax)$$

sujeito a:

$$Bx \leq d$$

$$x \in \{0,1\}, \lambda \geq 0$$

O que devemos observar é que o problema que obtemos depois de aplicar a relaxação Lagrangiana, para quaisquer $\lambda \geq 0$, nos dá um limitante superior para a solução ótima do problema original (maximização). Isso se deve ao fato de que a solução do problema original (P):

$$\text{Max } cx$$

sujeito a:

$$Ax \leq b$$

$$Bx \leq d$$

$$x \in \{0,1\}$$

tem valor menor ou igual a solução do problema

$$\text{Max } cx + \lambda(b - Ax)$$

sujeito a:

$$Ax \leq b$$

$$Bx \leq d$$

$$x \in \{0,1\}, \lambda \geq 0$$

Observe que sendo $\lambda \geq 0$ e $(b - Ax) \geq 0$, temos que $\lambda(b - Ax) \geq 0$. A solução do problema acima tem valor menor ou igual a solução do problema:

$$\text{Max } cx + \lambda(b - Ax) = (c - \lambda A)x + \lambda b$$

sujeito a:

$$Bx \leq d$$

$$x \in \{0,1\}, \lambda \geq 0$$

uma vez que o problema se torna menos restrito. Por este motivo, a solução deste último problema nos dá um limitante superior (Z_{UB}) para a solução do primeiro problema. Resta agora encontrar os multiplicadores de Lagrange que minimizem o limitante superior para o nosso problema de maximização original, ou seja:

$$\text{Min}_{\lambda \geq 0} \left\{ \begin{array}{l} \text{Max } (c - \lambda A)x + \lambda b \\ \text{s.a : } Bx \leq d \\ x \in \{0,1\} \end{array} \right\},$$

o qual é chamado problema dual Lagrangiano.

4.5.2.1. APLICAÇÃO DE RELAXAÇÃO LAGRANGIANA NO PCP DO PRODUTOR

Voltemos ao modelo do PCP do produtor definido em (3.1)-(3.4). Introduzindo os multiplicadores de Lagrange λ_{rs} (≥ 0) para todo $r \in X$ e todo $s \in Y$ da expressão (3.2), obtemos o seguinte problema Lagrangiano do limitante superior:

$$\text{Max } \sum_1^2 \sum_{\{p \in X | p \leq L-l_i\}} \sum_{\{q \in Y | q \leq W-w_i\}} x_{ipq} +$$

$$\sum_{r \in X} \sum_{s \in Y} \lambda_{rs} \left(1 - \sum_1^2 \sum_{\{p \in X | p \leq L-l_i\}} \sum_{\{q \in Y | q \leq W-w_i\}} a_{ipqrs} x_{ipq} \right)$$

sujeito a:

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

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

que pode ser reescrito por:

$$\text{Max} \quad \sum_{i=1}^2 \sum_{\{p \in X | p \leq L-l_i\}} \sum_{\{q \in Y | q \leq W-w_i\}} V_{ipq} x_{ipq} + \sum_{r \in X} \sum_{s \in Y} \lambda_{rs} \quad (4.6)$$

sujeito a:

$$P \leq \sum_{i=1}^2 \sum_{\{p \in X | p \leq L-l_i\}} \sum_{\{q \in Y | q \leq W-w_i\}} x_{ipq} \leq Q \quad (4.7)$$

$$x_{ipq} \in \{0,1\}, \quad i=1,2, \quad p \in X \text{ tal que } p \leq L-l_i; \quad q \in Y \text{ tal que } q \leq W-w_i \quad (4.8)$$

onde $V_{ipq} = 1 - \sum_{r \in X} \sum_{s \in Y} \lambda_{rs} a_{ipqrs}$.

Observe que, neste caso, teremos $|X||Y|$ multiplicadores, ou seja, um multiplicador para cada restrição. Sejam $\{x_{ipq}\}$ representando os valores das variáveis $\{x_{ipq}\}$ na solução do problema Lagrangiano (4.6)-(4.8), com valor dado por:

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

que corresponde a um limitante superior para o valor ótimo do problema original (3.1)-(3.4), para quaisquer $\{\lambda_{rs}\}$ não negativos. Note que, dados $\{\lambda_{rs}\}$, o problema (4.6)-(4.8) 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. Note que o problema Lagrangiano (4.6)-(4.8) 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 (4.7) são iguais a 1, e P e Q são números inteiros). Isto implica que o melhor limitante superior produzido pelo problema Lagrangiano (4.6)-(4.8) para o problema (3.1)-(3.4) não é melhor do que o limitante superior obtido pela relaxação de Programação Linear do problema (3.1)-(3.4).

No entanto, estamos interessados em encontrar os valores para os multiplicadores de Lagrange que minimizem o limitante superior, isto é, o limitante

superior que seja tão próximo quanto possível do valor ótimo. Isto envolve encontrar multiplicadores que correspondam a:

$$\left. \begin{array}{l} \text{Max} \sum_{i=1}^2 \sum_{\{p \in X | p \leq L - l_i\}} \sum_{\{q \in Y | q \leq W - w_i\}} V_{ipq} x_{ipq} + \sum_{r \in X} \sum_{s \in Y} \lambda_{rs} \\ \text{Min}_{\lambda_{rs} \geq 0} \text{s.a: } P \leq \sum_{i=1}^2 \sum_{\{p \in X | p \leq L - l_i\}} \sum_{\{q \in Y | q \leq W - w_i\}} x_{ipq} \leq Q \\ x_{ipq} \in \{0, 1\}, i = 1, 2, p \in X \text{ tal que } p \leq L - l_i; q \in Y \text{ tal que } q \leq W - w_i \end{array} \right\},$$

ou seja, $\min_{\lambda_{rs} \geq 0} \{\text{problema Lagrangiano (4.6)-(4.8)}\}$, o qual é chamado problema dual Lagrangiano.

Ilustração através do exemplo didático

Utilizamos aqui novamente o exemplo numérico da seção 3.2 do Capítulo 3, com o objetivo de ilustrar a aplicação de relaxação Lagrangiana. Assim temos:

$$\begin{aligned} \text{Max } & x_{100} + x_{102} + x_{120} + x_{122} + x_{200} + x_{220} + x_{230} + \lambda_{00}(1 - x_{100} - x_{200}) + \\ & \lambda_{02}(1 - x_{102} - x_{200}) + \lambda_{20}(1 - x_{100} - x_{120} - x_{220}) + \lambda_{22}(1 - x_{102} - x_{122} - x_{220}) + \\ & \lambda_{30}(1 - x_{120} - x_{220} - x_{230}) + \lambda_{32}(1 - x_{122} - x_{220} - x_{230}) \end{aligned}$$

sujeito a:

$$x_{100} + x_{102} + x_{120} + x_{122} + x_{200} + x_{220} + x_{230} \leq 3$$

$$x_{ipq} \in \{0, 1\}$$

onde $\lambda_{00}, \lambda_{02}, \lambda_{20}, \lambda_{22}, \lambda_{30}, \lambda_{32}$ correspondem aos multiplicadores de Lagrange associados ao conjunto de restrições.

Reescrevendo o problema acima temos:

$$\begin{aligned} \text{Max } & (1 - \lambda_{00} - \lambda_{20})x_{100} + (1 - \lambda_{02} - \lambda_{22})x_{102} + (1 - \lambda_{20} - \lambda_{30})x_{120} + \\ & (1 - \lambda_{22} - \lambda_{32})x_{122} + (1 - \lambda_{00} - \lambda_{02})x_{200} + (1 - \lambda_{20} - \lambda_{22} - \lambda_{30} - \lambda_{32})x_{220} + \\ & (1 - \lambda_{30} - \lambda_{32})x_{230} + \lambda_{00} + \lambda_{02} + \lambda_{20} + \lambda_{22} + \lambda_{30} + \lambda_{32} \end{aligned}$$

sujeito a:

$$x_{100} + x_{102} + x_{120} + x_{122} + x_{200} + x_{220} + x_{230} \leq 3$$

$$x_{ipq} \in \{0,1\}$$

Observe que ao considerarmos os valores iniciais para os multiplicadores de Lagrange, por exemplo, iguais a 0, ou seja, $\lambda_{00} = \lambda_{02} = \lambda_{20} = \lambda_{22} = \lambda_{30} = \lambda_{32} = 0$, temos o seguinte problema:

$$\text{Max } x_{100} + x_{102} + x_{120} + x_{122} + x_{200} + x_{220} + x_{230}$$

sujeito a:

$$x_{100} + x_{102} + x_{120} + x_{122} + x_{200} + x_{220} + x_{230} \leq 3$$

$$x_{ipq} \in \{0,1\}$$

A solução do problema com a aplicação de relaxação Lagrangiana é de fácil resolução, pois como podemos ver no modelo acima, devemos considerar as variáveis que maximizem o valor da função objetivo respeitando o conjunto de restrições. Assim, devemos ordenar as variáveis x_{ipq} por V_{ipq} (estritamente positivos) e, em seguida, devemos escolher as primeiras $Q=3$ variáveis, no caso, por exemplo, $x_{100} = x_{102} = x_{120} = 1$ com valor da função objetivo igual a 3, apesar desta não ser uma solução factível para o problema original. Note que o problema acima satisfaz a propriedade de integralidade.

4.5.2.2. OTIMIZAÇÃO DO SUBGRADIENTE PARA RELAXAÇÃO LAGRANGIANA

Para resolver o problema dual Lagrangiano, utilizamos o método de otimização do subgradiente, que está descrito a seguir. Para maiores detalhes do método, o leitor pode consultar, por exemplo, CAMERINI *et al.* (1975), CROWDER (1976), BEASLEY (1985b) e REEVES (1993). A seguir apresentamos o método conforme apresentado em FARAGO e 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 $\lambda_{rs} = 0$ para todo $s \in Y$, e todo $r \in X$, como valores iniciais para os multiplicadores.

Passo 2: Resolva o problema Lagrangiano (4.6)-(4.8) 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 (3.1)-(3.4), 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:

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

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*.

Existem várias maneiras de se variar o parâmetro f ao longo das iterações, para acelerar a convergência do método, conforme FARAGO e MORABITO (2000), que serão discutidas no Capítulo 6. Note que, se em alguma iteração obtivermos

$Z_{min} = Z_{LB}$ no *Passo 4*, então o procedimento acima termina com a solução ótima do problema original (3.1)-(3.4).

Em LUCENA (2004), o termo *relax and cut* é usado para denotar uma classe de algoritmos de relaxação Lagrangiana onde uma desigualdade, entre muitas, pode somente ser dualizada se esta viola a solução do problema com relaxação Lagrangiana. O autor apresenta algumas modificações *relax and cut* no procedimento de otimização do subgradiente, sendo que uma delas é similar à proposta por Beasley (REEVES, 1993). Beasley (REEVES, 1993) reporta uma boa prática para a convergência do procedimento de otimização do subgradiente ajustando os subgradientes antes do cálculo do tamanho do passo (t) (durante o *Passo 5*), usando $G_{rs} = 0$ sempre que $G_{rs} < 0$ e $\lambda_{rs} = 0$. Ou seja, se a restrição (r,s) não está violada (isto é, $G_{rs} < 0$) na iteração corrente, e não possui multiplicador associado não nulo (isto é, $\lambda_{rs} > 0$), então seu gradiente é desconsiderado, fixando-o em $G_{rs} = 0$ (*Passo 5*). Com isso, ele não contribui no cálculo do tamanho do passo t (*Passo 6*).

Uma outra maneira que pode ser usada para resolver o problema dual Lagrangiano utilizando o método do subgradiente é incorporar as modificações propostas por CAMERINI *et al.* (1975). Neste trabalho, o tamanho do passo (t^m) na iteração m é atualizado utilizando as propostas a) e c) em CAMERINI *et al.* (1975). Na primeira iteração do método do subgradiente é utilizada a proposta a) em CAMERINI *et al.* (1975), onde o tamanho é dado por um valor fixo ($t^1 = 10^{-5}$). Nas demais iterações é utilizada a proposta c), onde o tamanho do passo (t^m) é dado pela expressão apresentada a seguir e os subgradientes utilizados são combinações de subgradientes de iterações anteriores. Assim,

$$s^m = G_{rs} + \beta_m s^{m-1},$$

onde,

$$s^{m-1} = 0 \text{ para } m = 0$$

$$\beta_m = \begin{cases} -\gamma \frac{s^{m-1} G_{rs}}{\|s^{m-1}\|^2}, & \text{se } s^{m-1} G_{rs} < 0 \\ 0, & \text{caso contrário} \end{cases}$$

e

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

4.5.3. RELAXAÇÃO *SURROGATE* (SUR)

A relaxação *Surrogate* (SUR) foi introduzida na Programação Matemática por GLOVER (1965, 1968). Esta relaxação consiste, basicamente, em considerar um conjunto de restrições do problema original e, usando um vetor de multiplicadores, transformar estas restrições em apenas uma restrição e esta nova restrição é chamada de *Surrogate*. Desde a sua introdução, esta relaxação tem sido proposta por vários autores para uso na solução de problemas não convexos, principalmente problemas de Programação Inteira. Um tratamento teórico abrangente da dualidade *Surrogate* em Programação Matemática é dado por GREENBERG e PIERSKALLA (1970). GLOVER (1975) resumiu estes resultados e apresentou um tratamento unificado da teoria da dualidade *Surrogate*. Um estudo da relação entre o dual Lagrangiano e o dual *Surrogate*, para o caso de problemas inteiros, foi feito por KARVAN e RARDIN (1979). Procedimentos de busca para atualização dos multiplicadores *Surrogate* foram desenvolvidos por BANERJEE (1971), GLOVER (1965, 1968, 1975), DYER (1980) e SARIN *et al.* (1987).

Voltemos ao problema (P). Consideremos um vetor de multiplicadores $\mu \geq 0$, $\mu \neq 0$. A relaxação *Surrogate* do problema (P) pode ser formulada como:

$$\begin{aligned} & \text{Max } cx \\ & \text{sujeito a:} \\ & \mu(Ax - b) \leq 0 \\ & Bx \leq d \\ & x \in \{0,1\} \end{aligned}$$

Desta forma as restrições $Ax \leq b$ são substituídas por $\mu(Ax - b) \leq 0$.

Note que o problema original

$$\text{Max } cx$$

sujeito a:

$$\begin{aligned} Ax &\leq b \\ Bx &\leq d \\ x &\in \{0,1\} \end{aligned}$$

tem valor menor ou igual a solução do problema ($\mu \geq 0, \mu \neq 0$)

$$\text{Max } cx$$

sujeito a:

$$\begin{aligned} Ax &\leq b \\ \mu(Ax - b) &\leq 0 \\ Bx &\leq d \\ x &\in \{0,1\} \end{aligned}$$

que por sua vez, tem valor menor ou igual a solução do problema

$$\text{Max } cx$$

sujeito a:

$$\begin{aligned} \mu(Ax - b) &\leq 0 \\ Bx &\leq d \\ x &\in \{0,1\} \end{aligned}$$

De um modo geral, podemos dizer que os problemas resultantes das relaxações *Surrogate* são mais difíceis de se resolver do que os de relaxação Lagrangiana. O dual *Surrogate* é definido como:

$$\text{Min}_{\mu \geq 0, \mu \neq 0} \left\{ \begin{array}{l} \text{Max } cx \\ \text{s.a : } \mu(Ax - b) \leq 0 \\ Bx \leq d \\ x \in \{0,1\} \end{array} \right\}.$$

De acordo com GREENBERG e PIERSKALLA (1970) e GLOVER (1975) a solução do dual *Surrogate* fornece limitantes melhores ou iguais ao dual Lagrangiano. LORENA e LOPES (1994) e LORENA e NARCISO (1996), usaram, respectivamente, a relaxação *Surrogate* em heurísticas para o problema de Cobertura (*Set Covering Problem*) e para o problema de Atribuição Generalizado (*Generalized Assignment Problem*) utilizando relaxação *Surrogate* contínua (quando as restrições de integralidade também são relaxadas). Em ambos os casos eles reportaram que a relaxação *Surrogate* contínua pode ser usada com vantagens sobre a relaxação Lagrangiana, principalmente em relação ao tempo computacional. Uma das heurísticas de GALVÃO *et al.* (2000) utiliza também a relaxação *Surrogate* para resolver o problema de localização de máxima cobertura.

4.5.3.1. APLICAÇÃO DE RELAXAÇÃO *SURROGATE* NO PCP DO PRODUTOR

Como dito anteriormente, a relaxação *Surrogate* consiste, basicamente, em considerar um conjunto de restrições do problema original e usando um vetor de multiplicadores transformar estas restrições em apenas uma restrição. Dessa forma, introduzindo os multiplicadores *Surrogate* $\mu_{rs} \geq 0$, $\mu_{rs} \neq 0$ para todo $r \in X$ e todo $s \in Y$, temos a seguinte relaxação *Surrogate*:

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

sujeito a:

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

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

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

que pode ser reescrito por:

$$\text{Max} \sum_{i=1}^2 \sum_{\{p \in X | p \leq L-l_i\}} \sum_{\{q \in Y | q \leq W-w_i\}} x_{ipq} \quad (4.9)$$

sujeito a:

$$\sum_{i=1}^2 \sum_{\{p \in X | p \leq L-l_i\}} \sum_{\{q \in Y | q \leq W-w_i\}} V_{ipq} x_{ipq} \leq \sum_{r \in X} \sum_{s \in Y} \mu_{rs} \quad (4.10)$$

$$P \leq \sum_{i=1}^2 \sum_{\{p \in X | p \leq L-l_i\}} \sum_{\{q \in Y | q \leq W-w_i\}} x_{ipq} \leq Q \quad (4.11)$$

$$x_{ipq} \in \{0, 1\}, i=1, 2, p \in X \text{ tal que } p \leq L-l_i; q \in Y \text{ tal que } q \leq W-w_i \quad (4.12)$$

onde $V_{ipq} = \sum_{r \in X} \sum_{s \in Y} \mu_{rs} a_{ipqrs}$ (note que V_{ipq} sempre é não negativo, dado que $\mu_{rs} \geq 0, \mu_{rs} \neq 0$).

Com a aplicação da relaxação *Surrogate*, temos que o modelo (4.9)-(4.12) resulta em um problema 0–1 com coeficientes reais, sendo que este problema é fácil de se resolver (dados $\{\mu_{rs}\}$), 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 o limite, escolha o maior número possível de variáveis com menores valores V_{ipq} não ultrapassando o limite, e fixe-as em 1.
- Se $P \leq n < Q$, 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 o 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 o 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 e Q como um dos limitantes inferiores e superiores definidos neste capítulo), mostrando que o problema *Surrogate* (4.9)-(4.12) não satisfaz a propriedade de integralidade. Também não encontramos, ao longo dos experimentos computacionais realizados no Capítulo 6, um 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 *Surrogate* (4.9)-(4.12) para o problema (3.1)-(3.4) não deve ser melhor do que o limitante superior obtido pela relaxação de Programação Linear do problema (3.1)-(3.4).

Para ilustrar a relaxação *Surrogate*, consideremos o exemplo ilustrativo apresentado no Capítulo 3.

Ilustração através do exemplo didático

Como dito anteriormente, é utilizado o exemplo numérico da seção 3.2 do Capítulo 3 para ilustrar a aplicação de relaxação *Surrogate*. Devemos considerar um conjunto de restrições do problema original e, usando um vetor de multiplicadores, transformar estas restrições em apenas uma restrição. A relaxação *Surrogate* para este exemplo é dada por:

$$\text{Max } x_{100} + x_{102} + x_{120} + x_{122} + x_{200} + x_{220} + x_{230}$$

sujeito a:

$$\mu_{00}(x_{100} + x_{200}) + \mu_{02}(x_{102} + x_{200}) + \mu_{20}(x_{100} + x_{120} + x_{220}) + \mu_{22}(x_{102} + x_{122} + x_{220}) + \mu_{30}(x_{120} + x_{220} + x_{230}) + \mu_{32}(x_{122} + x_{220} + x_{230}) \leq \mu_{00} + \mu_{02} + \mu_{20} + \mu_{22} + \mu_{30} + \mu_{32}$$

$$x_{100} + x_{102} + x_{120} + x_{122} + x_{200} + x_{220} + x_{230} \leq 3$$

$$x_{ipq} \in \{0, 1\}$$

onde $\mu_{00}, \mu_{02}, \mu_{20}, \mu_{22}, \mu_{30}, \mu_{32}$ são os multiplicadores *Surrogate*.

Reescrevendo o modelo acima temos:

$$\text{Max } x_{100} + x_{102} + x_{120} + x_{122} + x_{200} + x_{220} + x_{230}$$

sujeito a:

$$\begin{aligned}
& (\mu_{00} + \mu_{20})x_{100} + (\mu_{02} + \mu_{22})x_{102} + (\mu_{20} + \mu_{30})x_{120} + (\mu_{22} + \mu_{32})x_{122} + (\mu_{00} + \mu_{02})x_{200} + \\
& (\mu_{20} + \mu_{22} + \mu_{30} + \mu_{32})x_{220} + (\mu_{30} + \mu_{32})x_{230} \leq \mu_{00} + \mu_{02} + \mu_{20} + \mu_{22} + \mu_{30} + \mu_{32} \\
& x_{100} + x_{102} + x_{120} + x_{122} + x_{200} + x_{220} + x_{230} \leq 3 \\
& x_{ipq} \in \{0,1\}
\end{aligned}$$

Considerando, por exemplo, $\mu_{00} = \mu_{02} = \mu_{20} = \mu_{22} = \mu_{30} = \mu_{32} = 1$ temos o seguinte problema:

$$\text{Max } x_{100} + x_{102} + x_{120} + x_{122} + x_{200} + x_{220} + x_{230} \quad (4.13)$$

sujeito a:

$$2x_{100} + 2x_{102} + 2x_{120} + 2x_{122} + 2x_{200} + 4x_{220} + 2x_{230} \leq 6 \quad (4.14)$$

$$x_{100} + x_{102} + x_{120} + x_{122} + x_{200} + x_{220} + x_{230} \leq 3 \quad (4.15)$$

$$x_{ipq} \in \{0,1\} \quad (4.16)$$

A solução do problema com a aplicação de relaxação *Surrogate* é de fácil resolução, pois como podemos ver no modelo (4.13)-(4.16), devemos considerar as variáveis que maximizem o valor da função objetivo respeitando o conjunto de restrições. Assim, devemos ordenar as variáveis x_{ipq} por V_{ipq} (em ordem crescente) e, em seguida, devemos escolher as primeiras $Q = 3$ variáveis que não ultrapassem o valor de $\sum_{r \in X} \sum_{s \in Y} \mu_{rs} = 6$, no caso, por exemplo, $x_{100} = x_{102} = x_{120} = 1$ com valor da função objetivo igual a 3, apesar desta não ser uma solução factível para o problema original. O que devemos observar é que ao substituímos um item que faz parte da solução por qualquer outro item que não faça parte da solução (ou até mesmos outros itens), a solução resultante sempre será igual ou pior que a solução considerada. Note que o exemplo (4.13)-(4.16) satisfaz a propriedade de integralidade

A aplicação de relaxação *Surrogate* nos fornece limitantes superiores para o modelo original para qualquer $\mu_{rs} \geq 0$, $\mu_{rs} \neq 0$. O problema central então é encontrar $\mu_{rs} \geq 0$, $\mu_{rs} \neq 0$ que minimizem estes limitantes superiores, ou seja, a escolha de valores para os multiplicadores *Surrogate* é a chave importante na qualidade dos limitantes superiores gerados. O problema dual *Surrogate* é dado por:

$$\left. \begin{array}{l} \text{Max} \sum_{i=1}^2 \sum_{\{p \in X | p \leq L-l_i\}} \sum_{\{q \in Y | q \leq W-w_i\}} x_{ipq} \\ \text{s.a.:} \sum_{i=1}^2 \sum_{\{p \in X | p \leq L-l_i\}} \sum_{\{q \in Y | q \leq W-w_i\}} V_{ipq} x_{ipq} \leq \sum_{r \in X} \sum_{s \in Y} \mu_{rs} \\ P \leq \sum_{i=1}^2 \sum_{\{p \in X | p \leq L-l_i\}} \sum_{\{q \in Y | q \leq W-w_i\}} x_{ipq} \leq Q \\ x_{ipq} \in \{0,1\}, i=1,2, p \in X \text{ tal que } p \leq L-l_i; q \in Y \text{ tal que } q \leq W-w_i \end{array} \right\},$$

ou seja, $\min_{\mu \geq 0, \mu \neq 0}$ {problema (4.9)-(4.12)}.

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). SARIN *et al.* (1987) desenvolveram um procedimento para a busca dos multiplicadores do dual *Surrogate* utilizando uma sucessão de buscas no dual Lagrangiano. Tal procedimento não será explorado neste trabalho e é uma perspectiva interessante para pesquisa futura.

4.5.3.2. OTIMIZAÇÃO DO SUBGRADIENTE PARA RELAXAÇÃO *SURROGATE*

Conforme mencionado acima, para resolver o problema dual *Surrogate*:

$\min_{\mu \geq 0, \mu \neq 0}$ {problema (4.9)-(4.12)} de forma aproximada, pode ser utilizado o método de otimização do subgradiente, porém, não há garantia teórica de convergência (GALVÃO *et al.*, 2000). O procedimento é similar ao da seção 4.5.2.2, trocando-se o problema Lagrangiano (4.6)-(4.8) pelo problema *Surrogate* (4.9)-(4.12), com algumas adaptações nos *Passos 1* e *6*.

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:

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

4.5.4. RELAXAÇÃO LAGRANGIANA *SURROGATE* (LAGSUR)

A relaxação Lagrangiana *Surrogate* (LAGSUR) é uma combinação das relaxações Lagrangiana e *Surrogate* com o objetivo de se conseguir melhores tempos computacionais na aplicação de otimização do subgradiente. Para um dado problema, inicialmente é derivada uma relaxação *Surrogate* de um conjunto adequado de restrições. Uma relaxação Lagrangiana da restrição *Surrogate* é então obtida.

Voltemos novamente ao problema (P) 0–1 :

$$\text{Max } cx$$

sujeito a:

$$Ax \leq b$$

$$Bx \leq d$$

$$x \in \{0,1\}$$

Considere a relaxação *Surrogate* em relação ao conjunto de restrições $Ax \leq b$ com $\mu \geq 0$, $\mu \neq 0$. A relaxação *Surrogate* do modelo acima pode ser formulada como:

$$\text{Max } cx$$

sujeito a:

$$\mu(Ax - b) \leq 0$$

$$Bx \leq d$$

$$x \in \{0,1\}$$

Aplicando agora a relaxação Lagrangiana temos:

$$\text{Max } cx + \lambda(\mu(b - Ax)) = (c - \lambda\mu A)x + \lambda\mu b$$

sujeito a:

$$Bx \leq d$$

$$x \in \{0,1\}$$

onde $\lambda \geq 0$ agora é um escalar. Resta agora encontrar os multiplicadores de Lagrange que minimizem o limitante superior para o nosso problema de minimização original, ou seja:

$$\text{Min}_{\lambda \geq 0, \mu \geq 0, \mu \neq 0} \left\{ \begin{array}{l} \text{Max } (c - \lambda\mu A)x + \lambda\mu b \\ \text{s.a.: } Bx \leq d \\ x \in \{0,1\} \end{array} \right\},$$

o qual é o dual Lagrangiano (com relaxação *Surrogate*). Note que fazendo $\lambda' = \lambda\mu$, voltamos à relaxação Lagrangiana original, que é um limitante superior para o problema original. Observe que a diferença entre a relaxação Lagrangiana *Surrogate* para a relaxação Lagrangiana é o fato de que λ é um escalar.

O uso da relaxação Lagrangiana *Surrogate* foi explorado em LORENA e SENNE (1996) que resolveram um problema de localização não capacitado, e em NARCISO e LORENA (1999) que resolveram o problema de alocação generalizado. A relaxação Lagrangiana *Surrogate* proposta envolve ainda a aplicação de um algoritmo para o cálculo do multiplicador λ . Este algoritmo pode ser de busca unidimensional ou eventualmente aquele que calcula o melhor multiplicador λ de uma forma exata (NARCISO, 1998).

4.5.4.1. APLICAÇÃO DE RELAXAÇÃO LAGRANGIANA SURROGATE NO PCP DO PRODUTOR

Poderíamos usar a relaxação Lagrangiana *Surrogate* descrita acima com o objetivo de acelerar a convergência em relação a relaxação Lagrangiana. No entanto, BEASLEY (1985b) sugeriu utilizar as duas restrições *Surrogate* (4.17) e (4.18) apresentadas abaixo, obtidas pela soma de todos os comprimentos possíveis $r \in X$, e

todas as larguras possíveis $s \in Y$, da restrição (3.2), ou seja, usando $\mu = 1$, para todo $r \in X$ e $s \in Y$:

$$\sum_{r \in X} \left(\sum_{i=1}^2 \sum_{\{p \in X | p \leq L-l_i\}} \sum_{\{q \in Y | q \leq W-w_i\}} a_{ipqrs} x_{ipq} \right) \leq |X|, \quad \text{para todo } s \in Y \quad (4.17)$$

$$\sum_{s \in Y} \left(\sum_{i=1}^2 \sum_{\{p \in X | p \leq L-l_i\}} \sum_{\{q \in Y | q \leq W-w_i\}} a_{ipqrs} x_{ipq} \right) \leq |Y|, \quad \text{para todo } r \in X \quad (4.18)$$

Note que isto é diferente do que foi apresentado acima por causa da escolha de $\mu = 1$. Substituindo no problema (3.1)-(3.4) a restrição (3.2) pelas restrições (4.17) e (4.18), temos o seguinte modelo:

$$\text{Max} \quad \sum_{i=1}^2 \sum_{\{p \in X | p \leq L-l_i\}} \sum_{\{q \in Y | q \leq W-w_i\}} x_{ipq} \quad (4.19)$$

sujeito a:

$$\sum_{r \in X} \left(\sum_{i=1}^2 \sum_{\{p \in X | p \leq L-l_i\}} \sum_{\{q \in Y | q \leq W-w_i\}} a_{ipqrs} x_{ipq} \right) \leq |X|, \quad \text{para todo } s \in Y \quad (4.20)$$

$$\sum_{s \in Y} \left(\sum_{i=1}^2 \sum_{\{p \in X | p \leq L-l_i\}} \sum_{\{q \in Y | q \leq W-w_i\}} a_{ipqrs} x_{ipq} \right) \leq |Y|, \quad \text{para todo } r \in X \quad (4.21)$$

$$P \leq \sum_{i=1}^2 \sum_{\{p \in X | p \leq L-l_i\}} \sum_{\{q \in Y | q \leq W-w_i\}} x_{ipq} \leq Q \quad (4.22)$$

$$x_{ipq} \in \{0, 1\}, \quad i=1, 2, p \in X \text{ tal que } p \leq L-l_i; \quad q \in Y \text{ tal que } q \leq W-w_i \quad (4.23)$$

Introduzindo os multiplicadores de Lagrange $g_s (\geq 0)$ para todo $s \in Y$ em (4.20) e $h_r (\geq 0)$ para todo $r \in X$ em (4.21), 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 (4.6)-(4.8).

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

sujeito a:

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

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

Reescrevendo o modelo acima temos:

$$\text{Max} \quad \sum_{i=1}^2 \sum_{\{p \in X | p \leq L-l_i\}} \sum_{\{q \in Y | q \leq W-w_i\}} V_{ipq} x_{ipq} + \sum_{s \in Y} g_s |X| + \sum_{r \in X} h_r |Y| \quad (4.24)$$

sujeito a:

$$P \leq \sum_{i=1}^2 \sum_{\{p \in X | p \leq L-l_i\}} \sum_{\{q \in Y | q \leq W-w_i\}} x_{ipq} \leq Q \quad (4.25)$$

$$x_{ipq} \in \{0,1\}, i=1,2, p \in X \text{ tal que } p \leq L-l_i; q \in Y \text{ tal que } q \leq W-w_i \quad (4.26)$$

$$\text{onde } V_{ipq} = 1 - \left(\sum_{r \in X} \sum_{s \in Y} (g_s + h_r) a_{ipqrs} \right).$$

Sejam $\{X_{ipq}\}$ representando os valores das variáveis $\{x_{ipq}\}$ na solução do problema Lagrangiano (4.24)-(4.26), com valor dado por:

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

que corresponde a um limitante superior para o valor ótimo do problema original (3.1)-(3.4), para quaisquer $\{g_s\}$ e $\{h_r\}$ não negativos. Note, entretanto, que este limitante não é melhor do que o limitante produzido pelo problema (4.6)-(4.8), uma vez que deriva de um problema Lagrangiano construído a partir de restrições *Surrogate* (com $\mu = 1$). Por outro lado, é mais fácil de ser computado, dado que envolve um menor número de multiplicadores de Lagrange.

Assim como no problema (4.6)-(4.8), dados $\{g_s\}$ e $\{h_r\}$, o problema (4.24)-(4.26) também pode ser facilmente resolvido por inspeção. O mesmo procedimento anteriormente descrito para resolver o problema (4.6)-(4.8) também pode ser aqui aplicado. O problema dual Lagrangiano (com relaxação *Surrogate*) é dado por:

$$\text{Min}_{g_s \geq 0, h_r \geq 0} \left\{ \begin{array}{l} \text{Max} \sum_{i=1}^2 \sum_{\{p \in X | p \leq L-l_i\}} \sum_{\{q \in Y | q \leq W-w_i\}} V_{ipq} x_{ipq} + \sum_{s \in Y} g_s |X| + \sum_{r \in X} h_r |Y| \\ \text{s.a: } P \leq \sum_{i=1}^2 \sum_{\{p \in X | p \leq L-l_i\}} \sum_{\{q \in Y | q \leq W-w_i\}} x_{ipq} \leq Q \\ x_{ipq} \in \{0,1\}, i=1,2, p \in X \text{ tal que } p \leq L-l_i; q \in Y \text{ tal que } q \leq W-w_i \end{array} \right\},$$

ou seja, $\min_{g_s \geq 0, h_r \geq 0} \{\text{problema Lagrangiano (4.24)-(4.26)}\}$.

Ilustração através do exemplo didático

O problema original para o exemplo numérico da seção 3.2 do Capítulo 3 é dado por:

$$\text{Max } x_{100} + x_{102} + x_{120} + x_{122} + x_{200} + x_{220} + x_{230}$$

sujeito a:

$$x_{100} + x_{200} \leq 1$$

$$x_{102} + x_{200} \leq 1$$

$$x_{100} + x_{120} + x_{220} \leq 1$$

$$x_{102} + x_{122} + x_{220} \leq 1$$

$$x_{120} + x_{220} + x_{230} \leq 1$$

$$x_{122} + x_{220} + x_{230} \leq 1$$

$$x_{100} + x_{102} + x_{120} + x_{122} + x_{200} + x_{220} + x_{230} \leq 3$$

$$x_{ipq} \in \{0,1\}$$

Desse conjunto de restrições do problema original, derivam duas famílias de restrições, resultado da soma de todos os comprimentos possíveis $r \in X$ e larguras $s \in Y$ (relaxação *Surrogate*). Com isso, o número de restrições é dado por $|X| + |Y|$.

- para $s = 0$: (0,0); (2,0) e (3,0)

$$(x_{100} + x_{200}) + (x_{100} + x_{120} + x_{220}) + (x_{120} + x_{220} + x_{230}) \leq 3$$

- para $s = 2$: $(0, \underline{2})$; $(2, \underline{2})$ e $(3, \underline{2})$
 $(x_{102} + x_{200}) + (x_{102} + x_{122} + x_{220}) + (x_{122} + x_{220} + x_{230}) \leq 3$
- para $r = 0$: $(\underline{0}, 0)$ e $(\underline{0}, 2)$
 $(x_{100} + x_{200}) + (x_{102} + x_{200}) \leq 2$
- para $r = 2$: $(\underline{2}, 0)$ e $(\underline{2}, 2)$
 $(x_{100} + x_{120} + x_{220}) + (x_{102} + x_{122} + x_{220}) \leq 2$
- para $r = 3$: $(\underline{3}, 0)$ e $(\underline{3}, 2)$
 $(x_{120} + x_{220} + x_{230}) + (x_{122} + x_{220} + x_{230}) \leq 2$

Após aplicarmos relaxação *Surrogate* podemos escrever o problema da seguinte forma:

$$\text{Max } x_{100} + x_{102} + x_{120} + x_{122} + x_{200} + x_{220} + x_{230}$$

sujeito a:

$$2x_{100} + x_{200} + 2x_{120} + 2x_{220} + x_{230} \leq 3$$

$$2x_{102} + x_{200} + 2x_{122} + 2x_{220} + x_{230} \leq 3$$

$$x_{100} + 2x_{200} + x_{102} \leq 2$$

$$x_{100} + x_{120} + 2x_{220} + x_{102} + x_{122} \leq 2$$

$$x_{120} + x_{122} + 2x_{220} + 2x_{230} \leq 2$$

$$x_{100} + x_{102} + x_{120} + x_{122} + x_{200} + x_{220} + x_{230} \leq 3$$

$$x_{ipq} \in \{0, 1\}$$

Na aplicação de relaxação Lagrangiana, somamos a função objetivo todas as restrições de sobreposição e ficamos com um problema Lagrangiano da seguinte forma:

$$\begin{aligned} \text{Max } & x_{100} + x_{102} + x_{120} + x_{122} + x_{200} + x_{220} + x_{230} + g_0(3 - 2x_{100} - x_{200} - 2x_{120} - 2x_{220} - x_{230}) + \\ & g_2(3 - 2x_{102} - x_{200} - 2x_{122} - 2x_{220} - x_{230}) + h_0(2 - x_{100} - 2x_{200} - x_{102}) + \\ & h_2(2 - x_{100} - x_{120} - 2x_{220} - x_{102} - x_{122}) + h_3(2 - x_{120} - x_{122} - 2x_{220} - 2x_{230}) \end{aligned}$$

sujeito a:

$$x_{100} + x_{102} + x_{120} + x_{122} + x_{200} + x_{220} + x_{230} \leq 3$$

$$x_{ipq} \in \{0,1\}$$

Reescrevendo o problema acima temos:

$$\begin{aligned} \text{Max } & (1-2g_0-h_0-h_2)x_{100} + (1-2g_2-h_0-h_2)x_{102} + \\ & (1-2g_0-h_2-h_3)x_{120} + (1-2g_2-h_2-h_3)x_{122} + \\ & (1-g_0-g_2-2h_0)x_{200} + (1-2g_0-2g_2-2h_2-2h_3)x_{220} + \\ & (1-g_0-g_2-2h_3)x_{230} + 3g_0 + 3g_2 + 2h_0 + 2h_2 + 2h_3 \end{aligned}$$

sujeito a:

$$x_{100} + x_{102} + x_{120} + x_{122} + x_{200} + x_{220} + x_{230} \leq 3$$

$$x_{ipq} \in \{0,1\}$$

Considerando, por exemplo, $g_0 = g_2 = h_0 = h_2 = h_3 = 0$ temos o seguinte problema:

$$\text{Max } x_{100} + x_{102} + x_{120} + x_{122} + x_{200} + x_{220} + x_{230}$$

sujeito a:

$$x_{100} + x_{102} + x_{120} + x_{122} + x_{200} + x_{220} + x_{230} \leq 3$$

$$x_{ipq} \in \{0,1\}$$

A solução do problema com a aplicação de relaxação Lagrangiana e *Surrogate* é de fácil resolução, pois como podemos ver no modelo acima, devemos considerar as variáveis que maximizem o valor da função objetivo respeitando o conjunto de restrições. Assim, devemos ordenar as variáveis x_{ipq} por V_{ipq} (estritamente positivos) e, em seguida, devemos escolher as primeiras $Q=3$ variáveis, no caso, por exemplo, $x_{100} = x_{102} = x_{120} = 1$ com valor da função objetivo igual a 3, apesar desta não ser uma solução factível para o problema original.

4.5.4.2. OTIMIZAÇÃO DO SUBGRADIENTE PARA RELAXAÇÃO LAGRANGIANA *SURROGATE*

Para resolver o problema dual Lagrangiano (com relaxação *Surrogate*): $\min_{g_s \geq 0, h_r \geq 0}$ {problema Lagrangiano (4.24)-(4.26)} podemos utilizar o método de otimização do subgradiente, e este é muito similar ao da seção 4.5.2.2, trocando-se o problema Lagrangiano (4.6)-(4.8) pelo problema (4.24)-(4.26), com pequenas modificações nos *Passos 1, 5 e 6*, conforme FARAGO e 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:

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

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

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:

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

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

CAPÍTULO 5

BUSCA EM ÁRVORE

5.1. INTRODUÇÃO

Neste capítulo são apresentados os métodos de solução exatos utilizados para a resolução do PCP do produtor, baseados na aplicação das técnicas de relaxação Lagrangiana (LAG), *Surrogate* (SUR) e Lagrangiana *Surrogate* (LAGSUR) na formulação de BEASLEY (1985b), discutida nos Capítulos 3 e 4. Conforme visto no capítulo anterior, 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 e WOLSEY, 1988). O problema é resolvido utilizando uma árvore binária com busca em profundidade e, em cada nó da árvore, é computado um limitante superior de relaxação Lagrangiana e/ou *Surrogate*.

Ainda, é possível aplicar, durante cada iteração do método de otimização do subgradiente, alguma heurística Lagrangiana ou *Surrogate* para gerar soluções factíveis para o problema original (3.1)-(3.4), a partir da solução obtida para o problema Lagrangiano e/ou *Surrogate* (modelos (4.6)-(4.8) (LAG), (4.9)-(4.12) (SUR) e (4.24)-(4.26)(LAGSUR)). Também podemos aplicar alguma técnica de redução do problema

para tentar fixar, sem perda de generalidade, alguma variável em 0 ou 1, desta maneira, reduzir o tamanho do problema (3.1)-(3.4).

A seguir, apresentamos a técnica de redução do problema e a heurística Lagrangiana, a heurística *Surrogate* e, em seguida, a busca em árvore utilizada para resolver o PCP do produtor.

5.2. HEURÍSTICA LAGRANGIANA E REDUÇÃO DO PROBLEMA

Como dito anteriormente, a solução do problema Lagrangiano do limitante superior pode ser utilizada na obtenção de soluções factíveis para o problema original através de alguma heurística Lagrangiana. É possível também reduzir o tamanho do problema através da técnica de redução, tentando fixar, sem perda de generalidade, certas variáveis do problema em 0 ou 1. Ressalta-se que tanto a heurística Lagrangiana quanto a técnica de redução do problema, pode ser aplicada no procedimento de otimização do subgradiente considerando apenas a relaxação Lagrangiana ou ainda a relaxação Lagrangiana *Surrogate*.

5.2.1. HEURÍSTICA LAGRANGIANA

O desempenho dos testes de redução depende da qualidade do limitante inferior (solução factível). Dessa forma, utilizamos um procedimento heurístico (FARAGO e MORABITO, 2000) simples e eficaz para tentar encontrar uma solução factível para qualquer solução do problema Lagrangiano. A seguir, apresentamos os passos da heurística Lagrangiana:

Passo 1: Coloque todas as variáveis fixadas em 1 pela redução do problema (veja a seguir) 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 a da iteração anterior. Neste caso, atualizamos esse limitante inferior Z_{LB} . A otimalidade desta solução é testada no decorrer do método de otimização do subgradiente. A heurística Lagrangiana, como dito anteriormente, é aplicada em cada iteração do procedimento de otimização do subgradiente. No capítulo anterior (seção 4.5.2.2) tal procedimento foi descrito e a heurística Lagrangiana é aplicada durante a resolução do problema Lagrangiano (4.6)-(4.8) e do problema (4.24)-(4.26) (com relaxação *Surrogate*), ou seja, durante o *Passo 3* deste procedimento. A seguir, por conveniência, reescrevemos o *Passo 3* do procedimento de otimização do subgradiente.

Passo 3: Teste de factibilidade: se a solução Lagrangiana é factível para o problema original (3.1)-(3.4), 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} .

Convém salientar que em FARAGO e MORABITO (2000) foi proposto um procedimento heurístico com a heurística acima, combinada com a otimização do subgradiente. No entanto, não foi possível encontrar a solução ótima em mais da metade dos exemplos analisados com tal procedimento, e seu desempenho não foi melhor do que outras heurísticas da literatura, como por exemplo, o procedimento recursivo de MORABITO e MORALES (1998). Testes com o uso ou não da heurística Lagrangiana foram considerados no Capítulo 6.

Consideramos também um procedimento de melhoria na heurística Lagrangiana 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 do procedimento acima descrito (por exemplo, apenas para as primeiras p variáveis fixadas em 1).

5.2.2. REDUÇÃO DO PROBLEMA

A técnica de redução consiste em tentar fixar valores de certas variáveis do problema, possibilitando assim, uma redução no tamanho do problema a se executado. A seguir, explicamos a aplicação desta técnica no PCP do produtor. Consideremos n o número de variáveis escolhidas pelo problema Lagrangiano numa dada iteração, ou seja,

$$n = \sum_{r \in X} \sum_{s \in Y} X_{irs}, \text{ para todo } i$$

e, consideremos ainda, P e Q o número máximo e mínimo de caixas, respectivamente, a serem colocadas sobre o palete. Convém lembrar que, $\{X_{ipq}\}$ representa os valores das variáveis $\{x_{ipq}\}$ na solução do problema Lagrangiano (4.6)-(4.8) e (4.24)-(4.26) (com relaxação *Surrogate*).

- Fixando $x_{ipq}=0$

Calculamos o novo limitante superior, Z_{UB}' , da seguinte forma:

$$Z_{UB}' = Z_{UB} + V_{ipq} - \min_{r \in X, s \in Y, X_{irs}} \{V_{irs}\}, \text{ se } n = Q \quad (5.1)$$

$$Z_{UB}' = Z_{UB} + V_{ipq} - \min_{r \in X, s \in Y, X_{irs}} \{0, (V_{irs})\}, \text{ se } n < Q \quad (5.2)$$

onde (V_{irs}) é o conjunto dos valores dos coeficientes das variáveis tal que $X_{irs}=1$, ou seja, são os valores dos coeficientes das variáveis que estão fazendo parte da solução corrente do problema Lagrangiano e iguais a 1. Para calcular este novo limitante superior devemos inicialmente, calcular o número de variáveis n que fazem parte da solução do problema Lagrangiano e que são iguais a 1 e, em seguida, comparamos este número com a quantidade máxima de caixas (Q) que devem ser colocadas sobre o palete.

Se $n=Q$, então consideramos cada variável X_{ipq} que ficou fora da solução e a colocamos na solução, uma por vez. Assim, como o número de caixas nesta nova

solução ultrapassou Q , devemos retirar da solução a variável com mínimo valor V_{irs} tal que $X_{irs}=1$ do problema Lagrangiano corrente.

Se $n < Q$, temos dois sub-casos:

- Se $P < n < Q$, então ao colocarmos a variável X_{ipq} na solução, não precisamos retirar nenhuma variável de valor V_{irs} , tal que $X_{irs}=1$ pois não atingimos a quantidade máxima de caixas permitidas e, assim, qualquer variável que escolhermos terá valor do coeficiente do problema Lagrangiano positivo.
- Se $P = n < Q$, então ao colocarmos a variável X_{ipq} na solução, ou não retiramos alguma variável na solução, pelo mesmo motivo anterior, ou retiramos a variável com menor valor caso esta tenha valor negativo.

Observe que, a princípio, não poderíamos ter valores dos coeficientes negativos na solução, mas se o problema Lagrangiano encontra todos os valores negativos e tivermos $P=1$, então neste caso teríamos uma variável com valor do coeficiente negativo a ser retirada. Desta forma, ilustramos as equações (5.1) e (5.2). Se o valor deste novo limitante superior for menor ou igual ao valor do limitante inferior, ou seja, se $Z_{UB}' \leq Z_{LB}$ então podemos fixar x_{ipq} em 0 na solução ótima e então retirá-la do problema.

- **Fixando $x_{ipq}=1$**

Calculamos o novo limitante superior, Z_{UB}' , da seguinte forma:

$$Z_{UB}' = Z_{UB} - V_{ipq} + \max_{r \in X, s \in Y, X_{irs}} \{V_{irs}\}, \text{ se } n = P \quad (5.3)$$

$$Z_{UB}' = Z_{UB} - V_{ipq} + \max_{r \in X, s \in Y, X_{irs}} \{0, (V_{irs})\}, \text{ se } n < P \quad (5.4)$$

onde (V_{irs}) é o conjunto dos valores dos coeficientes das variáveis tal que $X_{irs}=0$, ou seja, são os valores dos coeficientes das variáveis que estão fazendo parte da solução corrente do problema Lagrangiano e iguais a 0. Para calcular este novo limitante superior devemos inicialmente, calcular o número de variáveis n definido como antes e, em seguida, comparamos este número com a quantidade mínima de caixas (P) que devem ser colocadas sobre o palete.

Se $n=P$, então consideramos cada variável X_{ipq} que estava na solução do problema Lagrangiano corrente e a retiramos da solução, uma por vez. Assim, como o

número de caixas nesta nova solução atingiu P , devemos colocar na solução a variável com maior valor V_{irs} tal que $X_{irs}=0$.

Se $n < P$, temos também dois sub-casos:

- Se $P < n < Q$, então ao retirarmos uma variável X_{ipq} na solução, não precisamos colocar nenhuma variável de valor V_{irs} , tal que $X_{irs}=0$, pois estamos atendendo a quantidade mínima de caixas exigidas e, assim, qualquer variável que escolhermos terá valor do coeficiente do problema Lagrangiano negativo.
- Se $P = n < Q$, então ao retirarmos uma variável X_{ipq} na solução, ou não colocamos alguma variável na solução, pelo mesmo motivo anterior, ou colocamos a variável com maior valor caso esta tenha valor positivo.

Desta forma, ilustramos as equações (5.3) e (5.4). Se o valor deste novo limitante superior for menor ou igual ao valor do limitante inferior, ou seja, se $Z_{UB}' \leq Z_{LB}$ então podemos fixar x_{ipq} em 1 na solução ótima e então retirá-la do problema.

A técnica de redução do problema, como dito acima, tem como objetivo reduzir o tamanho do problema, tentando fixar, sem perda de generalidade, certas variáveis do problema em 0 ou 1. Logo após a aplicação da heurística Lagrangiana durante o *Passo 3* do procedimento de otimização do subgradiente (visto no capítulo anterior), a técnica de redução é aplicada, tentando assim reduzir o tamanho do problema.

No trabalho de FARAGO e MORABITO (2000) são feitos testes quanto a inclusão ou não do procedimento de redução tanto no problema envolvendo apenas a relaxação Lagrangiana quanto no problema com relaxação Lagrangiana *Surrogate*. Os autores sugerem que é melhor utilizar o método de redução do problema nos programas LAG e LAGSUR, dado que o esforço computacional de tentar fixar variáveis ao longo das iterações parece ser compensatório. Isto foi testado no Capítulo 6, observando-se que é melhor utilizar a redução do problema.

5.2.3. PROCEDIMENTO DE OTIMIZAÇÃO DO SUBGRADIENTE PARA OS MÉTODOS LAG E LAGSUR

A seguir apresentamos o pseudo-código do procedimento de otimização do subgradiente com a heurística Lagrangiana e o procedimento de redução do problema, para os métodos LAG e LAGSUR onde:

- *it*: contador de iterações;
- Z_{min} : valor do melhor limitante para a solução do problema;
- Z_{LB} : valor do limitante inferior para a solução do problema correspondente a uma solução factível, por exemplo, a solução homogênea com todas as caixas arranjadas sob a mesma orientação;
- f : parâmetro usado na cálculo do tamanho do passo ($0 < f \leq 2$);
- λ_{rs} : multiplicadores de Lagrange;
- *solve_LUBP*(x): resolve o problema Lagrangiano do limitante superior;
- *objfun*(x): valor da função objetivo na solução corrente;
- *feasible* (x): procedimento para verificar a factibilidade da solução;
- $Gmin_{rs}$: conjunto dos subgradientes associados com a melhor solução;
- λmin_{rs} : conjunto de multiplicadores de Lagrange associados com a melhor solução;
- x_{LB} : vetor correspondente ao limitante inferior;
- x_{min} : vetor correspondente ao melhor limitante;
- *optimal*: variável booleana.

▪ **Pseudo-código do procedimento de otimização do subgradiente para o método LAG**

```

1. it:=1 (Passo 1: Inicialização)
2.  $Z_{LB} :=$  solução_homogênea (limitante inferior correspondente a uma solução factível)
3.  $Z_{min} :=$  infinito
4. Iniciar o parâmetro  $f$  ( $f=2$  ou  $f=1$ )
5.  $\lambda_{rs} := 0$  (valores iniciais dos multiplicadores)
6. Enquanto  $f \geq 0,005$  e (not optimal) faça
7.    $Z_{UB} :=$  solve_LUBP( $x$ ) (Passo 2: Resolva o problema Lagrangiano)
8.    $nc :=$  objfun ( $x$ )
9.   Se feasible ( $x$ ) e ( $nc > Z_{LB}$ ) então (Passo 3: Teste de factibilidade)
10.  |    $Z_{LB} := nc$ 
11.  |    $X_{LB} := x$ 
12.  fim
13.  Se  $Z_{UB} < Z_{min}$  então (atualização do menor limitante superior)
14.  |    $Z_{min} := Z_{UB}$ 
15.  |    $Gmin_{rs} = G_{rs}$ 
16.  |    $\lambda min_{rs} := \lambda_{rs}$ 
17.  |    $x_{min} := x$ 
18.  fim
19.   $Z_{LB} :=$  heurística_Lagrangiana (aplicação da heurística Lagrangiana)
20.  Redução (procedimento de redução)
21.  Se  $Z_{min} \leq Z_{LB}$  então (Passo 4: Teste de otimalidade)
22.  |   optimal := true;
23.  |    $x_{min} := x_{LB}$ 
24.  fim
25.  Senão
26.  |   Cálculo_subgradientes (Passo 5: Cálculo dos subgradientes)
27.  |   Cálculo_tamanho_passo (Passo 6: Cálculo do tamanho do passo  $t$ )
28.  |   Redução do valor do parâmetro  $f$ 
29.  |   Atualiza_multiplicadores (atualiza os multiplicadores de Lagrange)
30.  fim
31.  it:=it+1
32. fim

```

No pseudo-código do procedimento de otimização do subgradiente para o método LAG, inicialmente deve-se: gerar uma solução factível por exemplo, a solução homogênea com todas as caixas arranjadas sob a mesma orientação, com Z_{LB} igual ao valor desta solução homogênea; iniciar um contador de iterações com $it=0$; iniciar o parâmetro f ($f=2$ ou $f=1$) e fixar $\lambda_{rs}=0$ para todo $s \in Y$, e todo $r \in X$, como valores iniciais para os multiplicadores (*Passo 1*, linhas 1 a 5). Enquanto $f \geq 0,005$ e a solução não seja a ótima (*Passo 7*, linha 6) deve-se: (i) resolver o problema Lagrangiano com o conjunto de multiplicadores atual, obtendo a solução $\{X_{ipq}\}$ e valor Z_{UB} (*Passo 2*, linhas 7 e 8); (ii) executar o teste de factibilidade, ou seja, se a solução Lagrangiana é factível para o problema original (3.1)-(3.4), 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} ; executar a heurística Lagrangiana para tentar encontrar uma solução factível para o problema Lagrangiano e, em seguida, executar o procedimento de redução do problema, para tentar fixar variáveis em 0 e 1 e dessa forma reduzir o tamanho do problema (*Passo 3*, linhas 9 a 20); (iii) executar o teste de otimalidade, isto é, pare se $Z_{min} = Z_{LB}$ (*Passo 4*, linhas 21 a 23); caso contrário, calcular os subgradientes (*Passo 5*, linha 26), definir o tamanho do passo e atualizar os multiplicadores de Lagrange (*Passo 6*, linha 27), reduzir o parâmetro f (*Passo 6*, linha 28); (iv) atualizar o contador de iterações (*Passo 6*, linha 31). Ressalta-se que o conjunto inicial de multiplicadores de cada nó é o conjunto associado com o melhor limitante superior encontrado no nó predecessor da árvore (linhas 15 e 16). Experimentos computacionais mostraram que é muito mais eficaz utilizar tal inicialização, do que simplesmente iniciar os multiplicadores com algum valor arbitrário, por exemplo, o valor nulo. Foram exploradas algumas variações para reduzir o valor de f ao longo das iterações como, por exemplo, dividir f a cada 60 (ou 30 ou 15) iterações, sucessivamente dividir f a cada 60 iterações (ou 30 ou 15) se Z_{min} não diminuir nas últimas 60 (ou 30 ou 15) iterações, etc.. Tais variações serão vistas com mais detalhes no Capítulo 6.

O procedimento de otimização do subgradiente para o método LAGSUR é como o descrito para o método LAG com modificações nos *Passos 1, 5 e 6*, como apresentado na seção 4.5.4.2. A seguir apresenta-se o pseudo-código do procedimento de otimização do subgradiente para o método LAGSUR.

▪ **Pseudo-código do procedimento de otimização do subgradiente para o método LAGSUR**

```

1. it:=1 (Passo 1: Inicialização)
2.  $Z_{LB} :=$  solução_homogênea (limitante inferior correspondente a uma solução factível)
3.  $Z_{min} :=$  infinito
4. Iniciar o parâmetro  $f$  ( $f=2$  ou  $f=1$ )
5.  $g_s := 0; h_r = 0$  (valores iniciais dos multiplicadores)
6. Enquanto  $f \geq 0,005$  e (not optimal) faça
7.    $Z_{UB} :=$  solve_LUBP( $x$ ) (Passo 2: Resolva o problema Lagrangiano Surrogate)
8.    $nc :=$  objfun ( $x$ )
9.   Se feasible ( $x$ ) e ( $nc > Z_{LB}$ ) então (Passo 3: Teste de factibilidade)
10.  |    $Z_{LB} := nc$ 
11.  |    $X_{LB} := x$ 
12.  fim
13.  Se  $Z_{UB} < Z_{min}$  então (atualização do menor limitante superior)
14.  |    $Z_{min} := Z_{UB}$ 
15.  |    $Gmin_s = G_s; Hmin_r = H_r$ 
16.  |    $gmin_s := g_s; hmin_r := h_r$ 
17.  |    $x_{min} := x$ 
18.  fim
19.   $Z_{LB} :=$  heurística_Lagrangiana (aplicação da heurística Lagrangiana)
20.  Redução (procedimento de redução)
21.  Se  $Z_{min} \leq Z_{LB}$  então (Passo 4: Teste de otimalidade)
22.  |   optimal := true;
23.  |    $x_{min} := x_{LB}$ 
24.  fim
25.  Senão
26.  |   Cálculo_subgradientes (Passo 5: Cálculo dos subgradientes)
27.  |   Cálculo_tamanho_passo (Passo 6: Cálculo do tamanho do passo  $t$ )
28.  |   Redução do valor do parâmetro  $f$ 
29.  |   Atualiza_multiplicadores (atualiza os multiplicadores de Lagrange)
30.  fim
31.  it:=it+1
32. fim

```

5.3. HEURÍSTICA *SURROGATE* E REDUÇÃO DO PROBLEMA

A solução do problema *Surrogate* pode ser utilizada na obtenção de soluções factíveis para o problema original através de alguma heurística *Surrogate*. É possível também, reduzir o tamanho do problema através da técnica de redução do problema, tentando fixar, sem perda de generalidade, certas variáveis do problema em 0 ou 1, como visto na seção 5.2.2.

O desempenho dos testes de redução, como dito anteriormente, depende da qualidade do limitante inferior (solução factível). Assim, utilizamos um procedimento heurístico simples para tentar encontrar uma solução factível para qualquer solução do problema *Surrogate*. A seguir, apresentamos os passos da heurística *Surrogate*:

Passo 1: Coloque todas as variáveis fixadas em 1 pela redução do problema (veja a seguir) 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} ;

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 a da iteração anterior. Neste caso, atualizamos esse 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. Podemos utilizar o procedimento de redução descrito na seção 5.2.2 para o problema Lagrangiano também para o problema envolvendo a relaxação *Surrogate* com o objetivo de tentar fixar valores de certas variáveis do problema em 0 ou 1, possibilitando assim, uma redução no tamanho do problema a ser executado.

5.3.1. PROCEDIMENTO DE OTIMIZAÇÃO DO SUBGRADIENTE PARA O MÉTODO SUR

A seguir apresentamos o pseudo-código do procedimento de otimização do subgradiente com a heurística *Surrogate* e o procedimento de redução do problema, para o método SUR onde:

- *it*: contador de iterações;
- Z_{min} : valor do melhor limitante para a solução do problema;
- Z_{LB} : valor do limitante inferior para a solução do problema correspondente a uma solução factível, por exemplo, a solução homogênea com todas as caixas arranjadas sob a mesma orientação;
- f : parâmetro usado na cálculo do tamanho do passo ($0 < f \leq 2$);
- μ_{rs} : multiplicadores *Surrogate*;
- *solve_SUBP*(x): resolve o problema *Surrogate* do limitante superior;
- *objfun*(x): valor da função objetivo na solução corrente;
- *feasible* (x): procedimento para verificar a factibilidade da solução;
- $Gmin_{rs}$: conjunto dos subgradientes associados com a melhor solução;
- μmin_{rs} : conjunto de multiplicadores *Surrogate* associados com a melhor solução;
- x_{LB} : vetor correspondente ao limitante inferior;
- x_{min} : vetor correspondente ao melhor limitante;
- *optimal*: variável booleana.

O procedimento de otimização do subgradiente para o método SUR é como o descrito para o método LAG, com modificações nos *Passos 1* e *6* como apresentado na seção 4.5.3.2.

▪ **Pseudo-código do procedimento de otimização do subgradiente para o método SUR**

```

1. it:=1 (Passo 1: Inicialização)
2.  $Z_{LB} :=$  solução _homogênea (limitante inferior correspondente a uma solução factível)
3.  $Z_{min} :=$  infinito
4. Iniciar o parâmetro  $f$  ( $f=2$  ou  $f=1$ )
5.  $\mu_{rs} := 1$  (valores iniciais dos multiplicadores)
6. Enquanto  $f \geq 0,005$  e (not optimal) faça
7.    $Z_{UB} :=$  solve_SUBP( $x$ ) (Passo 2: Resolva o problema Surrogate)
8.    $nc :=$  objfun ( $x$ )
9.   Se feasible ( $x$ ) e ( $nc > Z_{LB}$ ) então (Passo 3: Teste de factibilidade)
10.  |    $Z_{LB} := nc$ 
11.  |    $X_{LB} := x$ 
12.  fim
13.  Se  $Z_{UB} < Z_{min}$  então (atualização do menor limitante superior)
14.  |    $Z_{min} := Z_{UB}$ 
15.  |    $Gmin_{rs} = G_{rs}$ 
16.  |    $\mu min_{rs} := \mu_{rs}$ 
17.  |    $x_{min} := x$ 
18.  fim
19.   $Z_{LB} :=$  heurística_Surrogate (aplicação da heurística Surrogate)
20.  Redução (procedimento de redução)
21.  Se  $Z_{min} \leq Z_{LB}$  então (Passo 4: Teste de otimalidade)
22.  |   optimal := true;
23.  |    $x_{min} := x_{LB}$ 
24.  fim
25.  Senão
26.  |   Cálculo_subgradientes (Passo 5: Cálculo dos subgradientes)
27.  |   Cálculo_tamanho_passo (Passo 6: Cálculo do tamanho do passo  $t$ )
28.  |   Redução do valor do parâmetro  $f$ 
29.  |   Atualiza_multiplicadores (atualiza os multiplicadores Surrogate)
30.  fim
31.  it:=it+1
32. fim

```

5.4. BUSCA EM ÁRVORE

Para se resolver problemas de Programação Inteira com um número finito de soluções factíveis, é comum o uso de algum tipo de procedimento de enumeração para encontrar uma solução ótima. Infelizmente para muitos problemas práticos, este número finito é muito grande, sendo impraticável a enumeração exaustiva de todas as soluções factíveis, mesmo com a capacidade dos computadores atuais. Por isso é necessário investir em um procedimento de enumeração que seja inteligentemente estruturado, para que apenas uma pequena fração de soluções factíveis realmente precise ser examinada. Essa é a idéia da técnica de enumeração implícita que pode ser representada através de um procedimento de busca em árvore.

Como dito anteriormente, 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 e WOLSEY, 1988). O problema é resolvido utilizando uma árvore binária com busca em profundidade e, em cada nó da árvore, é computado um limitante superior de relaxação Lagrangiana e/ou *Surrogate*. Convém ressaltar que autores têm mostrado que os limitantes de relaxações Linear e Lagrangiana são bem apertados para o PCP do produtor, e dominam outros limitantes (LETCHEFORD e AMARAL, 2001; MORABITO e FARAGO, 2002).

Podemos dizer que há um número de diferentes e difíceis decisões que devem ser tomadas ao se escolher um caminho para percorrer a busca em árvore para um problema particular. Como dito acima, será utilizado uma árvore binária com busca em profundidade, pois dessa forma é possível encontrar facilmente soluções factíveis e ainda, é de fácil implementação computacional. A seguir é apresentada a estrutura de cada nó da árvore e, em seguida, apresentamos o algoritmo do procedimento de busca em árvore LAG, SUR e LAGSUR.

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 sendo Z_{LB} o valor ótimo da solução (Z_{LB} é o limitante inferior na solução do problema correspondente a uma solução factível). O conjunto inicial de multiplicadores de cada nó é o conjunto associado com o melhor limitante superior encontrado no nó predecessor da árvore.

Convém lembrar que, durante cada iteração do método de otimização do subgradiente, é aplicada a heurística Lagrangiana ou *Surrogate* (seções 5.2.1 e 5.3) para gerar soluções factíveis para o problema original (3.1)-(3.4), a partir da solução obtida para o problema Lagrangiano e/ou *Surrogate* (modelos (4.6)-(4.8) (LAG), (4.9)-(4.12) (SUR) e (4.24)-(4.26) (LAGSUR)). Também é aplicada a técnica de redução do problema da seção 5.2.2 para tentar fixar, sem perda de generalidade, alguma variável em 0 ou 1, e, desta maneira, reduzir o tamanho do problema (3.1)-(3.4).

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, teremos um novo nó a esquerda e outro a direita. Neste caso, selecionamos uma variável e esta será fixada com valor igual a 1 no nó da esquerda e, conseqüentemente, 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, assim devemos explorar a possibilidade de se colocar tais variáveis na solução do problema e avaliar se esta solução é viável ou não.

De acordo com a experiência de Beasley (REEVES, 1993), no nó inicial da árvore vale a pena aplicar um esforço computacional ao realizar muitas iterações. Isto porque se dá a heurística Lagrangiana a chance de se encontrar boas soluções factíveis e, ao procedimento de redução, a chance de reduzir o tamanho do problema e, ainda, a otimização do subgradiente, encontrar bons limitantes. Problemas difíceis, por definição, exigem muitos nós na busca em árvore para a solução, e o tempo gasto no nó inicial torna-se uma pequena fração de todo o tempo computacional gasto para se resolver tais problemas.

Ainda segundo Beasley (REEVES, 1993), nos outros nós da árvore, uma opção é simplesmente repetir, em cada nó da árvore, exatamente o mesmo procedimento usado no nó inicial. A desvantagem disso é tornar o cálculo do limitante em cada nó muito caro. Assim, o que se deve fazer é balancear o esforço computacional envolvido no cálculo do limitante em cada nó contra a capacidade de armazenamento nos nós da

árvore resultante do melhoramento do limitante. Beasley (REEVES, 1993) sugere o uso de uma das seguintes estratégias:

1. 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;
2. 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;
3. duplicar ambos os parâmetros (para 60 sendo dividido a cada 10 (ou 20)) nos nós da árvore associados com a sondagem.

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. Como dito anteriormente, foram feitos vários testes quanto a redução do parâmetro f no procedimento de otimização do subgradiente, que serão apresentados no Capítulo 6.

A seguir é apresentado o algoritmo do procedimento de busca em árvore (NEMHAUSER e WOLSEY, 1988). Consideremos:

- L : lista de problemas inteiros (P^i) que devem ser investigados ou 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 (relaxação);
- Z_{UB} : limitante superior para o valor objetivo de P (solução factível);
- Z : valor da função objetivo do problema P .

Passo 1: Inicialização: forme uma lista $L=\{P\}$ (sendo P o problema (3.1)-(3.4)) 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^0 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:

- se $Z_R^i \leq Z_{LB}$, vá para o *Passo 2*;
- se $x_R^i \notin S^i$, vá para o *Passo 5*;
- se $x_R^i \in S^i$ e $Z > Z_{LB}$, faça $Z_{LB} = Z$. Elimine de L todos os problemas com $Z_{UB}^i \leq Z_{LB}$. Se $Z = Z_R^i$, vá para o *Passo 2*, caso contrário vá para o *Passo 5*.

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, conseqüentemente, 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*.

Na Figura 5.1 apresentamos um diagrama ilustrativo de uma árvore binária de busca fixando arbitrariamente as variáveis na seqüência x_1, x_2, x_3 e x_4 .

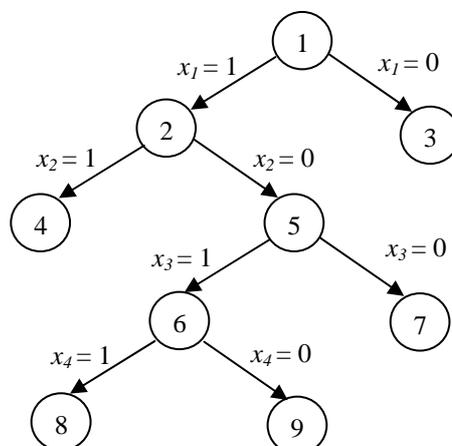


Figura 5.1. Diagrama ilustrativo de uma árvore binária de busca.

5.4.1. REGRA PARA SELEÇÃO DA VARIÁVEL A SER FIXADA

Como visto anteriormente, no *Passo 5* do algoritmo de procedimento de busca em árvore acima, devemos fazer a divisão dos nós (*branching*), ou seja, teremos dois novos subproblemas e devemos selecionar uma variável e esta será fixada com valor igual a 1 no nó da esquerda e, conseqüentemente, igual a 0 no nó da direita. A seguir, serão apresentadas algumas das estratégias para selecionar a variável a ser fixada na divisão dos nós.

- Proposta de BEASLEY (1985b)

Em qualquer nó da busca em árvore, 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,

$$R = \min(r \mid (i, r, s) \in G)$$

$$S = \min(s \mid (i, r, s) \in G)$$

$$V_{jRS} = \max(V_{iRS} \mid (i, R, S) \in G).$$

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 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 $x_{jRS} = 1$, ou seja, a variável que está mais abaixo e mais a esquerda, não podemos ter outra

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

- Proposta de BEASLEY (REEVES, 1993)

Voltemos ao modelo do PCP do produtor com aplicação de relaxação Lagrangiana apresentado no Capítulo 4. Por conveniência, rerepresentamos o modelo onde λ_{rs} (≥ 0) são os multiplicadores de Lagrange para todo $r \in X$ e todo $s \in Y$:

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

sujeito a:

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

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

onde $V_{ipq} = 1 - \sum_{r \in X} \sum_{s \in Y} \lambda_{rs} a_{ipqrs}$.

No procedimento da árvore de busca baseado no limitante de relaxação Lagrangiana, 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

$$\sum_{r \in X} \sum_{s \in Y} \lambda_{rs} \left(1 - \sum_{i=1}^2 \sum_{\{p \in X / p \leq L-l_i\}} \sum_{\{q \in X / q \leq W-w_i\}} a_{ipqrs} X_{ipq} \right) = \lambda G \text{ e escolha a restrição com}$$

maior valor absoluto neste vetor;

3. Selecione a seguir a variável com maior valor Lagrangiano, ou seja, com maior

$$V_{ipq}.$$

Ao término deste procedimento teremos selecionado a variável a ser fixada em 0 e em 1 na divisão dos nós.

- Proposta do maior valor de V_{ipq}

Uma estratégia é escolher a variável com maior valor Lagrangiano, ou seja, com maior V_{ipq} para ser fixada em 0 ou em 1.

- 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, ou seja, cada vez que uma determinada variável faz parte da solução do problema, a frequência é atualizada. A cada determinado número de iterações (por exemplo, 5 ou 10 iterações), escolhe-se uma variável que tenha baixa frequência (ou seja, uma variável que tenha feito parte da solução do problema poucas vezes) 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 estratégia de BEASLEY (1985b).

- Escolha aleatória

Podemos também escolher a variável a ser fixada na divisão dos nós de forma aleatória, sem considerar nenhum critério pré-definido.

5.4.2. PSEUDO-CÓDIGO DO MÉTODO DE BUSCA EM ÁRVORE

A seguir, é apresentado o pseudo-código da busca em árvore do método LAG, SUR e LAGSUR proposto, ou seja, da busca em árvore.

Considere:

- LL : lista que contém a rota do nó;
- $nc0$: limitante superior inicial para o número de caixas no palete. Utilizamos o limitante da Razão da Área, apresentado na seção 4.4.1 e o limitante de Barnes, apresentado na seção 4.4.2;
- Z_{LB} : valor do limitante inferior para a solução do problema;
- Z_{UBR} : valor do limitante superior obtido durante a execução do procedimento de otimização do subgradiente;

- *found*: variável booleana;
- x_{ipq} : vetor solução podendo ser 0, 1 ou -1 . No caso de ser -1 , isto quer dizer que a variável ainda não foi fixada (está livre);
- *nosolution*: nenhuma solução ainda foi encontrada para o problema;
- *feasible* (x): procedimento para verificar a factibilidade da solução;
- Z_{UBRO} : variável que recebe o valor da função objetivo para um determinado vetor solução;
- x_{LB} : vetor correspondente ao limitante inferior;
- seleção da variável: critério para escolha da variável x_{ipq} a ser ramificada;
- Atualiza_LL: atualiza a lista de nós após a ramificação;
- $LL.x_{ipq}$: após ser ramificada a variável $LL.x_{ipq}$ receberá valor zero no nó da direita e valor um no nó da esquerda;
- Deleta variável: $x_{jRS} = 1$ significa que devemos ter $x_{ipq} = 0$ se existe um ponto (r,s) tal que $a_{jRSrs} + a_{ipqrs} = 2$.

▪ **Pseudo-código para o método de Busca em Árvore binária**

```

1. nó=1; LL:={1}; (Passo 1: Inicialização)
2. nc0: limitante superior para o problema
3. zLB:= solução heurística
4. Enquanto (LL ≠ ∅) e (zLB < nc0) (Passo 2: Teste de término)
5.   zUBR :=otimização_subgradiente (Passo 3: Seleção do problema e relaxação)
6.   {checando por não solução} (Passo 4a: Infactibilidade ou valor de dominância)
7.   found:=false;
8.   se xipq=-1 então
9.     | found:= true;
10.  fim
11.  se found então
12.    | nosolution:=true
13.  fim
14.  senão
15.    | nosolution:=not feasible (x)
16.  fim
17.  se (not nosolution) e (ZUBR>ZLB) então
18.    {solução não dominada}
19.    se feasible (xR) então (Passo 4c: Otimalidade)
20.      | ZUBR0=objfun (xR)
21.    fim
22.    se (ZUBR>ZLB) então
23.      | ZLB= ZUBR0
24.      | xLB= xR
25.    fim
26.    senão (Passo 5: Divisão (branch))
27.      Seleção da variável xi,p,q
28.      {nó da direita}
29.      nó:=nó+1
30.      LL.xipq=0
31.      Atualiza_LL
32.      {nó da esquerda}
33.      no:=nó+1
34.      LL.xipq=1
35.      Deleta variável xi,p,q
36.      Atualiza_LL
37.    fim
38.  fim
39.  senão
40.    | não solução ou solução dominada
41.  fim
42. fim
43. {solução ótima}
44. ZLB; xLB; número de nós

```

No método de busca em árvore inicialmente deve-se formar uma lista $L=\{P\}$ (sendo P o problema (3.1)-(3.4)) com os problemas ainda não resolvidos, em seguida deve-se iniciar o limitante superior e limitante inferior (*Passo 1*, linhas 1 a 3). Para o cálculo do limitante superior foram utilizados os limitantes da Razão da Área e o de Barnes apresentados na seção 4.4. 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 e MORALES (1998). Esta é uma heurística simples e eficiente que pode ser vista como uma extensão da heurística de BISCHOFF e DOWSLAND (1982). A heurística de BISCHOFF e DOWSLAND (1982) é um método no qual o palete é inicialmente dividido em até 5 retângulos R_1, R_2, \dots, R_5 de dimensões $(L_1, W_1), (L_2, W_2), \dots, (L_5, W_5)$. Em cada retângulo as caixas são arranjadas em uma orientação horizontal pré-fixada, com exceção de R_3 onde a orientação é pós-fixada. A Figura 5.2 ilustra a localização dos retângulos no palete assim como a orientação das caixas.

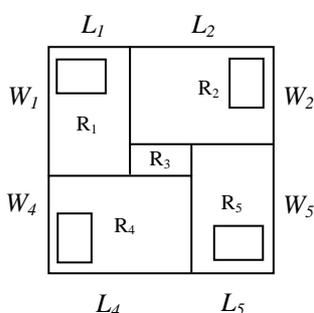


Figura 5.2. Divisão do palete em 5 blocos com orientação das caixas.

Basicamente, o método examina um número de cortes guilhotinados e não-guilhotinados de 1ª ordem, ou equivalentemente, um número de diferentes tamanhos (L_i, W_i) dos retângulos R_1, R_2, R_4 e R_5 (incluindo $L_i = 0$ e $W_i = 0$), e retorna o padrão que resulta em um número máximo de caixas. A busca pelo melhor padrão utiliza um procedimento recursivo e termina caso seja encontrado um padrão com área não utilizada menor do que a área de uma única caixa. Este resultado define uma condição suficiente, (mas não necessária), para verificar se uma solução é ótima para o PCP do produtor.

O teste de parada do método de busca em árvore é apresentado na linha 4 (*Passo 2*). Caso a solução não seja a ótima, deve-se selecionar e eliminar um problema da lista *LL*. Resolva a relaxação do problema selecionado utilizando o procedimento de otimização do subgradiente obtendo o valor da relaxação bem como a solução (*Passo 3*, linha 5). A seguir, deve-se realizar a sondagem dos nós pelos critérios de otimalidade, infactibilidade e valor de dominância (*Passo 4*, linhas 6 a 25). Caso a solução não seja a ótima deve-se realizar a divisão dos nós, assim teremos dois novos subproblemas e devemos selecionar uma variável e esta será fixada com valor igual a 1 no nó da esquerda e, conseqüentemente, igual a 0 no nó da direita. Esses novos subproblemas são incluídos na lista *L* (*Passo 5*, linhas 27 a 37). As estratégias consideradas para a escolha da variável a ser fixada na divisão dos nós foram apresentadas na seção 5.4.1. Convém lembrar que, 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 $x_{jRS} = 1$, ou seja, a variável que está mais abaixo e mais a esquerda, não podemos ter outra variável no ponto (r,s) , pois haverá sobreposição das peças e assim devemos retirar tais variáveis do problema, isto é, considerar tais variáveis iguais a zero ($x_{ipq} = 0$) (linha 35). Após a divisão dos nós, o método *branch and bound* retorna ao *Passo 2* (linha 4).

CAPÍTULO 6

RESULTADOS COMPUTACIONAIS

6.1. INTRODUÇÃO

Neste capítulo são descritos os resultados dos testes computacionais realizados com o objetivo de avaliar os métodos exatos LAG, SUR e LAGSUR, apresentados no capítulo anterior. As implementações foram realizadas em linguagem Pascal, compilador Delphi 5, e os testes foram realizados em um Pentium III 650 MHz.

Inicialmente foram realizados diversos testes computacionais preliminares para se verificar o desempenho de possíveis variações nos procedimentos de otimização do subgradiente e da busca em árvore. Algumas questões a serem respondidas na otimização do subgradiente são:

- i) utilizar ou não o método de redução do problema (seção 5.2.2);
- ii) utilizar ou não a heurística Lagrangiana/*Surrogate* (seções 5.2.1. e 5.3);
- iii) utilizar ou não a melhoria da heurística Lagrangiana (seção 5.2.2);
- iv) utilizar ou não modificações na atualização dos multiplicadores baseados em *relax and cut* (seção 4.5.2.2);
- v) utilizar ou não modificações propostas por CAMERINI *et al.* (1975) para cálculo do tamanho do passo t (seção 4.5.2.2);

vi) qual variação escolher para reduzir o parâmetro f do tamanho do passo t (seção 4.5.2.2).

Algumas questões a serem respondidas no procedimento de busca em árvore são:

- i) qual regra escolher para a seleção da variável a ser fixada na divisão dos nós (seção 5.4.1);
- ii) executar ou não um número maior de iterações no nó inicial da árvore (seção 5.4);
- iii) utilizar ou não valores de multiplicadores associados com o melhor limitante superior obtido no nó predecessor da árvore (seção 5.4);
- iv) utilizar ou não os conjuntos de *raster points* ao invés dos conjuntos normais (seção 4.4.5);
- v) utilizar ou não a heurística construtiva de MORABITO e MORALES (1998) para gerar bons limitantes inferiores iniciais (seção 5.4.2).

Estes testes preliminares envolveram diversos exemplos (incluindo alguns dos Grupos 1, 2, 3 e 4 definidos adiante), e as conclusões com tais resultados são discutidas a seguir. Como dito anteriormente, em cada iteração do procedimento de otimização do subgradiente é aplicada a heurística Lagrangiana (seção 5.2.1) ou *Surrogate* (5.3) para gerar soluções factíveis para o problema original (3.1)-(3.4), a partir do problema Lagrangiano e/ou *Surrogate* (modelos (4.6)-(4.8) (LAG), (4.9)-(4.12) (SUR) ou (4.24)-(4.26) (LAGSUR)). Também é aplicada a técnica de redução do problema da seção 5.2.2 para tentar fixar, sem perda de generalidade, alguma variável em 0 ou em 1, e, desta maneira, reduzir o tamanho do problema (3.1)-(3.4). Foram realizados testes preliminares com diversos exemplos quanto à aplicação ou não da heurística Lagrangiana; quanto à aplicação ou não do método de redução do problema e ainda, foi testada a exclusão do procedimento de redução juntamente com a heurística Lagrangiana. Podemos dizer diante dos resultados obtidos que o procedimento de redução do problema no método LAG é eficaz, dado que o esforço computacional de tentar fixar variáveis ao longo das iterações é compensatório. O mesmo acontece para o programa sem a inclusão da heurística Lagrangiana e do procedimento de redução. Já para o programa apenas sem a inclusão da heurística Lagrangiana, o tempo computacional foi um pouco menor (em média) do que com a inclusão da heurística Lagrangiana, o que sugere que a heurística não é efetiva nestes exemplos. Assim, todos

os testes apresentados adiante estão considerando a redução do problema e a heurística Lagrangiana. Ainda, foi testado o procedimento de melhoria na heurística Lagrangiana apresentado na seção 5.2.1. No entanto, tal procedimento não foi capaz de reduzir os tempos computacionais até então obtidos apenas com a utilização da heurística Lagrangiana. Dessa forma, os testes apresentados adiante não estão considerando tal modificação na heurística Lagrangiana.

Consideramos também a proposta do trabalho de LUCENA (2004), no qual o autor apresenta algumas modificações *relax and cut* no procedimento de otimização do subgradiente. Aplicamos uma delas que se assemelha à proposta por Beasley (REEVES, 1993) que reporta uma boa prática para a convergência do procedimento de otimização do subgradiente ajustando os subgradientes antes do cálculo do tamanho do passo (t) usando $G_{rs} = 0$ sempre que $G_{rs} < 0$ e $\lambda_{rs} = 0$ (problema de maximização). Tais testes computacionais foram realizados com diversos exemplos e os resultados não foram melhores do que os obtidos sem essas modificações. Uma outra maneira que pode ser utilizada para se resolver o problema dual Lagrangiano utilizando o procedimento de otimização do subgradiente é incorporar as modificações propostas por CAMERINI *et al.* (1975). Foram realizados testes computacionais com esta proposta e os resultados obtidos são mostrados adiante.

Ainda, o procedimento de otimização do subgradiente foi iniciado com os parâmetros $f = 2$ ou $f = 1$ (*Passo 6*, seção 4.5.2.2), sendo reduzido após um determinado número de iterações. Se a solução ótima é encontrada ou se $f < 0,005$ (*Passo 7*), o procedimento é terminado. Seis variações foram exploradas para reduzir o valor de f ao longo das iterações:

- a) dividir f a cada 60 iterações;
- b) dividir f a cada 30 iterações;
- c) dividir f a cada 15 iterações;
- d) sucessivamente dividir f a cada 60 iterações se Z_{min} não diminuir nas últimas 60 iterações;
- e) sucessivamente dividir f a cada 30 iterações se Z_{min} não diminuir nas últimas 30 iterações;
- f) sucessivamente dividir f a cada 15 iterações se Z_{min} não diminuir nas últimas 15 iterações.

Também consideramos dividir f a cada $|X||Y|$ 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 e MORABITO, 2000), no entanto os resultados obtidos não foram melhores do que os obtidos com as variações (a)-(f) acima.

Foram feitos vários testes preliminares também com as diferentes regras para a seleção da variável a ser fixada na divisão dos nós (*Passo 5* do algoritmo de procedimento de busca em árvore apresentado no Capítulo 5, seção 5.4). Dentre as cinco regras propostas (veja seção 5.4.1), a que apresentou melhores resultados quanto ao tempo computacional foi a proposta por BEASLEY (1985b). Assim, todos os testes apresentados adiante estão considerando tal regra. Consideramos também executar um número de iterações maior no nó inicial da árvore, conforme discutido na seção 5.4., Capítulo 5. No entanto, tal consideração não apresentou ganhos em relação ao tempo computacional e, desta forma, não foi incluída nos testes apresentados adiante.

Ressalta-se que 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, o que garante uma convergência mais rápida do método. No caso de não considerarmos os multiplicadores de Lagrange e/ou *Surrogate* associados com o melhor limitante superior encontrado no nó predecessor da árvore (por exemplo, inicializados com valor 0 ou 1), os tempos computacionais dos métodos são muito mais altos. Ainda, realizamos testes computacionais com o conjunto dos *raster points* (seção 4.4.5) e a heurística construtiva de MORABITO e MORALES (1998) (seção 5.4.2). Os testes apresentados adiante estão considerando tais variações no procedimento de busca em árvore.

Como já apresentado no Capítulo 5, utilizamos alguns limitantes para os métodos LAG, SUR e LAGSUR. O uso de um limitante superior apertado para o número máximo de caixas que podem ser empacotadas em um exemplo do PCP é útil em heurísticas para verificar a otimalidade, e em algoritmos exatos, pois permitem a redução do espaço de busca da solução. Assim, nos métodos propostos consideramos os limitantes da Razão da Área e o de Barnes, discutidos nas seções 4.4.1 e 4.4.2 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 e MORALES

(1998) (seção 5.4). Nos testes computacionais a seguir analisamos em detalhes os itens v) e vi) com relação ao procedimento de otimização do subgradiente, e os itens iv) e v) com relação ao procedimento de busca em árvore. Para isso, compilamos conjuntos de exemplos da literatura e reais, definidos como Grupos 1, 2, 3 e 4, conforme a seguir.

6.2. EXEMPLOS DA LITERATURA E REAIS

Nesta seção apresentamos os exemplos da literatura e os exemplos reais que foram utilizados para a realização dos testes computacionais. Inicialmente foram considerados 3 grupos de exemplos da literatura, cada um sendo subdividido em dois grupos menores. No primeiro grupo de exemplos (subdividido em Grupo 1A e 1B), selecionamos 20 exemplos, como mostrado nas Tabelas 6.1 e 6.2. Tais exemplos (L1-L20) são pertencentes ao conjunto Cover I e foram escolhidos aleatoriamente deste conjunto. Tanto o Grupo 1A como o Grupo 1B possuem 10 exemplos. O Grupo 1A contém exemplos que não apresentam *gap* entre o valor da solução ótima e o valor da solução da relaxação Linear de (3.1)-(3.4) (Z_{min}). Já o Grupo 1B contém exemplos que são considerados desfavoráveis para os métodos propostos, uma vez que todos têm valor da solução da relaxação Linear diferente do valor da solução ótima.

No Grupo 2A consideramos 9 exemplos (L21-L29) que foram retirados de FARAGO e MORABITO (2000) (Tabela 6.3). Estes exemplos têm sido utilizados em diversos artigos como DOWSLAND (1993), NELISSEN (1994), SCHEITHAUER e TERNO (1996) e MORABITO e MORALES (1998). Tais exemplos pertencem ao conjunto Cover I, os quais são resolvidos por alguns métodos heurísticos e, assim como o Grupo 1A, não apresentam *gap* entre o valor da solução da relaxação Linear de (3.1)-(3.4) e o valor da solução ótima. Os exemplos do Grupo 2B (L30-L39), como mostrado na Tabela 6.4, foram encontrados em LETCHFORD e AMARAL (2001) e também foram escolhidos por serem considerados desfavoráveis para os métodos propostos. Todos os exemplos dos Grupos 1A, 1B, 2A e 2B têm solução ótima não-guilhotinada de 1ª ordem (MORABITO e MORALES, 1998). Fizemos também experimentos limitados com todos os 3179 exemplos do conjunto Cover I analisado em MORABITO e MORALES (1998), que cobre os exemplos com solução ótima composta de até 50

caixas. Estes exemplos têm grande interesse nas aplicações práticas do PCP do produtor.

Consideramos ainda um terceiro grupo (Grupo 3A e 3B) com 32 exemplos da literatura (Tabelas 6.5 e 6.6) que são pertencentes ao conjunto Cover II. Os exemplos do Grupo 3A foram escolhidos aleatoriamente deste conjunto e todos têm solução ótima não-guilhotinada de 1ª ordem. Os do Grupo 3B são exemplos difíceis de se resolver com as heurísticas da literatura, pois todos eles não têm solução ótima não-guilhotinada de 1ª ordem (MORABITO e MORALES, 1998; LINS *et al.*, 2003). Incluímos estes exemplos, pois são considerados difíceis para os métodos heurísticos da literatura. Fizemos também experimentos limitados com todos 16938 exemplos do conjunto Cover II, analisado em MORABITO e MORALES (1998), que cobre exemplos com solução ótima composta de 51 a 100 caixas.

Os exemplos reais (Grupo 4) foram coletados no centro de distribuição da Transportadora Americana, em Campinas, São Paulo. Selecionamos arbitrariamente 30 exemplos como mostrado na Tabela 6.7. Estes exemplos utilizam o palete Brasil (PBR: 120x100 cm), também utilizado na maioria das empresas no Brasil.

Nas Tabelas 6.1 a 6.7 a coluna ' $2|X||Y|$ ' representa o número (aproximado) de variáveis 0–1 envolvidas no modelo (4.6)-(4.8) e a coluna ' $2|X'|Y|$ ' considera os *raster points* (seção 4.4.5) apresentando da mesma forma, o número (aproximado) de variáveis envolvidas no respectivo modelo. Observe nas Tabelas 6.1 a 6.7 que o número de variáveis envolvidas no modelo considerando os *raster points* (coluna ' $2|X'|Y|$ '), em geral, é bem menor do que sem os *raster points* (coluna ' $2|X||Y|$ '). Note nos exemplos reais que o número médio de variáveis 0–1 é bem menor (340 variáveis em média), se comparado com os exemplos dos Grupos 1A, 1B, 2A, 2B, 3A e 3B (milhares de variáveis). Note ainda que, o número médio de variáveis 0–1 nos Grupos 3A e 3B é bem maior, se comparado aos demais.

Tabela 6.1. Grupo 1A de exemplos para teste.

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 6.2. Grupo 1B de exemplos para teste.

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 6.3. Grupo 2A de exemplos para teste.

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 6.4. Grupo 2B de exemplos para teste.

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

Tabela 6.5. Grupo 3A de exemplos para teste.

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 6.6. Grupo 3B de exemplos para teste.

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 6.7. Exemplos reais coletados na Transportadora Americana 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

6.3. RESULTADOS UTILIZANDO O GAMS/CPLEX

A Tabela 6.8 apresenta os resultados (valores médios) encontrados pelo pacote GAMS/CPLEX (versão 7) para os exemplos da literatura (Grupos 1A, 1B, 2A, 2B, 3A e 3B) e exemplos reais (Grupo 4). São apresentados a solução ótima, a solução obtida em um tempo máximo de 3 horas de execução, o número de nós envolvidos e o tempo (em minutos) gasto para se resolver cada grupo de exemplos (computado apenas para os exemplos resolvidos em menos de 3 horas).

Tabela 6.8. Soluções obtidas utilizando o pacote GAMS/CPLEX (versão 7).

Grupo	Solução Ótima	Solução Obtida	Número de nós	Tempo (min)
1A	31,90	31,90	0	0,19
1B	44,20	44,20	0	0,81
2A	48,56	48,56	0	4,05
2B	42,60	42,60	25,7	4,87
3A	73,31	73,31	0	9,23
3B	80,06	79,94	1,14	17,55
4	20,10	20,10	0,50	18,52

Pela Tabela 6.8 podemos observar que, exceto no Grupo 3B, o pacote GAMS/CPLEX foi capaz de resolver todos os exemplos em um tempo computacional máximo de 3 horas. No Grupo 3B não foi possível encontrar a solução ótima em 2 exemplos, L69 e L70. Para estes exemplos as melhores soluções inteiras possíveis são 97 e 99, respectivamente e o GAMS/CPLEX encontrou 96 e 98.

6.4. RESULTADOS UTILIZANDO OS GRUPOS 1 E 2

Nesta seção, inicialmente é apresentada a análise dos resultados dos métodos LAG, SUR e LAGSUR utilizando o Grupo 1A, pois este grupo pertence ao conjunto Cover I e apresenta exemplos mais fáceis de se resolver. O procedimento de otimização do subgradiente foi iniciado com o parâmetro $f = 2$ (Passo 6, seção 4.5.2.2), sendo reduzido a cada 60 iterações. Consideramos nestes testes computacionais

a regra para a seleção da variável a ser fixada na divisão dos nós proposta por BEASLEY (1985b). A Tabela 6.9 apresenta a solução ótima, a solução obtida em um tempo computacional máximo de 3 horas, o número de nós envolvidos na busca em árvore e, em seguida, o tempo total (em minutos) gasto para se resolver o conjunto de exemplos (computado apenas para os exemplos resolvidos em menos de 3 horas) para os métodos LAG, SUR e LAGSUR.

Tabela 6.9. Resultados computacionais obtidos utilizando $f=2$ e divisão de f a cada 60 iterações com o Grupo 1A.

Método exato	Solução Ótima	Solução Obtida	Número de nós	Tempo (min)
LAG	31,90	31,90	4,00	0,24
SUR	31,90	31,50	3,67	0,06
LAGSUR	31,90	31,70	9,25	1,30

Podemos observar pela Tabela 6.9 que entre as três abordagens LAG, SUR e LAGSUR a que apresentou melhor resultado quanto ao valor da solução ótima e ao tempo computacional foi a que utiliza apenas relaxação Lagrangiana (lembre-se que os tempos são a média apenas dos exemplos resolvidos em menos de 3 horas). Além disso, o método LAG conseguiu resolver todos os exemplos do Grupo 1A. Outros experimentos foram realizados com diversos exemplos dos Grupos 1, 2, 3 e 4, variando-se as divisões do parâmetro f da otimização do subgradiente, e o método LAG continuou apresentando desempenho bem superior aos métodos SUR e LAGSUR. Uma vantagem do método LAG sobre o método LAGSUR é o número de multiplicadores bem maior ($|X||Y|$ ao invés de $|X|+|Y|$), o que contém bem 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 relaxação *Surrogate* não tem garantia de convergência, conforme discussão na seção 4.5.3.2. Assim, devido aos resultados obtidos, considerando que os métodos SUR e LAGSUR tiveram dificuldades para resolver mesmo exemplos mais fáceis como os do Grupo 1A, resolvemos abandoná-los, e os testes adiante são realizados considerando apenas o método LAG.

A seguir, é apresentada a análise dos resultados do método LAG utilizando os Grupos 1A, 1B, 2A e 2B, pois são os grupos de exemplos selecionados que pertencem ao conjunto Cover I. Estes grupos, por serem mais fáceis de se resolver, foram utilizados para se comparar outros parâmetros do método LAG. Como dito anteriormente, o procedimento de otimização do subgradiente foi iniciado com os parâmetros $f = 2$ ou $f = 1$ (*Passo 6*, seção 4.5.2.2), sendo reduzido após um determinado número de iterações. Seis variações foram exploradas para reduzir o valor de f ao longo das iterações, cujos resultados computacionais são mostrados nas tabelas a seguir. Também foram realizados experimentos com outras variações além dessas seis, mas os resultados obtidos não foram melhores.

Em todos estes testes computacionais consideramos a regra para a seleção da variável a ser fixada na divisão dos nós proposta por BEASLEY (1985b).

As Tabelas 6.10 a 6.15 apresentam os resultados computacionais (valores médios obtidos) para os grupos de exemplos 1A, 1B, 2A e 2B utilizando inicialmente o parâmetro $f = 2$ com as diferentes reduções de f . Em todas as tabelas a seguir, são apresentadas a solução ótima, a solução obtida em um tempo computacional máximo de 3 horas, o número de nós envolvidos na busca em árvore e, em seguida, o tempo total (em minutos) gasto para se resolver o conjunto de exemplos.

Tabela 6.10. Resultados computacionais obtidos com o método LAG utilizando $f=2$ e divisão de f a cada 60 iterações.

Grupo	Solução Ótima	Solução Obtida	Número de nós	Tempo (min)
1A	31,90	31,90	4,00	0,24
1B	44,20	44,20	20,40	2,10
2A	48,56	48,44	45,75	4,50
2B	42,60	42,60	196,20	17,73

Tabela 6.11. Resultados computacionais obtidos com o método LAG utilizando $f=2$ e divisão de f a cada 30 iterações.

Grupo	Solução Ótima	Solução Obtida	Número de nós	Tempo (min)
1A	31,90	31,90	9,40	0,38
1B	44,20	44,20	31,00	1,89
2A	48,56	48,44	63,75	3,22
2B	42,60	42,60	212,60	10,73

Tabela 6.12. Resultados computacionais obtidos com o método LAG utilizando $f=2$ e divisão de f a cada 15 iterações.

Grupo	Solução Ótima	Solução Obtida	Número de nós	Tempo (min)
1A	31,90	31,90	14,20	0,29
1B	44,20	44,20	65,00	2,02
2A	48,56	48,44	99,00	2,62
2B	42,60	42,60	311,20	8,10

Tabela 6.13. Resultados computacionais obtidos com o método LAG utilizando $f=2$ e divisão de f a cada 60 iterações se Z_{min} não diminuir nas últimas 60 iterações.

Grupo	Solução Ótima	Solução Obtida	Número de nós	Tempo (min)
1A	31,90	31,90	5,40	0,38
1B	44,20	44,20	12,60	2,88
2A	48,56	48,44	34,50	6,38
2B	42,60	42,60	65,00	14,28

Tabela 6.14. Resultados computacionais obtidos com o método LAG utilizando $f=2$ e divisão de f a cada 30 iterações se Z_{min} não diminuir nas últimas 30 iterações.

Grupo	Solução Ótima	Solução Obtida	Número de nós	Tempo (min)
1A	31,90	31,90	6,60	0,28
1B	44,20	44,20	21,00	2,26
2A	48,56	48,44	46,50	4,49
2B	42,60	42,60	97,20	12,51

Tabela 6.15. Resultados computacionais obtidos com o método LAG utilizando $f=2$ e divisão de f a cada 15 iterações se Z_{min} não diminuir nas últimas 15 iterações.

Grupo	Solução Ótima	Solução Obtida	Número de nós	Tempo (min)
1A	31,90	31,90	13,20	0,28
1B	44,20	44,20	33,20	1,71
2A	48,56	48,44	68,25	3,64
2B	42,60	42,60	152,40	10,02

Observe nas Tabelas 6.10 a 6.15 que para o Grupo 2A não foi possível encontrar a solução ótima em todos os exemplos em um tempo computacional de 3 horas. Observe também que, em média, reduzir o parâmetro f a cada 15 ou 30 iterações

apresenta melhores resultados quanto ao tempo computacional. Assim, consideraremos nas tabelas seguintes a redução de f a cada 30 iterações. Com relação ao número de nós envolvidos na árvore de busca, observa-se que quanto menor o número de iterações no procedimento do subgradiente, maior o número de nós envolvidos na busca em árvore, ou seja, um *trade-off* entre os tempos computacionais envolvidos na otimização do subgradiente e na busca em árvore, como era de se esperar. Podemos observar também que o tempo computacional gasto para se resolver os exemplos Grupos 1B e 2B são maiores em relação aos Grupos 1A e 2A respectivamente. Isto ocorre provavelmente porque os problemas dos Grupos 1B e 2B apresentam *gap* entre o limitante de relaxações Linear ou Lagrangiana e a solução ótima, diferente dos problemas dos Grupos 1A e 2A, o que os torna mais difíceis para o método LAG.

Pelos resultados apresentados nas tabelas anteriores, podemos dizer que em média, o tempo computacional gasto no nó inicial da árvore, é uma pequena fração do tempo total gasto para se resolver o exemplo. A seguir, na Tabela 6.16, apresentamos os valores médios obtidos com os grupos de exemplos utilizando o parâmetro $f = 1$ com a divisão de f a cada 30 iterações. Também foram realizados experimentos utilizando $f = 1$ e as outras variações de f , mas os resultados obtidos não foram melhores.

Tabela 6.16. Resultados computacionais obtidos com o método LAG utilizando $f=1$ e divisão de f a cada 30 iterações.

Grupo	Solução Ótima	Solução Obtida	Número de nós	Tempo (min)
1A	31,90	31,90	10,0	0,26
1B	44,20	44,20	29,20	1,27
2A	48,56	48,56	95,00	15,72
2B	42,60	42,60	131,60	6,30

Observe pela Tabela 6.16 que foi possível encontrar a solução ótima em todos os grupos de exemplos, diferente das tabelas anteriores 6.10 a 6.15. O tempo médio apresentado para o Grupo 2A na Tabela 6.16 é maior do que os apresentados nas tabelas anteriores pois, como dissemos anteriormente, todos os exemplos deste grupo foram resolvidos. Para os demais grupos, podemos observar pelas Tabelas 6.11 e 6.16 que considerar inicialmente o parâmetro $f = 1$ apresentou melhores resultados

computacionais. Assim, nas tabelas seguintes consideramos sempre o parâmetro $f = 1$ sendo reduzido a cada 30 iterações. Quanto ao número de nós, podemos notar que para o Grupo 2B, em média, esse número decresceu significativamente considerando inicialmente o parâmetro $f = 1$, enquanto que para os outros grupos não houve grandes modificações nos números de nós considerando $f = 2$ ou $f = 1$ com sua redução a cada 30 iterações.

Foram feitos ainda testes com os exemplos da literatura dos Grupos 1A, 1B, 2A e 2B com as modificações propostas por CAMERINI *et al.* (1975) apresentadas no Capítulo 4 (seção 4.5.2.2). A Tabela 6.17 apresenta os resultados computacionais com as modificações de CAMERINI *et al.* (1975) no procedimento de otimização do subgradiente.

Tabela 6.17. Resultados computacionais obtidos com o método LAG utilizando CAMERINI *et al.* (1975).

Grupo	Solução Ótima	Solução Obtida	Número de nós	Tempo (min)
1A	31,90	31,90	9,40	0,47
1B	44,20	44,20	27,80	2,66
2A	48,56	48,45	53,75	4,43
2B	42,60	42,60	93,60	8,82

Observe na Tabela 6.17 que no Grupo 2A não foi possível obter a solução ótima em todos os exemplos, em menos de 3 horas. Assim, podemos dizer que considerar o procedimento de otimização sem as modificações propostas por CAMERINI *et al.* (1975) apresenta melhores resultados quanto ao tempo computacional. A Tabela 6.18 apresenta os resultados computacionais do método LAG com os *raster points* apresentado na seção 4.4.5.

Tabela 6.18. Resultados computacionais obtidos com o método LAG utilizando os *raster points*.

Grupo	Solução Ótima	Solução Obtida	Número de nós	Tempo (min)
1A	31,90	31,90	10,0	0,18
1B	44,20	44,20	25,20	0,72
2A	48,56	48,56	76,00	4,55
2B	42,60	42,60	105,80	2,67

Podemos observar pela Tabela 6.18 que utilizando os *raster points* o método LAG foi capaz de resolver todos os exemplos em tempos computacionais menores do que os apresentados na Tabela 6.16, especialmente para os Grupos 2A e 2B. Assim, podemos dizer que considerar os *raster points* contribuiu para um melhor desempenho do método, devido ao número de variáveis envolvidas no modelo ser reduzido. Ainda, podemos notar pelas Tabelas 6.16 e 6.18 que o número de nós envolvidos na busca em árvore foi um pouco menor considerando os *raster points*. A tabela a seguir apresenta os resultados obtidos considerando a estratégia de escolha a ser fixada na divisão dos nós usando a idéia de memória de longo prazo da meta-heurística Busca Tabu apresentada na seção 5.4.1.

Tabela 6.19. Resultados computacionais obtidos com o método LAG utilizando os *raster points* com a ramificação dos nós usando a idéia de memória de longo prazo de Busca Tabu.

Grupo	Solução Ótima	Solução Obtida	Número de nós	Tempo (min)
1A	31,90	31,90	7,00	0,17
1B	44,20	44,20	22,60	1,17
2A	48,56	48,56	75,00	4,58
2B	42,60	42,60	113,60	3,34

Podemos observar pelas Tabelas 6.18 e 6.19 que a incorporação da estratégia de escolha a ser fixada na divisão dos nós usando a idéia de memória de longo prazo da meta-heurística Busca Tabu com os *raster points* apresentou resultados, em geral, um pouco piores do que os apresentados na Tabela 6.18. Assim, a estratégia de escolha da variável a ser fixada na divisão dos nós proposta por BEASLEY (1985b) foi mantida nos experimentos seguintes.

Na tabela a seguir apresentamos os resultados computacionais considerando a heurística de MORABITO e MORALES (1998) (seção 5.4.2) para produzir uma boa solução factível inicial (limitante inferior). Pelas Tabelas 6.18 e 6.20 podemos dizer que a heurística de MORABITO e MORALES (1998) foi capaz de reduzir os tempos computacionais 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.

Tabela 6.20. Resultados computacionais obtidos com o método LAG utilizando os *raster points* com a heurística de MORABITO e MORALES (1998).

Grupo	Solução Ótima	Solução Obtida	Número de nós	Tempo (min)
1A	31,90	31,90	1,00	< 0,01
1B	44,20	44,20	18,00	0,70
2A	48,56	48,56	1,00	0,02
2B	42,60	42,60	104,00	3,31

Podemos concluir diante dos resultados computacionais apresentados para os Grupos 1A, 1B, 2A e 2B que considerar os *raster points* com a heurística de MORABITO e MORALES (1998) apresentou melhores resultados quanto ao tempo computacional (veja Tabela 6.20). Verificamos que o procedimento de busca em árvore é capaz de resolver os exemplos dos Grupos 1A, 1B, 2A e 2B em um tempo computacional menor do que os apresentados pelo pacote GAMS/CPLEX (Tabela 6.8). Vemos pela Tabela 6.20 que, em média, o tempo do método LAG é menor que 0,01 minutos, 0,70 minutos, 0,02 minutos e 3,31 minutos para os Grupos 1A, 1B, 2A e 2B, respectivamente, enquanto que o encontrado pelo pacote GAMS/CPLEX (Tabela 6.8) é de 0,19 minutos, 0,81 minutos, 4,05 minutos e 4,87 minutos, respectivamente, para os mesmos grupos de exemplos.

Fizemos ainda testes computacionais com todos os 3179 exemplos do conjunto Cover I analisado em MORABITO e MORALES (1998), que cobre exemplos com solução ótima composta de até 50 caixas. Dentre esses exemplos, 116 apresentam *gap*, ou seja, possuem o valor da solução relaxada (Linear ou Lagrangiana) diferente do valor da solução ótima. A Tabela 6.21 apresenta resumidamente os resultados obtidos com o método LAG para este conjunto de exemplos. O número de elementos dos conjuntos normais e *raster points* são dados, respectivamente, por $|X| = 25,9$, $|Y| = 14,2$, $|X'| = 19,8$ e $|Y'| = 11,0$. O número (aproximado) de variáveis 0–1 envolvidas no modelo (4.6)-(4.8) é de $2|X||Y| = 735,56$ e o número de variáveis envolvidas no modelo considerando os *raster points* é de $2|X' ||Y'| = 435,60$. Note que 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 (conjunto Cover I).

Tabela 6.21. Resultados computacionais obtidos com o método LAG utilizando os *raster points* com a heurística de MORABITO e MORALES (1998) para o conjunto Cover I.

Cover I	Solução Ótima	Solução Obtida	Número de nós	Tempo (min)
Exemplos sem <i>gap</i> (3063/3179)	35,98	35,98	1,00	< 0,01
Exemplos com <i>gap</i> (116/3179)	41,82	41,82	23,79	0,79
Total (3179)	36,19	36,19	1,83	0,03

6.5. RESULTADOS UTILIZANDO O GRUPO 3

Nesta seção é apresentada a análise dos resultados do método LAG utilizando os Grupos 3A e 3B. Convém lembrar que o procedimento de otimização do subgradiente foi iniciado com o parâmetro $f = 1$ (*Passo 6*, seção 4.5.2.2), sendo reduzido a cada 30 iterações, pois como visto para os Grupos 1A, 1B, 2A e 2B de exemplos da literatura, foi o que apresentou melhor resultado computacional. É importante lembrar também que estes conjuntos de exemplos são pertencentes ao conjunto Cover II (que é composto de exemplos com soluções ótimas entre 50 e 100 caixas) e, em especial, o conjunto 3B possui exemplos considerados difíceis de se resolver com as heurísticas encontradas na literatura (LINS *et al.*, 2003).

Na Tabela 6.22 apresentamos os resultados dos testes computacionais dos Grupos 3A e 3B. São apresentados, em média, o valor da solução ótima, a solução obtida em um tempo computacional máximo limitado a 3 horas, o número de nós envolvidos na busca em árvore e o tempo médio (em minutos) gasto para se executar o conjunto de exemplos.

Tabela 6.22. Resultados computacionais obtidos com o método LAG utilizando $f=1$ e a divisão de f a cada 30 iterações.

Grupo	Solução Ótima	Solução Obtida	Número de nós	Tempo (min)
3A	73,31	73,25	91,67	26,71
3B	80,06	79,75	192,09	41,60

Observe pela Tabela 6.22 que não foi possível encontrar a solução ótima em todos os exemplos em um tempo computacional de 3 horas. No Grupo 3A, o método LAG não encontrou a solução ótima em apenas 1 exemplo, enquanto que no Grupo 3B o número de exemplos não resolvidos foi maior, 5 exemplos. Foram feitos testes com os exemplos dos Grupos 3A e 3B com as modificações propostas no trabalho de CAMERINI *et al.* (1975) apresentadas no Capítulo 4 (seção 4.5.2.2), os quais são apresentados na tabela a seguir. Pela Tabela 6.23 podemos observar que considerar a proposta de CAMERINI *et al.* (1975) não apresentou melhores resultados quanto ao tempo computacional (veja Tabela 6.22). Assim, podemos concluir que considerar o procedimento de otimização do subgradiente como proposto no Capítulo 4, seção 4.5.2.2, apresenta melhores resultados computacionais.

Tabela 6.23. Resultados computacionais obtidos com o método LAG utilizando CAMERINI *et al.* (1975).

Grupo	Solução Ótima	Solução Obtida	Número de nós	Tempo (min)
3A	73,31	73,19	70,00	29,81
3B	80,06	79,75	150,09	42,44

A Tabela 6.24 apresenta os resultados computacionais do método LAG utilizando os *raster points* (seção 4.4.5). Podemos observar que com a consideração dos *raster points*, o método LAG foi capaz de resolver um número maior de exemplos e ainda, os tempos computacionais obtidos foram bem menores dos que os apresentados na Tabela 6.22. No Grupo 3B o método LAG não resolveu otimamente apenas 2 exemplos.

Tabela 6.24. Resultados computacionais obtidos com o método LAG utilizando os *raster points*.

Grupo	Solução Ótima	Solução Obtida	Número de nós	Tempo (min)
3A	73,31	73,31	122,20	15,47
3B	80,06	79,94	254,38	21,48

Na tabela a seguir apresentamos os resultados obtidos considerando a estratégia de escolha da variável a ser fixada na divisão dos nós usando a idéia de memória de longo prazo de Busca Tabu apresentada no Capítulo 5 (seção 5.4.1). Podemos observar que os tempos computacionais obtidos considerando a estratégia de

escolha da variável a ser fixada na divisão dos nós usando a idéia de memória de longo prazo de Busca Tabu considerando os *raster points* foram maiores dos que os obtidos pela Tabela 6.24. Além disso, podemos dizer que, para o Grupo 3B, o número de exemplos resolvidos foi maior usando a estratégia de BEASLEY (1985b) (veja Tabela 6.24).

Tabela 6.25. Resultados computacionais obtidos com o método LAG utilizando os *raster points* com a ramificação dos nós usando a idéia de memória de longo prazo de Busca Tabu.

Grupo	Solução Ótima	Solução Obtida	Número de nós	Tempo (min)
3A	73,31	73,31	135,53	21,16
3B	80,06	79,88	314,38	29,50

A Tabela 6.26 apresenta os resultados utilizando os *raster points* e a heurística de MORABITO e MORALES (1998). Podemos dizer que considerar os *raster points* e a heurística de MORABITO e MORALES (1998) foi capaz de reduzir significativamente os tempos computacionais nos exemplos do Grupo 3A (de 15,47 minutos para 1,34 minutos), cujas soluções ótimas são não-guilhotinadas de 1ª ordem. Já no Grupo 3B, estes tempos foram bem similares, mas o número de exemplos não resolvidos (3 exemplos) é maior do que o da Tabela 6.24 (2 exemplos).

Tabela 6.26. Resultados computacionais obtidos com o método LAG utilizando os *raster points* com a heurística de MORABITO e MORALES (1998).

Grupo	Solução Ótima	Solução Obtida	Número de nós	Tempo (min)
3A	73,31	73,31	12,63	1,34
3B	80,06	79,88	249,31	22,76

Observe pelas Tabelas 6.8 e 6.26 que o tempo médio obtido pelo pacote GAMS/CPLEX para o Grupo 3A (9,23 minutos) foi bem maior do que o obtido pelo método LAG (1,34 minutos). Por outro lado, para o Grupo 3B, note que tempo computacional médio encontrado pelo método LAG (22,76 minutos) foi superior ao encontrado pelo pacote GAMS/CPLEX (17,55 minutos).

LINS *et al.* (2003) realizaram testes computacionais com o Grupo 3B e os resultados apresentados mostraram que o método exato proposto pelos autores 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 microcomputador utilizado foi um Pentium III 700 MHz, ou seja, comparável ao microcomputador utilizado no presente trabalho (Pentium III 650 MHz). A seguir, apresentamos os resultados encontrados em LINS *et al.* (2003) e comparamos com os melhores resultados obtidos com o método LAG.

Tabela 6.27. Resultados computacionais obtidos em LINS *et al.* (2003) e com o método LAG.

Exemplo	Solução Ótima	LINS <i>et al.</i> (2003)		Método LAG	
		Solução Obtida	Tempo (min)	Solução Obtida	Tempo (min)
L56	53	53	2,83	53	2,52
L57	57	57	6,27	57	2,85
L58	69	69	13,9	69	17,95
L59	69	69	35,12	69	73,60
L60	71	71	31,65	71	8,56
L61	75	75	52,08	75	10,08
L62	77	77	29,9	77	17,71
L63	77	77	23,15	77	11,03
L64	81	81	46,08	81	19,48
L65	82	82	131,52	82	22,60
L66	82	82	134,78	82	23,68
L67	96	95	180,00	96	47,07
L68	96	95	180,00	95	180,00
L69	97	97	99,87	96	180,00
L70	99	99	179,3	98	180,00
L71	100	99	180,00	100	38,79
Média	80,06	79,88	60,50	79,88	22,76

Pela Tabela 6.27 podemos observar que o método LAG resolveu a maioria dos exemplos em um tempo computacional menor do que os apresentados em LINS *et al.* (2003). Entretanto, podemos notar que o método LAG não resolveu 3 exemplos (L68, L69 e L70) em um tempo máximo de 3 horas, enquanto que o método

em LINS *et al.* (2003) também não resolveu 3 exemplos dentro deste limite (L67, L68 e L71). Recentemente BIRGIN *et al.* (2004) realizaram experimentos computacionais com a abordagem *L* de LINS *et al.* (2003), explorando estruturas de dados diferentes daquelas utilizadas em LINS *et al.* (2003), mostrando que os tempos computacionais da abordagem *L* podem ser reduzidos. No entanto, os requisitos de memória computacional são bem maiores do que os anteriores.

Também fizemos testes computacionais com todos os 16938 exemplos do conjunto Cover II analisado em MORABITO e MORALES (1998), que cobre exemplos com solução ótima composta de 51 a 100 caixas. A Tabela 6.28 apresenta resumidamente os resultados obtidos com o método LAG para este conjunto de exemplos. O número de elementos dos conjuntos normais e *raster points* são dados, respectivamente, por $|X| = 54,6$, $|Y| = 29,7$, $|X'| = 38,4$ e $|Y'| = 20,8$. O número (aproximado) de variáveis 0–1 envolvidas no modelo (4.6)-(4.8) é de $2|X||Y| = 3243,24$ e o número de variáveis envolvidas no modelo considerando os *raster points* é de $2|X'| |Y'| = 1597,44$.

Tabela 6.28. Resultados computacionais obtidos com o método LAG utilizando os *raster points* com a heurística de MORABITO e MORALES (1998) para o conjunto Cover II.

Cover II	Solução Ótima	Solução Obtida	Número de nós	Tempo (min)
Total (16938)	80,22	80,16	18,32	3,27

Pela Tabela 6.28 podemos notar que não foi possível encontrar a solução ótima em todos os exemplos em um tempo máximo de 3 horas. O método LAG, de um total de 16938 exemplos, não resolveu 82 exemplos deste conjunto. Convém observar que, nos exemplos não resolvidos pelo método LAG do conjunto Cover II, a melhor solução obtida no tempo máximo de 3 horas ou fica a 1 caixa da solução ótima ou, é ótima mas o método não conseguiu provar que ela é a ótima.

6.6. RESULTADOS OBTIDOS COM OS EXEMPLOS REAIS

Nesta seção apresentamos os resultados computacionais para os exemplos reais (Grupo 4) coletados no centro de distribuição da Transportadora Americana, em Campinas, São Paulo. Como dito anteriormente, selecionamos arbitrariamente 30 exemplos e, nestes exemplos estamos considerando o palete Brasil (PBR: 120x100 cm) como palete padrão, já que este é o usado na Transportadora Americana e na maioria das empresas no Brasil.

As Tabelas 6.29 e 6.30 apresentam os resultados dos testes computacionais partindo de $f = 1$ e dividindo-o a cada 30 iterações considerando os *raster points* e também considerando os *raster points* com a heurística de MORABITO e MORALES (1998), respectivamente. São apresentados os resultados dos testes computacionais dos exemplos R1-R30 com a solução utilizada pela Transportadora Americana. Apresentamos também o limitante superior (Z_{min}) e o limitante inferior (Z_{LB}) no nó inicial da árvore. Ainda são apresentados o valor da solução ótima, o número de nós na árvore de busca e o tempo total (em minutos) gasto para se executar cada exemplo. Apresentamos também, a diferença entre a solução apresentada pela Transportadora Americana e a solução ótima encontrada pelo método LAG.

Convém lembrar que para estes testes foi considerado a estratégia de escolha da variável a ser fixada na divisão dos nós proposta por BEASLEY (1985b) pois, como visto para os Grupos 1A, 1B, 2A, 2B, 3A e 3B, foi a estratégia que apresentou melhor resultado quanto ao tempo computacional.

Tabela 6.29. Resultados computacionais do Grupo 4 obtidos com o método LAG com a divisão de f a cada 30 iterações com os *raster points*.

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	15	16	1	0,02	2
R2	10	12	12	12	1	< 0,01	2
R3	13	15	15	15	1	< 0,01	2
R4	12	12	9	12	1	< 0,01	0
R5	20	21	18	20	1	0,01	0
R6	18	19	16	19	1	0,02	1
R7	18	20	18	20	3	0,03	2
R8	10	10	9	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	21	23	3	0,05	0
R15	10	10	9	10	1	0,01	0
R16	25	27	24	26	1	< 0,01	1
R17	18	19	16	19	1	0,02	1
R18	46	47	40	46	85	3,90	0
R19	26	26	25	26	3	0,04	0
R20	15	15	15	15	1	< 0,01	0
R21	20	23	20	22	3	0,04	2
R22	41	43	42	43	37	0,59	2
R23	8	8	8	8	1	< 0,01	0
R24	6	6	6	6	1	< 0,01	0
R25	14	15	12	15	1	0,02	1
R26	14	17	15	15	1	0,01	1
R27	40	42	42	42	1	< 0,01	2
R28	27	31	28	31	3	0,06	4
R29	15	16	15	15	1	< 0,01	0
R30	37	38	36	38	11	0,14	1
Média				20,10	5,67	0,17	0,97

Tabela 6.30. Resultados computacionais do Grupo 4 obtidos com o método LAG com a divisão de f a cada 30 iterações com os *raster points* e com a heurística de MORABITO e MORALES (1998).

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

Pelas Tabelas 6.29 e 6.30 podemos dizer que o programa foi capaz de resolver otimamente todos os exemplos em um tempo computacional relativamente pequeno (0,17 e 0,10 minutos em média, respectivamente), o que sugere que ele pode ser uma boa alternativa para resolver exemplos reais. Observamos ainda que em vários exemplos a solução apresentada pela Transportadora Americana não era a solução ótima e dessa forma podemos dizer que, em média, o método LAG é capaz de melhorar em 0,97 caixas (por camada do palete) a solução apresentada pela Transportadora. Observe nos exemplos R9 e R28 que a solução exata é capaz de colocar 4 caixas a mais que a solução apresentada pela Transportadora Americana. Note que dos 30 exemplos, 16 das soluções apresentadas pela Transportadora Americana não são ótimas. Ainda, podemos dizer que considerar os *raster points* com a heurística de MORABITO e MORALES (1998) (Tabela 6.30) apresentou melhores resultados quanto ao tempo computacional e, também, um número menor de nós na busca em árvore.

Observe pelas Tabelas 6.8 e 6.30 que, em média, o método LAG proposto resolveu todos os 30 exemplos em um tempo computacional (0,10 minutos) muito menor do que o obtido pelo pacote GAMS/CPLEX (18,52 minutos). Assim, como dito anteriormente, podemos dizer que o método LAG proposto pode ser uma boa alternativa para resolver exemplos reais.

6.7. CONSIDERAÇÕES FINAIS

A Tabela 6.31 apresenta um resumo dos valores médios obtidos para os grupos de exemplos da literatura e reais. Podemos dizer que o método exato proposto resolve os conjuntos de exemplos em um tempo computacional aceitável. Podemos observar ainda que, exceto o Grupo 3B, em todos os outros grupos o método LAG resolveu todos os exemplos.

Tabela 6.31. Resultados computacionais obtidos com o método LAG utilizando os *raster points* com a heurística de MORABITO e MORALES (1998).

Grupos	Solução Ótima	Solução Obtida	Número de nós	Tempo (min)
1A	31,90	31,90	1,00	< 0,01
1B	44,20	44,20	18,00	0,70
2A	48,56	48,56	1,00	0,02
2B	42,60	42,60	104,00	3,31
3A	73,31	73,31	12,63	1,34
3B	80,06	79,88	249,31	22,76
4	20,10	20,10	3,47	0,10

Podemos dizer que o método LAG apresentou um bom desempenho computacional em relação à outros métodos exatos e ao *software* GAMS/CPLEX. Pelas Tabelas 6.8 e 6.31 podemos observar que, a menos do Grupo 3B, o método LAG resolveu os grupos de exemplos em um tempo menor do que o apresentado pelo GAMS/CPLEX. No grupo de exemplos reais (Grupo 4) podemos notar que o tempo gasto pelo método LAG (0,10 minutos) é, em média, muito menor do que o obtido pelo pacote GAMS/CPLEX (18,52 minutos), o que mostra que o método LAG é eficiente para resolver problemas reais. Nos exemplos não resolvidos pelo método LAG (exemplos do Grupo 3B), a melhor solução obtida pelo método, em um tempo máximo de 3 horas, fica apenas a 1 caixa do valor ótimo.

CAPÍTULO 7

CONCLUSÕES E PROPOSTAS FUTURAS

7.1. CONCLUSÕES

Esse trabalho abordou o PCP do produtor. Mais especificamente, o problema de arranjar ortogonalmente e sem sobreposição o máximo número de retângulos de dimensões (l,w) ou (w,l) sobre um retângulo maior. Devido à escala e extensão de certos sistemas logísticos, um pequeno aumento no número de produtos carregados sobre o palete pode resultar em economias substanciais. O PCP do produtor também é importante do ponto de vista computacional, pois aparentemente simples de ser resolvido otimamente, o problema não é resolvido em tempo polinomial, embora ainda não seja provado pertencer a classe NP-difícil. A solução deste problema possui importância técnica e econômica nos processos industriais e logísticos. No nível estratégico, uma análise estatística baseada na solução do PCP do produtor pode ser útil para escolher o palete mais adequado para uma empresa (dentro um conjunto de paletes padronizados disponíveis), assim como para projetar novos paletes (MORALES *et al.*, 1997), ou ainda, o desenvolvimento de novas embalagens de produtos, que depende de fatores logísticos e mercadológicos. No nível operacional, métodos mais efetivos para resolver o PCP do produtor permitem encontrar melhores padrões de empacotamento para a armazenagem e distribuição de um produto.

Os objetivos deste trabalho foram revisar o estado da arte do PCP do produtor, desenvolver métodos exatos competitivos baseados em relaxação Lagrangiana e/ou *Surrogate* para resolver este problema e avaliar o desempenho destes métodos em relação a outros métodos exatos com exemplos da literatura e situações práticas. Convém lembrar que na literatura há poucos autores que propuseram métodos exatos para resolver o PCP do produtor. Foram apresentadas diferentes formulações matemáticas para resolver o PCP do produtor, entre elas, o modelo inteiro 0–1 (modelo (3-1)-(3.4), apresentado no Capítulo 3) que pode ser visto como um caso particular do proposto em BEASLEY (1985b) para o problema mais geral de corte e empacotamento. Três métodos foram estudados baseados neste modelo: (i) o método LAG utilizando relaxação Lagrangiana; (ii) o método SUR utilizando relaxação *Surrogate* e (iii) o método LAGSUR combinando relaxação Lagrangiana e relaxação *Surrogate*. Este último é similar ao método proposto em BEASLEY (1985b) para o problema mais geral de corte bidimensional não-guilhotinado. Foi utilizado o método *branch and bound*: consideramos uma árvore binária com busca em profundidade e, em cada nó da árvore, computamos um limitante superior de relaxação Lagrangiana e/ou *Surrogate* por meio de um procedimento de otimização do subgradiente. Convém lembrar que durante cada iteração deste procedimento foram aplicadas heurísticas Lagrangiana ou *Surrogate* para gerar soluções factíveis para o problema original (3.1)-(3.4) e técnicas de redução do problema para tentar fixar, sem perda de generalidade, alguma variável em 0 ou 1, e desta maneira, reduzir o tamanho do problema (3.1)-(3.4).

Após realizar as implementações dos métodos LAG, SUR e LAGSUR como descritas no Capítulo 5, iniciaram-se os testes computacionais utilizando os grupos de exemplos indicados no Capítulo 6. Para os testes computacionais consideramos exemplos da literatura (71 exemplos, Grupos 1, 2 e 3) e exemplos reais (30 exemplos, Grupo 4). Realizamos ainda testes com os conjuntos da literatura Cover I (3179 exemplos) e Cover II (16938 exemplos), que cobrem famílias de exemplos com soluções até 100 caixas.

Inicialmente foram realizados diversos testes computacionais para se verificar o desempenho de possíveis variações nos procedimentos de otimização do subgradiente e da busca em árvore. No procedimento de otimização do subgradiente foram realizados testes com diversos exemplos quanto à aplicação ou não da heurística

Lagrangiana; quanto à aplicação ou não do método de redução do problema e ainda, foi testada a exclusão do procedimento de redução juntamente com a heurística Lagrangiana. Podemos dizer diante dos resultados obtidos que o procedimento de redução do problema no método LAG é eficaz, dado que o esforço computacional de tentar fixar variáveis ao longo das iterações é compensatório. O mesmo acontece para o programa sem a inclusão da heurística Lagrangiana e do procedimento de redução. Já para o programa apenas sem a inclusão da heurística Lagrangiana, o tempo computacional foi um pouco menor (em média) do que com a inclusão da heurística Lagrangiana, o que sugere que a heurística não é efetiva nestes exemplos. Ainda, foi testado o procedimento de melhoria na heurística Lagrangiana apresentado na seção 5.2.1. No entanto, tal procedimento não foi capaz de reduzir os tempos computacionais até então obtidos apenas com a utilização da heurística Lagrangiana. Consideramos também a proposta do trabalho de LUCENA (2004), no qual o autor apresenta algumas modificações *relax and cut* (seção 4.5.2.2). Tais testes computacionais foram realizados com diversos exemplos e os resultados não foram melhores do que os obtidos sem essas modificações. Uma outra maneira que pode ser utilizada para se resolver o problema dual Lagrangiano utilizando o procedimento de otimização do subgradiente é incorporar as modificações propostas por CAMERINI *et al.* (1975). Foram realizados vários testes computacionais apresentados no Capítulo 6, os quais também não resultaram em melhoria no tempo computacional. Ainda, o procedimento de otimização do subgradiente foi iniciado com os parâmetros $f=2$ ou $f=1$ (Passo 6, seção 4.5.2.2), sendo reduzido após um determinado número de iterações. Foram exploradas algumas variações para reduzir o valor de f ao longo das iterações como apresentado no Capítulo 6, e os melhores resultados computacionais foram obtidos considerando-se inicialmente $f = 1$ sendo reduzido a cada 30 iterações.

Na busca em árvore foram também feitos vários testes com as diferentes regras para a seleção da variável a ser fixada na divisão dos nós. Dentre as cinco regras propostas (veja seção 5.4.1), a que apresentou melhores resultados quanto ao tempo computacional foi a proposta por BEASLEY (1985b). Consideramos também executar um número de iterações maior no nó inicial da árvore, conforme discutido na seção 5.4, Capítulo 5. No entanto, tal consideração não apresentou ganhos em relação ao tempo computacional. Ressalta-se que 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, o que garante uma convergência mais rápida do método. No caso de não considerarmos os multiplicadores de Lagrange e/ou *Surrogate* associados com o melhor limitante superior encontrado no nó predecessor da árvore (por exemplo, inicializados com valor 0 ou 1), os tempos computacionais dos métodos são muito mais altos. Ainda, realizamos testes computacionais com o conjunto dos *raster points* (seção 4.4.5) e podemos dizer que considerar os *raster points* contribuiu para um melhor desempenho do método, devido ao número de variáveis envolvidas no modelo ser reduzido. Também consideramos a heurística de MORABITO e MORALES (1998) (seção 5.4.2) e diante dos resultados computacionais obtidos podemos dizer que considerar os *raster points* com a heurística de MORABITO e MORALES (1998) apresentou melhores resultados quanto ao tempo computacional.

Em seguida realizamos testes computacionais com o Grupo 1A com as três abordagens LAG, SUR e LAGSUR, e a que apresentou melhor resultado quanto ao tempo computacional foi a que utiliza apenas relaxação Lagrangiana. O método LAG conseguiu resolver todos os exemplos do Grupo 1A em menos de 3 horas, diferente dos demais. Outros experimentos foram realizados com diversos exemplos dos Grupos 1, 2, 3 e 4, variando-se as divisões do parâmetro f da otimização do subgradiente, e o método LAG continuou apresentando desempenho bem superior aos métodos SUR e LAGSUR. Uma vantagem do método LAG sobre o método LAGSUR é o número de multiplicadores bem maior ($|X||Y|$ ao invés de $|X|+|Y|$), o que contém bem 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 relaxação *Surrogate* não tem garantia de convergência, conforme discussão na seção 4.5.3.2. Assim, devido aos resultados obtidos, considerando que os métodos SUR e LAGSUR tiveram dificuldades para resolver mesmo exemplos mais fáceis como os do Grupo 1A, resolvemos abandoná-los, e os todos os outros testes foram realizados considerando apenas o método LAG.

Podemos dizer pelos resultados apresentados nas Tabelas 6.20, 6.26 e 6.30 que o método LAG resolve os conjuntos de exemplos em um tempo computacional aceitável levando-se em conta as decisões envolvidas na prática, tanto estratégicas quanto operacionais. Podemos observar ainda que, exceto o Grupo 3B (com padrões ótimos não-guilhotinados de ordem superior), em todos os outros grupos o método exato resolveu todos os exemplos dentro do limite de tempo. Ainda, pelas Tabelas 6.8, 6.20 e 6.26 podemos observar que, a menos do Grupo 3B, o método LAG resolveu os grupos de exemplos em um tempo médio menor do que o apresentado pelo pacote GAMS/CPLEX. Ressalta-se que no Grupo 3B os exemplos são considerados difíceis de se resolver com as heurísticas encontradas na literatura e incluímos este grupo com o objetivo de verificar o comportamento de tais exemplos em relação ao método exato. Comparamos também os resultados obtidos pelo método LAG, considerando os exemplos do Grupo 3B, com os resultados apresentados em LINS *et al.* (2003). Pela Tabela 6.27 podemos observar que o método LAG resolveu a maioria dos exemplos em um tempo computacional menor do que os apresentados em LINS *et al.* (2003). Ressalta-se também que no grupo de exemplos reais coletados da transportadora (Grupo 4, Tabela 6.30), podemos notar que o tempo gasto pelo método exato é, em média, muito menor do que o obtido pelo pacote GAMS/CPLEX, o que sugere que o método exato seja eficiente para aplicações práticas.

A partir dos resultados obtidos, podemos dizer que o método LAG apresentou um bom desempenho computacional e pode-se concluir que esta tese contribuiu com métodos exatos baseados em relaxação Lagrangiana e/ou *Surrogate* para o PCP do produtor, com bom desempenho em relação a outros métodos exatos e ao pacote GAMS/CPLEX. Apesar do desempenho do método LAG ser competitivo, capaz de gerar soluções ótimas e provar suas otimalidades em tempos computacionais relativamente pequenos (considerando as decisões envolvidas), nossa expectativa inicial era de que ele teria desempenho computacional melhor do que o obtido ao longo dos experimentos.

7.2. PROPOSTAS FUTURAS

Várias propostas para pesquisas futuras foram apontadas ao longo do texto e uma delas é tentar reduzir os tempos computacionais dos métodos LAG, SUR e LAGSUR. Em particular pretende-se considerar as idéias apresentadas no trabalho recente de ALVAREZ-VALDES *et al.* (2004) 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 aplicar técnicas de *relax and cut* no PCP do produtor, além daquelas utilizadas neste trabalho.

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 e LETCHFORD (2001) 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. Lembramos que não encontramos um contra-exemplo do PCP do produtor (usando P e Q como um dos limitantes inferiores e superiores definidos no Capítulo 4), mostrando que o problema *Surrogate* não satisfaz a propriedade de integralidade. Também não encontramos, ao longo dos experimentos computacionais do Capítulo 6, um exemplo com limitante produzido pela relaxação *Surrogate* melhor do que o limitante produzido pela relaxação Lagrangiana.

REFERÊNCIAS BIBLIOGRÁFICAS

- ALVAREZ-VALDES, R.; PARREÑO, F. e TAMARIT, J. M., A Branch-and-Cut Algorithm for the Pallet Loading Problem. Aceito para publicação no **Computers & Operations Research**, 2004.
- AMARAL, A. e LETCHFORD, A., Comment on "An Exact Algorithm for General, Orthogonal, Two-dimensional Knapsack Problems". Submetido para **European Journal of Operational Research**, 2001.
- ARENALES, M. e MORABITO, R., An AND/OR-Graph to the Solution of Two-Dimensional Non-guilhotine Cutting Problems. **European Journal of Operational Research**, v. 84, p. 599-617, 1995.
- ARENALES, M.; MORABITO, R. e YANASSE, H. (eds), Special Issue: Cutting and Packing Problems. **Pesquisa Operacional**, v. 19, n. 2, p. 107-299, 1999.
- BALLOU, R. H., **Logística Empresarial: Transportes, Administração de Materiais e Distribuição Física**. Tradução: Hugo T. Y. Yoshizaki, São Paulo: Atlas, 1993.
- BALLOU, R. H., **Business Logistics Management**. Weatherhead School of Management: Prentice Hall, 1999.

- BANERJEE, K., **Generalized Lagrange Multipliers in Dynamic Programming.** Research Report No. OCR 71-12, Operations Research Center, University of California, Berkeley, California, 1971.
- BARNES, F. W., Packing the Maximum Number of $m \times n$ Tiles in a Large $p \times q$ Rectangle. **Discrete Mathematics**, v. 26, p. 93-100, 1979.
- BARNETT, S. e KYNCH, G. J., Exact Solution of a Simple Cutting Problem. **Operations Research**, v. 15, p. 1051-1056, 1967.
- BEASLEY, J., Bounds for Two-Dimensional Cutting. **Journal of the Operational Research Society**, v. 36, p. 71-74, 1985a.
- BEASLEY, J., An Exact Two-Dimensional Non Guillotine Cutting Tree Search Procedure. **Operations Research**, v. 33, p. 49-64, 1985b.
- BEASLEY, J., A Population Heuristic for Constrained Two-Dimensional Non-Guillotine Cutting. **European Journal of Operational Research**, v. 156, n. 3, p. 601-627, 2004.
- BHATTACHARYA, S.; ROY, R. e BHATTACHARYA, S., An Exact Depth-First Algorithm for the Pallet Loading Problem. **European Journal of Operational Research**, v. 110, p. 610-625, 1998.
- BIRGIN, E. G.; MORABITO, R. e NISHIHARA, F. H., A Note on a L-Approach for Solving the Manufacturer's Pallet Loading Problem. Submetido para **Journal of the Operational Research Society**, 2004.
- BIRÓ, M. e BOROS, E., Network Flows and Non-Guillotine Cutting Patterns. **European Journal of Operational Research**, v. 16, p. 215-221, 1984.
- BISCHOFF, E. e DOWSLAND, W., An Application of the Micro to Product Design and Distribution. **Journal of the Operational Research Society**, v. 33, p. 271-280, 1982.
- BISCHOFF, E. e RATCLIFF, M. S., Loading Multiple Pallets. **Journal of the Operational Research Society**, v. 46, p. 1322-1336, 1995.

- BISCHOFF, E. e WAESCHER, G., Special Issue on Cutting and Packing Problems. **European Journal of Operational Research**, v. 84, n. 3, p. 503-712, 1995.
- BROOKE, A.; KENDRICK, D.; MEERAUS, A. e RAMAN, R., **GAMS: A User's Guide**. GAMS Development Corporation, Washington, 1998.
- BUARQUE DE HOLANDA, A., **Novo Dicionário Aurélio**. Segunda Edição, Ed. Nova Fronteira, Rio de Janeiro, 1986.
- CAMERINI, P. M.; FRATTA, L. e MAFFIOLI, F., On Improving Relaxation Methods by Gradient Techniques. **Mathematical Programming Study** v. 3, p. 26-34, 1975.
- CHEN, C. S.; LEE, S. M. e SHEN, Q. S., An Analytical Model for the Container Loading Problem. **European Journal of Operational Research**, v. 80, p. 68-76, 1995.
- CHENG, C. H.; FEIRING, B. R. e CHENG, T. C. E., The Cutting Stock Problem – A Survey. **International Journal of Production Economics**, v. 36, p. 291-305, 1994.
- CHRISTOFIDES, N. e WHITLOCK, C., An Algorithm for Two-Dimensional Cutting Problems. **Operations Research**, v. 25, p. 30-45, 1977.
- CROWDER, H. Computational Improvements for Subgradient Optimization. **Symposio Matematica**, Rome, v. 17, p. 357-372, 1976.
- DOWSLAND, K. A., Determining an Upper Bound for a Class of Rectangular Packing Problems. **Computers and Operations Research**, v. 12, p. 201-205, 1985.
- DOWSLAND, K., An Exact Algorithm for the Pallet Loading Problem. **European Journal of Operational Research**, v. 31, p. 78-84, 1987.
- DOWSLAND, K., Efficient Automated Pallet Loading. **European Journal of Operational Research**, v. 44, p. 232-238, 1990.
- DOWSLAND, K., Some Experiments With Simulated Annealing Techniques for Packing Problems. **European Journal of Operational Research**, v. 68, p. 389-399, 1993.

- DOWSLAND, K. e DOWSLAND, W. B., Packing Problems. **European Journal of Operational Research**, v. 56, p. 2-14, 1992.
- DOWSLAND, W. B., Improving Palletization Efficiency - The Theoretical Basis and Practical Application. **International Journal of Production Research**, v. 33, n. 8, p. 213-222, 1995.
- DYCKHOFF, H., A Typology of Cutting and Packing Problems. **European Journal of Operational Research**, v. 44, p. 145-159, 1990.
- DYCKHOFF, H. e FINKE, U., **Cutting and Packing in Production and Distribution: Typology and Bibliography**, Springer-Verlag Co, Heidelberg, 1992.
- DYCKHOFF, H.; SCHEITHAUER, G. e TERNO, J., **Cutting and Packing**, In Annotated Bibliographies in Combinatorial Optimisation, edited by M. Amico, F. Maffioli e S. Martello (New York, NY:Wiley), p. 393-414, 1997.
- DYER, M. E., Calculating Surrogate Constraints. **Mathematical Programming**, v. 19, p. 255-278, 1980.
- ERDÖS, P. e GRAHAM, R. L., On Packing Squares with Equal Squares. **Journal of Combinatorial Theory, Series A**, v. 19, p. 119-123, 1975.
- FARAGO, R., **Um Método Heurístico Baseado em Relaxação Lagrangiana para Resolver o Problema de Carregamento de Paletes do Produtor**. Dissertação de Mestrado, Departamento de Engenharia de Produção, Universidade Federal de São Carlos, 1999.
- FARAGO, R. e MORABITO, R., Um Método Heurístico Baseado em Relaxação Lagrangiana para o Problema de Carregamento de Paletes do Produtor. **Pesquisa Operacional**, v. 2, n. 20, p. 197-212, 2000.
- G, Y. e KANG, M., A Fast Algorithm for Two-dimensional Pallet Loading Problems of Large Size. **European Journal of Operational Research**, v. 134, p. 193-200, 2001.

- GALVÃO, R. D.; ESPEJO, L. G. A. e BOFFEY, B., A Comparison of Lagrangean and Surrogate Relaxations for the Maximal Covering Location Problem. **European Journal of Operational Research**, v. 124, p. 377-389, 2000.
- GLOVER, F., A Multiphase Dual Algorithm for the Zero-One Integer Programming Problem. **Operations Research**, v. 13, p. 879-919, 1965.
- GLOVER, F., Surrogate Constraints. **Operations Research**, v. 16, n. 4, p. 741-749, 1968.
- GLOVER, F., Surrogate Constraints Duality in Mathematical Programming. **Operations Research**, v. 23, p. 434-451, 1975.
- GREENBERG, H. J. e PIERSKALLA, W. P., Surrogate Mathematical Programming. **Operations Research**, v. 18, p. 924-939, 1970.
- HADJICONSTANTINO, E. e CHRISTOFIDES, N., An Exact Algorithm for General, Orthogonal, Two-Dimensional Knapsack Problems. **European Journal of Operational Research**, v. 83, p. 39-56, 1995.
- HELD M. e KARP, R. M., The Traveling Salesman Problem and Minimum Spanning Trees. **Operations Research**, v. 18, p. 1138-1162, 1970.
- HELD M. e KARP, R. M., The Traveling Salesman Problem and Minimum Spanning Trees: Part II. **Mathematical Programming**, v. 1, p. 6-25, 1971.
- HERBERT, A. e DOWSLAND, K., A Family of Genetic Algorithms for the Pallet Loading Problem. **Annals of Operations Research**, v. 63, p. 415-436, 1996.
- HIFI, M. (eds), Special Issue: Cutting and Packing Problems. **Studia Informatica Universalis**, v. 2, p. 1-161, 2002.
- HODGSON, T., A Combined Approach to the Pallet Loading Problem. **IIE Transactions**, v. 14, n. 3, p. 176-182, 1982.
- HOFFMAN, K. e KAPLINSKY, R., **The Point of Transition – from Machinofacture to Systemofacture**. In: Hoffman, K.; Kaplinsky, R. Driving Force: the Global

Restructuring of Technology, Labour and Investment in the Automobile and Components Industries. London, Westview, 1988.

INTERLOGIS, Disponível em: <www.interlogis.com.br/artigos>. Acesso em: 02/08/2004.

KARVAN, M. H. e RARDIN, R. L., Some Relationship between Lagrangian and Surrogate Duality in Integer Programming. **Mathematical Programming**, v. 17, p. 320-334, 1979.

LETCHEFORD, A. N. e AMARAL, A., Analysis of Upper Bounds for the Pallet Loading Problem. **European Journal Of Operational Research**, v. 3, n. 132, p. 582-593, 2001.

LEUNG, T. W.; YUNG, C. H. e TROUTT, M. D., Applications of Genetic Search and Simulated Annealing to the Two-Dimensional Non-Guillotine Cutting Stock Problem. **Computers & Industrial Engineering**, v. 40, p. 201-214, 2001.

LINS, L.; LINS, S. e MORABITO, R., An N-thet Graph Approach for Non-Guillotine Packings of N-Dimensional Boxes into an N-Container. **European Journal Of Operational Research**, n. 141, p. 421-439, 2002.

LINS, L.; LINS, S. e MORABITO, R., An L-Approach for Packing (l,w) -Rectangles into Rectangular and L-Shaped Pieces. **Journal of the Operational Research Society**, v. 54, p.777-789, 2003.

LODI, A.; MARTELLO, S. e MONACI, M., Two-Dimensional Packing Problems: A Survey. **European Journal of Operational Research**, v. 141, p. 241-252, 2002.

LORENA, L. A. N. e LOPES, F. B., A Surrogate Heuristic for the Set Covering Problems. **European Journal of Operational Research**, v. 79, p. 138-150, 1994.

LORENA, L. A. N. e NARCISO, M. G., Relaxation Heuristics for a Generalized Assignment Problem. **European Journal of Operational Research**, v. 91, p. 600-610, 1996.

- LORENA, L. A. N. e SENNE, E. L. F., A Lagrangean/Surrogate Heuristic for Uncapacitated Facility Location Problems. In: **Anais dos resumos estendidos VIII CLAIO e VXVIII SBPO**, Rio de Janeiro, p. 854-859, 1996.
- LUCENA, A., Relax and Cut Algorithms. Submetido para publicação, 2004.
- MAGGE, J. F., **Logística Industrial: Análise e Administração dos Sistemas de Suprimento e Distribuição**. Tradução: Ana Lúcia Boucinhas. São Paulo: Pioneira, 1977.
- MARTELLO, S. (ed.), Special Issue: Knapsack, Packing and Cutting – Part I: One Dimensional Knapsack Problems. **INFOR**, v. 32, n. 2, 1994a.
- MARTELLO, S. (ed.), Special Issue: Knapsack, Packing and Cutting – Part II: Multidimensional Knapsack and Cutting Stock Problems. **INFOR**, v. 32, n. 4, 1994b.
- MARTELLO, S. e TOTH, P., **Knapsack Problems: Algorithms and Computer Implementations**. Chichester : John Wiley & Sons, 1990.
- MARTINS, G. H. A., **Packing in Two and Three Dimensions**. Tese de Doutorado, Naval Postgraduate School, Monterey, California, 2002.
- MORABITO, R., **Uma Abordagem em Grafo-E/OU para o Problema do Empacotamento: Aplicação ao Carregamento de Paletes e Contêineres**, Tese de Doutorado, Escola de Engenharia de São Carlos, Universidade de São Paulo, 1992.
- MORABITO, R. e ARENALES, M., Um Exame dos Problemas de Corte e Empacotamento. **Pesquisa Operacional**, v. 12, n. 1, p. 1-20, 1992.
- MORABITO, R. e ARENALES, M., An AND/OR-Graph Approach to the Container Loading Problem. **International Transactions in Operational Research**, v. 1, n. 1, p. 59-73, 1994.
- MORABITO, R. e MORALES, S., A Simple and Effective Recursive Procedure for the Manufacturer's Pallet Loading Problem. **Journal of the Operational Research Society**, v. 49, p. 819-828, 1998.

- MORABITO, R. e MORALES, S. R., Errata A Simple and Effective Recursive Procedure for the Manufacturer's Pallet Loading Problem. **Journal of the Operational Research Society** v. 50, 876, 1999.
- MORABITO, R., MORALES, S. R. e WIDMER, J. A., **Loading Optimization of Palletized Products on Trucks**, Transportation Research v. 36, p. 285-296, 2000.
- MORABITO, R. e FARAGO, R., A Tight Lagrangean Relaxation Bound for the Manufacturer's Pallet Loading Problem. **Studia Informatica Universalis**, v. 2, n. 1, p. 57-76, 2002.
- MORALES, S. R., **Otimização no Carregamento de Paletes: Uma Abordagem Heurística para Resolver o Problema do Produtor**, Dissertação de Mestrado, Departamento de Engenharia de Produção, Universidade Federal de São Carlos, 1995.
- MORALES, S. R. e MORABITO, R., Uma Heurística Simples e Eficiente para Resolver o Problema do Carregamento de Palestes do Produtor. **Gestão & Produção**, v. 4, n. 1, p. 234-250, 1997.
- MORALES, S. R.; MORABITO, R. e WIDMER, J. A., Otimização do Carregamento de Produtos Paletizados em Caminhões. **Gestão & Produção**, v. 4, n. 2, p. 234-250, 1997.
- MUKHACHEVA, E. A., **Decision making Under Conditions of Uncertainty: Cutting Packing Problems** (Ufa. Russia: The International Scientific Collection), 1997.
- NARCISO, M. G., **A Relaxação Lagrangiana/Surrogate e algumas Aplicações em Otimização Combinatória**, Tese de Doutorado, Computação Aplicada, Instituto Nacional de Pesquisas Espaciais, 1998.
- NARCISO, M. G. e LORENA, L. A. N., Lagrangean/surrogate Relaxation for Generalized Assignment Problems. **European Journal of Operational Research**, v. 114, n. 1, p. 165-177, 1999.

- NELISSEN, J., **New Approaches to the Pallet Loading Problem**. Working paper, RWTH Aachen [ftp://ftp.informatik.rwth-aachen.de/pub/reports/pallet.ps.Z], 1993.
- NELISSEN, J., **Solving the Pallet Loading Problem More Efficiently**, Working Paper, Graduiertenkolleg Informatik und Technik, Aachen, 1994.
- NELISSEN, J., How to Use Constraints to Compute an Upper Bound for the Pallet Loading Problem. **European Journal of Operational Research**, v. 84, p. 662-680, 1995.
- NEMHAUSER, G. L. e WOLSEY, L. A., **Integer and Combinatorial Optimization**, John Wiley & Sons, Inc, 1988.
- NOVAMAD** **PALLETS**, Disponível em: <www.lpnet.com.br/novamad/palletscom.htm>. Acesso em: 02/08/2004.
- PUREZA, V. e MORABITO, R., Uma Heurística de Busca Tabu Simples para o Problema de Carregamento de Paletes do Produtor. **Pesquisa Operacional**, v. 23, n. 2, p. 259-378, 2003.
- PUREZA, V. e MORABITO R., Some Experiments with a Simple Tabu Search Algorithm for the Manufacturer's Pallet Loading Problem. Aceito para publicação no **Computers & Operations Research**, 2004.
- RAM, B., The Pallet Loading Problem: A Survey. **International Journal of Production Economics**, v. 28, p. 217-225, 1992.
- REEVES, C. R., **Modern Heuristic Techniques for Combinatorial Problem**. Departament of Statistics and Operacional Research, School of Mathematical and Information Sciences, Coventry University, 1993.
- SARIN, S.; KARVAN, M. H. e RARDIN, R. L., A New Surrogate Dual Multiplier Search Procedure. **Naval Research Procedure**, v. 34, p. 431-450, 1987.
- SCHEITHAUER, G. e TERNO, J., The G4-Heuristic for the Pallet Loading Problem. **Journal of the Operational Research Society**, v. 47, p. 511-522, 1996.

- SCHEITHAUER, G. e SOMMERWEISS, G., 4-block Heuristic for the Rectangle Packing Problem. **European Journal of Operational Research**, v. 108, n. 3, p. 509-526, 1988.
- SENNE, E. L. F. e LORENA, L. A. N., **Lagrangean/surrogate Heuristics for p-median Problems**, In Computing Tools for Modeling, Optimization and Simulation: Interfaces in Computer Science and Operations Research, M. Laguna and J. L. Gonzalez-Velarde (eds.), Kluwer Academic Publishers, p. 115-130, 2000.
- SHAPIRO, J. F., Generalized Lagrange Multipliers in Integer Programming. **Operations Research**, v. 19, p. 68-76, 1971.
- SHAPIRO, J. F. **Modeling the Supply Chain** Massachusetts Institute of Technology, 2001.
- SICUP**, Special Interest Group on Cutting and Packing, online library, Disponível em: <www.apdio.pt/sicup/Sicuphomepage/library.html>. Acesso em: 2004.
- SMITH, A. e DE CANI, P., An Algorithm to Optimize the Layout of Boxes in Pallets. **Journal of the Operational Research Society**, v. 31, p. 573-578, 1980.
- SOBRAL, F. H. A., **Técnicas de Unitização de Cargas em Paletes**. Associação Brasileira de Logística (ASLOG) – Comitê de Unitização de Cargas, 1996.
- STEUDEL, H., Generating Pallet Loading Patterns: a Special Case of the Two-Dimensional Cutting Stock Problem. **Management Science**, v. 10, p. 997-1004, 1979.
- SULE, D. R., **Manufacturing Facilities – Location, Planning e Design**. PWS-KENT Publishing Company, 1988.
- SWEENEY, P. e PATERNOSTER, E., Cutting and Packing Problems: A Categorized, Application-oriented Research Bibliography. **Journal of the Operational Research Society**, v. 43, p. 691-706, 1992.

- TARNOWSKI, A. G., TERNO, J. e SCHEITHAUER, G., A Polynomial Time Algorithm for the Guillotine Pallet Loading Problem. **Information Systems and Operational Research**, v. 32, p. 275-287, 1994.
- TSAI, R. D.; MALSTROM, E. M. e MEEKS H. D., A Two-Dimensional Palletizing Procedure for Warehouse Loading Operations. **IEE Transactions**, v. 20, n. 4, p. 418-424, 1988.
- TSAI, R. D.; MALSTROM, E. M. e KUO, W., Three Dimensional Palletization of Mixed Box Sizes. **IEE Transactions**, v. 25, n. 4, p. 64-75, 1993.
- UELZE, R., **Logística Empresarial: uma Introdução à Administração de Transportes**. São Paulo: Pioneira, 1974.
- WANG, P. Y. e WÄSCHER, G., Cutting and Packing (special issue). **European Journal of Operational Research**, vol. 141, 2002.
- YAMASSAKI, C. e PUREZA, V., Um Refinamento do Algoritmo Tabu de Dowsland para o Problema de Carregamento de Paletes do Produtor, **Produção**, v. 13, n. 3, p. 6-17, 2003.

APÊNDICE A

Programa GAMS do modelo (3.1)-(3.4) de BEASLEY (1985b) descrito na seção 3.2 do Capítulo 3.

* constrained non-guillotine bidimensional cutting problem---Beasley (1985b)
* (5,4,3,2) pallet loading example

option OPTCR = 0.00, ITERLIM = 50000;

sets

I0 index of L0 / 0 * 5 /
I(I0) index of horizontal cuts / 0 * 5 /
J(I0) index of vertical cuts / 0 * 5 /
K index of piece type / 1 * 2 /
alias(I0,J0)
alias(I,I1)
alias(J,J1);

* data

scalars

L0 plate length / 5 /

W0 plate width / 4 /;

parameters

L(K) piece length

/ 1 3

2 2 /

W(K) piece width

/ 1 2

2 3 /

C(K) piece value

/ 1 1

2 1 /

X0(I0)

X(I) horizontal cuts

Y(J) vertical cuts

H(I,J,I1,J1,K);

X0(I0) = -1;

loop(I\$((ord(I)-1) * L("1") le L0),

loop(J\$((ord(J)-1) * L("2") le L0),

X0(I0)\$((ord(I0)-1) eq (ord(I)-1)*L("1") + (ord(J)-1)*L("2")) =
ord(I0) - 1));

display X0;

loop(J0, X0(I0)\$X0(I0) eq -1) = X0(I0 + 1);

X0(I0 + 1)\$X0(I0) eq X0(I0 + 1) = -1);

display X0;

X(I) = X0(I)\$((X0(I) ge 0) and (X0(I) le L0 - L("2")));

loop(I, X(I)\$X(I) ne -1) = X(I);

X(I)\$X(I) eq 0) = -1;

X("0")\$X("0") eq -1) = 0);

```

display X;
Y(J) = X0(J)$((X0(J) ge 0) and (X0(J) le W0 - L("2")));
loop(J, Y(J)$ (Y(J) ne -1) = Y(J);
    Y(J)$ (Y(J) eq 0) = -1;
    Y("0")$ (Y("0") eq -1) = 0);
display Y;

H(I,J,I1,J1,K)$((X(I) ne -1) and (Y(J) ne -1) and
    (X(I1) ne -1) and (Y(J1) ne -1)) = 0;
H(I,J,I1,J1,K)$((X(I) ne -1) and (Y(J) ne -1) and
    (X(I1) ne -1) and (Y(J1) ne -1)) = 1$((X(I1) ge X(I) and
    (X(I1) le X(I)+L(K)-1) and (Y(J1) ge Y(J) and
    (Y(J1) le Y(J)+W(K)-1)));

display H;

variables
    A(I,J,K) if piece K is placed at X(I) and Y(J) cuts
    Z      total value of placed pieces;

binary variables A;

equations
    OBJFUN      objective function
    OVERLAP(I1,J1)  overlap constraints at X(I1) and Y(J1) cuts;

*OBJFUN.. Z =e= sum(K, sum(I, sum(J, C(K) * A(I,J,K) )));
OBJFUN.. Z =e= sum(K, sum(I$(X(I) ne -1) and (X(I) le L0-L(K))),
    sum(J$(Y(J) ne -1) and (Y(J) le W0-W(K))),
    C(K) * A(I,J,K) ));
OVERLAP(I1,J1)$((X(I1) ne -1) and (Y(J1) ne -1))..
    sum(K, sum(I$(X(I) ne -1), sum(J$(Y(J) ne -1), H(I,J,I1,J1,K) *
    A(I,J,K) ))) =l= 1;

```

```
model BEASLEY / all /;
```

```
solve BEASLEY maximizing Z using MIP;
```

```
display A.1, X, Y;
```

APÊNDICE B

Programa GAMS do modelo de TSAI *et al.* (1993) descrito na seção 3.3 do Capítulo 3, junto com a solução do exemplo apresentado.

EXEMPLO NUMÉRICO DE TSAI ET AL. (1993)

Como para o modelo (3.1)-(3.4), apresentamos o mesmo exemplo numérico com o objetivo de ilustrar a aplicação do modelo proposto por TSAI *et al.* (1993) para o PCP do produtor, apresentado no Capítulo 3, seção 3.3. Consideremos novamente um palete com dimensões $(L,W) = (5,4)$ e também 4 caixas, ou seja, $k = 1, \dots, 4$. Assim temos que $S = \{b_1, b_2, b_3, b_4\}$, onde:

$$\begin{aligned} b_1 &= (3,2); & b_3 &= (3,2); \\ b_2 &= (2,3); & b_4 &= (2,3). \end{aligned}$$

Consideremos ainda $(X^0, Y^0) = (100, 100)$ as coordenadas do canto inferior esquerdo do palete no plano cartesiano e $M = 500$ um número suficientemente grande.

Assim, o modelo original de TSAI *et al.* (1993) para este exemplo é dado por:

$$\text{Max } P_1 + P_2 + P_3 + P_4$$

sujeito a:

$$x_2 - x_1 \leq -2 + 500 * (u_{12}^1 + u_{12}^2)$$

$$x_1 - x_2 \leq -3 + 500 * [1 - (u_{12}^2 - u_{12}^1)]$$

$$y_2 - y_1 \leq -3 + 500 * [1 - (u_{12}^1 - u_{12}^2)]$$

$$y_1 - y_2 \leq -2 + 500 * [2 - (u_{12}^1 + u_{12}^2)]$$

$$x_3 - x_1 \leq -3 + 500 * (u_{13}^1 + u_{13}^2)$$

$$x_1 - x_3 \leq -3 + 500 * [1 - (u_{13}^2 - u_{13}^1)]$$

$$y_3 - y_1 \leq -2 + 500 * [1 - (u_{13}^1 - u_{13}^2)]$$

$$y_1 - y_3 \leq -2 + 500 * [2 - (u_{13}^1 + u_{13}^2)]$$

$$x_4 - x_1 \leq -2 + 500 * (u_{14}^1 + u_{14}^2)$$

$$x_1 - x_4 \leq -3 + 500 * [1 - (u_{14}^2 - u_{14}^1)]$$

$$y_4 - y_1 \leq -3 + 500 * [1 - (u_{14}^1 - u_{14}^2)]$$

$$y_1 - y_4 \leq -2 + 500 * [2 - (u_{14}^1 + u_{14}^2)]$$

$$x_3 - x_2 \leq -3 + 500 * (u_{23}^1 + u_{23}^2)$$

$$x_2 - x_3 \leq -2 + 500 * [1 - (u_{23}^2 - u_{23}^1)]$$

$$y_3 - y_2 \leq -2 + 500 * [1 - (u_{23}^1 - u_{23}^2)]$$

$$y_2 - y_3 \leq -3 + 500 * [2 - (u_{23}^1 + u_{23}^2)]$$

$$x_4 - x_2 \leq -2 + 500 * (u_{24}^1 + u_{24}^2)$$

$$x_2 - x_4 \leq -2 + 500 * [1 - (u_{24}^2 - u_{24}^1)]$$

$$y_4 - y_2 \leq -3 + 500 * [1 - (u_{24}^1 - u_{24}^2)]$$

$$y_2 - y_4 \leq -3 + 500 * [2 - (u_{24}^1 + u_{24}^2)]$$

$$x_4 - x_3 \leq -2 + 500 * (u_{34}^1 + u_{34}^2)$$

$$x_3 - x_4 \leq -3 + 500 * [1 - (u_{34}^2 - u_{34}^1)]$$

$$y_4 - y_3 \leq -3 + 500 * [1 - (u_{34}^1 - u_{34}^2)]$$

$$y_3 - y_4 \leq -2 + 500 * [2 - (u_{34}^1 + u_{34}^2)]$$

$$x_1 \geq 100 * P_1$$

$$x_2 \geq 100 * P_2$$

$$x_3 \geq 100 * P_3$$

$$x_4 \geq 100 * P_4$$

$$y_1 \geq 100 * P_1$$

$$y_2 \geq 100 * P_2$$

$$y_3 \geq 100 * P_3$$

$$y_4 \geq 100 * P_4$$

$$x_1 \leq (100 + 5) - 3$$

$$x_2 \leq (100 + 5) - 2$$

$$x_3 \leq (100 + 5) - 3$$

$$x_4 \leq (100 + 5) - 2$$

$$y_1 \leq (100 + 4) - 2$$

$$y_2 \leq (100 + 4) - 3$$

$$y_3 \leq (100 + 4) - 2$$

$$y_4 \leq (100 + 4) - 3$$

$$P_k \in \{0, 1\}$$

$$x_k \geq 0, y_k \geq 0$$

$$u_{ij}^1, u_{ij}^2 \in \{0, 1\}$$

$$k = 1, \dots, 4; i = 1, \dots, 3; j = i + 1, \dots, 4$$

Note que este modelo tem 40 restrições, enquanto o modelo de BEASLEY (1985b) tem apenas 7 restrições.

Este problema também foi resolvido utilizando o pacote GAMS/CPLEX. A solução ótima obtida é dada por:

$$(P_1, P_2, P_3, P_4) = (1, 0, 1, 1)$$

$$(x_1, y_1) = (100, 102)$$

$$(x_3, y_3) = (100, 100)$$

$$(x_4, y_4) = (103, 100)$$

$$(u_{12}^1, u_{12}^2) = (0, 0)$$

$$(u_{13}^1, u_{13}^2) = (1, 0)$$

$$(u_{14}^1, u_{14}^2) = (0, 1)$$

$$(u_{23}^1, u_{23}^2) = (0, 1)$$

$$(u_{24}^1, u_{24}^2) = (0, 1)$$

$$(u_{34}^1, u_{34}^2) = (0, 1).$$

Esta solução nos dá uma função objetivo ótima com valor 3. Relembramos que as coordenadas do canto esquerdo do palete é $(X^0, Y^0) = (100, 100)$. Subtraindo (X^0, Y^0) do resultado obtido para colocação de uma caixa, (x_i, y_i) pode ser convertida para as coordenadas originais. Isto assegura,

$$(x_1, y_1) = (0, 2)$$

$$(x_3, y_3) = (0, 0)$$

$$(x_4, y_4) = (3, 0).$$

Esta solução corresponde ao padrão ótimo como mostra a Figura B.1.

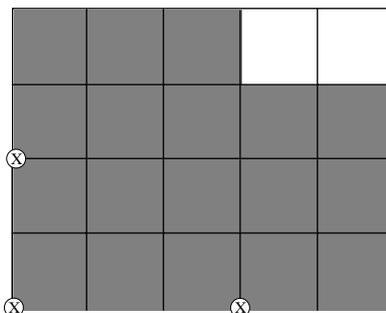


Figura B.1. Padrão de carregamento ótimo para o exemplo utilizando o modelo GAMS/CPLEX de TSAI *et al.* (1993).

A seguir, apresenta-se o modelo GAMS de TSAI *et al.* (1993) resolvido pelo GAMS/CPLEX (versão 7).

MODELO GAMS DE TSAI ET AL. (1983)

* three dimensional palletization of mixed box sizes - Tsai et al. (1993)

* (5,4) pallet loading example

* b1=(3,2)

* b2=(2,3)

* b3=(3,2)

* b4=(2,3)

option OPTCR = 0.0, ITERLIM = 50000;

sets

K index of piece type / 1 * 4 /

alias(I,K)

alias(J,K);

* data

scalars

M / 500 /

X0 / 100 /

Y0 / 100 /

L0 plate length / 5 /

W0 plate width / 4 /;

parameters

L(K) piece length

/ 1 3

2 2

3 3

$4 \quad 2 /$
 $W(K)$ piece width
 $/ 1 \quad 2$
 $2 \quad 3$
 $3 \quad 2$
 $4 \quad 3 /;$

variables

$P(K)$ if piece K is placed is cuts
 $U(I,J,K)$
 $X(K)$
 $Y(K)$
 Z total value of placed pieces;

binary variables P,U ;

equations

$OBJFUN$ objective function
 $R1(I,J)$
 $R2(I,J)$
 $R3(I,J)$
 $R4(I,J)$
 $R5(I,J)$
 $R6(I,J)$
 $R7(K)$
 $R8(K)$
 $R9(K)$
 $R10(K);$

$OBJFUN.. Z =e= \text{sum}(K, P(K));$

$R1(I,J)\$((\text{ord}(I) \neq \text{ord}(J)) \text{ and } (\text{ord}(I) \leq \text{ord}(J))). X(J)-X(I) =l= -L(J) + M*(U(I,J,"1") + U(I,J,"2"));$

R2(I,J)\$((ord(I) ne ord(J)) and (ord(I) le ord(J))). X(I)-X(J) =l= -L(I) + M*(1-(U(I,J,"2") - U(I,J,"1")));

R3(I,J)\$((ord(I) ne ord(J)) and (ord(I) le ord(J))). Y(J)-Y(I) =l= -W(J) + M*(1-(U(I,J,"1") - U(I,J,"2")));

R4(I,J)\$((ord(I) ne ord(J)) and (ord(I) le ord(J))). Y(I)-Y(J) =l= -W(I) + M*(2-(U(I,J,"1") + U(I,J,"2")));

R5(I,J)\$((ord(I) ne ord(J)) and (ord(I) le ord(J))). U(I,J,"1")+U(I,J,"2") =l= 1;

R6(I,J)\$((ord(I) ne ord(J)) and (ord(I) le ord(J))). U(I,J,"1")+U(I,J,"2") =g= 0;

R7(K).. X(K) =g= X0 * P(K);

R8(K).. Y(K) =g= Y0 * P(K);

R9(K).. X(K) =l= (X0 + L0) - L(K);

R10(K).. Y(K) =l= (Y0 + W0) - W(K);

model TSAI / all /;

solve TSAI maximizing Z using MIP;

display X.l, Y.l;

APÊNDICE C

Programa GAMS do modelo de HADJICONSTANTINOU e CHRISTOFIDES (1995) e o modelo CPLEX de AMARAL e LETCHFORD (2001), descritos na seção 3.4 do Capítulo 3, junto com a solução do exemplo apresentado.

EXEMPLO NUMÉRICO DE HADJICONSTANTINOU E CHRISTOFIDES

(1995)

Como para o modelo (3.1)-(3.4), apresentamos o mesmo exemplo numérico com o objetivo de ilustrar a aplicação do modelo proposto por HADJICONSTANTINOU e CHRISTOFIDES (1995) para o PCP do produtor, apresentado no Capítulo 3, seção 3.4. Consideremos um palete com dimensões $(L, W) = (5, 4)$ e caixas $(l, w) = (3, 2)$. As caixas podem ser colocadas em duas posições, ou seja, na horizontal e na vertical, com seus lados paralelos aos lados do palete. Assim, temos $(l_1, w_1) = (3, 2)$ e $(l_2, w_2) = (2, 3)$.

Como dito anteriormente, ao se considerar apenas duas peças (isto é, $m = 2$) com $(l_1, w_1) = (l, w)$, $(l_2, w_2) = (w, l)$ e $v_1 = v_2 = 1$, temos o caso especial do PCP do produtor. Assim, temos que $i = 1, 2$. Admitimos ainda, $Q_1 = Q_2 = 3$.

A formulação original para o modelo é dada por:

$$\text{Max } x_{110} + x_{111} + x_{112} + x_{120} + x_{121} + x_{122} + x_{130} + x_{131} + x_{132} + x_{210} + x_{211} + x_{212} + x_{213} + x_{220} + x_{221} + x_{222} + x_{223} + x_{230} + x_{231} + x_{232} + x_{233}$$

sujeito a:

$$z_{00} + z_{01} + z_{10} + z_{11} + z_{20} + z_{21} \leq 6*(2 - x_{110} - y_{110})$$

$$z_{00} + z_{01} + z_{10} + z_{11} + z_{20} + z_{21} \leq 6*(2 - x_{120} - y_{120})$$

$$z_{00} + z_{01} + z_{10} + z_{11} + z_{20} + z_{21} \leq 6*(2 - x_{130} - y_{130})$$

⋮

$$x_{110} + x_{111} + x_{112} \leq 1$$

$$x_{120} + x_{121} + x_{122} \leq 1$$

$$x_{130} + x_{131} + x_{132} \leq 1$$

⋮

$$x_{110} + x_{111} + x_{112} = y_{110} + y_{111} + y_{112}$$

$$x_{120} + x_{121} + x_{122} = y_{120} + y_{121} + y_{122}$$

$$x_{130} + x_{131} + x_{132} = y_{130} + y_{131} + y_{132}$$

⋮

$$2*(x_{110} + x_{120} + x_{130}) + 3*(x_{210} + x_{220} + x_{230}) + z_{00} + z_{01} + z_{02} + z_{03} = 4$$

$$2*(x_{110} + x_{111} + x_{120} + x_{121} + x_{130} + x_{131}) +$$

$$3*(x_{210} + x_{211} + x_{220} + x_{221} + x_{230} + x_{231}) + z_{10} + z_{11} + z_{12} + z_{13} = 4$$

$$2*(x_{110} + x_{111} + x_{112} + x_{120} + x_{121} + x_{122} + x_{130} + x_{131} + x_{132}) +$$

$$3*(x_{211} + x_{212} + x_{221} + x_{222} + x_{231} + x_{232}) + z_{20} + z_{21} + z_{22} + z_{23} = 4$$

⋮

$$3*(y_{110} + y_{120} + y_{130}) + 2*(y_{210} + y_{220} + y_{230}) + z_{00} + z_{10} + z_{20} + z_{30} + z_{40} = 5$$

$$3*(y_{110} + y_{111} + y_{120} + y_{121} + y_{130} + y_{131}) +$$

$$2*(y_{210} + y_{211} + y_{220} + y_{221} + y_{230} + y_{231}) + z_{01} + z_{11} + z_{21} + z_{31} + z_{41} = 5$$

$$3*(y_{111} + y_{112} + y_{121} + y_{122} + y_{131} + y_{132}) +$$

$$2*(y_{210} + y_{211} + y_{220} + y_{221} + y_{230} + y_{231}) + z_{02} + z_{12} + z_{22} + z_{32} + z_{42} = 5$$

⋮

O modelo para o nosso exemplo contém 72 restrições, enquanto os modelos de BEASLEY (1985b) e TSAI *et al.* (1993) continham 7 e 40 restrições, respectivamente. Redefinindo $p \in X$ e $q \in Y$ (conjuntos normais) o modelo ficaria com

57 restrições e 50 variáveis. Por esse motivo, apresentamos somente as 3 primeiras restrições de cada conjunto de restrições do modelo (3.24)-(3.32). A solução ótima obtida é $x_{110} = x_{120} = x_{223} = 1$ e $y_{110} = y_{122} = y_{221} = 1$ com o valor da função objetivo igual a 3. A Figura C.1 ilustra a solução ótima encontrada pelo pacote GAMS/CPLEX .

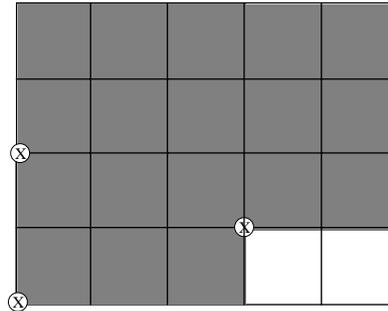


Figura C.1. Padrão de carregamento ótimo para o exemplo utilizando o modelo GAMS/ CPLEX de HADJICONSTANTINOU e CHRISTOFIDES (1995).

A correção da formulação de HADJICONSTANTINOU e CHRISTOFIDES (1995) segundo AMARAL e LETCHFORD (2001) para o nosso exemplo é dada por:

$$\begin{aligned} \text{Max } & x_{110} + x_{111} + x_{112} + x_{120} + x_{121} + x_{122} + x_{130} + x_{131} + x_{132} + x_{210} + x_{211} + \\ & x_{212} + x_{213} + x_{220} + x_{221} + x_{222} + x_{223} + x_{230} + x_{231} + x_{232} + x_{233} \end{aligned}$$

sujeito a:

$$x_{110} + x_{111} + x_{112} \leq 1$$

$$x_{120} + x_{121} + x_{122} \leq 1$$

$$x_{130} + x_{131} + x_{132} \leq 1$$

⋮

$$x_{110} + x_{111} + x_{112} = y_{110} + y_{111} + y_{112}$$

$$x_{120} + x_{121} + x_{122} = y_{120} + y_{121} + y_{122}$$

$$x_{130} + x_{131} + x_{132} = y_{130} + y_{131} + y_{132}$$

⋮

$$x_{120} + y_{120} + x_{110} + y_{110} \leq 3$$

$$x_{130} + y_{130} + x_{110} + y_{110} \leq 3$$

$$x_{110} + y_{110} + x_{120} + y_{120} \leq 3$$

$$\vdots$$

O modelo para o nosso exemplo contém 4872 restrições (o modelo original de HADJICONSTANTINOU e CHRISTOFIDES (1995) contém 72 restrições). Assim, como para o modelo anterior, apresentamos somente as 3 primeiras restrições de cada uma das restrições do modelo (3.33)-(3.38). A solução ótima obtida é $x_{112} = x_{132} = x_{220} = 1$ e $y_{110} = y_{132} = y_{220} = 1$ com o valor da função objetivo igual a 3. A Figura C.2 ilustra a solução ótima encontrada utilizando o *software* CPLEX .

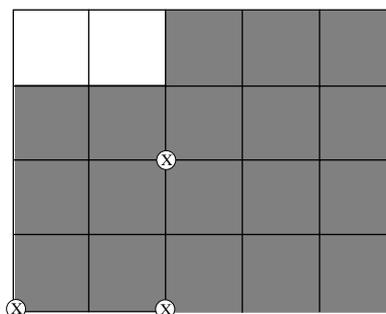


Figura C.2. Padrão de carregamento ótimo para o exemplo utilizando o *software* CPLEX no modelo de AMARAL e LETCHFORD (2001).

A seguir, apresentamos o modelo GAMS de HADJICONSTANTINOU e CHRISTOFIDES (1995) resolvido pelo GAMS/CPLEX e o modelo CPLEX de AMARAL e LETCHFORD (2001).

MODELO GAMS DE HADJICONSTANTINOU E CHRISTOFIDES

(1995)

* non-guillotine bidimensional cutting problem --- Hadjiconstantinou e Christofides (1995)

* (5,4,3,2) pallet loading example

option OPTCR = 0.0, ITERLIM = 50000;

sets

P0 index of L0 / 0 * 5 /
 P(P0) index of horizontal cuts / 0 * 5 /
 Q(P0) index of vertical cuts / 0 * 3 /
 I index of piece type / 1 * 2 /
 J max number of boxes / 1 * 3 /
 alias(P0,Q0)
 alias(P,R)
 alias(Q,S);

* data

scalars

L0 plate length / 5 /
 W0 plate width / 4 /;

parameters

L(I) piece length
 / 1 3
 2 2 /
 W(I) piece width
 / 1 2
 2 3 /
 V(I) piece value
 / 1 1
 2 1 /
 X0(P0)
 Y0(P0)
 X(P) horizontal cuts
 Y(Q) vertical cuts;

* conjuntos discretizados

X0(P0) = -1;

```

loop(P$((ord(P)-1) le L0),
  loop(Q$((ord(Q)-1) le L0),
    X0(P0)$((ord(P0)-1) eq (ord(P)-1) + (ord(Q)-1)) = ord(P0)-1));
display X0;
loop(Q0, X0(P0)$X0(P0) eq -1) = X0(P0 + 1);
  X0(P0 + 1)$X0(P0) eq X0(P0 + 1)) = -1 );
display X0;

```

```

X(P) = X0(P)$((X0(P) ge 0) and (X0(P) le L0-L("2")));
loop(P, X(P)$X(P) ne -1) = X(P);
  X(P)$X(P) eq 0 ) = -1;
  X("0")$X("0") eq -1) = 0);
display X;

```

```

Y(Q) = X0(Q)$((X0(Q) ge 0) and (X0(Q) le W0 - L("2")));
loop(Q, Y(Q)$Y(Q) ne -1) = Y(Q);
  Y(Q)$Y(Q) eq 0 ) = -1;
  Y("0")$Y("0") eq -1) = 0);
display Y;

```

variables

```

A(I,J,P)  if piece K is placed is cuts at x-position
B(I,J,Q)  if piece K is placed is cuts at y-position
C(R,S)    if point (r and s) in (L0 and W0)=Ao
Z         total value of placed pieces;

```

binary variables A,B,C;

equations

```

OBJFUN    objective function
R1(I,J,P,Q)
R2(J,I)

```

R3(J,I)

R4(R)

R5(S);

OBJFUN.. Z =e= sum (I, sum (J,
sum (P\$(X(P) ne -1) and (X(P) ge 0) and (X(P) le
L0-L(I)), V(I)*A(I,J,P))));

* /primeira restricao do modelo/

R1(I,J,P,Q)\$((X(P) ne -1) and (X(P) le L0-L(I)) and (Y(Q) ne -1) and
(Y(Q) le W0-W(I)).. sum (S\$((ord(S) ge ord(Q)) and (ord(S)
le (ord(Q)+ W(I)-1))), sum (R\$((ord(R)
ge ord(P)) and (ord(R) le (ord(P)+
L(I)-1))), C(R,S))) =l= (2-A(I,J,P)-
B(I,J,Q))*L(I)*W(I);

* /segunda restricao do modelo/

R2(J,I).. sum (P\$(X(P) ne -1) and (X(P) ge 0) and (X(P) le L0-L(I))),
A(I,J,P)) =l= 1;

* /terceira restricao do modelo/

R3(J,I).. sum (P\$(X(P) ne -1) and (X(P) ge 0) and (X(P) le L0-L(I))),
A(I,J,P)) =e= sum (Q\$((Y(Q) ne -1) and (Y(Q) ge 0) and
(Y(Q) le W0-W(I))),B(I,J,Q));

* /quarta restricao do modelo/

R4(R)\$((ord(R) le L0).. sum (I, sum (J, sum (P \$(X(P) ne -1) and (X(P) le
L0-L(I)) and (ord(P) ge
(max(0,ord(R)-L(I)+1))) and (ord(P) le
(min(ord(R),(L0+1)-
L(I))))),W(I)*A(I,J,P))))) + sum (S
\$((ord(S) ge 0) and (ord(S) le W0)),

$$C(R,S) = e = W_0;$$

* /quinta restricao do modelo/

```

R5(S)$((ord(S) le W0).. sum (I, sum (J, sum (Q $((Y(Q) ne -1) and (Y(Q) le
      W0-W(I)) and (ord(Q) ge (max(0,ord(S)-
      W(I)+1))) and (ord(Q) le
      (min(ord(S),(W0+1)-W(I))))),
      L(I)*B(I,J,Q)))) +
      sum (R $((ord(R) ge 0) and (ord(R) le
      L0)), C(R,S)) =e= L0;

```

model HADJICONSTANTINO / all /;

solve HADJICONSTANTINO maximizing Z using MIP;

display A.I,B.I,C.I,X,Y;

MODELO CPLEX DE AMARAL E LETCHFORD (2001)

```

/*****
/* Este programa imprime o modelo (modificado) de */
/* Hadjiconstantinou e Christofides (1995) */
/* para o problema de corte bidimensional nao guilotinado(Após */
/* gerado o modelo poderá ser resolvido usando CPLEX */
/*****
/***** PROBLEMA LINEAR INTEIRO MISTO *****/

```

```
#include <stdio.h>
```

```
#include <limits.h>
```

```
#include <math.h>
```

```
#include <ctype.h>
```

```
#include <string.h>
```

```
/*#include <conio.h>*/
#include <stdlib.h>
#include <malloc.h>

int M,Qi,Qk;
double L[10],W[10],v[10];

void Dados_iniciais()
{
M=2;
Qi=3;
Qk=3;
L[0]=5;
W[0]=4;
L[1]=3;
W[1]=2;
L[2]=2;
W[2]=3;
v[1]=1;
v[2]=1;
}

int main(void)
{
FILE *fp;
int cont,cont1;
int i, j, p, q, k, l, r, s,a,b;

cont=1;

Dados_iniciais();
```

```

fp = fopen ("problema.lp", "w");

if ((fp = fopen ("problema.lp", "w")) == NULL)
{
printf ("Erro na abertura do arquivo para gravar dados !!!\n");
exit (0);
}
/* Definição da função objetivo */
/* Maximizar (somatorio(i)somatorio(j)somatorio(p)vi.Xijp) para
i=1,...,M,j=1,...,Qi e 0<=p<=alfa0-alfai*/
fprintf(fp, "\nMAX\n");
for (i=1; i<=M; i++)
{
for (j=1; j<=Qi; j++)
for (p=0; p<=L[0]-L[i]; p++)
{
if ((i==1) && (j==1) && (p==0))
fprintf(fp," %5.2lf X%d%d%d ",v[i],i,j,p);
else
{
if (cont >=4 )
{
fprintf(fp," + %5.2lf X%d%d%d\n ",v[i],i,j,p);
cont=0;
}
else
{
fprintf(fp," + %5.2lf X%d%d%d ",v[i],i,j,p);
cont++;
}
}
}
}
}

```

```

}

fprintf(fp, "\n");
fprintf(fp, "\n");

/* Definição do conjunto de restrições */
fprintf(fp,"SUBJECT TO\n");

/* definição das restrições de impõem que cada peça é cortada uma vez
de A0

$$\sum_{j=1}^{Q_i} X_{ij} \leq 1 \quad 0 \leq p \leq L_0 - L_i \quad */$$


cont1=1;
for (i=1; i<=M; i++)
for (j=1; j<=Qi; j++)
{
for (p=0; p<=L[0]-L[i]; p++)
{
if (p==0)
fprintf(fp,"R%d: X%d%d%d ",cont1,i,j,p);
else
{
fprintf(fp," + X%d%d%d ",i,j,p);
}
}
fprintf(fp," <= 1.0\n");
cont1++;
}

fprintf(fp, "\n");

/* definição das restrições de impõem que cada peça é cortada uma vez

```

```

de A0
    somatorio(0<=p<=L0-Li)Xijp<=1          */
for (i=1; i<=M; i++)
    for (j=1; j<=Qi; j++)
    {
        for (p=0; p<=L[0]-L[i]; p++)
        {
            if (p==0)
                fprintf(fp,"R%d: X%d%d%d",cont1,i,j,p);
            else
                fprintf(fp," + X%d%d%d",i,j,p);
        }
        for (q=0; q<=W[0]-W[i]; q++)
        {
            if (q==0)
                fprintf(fp," - Y%d%d%d",i,j,q);
            else
                fprintf(fp," - Y%d%d%d",i,j,q);
        }
        fprintf(fp," = 0.0\n");
        cont1++;
    }

fprintf(fp,"\n");

/* Restrições de Amaral & Letchford para evitar a sobreposicao das
peças
Xijp + Yijq + Xklr + Ykls <= 3          */

for (i=1;i<=M;i++)
    for (p=0; p<=L[0]-L[i]; p++)
        for (q=0; q<=W[0]-W[i]; q++)

```

```

for (k=1;k<=M;k++)
  for (r=0; r<=L[0]-L[k]; r++)
    for (s=0; s<=W[0]-W[k]; s++)
      for (a=r; a<r+L[k]; a++)
        for (b=s; b<s+W[k];b++)
          if (a>=p && a<p+L[i] && b>=q && b<q+W[i])
            {
              for (l=1;l<=Qk;l++)
                for (j=1;j<=Qi;j++)
                  if ((i!=k) || (j!=l))
                    {
                      fprintf(fp,"R%d: X%d%d%d",cont1,i,j,p);
                      fprintf(fp," + Y%d%d%d",i,j,q);
                      fprintf(fp," + X%d%d%d",k,l,r);
                      fprintf(fp," + Y%d%d%d",k,l,s);
                      fprintf(fp," <=3\n");
                      cont1++;
                    }
            }

fprintf(fp, "\n");
/* VARIAVEIS INTEIRAS */
/* Xijp=0 ou 1, i=1,...,M e j=1,..Q, p=0,...,L0-Li */
/* Yijq=0 ou 1, i=1,...,M e j=1,..Q, q=0,...,W0-Wi */
/* Xklr=0 ou 1, k=1,...,M e l=1,..Q, r=0,...,L0-Li */
/* Ykls=0 ou 1, k=1,...,M e l=1,..Q, s=0,...,W0-Wi */

fprintf(fp,"BOUNDS\n");
for (i=1;i<=M;i++)
  {
    for (j=1; j<=Qi; j++)
      for (p=0; p<=W[i]; p++)

```

```

        fprintf(fp,"0.0 =< X%d%d%d <= 1.0\n", i,j,p);
    }

for (i=1;i<=M;i++)
    {
    for (j=1; j<=Qi; j++)
        for (q=0; q<=L[i]; q++)
            fprintf(fp,"0.0 =< Y%d%d%d <= 1.0\n", i,j,q);
    }

fprintf(fp,"INTEGERS\n");
for (i=1;i<=M;i++)
    {
    for (j=1; j<=Qi; j++)
        for (p=0; p<=W[i]; p++)
            fprintf(fp," X%d%d%d \n", i,j,p);
    }
for (i=1;i<=M;i++)
    {
    for (j=1; j<=Qi; j++)
        for (q=0; q<=L[i]; q++)
            fprintf(fp," Y%d%d%d \n", i,j,q);
    }

fprintf(fp,"\n");
fprintf(fp,"END\n");

fclose(fp);

system("cplex<plex.bat");

}/*fim da rotina*/

```