

# UM ALGORITMO GENÉTICO PARA SOLUÇÃO DO PROBLEMA DE CARREGAMENTO DE CONTAINER

Lúcio Lopes Rodrigues Neto

TESE SUBMETIDA AO CORPO DOCENTE DA COORDENAÇÃO DOS  
PROGRAMAS DE PÓS-GRADUAÇÃO DE ENGENHARIA DA UNIVERSIDADE  
FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS  
PARA A OBTENÇÃO DO GRAU DE MESTRE EM ENGENHARIA ELÉTRICA

Aprovado por:

---

Prof. Amit Bhaya, Ph. D.

---

Prof. Eugenius Kaszkurewicz, D. Sc.

---

Prof. Roberto Diéguez Galvão, Ph. D.

---

Prof<sup>a</sup>. Marley Maria Bernardes Rebuszi Vellasco, Ph. D.

RIO DE JANEIRO, RJ – BRASIL

ABRIL DE 2005

RODRIGUES NETO, LÚCIO LOPES

Um Algoritmo Genético para a Solução  
do Problema de Carregamento de  
Container [Rio de Janeiro] 2005

XII, 105 p. 29,7 cm (COPPE/UFRJ,  
M.Sc., Engenharia Elétrica, 2005)

Tese – Universidade Federal do Rio de  
Janeiro, COPPE

1. Otimização com Algoritmo Genético
2. Carregamento de container

I. COPPE/UFRJ II. Título (série)

Dedico este trabalho aos meus pais, Lúcio e Fátima, pelos sacrifícios e dedicação durante toda a vida, que me proporcionaram uma educação e formação humana correta, a minha irmã, Renata, pela amizade e amor, a minha avó Cléia, por não deixar de acreditar em nenhum momento, e aos amigos, essenciais em todas as horas.

## AGRADECIMENTOS

Gostaria de agradecer a todos que de alguma forma contribuíram para que esse trabalho pudesse ser realizado, contudo, gostaria de fazer agradecimentos especiais aos que seguem abaixo:

Em primeiro lugar, agradeço a Deus, fonte da vida e de inspiração, que me deu saúde, coragem e inteligência.

Ao meu orientador Amit Bhaya Ph.D., ao meu co-orientador Eugenius Kaszkurewicz, D. Sc., pelo apoio, confiança, orientação e incentivo, essenciais para a realização deste trabalho.

À Universidade Federal do Amazonas e à Universidade Federal do Rio de Janeiro que tiveram a iniciativa de realizar o mestrado em automação industrial voltado para o público de Manaus. E a SUFRAMA (Superintendência da Zona Franca de Manaus) que tornou possível a realização do mesmo, através dos recursos alocados. Que outras iniciativas como essa possam ser pensadas e realizadas.

Aos professores do mestrado, pela dedicação, que nos proporcionaram a aquisição de novos conhecimentos e desenvolvimento profissional e intelectual, em especial, aos coordenadores do mestrado, Prof. Ramon e Prof<sup>a</sup>. Marly Guimarães.

Aos colegas de mestrado, que incentivavam uns aos outros, compartilhavam conhecimento e contribuíram para que eu pudesse cursar todo o programa.

Aos colegas de trabalho, em diversos níveis, pelo apoio e incentivo. Obrigado!

A todos os amigos, em especial, Alexandre e Alex Sandra Leocádio, Eduardo e Fabíola Nakamura, Adriana Mourão, Auxiliadora Barbosa, Pe. Zózimo e Glaura Dias, que se fizeram presentes em momentos importantes nesse período.

Aos meus pais, educadores, companheiros e amigos de sempre. À minha irmã, Renata, que com sua doçura e amor sempre me incentivava.

E a todos os amigos e familiares não citados aqui.

Resumo da tese apresentada à COPPE/ UFRJ como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências (M. Sc.)

UM ALGORITMO GENÉTICO PARA SOLUÇÃO DO PROBLEMA DE  
CARREGAMENTO DE CONTAINER

Lúcio Lopes Rodrigues Neto

Abril / 2005

Orientadores: Amit Bhaya

Eugenius Kaszkurewicz

Programa: Engenharia Elétrica.

Este trabalho aborda o uso de técnicas metaheurísticas em problemas de otimização, especialmente de carregamento de *containers*. Desenvolve um algoritmo genético para a solução do problema de carregamento de *container*, utilizando algumas características reais e adaptadas do processo de carregamento e dos dados das cargas de uma empresa de produto eletrônico. O algoritmo desenvolvido possui dois modelos de carregamento (preenchimento do *container*). São realizados testes com os dois modelos de carregamento com diferentes configurações de cargas disponíveis, que colaboram para as conclusões apresentadas no trabalho.

Abstract of Thesis presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Master of Science (M.Sc.)

## A GENETIC ALGORITHM FOR SOLVING CONTAINER LOADING PROBLEM

Lúcio Lopes Rodrigues Neto

April / 2005

Advisors:     Amit Bhaya  
                  Eugenius Kaszkurewicz

Department: Electrical Engineering.

This work presents the use of metaheuristic techniques for solving optimization problems, specifically in the container loading problem. It develops a genetic algorithm to solve the container loading problem, using some real features of a container loading problem as well as real cargo data from an electronics factory. The algorithm that is developed uses two kinds of loading models. Tests are performed with various available cargo data, confirming conclusions that are presented in this work.

## ÍNDICE

1.	INTRODUÇÃO	1
1.1	Caracterização do Problema	1
1.2	Objetivos	4
1.3	Estrutura da Tese	5
2.	ABORDAGENS CLÁSSICAS E METAHEURÍSTICAS PARA O PROBLEMA DE CARREGAMENTO	6
2.1	Revisão bibliográfica das formulações utilizando programação matemática	7
2.2	Formulação matemática do problema de carregamento	9
2.3	Soluções Metaheurísticas	14
2.4	Os algoritmos de GEHRING, BORTFELDT (2001) e HE, CHA (2002)	20
3.	O ALGORITMO GENÉTICO PROPOSTO	26
3.1	Objetivos do Algoritmo Genético Desenvolvido	27
3.2	Estrutura dos Dados	29
3.3	População Inicial de Cromossomos	30
3.4	Avaliando pela Função de Aptidão Total	31
3.4.1	Sub-função de Aptidão do Volume (R)	32
3.4.2	Sub-função de Aptidão do Peso (W)	32
3.4.3	Sub-função de Aptidão do Centro de Gravidade (G)	33
3.4.4	Sub-função de Aptidão do Valor (V)	34
3.5	Operadores do Algoritmo Genético	35
3.5.1	Cruzamento	35
3.5.2	Mutação	37

3.6	Decodificação para obtenção do padrão de carregamento	40
3.6.1	Seleção da Primeira Caixa	43
3.6.2	Decomposição L, M e R	45
3.6.3	Decomposição M, L e R	61
4.	TESTES E RESULTADOS	74
4.1	Testes com dimensões de caixas utilizadas por HE, CHA (2002)	75
4.1.1	Testes do Algoritmo Genético com o Preenchimento L, M e R	77
4.1.2	Testes do Algoritmo Genético com o Preenchimento M, L e R	80
4.1.3	Comparação dos Resultados entre os Dois Modelos de Carregamento	82
4.2	Testes com dimensões de caixas reais do problema estudado	84
4.2.1	Testes com 100 caixas de Áudio	85
4.2.2	Testes com caixas mistas	90
5.	CONCLUSÃO	98
5.1	Conclusão	98
5.2	Sugestão para Trabalhos Subseqüentes	99
6.	REFERÊNCIAS BIBLIOGRÁFICAS	102



## ÍNDICE DE FIGURAS

Figura 1 - Ilustração do posicionamento da caixa, de acordo com suas dimensões e das dimensões do container. _____	9
Figura 2 - Métodos de Resolução de Problemas de Otimização (RODRIGUES, 2000). _____	15
Figura 3 - Ilustração do posicionamento da caixa em cima da caixa de base e do espaço vazio da torre. _____	22
Figura 4 - Ilustração do posicionamento da primeira caixa e decomposição do espaço disponível em três subespaços: lateral (L), superior (M) e frontal (R). _____	25
Figura 5 - Fluxograma do processo de determinação da solução para o problema de carregamento de container com o algoritmo genético desenvolvido. _____	28
Figura 6 - Exemplo da operação de cruzamento para obtenção de dois novos cromossomos a partir de um par de cromossomos. _____	37
Figura 7 - Exemplo da operação de mutação que resulta em novo cromossomo pela troca de posição de genes de um cromossomo original. _____	39
Figura 8 - Fluxograma Macro do Preenchimento de Container L, M e R _____	42
Figura 9 - Fluxograma Macro do Preenchimento de Container M, L e R _____	42
Figura 10 - Fluxograma da Seleção da Primeira Caixa _____	45
Figura 11 - Ilustração do posicionamento da primeira caixa e formação dos subespaços da decomposição L, M e R. _____	46
Figura 12 - Ilustração do posicionamento de nova caixa no subespaço R, e formação de novos subespaços L e M. _____	57
Figura 13 - Processo de seleção de caixas e preenchimento de subespaços L e M, da seqüência L, M e R. _____	59
Figura 14 - Processo de seleção de caixas e preenchimento de subespaços R, da seqüência L, M e R. _____	60

Figura 15 - Ilustração do posicionamento da primeira caixa e formação dos subespaços M, L e R, da decomposição M, L e R. _____	62
Figura 16 - Processo de seleção de caixas e preenchimento dos subespaços M e L, na seqüência M, L e R. _____	67
Figura 17 - Processo de seleção de caixas e preenchimento do subespaço R, na seqüência M, L e R. _____	68
Figura 18 - Gráfico demonstrativo da evolução do algoritmo genético _____	79
Figura 19 - Ilustração das decomposições de espaço L, M e R e M, L e R. _____	83
Figura 20 - Demonstrativo do preenchimento médio mensal do espaço de containers _____	89
Figura 21 - Gráfico demonstrativo da interação dos fatores A e B e da relação entre os fatores (parâmetros) e a saída (função de aptidão total) para o carregamento M, L e R com 285 caixas. _____	95

## ÍNDICE DE TABELAS

Tabela 1 - Descrição das denominações de elementos do algoritmo genético e do problema de carregamento. _____	30
Tabela 2 - Dimensões das caixas (1 a 40) de HE, CHA (2002). _____	76
Tabela 3 - Dimensões das caixas (41 a 50) de HE, CHA (2002). _____	77
Tabela 4 - Testes do GA com preenchimento L, M e R, com caixas utilizadas na tabela 2 e 3. _____	77
Tabela 5 - Média e desvio padrão da função de aptidão total e das sub-funções de aptidão do preenchimento L, M e R com as caixas da tabela 2 e 3. _____	78
Tabela 6 - Demonstrativo do melhor resultado obtido no preenchimento L, M e R com as caixas da tabela 2 e 3. _____	78
Tabela 7 - Testes do GA com preenchimento M, L e R, com caixas utilizadas na tabela 2 e 3. _____	80
Tabela 8 - Média e desvio padrão da função de aptidão total e das sub-funções de aptidão do preenchimento M, L e R com as caixas da tabela 2 e 3. _____	81
Tabela 9 - Demonstrativo do melhor resultado obtido no preenchimento M, L e R com as caixas da tabela 2 e 3. _____	81
Tabela 10 - Dados e Resultado do teste estatístico de diferenças de média entre os preenchimentos L,M e R e M, L e R para o teste com 50 caixas. _____	82
Tabela 11 - Dados dimensionais, pesos e valores das caixas do teste com 100 caixas de áudio. _____	85
Tabela 12 - Resultados dos testes com caixas de áudio com o preenchimento L, M e R. _____	86
Tabela 13 - Média e desvio padrão dos valores dos testes da tabela 12 (preenchimento L, M e R). _____	86

Tabela 14 - Demonstrativo do melhor resultado obtido no preenchimento L, M e R nos testes realizados conforme a tabela 12. _____	86
Tabela 15 - Resultados dos testes com caixas de áudio com o preenchimento M, L e R. _____	87
Tabela 16 - Média e desvio padrão dos valores dos testes da tabela 15 (preenchimento M, L e R). _____	87
Tabela 17 - Demonstrativo do melhor resultado obtido no preenchimento M, L e R nos testes realizados conforme a tabela 15. _____	88
Tabela 18 - Dados dimensionais, pesos e valores das caixas do teste com 100 caixas de Áudio, TV e DVD. _____	90
Tabela 19 - Resultado dos testes com combinações dos parâmetros, “número de gerações” e “probabilidade de mutação”, no carregamento L, M e R. _____	92
Tabela 20 - Resultado dos testes com os parâmetros número de gerações e probabilidade de mutação ajustados em seus níveis máximos, para carregamento L, M e R. _____	93
Tabela 21 - Média e desvio padrão para a função de aptidão total e para as sub-funções de aptidão do teste com carga mista para o carregamento L, M e R. _____	94
Tabela 22 - Resultado dos testes com combinações dos parâmetros “número de gerações” e “probabilidade de mutação”, no carregamento M, L e R. _____	94
Tabela 23 - Média e desvio padrão para a função de aptidão total e para as sub-funções de aptidão do teste com carga mista para o carregamento M, L e R. _____	96
Tabela 24 - Dados e Resultado do teste estatístico de diferenças de média entre os preenchimentos L,M e R e M, L e R para o teste com 285 caixas. _____	97

# 1. INTRODUÇÃO

## 1.1 Caracterização do Problema

O carregamento de *containers* é um problema clássico em pesquisas de otimização, cujo objetivo, em geral, é aumentar o volume utilizado em relação ao volume total disponível. Em outras palavras, deseja-se encontrar uma configuração de arrumação de cargas de tal maneira que o volume ocioso do *container* seja o mais próximo de zero.

Quando existem mais cargas a serem carregadas do que volume disponível nos *containers*, acrescenta-se ao problema a necessidade de reduzir o número de *containers* para dispor todas as cargas.

A melhor utilização do volume disponibilizado pelo *container* e a menor quantidade de *containers* utilizados reduz despesas com o transporte de mercadorias, o que deve refletir no preço dos produtos e nos resultados financeiros das companhias.

Contudo a solução para esse problema não é trivial. Várias restrições com relação ao empilhamento de cargas, pesos, restrições dimensionais, centro de gravidade, quanto a orientação do posicionamento das cargas, valores monetários e outros, dependendo da característica da carga são fatores que dificultam a

determinação de uma solução ótima. A solução do problema não pode ser determinada analiticamente, e, em termos computacionais, é pouco provável que exista um algoritmo de baixa complexidade, com base em uma abordagem matemática e determinística.

Esse tipo de problema requer, portanto, o uso de técnicas de otimização chamadas metaheurísticas ou apenas heurísticas, para resolução aproximada do problema, considerando as restrições que o mesmo apresenta, levando à determinação de uma solução ótima ou sub-ótima que possa ser utilizada.

Para a solução do problema do carregamento de *containers* existem várias abordagens e técnicas de otimização que, ao longo do tempo, vêm sendo pesquisadas e implementadas. A evolução das pesquisas e as restrições peculiares a cada tipo de mercadoria e de operação de carregamento e transporte de mercadoria, como o tipo de produto, seu formato geométrico e a quantidade de carga, levam a adequações diferenciadas dessas técnicas e ao desenvolvimento de heurísticas adaptativas a cada problema específico.

Neste contexto, o estudo e o emprego de algoritmos genéticos em problemas práticos de otimização, como o de carregamento de *container*, é cada vez maior. Esses algoritmos se inspiram em um modelo matemático da evolução genética das espécies para a busca de um conjunto de boas soluções para o problema estudado. Eles iteram uma população de soluções possíveis, que evolui para um conjunto de soluções melhores. Isso, por si só, destaca o algoritmo genético entre outros que iteram um único ponto do espaço de soluções (MICHALEWICZ, 1994).

Neste trabalho, faz-se o estudo do problema de carregamento de *container* em uma empresa fabricante de produtos eletroeletrônicos, que necessita fazer parte do seu transporte de mercadoria via sistema rodoviário.

A empresa apresenta certas peculiaridades em sua operação que devem ser considerados no caso da busca de uma solução para o seu problema real: não existência de estoques de mercadorias, produtos enviados diretamente da linha de

produção para o carregamento das carretas (*containers*), a existência de restrições peculiares ao armazenamento das cargas nas carretas, conforme os produtos que serão carregados na mesma, televisores, dvds e aparelhos de áudio, e um valor limite do total de cargas carregadas por *container*. O carregamento ainda é caracterizado por ser feito de forma paralela, ou seja, várias carretas são preenchidas com produtos ao mesmo tempo.

Contudo, para a pesquisa apresentada, estuda-se um problema modificado, mais simples, fazendo-se considerações com relação às características da operação da empresa e de seus produtos.

Essencialmente, o estudo aqui apresentado considera processo de carregamento com estoque de mercadorias, onde a quantidade de produtos disponíveis de cada modelo para cada carregamento é determinada pelo plano de produção diário da empresa e onde não se tem restrições de disponibilidades de produtos por conta de ciclos de produção e outras características do processo produtivo. Outra característica desconsiderada nesse estudo é o processo de carregamento paralelo das carretas, o estudo é realizado simulando uma situação de carregamento de uma carreta por vez.

A configuração de carga a ser determinada deveria atender a quatro objetivos:

maximização do volume utilizado;

maximização do peso da carga;

maximização do equilíbrio (centro de gravidade) e

maximização do valor monetário associado à carga.

Considerando os objetivos acima, o problema, em questão, deveria ser tratado como um problema de otimização multi-objetivo (DEB, 2001, ESPEJO, GALVÃO, 2004). Contudo, um problema dessa natureza, com quatro objetivos, tornaria a busca por sua solução demasiadamente complexa, uma vez que a determinação numérica da fronteira eficiente (fronteira Pareto-ótima) é difícil. Além, disso, a interpretação e

visualização desta fronteira é uma tarefa não trivial que talvez não fosse facilitar o processo decisório.

Para tanto, optou-se por tratar o problema como um problema de um único objetivo, para o qual, a avaliação das características do padrão de carregamento é feita por uma única função objetivo, também chamada de aptidão total (*fitness function*), que deve ser maximizada. Esta, por sua vez, é construída como combinação convexa dos quatro objetivos descritos acima. Estes quatro objetivos são denominados sub-funções de aptidão de: volume, peso, centro de gravidade e valor. Os pesos da combinação convexa são apontados conforme as prioridades definidas pela empresa. Certamente, cada combinação define um novo problema com nova solução. Como, no caso do problema tratado, havia uma escolha pré-estabelecida dos pesos, investigou-se esta escolha. Entretanto, deve ser enfatizado que o algoritmo proposto funcionaria com qualquer combinação de pesos, podendo, portanto, ser utilizado através de uma discretização apropriada do espaço de pesos na construção da fronteira eficiente.

## 1.2 Objetivos

Este trabalho visa fazer um estudo sobre a aplicação de algoritmos genéticos como técnica de otimização em um problema de carregamento de carretas real, adotando algumas hipóteses simplificadoras, porém utilizando dados reais de dimensões das caixas e carretas, restrições quanto a empilhamento, valor de carregamento, peso e equilíbrio.

O algoritmo genético deverá apresentar dois tipos de formação de carga, sempre através da formação de camadas de caixas: a primeira, através de um preenchimento que começa horizontalmente e segue-se no espaço vertical disponível, vindo do fundo do *container* para parte frontal do mesmo; a segunda, através de



formação de torres de caixas, onde as mesmas preencherão o espaço disponível da esquerda para a direita do *container* e de trás para frente.

### 1.3 Estrutura da Tese

Esta tese está estruturada na seguinte forma:

- Introdução, com motivações que levaram ao problema e objetivos da pesquisa, e com a descrição do processo atual de carregamento de *container* na empresa, considerando todas as características e restrições do problema real e do problema simplificado para os efeitos deste trabalho;
- Apresentação da formulação matemática e das técnicas metaheurísticas utilizadas para soluções de problemas complexos;
- Descrição do algoritmo genético desenvolvido, seus operadores, e parâmetros associados;
- Apresentação de testes com diversas configurações de cargas disponíveis;
- Conclusões.

## 2. ABORDAGENS CLÁSSICAS E METAHEURÍSTICAS PARA O PROBLEMA DE CARREGAMENTO

Este capítulo apresenta uma formulação matemática para o problema de otimização a ser abordado neste trabalho, bem como uma revisão bibliográfica. O objetivo é mostrar a complexidade da formulação em termos de programação matemática para justificar a consideração de metaheurísticas como o algoritmo genético a ser proposto neste trabalho.

Inicia-se o capítulo com uma breve revisão bibliográfica, através da abordagem clássica do problema de otimização de carregamento; segue-se com a apresentação da formulação matemática do problema; uma abordagem sobre soluções metaheurísticas e, por fim, apresenta-se dois trabalhos com propostas de soluções para o problema de carregamento de *container*, utilizando algoritmo genético, e que inspiraram a solução proposta por este trabalho, que unifica as duas propostas mencionadas.

## 2.1 Revisão bibliográfica das formulações utilizando programação matemática

Na literatura técnica, existem duas versões do problema do carregamento do *container* que recebem, respectivamente, as denominações *bin packing* e *knapsack problem* (MICHALEWICZ, 1994, RAIDL, 1999). O *bin packing* é um problema de determinação de configuração de carregamento cujo objetivo é determinar o menor número de *containers* que possa armazenar todos os itens do carregamento. O *knapsack problem* é um problema de otimização onde itens são associados a valores e pesos (penalizações), e busca-se determinar uma coleção de itens tal que o valor total dos itens seja maximizado atendendo a uma restrição de peso máximo para a coleção, previamente definido (MICHALEWICZ, 1994). Trazendo para o problema de carregamento de *container*, o *knapsack problem* tem por objetivo encontrar uma configuração de carregamento para um único *container* tal que o volume utilizado do *container* seja maximizado, considerando as restrições dimensionais do *container*.

O problema combinando *knapsack* e *bin packing* é denominado *Multiple Container Packing Problem* (MCP) ou Problema de Empacotamento em Múltiplos *Containers* (RAIDL, 1999), onde se tem o objetivo múltiplo de maximizar o espaço utilizado e, ao mesmo tempo, minimizar o número de *containers*. O *knapsack* pode ser encarado como um problema MCP com apenas um *container*, onde o objetivo passa a ser unicamente de maximizar o espaço utilizado.

Em certos casos o objetivo não é o de maximizar o espaço utilizado ou minimizar o espaço vazio, mas sim maximizar o peso do carregamento ou, ainda, maximizar o valor da carga no *container*, quando são atribuídos valores aos itens de carregamento (RAIDL, 1999).

Outras variantes do problema existem em diversos estudos. Ainda que o problema de carregamento de *container* esteja normalmente associado a

carregamento de itens tridimensionais, existem vários estudos para problemas de utilização de espaço disponível em uma dimensão (CSRIK, JOHSON, *et al.*, 1999) e duas dimensões (HOPPER, TURTON, 1997). Estudos de problemas bidimensionais são comuns, especialmente em problemas de definições de *lay-out*, armazenamento de *pallets* e outros, tanto com objetos de formas regulares quanto irregulares.

Contudo, os problemas envolvendo itens tridimensionais são muito estudados, especialmente pelas inúmeras aplicações reais, carregamento de caminhões, de navios, aviões, preenchimento de estoques e armazéns, arrumação de caixas em *pallets* e outros, na sua maioria com caixas retangulares, porém de tamanhos diferentes (CHIEN, WU, 1998, HOPPER, TURTON, 1997). Objetos cilíndricos e de outros formatos regulares e irregulares também são estudados.

Problemas tridimensionais com caixas retangulares são amplamente estudados. Em geral, esses problemas visam maximizar o volume utilizado dos *containers*, buscando, ao mesmo tempo, atender às restrições das dimensões dos *containers*. É muito comum em problemas reais a inclusão de restrições com relação ao peso e ao equilíbrio da carga (HE, CHA, 2002, GEHRING, BORTFELDT, 2001).

Alguns estudos propõem modelos de programação matemática para o problema do carregamento de *containers*. Por exemplo, MARQUES e ARENALES (2002) propõem um modelo de otimização não linear, cujo *container*, que é denominada mochila, é dividido em compartimentos e onde se busca a maior utilização possível desses compartimentos e o menor número dos mesmos a serem agregados à mochila. Para tanto é incluso um fator de penalização a cada compartimento adicionado. LI, TSAI, *et al.* (2003) também desenvolveram uma formulação de programação matemática para o problema, caracterizada pela mistura de variáveis contínuas e binárias, no intuito de se encontrar uma solução ótima que seria um *container* com volume mínimo disponibilizado para o carregamento de todas as caixas. FAINA (2000), por sua vez, desenvolve um modelo geométrico, onde se

define zonas de espaços, cujos preenchimentos e escolha das caixas se dá utilizando o método de *simulated annealing*, que será resumidamente abordado neste capítulo.

CHEN, LEE, *et al.* (1995) desenvolveram um modelo analítico para o problema de carregamento de *container*, utilizando variáveis inteiras e binárias, onde são consideradas caixas retangulares e de tamanhos não uniformes e é previsto um carregamento em diferentes *containers* de dimensões possivelmente diferentes entre si. No seu modelo matemático, o objetivo é encontrar uma solução que minimize o espaço desperdiçado do *container* e minimize o número de *containers* necessários para carregar todas as caixas.

## 2.2 Formulação matemática do problema de carregamento

Abaixo, segue-se o modelo matemático de CHEN, LEE, *et al.* (1995), considerando o carregamento em apenas um *container* e fazendo-se adaptações ao problema aqui estudado. Ainda que no desenvolvimento matemático original sejam permitidas rotações das caixas em todos os eixos, aqui, a dimensão altura da caixa terá orientação fixa (como na aplicação real pretendida) e sempre estará paralela ao eixo Z, onde se encontra a altura do *container*. A largura será a dimensão horizontal de maior valor e o comprimento, a dimensão horizontal de menor valor. A figura 1, abaixo, auxilia no entendimento da descrição acima.

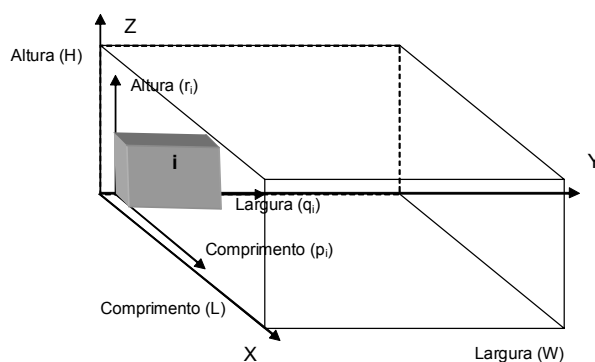


Figura 1 - Ilustração do posicionamento da caixa, de acordo com suas dimensões e das dimensões do *container*.

Supõe-se que as dimensões das caixas e do *container* são conhecidas, bem como a quantidade de caixas para o carregamento. As caixas são independentes e são posicionadas ortogonalmente no *container*, ou seja, são posicionadas em paralelo ou perpendicular aos eixos. As caixas podem sofrer rotações de  $90^\circ$  no posicionamento, de tal forma que a largura e o comprimento da caixa não sejam posicionados obrigatoriamente em paralelo à largura e ao comprimento do *container*, respectivamente, contudo, a altura da caixa sempre será posicionada em paralelo ao eixo da altura do *container*.

As notações listadas abaixo são necessárias para o entendimento do modelo matemático:

$N$  – número de caixas disponíveis para carregamento;

$s_i$  – variável binária, que indica se a caixa foi posicionada no *container*. Quando isso ocorre,  $s_i = 1$ , caso contrário,  $s_i = 0$ ;

$(L, W, H)$  – triplo que indica comprimento, largura e altura do *container*, respectivamente;

$(p_i, q_i, r_i)$  – triplo que indica comprimento, largura e altura da caixa, respectivamente;

$(x_i, y_i, z_i)$  – triplo que indica a localização da caixa pelo canto inferior esquerdo traseiro;

$(l_{xi}, l_{yi}, l_{zi})$  – triplo binário que indica para qual eixo o comprimento da caixa está em paralelo. Como a altura da caixa sempre está em paralelo com a altura do *container*, podemos trabalhar com o triplo  $(l_{xi}, l_{yi}, 0)$ ;

$(w_{xi}, w_{yi}, w_{zi})$  – variável binária que indica para qual eixo a largura da caixa está em paralelo. Como a altura da caixa sempre está em paralelo com a altura do *container*, essa variável pode ser modificada para  $(w_{xi}, w_{yi}, 0)$ ;

$(h_{xi}, h_{yi}, h_{zi})$  – triplo binário que indica para qual eixo a altura da caixa está em paralelo. Como a altura da caixa sempre estará em paralelo com a altura do *container*, esse triplo será fixo:  $(0, 0, 1)$ ;

Ainda existem outras variáveis que são usadas para indicar o posicionamento das caixas em relação a outras caixas:

$a_{ik}$  – caso seja 1, indica que a caixa  $i$  está à esquerda da caixa  $k$ ;

$b_{ik}$  – caso seja 1, indica que a caixa  $i$  está à direita da caixa  $k$ ;

$c_{ik}$  – caso seja 1, indica que a caixa  $i$  está atrás da caixa  $k$ ;

$d_{ik}$  – caso seja 1, indica que a caixa  $i$  está à frente da caixa  $k$ ;

$e_{ik}$  – caso seja 1, indica que a caixa  $i$  está abaixo da caixa  $k$ ;

$f_{ik}$  – caso seja 1, indica que a caixa  $i$  está acima da caixa  $k$ ;

O problema é, então, formulado, conforme abaixo, onde se observa que o objetivo é minimizar o espaço não preenchido do *container*:

$$\text{Min } L \cdot W \cdot H - \sum_{i=1}^N p_i \cdot q_i \cdot r_i \cdot s_i$$

Sujeito a

(Evitar sobreposição de caixas no *container*)

$$x_i + p_i \cdot l_{xi} + q_i \cdot w_{xi} + r_i \cdot h_{xi} \leq x_k + (1 - c_{ik}) \cdot M, \forall i, k, i < k \quad (1)$$

$$x_k + p_k \cdot l_{xk} + q_k \cdot w_{xk} + r_k \cdot h_{xk} \leq x_i + (1 - d_{ik}) \cdot M, \forall i, k, i < k \quad (2)$$

$$y_i + p_i \cdot l_{yi} + q_i \cdot w_{yi} + r_i \cdot h_{yi} \leq y_k + (1 - a_{ik}) \cdot M, \forall i, k, i < k \quad (3)$$

$$y_k + p_k \cdot l_{yk} + q_k \cdot w_{yk} + r_k \cdot h_{yk} \leq y_i + (1 - b_{ik}) \cdot M, \forall i, k, i < k \quad (4)$$

$$z_i + p_i \cdot l_{zi} + q_i \cdot w_{zi} + r_i \cdot h_{zi} \leq z_k + (1 - e_{ik}) \cdot M, \forall i, k, i < k \quad (5)$$

$$z_k + p_k \cdot l_{zk} + q_k \cdot w_{zk} + r_k \cdot h_{zk} \leq z_i + (1 - f_{ik}) \cdot M, \forall i, k, i < k \quad (6)$$

(Garantir que o par de caixas avaliados nas equações 1 a 6 está no *container*)

$$a_{ik} + b_{ik} + c_{ik} + d_{ik} + e_{ik} + f_{ik} \geq s_i + s_k - 1 \quad \forall i, k, i < k \quad (7)$$

(Garantir que o posicionamento das caixas obedeça às limitações físicas dimensionais do *container*)

$$x_i + p_i \cdot l_{xi} + q_i \cdot w_{xi} + r_i \cdot h_{xi} \leq L + (1 - s_i) \cdot M, \forall i \quad (8)$$

$$y_i + p_i \cdot l_{yi} + q_i \cdot w_{yi} + r_i \cdot h_{yi} \leq W + (1 - s_i) \cdot M, \forall i \quad (9)$$

$$z_i + p_i \cdot l_{zi} + q_i \cdot w_{zi} + r_i \cdot h_{zi} \leq H + (1 - s_i) \cdot M, \quad \forall i \quad (10)$$

M é um número inteiro arbitrário e grande.

Os triplos binários que identificam o posicionamento das caixas com relação aos eixos são variáveis dependentes e se relacionam conforme abaixo:

$$l_{xi} + l_{yi} + l_{zi} = 1 \quad (11)$$

$$w_{xi} + w_{yi} + w_{zi} = 1 \quad (12)$$

$$z_{xi} + z_{yi} + z_{zi} = 1 \quad (13)$$

$$l_{xi} + w_{xi} + z_{xi} = 1 \quad (14)$$

$$l_{yi} + w_{yi} + z_{yi} = 1 \quad (15)$$

$$l_{zi} + w_{zi} + z_{zi} = 1 \quad (16)$$

Como já mencionado, o triplo  $(h_{xi}, h_{yi}, h_{zi}) = (0, 0, 1)$ , implica que  $h_{xi} = h_{yi} = 0$ , portanto, as equações (1) a (10) podem ser substituídas, e a formulação fica conforme abaixo:

$$\text{Min} \quad L \cdot W \cdot H - \sum_{i=1}^N p_i \cdot q_i \cdot r_i \cdot s_i$$

Sujeito a

(Evitar sobreposição de caixas no *container*)

$$x_i + p_i \cdot l_{xi} + q_i \cdot w_{xi} \leq x_k + (1 - c_{ik}) \cdot M, \quad \forall i, k, i < k \quad (17)$$

$$x_k + p_k \cdot l_{xk} + q_k \cdot w_{xk} \leq x_i + (1 - d_{ik}) \cdot M, \quad \forall i, k, i < k \quad (18)$$

$$y_i + p_i \cdot l_{yi} + q_i \cdot w_{yi} \leq y_k + (1 - a_{ik}) \cdot M, \quad \forall i, k, i < k \quad (19)$$

$$y_k + p_k \cdot l_{yk} + q_k \cdot w_{yk} \leq y_i + (1 - b_{ik}) \cdot M, \quad \forall i, k, i < k \quad (20)$$

$$z_i + p_i \cdot l_{zi} + q_i \cdot w_{zi} + r_i \cdot h_{zi} \leq z_k + (1 - e_{ik}) \cdot M, \quad \forall i, k, i < k \quad (21)$$

$$z_k + p_k \cdot l_{zk} + q_k \cdot w_{zk} + r_k \cdot h_{zk} \leq z_i + (1 - b_{ik}) \cdot M, \quad \forall i, k, i < k \quad (22)$$



(Garantir que o par de caixas avaliados nas equações 17 a 22 está no *container*)

$$a_{ik} + b_{ik} + c_{ik} + d_{ik} + e_{ik} + f_{ik} \geq s_i + s_k - 1 \quad \forall i, k, i < k \quad (23)$$

(Garantir que o posicionamento das caixas obedeça às limitações físicas dimensionais do *container*)

$$x_i + p_i.l_{xi} + q_i.w_{xi} \leq L + (1 - s_i).M, \forall i \quad (24)$$

$$y_i + p_i.l_{yi} + q_i.w_{yi} \leq W + (1 - s_i).M, \forall i \quad (25)$$

$$z_i + p_i.l_{zi} + q_i.w_{zi} + r_i.h_{zi} \leq H + (1 - s_i).M, \forall i \quad (26)$$

Ainda que o problema tenha sido simplificado com a eliminação de algumas variáveis, continua sendo um problema complexo de solução não trivial, sendo classificado como um problema de classe *NP-Complete* (NP completo) (*Non-deterministic Polynomial-time Complete*), já que tanto o *knapsack problem* quanto o *bin packing problem* que formam o problema de carregamento múltiplo de *container* são NP-completos (RAIDL, 1999). Isso significa que o problema não apresenta uma solução polinomial determinística em tempo polinomial (RODRIGUES, 2000, LITVINENKO *et al.*, 2002).

CHEN, LEE, *et al.* (1995) afirmam que ainda que a formulação matemática leve a uma solução ótima para o problema de carregamento de *container* utilizando um modelo de programação linear inteiro e binário mista, esta não se apresenta como uma solução eficiente para um problema de larga escala, com um número elevado de caixas, já que os números de variáveis e de restrições tornam-se muito grandes, onde o crescimento dos mesmos é dado por  $2n^2$  (FAINA, 2000), onde  $n$  é o número de caixas.

A inclusão de novas restrições, tais como peso de carga, valores e outros, aumentaria ainda mais a complexidade do problema, o que, por sua vez, aumentaria a dificuldade de se encontrar uma solução.

Considerando a complexidade da formulação apresentada acima e o tempo para que se encontre a solução com esta formulação, técnicas de otimização metaheurísticas, como algoritmos genéticos, apresentam-se como métodos mais eficientes, simples e possíveis de serem aplicados em problemas complexos de carregamento de *container* (HE, CHA, 2002, MICHALEWICZ, 1994, SOUZA, 2002).

### 2.3 Soluções Metaheurísticas

Grande parte dos problemas práticos de otimização combinatória é classificada como problemas NP-Completo ou problemas NP-Difícil. Esses problemas não podem ser resolvidos por algoritmos simples em tempo polinomial (SOUZA, 2002).

Técnicas clássicas de otimização não conseguem determinar solução, em tempo hábil, de problemas complexos, como o problema de carregamento de *container*. Mesmo o desenvolvimento de heurísticas simples para a procura de soluções não tem sido suficiente para a determinação de soluções ótimas ou sub-ótimas para problemas NP-difíceis. Métodos metaheurísticos têm sido cada vez mais utilizados nesse tipo de problema.

Metaheurísticas de otimização apresentam como características principais um componente probabilístico, quando comparado a técnicas clássicas, e propõem uma heurística geral que independe da natureza do problema específico a ser tratado, o que motiva o uso do prefixo meta no termo metaheurística. De uma forma geral, essas técnicas conseguem evitar ótimos locais, por realizarem a procura do ótimo em diversas regiões do espaço, utilizando uma aleatoriedade “direcionada” pela heurística codificada na função de aptidão.

Algumas das técnicas classificadas como metaheurística desenvolvidas e utilizadas nos últimos anos são: Algoritmos Genéticos, Busca Tabu, *Simulated Annealing*, *Grasp*, VNS e outros. As técnicas clássicas de método de descida, método

de descida randômico e outros, que são baseados em noção de vizinhança, ainda são utilizados para alguns problemas de otimização menos complexos que o problema descrito acima.

RODRIGUES (2000) resume os métodos de resolução de problemas de otimização em dois grupos: métodos exatos e métodos aproximados (heurísticas, metaheurísticas e especiais), conforme figura 2.

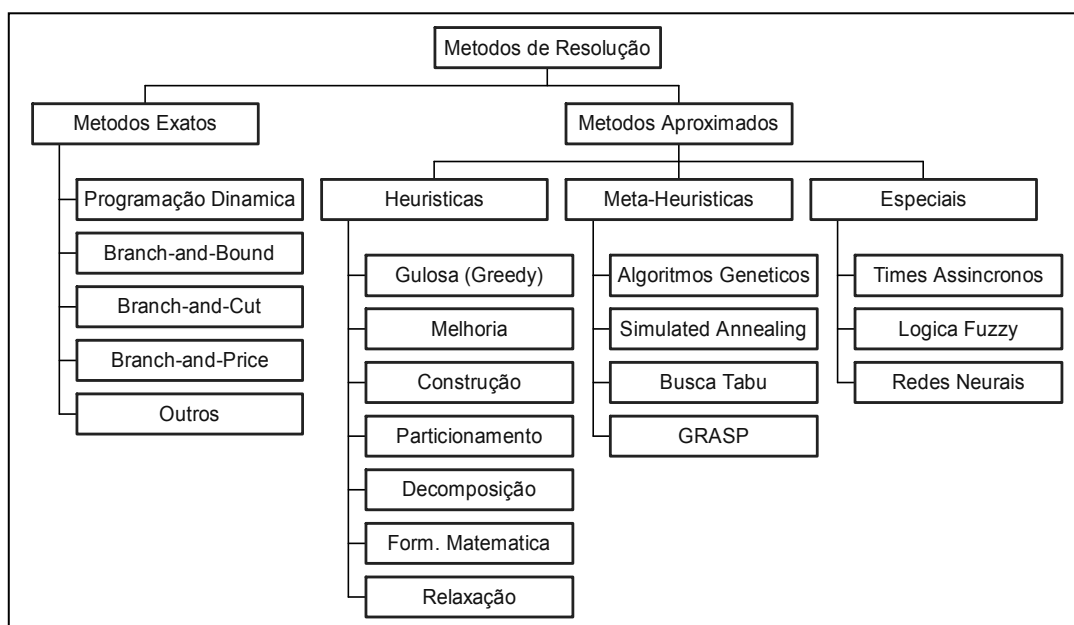


Figura 2 - Métodos de Resolução de Problemas de Otimização (RODRIGUES, 2000).

Os principais métodos de metaheurísticas citados acima podem ser resumidos conforme abaixo:

- *Simulated Annealing* é uma técnica de busca local probabilística, que se fundamenta em uma analogia com termodinâmica, simulando resfriamento de um conjunto de átomos aquecidos. A técnica utiliza-se de uma solução inicial qualquer para prover a busca pela solução final. Um *loop* gera aleatoriamente, em cada direção, um único vizinho da solução corrente. O método possui um comportamento similar ao método de

descida (KIRKPATRICK, GELLAT, *et al.*, 1983, SOUZA, 2002, BAILER-JONES, 2002).

- Busca Tabu é um método de descida que utiliza-se de um procedimento adaptativo que utiliza uma estrutura de memória como guia, que, por sua vez, armazena os últimos movimentos realizados, e é chamada de lista tabu. Essa lista funciona como uma fila, onde um novo movimento é adicionado e o mais antigo retirado e permite que se decida pela continuidade da exploração do espaço de soluções mesmo na ausência de movimentos de melhora, evitando que haja retorno a um ótimo local previamente visitado (GLOVER, 1989, GLOVER, 1990, SOUZA, 2002).
- Algoritmos Genéticos, por sua vez, utilizam a analogia com processos naturais de evolução, onde as características de uma possível solução são medidas por uma função de aptidão, determinando assim que os indivíduos mais aptos têm maiores chances de sobreviver e reproduzir. Para reproduzir ou criar uma nova geração, novamente em analogia com a biologia, operações de cruzamento, mutação e seleção são utilizadas. Quando se atinge uma população onde todos apresentam um determinado nível de aptidão, o método termina e o melhor indivíduo (cromossomo) que apresenta características mais adequadas à solução esperada (maximização ou minimização) é selecionado como solução (MICHALEWICZ, 1994, SOUZA, 2002, BAILER-JONES, 2002).
- GRASP: *Greedy Randomized Adaptive Search Procedure* ou Procedimento de Busca Adaptativa Gulosa e Aleatória é um método iterativo de duas etapas. A primeira etapa consiste na construção de soluções, elemento a elemento, e numa segunda etapa faz-se busca local para determinar um ótimo local na vizinhança de uma solução construída. O processo de seleção de soluções é baseado em uma

função adaptativa gulosa. Em cada interação são geradas diferentes soluções (SOUZA, 2002).

Das técnicas metaheurísticas apresentadas acima, Algoritmos Genéticos estão entre as mais utilizadas, pela sua flexibilidade, facilidade de programação, possibilidade de busca de solução em um espaço grande de soluções e pelos estudos e aplicações já realizadas que demonstram sua eficácia (HE, CHA, 2002).

Baseado na evolução natural da espécie, teoria oriunda de Darwin, a idéia por trás dos algoritmos genéticos é o de busca de soluções sempre melhores que o subconjunto de soluções atuais. Cromossomos são utilizados como indivíduos de uma população de soluções, onde os mesmos representam uma possível solução para o problema. Cada elemento do cromossomo, chamado de gene, é um componente da solução, ou seja, no problema do Caixeiro Viajante, o gene pode representar uma localidade, já no *knapsack* ou *bin packing problem* deverá representar uma caixa, cartão, cilindro, ou seja, o elemento que será carregado. A posição dos genes em cada cromossomo é chamada de *locus*.

Os cromossomos com relação às características apresentadas são avaliados através de uma função objetivo ou função de aptidão. A função aptidão é usada para medir numericamente o quanto o indivíduo apresenta-se como uma boa solução para o problema, ou seja, se o problema for de minimização, quanto menor o resultado, melhor é o cromossomo para a população; se o problema for de maximização, quanto maior o resultado numérico, melhor é a solução.

No caso dos problemas clássicos de carregamento de *container*, as funções de aptidão são determinadas de forma a minimizar o espaço não ocupado ou maximizar o espaço disponível. Para o *bin packing problem*, a função de aptidão é construída de forma que o objetivo seja minimizar o número de *containers* necessários para o carregamento.

O operador de cruzamento, como o nome sugere, faz o cruzamento de dois cromossomos para gerar outros dois cromossomos. Os cromossomos (filhos) gerados

carregam consigo partes dos cromossomos originais (pais). O cruzamento se dá de várias formas, através da escolha aleatória de genes de dois cromossomos pais, através de uma escolha do comprimento ou número de genes de um determinado pai e de outro, não necessariamente com o mesmo número, que permitirá escolhas aleatórias ou não, seqüenciais ou não, dos genes de cada um dos pais que determinarão o novo indivíduo. Na literatura é possível encontrar diversos tipos de cruzamento, como: *single-point*, *multi-point*, em linha, intermediária e aleatória (MICHALEWICZ, 1994).

Exemplificando um operador de cruzamento *single-point*, têm-se dois cromossomos representados por genes binários (não necessariamente os genes devem ser binários, podem ser números reais ou inteiros):  $C_1 = [1\ 0\ 0\ 1\ 0\ 0\ 0\ 1]$  e  $C_2 = [1\ 1\ 0\ 0\ 1\ 1\ 0\ 0]$ , dois novos cromossomos são formados com partes dos dois cromossomos:  $C_A = [1\ 0\ 0\ 0\ 1\ 1\ 0\ 0]$  e  $C_B = [1\ 1\ 0\ 1\ 0\ 0\ 0\ 1]$ , onde  $C_A$  possui, como seus três primeiros genes, os três primeiros genes de  $C_1$  e do quarto ao sétimo gene são respectivamente os genes 4 a 7 dos genes de  $C_2$ ; já  $C_B$ , por sua vez, possui, como seus três primeiros genes, os três primeiros genes de  $C_2$  e do quarto ao sétimo gene são genes oriundos da mesma posição de  $C_1$ .

A mutação, por sua vez, é uma operação onde os genes mudam de posição, normalmente, de forma aleatória. Dependendo como o operador for construído, um número variável de genes pode sofrer a troca de posição e suas trocas podem ser, também, aleatórias. Observa-se abaixo um cromossomo de genes inteiros, onde dois genes sofrem mutação.

$$C_1 = [4\ 5\ 2\ 8\ 7\ 1\ 9\ 3\ 6] \rightarrow \text{mutação} \rightarrow C_A = [4\ 9\ 2\ 8\ 7\ 1\ 5\ 3\ 6]$$

Repara-se que o segundo e o sétimo gene trocam de posição. A operação de mutação não precisa ser somente de troca de posições, ela também é utilizada para mudança de aleatória de um ou mais genes. No exemplo abaixo é possível perceber que um único gene modifica-se, sem que haja uma troca de *locus* com outro gene.

$$C_2 = [1\ 0\ 0\ 0\ 0\ 1\ 1\ 1] \rightarrow \text{mutação} \rightarrow C_B = [1\ 0\ 0\ 1\ 0\ 1\ 1\ 1]$$

Em uma população de cromossomos pode-se limitar ou determinar aleatoriamente o número de cromossomos que sofrerão cruzamento e ou mutação, associando-os a um fator de probabilidade, que pode ser fixo ou adaptativo.

Operadores de seleção combinados a operações de cruzamento e mutação também são utilizados nos algoritmos genéticos para determinação de uma nova população de cromossomos. Um dos mais clássicos é o da roleta (*Roulette Wheel*), que utiliza um mecanismo probabilístico de seleção baseado em medições de desempenho. Seleções aleatórias ou ainda um processo de seleção cujos melhores cromossomos, com relação as suas características, são selecionados, também, são utilizados (MICHALEWICZ, 1994).

O número de populações geradas é, em geral, limitado para assegurar o término do algoritmo genético em tempo razoável. Contudo, outros critérios como, por exemplo, um número máximo de gerações sem apresentar melhora na função objetivo, poderiam ser utilizados como regras de parada do algoritmo genético.

Outro item chave em um algoritmo genético é a geração da população inicial, já que é a partir da mesma que as avaliações e a formação de novas gerações serão feitas. A população inicial pode ser gerada de forma aleatória, através de uma heurística própria ao problema em questão, considerando suas características, ou ainda ser fixa, a partir de uma solução previamente conhecida e, relativamente, satisfatória.

O mecanismo descrito acima garante uma busca multi-direcional em um determinado espaço, mantendo-se uma população de soluções potenciais de aptidão elevada, simulando uma evolução e garantindo, se não uma solução ótima, uma solução sub-ótima.

Em seguida, são apresentados dois trabalhos que inspiraram a proposta de solução desta pesquisa, a ser apresentada no capítulo seguinte.

## 2.4 Os algoritmos de GEHRING, BORTFELDT (2001) e HE, CHA (2002)

GEHRING, BORTFELDT (2001) apresentam um problema de carregamento de *container* onde a carga é formada por caixas regulares retangulares que devem ser posicionadas em paralelo às paredes do *container*. No seu trabalho, é proposta uma solução combinando uma heurística para a formação de torres disjuntivas de caixas, ou seja, torres independentes onde nenhuma caixa pode pertencer a mais de uma torre, com um algoritmo genético cujo objetivo é otimizar a disposição das torres no chão do *container*.

No problema apresentado por GEHRING e BORTFELDT (2001), assume-se que o centro de gravidade das caixas é sempre no centro geométrico das mesmas. As caixas são sempre posicionadas completamente no chão no *container* ou acima de uma outra caixa. O objetivo é determinar uma série de caixas para o carregamento que maximize o valor total das caixas no *container* e ao mesmo tempo atenda às restrições do problema. Entende-se por valor: volume ou taxa de frete.

O *container* é posicionado sobre o eixo cartesiano, onde o canto inferior esquerdo é chamado de origem do sistema. A profundidade ou comprimento do *container* é orientado na direção do eixo  $x$ , a largura na direção do eixo  $y$  e a altura na direção do eixo  $z$ .

As restrições do problema são apresentadas abaixo:

1. Restrições de orientação: as caixas podem ter dimensões (uma ou duas) que não possam ser posicionadas na orientação vertical.
2. Restrições de posicionamento: nenhuma caixa mais pesada pode ser posicionada sobre uma caixa mais leve.
3. Restrições de peso: o peso da carga completa não pode ultrapassar o limite de peso do *container*.



4. Restrições de estabilidade: um índice para estabilidade é utilizado para avaliar a estabilidade, calculado através da proporção da área de baixo da caixa em contato com as caixas abaixo para a área total da caixa. A estabilidade de todas as caixas não pode estar abaixo do total dado.
5. Restrições de equilíbrio: avalia-se o equilíbrio da carga, através da distância entre a coordenada  $x$  do centro de gravidade da carga para o ponto médio da profundidade do *container* (eixo  $x$  do *container*). A referente distância não pode exceder ao total dado. Analogamente, o mesmo critério para o eixo  $y$  é adotado. Na verdade, busca-se idealmente uma coincidência entre o centro de gravidade da carga e o centro geométrico do *container*.

Em um primeiro passo, as caixas são formadas em torres disjuntivas, através de um algoritmo “guloso”, que tende a minimizar o espaço vazio acima da caixa de base de cada torre. Salienta-se que cada torre apresenta uma única caixa de base que é posicionada no chão do *container*; as demais caixas são posicionadas acima da caixa de base.

No algoritmo de formação da torre, a caixa de base é sempre posicionada de tal forma que a dimensão horizontal de maior valor é posicionada na direção do eixo  $x$ . O posicionamento das caixas acima da caixa de base é sempre no canto inferior esquerdo e rotações da caixa são permitidas de modo que a mesma fique mais bem acomodada na torre. Preferencialmente, escolhe-se caixa de maior volume, que seja menor que o volume do espaço vazio da torre, que é dividido em três subespaços (lateral, frontal e superior), e que possa ser posicionada na mesma sem que as restrições de dimensões, posicionamento e orientação sejam violadas, conforme figura 3. As caixas vão sendo posicionadas na torre até que a mesma atinja a altura do *container* ou até que não haja mais caixa que possa ser selecionada para a torre sem que o limite de altura seja violado.

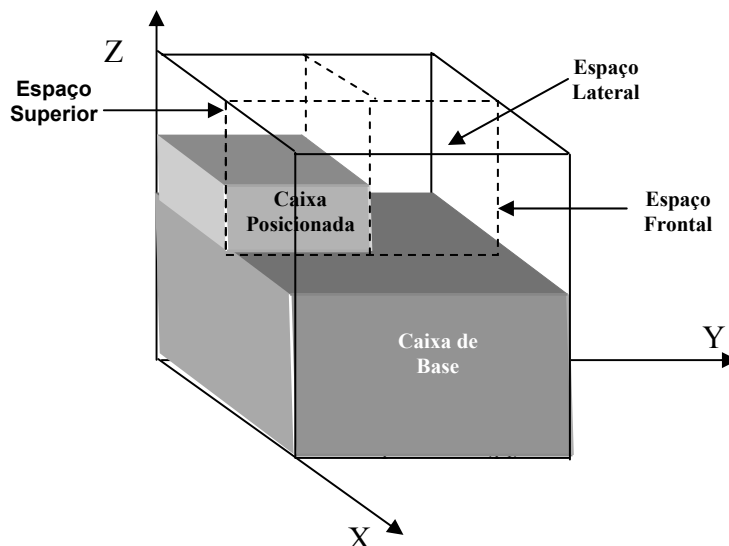


Figura 3 - Ilustração do posicionamento da caixa em cima da caixa de base e do espaço vazio da torre.

A série de torres, seguindo a descrição acima é formada até que todas as caixas estejam posicionadas em alguma torre.

Em um segundo passo, um algoritmo genético é adaptado ao problema, onde o objetivo do mesmo é determinar o posicionamento das torres de caixas no chão do *container*. Os cromossomos representam a seqüência de posicionamento das torres e os genes representam os índices das torres, onde cada torre é identificada pelo seu índice.

O algoritmo genético é desenvolvido de forma a permitir que as dimensões horizontais da torre, que ficam em paralelas aos eixos  $x$  e  $y$ , possam sofrer rotações, em outras palavras, a dimensão altura da torre tem orientação fixa (eixo  $z$ ). Operadores de cruzamento e de mutação são utilizados para a geração de nova população e um operador de seleção é utilizado para selecionar somente novos indivíduos que possuam valor de aptidão maior do que os valores dos indivíduos atuais da população e, assim, substituí-los.

O processo de formação de série de torres e de posicionamento no chão do *container* citados acima é repetido em um número de vezes especificado até que, no final, a melhor solução gerada é selecionada como solução do problema.

HE, CHA (2002), por sua vez, desenvolvem um algoritmo genético para solucionar o problema de carregamento de *container*, cujos objetivos da solução são:

1. Maximizar o volume de utilização do *container*.
2. Maximizar o peso de utilização do *container*.
3. Minimizar a altura do centro de gravidade.

Os cromossomos representam a seqüência de posicionamento das caixas e os genes representam os índices das caixas. A determinação da configuração de carregamento é descrita ainda nesta seção.

A função de aptidão total é formada por uma combinação convexa de três sub-funções de aptidão que avaliam as adequações a cada um dos objetivos apresentados acima. Cada uma dessas sub-funções de aptidão está associada a um peso na função de aptidão total, que determina o seu grau de importância. O objetivo final, portanto, é maximizar o valor da função de aptidão total apresentada abaixo:

$$\varpi = k_1 \times R + k_2 \times W + k_3 \times G ,$$

na qual  $k_1$ ,  $k_2$  e  $k_3$  são os pesos não-negativos cuja soma é 1, atribuídos às sub-funções de aptidão do volume ( $R$ ), do peso ( $W$ ) e do centro de gravidade ( $G$ ), respectivamente.

Por sua vez as sub-funções de aptidão são determinadas conforme abaixo.

a) Sub-função de aptidão do volume ( $R$ ):

$$R = \frac{\left( \sum_{i=1}^m R_{Bi} \right)}{R_c} \times 100 , \text{ onde } R_{Bi} \text{ é o volume da caixa de índice } i, R_c \text{ é o volume}$$

do *container* e  $m$  é o número de caixas carregadas.

b) Sub-função de aptidão do peso ( $W$ ):

$$W = \begin{cases} 0 & \sum_{i=1}^m W_{Bi} > W_C \\ \frac{\sum_{i=1}^m W_{Bi}}{W_C} \times 100 & \sum_{i=1}^m W_{Bi} \leq W_C \end{cases}, \text{ onde } W_{Bi} \text{ é o peso da caixa e } W_C \text{ é o peso}$$

máximo suportado pelo *container*.

c) Sub-função de aptidão do centro de gravidade ( $G$ ):

$$G = \left[ \left( H_C \times 1.5 - \frac{\sum_{i=1}^m W_{Bi} \times G_{Bi}}{\sum_{i=1}^m W_{Bi}} \right) / H_C \right] \times 100, \text{ onde } H_C \text{ é a altura do } \textit{container},$$

$G_{Bi}$  é o centro de gravidade da caixa  $i$ , que é dado pela metade da altura da caixa.

Como as sub-funções apresentadas acima são utilizadas no algoritmo proposto, no próximo capítulo é feita uma descrição um pouco mais detalhada sobre as mesmas.

A avaliação da configuração de carregamento pela função de aptidão total é feita para o que HE, CHA (2002) chamam de “padrão de carregamento”. O padrão de carregamento é uma seqüência de caixas para o carregamento, resultante da decomposição dos cromossomos, que é obtida através de um processo sistemático e seqüencial de preenchimento do espaço disponível do *container*, conforme a ordem dos genes no cromossomo.

Seguindo a seqüência dos genes, determina-se a primeira caixa do carregamento, que deverá ser posicionada na origem do *container* (canto inferior esquerdo traseiro). O posicionamento da caixa determina uma decomposição do espaço disponível em três subespaços: lateral (L), superior (M) e frontal (R), conforme figura 4. O preenchimento, então, é feito necessariamente na seqüência L, M e R, repetidamente, até que não haja mais caixas disponíveis para o preenchimento ou que não haja mais espaço disponível no *container*. A seqüência com que as caixas preenchem os subespaços determina o padrão de carregamento.

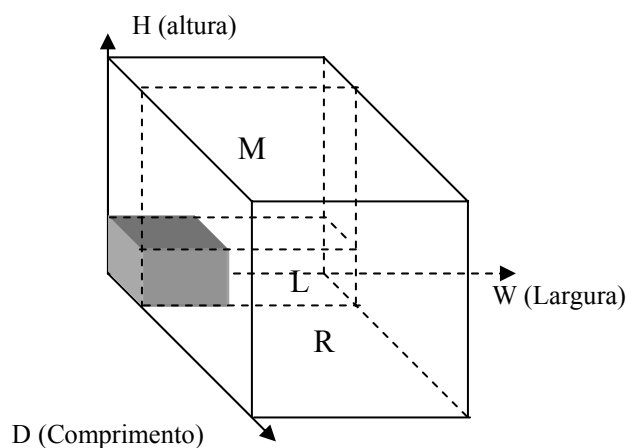


Figura 4 - Ilustração do posicionamento da primeira caixa e decomposição do espaço disponível em três subespaços: lateral (L), superior (M) e frontal (R).

O algoritmo genético desenvolve seqüências repetidas de geração de novas populações utilizando operadores de cruzamento, mutação e de um operador de seleção de indivíduos, para garantir a evolução da população. Após um número limitado de gerações, definido como regra de parada, determina-se o padrão de carregamento final, que será adotado como solução para o problema em questão.

### 3. O ALGORITMO GENÉTICO PROPOSTO

Conforme mencionado nos capítulos anteriores, soluções metaheurísticas, especialmente algoritmos genéticos, são amplamente utilizados em problemas de otimização não determinísticos (e, também, em problemas determinísticos), como o problema de carregamento de *container*. RAIDL (1999), LITVINENKO, BURGHER, *et al.* (2002), JAKOBS (1996), GEHRING, BORTFELDT (2001), HE, CHA (2002) apresentam métodos de soluções utilizando algoritmos genéticos, normalmente associados a algum esquema complementar para decodificação dos cromossomos.

O referente trabalho propõe um algoritmo genético inspirado no algoritmo de HE, CHA (2002), adequado às características e ao contexto do problema real, com possibilidade de escolha entre dois métodos de preenchimento, um dos quais se assemelha ao método de preenchimento adotado por HE, CHA (2002) e o outro que se aproxima a uma combinação entre os métodos de preenchimento utilizado por HE, CHA (2002) e por GEHRING, BORTFELDT (2001), respectivamente, Lateral, Superior e Frontal (doravante denominado L, M e R, mantendo o mesmo padrão de denominação utilizado por HE, CHA (2002)) e Superior, Lateral e Frontal (denominado M, L e R), através de formação de torres de caixas no preenchimento do *container*. Em outras palavras, propõem-se dois métodos de preenchimento, sendo um igual ao proposto por HE, CHA (2002) (L, M e R) e outro através de uma alteração na determinação dos subespaços e na ordem de seus preenchimentos (M, L e R),

levando a formação de torres de caixas, fazendo com que duas metodologias distintas sejam unificadas, passando a diferir apenas na ordem de preenchimento dos subespaços do *container*. Esta unificação e os resultados gerados a partir dela são as contribuições principais deste trabalho.

### 3.1 Objetivos do Algoritmo Genético Desenvolvido

A solução do algoritmo desenvolvido deve atender basicamente quatro objetivos:

1. Maximizar o volume de utilização do *container*.
2. Maximizar o peso permitido de utilização do *container*.
3. Maximizar a estabilidade da carga, através da minimização da altura do centro de gravidade.
4. Maximizar o valor monetário final da carga, considerando um limite dado.

Comparando-se aos objetivos do trabalho de HE, CHA (2002), acrescentou-se mais um objetivo que é o de maximização do valor monetário da carga.

O atendimento aos quatro objetivos é feito através da avaliação pela função de aptidão, tratando o problema como um problema de um objetivo, e pelas sub-funções de aptidão, como poderá ser observado no decorrer deste capítulo.

Todo o processo de determinação da solução para o carregamento feito pelo algoritmo genético é apresentado no fluxograma da figura 5.

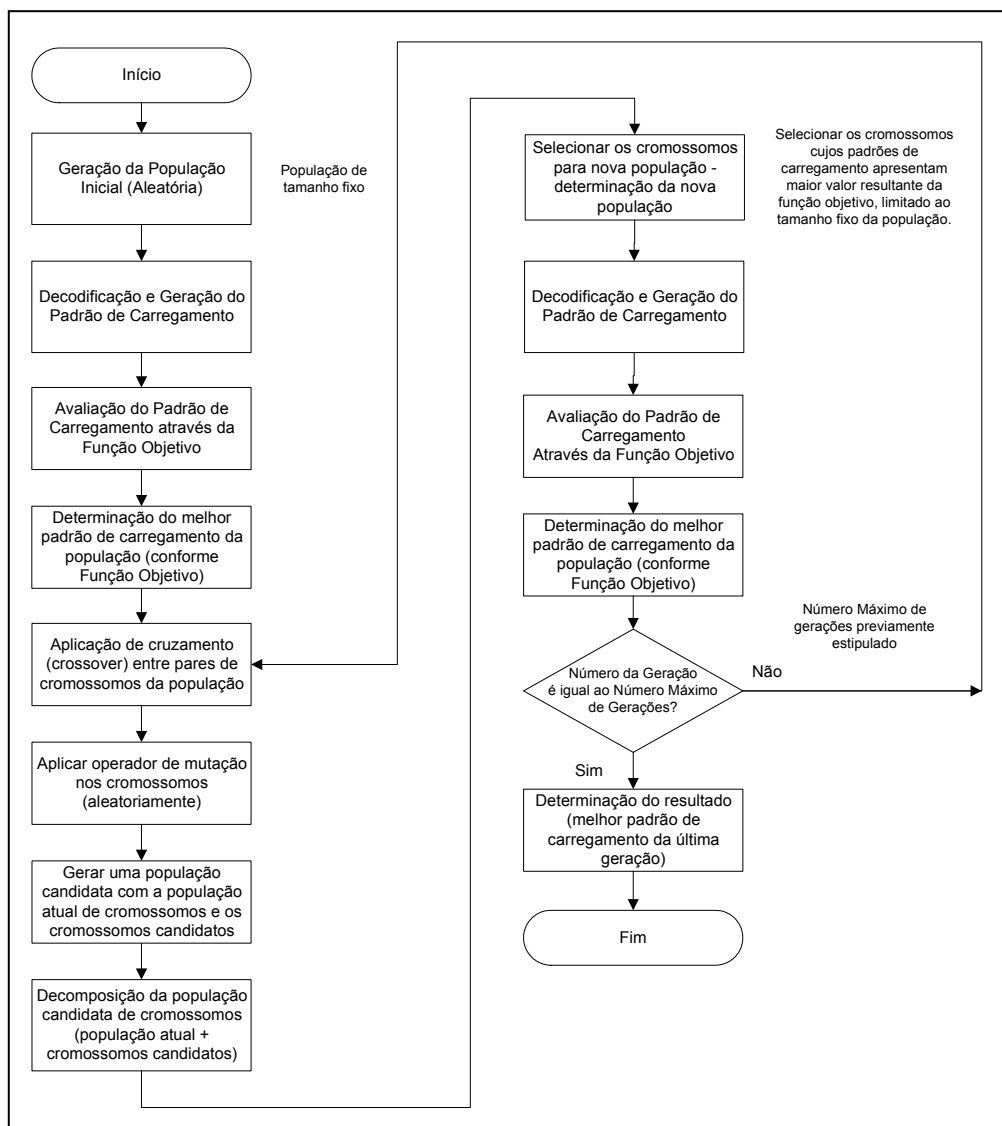


Figura 5 - Fluxograma do processo de determinação da solução para o problema de carregamento de *container* com o algoritmo genético desenvolvido.

Detalhadamente, é descrita, abaixo, a estrutura dos dados utilizados no algoritmo genético desenvolvido, a função de aptidão, a decodificação do cromossomo para obtenção do padrão de carregamento e os operadores utilizados para geração de novas populações.



### 3.2 Estrutura dos Dados

A estrutura dos dados ou codificação é extremamente importante para o algoritmo genético (JAKOBS, 1996), pois irá determinar a representação dos itens do problema em questão e o seu significado. JAKOBS (1996) descreve que uma representação natural, ou seja, de posicionamento das caixas por meio de coordenadas, não pode ser utilizada devido à complexidade presente com posicionamento de objetos tridimensionais (HE, CHA, 2002).

JAKOBS (1996) ainda propõe uma representação alternativa para um padrão de carregamento, denominada permutação  $P$ , onde esta permutação é formada por índices de caixas  $b_i$ :

$P = \{b_1, b_2, b_3, \dots, b_n\}$ , onde  $i = 1, \dots, n$  são índices das caixas.

A permutação descrita acima representa um padrão de carregamento, determinado a partir de um cromossomo da população. A seqüência  $b_1, b_2, \dots, b_n$  representa a ordem com que as caixas devem ser posicionadas no *container*, seguindo as regras de preenchimento L, M e R ou M, L e R, previamente citadas.

A população de cromossomos que determinará os padrões de carregamento (possíveis soluções) para cada população é formada por um número limitado de cromossomos. Já os cromossomos são formados por genes que representam os índices de todas as caixas disponíveis para carregamento, na ordem com que as mesmas devem ser avaliadas para determinação do padrão de carregamento, conforme poderá ser observado adiante.

Salienta-se que para um melhor entendimento da descrição dos operadores de mutação e cruzamento, decodificação de cromossomos e obtenção do padrão de carregamento, muitas das vezes é necessário utilizar mais de uma denominação para o mesmo item. Para facilitar a compreensão, a tabela 1, abaixo, demonstra as principais denominações utilizadas.

Tabela 1 - Descrição das denominações de elementos do algoritmo genético e do problema de carregamento.

<b>Denominação</b>	<b>Denominação Alternativa</b>	<b>Descrição</b>
Genes	Caixas	Os genes representam os índices de caixas no cromossomo e no padrão de carregamento.
Cromossomo	Indivíduo	Cada indivíduo de uma população é um cromossomo, com os genes que representam os índices de caixa, que será utilizado para a determinação do padrão de carregamento.
População	Geração	População é um conjunto de cromossomos. No algoritmo genético, o processo de obtenção de nova população de cromossomos também pode ser descrito como formação de uma nova geração de cromossomos.

### **3.3 População Inicial de Cromossomos**

A população inicial de cromossomos, no algoritmo desenvolvido, é determinada de forma aleatória, ou seja, todos os cromossomos são definidos de forma aleatória considerando alguns princípios básicos. Garante-se que o tamanho dos cromossomos será determinado pelo número de caixas disponibilizadas para o carregamento; como os genes representam índices de caixas, não pode haver repetição dos mesmos em um mesmo cromossomo; a quantidade de cromossomos que fazem parte de uma população é previamente definida.

### 3.4 Avaliando pela Função de Aptidão Total

A avaliação do padrão de carregamento (*packing pattern*) é feita através da função de aptidão total ou somente função de aptidão, que avalia a configuração da carga carregada conforme o volume ocupado, o peso dos produtos carregados, o centro de gravidade e o valor total dos produtos. Cada um desses aspectos avaliados demonstra o atendimento da função em relação às restrições impostas pelo problema: volume disponível, peso permitido de carregamento, centro de gravidade da carga e valor máximo das cargas. Essas avaliações específicas são feitas por meio de sub-funções de aptidão.

Cada sub-função específica de aptidão da função de aptidão total possui um peso associado que prioriza as restrições mais importantes em detrimento a restrições de menor importância. O valor final da função de aptidão é ponderado conforme os pesos das sub-funções de aptidão e o objetivo é obter o valor máximo da função de aptidão, conforme os carregamentos avaliados.

A fórmula abaixo demonstra como o valor associado à função de aptidão é calculado:

$$\varpi = \frac{k_1 \times R + k_2 \times W + k_3 \times G + k_4 \times V}{k_1 + k_2 + k_3 + k_4}$$

As sub-funções de aptidão para as restrições são as seguintes:

$R$  – Sub-função de aptidão do volume;

$W$  – Sub-função de aptidão do peso;

$G$  – Sub-função de aptidão do centro de gravidade e

$V$  – Sub-função de aptidão ao máximo valor dos produtos carregados.

Os pesos são determinados por  $k_1$ ,  $k_2$ ,  $k_3$  e  $k_4$ , que estão associados às sub-funções de aptidão do volume ( $R$ ), do peso ( $W$ ), do centro de gravidade ( $G$ ) e do valor da carga ( $V$ ), respectivamente.

As sub-funções de aptidão do volume, do peso e do centro de gravidade são as mesmas utilizadas por HE, CHA (2002), e todas as sub-funções são descritas abaixo.

#### 3.4.1 Sub-função de Aptidão do Volume ( $R$ )

Essa sub-função avalia o padrão de carregamento conforme o volume ocupado em relação ao volume disponibilizado pela carreta e o resultado é uma relação percentual entre a somatória dos volumes das caixas carregadas pelo volume disponibilizado pelo *container*, conforme expressão matemática abaixo.

$$R = \frac{\sum_{i=1}^m R_{Bi}}{R_C} \times 100$$

Na expressão,  $R_{Bi}$  é o volume de cada caixa carregada, onde  $i$  determina o índice da caixa, e  $R_C$  é o volume disponibilizado pelo *container*.

O objetivo é ocupar o máximo do volume disponível do *container* com as caixas. Como na obtenção do padrão de carregamento, através da decomposição do espaço, já leva em consideração o volume máximo disponibilizado pelo *container*, conforme suas dimensões, o valor que a função  $R$  pode atingir está entre 0 e 100.

#### 3.4.2 Sub-função de Aptidão do Peso ( $W$ )

A sub-função de aptidão ao peso ( $W$ ) avalia o peso total das caixas do padrão de carregamento em relação ao peso máximo permitido pelo *container*.

Considerando que o peso da carga não pode ultrapassar o peso máximo permitido do *container*, a sub-função apresenta duas possibilidades de resultados. Caso o peso das caixas carregadas no *container* ultrapassar o peso máximo do *container* ( $W$ ), a sub-função recebe o valor “0” (zero). Se a somatória dos pesos das

caixas não ultrapassar o valor do peso máximo do *container*, então, calcula-se a relação entre o peso total da carga pelo peso máximo de carregamento do *container*.

A fórmula abaixo faz uma descrição matemática da sub-função de aptidão ao peso:

$$W = \begin{cases} 0 & \sum_{i=1}^m W_{Bi} > W_C \\ \frac{\sum_{i=1}^m W_{Bi}}{W_C} \times 100 & \sum_{i=1}^m W_{Bi} \leq W_C \end{cases}$$

$W_C$  é o peso máximo suportado pelo *container*,  $W_{Bi}$  é o peso da caixa de índice  $i$  e  $m$  representa o número de caixas da carga.

O objetivo é maximizar o peso de caixas carregadas no *container* sem ultrapassar o limite máximo de peso suportado pelo *container*.

### 3.4.3 Sub-função de Aptidão do Centro de Gravidade (G)

O carregamento das caixas também deve ser avaliado com relação à estabilidade do *container* após as caixas serem posicionadas no mesmo. O objetivo é que o centro de gravidade seja o mais baixo possível. Para tanto, a avaliação da sub-função é realizada conforme a fórmula abaixo, onde  $W_{Bi}$  representa o peso da cada caixa carregada,  $G_{Bi}$  representa o centro de gravidade de cada caixa, conforme índice  $i$ , e assume-se que o valor médio da altura da caixa é o seu centro de gravidade, e  $H_C$  é a altura do *container*.

Pela expressão abaixo, considera-se como ideal um centro de gravidade no centro geométrico (metade da altura) do *container*, ou seja, a relação dada pela somatória dos produtos do peso e centro de gravidade da caixa pelo peso da caixa, deverá resultar no valor 0,5. Nesse caso, o valor da sub-função  $G$  é 100. Caso o centro de gravidade esteja abaixo do valor médio da altura do *container*, o valor de  $G$

será maior que 100 e caso o centro de gravidade esteja acima da metade da altura do *container*,  $G$  assumirá um valor abaixo de 100. Em outras palavras, quanto menor o centro de gravidade, maior o valor percentual resultante da sub-função  $G$ .

$$G = \left[ \left( H_C \times 1.5 - \frac{\sum_{i=1}^m W_{Bi} \times G_{Bi}}{\sum_{i=1}^m W_{Bi}} \right) / H_C \right] \times 100$$

#### 3.4.4 Sub-função de Aptidão do Valor (V)

A função de aptidão total também deve avaliar a configuração de carregamento conforme o valor máximo a ser permitido para a carga do *container*. A somatória dos valores atribuídos a cada caixa deve ser igual ou inferior ao valor máximo previamente determinado. No problema real, esse valor é determinado pela seguradora, conforme contrato com a empresa fabricante dos produtos (caixas) ou com a transportadora.

Assim como a sub-função de aptidão do peso, a sub-função de aptidão do valor recebe zero quando o valor total dos produtos carregados for superior ao limite estabelecido. Quando o valor dos produtos carregados está dentro do limite estipulado, é feita a relação entre o valor da carga em relação ao valor máximo estipulado. A fórmula abaixo demonstra como a sub-função determina o valor com relação ao atendimento da restrição ao valor.

$$V = \begin{cases} 0 & \sum_{i=1}^m V_{Bi} > V_C \\ \frac{\sum_{i=1}^m V_{Bi}}{V_C} \times 100 & \sum_{i=1}^m V_{Bi} \leq V_C \end{cases}$$

$V_{Bi}$  é o valor associado a cada produto e  $V_C$  é o valor monetário máximo que a carga do *container* deve possuir.

### 3.5 Operadores do Algoritmo Genético

A obtenção de nova população de cromossomos, que dará origem a novas configurações de carregamento possíveis, é feita por meio dos operadores de cruzamento, de mutação e pela seleção dos cromossomos, conforme anteriormente descrito.

O algoritmo genético desenvolvido avalia os padrões de carregamento por meio da função de aptidão total, conforme as quatro sub-funções de aptidão. Em seguida, é feita uma classificação dos cromossomos, conforme o valor obtido da função de aptidão para o padrão de carregamento respectivo a cada indivíduo, e, então, as operações de cruzamento e mutação são realizadas. Finalmente, a seleção é executada de tal forma que somente os indivíduos (cromossomos) cujos padrões de carregamento que possuem os maiores valores da função de aptidão total sejam aproveitados para a nova geração, e, assim, garantir um processo de evolução das possíveis configurações de carregamento, para obtenção da maximização desejada.

Os operadores de cruzamento e de mutação foram desenvolvidos especificamente para a aplicação no algoritmo genético aqui apresentado. Cabe enfatizar que estes operadores, embora se assemelhem aos operadores “clássicos” de cruzamento e mutação, apresentam importantes diferenças impostas pelo problema específico sendo estudado.

#### 3.5.1 Cruzamento

O operador de cruzamento ou *crossover* executa o cruzamento de um par de cromossomos (pais) para gerar dois novos cromossomos (filhos) que irão formar uma nova geração e que possuirão os mesmos números de genes dos cromossomos pais.

Esses novos cromossomos gerados pelo operador cruzamento contêm informações dos dois cromossomos geradores.

Os pares de cromossomos são formados conforme a classificação pelo valor da função de aptidão total e um cromossomo somente participa de um único par de cruzamentos. Quando a quantidade de cromossomos de uma população for um número ímpar, o último cromossomo da classificação não fará cruzamento com nenhum outro, e para garantir que o tamanho da população de cromossomos candidatos será a mesma da população original, esse será repetido na população candidata à nova geração de cromossomos. Os demais cromossomos participarão do processo de cruzamento.

O processo inicia-se pela determinação da posição ( $x$ ) do primeiro elemento oriundo de um dos cromossomos pais e do número de genes ( $y$ ) provenientes desse mesmo cromossomo pai, que a partir do elemento inicial, determinado por ( $x$ ), deverá compor um dos novos indivíduos. Os elementos restantes são oriundos do outro cromossomo pai, seguindo a seqüência dos elementos desse cromossomo e não repetindo nenhum elemento que já tenha sido originado do primeiro cromossomo pai. O número de elementos (genes) de um cromossomo gerado é o mesmo dos cromossomos originais. O processo é semelhante ao utilizado por HE, CHA (2002).

No exemplo abaixo (figura 6) é possível observar como é feito o cruzamento e a geração de dois novos cromossomos a partir de um par de pais geradores. Neste exemplo, os cromossomos são formados por números inteiros, que representam índice de caixas.



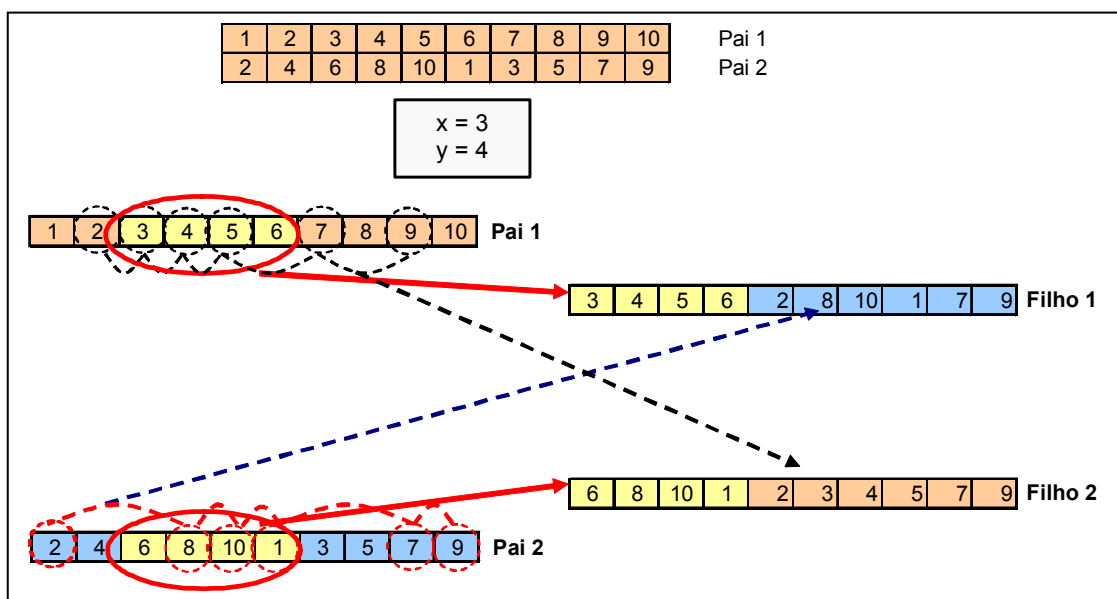


Figura 6 - Exemplo da operação de cruzamento para obtenção de dois novos cromossomos a partir de um par de cromossomos.

A posição inicial ( $x$ ) é 3 e o número de genes ( $y$ ) do primeiro cromossomo pai para um dos filhos é 4. Percebe-se, então, que os primeiros quatro genes do filho 1 são oriundos da seqüência de genes do cromossomo pai 1, a partir da posição 3. Os demais genes são oriundos do cromossomo pai 2, seguindo a seqüência de genes, do primeiro ao último, não repetindo nenhum elemento (número inteiro) que já faça parte do cromossomo gerado. O mesmo processo ocorre para a geração do filho 2, mas tendo o pai 2 como gerador dos primeiros genes do novo cromossomo e os genes do pai 1 como complemento.

### 3.5.2 Mutação

Construiu-se um operador de mutação específico para a operação de mutação a ser realizada conforme a estrutura de dados propostas. O referente operador de mutação, como os demais, a partir de uma modificação nos genes do cromossomo,

gera um novo membro da mesma classe. Como para a estrutura dos dados definidos no trabalho aqui apresentado, o cromossomo é formado por números inteiros, que não se repetem, e que representam o índice das caixas, o operador de mutação deve gerar um novo cromossomo com as mesmas características.

O processo de mutação de cromossomos se dá através da mudança aleatória das posições dos genes de um cromossomo. Essa mudança aleatória ocorre através da determinação do número aleatório de genes que sofrerão mutação, esse número é associado a um fator de probabilidade fixo  $p_m$ , sendo  $0 \leq p_m \leq 1$ , que limita o número de genes que participarão do processo de mutação.

As modificações ocorrem sempre com mais de um gene, ou seja, se para um determinado indivíduo, três genes deverão sofrer mutação, esses três genes deverão trocar de posição entre eles. Isso garante que os genes, que representam o índice das caixas, jamais se repetirão no mesmo cromossomo e que o tamanho do cromossomo não será alterado em função do processo de mutação.

Na execução do operador de mutação, é feito a determinação aleatória do número de genes que participarão da mutação ( $x$ ), em seguida, considerando-se esse número e o número de genes do cromossomo, determina-se quais os genes que participarão da mutação. Os genes participantes deverão trocar as posições entre eles. Como os genes representam índices de caixas, isso determinará um novo posicionamento das caixas para determinação posterior do padrão de carregamento.

Na figura 7, é possível verificar um exemplo da execução do processo de mutação.

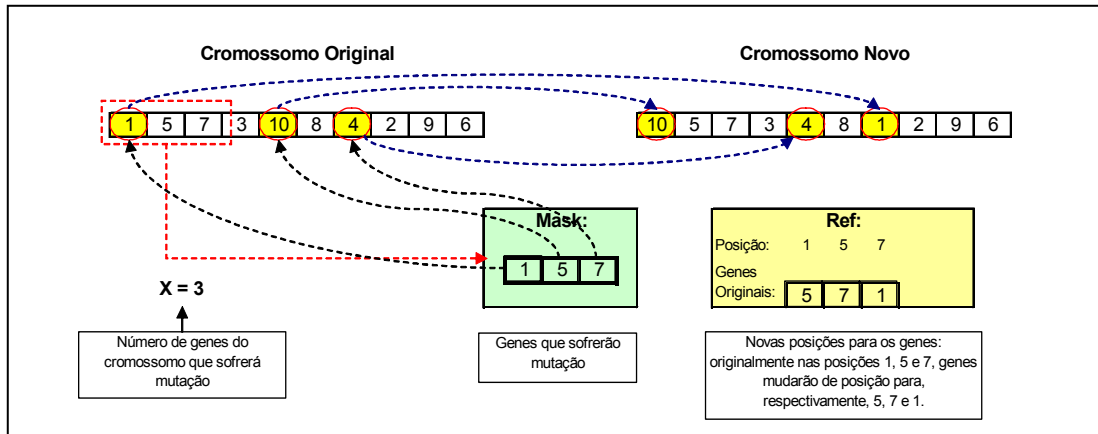


Figura 7 - Exemplo da operação de mutação que resulta em novo cromossomo pela troca de posição de genes de um cromossomo original.

No exemplo acima, o número aleatório de genes ( $x$ ) que participará do processo de mutação é 3. Em ordem aleatória, os valores referentes aos três primeiros genes são tomados para referência das posições que sofrerão mutação, que nesse caso são as posições 1, 5 e 7. A máscara ou *mask* indica, a partir do número aleatório de genes que participarão da mutação ( $x$ ), quais os genes (posições) que sofrerão a mutação. Em seguida, determina-se as novas posições dos genes anteriormente selecionados (1, 5 e 7), por meio de uma operação de mudança aleatória do posicionamento entre eles, como pode ser observado na figura 7. E, então, efetua-se as trocas de posições. No caso deste exemplo, o gene de posição 1 do novo cromossomo recebe o valor do gene de posição 5 do cromossomo original, que é 10; o gene de posição 5 recebe o valor do gene de posição 7 do cromossomo original, que é 4; e, por fim, o gene da posição 7 recebe o valor do gene de posição 1 do cromossomo inicial, que é 1.

Como o processo de mutação está associado a uma determinação aleatória do número de genes que participarão da mudança, conforme o fator de probabilidade  $p_m$ , em muitos dos casos, dependendo do valor de  $p_m$ , os cromossomos permanecem inalterados,  $x = 0$  ou 1.

### 3.6 Decodificação para obtenção do padrão de carregamento

A decodificação do cromossomo para obtenção da configuração do carregamento é realizada através do preenchimento do espaço do *container* seguindo a seqüência dos genes dos cromossomos.

O processo decorre de um método sistemático de preenchimento de subespaços, considerando as restrições que as caixas e o *container* apresentam, como limites dimensionais e de rotação, até que todo o espaço do *container* esteja preenchido ou que não haja mais caixas que possam ocupar o espaço restante.

Conforme descrito anteriormente, HE, CHA (2002) apresentam uma decodificação através de um processo de decomposição de espaços, onde originalmente o espaço disponível é dado pelo volume do *container*. Seguindo a ordem do cromossomo, verifica-se se a primeira caixa não viola as restrições dimensionais do *container*. Caso suas dimensões não ultrapassem os limites impostos pela dimensão do *container*, essa caixa é selecionada para o carregamento no *container*, sendo posicionada no canto inferior esquerdo, que será chamado de origem do *container*. Caso essa primeira caixa viole os limites do *container*, essa caixa não é selecionada, e, seguindo a seqüência dos genes no cromossomo, avalia-se a próxima caixa, até que uma caixa que seja compatível com as dimensões do *container* seja selecionada e posicionada no *container*.

A primeira caixa selecionada, que é posicionada na origem, determina uma decomposição do espaço restante do *container* em três subespaços: lateral – L, superior – M e frontal – R. Os subespaços devem ser preenchidos, conforme a seqüência dos genes, até que todo o espaço esteja preenchido ou que não haja mais caixa que possa atender às restrições dimensionais do espaço restante.

As caixas que foram selecionadas para o preenchimento do *container* formarão uma série que será denominada “padrão de carregamento” ou “*packing pattern*”. É por

esse padrão que a configuração será avaliada, através da função de aptidão total e de suas sub-funções de aptidão.

HE, CHA (2002) desenvolveram a decomposição levando a um preenchimento dos subespaços na seqüência L, M e R, respectivamente. Para o algoritmo proposto neste trabalho foram desenvolvidos dois tipos de decomposição: L,M e R, semelhante ao desenvolvido por HE, CHA (2002) e M, L e R, que inspira-se na formação de torres disjuntivas, conforme GEHRING e BORTFELDT (2001), onde a seqüência de preenchimento é iniciada pelo subespaço superior. O preenchimento completo do subespaço superior determina a formação de uma torre. À medida que novas caixas são posicionadas no chão do *container*, para preenchimentos dos subespaços L e R, novas torres são formadas.

Diferentemente de GEHRING e BORTFELDT (2001), a formação das torres não se dá através de heurística independente para posterior uso de um algoritmo genético para preenchimento do chão do *container*. No algoritmo desenvolvido, a formação de torres se dá na decodificação dos cromossomos, mas já executando o preenchimento dos subespaços do *container* nas suas três dimensões, semelhante ao preenchimento L, M e R de HE, CHA (2002). Certamente para atender a uma condição específica da formação de torres, acrescenta-se uma restrição no preenchimento M, L e R em relação ao preenchimento L, M e R; a caixa que se posicionará acima de uma outra caixa, não poderá apresentar em nenhuma das dimensões horizontais, comprimento ou largura, valor superior à dimensão respectivamente paralela da caixa de baixo, isso para evitar que a torre possa apresentar problemas de equilíbrio.

As configurações dos subespaços lateral L e superior M são ligeiramente diferente para os dois modelos de preenchimento, como poderá ser observado mais adiante.

Os dois fluxogramas abaixo (figuras 8 e 9) mostram de maneira geral como é feito o processo de preenchimento do *container* na seqüência L, M e R e em seguida na seqüência M, L e R, com formação de torres.

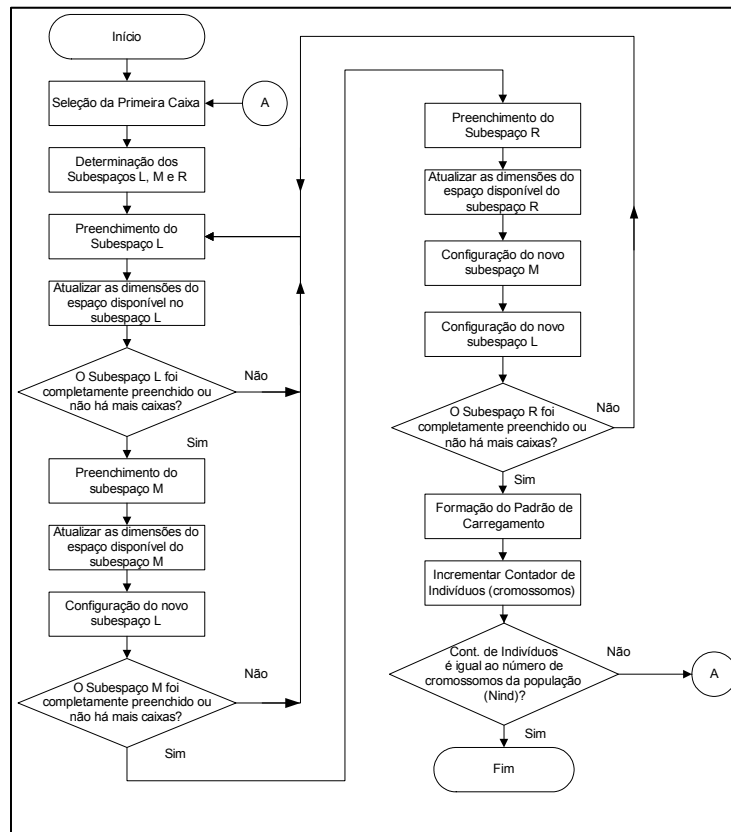


Figura 8 - Fluxograma Macro do Preenchimento de *Container* L, M e R

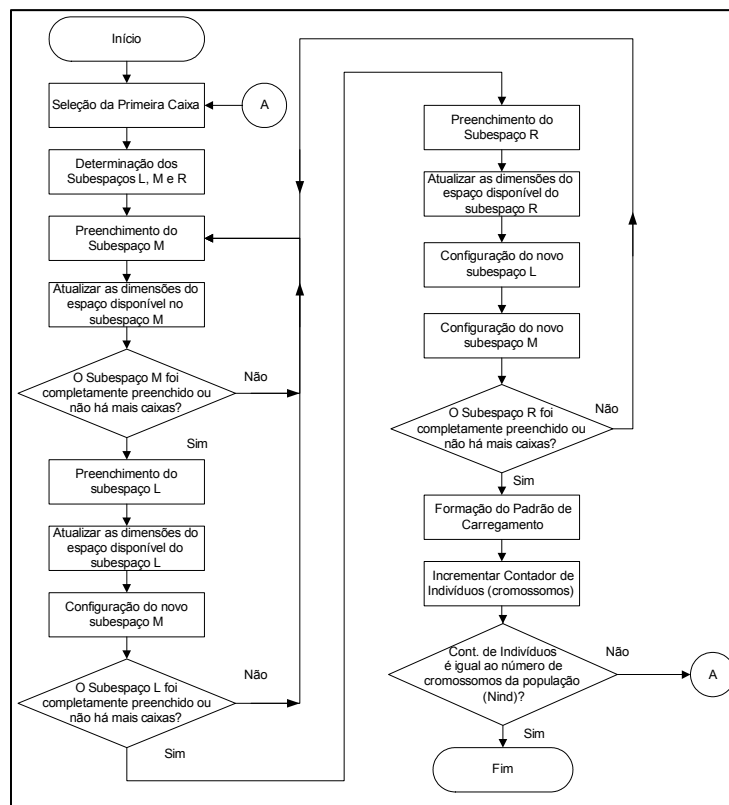


Figura 9 - Fluxograma Macro do Preenchimento de *Container* M, L e R

Abaixo, segue-se o procedimento de seleção da primeira caixa e o desenvolvimento das decomposições e preenchimentos na sequência L, M e R, bem como, na sequência M, L e R.

### 3.6.1 Seleção da Primeira Caixa

Conforme descrito acima, no primeiro momento o volume do *container* é todo o espaço disponível para o carregamento. Logo, conforme a sequência de genes (índices de caixas) apresentados no cromossomo, é selecionada a primeira caixa para o carregamento, que será posicionada na origem, e que será a primeira caixa do padrão de carregamento.

A seleção da caixa, além de obedecer a sequência dos genes do cromossomo, envolve um processo de comparações entre as dimensões da caixa e do *container*.

1. Verifica-se se o volume da caixa é menor que o volume do *container*.

$$R_{B_i} < R_C;$$

2. Verifica-se se a área da superfície horizontal inferior da caixa é menor que a área da superfície horizontal do *container*.

$$A_{B_i} < A_C;$$

3. Verifica-se se a altura da caixa é menor que a altura do *container*.

$$H_{B_i} < H_C;$$

Salienta-se que as caixas têm suas alturas com orientações fixas, não podendo sofrer rotações desta dimensão, por conta das restrições de carregamento e armazenamento dos produtos.

4. Verifica-se se a menor dimensão horizontal da caixa, aqui denominada de comprimento da caixa, é menor que o comprimento do *container*.

$$L_{B_i} < L_C;$$

5. Verifica-se se a maior dimensão horizontal da caixa, aqui denominada de largura da caixa, é menor que a largura do *container*,

$$W_{Bj} < W_C;$$

6. Caso a maior dimensão horizontal da caixa, não seja menor que a largura do *container*, item 4, verifica-se o enquadramento da caixa no espaço do *container*, executando uma rotação de 90° na caixa. A verificação a ser executada agora é se a menor dimensão da caixa, comprimento, é menor que a largura do *container*.

$$L_{Bj} < W_C$$

7. Em seguida, verifica-se se a maior dimensão da caixa, largura, é menor que o comprimento do *container*.

$$W_{Bj} < L_C;$$

Caso os itens de 1 a 5 ou caso os itens de 1 a 3 e 6 a 7 sejam satisfeitos, a caixa é selecionada e posicionada no canto inferior esquerdo. A partir desse posicionamento é feita a decomposição do espaço disponível nos três subespaços já mencionados.

O processo de seleção da primeira caixa pode ser visualizado no fluxograma a seguir, onde se observa vários pontos de decisão, que representam as verificações das dimensões da caixa com relação às dimensões do *container*.



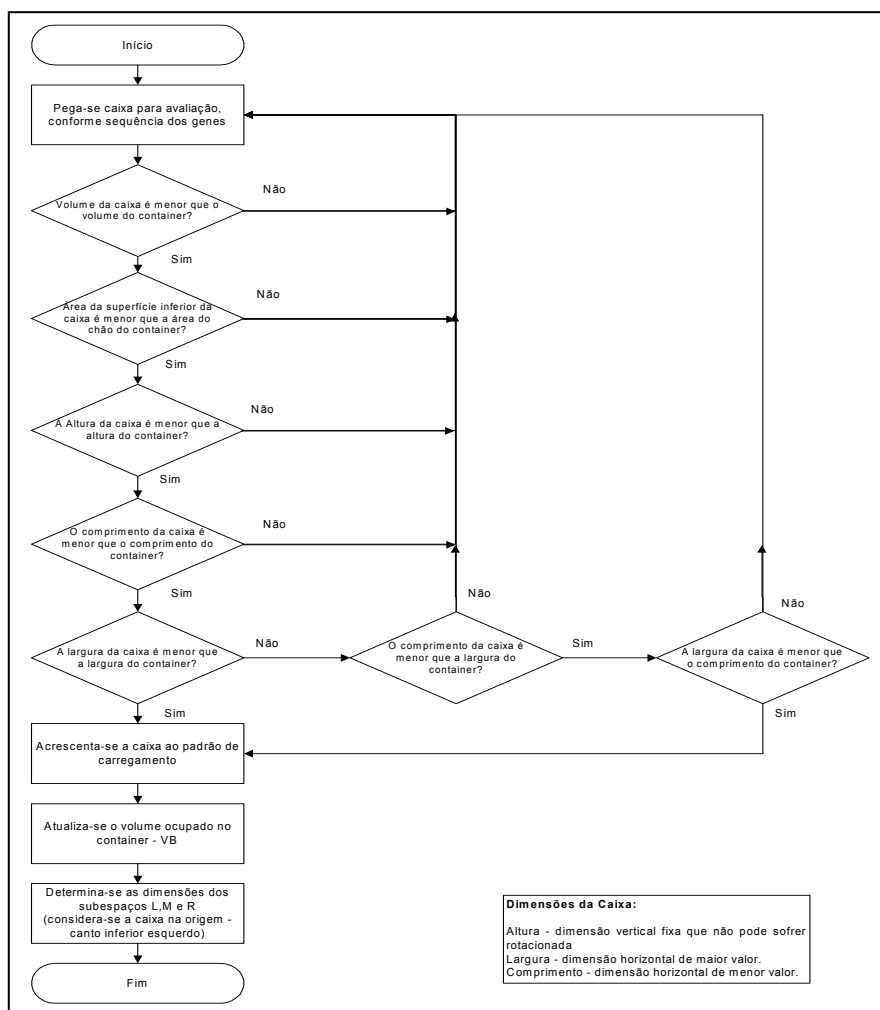


Figura 10 - Fluxograma da Seleção da Primeira Caixa

### 3.6.2 Decomposição L, M e R

Esta subseção descreve com detalhes os passos da determinação dos subespaços disponíveis e a formação do padrão de carregamento (*packing pattern*) através do preenchimento do espaço disponível na ordem L, M e R, ou seja, lateral, superior e frontal.

### 3.6.2.1 Obtenção dos Subespaços

A primeira caixa posicionada determinará o espaço disponível restante e os subespaços L, M e R, que são provenientes da decomposição do espaço total disponível, conforme pode ser observado na figura abaixo.

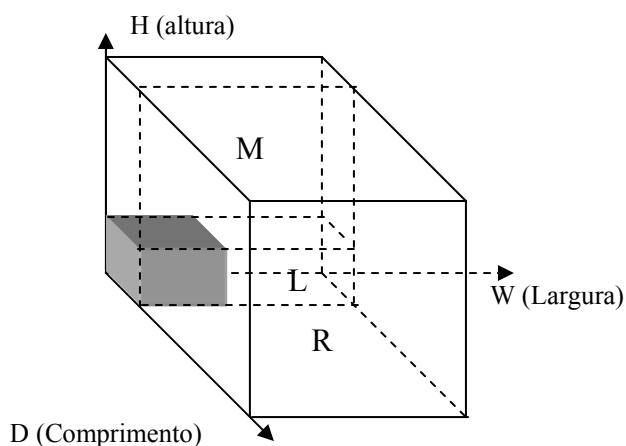


Figura 11 - Ilustração do posicionamento da primeira caixa e formação dos subespaços da decomposição L, M e R.

Os eixos sobre os quais estão posicionadas as dimensões do *container* são, H, W e D para as dimensões altura, largura e comprimento, respectivamente. Para o comprimento utilizou-se D, para evitar conflito de notação com o subespaço L e é oriundo do termo em inglês *deep* (profundidade), que também é utilizado para designação do comprimento do *container*.

- a) L é o subespaço lateral cuja altura é determinada pela altura da caixa. A dimensão paralela ao comprimento do *container*, que é denominada de comprimento do subespaço, é determinada pela dimensão da caixa que fica paralela ao eixo do comprimento do *container*, caso a caixa não sofra rotação. E a dimensão perpendicular ao comprimento, que é denominada de largura do subespaço, é determinada pela diferença entre a largura do *container* e a largura da caixa, caso a caixa não sofra rotação. Matematicamente, tem-se:

$$\begin{aligned}
 H_L &= H_{B1} \\
 L_L &= L_{B1} \\
 W_L &= W_C - W_{B1}
 \end{aligned}$$

No qual:

$H_L$  - Altura do subespaço L;

$L_L$  - Comprimento do subespaço L;

$W_L$  - Largura do subespaço L;

$H_{B1}$  - Altura da primeira caixa;

$L_{B1}$  - Comprimento da primeira caixa;

$W_{B1}$  - Largura da primeira caixa;

$W_C$  - Largura do *container*.

Caso a caixa sofra rotação de 90°, as dimensões horizontais do subespaço L são alteradas. O comprimento do subespaço L passa a ser determinado pela dimensão da caixa que fica paralela ao eixo do comprimento do *container*, que agora é a largura da caixa, que por sua vez, é a maior dimensão horizontal da caixa. A largura do subespaço L passa a ser determinado pela diferença entre a largura do *container* e o comprimento da caixa (menor dimensão horizontal da caixa). Podemos reescrever matematicamente a obtenção das dimensões, conforme abaixo:

$$\begin{aligned}
 H_L &= H_{B1} \\
 L_L &= W_{B1} \\
 W_L &= W_C - L_{B1}
 \end{aligned}$$

A área da superfície horizontal do subespaço L é determinada pelo produto do comprimento e largura do mesmo, conforme abaixo.

$$A_L = L_L \times W_L$$

O volume do subespaço L é, portanto, dado pelo produto das três dimensões, comprimento, largura e altura.

$$V_L = L_L \times W_L \times H_L$$

b) O subespaço M, que é o subespaço superior, tem sua altura determinada pela diferença entre a altura do *container* e a altura da caixa. O comprimento do subespaço é dado pelo comprimento da caixa, caso esta não sofra rotação. Quando há rotação da caixa no seu posicionamento, o comprimento do subespaço é determinado pela largura da caixa. A largura do subespaço M, por sua vez, é determinada pela largura do *container*.

No caso de não haver rotação na primeira caixa, matematicamente, as dimensões do subespaço M podem ser determinadas, conforme abaixo:

$$\begin{aligned}H_M &= H_C - H_{B1} \\L_M &= L_{B1} \\W_M &= W_C\end{aligned}$$

No qual:

$H_M$  – Altura do subespaço M;

$L_M$  – Comprimento do subespaço M;

$W_M$  – Largura do subespaço M;

$H_C$  – Altura do *container*.

No caso de rotação da primeira caixa, as dimensões podem ser obtidas da seguinte forma:

$$\begin{aligned}H_M &= H_C - H_{B1} \\L_M &= W_{B1} \\W_M &= W_C\end{aligned}$$

A área da superfície inferior do subespaço M é resultado do produto do comprimento do subespaço pela largura do mesmo.

$$A_M = L_M \times W_M$$

O volume do subespaço M é obtido pelo produto das três dimensões do mesmo.

$$V_M = L_M \times W_M \times H_M$$

c) Por sua vez o subespaço R, que é o subespaço frontal, tem suas dimensões determinadas conforme abaixo.

A altura do subespaço R é a própria altura do *container*, a largura do subespaço R é a largura do *container*. Já o comprimento do subespaço R é determinado pela diferença entre o comprimento do *container* e o comprimento da caixa, quando a caixa não sofre rotação. Quando a caixa sofre rotação, o comprimento do subespaço R é dado pela diferença entre o comprimento do *container* e a largura da caixa.

No processo de preenchimento L, M e R, as caixas devem preencher os subespaços obedecendo à seqüência dos índices das caixas e a seqüência dos subespaços a serem preenchidos: lateral (L), superior (M) e frontal (R), respectivamente. As dimensões dos subespaços são atualizadas a cada caixa selecionada para o padrão de carregamento.

O processo de preenchimento e atualização das dimensões do subespaço é descrito mais detalhadamente abaixo.

### **3.6.2.2 Preenchimento dos Subespaços**

Assim como ocorre na seleção da primeira caixa, a seleção das caixas para o preenchimento dos subespaços segue um procedimento de comparações entre as dimensões das caixas e dimensões do subespaço.

#### **3.6.2.2.1 Preenchimento do subespaço L**

Seguindo a seqüência dos genes (caixas) do cromossomo, é feita a seguinte avaliação:

1. Verifica-se se o volume da caixa é menor que o volume disponível do subespaço L ( $V_{LD}$ ), que é inicialmente todo o subespaço L ( $V_L$ ), já que inicialmente todo o subespaço L está disponível. Observa-se que o subscrito adicional D, utilizado na descrição dos preenchimentos dos subespaços nos dois modelos de preenchimento, origina-se da palavra “disponível”, aqui associada a subespaços e à área.

$$V_{Bi} \leq V_{LD}$$

2. Em seguida, verifica-se se a área da superfície inferior da caixa é menor que a área superficial do espaço disponível do subespaço L ( $A_{LD}$ ).

$$A_{Bi} \leq A_{LD}$$

3. Segue-se a verificação da altura da caixa, que deve ser menor que a altura do espaço disponível do subespaço L ( $H_{LD}$ ).

$$H_{Bi} \leq H_{LD}$$

4. O comprimento da caixa (menor dimensão horizontal) deverá ser menor que o comprimento do espaço disponível do subespaço L ( $L_{LD}$ ).

$$L_{Bi} \leq L_{LD}$$

5. Verifica-se, agora, se a largura da caixa (maior dimensão horizontal) é menor que a largura do espaço disponível do subespaço L ( $W_{LD}$ ).

$$W_{Bi} \leq W_{LD}$$

6. Caso a largura da caixa seja maior que a largura do espaço disponível do subespaço L, a caixa deverá sofrer rotação para verificar se, desta maneira, a mesma pode ser posicionada no subespaço L. Neste caso, o comprimento da caixa deverá ser menor que a largura do espaço disponível do subespaço em questão, e a largura da caixa deverá ser menor que o comprimento do espaço disponível do subespaço.

7. Se a caixa tiver dimensões compatíveis com as dimensões do espaço disponível do subespaço L, é necessário garantir que a caixa em avaliação não foi selecionada anteriormente.

Se a caixa avaliada for compatível com as dimensões do subespaço, ou seja, conseguir atender todos os critérios descritos acima, a caixa é selecionada, posicionada no subespaço, no canto inferior esquerdo do espaço disponível, e incluída na série de padrão de carregamento. O volume preenchido do *container* e o volume preenchido do subespaço L deverão ser atualizados, acrescentando a eles o volume da caixa selecionada.

As dimensões do espaço disponível do subespaço L deverão ser atualizadas: a altura do espaço disponível permanece inalterada, assim como o comprimento. Já a largura do espaço disponível do subespaço L, deverá ser subtraída da dimensão da caixa posicionada em paralelo a largura do *container*, ou seja, a largura, no caso em que a caixa não sofre rotação, ou comprimento da caixa, quando a mesma sofre rotação para ser posicionada no subespaço.

Matematicamente, tem-se:

$$H_{LD}(i) = H_{LD}(i - 1)$$

$$L_{LD}(i) = L_{LD}(i - 1)$$

$W_{LD}(i) = W_{LD}(i - 1) - W_B(j)$ , quando a caixa não sofre rotação no posicionamento.

$W_{LD}(i) = W_{LD}(i - 1) - L_B(j)$ , quando a caixa sofre rotação de 90° na superfície horizontal no posicionamento.

Os índices  $i$  e  $j$  são respectivamente os índices de caixa selecionada para o subespaço L e índice geral da caixa.

O volume e a área superficial do espaço disponível são automaticamente atualizados:

$$V_{LD}(i) = W_{LD}(i) \cdot L_{LD}(i) \cdot H_{LD}(i), \text{ sendo } i \text{ o índice de caixa selecionado.}$$

Os espaços disponíveis dos subespaços M e R permanecem inalterados.

Os passos acima são repetidos até que todo o subespaço L esteja preenchido; isso ocorre quando não há mais espaço disponível no subespaço L ou quando não há mais caixas que ainda não tenham sido selecionadas e que possuam dimensões compatíveis com o espaço disponível do subespaço.

Quando não é possível mais preencher o subespaço L, passa-se a selecionar uma caixa para ser posicionada no subespaço M.

### 3.6.2.2.2 Preenchimento do Subespaço M

Os critérios para seleção da caixa para preenchimento do subespaço M são similares aos critérios adotados do subespaço L, permanecendo as mesmas comparações de dimensões, mudando apenas o espaço disponível que passa ser do subespaço M.

A seqüência das caixas no cromossomo deve ser obedecida, ou seja, verifica-se a primeira caixa e caso essa não seja selecionada, conforme os critérios abaixo, avalia-se a caixa seguinte até que uma caixa seja selecionada. Lembra-se, também, que inicialmente, todo o subespaço M está disponível.

Seguem-se os passos de seleção da caixa para posicionamento no subespaço M:

1. Volume da caixa deve ser menor que o volume disponível do subespaço M ( $V_{MD}$ ), que é inicialmente todo o subespaço M ( $V_M$ ).

$$V_{Bi} \leq V_{MD}$$

2. A área da superfície inferior da caixa deve ser menor que a área da superfície horizontal do espaço disponível do subespaço M ( $A_{MD}$ ).

$$A_{Bi} \leq A_{MD}$$



3. A altura da caixa deve ser menor que a altura do espaço disponível do subespaço M ( $H_{MD}$ ).

$$H_{Bi} \leq H_{MD}$$

4. O comprimento da caixa deve ser menor que o comprimento do espaço disponível do subespaço M ( $L_{MD}$ ).

$$L_{Bi} \leq L_{MD}$$

5. A largura da caixa deve ser menor que a largura do espaço disponível do subespaço M ( $W_{MD}$ ).

$$W_{Bi} \leq W_{MD}$$

6. Caso a largura da caixa seja maior que a largura do espaço disponível do subespaço M, a caixa deverá sofrer rotação para verificar se, desta maneira, a mesma pode ser posicionada no subespaço. Neste caso, o comprimento da caixa deverá ser menor que a largura do espaço disponível do subespaço e avalia-se agora se a largura da caixa é menor que o comprimento do espaço disponível do subespaço.
7. Se todos os passos acima descritos forem contemplados, será necessário garantir que a caixa em avaliação não foi selecionada anteriormente.

Cumprindo todos os passos descritos acima, a caixa será incluída no padrão de carregamento e será posicionada no canto inferior esquerdo do espaço em questão. O volume ocupado do *container* e o volume ocupado do espaço M devem ser atualizados.

As dimensões do espaço disponível do subespaço M são atualizadas, conforme abaixo:

A altura é dada por:  $H_{MD}(i) = H_{MD}(i-1) - H_B(j)$ .

O comprimento é dado por  $L_{MD}(i) = L_{MD}(i-1)$ .

E a largura é dada por  $W_{MD}(i) = W_{MD}(i-1)$ .

Sendo  $i$  o índice de caixa selecionada do subespaço e  $j$ , o índice geral da caixa.

O posicionamento de uma caixa no subespaço M origina um novo subespaço L, cujas dimensões são obtidas de forma similar quando do posicionamento da primeira caixa. Esse novo subespaço, que inicialmente está totalmente disponível para preenchimento, deverá ser preenchido, conforme a descrição feita acima para o subespaço L, fazendo-se sempre uma verificação da aptidão das caixas ao espaço disponível. O comprimento do subespaço L permanece o mesmo.

As dimensões do novo subespaço L são obtidas da seguinte forma:

$$\begin{aligned}H_L &= H_{BM1} \\L_L &= L_{MD}(i) \\W_L &= W_C - W_{BM1}\end{aligned}$$

Sendo,  $H_{BM1}$ ,  $L_{BM1}$ , e  $W_{BM1}$ , a altura, o comprimento e a largura da caixa posicionada no subespaço M e  $H_L$ ,  $L_L$ , e  $W_L$ , a altura, o comprimento e a largura do novo subespaço L.

A obtenção da largura do novo subespaço L descrita acima é válida somente quando a caixa não sofre rotação no seu posicionamento. Caso a caixa seja posicionada com rotação de  $90^\circ$  no subespaço M, a largura do subespaço L serão obtidos conforme abaixo:

$$W_L = W_C - L_{BM1}$$

Esse novo subespaço L deverá ser preenchido até que não haja mais espaço vazio no mesmo ou que não haja mais caixa que possa preencher os espaços, conforme os critérios já mencionados. Após isso, novamente seleciona-se uma nova caixa para ser posicionada no subespaço M, cujas dimensões do seu espaço disponível deverão ser novamente atualizadas e um novo subespaço L deverá ser gerado.

Esse processo deverá ser repetido até que todo o subespaço M e o subespaço gerado L estejam totalmente preenchidos ou que não haja mais caixas que ainda não

tenham sido selecionadas anteriormente e que não possam ocupar os espaços disponíveis. Quando isso ocorre, parte-se agora para a seleção da caixa que deverá preencher o subespaço frontal R.

### 3.6.2.2.3 Preenchimento do Subespaço R

Para o preenchimento do subespaço R, seguem-se os mesmos critérios utilizados para a seleção de caixas nos subespaços L e M. A seqüência das caixas no cromossomo deve ser obedecida e considera-se inicialmente todo o subespaço R disponível para posicionamento de caixas.

Os passos anteriormente descritos para a seleção das caixas para os subespaços L e M valem também para o subespaço R, conforme abaixo:

1. Volume da caixa deve ser menor que o volume disponível do subespaço R ( $V_{RD}$ ).

$$V_{Bi} \leq V_{RD}$$

2. A área da superfície inferior da caixa deve ser menor que a área da superfície horizontal do espaço disponível do subespaço R ( $A_{RD}$ ).

$$A_{Bi} \leq A_{RD}$$

3. A altura da caixa deve ser menor que a altura do espaço disponível do subespaço R ( $H_{RD}$ ).

$$H_{Bi} \leq H_{RD}$$

4. O comprimento da caixa deve ser menor que o comprimento do espaço disponível do subespaço R ( $L_{RD}$ ).

$$L_{Bi} \leq L_{RD}$$

5. A largura da caixa deve ser menor que a largura do espaço disponível do subespaço R ( $W_{RD}$ ).

$$W_{Bi} \leq W_{RD}$$

6. Caso a largura da caixa seja maior que a largura do espaço disponível do subespaço R, a caixa deverá sofrer rotação para verificar se, desta maneira, a mesma pode ser posicionada no subespaço. Neste caso, o comprimento da caixa deverá ser menor que a largura do espaço disponível do subespaço e avalia-se agora se a largura da caixa é menor que o comprimento do espaço disponível do subespaço.
7. Após os passos acima, é necessário garantir que a caixa em avaliação não foi selecionada anteriormente.

Contemplando-se todos os critérios acima descritos, a caixa é selecionada, posicionada no espaço R, sempre no canto inferior esquerdo do espaço disponível, e é incluída no padrão de carregamento.

Essa caixa posicionada no subespaço R determinará um novo subespaço M e um novo subespaço L. As dimensões desses novos subespaços deverão ser obtidas conforme o posicionamento da primeira caixa, ou seja, o espaço entre a dimensão da caixa em paralelo com a largura do *container* até a lateral do *container*, limitado pela altura e dimensão da caixa posicionada em paralelo ao comprimento do *container*, determinará o subespaço L. O espaço superior M é compreendido pelo espaço entre a caixa e o teto do *container*, estendendo-se lateralmente por toda a largura do *container*, limitado pela dimensão da caixa paralela ao comprimento do *container*.

Na figura 12 é possível observar os novos subespaços, sendo a caixa de cor verde, a caixa posicionada no subespaço R, que deu origem aos novos subespaços L e M, e que diminuiu o espaço disponível em R. A ilustração permite que haja uma interpretação do preenchimento, como um preenchimento em camadas e que a cada nova caixa posicionada no subespaço R, nova camada é gerada.

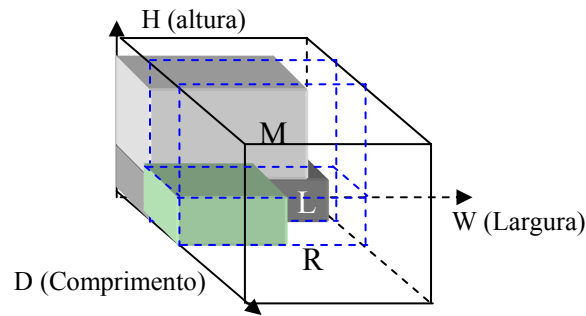


Figura 12 - Ilustração do posicionamento de nova caixa no subespaço R, e formação de novos subespaços L e M.

Das dimensões do espaço disponível do subespaço R, somente o comprimento é alterado, reduzindo do seu tamanho anterior, a dimensão da caixa que está paralela ao comprimento do *container*, que é o próprio comprimento da caixa, quando a mesma não sofre rotação no posicionamento, ou é a largura da caixa, quando esta sofreu uma rotação de  $90^\circ$  no posicionamento no subespaço. Tem-se:

$$L_{RD}(i) = L_{RD}(i-1) - L_B(j), \text{ para caixa posicionada sem rotação.}$$

$$L_{RD}(i) = L_{RD}(i-1) - W_B(j), \text{ para caixa posicionada com rotação.}$$

As dimensões dos novos subespaços podem ser obtidas, conforme abaixo:

Subespaço M, sem rotação no posicionamento da caixa:

$$H_M = H_C - H_{BR1}$$

$$L_M = L_{BR1}$$

$$W_M = W_C$$

Subespaço M, com rotação no posicionamento da caixa:

$$H_M = H_C - H_{BR1}$$

$$L_M = W_{BR1}$$

$$W_M = W_C$$

Subespaço L, sem rotação no posicionamento da caixa:

$$H_L = H_{BR1}$$

$$L_L = L_{BR1}$$

$$W_L = W_C - W_{BR1}$$

Subespaço L, com rotação no posicionamento da caixa:

$$H_L = H_{BR1}$$

$$L_L = W_{BR1}$$

$$W_L = W_C - L_{BR1}$$

Após o posicionamento da caixa no subespaço R, o novo subespaço L deve ser preenchido e, em seguida, o novo subespaço M. Quando os novos subespaços forem preenchidos, uma nova caixa é selecionada para o espaço disponível do subespaço R. O processo se repete até que todo o subespaço R seja preenchido ou que não haja mais caixa que possa ser posicionada nos espaços vazios. Quando isso acontece, termina-se a formação da série padrão de carregamento ou *packing pattern*, que será avaliado conforme função de aptidão total e sub-funções de aptidão.

As figuras 13 e 14 mostram o fluxograma do processo de seleção de caixas para preenchimento dos subespaços L, M e R.

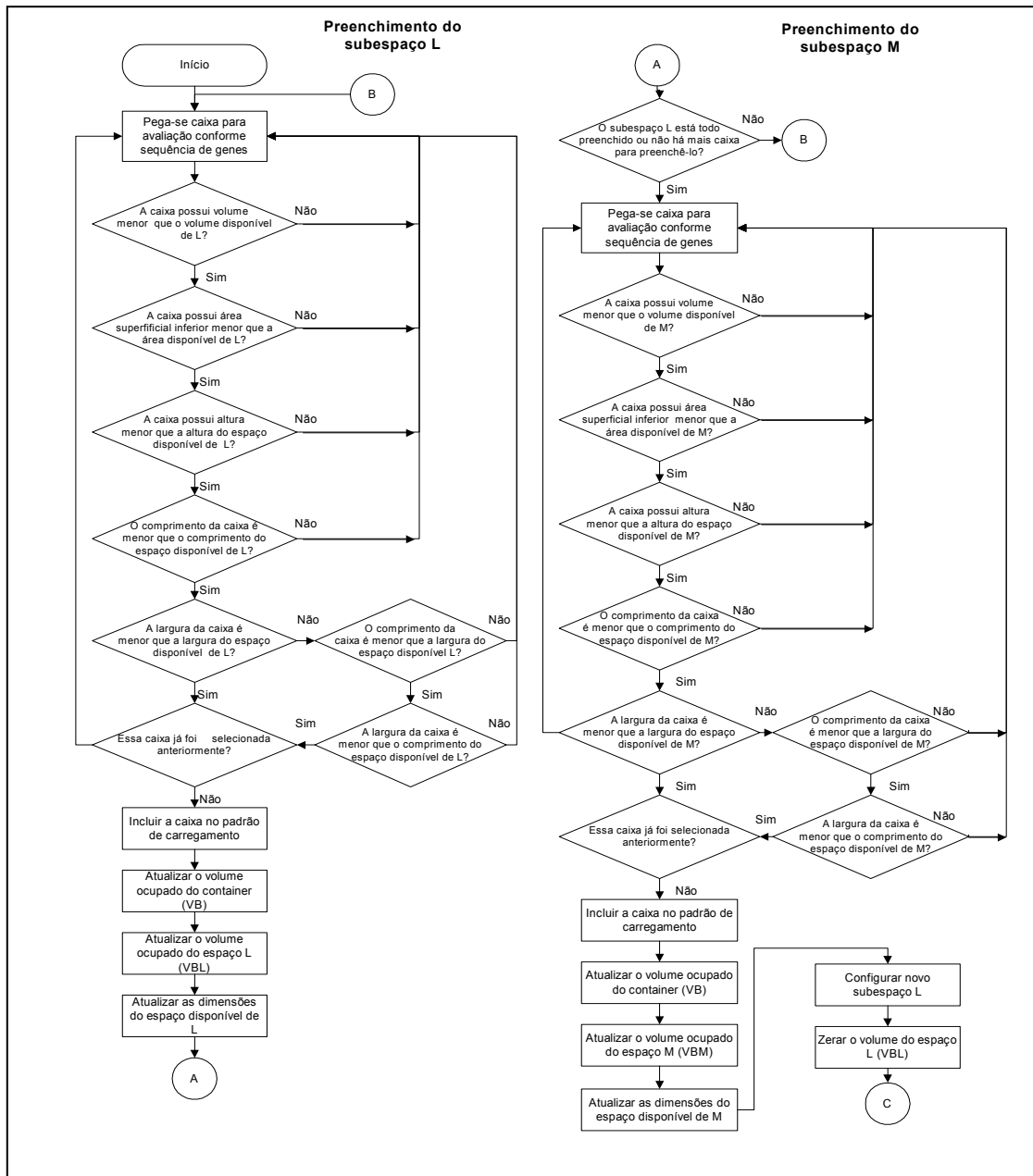


Figura 13 - Processo de seleção de caixas e preenchimento de subespaços L e M, da seqüência L, M e R.



Figura 14 - Processo de seleção de caixas e preenchimento de subespaços R, da seqüência L, M e R.



### 3.6.3 Decomposição M, L e R

Como já mencionado, a decomposição M, L e R, caracteriza-se pela formação de torres de caixas que deverão ser posicionadas no *container*. Essa decomposição e preenchimento por torres podem ser obtidos de maneira similar à decomposição L, M e R, contudo a seqüência de preenchimento dos subespaços é diferente.

Na seqüência L, M e R, a seqüência de preenchimento privilegia o preenchimento do subespaço lateral L, em seguida o subespaço superior M e por último o subespaço frontal R. Essa seqüência de preenchimento caracteriza-se por um preenchimento de fileiras de caixas, do chão até o teto do *container*, já que se preenche todo o subespaço lateral antes de se colocar uma nova caixa no subespaço superior, que formará uma nova fileira de caixas, e quando o subespaço lateral e superior já estão completamente preenchidos, uma outra fileira de caixas será formada com o posicionamento de uma nova caixa no subespaço frontal R.

Para a formação de torres, a seqüência de preenchimento das caixas passa a ser, o preenchimento do subespaço superior M, que agora terá uma configuração diferente do subespaço superior da seqüência L, M e R, depois se preenche o subespaço lateral L, que também é diferente do subespaço L da seqüência anteriormente estudada e, por último, o subespaço frontal R.

As diferenças na formação e no preenchimento dos subespaços são tratadas abaixo.

#### 3.6.3.1 Obtenção dos Subespaços

Assim como ocorre para a decomposição L, M e R, na decomposição M, L e R, a primeira caixa posicionada determinará o espaço disponível restante e os subespaços. A seleção da primeira caixa é feita da mesma forma, seleciona-se uma

caixa, conforme a seqüência de genes cujas dimensões permitem que a mesma seja posicionada dentro do *container*, sem violar suas restrições dimensionais e sem provocar uma rotação na caixa que não seja de suas dimensões horizontais (largura e comprimento).

A figura 15, abaixo, auxiliará no entendimento da determinação dos subespaços.

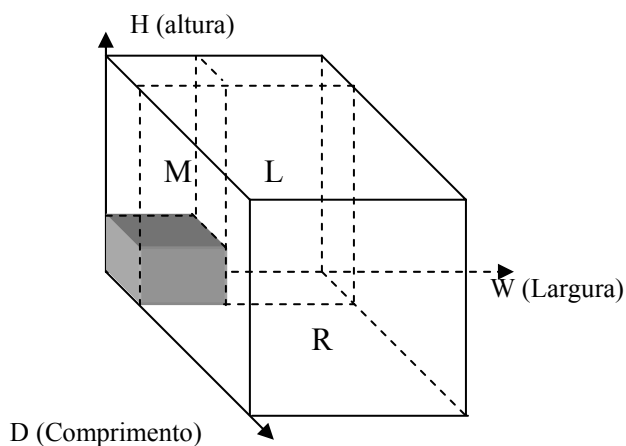


Figura 15 - Ilustração do posicionamento da primeira caixa e formação dos subespaços M, L e R, da decomposição M, L e R.

a) O subespaço M é o subespaço que determinará a formação de cada torre, já que o mesmo delimita as dimensões da mesma, como pode ser observado na figura 15. A altura do subespaço original M é dada pela diferença entre a altura do *container* e a altura da caixa posicionada na origem. O comprimento é dado pela dimensão horizontal da caixa (comprimento ou largura) posicionada em paralelo ao comprimento do *container* (D). A largura do subespaço M é dada pela dimensão horizontal da caixa posicionada em paralelo a largura do *container*. Quando não há rotação da caixa no seu posicionamento, o comprimento do subespaço M é dado pelo comprimento da caixa (menor dimensão horizontal da caixa) e a largura do subespaço pela largura da caixa (maior dimensão horizontal da caixa). Quando a primeira caixa sofre rotação de  $90^\circ$  no seu posicionamento, o comprimento do subespaço M

passa a ser dado pela largura da caixa e a largura do subespaço pelo comprimento da caixa.

Quando não há rotação da caixa:

$$H_M = H_C - H_{B1}$$

$$L_M = L_{B1}$$

$$W_M = W_{B1}$$

No qual:

$H_M$  - Altura do subespaço M;

$L_M$  - Comprimento do subespaço M;

$W_M$  - Largura do subespaço M;

$H_{B1}$  - Altura da primeira caixa;

$L_{B1}$  - Comprimento da primeira caixa;

$W_{B1}$  - Largura da primeira caixa;

$H_C$  - Altura do *container*;

$W_C$  - Largura do *container*.

Quando na rotação de 90°, o comprimento e a largura do subespaço são obtidos por:

$$L_M = W_{B1}$$

$$W_M = L_{B1}$$

b) O subespaço L, por sua vez compreende todo o espaço lateral à caixa posicionada, limitada à altura do *container* e ao comprimento da caixa, conforme pode ser observado na figura 15. A altura, portanto é dada pela própria altura do *container*. A largura é a diferença entre a largura do *container* e a dimensão da caixa posicionada em paralelo à largura do *container*. Quando a caixa não sofre rotação no posicionamento, a largura do subespaço é dada pela diferença entre a largura do *container* e a largura da caixa, quando há rotação, a largura do subespaço é dada pela diferença entre largura do

*container* e o comprimento da caixa. O comprimento do subespaço é dado pela dimensão da caixa posicionada em paralelo ao comprimento do *container*, comprimento da caixa, quando a mesma não sofre rotação, e largura da caixa, quando a mesma sofre rotação no posicionamento.

Quando não há rotação da caixa:

$$\begin{aligned}H_L &= H_C \\L_L &= L_{B1} \\W_L &= W_C - W_{B1}\end{aligned}$$

No qual:

$H_L$  - Altura do subespaço L;

$L_L$  - Comprimento do subespaço L;

$W_L$  - Largura do subespaço L;

$L_{B1}$  - Comprimento da primeira caixa;

$W_{B1}$  - Largura da primeira caixa.

Quando há rotação no posicionamento da caixa, a altura do subespaço permanece o mesmo, porém a largura e o comprimento são obtidos conforme abaixo:

$$\begin{aligned}L_L &= W_{B1} \\W_L &= W_C - L_{B1}\end{aligned}$$

c) O subespaço R é obtido da mesma forma como na decomposição L, M e R. A altura é a altura do *container*, a largura é a largura do *container*. O comprimento do subespaço R é dado pela diferença entre o comprimento do *container* e o comprimento da primeira caixa, quando a mesma não sofre rotação, quando a caixa sofre rotação, o comprimento de R é a diferença entre o comprimento do *container* e a largura da caixa.

Quando não há rotação no posicionamento da primeira caixa:

$$\begin{aligned}
 H_R &= H_C \\
 L_R &= L_C - L_{B1} \\
 W_R &= W_C
 \end{aligned}$$

No qual:

$H_R$  - Altura do subespaço R;

$L_R$  - Comprimento do subespaço R;

$W_R$  - Largura do subespaço R;

$L_C$  - Comprimento do *container*.

Quando há rotação no posicionamento da primeira caixa, a largura do subespaço R é dado por:

$$L_R = L_C - W_{B1}$$

### 3.6.3.2 Preenchimento dos Subespaços

O preenchimento dos subespaços na seqüência M, L e R é semelhante ao preenchimento L, M e R.

Considerando a seqüência dos genes (índice de caixas) e a seqüência dos subespaços, avalia-se se a caixa pode ser posicionada dentro do subespaço. Quando a caixa é selecionada, a mesma é incluída no padrão de carregamento e o espaço disponível do subespaço em questão é atualizado.

Enfatiza-se que primeiramente são selecionadas caixas para o subespaço M para o preenchimento do mesmo e formação da torre. Quando o subespaço M é totalmente preenchido ou quando não há mais caixas que possam ser posicionadas no mesmo, seleciona-se uma caixa para o subespaço L e determina-se um novo subespaço M, que em outras palavras também significa iniciar a formação de uma nova torre. O processo é repetido até que todo o subespaço L esteja totalmente

preenchido ou que não haja mais caixas que possam preencher o mesmo. Após isso, seleciona-se uma caixa para ser posicionada no subespaço R, que determinará novos subespaços L e M.

O posicionamento da caixa nos subespaços continua privilegiando o posicionamento da menor dimensão da caixa (comprimento) em paralelo ao comprimento do *container* e a maior dimensão (largura) em paralelo a largura do *container*, contudo, quando isso não é possível, o posicionamento da caixa com rotação de 90° é permitido, desde que as restrições dimensionais dos subespaços sejam obedecidas.

#### **3.6.3.2.1 O processo de seleção das caixas para os subespaços**

O processo de seleção das caixas é similar aos processos descrito anteriormente. Devido à formação de torres, novas restrições precisam ser consideradas na seleção da caixa.

Tomando como base algumas dessas características mencionadas por GEHRING e BORTFELDT (2001), as torres devem ser formadas respeitando-se as restrições das caixas com relação às dimensões físicas, isto é, a caixa posicionada em cima de outra não pode apresentar área de base (superfície inferior da caixa) maior que a mesma área da caixa posicionada abaixo, e essa restrição deve ser considerada na seleção de cada nova caixa que será posicionada, de forma a propiciar a formação de uma torre, e não somente no posicionamento em cima da caixa base, que é a primeira caixa da torre. Além disso, individualmente, as dimensões paralelas ao comprimento e largura do *container* também devem ser iguais ou menores das da caixa posicionada imediatamente embaixo da mesma. Isso implica em atualizar as dimensões horizontais (comprimento e largura) do espaço disponível dos subespaços

a cada posicionamento de uma nova caixa, e a altura quando aplicado, como no caso do subespaço M.

Os fluxogramas apresentados nas figuras 16 e 17 mostram todo o processo de seleção de caixas, de posicionamento e de atualização dos espaços disponíveis.

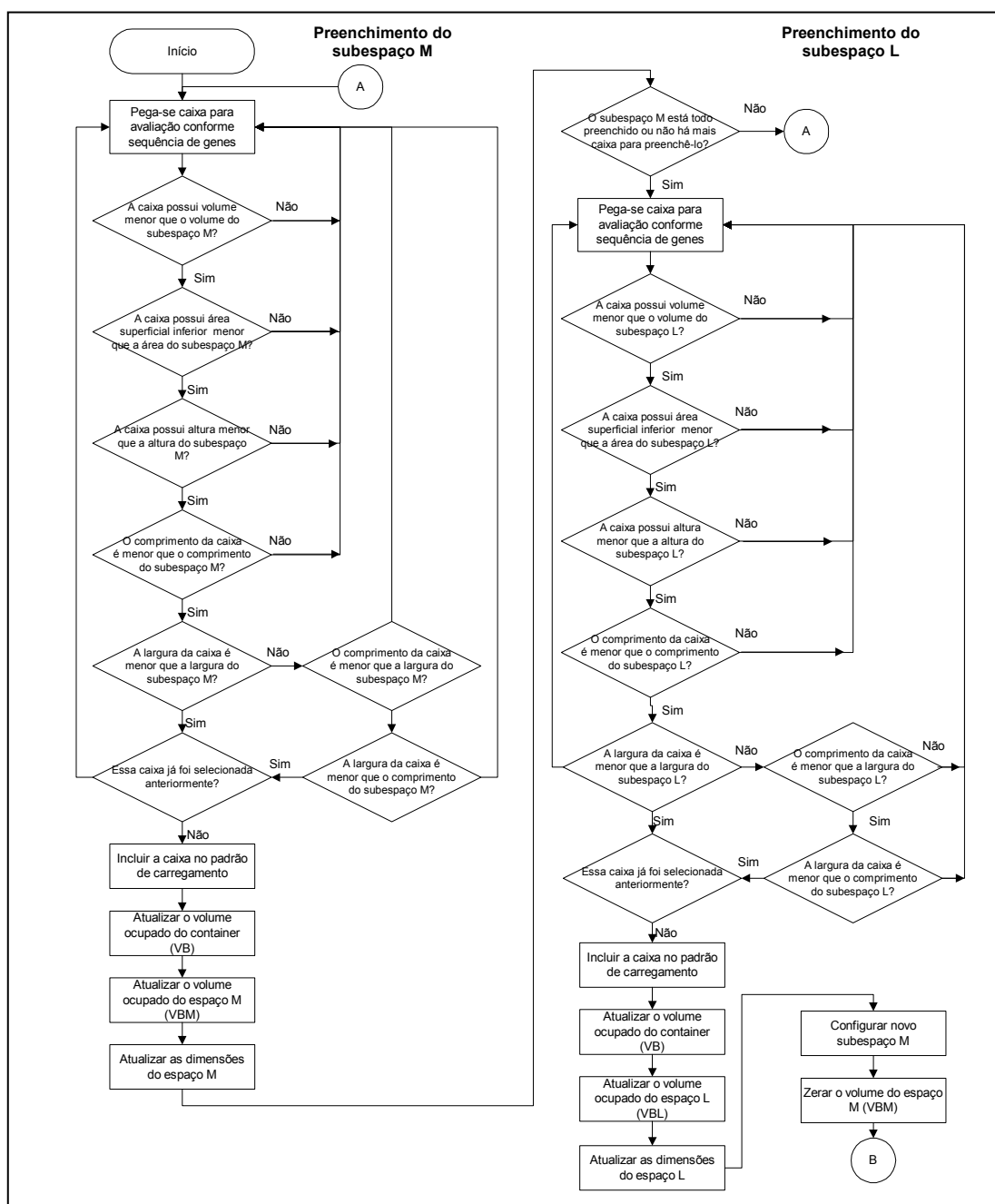


Figura 16 - Processo de seleção de caixas e preenchimento dos subespaços M e L, na seqüência M, L e R.

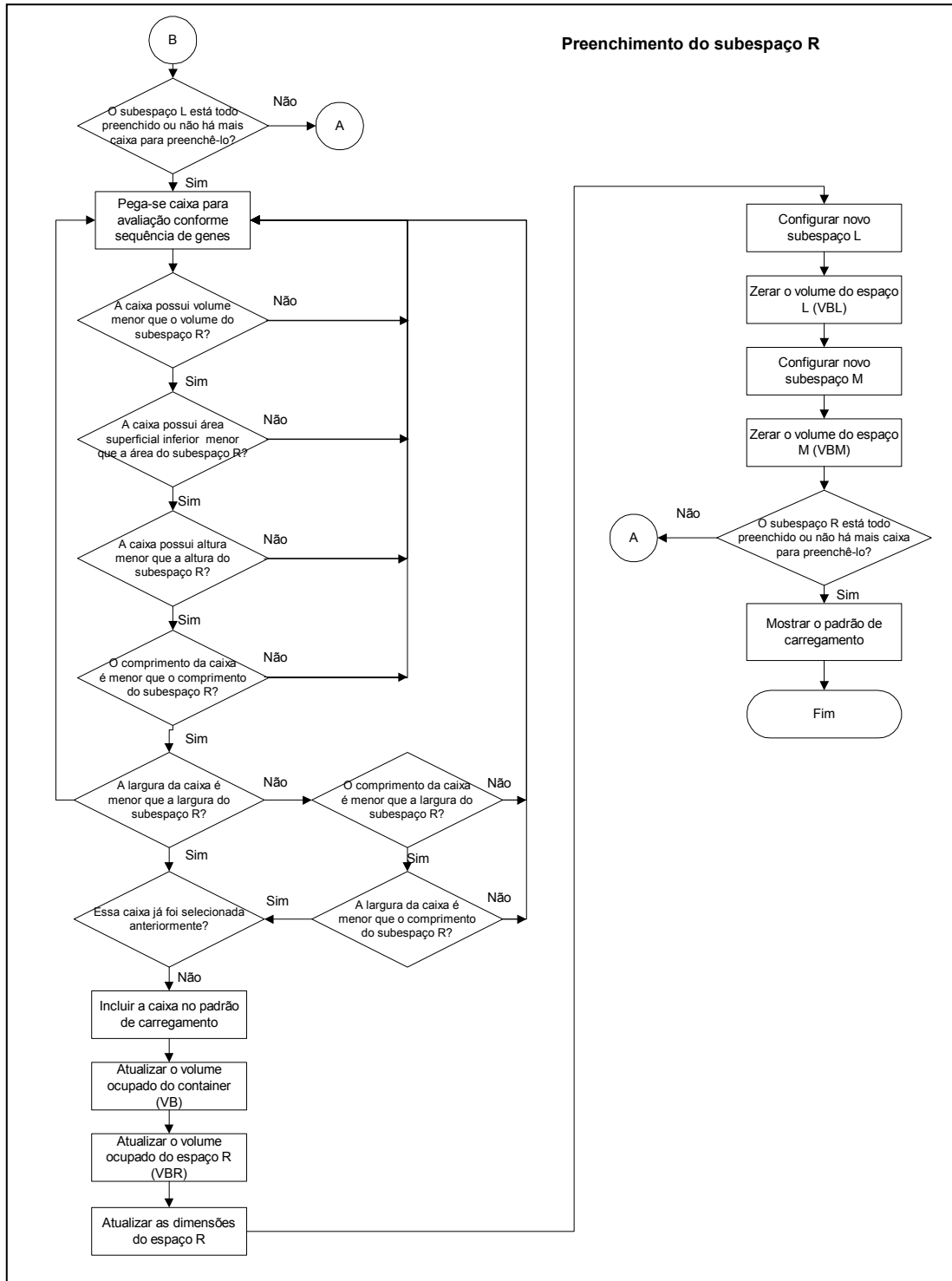


Figura 17 - Processo de seleção de caixas e preenchimento do subespaço R, na seqüência M, L e R.

Como pode ser observado nos fluxogramas acima, os critérios para seleção das caixas são:



1. A caixa deve ter volume ( $V_{Bi}$ ) menor ou igual ao volume do espaço disponível do subespaço ( $V_{SD}$ ).

$$V_{Bi} \leq V_{SD}$$

2. A caixa deve ter área da superfície inferior ( $A_{Bi}$ ) menor ou igual à área de superfície do espaço disponível do subespaço ( $A_{SD}$ ).

$$A_{Bi} \leq A_{SD}$$

3. A caixa deve possuir altura ( $H_{Bi}$ ) igual ou menor a altura do espaço disponível do subespaço ( $H_{SD}$ ).

$$H_{Bi} \leq H_{SD}$$

4. A caixa deve possuir comprimento (menor dimensão horizontal) ( $L_{Bi}$ ) menor que o comprimento do espaço disponível do subespaço ( $L_{SD}$ ).

$$L_{Bi} \leq L_{SD}$$

5. A caixa deve possuir largura (maior dimensão horizontal) ( $W_{Bi}$ ) menor que a largura do espaço disponível do subespaço ( $W_{SD}$ ).

$$W_{Bi} \leq W_{SD}$$

6. Caso a caixa não possua largura ( $W_{Bi}$ ) menor que a largura do espaço disponível do subespaço ( $W_{SD}$ ), a caixa deve possuir comprimento ( $L_{Bi}$ ) menor que a largura do subespaço ( $W_{SD}$ ) e largura menor que o comprimento do subespaço ( $W_{Bi}$ ) menor que o comprimento do subespaço.

$$L_{Bi} \leq W_{SD}$$

$$W_{Bi} \leq L_{SD}$$

7. A mesma caixa não pode ser selecionada mais de uma vez.

Lembra-se que inicialmente o espaço disponível é todo o subespaço.

Quando a caixa contempla todos os critérios acima, a caixa é selecionada e incluída no padrão de carregamento.

As restrições dimensionais para a formação da torre estão inclusas nos critérios acima. Essas restrições são consideradas nos cálculos das dimensões do espaço disponível dos subespaços, como poderá ser observado abaixo.

### 3.6.3.2.2 Preenchimento do Subespaço M

O subespaço M é o primeiro espaço a ser preenchido. A seleção das caixas segue a seqüência de genes e a seqüência de avaliações descritas acima. A cada seleção de caixas para o subespaço M, a torre de caixas vai sendo formada e as dimensões do espaço disponível para novas caixas devem ser atualizadas.

Na atualização das dimensões, a altura do espaço disponível será a altura anterior do espaço disponível menos a altura da nova caixa selecionada. O comprimento do espaço disponível será o comprimento da caixa selecionada e a largura será a largura da caixa selecionada, quando não ocorre rotação no posicionamento da caixa. Caso ocorra rotação da caixa, comprimento do espaço disponível do subespaço será a largura da caixa selecionada e a largura, o comprimento da caixa selecionada.

A forma como a atualização se apresenta atende a restrição da formação torre, a qual a área de base e as dimensões horizontais (largura e comprimento) da caixa selecionada deve ser menor ou igual a caixa imediatamente abaixo.

Para caixa que não sofreu rotação:

$$H_{MD}(i) = H_{MD}(i - 1) - H_B(j)$$

$$W_{MD}(i) = W_B(j)$$

$$L_{MD}(i) = L_B(j)$$

Com rotação, a obtenção da altura permanece da mesma forma, porém a largura e comprimento passam a ser:

$$W_{MD}(i) = L_B(j)$$

$$L_{MD}(i) = W_B(j)$$

Sendo,  $i$  o índice da caixa selecionada no subespaço e  $j$  o índice geral da caixa.  $H_{MD}$  é a altura do espaço disponível de  $M$ ,  $L_{MD}$  é o comprimento do espaço disponível de  $M$  e  $W_{MD}$  é a largura disponível do subespaço  $M$ .

### 3.6.3.2.3 Preenchimento do Subespaço L

Quando não há mais caixas que possam ser selecionadas para o subespaço  $M$ , passa-se a selecionar caixas para o subespaço  $L$ . Os critérios para a seleção da caixa continua sendo os mesmos, conforme descrito acima.

Quando da seleção de uma caixa, o espaço disponível do subespaço deve ser atualizado. A altura do espaço disponível continua a mesma, o comprimento continua o mesmo, porém a largura do espaço disponível passa a ser a diferença da largura anterior para a dimensão da caixa que está paralela a largura do *container* (largura, caso não haja rotação ou comprimento, caso a caixa sofra rotação no seu posicionamento).

Atualização das dimensões do espaço disponível do subespaço  $L$ , quando a caixa não sofreu rotação:

$$H_{LD}(i) = H_{LD}(i - 1)$$

$$W_{LD}(i) = W_{LD}(i - 1) - W_B(j)$$

$$L_{LD}(i) = L_{LD}(i - 1)$$

No caso de rotação da caixa no seu posicionamento, somente a largura do espaço disponível será calculada de outra forma:

$$W_{LD}(i) = W_B(i - 1) - L_B(j)$$

Sendo  $i$  o índice da caixa selecionada no subespaço e  $j$  o índice geral da caixa.  $H_{LD}$  é a altura do espaço disponível de L,  $L_{LD}$  é o comprimento do espaço disponível de L e  $W_{LD}$  é a largura disponível de L.

Além da atualização do subespaço L, é criado um novo subespaço M, cujas dimensões são calculadas da mesma forma quando do posicionamento da primeira caixa. Esse novo subespaço deve ser preenchido, até que não haja mais caixas compatíveis com o espaço, sempre seguindo a seqüência dos genes e após o complemento do mesmo, nova caixa deverá ser selecionada para o subespaço L até que o referente subespaço esteja totalmente preenchido ou que não haja mais caixas.

#### 3.6.3.2.4 Preenchimento do Subespaço R

Quando o subespaço L e M estão completos ou quando não há mais caixas que possam ser posicionadas nos mesmos, passa-se para a seleção de caixa para posicionamento no subespaço frontal R.

Todos os critérios já mencionados acima devem ser contemplados na seleção da caixa.

Quando uma caixa é selecionada e é incluída no padrão de carregamento, o espaço disponível do subespaço R diminui, já que o comprimento do mesmo passa a ser a diferença entre o comprimento anterior e a dimensão horizontal da caixa posicionada em paralelo ao comprimento do *container*.

Para o caso em que a caixa não sofre rotação, tem-se:

$$H_{RD}(i) = H_{RD}(i - 1)$$

$$W_{RD}(i) = W_{RD}(i - 1)$$

$$L_{LD}(i) = L_{LD}(i) - L_B(j)$$

Quando a caixa posicionada sofreu rotação de  $90^\circ$ , o comprimento do espaço disponível passa a ser:

$$L_{LD}(i) = L_{LD}(i) - W_B(j)$$

Novos subespaços L e M são criados, da mesma forma como foram criados quando do posicionamento da primeira caixa, sendo necessário que se repitam os processos de preenchimento dos subespaços M, L e R, respectivamente, até que não haja mais espaço disponível ou que não haja mais caixa compatível com os espaços.

## 4. TESTES E RESULTADOS

O algoritmo genético apresentado no capítulo anterior, com seus dois modelos de preenchimento (L, M e R e M, L e R) foi testado para avaliação de desempenho do preenchimento, considerando a função de aptidão total, e para que comparações entre os dois modelos de preenchimento pudessem ser feitas.

Em um primeiro momento, foram realizados testes com as mesmas caixas apresentadas por HE, CHA (2002), mantendo a mesma quantidade e dimensões de caixas. Às mesmas foram atribuídos valores aleatórios. Para esse teste, também se utilizou a mesma dimensão do *container* utilizado por HE, CHA (2002).

Numa segunda etapa, foram realizados testes com dimensões de caixas reais de produtos eletroeletrônicos, com quantidade e tipo de caixas semelhantes ao problema real. Nestes testes com dimensões reais de caixas, também foram avaliados diferentes ajustes, como número de gerações e de probabilidade de mutação (pm), para que pudessem ser determinadas as melhores condições de ajustes para o algoritmo genético.

O algoritmo e seus dois modelos de preenchimento foram desenvolvidos e testados em Matlab 5.3 e os testes foram realizados em PC Intel Pentium III (1000 MHz) / 261.100 kB de memória RAM, com sistema operacional Windows 2000.

#### 4.1 Testes com dimensões de caixas utilizadas por HE, CHA (2002)

Conforme citado acima, os primeiros testes realizados com o algoritmo desenvolvido utilizaram o mesmo número de caixas e suas respectivas dimensões do trabalho desenvolvido por HE, CHA (2002), portanto, 50 caixas, cujas dimensões são descritas na tabela 2. O número de indivíduos por população de cromossomos utilizados no teste foi de 40, o número de gerações foi de 200, parâmetros iguais aos de HE, CHA (2002). Os pesos respectivos a cada sub-função de aptidão, volume (R), peso (W), centro de gravidade (G) e valor (V), foram  $k_1= 6$ ,  $k_2= 1$ ,  $k_3= 1$  e  $k_4= 2$ . A definição dos pesos foi feita conforme a definição de prioridade para os objetivos do problema de carregamento na empresa onde o mesmo foi estudado, da mesma forma como foi feito nos outros testes. No trabalho de HE, CHA (2002) não são descritos os pesos utilizados.

As dimensões do *container* utilizadas no teste foram:

altura = 2.352 m;

largura = 2.388 m;

comprimento = 5.899 m;

carga máxima = 18070.00 Kg;

valor máximo da carga = R\$ 10,000.00.

A probabilidade de mutação pm utilizada foi de 0.05.

O valor máximo de carga apresentado acima foi acrescentado aos dados originais do exemplo de HE, CHA (2002), cujo valor foi atribuído aleatoriamente, bem como os valores respectivos a cada caixa, apresentados nas tabelas 2 e 3, que nesses primeiros testes foram os mesmos para todas as caixas.

Tabela 2 - Dimensões das caixas (1 a 40) de HE, CHA (2002).

Id	Max (cm)	Mid (cm)	Min (cm)	Vol (cm <sup>3</sup> )	Weight (kg)	Valor (R\$)
1	239	235	120	6739800	3675.25	100.00
2	180	235	100	4230000	2306.64	100.00
3	128	235	139	4181120	2279.99	100.00
4	163	200	120	3912000	2133.23	100.00
5	111	235	139	3625815	1977.17	100.00
6	120	235	111	3130200	1706.91	100.00
7	119	122	111	1611498	878.75	100.00
8	76	168	120	1532160	835.49	100.00
9	120	119	100	1428000	999.60	100.00
10	139	111	90	1388610	972.02	100.00
11	145	111	79	1271505	890.05	100.00
12	120	119	85	1213800	849.66	100.00
13	120	115	80	1104000	772.80	100.00
14	139	115	68	1086980	760.88	100.00
15	145	111	60	965700	675.99	100.00
16	139	110	60	917400	642.18	100.00
17	59	130	100	767000	418.24	100.00
18	139	90	60	750600	525.42	100.00
19	120	119	50	714000	499.80	100.00
20	163	35	120	684600	373.21	100.00
21	139	70	68	661640	198.49	100.00
22	113	111	51	639693	191.90	100.00
23	51	113	111	639693	348.82	100.00
24	76	67	120	611040	333.20	100.00
25	100	100	60	600000	180.00	100.00
26	120	115	40	552000	165.60	100.00
27	111	72	68	543456	163.03	100.00
28	68	72	111	543456	296.34	100.00
29	120	100	43	516000	154.80	100.00
30	139	68	50	472600	141.78	100.00
31	120	120	30	432000	129.60	100.00
32	120	70	50	420000	126.00	100.00
33	100	100	40	400000	120.00	100.00
34	120	65	40	312000	93.60	100.00
35	76	67	60	305520	91.65	100.00
36	76	67	60	305520	91.65	100.00
37	120	50	50	300000	90.00	100.00
38	139	60	35	291900	87.57	100.00
39	111	68	37	279276	223.42	100.00
40	68	37	111	279276	152.29	100.00



Tabela 3 - Dimensões das caixas (41 a 50) de HE, CHA (2002).

Id	Max (cm)	Mid (cm)	Min (cm)	Vol (cm <sup>3</sup> )	Weight (kg)	Valor (R\$)
41	70	64	59	264320	211.45	100.00
42	59	64	70	264320	144.13	100.00
43	120	55	40	264000	211.20	100.00
44	93	80	35	260400	208.32	100.00
45	80	70	35	196000	156.80	100.00
46	105	59	30	185850	148.68	100.00
47	59	105	30	185850	101.34	100.00
48	70	59	41	169330	135.46	100.00
49	59	41	70	169330	92.33	100.00
50	100	45	35	157500	126.00	100.00

#### 4.1.1 Testes do Algoritmo Genético com o Preenchimento L, M e R

Foi realizada uma seqüência de execuções do algoritmo genético com o preenchimento L, M e R, onde os resultados obtidos podem ser observados na tabela 4.

Tabela 4 - Testes do GA com preenchimento L, M e R, com caixas utilizadas na tabela 2 e 3.

Seq.	Função Aptidão - $\pi$ (%)	R (%)	W (%)	G (%)	V (%)
1	73.1499	79.0498	78.4584	124.7416	27
2	72.5780	78.5337	79.4431	125.1342	25
3	72.4260	78.2430	76.6128	122.1897	28
4	72.8667	77.4324	75.3616	122.7116	33
5	73.1029	77.5897	82.9691	124.5215	29
6	72.4188	78.2152	80.7190	122.1783	26
7	74.3242	80.1490	78.4853	121.8630	31
8	72.1649	76.8683	79.9443	124.4955	28
9	71.8409	76.0902	76.1542	123.7141	31
10	72.5307	76.4762	78.0752	124.3746	32
11	72.4009	77.6069	76.2245	122.1429	30
12	73.6237	79.6334	79.8613	124.5749	27
13	73.0439	78.2776	78.3335	122.4395	30
14	71.5697	75.1646	76.6837	124.0249	32
15	73.9744	79.3015	81.2100	124.7250	29
16	72.1924	75.6129	75.6703	124.5760	34
17	73.5412	79.4814	79.9287	124.5951	27
18	73.9451	79.6737	77.2751	122.1332	31
19	75.3623	79.6082	81.6376	124.3366	35
20	73.6270	78.7369	77.3274	124.5214	31
21	73.5207	78.0333	76.0090	124.9981	33
22	71.6645	76.7162	75.7392	122.6079	29

A média dos resultados da função de aptidão total e das sub-funções de aptidão do volume (R), peso (W), centro de gravidade (G) e o valor (V), bem como, os desvios padrões são apresentados na tabela 5.

Tabela 5 - Média e desvio padrão da função de aptidão total e das sub-funções de aptidão do preenchimento L, M e R com as caixas da tabela 2 e 3.

	Função Aptidão - $\bar{x}$ (%)	R (%)	W (%)	G (%)	V (%)
Média	72.99403636	78.0224591	78.2783318	123.709073	29.9090909
Desv. Pad.	0.93914152	1.41329685	2.1841932	1.14989229	2.67099216

Na tabela 6, observa-se o melhor resultado obtido nas seqüências dos testes realizados.

Tabela 6 - Demonstrativo do melhor resultado obtido no preenchimento L, M e R com as caixas da tabela 2 e 3.

	Função de Aptidão - $\bar{x}$ (%)	R (%)	W (%)	G (%)	V (%)
Melhor Resultado	75.3623	79.6082	81.6376	124.3366	35

Observa-se nos resultados dos testes que a média da função de aptidão foi de 72.994%, com um desvio padrão de 0.939%, chegando a obter um resultado de 75.362%.

Especificamente com relação à taxa de ocupação do volume disponível, avaliado pela sub-função de aptidão R, que é a sub-função de maior peso associado, observa-se uma taxa de ocupação média de 78.022%, com um desvio padrão de 1.413%, conseguindo alcançar, como um melhor resultado, uma taxa de ocupação de 80.149% do volume disponível. O peso da carga, por sua vez, obteve uma média de 78.278% do peso máximo de carregamento, a sub-função de aptidão do centro de gravidade obteve uma média de 123.709% e o valor alcançou uma média de 29.909% do valor máximo permitido pela carga.

Pode ser observado na figura 18, um gráfico exemplificando a evolução propiciada pelo algoritmo genético, conforme a função de aptidão total.

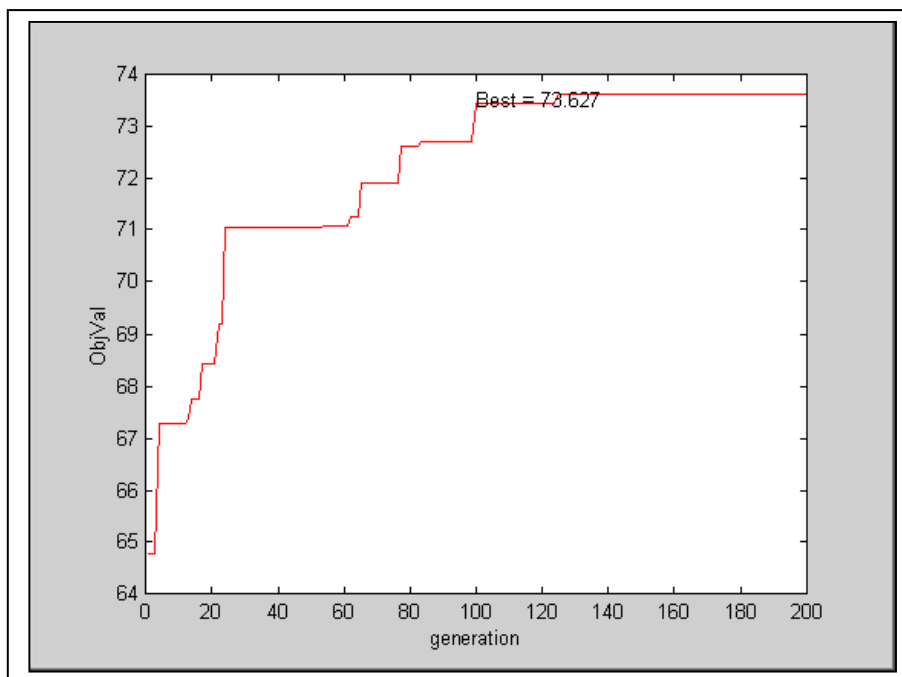


Figura 18 - Gráfico demonstrativo da evolução do algoritmo genético

Comparando numericamente o resultado apresentado acima aos resultados do algoritmo genético desenvolvido por HE, CHA (2002), observa-se uma inferioridade nos valores encontrados para a função de aptidão total, cujo melhor valor foi de 97.48%, no algoritmo de HE, CHA (2002), contra um valor médio do algoritmo aqui descrito de 72.994% e pico de 75.362%, e com uma taxa de ocupação do volume máxima de 95.32% contra uma taxa média determinada de 78.022% e pico de 80.149%. Contudo, o algoritmo genético com preenchimento L, M e R, descrito nesse trabalho, apresenta um item a mais para avaliação que é a sub-função de aptidão do valor e, acrescenta-se, as restrições com relação a rotações das caixas durante o seu posicionamento, cuja idéia seria simular as condições reais das cargas utilizadas, onde existem restrições de rotações das cargas, com direcionamento específico de empilhamento. O que se quer dizer aqui é que, no algoritmo genético desenvolvido por HE, CHA (2002), para determinação do padrão de carregamento, as caixas poderiam

sofrer rotações em qualquer um dos seus eixos, o que permite um número ainda maior de combinações de resultados possíveis e, por conseqüência, maior possibilidade de preenchimento dos espaços disponíveis.

Soma-se a característica descrita acima o fato do trabalho de HE, CHA (2002) não apresentar os valores dos pesos associados às sub-funções e não descrever com detalhes o preenchimento dos subespaços, o que dificulta uma comparação dos resultados do algoritmo genético proposto neste trabalho e do algoritmo proposto por HE, CHA (2002).

#### 4.1.2 Testes do Algoritmo Genético com o Preenchimento M, L e R

Os resultados dos testes do algoritmo genético com o preenchimento M, L e R são apresentados na tabela 7.

Tabela 7 - Testes do GA com preenchimento M, L e R, com caixas utilizadas na tabela 2 e 3.

Seq.	Função de Aptidão - $\pi$ (%)	R (%)	W (%)	G (%)	V (%)
1	70.2543	74.0793	76.3818	123.6850	29.0000
2	71.1380	76.4147	79.2494	121.6421	26.0000
3	71.1380	76.4147	79.2494	121.6421	26.0000
4	71.5349	77.0041	77.9459	121.3788	27.0000
5	70.8548	76.4563	76.0905	121.7197	26.0000
6	71.0803	76.3619	77.0300	121.6012	27.0000
7	69.4809	73.9627	75.1749	121.8579	27.0000
8	71.1681	76.1795	76.5155	122.0881	28.0000
9	71.9736	77.8773	77.3912	121.0807	27.0000
10	71.5850	76.9149	76.7866	121.5743	28.0000
11	69.9857	75.5304	78.3996	124.2751	22.0000
12	71.6914	76.8908	77.2545	122.3151	28.0000
13	72.3457	78.3180	77.9596	121.5892	27.0000
14	71.2920	75.1037	73.4068	122.8907	33.0000
15	69.9871	73.5841	71.8127	124.5534	31.0000
16	69.7539	74.4290	77.3540	121.6114	26.0000
17	70.8069	76.1603	75.7498	121.3569	27.0000
18	71.4618	75.6645	76.9533	123.6778	30.0000
19	68.6574	71.7230	72.8438	121.3925	31.0000
20	71.1836	75.9730	76.0999	121.8982	29.0000
21	70.4998	75.4388	76.2196	124.1456	26.0000
22	71.2529	76.6731	75.0159	121.4739	28.0000

A média dos resultados da função de aptidão total para o preenchimento M, L e R, bem como, das sub-funções de aptidão do volume (R), peso (W), centro de gravidade (G) e o valor (V), e os desvios padrões são apresentados na tabela 8, a seguir.

Tabela 8 - Média e desvio padrão da função de aptidão total e das sub-funções de aptidão do preenchimento M, L e R com as caixas da tabela 2 e 3.

	Função de Aptidão - $\varpi$ (%)	R (%)	W (%)	G (%)	V (%)
Média	70.8694	75.7797	76.4039	122.2477	27.6818
Desv. Pad.	0.879024	1.504786	1.893022	1.087153	2.275885

O melhor resultado obtido no preenchimento M, L e R, para os testes descritos acima, é apresentado na tabela 9.

Tabela 9 - Demonstrativo do melhor resultado obtido no preenchimento M, L e R com as caixas da tabela 2 e 3.

	Função de Aptidão - $\varpi$ (%)	R (%)	W (%)	G (%)	V (%)
Melhor Resultado	72.3457	78.318	77.9596	121.5892	27

A média obtida dos resultados da função de aptidão total ( $\varpi$ ) do algoritmo genético foi de 70.8694%, com um desvio padrão de 0.879%, obtendo como melhor resultado, 72.3457%. Especificamente, a taxa de ocupação do volume, avaliada pela sub-função de aptidão (R), obteve uma média de 75.7797% do espaço disponível, conseguindo obter uma ocupação máxima de 78.318%. As médias dos resultados para as demais sub-funções de aptidão, peso (W), centro de gravidade (G) e valor (V), respectivamente, foram 76.4039%, 122.2477% e 27.6818%.

#### 4.1.3 Comparação dos Resultados entre os Dois Modelos de Carregamento

Quando comparados os resultados obtidos para os dois tipos de carregamento, observa-se que o preenchimento L, M e R apresenta resultados ligeiramente melhores, com relação à função de aptidão total, comprovados através de teste estatístico de diferença de médias (teste-z para duas amostras), onde o resultado pode ser observado na tabela 10, cujos dados foram extraídos do *software* estatístico (SSS\_Stat) utilizado para a execução do referente teste. Na referente tabela, observa-se que a probabilidade de significância (p) é menor que o valor referente da região crítica (R), 2.698, para um nível de significância de 0.01.

Tabela 10 - Dados e Resultado do teste estatístico de diferenças de média entre os preenchimentos L,M e R e M, L e R para o teste com 50 caixas.

	<b>L-M-R</b>	<b>M-L-R</b>	<b>Dif. (X1-X2)</b>
No. De Itens	22	22	0
Média	72.99403636	70.8694	2.12463636
Desvio Padrão:n	0.93914152	0.879024	0.06011752
Estatística	7.747119501		
Grau de Liberdade	42		
<b>0.5-% ponto</b>	<b>2.698070602</b>		
2.5-% ponto	2.018082341		
<b>Valor P</b>	<b>6.32804E-10</b>		
Marca de decisão	[**]		

A média do preenchimento L, M e R foi de 72.994% enquanto que o preenchimento M, L e R obteve uma média de 70.8694%. O melhor resultado da função de aptidão total do preenchimento L, M e R foi de 75.3623% contra 72.3457% do preenchimento M, L e R. Comparando-se a taxa de preenchimento do volume disponível, também se observa que o preenchimento L, M e R apresenta resultados melhores: a média do preenchimento L, M e R foi de 78.022% e o preenchimento M, L e R foi de 75.7797%. Comparando os resultados da taxa de ocupação do volume para os melhores padrões de carregamento entre os dois modelos de preenchimento,

observa-se também um resultado ligeiramente superior do preenchimento L, M e R, cujo valor foi de 79.6082% contra 78.318% do preenchimento M, L e R.

A diferença encontrada nos valores das funções objetivos e das sub-funções de aptidão dos dois modelos, para as caixas disponíveis para o carregamento, conforme tabela 2, pode ser explicado pela combinação de dois aspectos importantes que os diferenciam: a obtenção dos subespaços de preenchimento e o método de preenchimento. Os subespaços ou decomposições do espaço do *container* podem ser observados na figura 19. No preenchimento, L, M e R, como o nome já sugere, o primeiro espaço a ser preenchido é o subespaço L, seguido dos subespaços M e R; já no preenchimento M, L e R, o preenchimento inicia-se pelo subespaço M, seguido dos subespaços L e R, além disso, como pode ser observado na figura 19, os subespaços L e M entre os dois modelos de carregamento são diferentes.

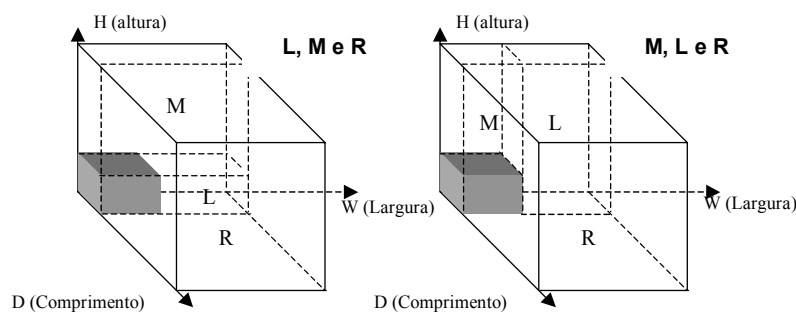


Figura 19 - Ilustração das decomposições de espaço L, M e R e M, L e R.

Comparando-se a forma de obtenção dos subespaços L, no preenchimento L, M e R e do subespaço M, no preenchimento M, L e R, pode-se dizer que as metodologias são similares, sendo que uma determina um subespaço no sentido horizontal e o outro no sentido vertical. Contudo, quando se compara o método de seleção das caixas para os preenchimentos dos respectivos subespaços, observa-se uma significativa diferença. A seleção das caixas para preenchimento do subespaço L no preenchimento L, M e R, se dá somente pela comparação entre as dimensões da

caixa e as dimensões do espaço disponível, se as dimensões da caixa forem iguais ou menores que as do espaço L disponível a mesma é selecionada. No preenchimento do subespaço M, no método M, L e R, além das dimensões da caixa terem que atender a restrição das dimensões do espaço disponível, as dimensões horizontais, comprimento e largura devem ser tais que quando posicionada a caixa, a caixa abaixo deve possuir dimensões posicionadas em paralelo aos eixos de comprimento e largura do *container* iguais ou maiores que da caixa acima, para garantir o empilhamento correto das caixas.

Para essas caixas disponíveis para o carregamento, de acordo com suas dimensões e quantidades disponíveis, conforme as diferenças de obtenção dos espaços e do preenchimento, o modelo L, M e R foi favorecido em relação ao modelo M, L e R.

O tempo médio de execução do algoritmo genético nos dois modelos de preenchimento para os testes com 50 caixas e 200 gerações foi de 7.5 minutos.

## **4.2 Testes com dimensões de caixas reais do problema estudado**

Nos testes desenvolvidos com dimensões reais de caixas do problema foram utilizadas configurações de cargas disponíveis proporcionalmente similares ao problema real, onde se apresenta vários tipos de caixas, que representam as embalagens dos modelos de aparelho, e em quantidade proporcional a produção diária dos mesmos. Para os testes também se considerou a variedade de modelos que são disponibilizados para carregamento.



#### 4.2.1 Testes com 100 caixas de Áudio

Na primeira seqüência de testes com caixas da situação real, foram utilizadas 100 caixas de modelos de áudio, numa configuração de carga de um mesmo tipo de produto, onde as caixas apresentam similaridades em suas dimensões e peso.

A tabela 11 apresenta as dimensões das caixas e as quantidades de tipo de caixa (produto), peso, os valores respectivos e os índices para cada modelo de aparelho.

Tabela 11 - Dados dimensionais, pesos e valores das caixas do teste com 100 caixas de áudio.

Modelo	Altura (cm)	Compr. (cm)	Largura (cm)	Volume (cm <sup>3</sup> )	Peso Bruto (Kg)	Valor (R\$)	Qtd	Índice	
								Início	Final
A	47.7	46	36	78991.2	14.1	1253.00	20	1	20
B	41.9	56.3	35.8	84451.1	9.6	626.00	25	21	45
C	41.9	56.3	35.8	84451.1	10	704.00	20	46	65
D	41.9	56.3	35.8	84451.1	10.6	791.00	15	66	80
E	41.9	56.3	35.8	84451.1	10.6	1018.00	10	81	90
F	45.6	47.3	38.2	82392.8	13.7	1488.00	5	91	95
G	44.5	44.1	33.5	65742.1	10.7	1096.00	5	96	100

As dimensões do *container* utilizado nos testes são, respectivamente, de 135 cm de altura, 500 cm de largura e 108 cm de comprimento (*container* estreito). O peso máximo de carregamento é de 10.000 Kg, o valor máximo das cargas no *container* é de R\$ 150000.00.

Por sua vez, os parâmetros do algoritmo genético utilizados foram, respectivamente, tamanho da população igual a 100 indivíduos, número de gerações igual a 200 e probabilidade de mutação igual a 0.1. Os pesos respectivos a cada sub-função de aptidão foram  $k_1 = 7$ ,  $k_2 = 0.5$ ,  $k_3 = 0.5$  e  $k_4 = 2$ , respectivamente, sub-função de adequação do volume, do peso, do centro de gravidade e do valor.

Os resultados do algoritmo genético com o preenchimento L, M e R são apresentados na tabela 12.

Tabela 12 - Resultados dos testes com caixas de áudio com o preenchimento L, M e R.

Seq.	Função de Aptidão - $\pi$ (%)	R (%)	W (%)	G (%)	V (%)
1	73.4930	82.2991	8.1870	133.7038	43.9453
2	73.3589	82.4170	8.0210	133.8637	42.8640
3	73.5127	82.3740	8.1500	133.7374	43.7827
4	73.5684	82.2991	8.2070	133.7057	44.3167
5	73.5684	82.2991	8.2070	133.7057	44.3167
6	73.5512	82.4489	8.1390	133.7736	43.7067
7	73.5684	82.2991	8.2070	133.7057	44.3167
8	73.5684	82.2991	8.2070	133.7057	44.3167
9	73.5684	82.2991	8.2070	133.7057	44.3167
10	73.5684	82.2991	8.2070	133.7057	44.3167

Na tabela 13, verifica-se as médias para a função de aptidão total e para as sub-funções de aptidão respectivas, volume (R), peso (W), centro de gravidade (G) e valor (V) para o preenchimento L, M e R.

Tabela 13 - Média e desvio padrão dos valores dos testes da tabela 12 (preenchimento L, M e R).

	Função de Aptidão - $\pi$ (%)	R (%)	W (%)	G (%)	V (%)
Média	73.5326	82.3334	8.1739	133.7313	44.0199
Desv. Padrão	0.0668096	0.0579396	0.0595473	0.0516878	0.4742912

O melhor resultado nos respectivos testes é apresentado na tabela 14.

Tabela 14 - Demonstrativo do melhor resultado obtido no preenchimento L, M e R nos testes realizados conforme a tabela 12.

	Função de Aptidão - $\pi$ (%)	R (%)	W (%)	G (%)	V (%)
Melhor Resultado	73.5684	82.2991	8.2070	133.7057	44.3167

Como observado na tabela acima, a média para a função de aptidão total nos testes desenvolvidos com o preenchimento L, M e R, foi de 73.5326%, com um desvio

padrão de 0.066% e pico de 73.5684%. A taxa de ocupação média do volume, verificada pela função de aptidão R, foi de 82.3334%, com um desvio padrão de 0.057%.

Os testes realizados para as mesmas caixas da tabela 11, com o mesmo *container*, porém com o preenchimento M, L e R tem seus resultados apresentados na tabela 15.

Tabela 15 - Resultados dos testes com caixas de áudio com o preenchimento M, L e R.

Seq.	Função de Aptidão - $\varpi$ (%)	R (%)	W (%)	G (%)	V (%)
1	51.7263	56.6143	50.5400	134.3616	25.6273
2	51.7685	56.6143	5.0600	134.3618	25.8367
3	51.7685	56.6143	5.0600	134.3618	25.8367
4	51.7685	56.6143	5.0600	134.3618	25.8367
5	51.6854	56.6143	5.0500	134.3615	25.4240
6	51.7685	56.6143	5.0600	134.3618	25.8367
7	51.6854	56.6143	5.0500	134.3615	25.4240
8	51.7579	56.6143	5.0560	134.3617	25.7847
9	51.7488	56.6143	5.0640	134.3619	25.7373
10	51.4795	56.3577	5.0330	134.3406	25.3020

A média dos resultados da função de aptidão total  $\varpi$  e os desvios padrões das suas respectivas funções de aptidão são apresentadas na tabela 16 e o melhor resultado é apresentado na tabela 17.

Tabela 16 - Média e desvio padrão dos valores dos testes da tabela 15 (preenchimento M, L e R).

	Função de Aptidão - $\varpi$ (%)	R (%)	W (%)	G (%)	V (%)
Média	51.7157	56.5886	9.6033	134.3596	25.6646
Desv. Padrão	0.0893	0.0811	14.3837	0.0067	0.2073

Tabela 17 - Demonstrativo do melhor resultado obtido no preenchimento M, L e R nos testes realizados conforme a tabela 15.

	Função de Aptidão - $\omega$ (%)	R (%)	W (%)	G (%)	V (%)
Melhor Resultado	51.7685	56.6143	5.0600	134.3618	25.8367

No preenchimento M, L e R, a média dos resultados dos testes para a função de aptidão total foi de 51.7157% com um desvio padrão de 0.0893%. A sub-função de aptidão de ocupação do volume R apresentou uma média de 56.5886% de preenchimento do espaço disponível e um desvio padrão de 0.0811%. O melhor resultado apresentado na seqüência de teste foi de 51.7685% e 56.6143%, para a função de aptidão total e sub-função de aptidão do volume, respectivamente.

De maneira bem evidente, comparando-se os resultados dos modelos de preenchimento, o preenchimento L, M e R apresenta um preenchimento com valores superiores em relação ao preenchimento M, L e R. De fato, os valores encontrados para o preenchimento M, L e R, com a média de apenas 51.7157% para a função de aptidão total e com a taxa de preenchimento do volume disponível do *container* de 56.5886% são valores inferiores aos encontrados com o preenchimento L, M e R, que apresentou como resultados médios, 73.536% para a função de aptidão total e 82.334% para a taxa de ocupação do volume.

A diferença entre os resultados obtidos nos testes dos dois modelos de preenchimento, conforme as caixas disponíveis para a formação do padrão de carregamento, se dá pelos mesmos motivos apresentados nos testes com as caixas de HE, CHA (2002). A diferença na obtenção dos subespaços L, M e R e a restrição adicional ao preenchimento no modelo M, L e R, que para a formação da torre, restringe a seleção da caixa à compatibilidade das dimensões horizontais da mesma caixa, que deve ser menor ou igual às respectivas dimensões da caixa que está logo abaixo da caixa selecionada, além das restrições em relação às dimensões do

subespaço em questão, subespaço M. Certamente, a combinação das caixas disponíveis para os testes acima não favoreciam a formação de torres, através do preenchimento M, L e R.

O resultado obtido na ocupação do preenchimento L, M e R é compatível com o resultado prático obtido no carregamento na empresa, onde os dados das caixas e o modelo adaptado foram extraídos, cuja média da taxa de volume preenchido no início do desenvolvimento do referente trabalho estava em 84.5%, conforme pode ser observado no gráfico demonstrativo na figura 20. Salientando, ainda, que a heurística utilizada na referente empresa não seguia padrões rígidos, como os dois modelos aqui apresentados. Em outras palavras, se necessário para ocupação de espaços disponíveis, especialmente em espaços superiores, as caixas são posicionadas acima de duas caixas ou de duas camadas de preenchimento, não seguindo nenhuma das regras de posicionamento que são adotadas nos dois modelos de preenchimento.

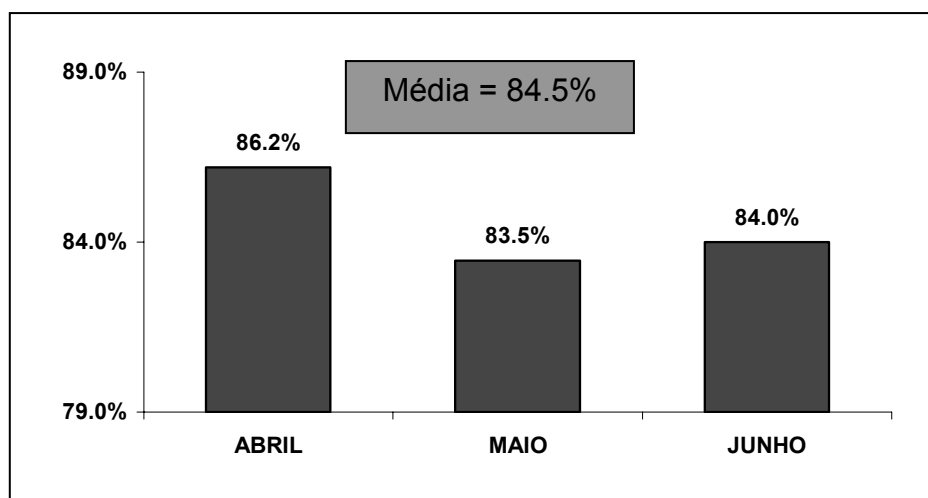


Figura 20 - Demonstrativo do preenchimento médio mensal do espaço de *containers*

O tempo médio de execução do algoritmo genético para os testes com 100 caixas e 200 gerações foi de 35 minutos para o preenchimento L, M e R e 45 minutos para o preenchimento M, L e R.

#### 4.2.2 Testes com caixas mistas

Os referentes testes foram feitos com uma configuração de carga semelhante à configuração real de cargas. A carga é formada por caixas de diferentes modelos de áudio, DVD e vídeo, que é uma situação semelhante ao problema original.

A tabela 18 apresenta as caixas, respectivas a cada modelo, suas dimensões, a quantidade de cada modelo disponível para carregamento e os valores associados.

Tabela 18 - Dados dimensionais, pesos e valores das caixas do teste com 100 caixas de Áudio, TV e DVD.

Modelo	Altura (cm)	Compr. (cm)	Largura (cm)	Volume (cm <sup>3</sup> )	Peso Bruto (Kg)	Valor (R\$)	Qty	Índice	
								Início	Final
A	18.3	50.0	38.2	34953.00	4.2	548	55	1	55
B	15.6	50.0	38.2	29796.00	3.9	508	30	56	85
C	69.6	59.3	86.9	358660.63	53.0	1410	68	86	153
D	69.6	57.6	89.8	360004.61	82.0	3135	26	154	179
E	41.9	56.3	35.8	84451.126	9.6	626	45	180	224
F	41.9	56.3	35.8	84451.126	10.6	791	43	225	267
G	41.9	56.3	35.8	84451.126	10.6	791	18	268	285

Como já mencionado, a carga acima é formada por diferentes tipos de produtos eletro-eletrônicos. Os modelos A e B são modelos de DVD, os modelos C e D são aparelhos de televisão de 29" e 34", respectivamente, os modelos E, F e G são aparelhos de áudio.

A quantidade de caixas disponíveis para carregamento é de 285. As dimensões do *container* são de 2.352 m de altura, 2.388 m de largura e 5.899 m de comprimento. A carga máxima suportada pelo *container* é de 18070 kg e o valor máximo da carga para o *container* é de R\$ 300000.00. Os pesos associados a cada função de aptidão total (volume - R, peso - W, centro de gravidade - G e valor - V) são, respectivamente,  $k_1 = 7$ ,  $k_2 = 0.5$ ,  $k_3 = 0.5$  e  $k_4 = 2$ , determinados conforme a prioridade de objetivos definida pela empresa onde o trabalho foi estudado.

O número de cromossomos (indivíduos) foi de 100, o número de caixas disponíveis, como já citado acima, era de 285 caixas. Contudo, dois parâmetros foram “experimentados” através de um “projeto de experimento” (DOE – *Design of Experiment*) (KIEMELE, SCHMIDT, *et al.*, 1997), número de gerações e probabilidade de mutação, para o qual foram definidos os níveis mínimo e máximo, onde o objetivo era determinar quais seriam os valores mais adequados para os respectivos parâmetros no intuito de maximizar a função de aptidão total. No projeto de experimento supracitado, avalia-se a maximização da função de aptidão total em função da adequação dos parâmetros escolhidos para serem experimentados. Conforme citado acima, o número de cromossomos por população também poderia ser experimentado, contudo, definiu-se por fixar um dos três parâmetros e variar os outros dois para identificar a melhor combinação.

O valor mínimo (-1) estipulado de gerações foi de 250 e o máximo (+1) foi de 350. Os valores mínimo (-1) e máximo (+1) para a probabilidade de mutação foram, respectivamente, de 0.05 e 0.1.

Pelo número de fatores (parâmetros) e número de níveis para cada um dos fatores, utilizou um projeto de experimento com fatorial completo de dois níveis, ou seja, quatro combinações de parâmetros. Para cada combinação foram realizados 5 testes, onde foi possível determinar a média e o desvio padrão para função de aptidão total em cada uma das combinações e, assim, determinar um modelo matemático de previsão que possibilite a determinação dos valores mais adequados para os parâmetros experimentados.

#### 4.2.2.1 Testes com o carregamento L, M e R

Os testes realizados para o carregamento L, M e R com as quatro combinações entre os parâmetros “número de gerações” e “probabilidade de mutação (Pm)” apresentaram os resultados que podem ser observados na tabela 19.

Tabela 19 - Resultado dos testes com combinações dos parâmetros, “número de gerações” e “probabilidade de mutação”, no carregamento L, M e R.

Gerações	Pm	1	2	3	4	5	Média	Desv. Pad
1	-1	-1	66.8373	66.8373	66.8373	66.8373	66.8373	0
2	-1	+1	67.7061	67.7693	67.7693	67.7693	67.7693	0.0282639
3	+1	-1	68.3961	64.0139	68.3961	68.3961	68.3961	1.9597794
4	+1	+1	67.9435	67.9435	67.9435	67.8382	67.9435	0.0470916

Utilizando um *software* estatístico, *SSS\_Stat*, é possível obter através de uma análise de regressão múltipla um modelo matemático entre o resultado da função de aptidão total  $\varpi$  (saída) e os parâmetros que foram combinados (fatores ou entradas).

O modelo obtido é apresentado abaixo:

$\hat{Y} = 67.5090 + 0.2120A + 0.3305B - 0.1291AB$ , onde  $A$  é o número de gerações e  $B$  a probabilidade de mutação (Pm).

Como o objetivo é maximizar a função de aptidão total, os valores de  $A$  e  $B$  devem ser tais que propiciem essa maximização. Quando posicionados os valores  $A$  e  $B$  nos níveis máximos, pelo modelo matemático determinado, é possível obter o valor máximo da função de aptidão total do referente algoritmo genético.

Na figura 20, abaixo, observa-se um gráfico de superfície em que é demonstrada a relação entre os fatores  $A$  e  $B$  e a saída e a interação entre os fatores, onde poderá ser observado que quando os fatores  $A$  e  $B$  estão ajustados em níveis máximos estipulados, pode-se obter o valor máximo para a função de aptidão total.



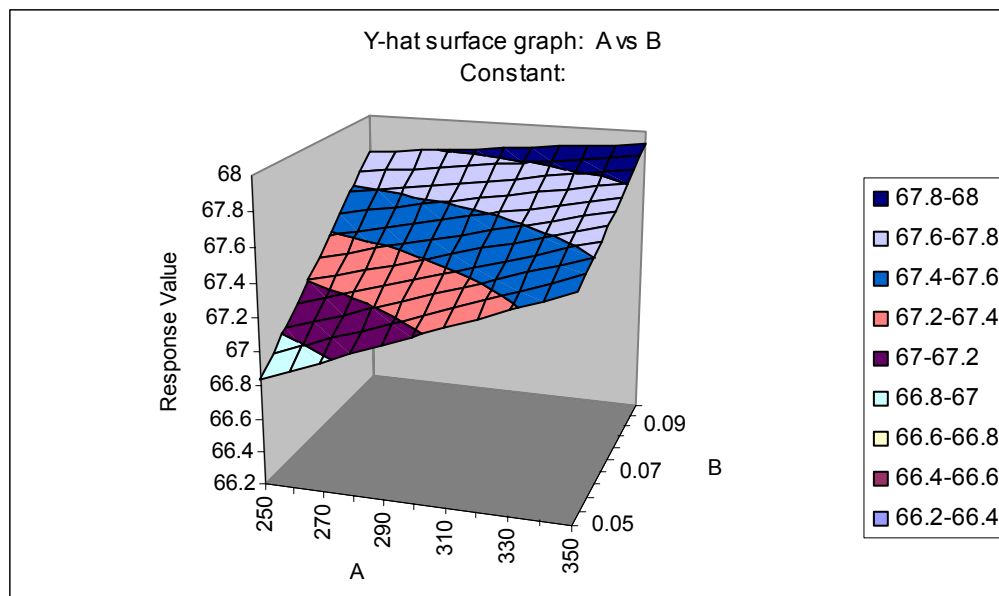


Figura 20 – Gráfico demonstrativo da interação dos fatores *A* e *B* e da relação entre os fatores e a saída para o carregamento L, M e R com 285 caixas.

Conforme o modelo matemático apresentado acima, quando *A* e *B* estão em níveis +1, a função de aptidão total apresenta como resultado 67.9224. Executando novamente os testes com os parâmetros número de gerações (*A*) e probabilidade de mutação (*B*) ajustados, respectivamente, para 350 e 0.1, obteve-se os resultados apresentados na tabela 20.

Tabela 20 - Resultado dos testes com os parâmetros número de gerações e probabilidade de mutação ajustados em seus níveis máximos, para carregamento L, M e R.

Seq.	Função de Aptidão - $\pi$ (%)	R (%)	W (%)	G (%)	V (%)
1	67.9435	70.0805	21.3094	136.5642	54.9677
2	67.9435	70.0805	21.3094	136.5642	54.9677
3	67.9435	70.0805	21.3094	136.5642	54.9677
4	67.8382	70.4395	20.7499	136.6473	53.3033
5	67.9435	70.0805	21.3094	136.5642	54.9677
6	67.9435	70.0805	21.3094	136.5642	54.9677
7	67.4926	70.1522	20.5257	136.6560	52.6347
8	67.9435	70.0805	21.3094	136.5642	54.9677

A média e o desvio padrão obtido para a função de aptidão total e para as funções de aptidão podem ser verificados na tabela 21, abaixo.

Tabela 21 - Média e desvio padrão para a função de aptidão total e para as sub-funções de aptidão do teste com carga mista para o carregamento L, M e R.

	Função de Aptidão - $\pi$ (%)	R (%)	W (%)	G (%)	V (%)
Média	67.8740	70.1343	21.1415	136.5861	54.4680
Desv. Padrão	0.1584429	0.125831	0.3166121	0.0405482	0.942316

Observa-se que a média dos resultados da função de aptidão total foi de 67.8740% com um desvio padrão de 0.158% e um valor máximo de 67.9435% e a taxa de ocupação média do volume disponível do *container* foi de 70.1343% com um desvio padrão de 0.125%. Os demais resultados das funções de aptidão podem ser observados na mesma tabela.

#### 4.2.2.2 Testes com carregamento M, L e R

Para o carregamento M, L e R foi desenvolvido a mesma avaliação realizada para o carregamento L, M e R, ou seja, foram realizadas avaliações para as quatro combinações dos parâmetros “número de gerações” e “probabilidade de mutação”. Os resultados dos experimentos para os parâmetros “número de gerações” e “probabilidade de mutação”, cujos níveis mínimos e máximos são, respectivamente, 250 (-1) e 350 (+1) e 0.05 (-1) e 0.1 (+1), são apresentados na tabela 22.

Tabela 22 - Resultado dos testes com combinações dos parâmetros “número de gerações” e “probabilidade de mutação”, no carregamento M, L e R.

Gerações	Pm	1	2	3	4	5	Média	Desv. Pad.	
1	-1	-1	67.8668	71.3546	72.0799	72.1085	72.1873	71.1194	1.8490548
2	-1	+1	72.0586	72.1003	72.0586	71.3781	72.0253	71.9242	0.3064247
3	+1	-1	72.1873	71.3757	72.1873	72.1873	72.1873	72.0250	0.3629586
4	+1	+1	72.4034	72.1683	71.5922	72.4476	72.1933	72.1610	0.3411478

Utilizando o mesmo *software* estatístico, anteriormente citado, foi possível obter o modelo matemático abaixo:

$\hat{Y} = 71.8074 + 0.2856A + 0.2352B - 0.1672AB$ , onde  $\hat{Y}$  é o valor estimado da função de aptidão total,  $A$  refere-se aos níveis do parâmetro “número de gerações” e  $B$  refere-se a “probabilidade de mutação”.

Para obtenção da maximização da função de aptidão total, conforme modelo matemático determinado, o parâmetro  $A$  (número de gerações) deve estar no nível máximo (+1), 350 gerações, e  $B$  (probabilidade de mutação), por sua vez, também deve estar no nível máximo (+1), 0.1. Para os valores dos respectivos parâmetros, o valor estimado para a função de aptidão total, conforme o modelo, é 72.161%.

A figura 21 apresenta um gráfico de superfície onde podem ser observados os níveis para os parâmetros  $A$  e  $B$ , suas interações e suas relações com o valor da função de aptidão total, conforme os testes apresentados acima.

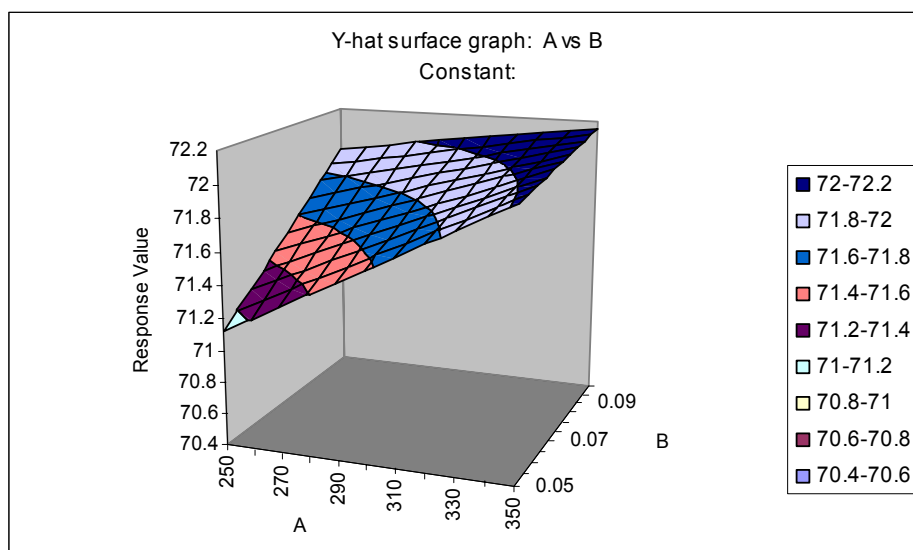


Figura 21 - Gráfico demonstrativo da interação dos fatores  $A$  e  $B$  e da relação entre os fatores (parâmetros) e a saída (função de aptidão total) para o carregamento M, L e R com 285 caixas.

Para os referentes parâmetros, a média e o desvio padrão da função de aptidão total e das funções de aptidão são apresentados na tabela 23.

Tabela 23 - Média e desvio padrão para a função de aptidão total e para as sub-funções de aptidão do teste com carga mista para o carregamento M, L e R.

	Função de Aptidão - $\pi$ (%)	R (%)	W (%)	G (%)	V (%)
Média	72.1610	74.6817	22.2702	136.8053	59.6501
Desv. Padrão	0.341147817	0.519207702	0.452372549	0.132953345	1.650042088

Observa-se que para os testes com cargas mistas, num total de 285 caixas disponíveis, com o método de preenchimento M, L e R, a média obtida para a função de aptidão total foi de 72.1610%, com um desvio padrão de 0.341%, e um melhor resultado, nos testes realizados, de 72.4476%. A taxa de ocupação do volume disponível foi de 74.6817%, com um desvio padrão de 0.519%.

#### 4.2.2.3 Comparação dos Resultados entre os Dois Modelos de Carregamento para Cargas Mistas de Áudio, Vídeo e TV

Diferentemente do que aconteceu nos testes com outros tipos e quantidades de caixas, a configuração aqui utilizada, com sete grupos diferentes de caixas (de dimensões, pesos e valores diferentes), em quantidades variadas, favoreceu o carregamento através do modelo de preenchimento M, L e R, que através de sua formação de torres apresentou resultados superiores ao modelo L, M e R. Como pode ser observado, a média da função de aptidão total para o preenchimento L, M e R foi de 67.8740%, enquanto que para o preenchimento M, L e R, a média foi de 72.1610%. A taxa de ocupação para o preenchimento L, M e R foi de 70.1343%, enquanto que para o preenchimento M, L e R a média foi de 74.6817%. A superioridade nos resultados apresentados pelo preenchimento M, L e R em relação ao preenchimento L, M e R, pode ser comprovada através de teste estatístico (teste-z para duas amostras), conforme pode ser observado na tabela 24, retirada do *software* SSS\_Stat.

Tabela 24 - Dados e Resultado do teste estatístico de diferenças de média entre os preenchimentos L,M e R e M, L e R para o teste com 285 caixas.

	<b>L-M-R</b>	<b>M-L-R</b>	<b>Dif. (X1-X2)</b>
No. de Itens	8	8	0
Média	67.874	72.161	-4.287
Desvio Padrão:n	0.1584429	0.341147817	-0.182704917
Estatística	32.23604585		
Grau de Liberdade	14		
<b>0.5-% ponto</b>	<b>2.976848918</b>		
2.5-% ponto	2.144788596		
<b>Valor P</b>	<b>7.72799E-15</b>		
Marca de decisão	[**]		

Para os preenchimentos os testes com 350 gerações e probabilidade de mutação de 0.1, o tempo médio de execução dos algoritmos foi de aproximadamente 6.16 horas no modelo de preenchimento L, M e R e 6.5 horas no modelo M, L e R.

## 5. CONCLUSÃO

### 5.1 Conclusão

Conforme o objetivo proposto para o referente trabalho, pode-se confirmar que algoritmos genéticos são eficientes para solucionar problemas de carregamento, especialmente, quando se tem um problema de carregamento complexo e com múltiplas restrições. Técnicas clássicas de otimização e modelos matemáticos, que se tornam bastantes complexos para esse tipo de problema, não conseguem ser utilizados para obtenção de resultados satisfatórios em tempo hábil.

Comparando os dois modelos de carregamento de preenchimento do algoritmo genético desenvolvido, observou-se que o modelo L, M e R se mostrou bastante eficiente nas configurações de caixas disponíveis utilizadas no trabalho de HE, CHA (2002) e nos testes com 100 caixas de produtos de áudio, cujas dimensões eram similares, contudo nos testes com cargas mistas de 285 caixas disponíveis, o preenchimento M, L e R apresentou melhores resultados na função de aptidão total e na taxa de ocupação do volume disponível e nas demais sub-funções de aptidão em relação ao preenchimento L, M e R.

Percebe-se, portanto, que o resultado dos dois modelos de preenchimento é influenciado pelas dimensões e pelas quantidades de caixas disponíveis para a

formação do padrão de carregamento, bem como pelas dimensões do *container*. O que leva a concluir-se que dependendo das características e da quantidade das caixas, um modelo pode ser mais adequado que o outro.

As diferenças na obtenção e preenchimento dos subespaços, e as diferenças de restrições na seleção das caixas são os fatores que diferenciam significativamente os dois tipos de preenchimento, e aliados às características das caixas influenciam no resultado final do padrão de carregamento obtido.

O relato acima demonstra que a utilização dos dois modelos como objeto de comparação para obtenção da melhor configuração de cargas pode ser adotado a fim de se obter o padrão de carregamento mais adequado, através da maximização da função de aptidão total.

Como se observa, algoritmos genéticos são técnicas eficientes e que se adaptam adequadamente a problemas complexos de carregamento, com diversas restrições e diferentes configurações de caixas, ainda que os tempos sejam altos, superior a 6 horas, para um número de gerações elevado e com maior quantidade de caixas, como no teste com cargas mistas, cuja quantidade de carga disponível era de 285 e o número de gerações era de 350.

Por fim, comprovou-se que duas metodologias de soluções distintas puderam ser unificadas fazendo uma alteração no processo de obtenção dos subespaços e na ordem de preenchimento dos mesmos, acrescentando uma restrição específica para a seleção de caixas no preenchimento M, L e R que permitisse a formação de torres, conforme proposto.

## **5.2 Sugestão para Trabalhos Subseqüentes**

Os trabalhos subseqüentes a este podem acrescentar outras características do problema real, como, a disponibilidade e carregamento de *containers* em paralelo, um

número maior de caixas, uma modelagem que determine as caixas que estariam sendo disponibilizadas a cada instante. O desenvolvimento do algoritmo genético com essas novas características direciona a uma solução para um problema de carregamento mais próximo as características reais do problema estudado.

Ao invés de tratar de maneira genérica as restrições quanto aos posicionamentos das caixas, um trabalho subsequente poderia, também, estudar restrições específicas para cada tipo de caixa e produto, flexibilizando as restrições de posicionamento de caixas, rotação e equilíbrio, principalmente na formação de torres, o que poderia direcionar o algoritmo genético para resultados ainda melhores, com relação à função de aptidão total e as sub-funções de aptidão.

Os operadores de cruzamento e de mutação poderiam ter seus mecanismos associados a taxas adaptativas, bem como, o operador de seleção poderia ser de forma elitista, selecionando como maior parte da nova população cromossomos com maior valor associado à função de aptidão, enquanto mantém alguns cromossomos com valores inferiores associados à função de aptidão.

Como a disponibilidade de produtos para o carregamento está associada a planos de produção, cujas alterações diárias não são tão significativas, uma outra sugestão seria a determinação de população inicial conhecida a partir de uma configuração de carga anteriormente utilizada cujo resultado tivesse sido satisfatório, o que poderia direcionar para resultados ainda melhores e até poder-se-ia estudar a possibilidade de utilizar um menor número de gerações no algoritmo genético, diminuindo sensivelmente os tempos computacionais.

Por fim, poderia ser feito um estudo para determinação de soluções tratando o problema como um problema de múltiplos objetivos, restringindo a dois objetivos principais, para que a solução seja factível, possibilitando a determinação da “Fronteira Eficiente (Pareto-ótima)”, que poderia ser utilizada como ferramenta auxiliar no processo decisório prático do carregamento de *container*. Existem algoritmos evolucionários dirigidos especificamente aos problemas multi-objetivo (DEB, 2001). A



determinação dos possíveis resultados (diferentes pontos na fronteira eficiente) poderia ser feita dentro do próprio algoritmo genético.

## 6. REFERÊNCIAS BIBLIOGRÁFICAS

BAILER-JONES, Daniela M., BAILER-JONES, Coryn A. L.. "Modelling Data: Analogies in Neural Networks, Simulated Annealing and Genetic Algorithms". In: *Model-Based Reasoning: Science, Technology, Values*, L. Magnani and N. Nersessian (eds). New York, Kluwer Academic/ Plenum Publishers, 2002.

CHEN, C. S., LEE, S. M., SHEN, Q. S.. "An Analytical Model for the Container Loading Problem", *European Journal of Operational Research*, v. 80, pp. 68-76, 1995.

CHIEN, C. F., WU, W. T.. "A Recursive Computational Procedure for Container Loading Problem", *Computers Industry Engineering*, v. 35, n. 1-2, pp. 319-322, 1998.

COFFMAN, E. G., JOHNSON, D. S., SHOR, P. W., et al.. "Bin Packing with Discrete Item Sizes, Part II: Tight Bounds on First Fit". *Random Structures and Algorithms*, v. 10, pp. 69-110, 1997.

CSIRIK, Janos, JOHNSON, David S., KENYON, Claire, et al.. "A Self Organizing Bin Packing Heuristic", *Algorithm Engineering and Experimentation*, pp. 246-265, 1999.

DEB, Kalyanmoy. *Multi-Objective Optimization Using Evolutionary Algorithms*. 1 ed. Chichester, John Wiley & Sons, 2001.

ESPEJO, Luis Gonzalo Acosta, GALVÃO, Roberto Diéguez. “Uma Aproximação da Fronteira Eficiente para um Problema de Localização Hierárquico de Máxima Cobertura”, *Pesquisa Operacional*, v. 24, n. 2, pp. 303-321, 2004.

FAINA, Loris. “A Global Optimization Algorithm for Three-Dimensional Packing Problem”, *European Journal of Operational Research*, v. 126, pp. 340-354, 2000.

GEHRING, H.. BORTFELDT A.. “A Genetic Algorithm for Solving the Container Loading Problem”, *European Journal of Operational Research*, v. 131, pp. 143-161, 2001.

GLOVER, F.. “Tabu Search, Part I”. *ORSA Journal on Computing*, v.1, pp. 190-206, 1989.

GLOVER, F.. “Tabu Search, Part II”. *ORSA Journal on Computing*, v.2, pp. 4-32, 1990.

HE, D. Y., CHA, J. Z.. “Research on Solution to Complex Container Loading Problem Based on Genetic Algorithm”. In: *Proceeding of First International Conference on Machine Learning and Cybernetics*, v. 1, pp. 78-82, Beijing, November, 2002.

HOPPER, E., TURTON, B.. “Application of Genetic Algorithms to Packing Problems – A Review”, In: *Proceedings of the 2<sup>nd</sup> On-line World Conference on Soft Computing in Engineering Design and Manufacturing*, pp. 279-288, London, 1997.

JAKOBS, Stefan. "On Genetic Algorithms for the Packing of Polygons", *European Journal of Operational Research*, v. 88, pp. 165-181, 1996.

KIEMELE, Mark J., SCHMIDT, Stephen, BERDINE, Ronald. *Basic Statistics*. 4 ed. Colorado, Air Academy Press, 1997.

KIRKPATRICK, S., GELATT, C. D., VECCHI, M. P.. "Optimisation by Simulated Annealing", *Science*, v. 220, pp. 671-680, 1983.

LI, Han-Lin, TSAI, Jung-Fa, HU, Nian-Ze. "A distributed global optimization method for packing problems". *Journal-of-the-Operational-Research-Society*, v. 54, pp. 419-425, 2003.

LITVINENKO, Vladimir I., BURGHER, J. A., et al. "The Application of the Distributed Genetic Algorithm to the Decision of the Packing in Containers Problem", In: *Proceeding of the 2002 IEEE International Conference on Artificial Intelligence Systems (ICAIS'02)*, v.1, Divnomorskoe, September, 2002.

MARQUES, Fabiano do Prado, ARENALES, Marcos Nereu. "O problema da Mochila Compartimentada e Aplicações", *Pesquisa Operacional*, n. 3, v. 22, pp. 285-304, 2002.

MICHALEWICZ, Zbigniew. *Genetic Algorithms + Data Structure = Evolution Programs*. 2 ed. New York, Springer-Verlag, 1994.

RAIDL, Günther R.. "A Weight-Coded Genetic Algorithm for the Multiple Container Packing Problem". In: *Proceeding of the 14<sup>th</sup> ACM Symposium on Applied Computing*, v. 1, pp. 596-603, San Antonio, July, 1999.

RODRIGUES, Rosiane de Freitas. *Complexidade Computacional – NP-Completo*.

ICE – Departamento da Ciência da Computação, 2000.

SOUZA, Marcone Jamilson Freitas. *Inteligência Computacional para Otimização*.

ICEB, Universidade Federal de Ouro Preto, 2002.