ANDERSON OLIVEIRA FARIA

OTIMIZAÇÃO DO PROBLEMA DE CORTE E EMPACOTAMENTO UNIDIMENSIONAL UTILIZANDO ALGORITMO GENÉTICO

Monografia apresentada ao Departamento de Ciência da Computação da Universidade Federal de Lavras como parte das exigências do Curso de Ciência da Computação para obtenção do título de Bacharel em Ciência da Computação.

Lavras
Minas Gerais - Brasil
2006

ANDERSON OLIVEIRA FARIA

OTIMIZAÇÃO DO PROBLEMA DE CORTE E EMPACOTAMENTO UNIDIMENSIONAL UTILIZANDO ALGORITMO GENÉTICO

Monografia apresentada ao Departamento de Ciência da Computação da Universidade Federal de Lavras como parte das exigências do Curso de Ciência da Computação para obtenção do título de Bacharel em Ciência da Computação.

Área de Concentração:

Otimização Combinatória

Orientador:

Prof. Guilherme Bastos Alvarenga

Lavras
Minas Gerais - Brasil
2006

Ficha Catalográfica preparada pela Divisão de Processos Técnico

da Biblioteca Central da UFLA

Faria, Anderson Oliveira

Otimização do Problema de Corte e Empacotamento Unidimensional Utilizando Algoritmo Genético / Anderson Oliveira Faria. Lavras – Minas Gerais, 2006. XXp.

Monografia de Graduação – Universidade Federal de Lavras. Departamento de Ciência da Computação.

1. Informática. 2. Otimização Combinatória. 3. Algoritmos Genéticos. I. FARIA, A. O. II. Universidade Federal de Lavras. III. Otimização do Problema de Corte e Empacotamento Unidimensional Utilizando Algoritmo Genético.

ANDERSON OLIVEIRA FARIA

OTIMIZAÇÃO DO PROBLEMA DE CORTE E EMPACOTAMENTO UNIDIMENSIONAL UTILIZANDO ALGORITMO GENÉTICO

Monografia de graduação apresentada ao Departamento de Ciência da Computação da Universidade Federal de Lavras como parte das exigências do Curso de Ciência da Computação para obtenção do título de Bacharel em Ciência da Computação.

APROVADA em *27 de Abril de 2006*

Prof. Luciano Mendes dos Santos

Prof. Reginaldo Ferreira de Souza

Prof. Guilherme Bastos Alvarenga
(Orientador)

LAVRAS
MINAS GERAIS – BRASIL

DEDICATÓRIA

Dedico este trabalho à minha família: Maria José de Oliveira Faria, Dagilberto José de Faria e Alisson José de Oliveira Faria, pelo total apoio e confiança que tiveram em mim durante toda esta caminhada. E a Deus, meu muito obrigado por mais esta realização.

AGRADECIMENTOS

Agradeço a todos os professores que passaram pela minha vida.

Agradeço a todos meus amigos, tanto aos que estão distantes devido a situações que a dinamicidade da vida nos coloca, quanto os que puderam acompanhar de perto minha caminhada, a todos os amigos que passaram na república ao longo do tempo e a todos os grandes novos amigos que pude conhecer ao longo destes quatro anos, em especial a toda minha turma.

Enfim, agradeço a todos que de alguma forma torceram pelo meu sucesso durante esta jornada.

“Nossas dúvidas são traidoras e nos fazem perder o bem que poderíamos conquistar se não fosse o medo de tentar...”

(Willian Shakespeare)

RESUMO

Problemas de Corte e Empacotamento (*Bin Packing Problem*) buscam determinar um arranjo ótimo de unidades menores dentro de unidades maiores, obedecendo a certas restrições, e sempre visando minimizar custos, seja maximizando a ocupação de espaços ou minimizando desperdícios. Assim, o grande interesse no estudo destes problemas foi maximizar espaços, com o objetivo de sempre utilizar o menor número possível de unidades maiores. Visto que este problema é *NP-Difícil*, alguns métodos heurísticos têm sido propostos para resolvê-lo. Os métodos heurísticos encontram uma boa solução melhorando uma solução inicial através de técnicas de pesquisas em vizinhanças. Dentre os métodos heurísticos mais usados para tratar este problema pode-se citar o Algoritmo Genético. Este trabalho implementa um algoritmo genético para construção de um sistema que possa ser utilizado em instâncias dentro da realidade de algumas indústrias, tais como indústrias de papel, de vidro, metalúrgica, plástica e têxtil.

Palavras chave: Otimização Combinatória, Problema de Corte e Empacotamento, Algoritmos Genéticos.

ABSTRACT

Bin Packing Problems are related with to obtain an optimal arrangement of smaller units inside of bigger units, obeying certain constrains, and always aiming at to minimize costs, either by maximizing occupation of spaces or minimizing wastefulness. Thus, the great interest in the study of these problems was to maximize spaces, with the objective to always use the lesser possible number of bigger units. Since Bin Packing Problem is NP-Hard, heuristic methods have been considered to solve it. Heuristic methods find a good solution by improving an initial solution through search techniques in neighborhoods. Amongst the heuristic methods used to deal with this problem the Genetic Algorithm can be cited. This work implements a genetic algorithm for a system construction that can be used to solve real instances of some problems in some industries, such as paper industries, glass industries, metallurgical, plastic and fabric industries.

Key-words: Combinatorial Optimization, Bin Packing Problem, Genetic Algorithm.

SUMÁRIO

LISTA DE FIGURAS

LISTA DE TABELAS

1 INTRODUÇÃO

Esta seção consiste em apresentar as considerações iniciais; os objetivos e justificativas e o escopo para trabalho desenvolvido.

1.1 Considerações Iniciais

No decorrer das últimas décadas vem-se observando uma aplicação crescente de métodos científicos na solução de problemas tecnológicos complexos. O desenvolvimento tecnológico tem contribuído fortemente para a evolução desses métodos, principalmente na área de otimização.

A solução de problemas de elevado nível de complexidade computacional tem sido um desafio constante para os pesquisadores de diversas áreas, tais como, Otimização, Pesquisa Operacional, Ciência da Computação, Matemática e Engenharias, as quais freqüentemente deparam se com problemas altamente combinatórios, cuja solução ótima em muitos casos ainda está limitada somente a pequenas instâncias. Conforme Johnson (Johnson, 1979) este problema é *NP-Difícil* ,o que significa que é pouco provável que exista um algoritmo de otimização em tempo polinomial para ele,

Os métodos exatos tradicionais de otimização se caracterizam pela rigidez de seus modelos matemáticos representados através de seus Teoremas, dificultando a representação de situações reais cada vez mais complexas e dinâmicas. Assim, um dos maiores problemas enfrentados é a dificuldade de inclusão dos modelos objetivos e restrições do mundo real, o que foi um pouco reduzido a partir do momento em que se passou a associar técnicas de Otimização com as ferramentas de busca heurística. Mas o o problema maior dos métodos exatos é o tempo para a parada do algoritmo, que é um tempo muito grande tornado inviável sua espera.

Dentro das heurísticas disponíveis estão os Algoritmos Genéticos (AGs) que são algoritmos probabilísticos muito eficientes para busca de soluções otimizadas em problemas combinatórios. O AG se destaca por permitir com facilidade incluir restrições complexas do mundo real.

Neste contexto, os problemas de corte e empacotamento (*Bin Packing Problem*), tais quais outros de natureza combinatória, podem ser facilmente formulados e compreendidos, escondendo atrás de sua aparente simplicidade, um problema de difícil solução e um grande desafio para a comunidade científica.

O *Bin Packing Problem* consiste basicamente na determinação dos melhores padrões de corte de unidades de material, sendo estes barras, placas, caixas e outros. Utilizando-se dos métodos heurísticos aliados às funções de avaliação específicas da área de otimização, tornou-se possível gerar, combinar e selecionar soluções de modo a gerar soluções próximas da solução ótima, desenvolvendo-se um sistema que permite agilizar e melhorar o aproveitamento no cálculo de encaixe e corte de peças, essenciais para o planejamento da produção em algumas indústrias. Podem tomar-se como exemplo indústrias de papel, de vidro, metalúrgica, plástica e têxtil.

1.2 Objetivos e Justificativas

O objetivo deste trabalho é apresentar uma proposta de solução para Problemas de Cortes (*Bin Packing Problem*) através do Algoritmo Genético, criando novos operadores e aproveitando outros da literatura.

O algoritmo genético foi escolhido pelo fato de possuir bons resultados tanto para problemas de corte e empacotamento (*Bin Packiong Problem*) quanto para outros problemas da classe *NP-difícil*, (GEORGE, J.A. e ROBINSON, D.F. (1980); MORABITO, R. e ARENALES, M. (1994)).

Os Problemas de Corte e Empacotamento são aplicáveis a várias instâncias do mundo real, nos mais diversos ramos industriais, visando sempre uma minimização de custos, sendo este o principal fator de motivação para se obter bons resultados, justificando assim o seu estudo.

1.3 Escopo do Trabalho

Este trabalho descreve uma visão geral dos problemas de cortes na seção 2.4 e apresenta a meta-heurística algoritmo genético (seção 2.6). A seção 3 descreve o que foi desenvolvido neste trabalho, subdividido em:

• Caracterização do Problema (seção 3.2.1);

• População Inicial (seção 3.2.2);

• Seleção (seção 3.2.3);

• Operador de Cruzamento (seção 3.2.4);

• Operadores de Mutação (seção 3.2.5).

• Correção de Indivíduos (seção 3.2.6).

• Elitismo (seção 3.2.7).

Na seção 4 são apresentados os resultados encontrados pelo algoritmo implementado neste trabalho para as instâncias de teste utilizadas.

A conclusão e proposta para futuros trabalhos são encontradas respectivamente nas seções 5 e 6.

2 REFERENCIAL TEÓRICO

Esta seção consiste em apresentar uma base de conhecimento para o tema abordado, e para o entendimento do trabalho desenvolvido. Serão apresentados conceitos de Pesquisa Operacional, conceitos de Otimização e Otimização Combinatória; detalhamento dos Problemas de Corte e Empacotamento, Métodos Heurísticos para solucionar o Problema de Corte e Empacotamento Unidimensional e detalhamento do funcionamento dos Algoritmos Genéticos.

2.1 Pesquisa Operacional

A Pesquisa Operacional nasceu durante a Segunda Guerra Mundial, quando os Aliados se viram confrontados com problemas (de natureza logística e de tática e estratégia militar) de grande dimensão e complexidade (CAMPOS, 98). Para apoiar os comandos operacionais na resolução desses problemas, foram então criados grupos multidisciplinares de cientistas em que se incluíam matemáticos, físicos e engenheiros, a par de outros oriundos das ciências sociais. Esses cientistas não fizeram mais do que aplicar o método científico, que tão bem conheciam, aos problemas que lhes foram sendo colocados. Desenvolveram então a idéia de criar modelos matemáticos, apoiados em dados e fatos, que lhes permitissem perceber os problemas em estudo e ensaiar e avaliar o resultado hipotético de estratégias ou decisões alternativas.

O sucesso e credibilidade ganhos durante a guerra foram tão grandes que, terminado o conflito, esses grupos de cientistas e a sua nova metodologia de abordagem dos problemas se transferiram para as empresas que, com o "boom" econômico que se seguiu, se viram também confrontadas com problemas de decisão de grande complexidade. Fizeram se desde então grandes desenvolvimentos técnicos e metodológicos que hoje, com o apoio de meios computacionais de crescente capacidade e disseminação, permitem trabalhar com enormes volumes de dados sobre as atividades das empresas e, através de adequados modelos de base quantitativa, ensaiar e avaliar linhas de ação alternativas e encontrar as soluções que melhor servem os objetivos dos indivíduos ou organizações.

Pesquisa Operacional é um método científico de tomada de decisões. Trata-se de uma ciência aplicada voltada para a resolução de problemas reais, em que se procura trazer para o campo da tomada de decisões (sobre a concepção, o planejamento ou a operação de sistemas) a atitude e os métodos próprios de outras áreas científicas. Através de desenvolvimentos de base quantitativa, a Pesquisa Operacional visa também introduzir elementos de objetividade e racionalidade nos processos de tomada de decisão, sem descuidar, no entanto dos elementos subjetivos e de enquadramento organizacional que caracterizam os problemas (MEDEIROS (1998)).

2.2 Otimização

Um problema de otimização (PO) pode ser um problema de minimização ou de maximização e é especificado por um conjunto de instâncias do problema

Uma instância de um problema de otimização, segundo (PAPADIMITRIOU, 1982) citado por (ANDRADE, C.E.; BATISTA, F.L.C.; TOSO, R.F(2004)), consiste no par (*F*, *c*), onde *F* é um conjunto qualquer, constituído pelos pontos viáveis, e c é uma função de custo, um mapeamento:

$$c : F \mapsto R^1 \tag{2.1}$$

O problema consiste em encontrar um *f* Є F para cada

$$c(f) \leq c(y) \forall y \in F \tag{2.2}$$

Cada ponto *f* é denominado ótimo global para a instância, e é comumente referenciado por ótimo.

Desta forma, um Problema de Otimização estende-se ao conjunto ***I*** de instâncias de um problema de otimização. Os principais constituintes de um problema de otimização são:

- Vizinhança: dado um ponto viável *f* Є *F* num determinado problema com instâncias (*F*, *c*), sua vizinhança consiste no mapeamento:

$$N : F \mapsto 2^F \tag{2.3}$$

definido para cada instância.

- Ótimo Local: em certas instâncias de problemas, encontrar uma solução ótima pode ser uma tarefa impossível do ponto de vista computacional. Nesses casos, há como encontrar uma solução *f*, sendo essa a melhor solução na vizinhança *N*(*f*).

Em uma instância (*F, c*) de um problema de otimização, com vizinhança *N*, a solução viável *f* Є *F* é definida como sendo *ótimo local* em relação a *N* se:

$$c(f) \leq cg) \forall g \in N(f) \qquad (2.4)$$

Na figura 2.1, observando os pontos ótimos assinalados, pode-se ter a clara distinção entre o ótimo local e o ótimo global: os pontos *A*, *B* e *C* são ótimos locais, mas apenas o ponto *B* é o ótimo global.

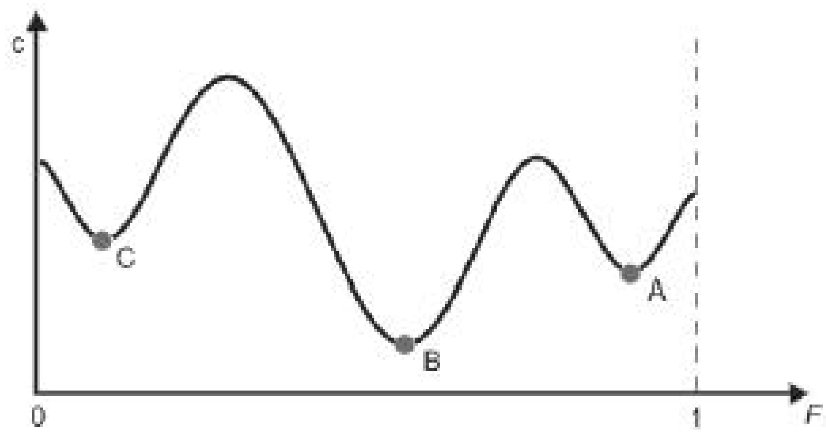

Figura 2.2.1 - Problema Euclidiano Unidimensional

No exemplo, utilizado por (PAPADIMITRIOU, 1982), o problema abordado é conhecido por *1-dimensional Euclidean Problem*, ou simplesmente Problema Euclideano Unidimensional. Considerando a instância (*F, c*) onde $F = [0, 1] \subseteq R^1$ é a função de custos *c* traçada na Figura 2.1, a vizinhança pode ser definida por:

$$N \in (f) = \{x : x \in F\varepsilon \ |x - \mathrm{f}| \leq \varepsilon \} \qquad (2.5)$$

2.2.1 Técnicas Exatas para Solução dos Problemas de Otimização

Segundo fonte de ANDRADE, C.E.; BATISTA, F.L.C.; TOSO, R.F(2004), os métodos de busca por soluções denominados exatos são aqueles que sempre encontram a solução ótima para o problema se é que ela existe. Esta solução deve satisfazer de forma ótima a função objetivo correspondente ao problema em questão, respeitando todos os parâmetros que se aplicam à resolução do problema.

Um problema pode ou não possuir um algoritmo exato para sua solução. Existindo esse algoritmo, o mesmo pode não encontrar um ótimo em tempo hábil, ou seja, o algoritmo pode levar décadas para encontrar a solução desejada. Nestes casos, diz-se que o algoritmo é inviável para a instância em questão (CORMEN et al, 2001).

2.3 Otimização Combinatória

Dentro da área de Pesquisa Operacional, encontra-se a Otimização Combinatória, que trata basicamente do estudo matemático para encontrar um arranjo, agrupamento ou seleção ótima de objetos discretos, tendo sido aplicada extensivamente aos mais diversos ramos do conhecimento, não permitindo, a utilização de métodos clássicos de Otimização Contínua (RAO, 1978).

Problemas de Otimização Combinatória ocorrem em áreas tão diversas como projetos de sistemas de distribuição de energia elétrica, posicionamento de satélites, projetos de computadores e de chips VLSI, roteamento ou escalonamento de veículos, alocação de trabalhadores, máquinas a tarefas, empacotamento de caixas em *containers*, corte de barras e placas, sequenciamento de genes e DNA, classificação de plantas e animais (LUNA & GOLDBARG, 2000) citado por SILVA (2005).

Em RAUPP (2003) o Problema de Otimização Combinatório é definido da seguinte maneira:

"Encontrar um vetor de variáveis de decisão (solução) que satisfaça algumas restrições e otimize uma função objetivo". Formalmente, segundo fonte RAUPP (2003) citado por SILVA (2005), o problema pode ser enunciado como:

Encontrar x para:

$$
\begin{array}{l}
\text{Minimizar}\left(\text{ou Maximizar}\right) \quad f\left(x\right) \\
\text{sujeito a:} \\
\qquad g_i\left(x\right) \geq 0 \quad i = 1, \ldots, m \\
\qquad h\left(x\right) = 0 \quad j = 1, \ldots, p
\end{array}
$$

Onde, $f, g_i\ e\ h_j$, são funções gerais de parâmetros. A variável de decisão x assume valores discretos e pertence ao conjunto de soluções possíveis para uma dada instância de um problema de otimização.

Conforme RAUPP (2003) citado por SILVA (2005), o Problema de Otimização Combinatória pode ser denominado como a ação de maximizar ou minimizar uma função de diversas variáveis sujeita a um conjunto de restrições, dentro de um domínio finito, geralmente um subconjunto de N.

Visando o problema proposto neste trabalho, suponha que um objeto deva ser cortado ao longo de seu comprimento em itens (pedaços) de comprimentos especificados. Cada item possui um valor associado que chamamos de valor de utilidade. Itens cujos comprimentos não foram especificados são considerados perdas e têm valores de utilidade nulos. De acordo com YANASSE (1997) e GRAMANI (2003), surge então um problema de otimização combinatória:

Maximizar VALOR DE UTILIDADE TOTAL OU

Minimizar AS PERDAS.

O problema é achar a melhor forma de cortar o objeto para produzir os itens, de modo que, o valor de utilidade total seja máximo (equivale a dizer que a perda de material seja mínima).

2.4 Problemas de Corte e Empacotamento

Para Vieira (VIEIRA, 99), dentre os problemas clássicos da Otimização Combinatória estão os Problemas de Corte e Empacotamento (*Bin Packing Problem*) extensivamente estudados por serem problemas de difícil solução, onde a possibilidade de existir algoritmos exatos que os resolvam em tempo de execução razoável é muito pequena. Com isso, faz-se necessário o uso de algoritmos heurísticos e aproximativos (ou de aproximação) que buscam encontrar soluções muito próximas da ótima em tempos de execução razoáveis.

Em Otimização Combinatória, é comum encontrar na literatura vários algoritmos, muitos com excelente desempenho, para resolução de determinado problema. Muitos dos algoritmos propostos para um problema particular são aplicáveis à solução de outro problema através de pequenas adaptações. Assim, se tornaria mais simples o estudo e proposição de novos algoritmos se houvesse uma biblioteca que reunisse os problemas mais clássicos com os algoritmos para sua resolução.

Segundo (DYCKHOFF e FINKE, 92), problemas de corte e empacotamento são, em geral, problemas de otimização combinatória que buscam determinar o arranjo ótimo de unidades menores (itens) dentro de unidades maiores (objetos) satisfazendo a determinadas restrições, ou seja, consiste em alocar uma coleção de objetos com tamanhos definidos no menor número de compartimentos possível. A capacidade do compartimento é previamente definida, de forma que nenhum destes seja preenchido além de sua capacidade, assim como nenhum objeto seja fracionado em objetos menores. Dependendo do tipo de objeto (barra, placas, caixas, dentre outros) temos os chamados problemas unidimensional, bidimensional, tridimensional e outros.

Problemas de corte e empacotamento são fundamentais no planejamento da produção de vários segmentos industriais, cujo objetivo é minimizar os efeitos negativos gerados por desperdício de materiais e igualmente relevantes no planejamento de operações logísticas como armazenagem, movimentação e transporte, visando a minimização de espaços ociosos (GOLDEN, 76).

Devido a aspectos principalmente econômicos, o desempenho global de certos sistemas industriais depende, em grande parte, de uma programação efetiva da produção nos processos de corte e empacotamento. Algumas das aplicações de problemas de corte e empacotamento são: alocação de comerciais em TV, alocação de programas em discos e fitas magnéticas, carregamento de veículos, problema da programação de veículos, corte de bobinas (papel, ...), corte de vigas (em madeireiras, construção civil, ...), pré-paginação, corte de placas (vidro, chapas, madeira, compensado, ...), corte de retalhos em fábrica de tecidos, ou em confecção de roupas, empacotamento de caixas em galpões, empacotamento em contêineres, carregamento de cargas em furgões e corte de espumas para colchões.

2.4.1 Tipologia dos Problemas de Corte e Empacotamento

Os Problemas de Corte e Empacotamento podem ser classificados como de tipos diferentes a considerar-se alguns aspectos. Para determinação do tipo de problema que se trata os aspectos básicos a serem focados com respeito à estrutura são (ARENALES, 93):

- Dimensionalidade;
- Tipo de seleção dos objetos/itens;
- Características dos objetos/itens;
- Restrições associadas aos padrões;
- Objetivos;
- Status da informação e variabilidade dos dados;
- Métodos de solução.

As três primeiras características da estrutura lógica são consideradas fundamentais, e são responsáveis pelo esquema que permite relacionar problemas de corte e empacotamento entre si. Entretanto, se levará em consideração neste momento, apenas a dimensionalidade do objeto, o que leva a uma abordagem importante quanto ao tipo de corte do objeto em estudo.

De uma forma geral, os problemas de empacotamento consistem em empacotar uma lista de itens em recipientes. Dependendo da natureza dos itens sendo cortados, estes podem ser ditos unidimensionais, bidimensionais ou tridimensionais. Nas próximas secções seram descritos detalhes destes três tipo de problemas.

2.4.2 Problema Unidimensional

Quando se trata de uma dimensão relevante e fixa, como no caso do corte de barras de ferro (construção civil) apenas o comprimento da barra é relevante na definição de um padrão associado a ela.

Por exemplo, considerando-se a estrutura de concreto armado de um prédio. Mais precisamente, considerando-se as colunas de concreto contendo barras de aço em seu interior. Estas colunas podem ter diferentes comprimentos e consequentemente cada uma delas requer barras com certo comprimento. Em geral, a fábrica que produz estas barras somente as produz com um comprimento específico, supondo 12 metros. Assim, a construtora deve comprar estas barras grandes e cortá-las conforme sua necessidade. Claramente, para minimizar os custos, o seu objetivo é comprar o menor número de barras necessário na construção. Este é um problema muito comum que também aparece em problemas de corte de vigas, estruturas metálicas, canos, esquadrias, etc.

O problema *Bin Packing* clássico, unidimensional, objetiva minimizar o número de barras (*bins*) (regiões bem definidas como: caixa, vagões, trilhas de um disco rígido, containeres, etc.) de igual capacidade, necessários para o acomodamento de uma dada lista de objetos, considerando-se apenas uma dimensão, onde o somatório dos itens contidos em uma barra (*bin*) nunca é superior a capacidade da mesma. Para cada uma das instâncias do problema temos que definir:

- A capacidade c da barra;
- Uma lista de n objetos, L;
- Um número N de barras.

Dessa forma, o problema de decisão é: a lista L pode ser empacotada em N ou menos barras, sendo que todas essas barras possuem capacidade máxima c? Conforme (Johnson, 1979) este problema é NP-completo no sentido forte, o que significa que é pouco provável que exista até mesmo um algoritmo de otimização em tempo pseudo-polinomial para ele.

Neste trabalho, desenvolveu se um Algoritmo Genético considerando o fato que a lista de itens é conhecida a priori. Portanto, o enfoque é voltado para a resolução do problema conhecido como *Bin Packing* 1-D off-line.

2.4.3 Problema Bidimensional

Quando se trata de duas dimensões relevantes e fixas, como no caso do corte de chapas de madeira (confecção de móveis) a largura e o comprimento da chapa são relevantes na definição de um padrão.

Dando continuidade no exemplo da construção do prédio, considere as janelas e divisórias de vidro retangulares, necessárias em quantidade e tamanhos diferentes. A construtora faz o pedido de compra a uma vidraçaria, especificando as dimensões e quantidades de cada item. A vidraçaria deve cortar estes pedaços a partir de placas de vidro de tamanhos fixos, supondo 3 metros por 3 metros, que são fornecidas pelo fabricante de vidro. Novamente, para minimizar os custos, o objetivo da vidraçaria é usar o menor número destas placas 3x3. Este problema também aparece no corte de chapas, divisórias, compensados, etc.

Tem-se aqui um problema de corte bidimensional.

2.4.4 Problema Tridimensional

Quando se trata de três dimensões relevantes e fixas, como no caso do acondicionamento de caixas em um contêiner, onde se considera a largura, o comprimento e a altura na definição de um padrão de empacotamento.

Por exemplo, um problema que ocorre está relacionado ao transporte de produtos e cargas. Vários produtos (como vidros, pisos, azulejos) devem ser encaixotados e então transportados em furgões de dimensões fixas. Para minimizar os custos o objetivo aqui é fazer o menor número de viagens da loja de materiais para o local de construção. Trata-se aqui de um problema de corte tridimensional.

Problemas de empacotamento tridimensional são em geral NP-difíceis e freqüentemente não é possível encontrar soluções exatas dentro de um limite de tempo razoável (DOWSLAND e DOWSLAND, 1992) citado por CECILIO (2003). Assim, devido à dificuldade com a complexidade dos métodos exatos, a maioria das abordagens propostas na literatura refere-se a métodos aproximados (heurísticas, metaheurísticas, algoritmos aproximativos, etc.) MARTINS (2002) e SILVA (2003) citados por CECILIO (2003).

2.4.5 Formulação Matemática para o Problema Unidimensional

Segundo (VINK, 97), o problema de corte unidimensional pode ser formulado da seguinte maneira: Dados uma barra (*bin*) de capacidade c e n itens de peso (ou tamanho) w_i. A solução ideal será alocar os itens nas *bins* de maneira que o tamanho total dos itens não exceda a capacidade c de cada *bin* e o número de *bins* usadas seja o mínimo.

Uma definição mais formal seria a seguinte: Dados n itens e n *bins;* com cada item j de tamanho w_j e c a capacidade de cada *bin*. $N = \{0, 1, 2,...,N\}$. Define-se:

$$y_i = \begin{cases} 1 \text{ se a } \textit{bin } i \text{ for usada} \\ 0 \text{ caso contrário} \end{cases}$$

$$x_{ij} = \begin{cases} 1 \text{ se o objeto } j \text{ é atribuído a } \textit{bin } i \\ 0 \text{ caso contrário} \end{cases}$$

Considerando-se $0 < w_j < c$ para j e $c > 0$.

Possível solução: Uma instanciação das variáveis x_{ij} tais que:

$$\sum_{j=1}^{n} w_j x_{ij} \leq c; \quad i \in N$$

$$\sum_{i=1}^{n} x_{ij} = 1; j \in N$$

$$y_i = 0 \text{ ou } 1, i \in N$$

$$x_{ij} = 0 \text{ ou } 1, i \in N, j \in N$$

Função Objetivo: $\sum_{i=1}^{n} = y_i$

Solução Ótima: A relação das variáveis x_{ij} que minimiza $\sum_{i=1}^{n} = y_i$.

2.5 Métodos heurísticos para solucionar o Problema de Corte e Empacotamento

Nesta seção serão apresentados duas heurísticas e seu funcionamento, utilizados para resolver o Problema de Corte e Empacotamento Unidimensional. O Algoritmo Genético também é uma heurística para solucionar o Problema de Corte e Empacotamento

Unidimensional, mas por ele ser o método de heurístico utilizado neste trabalho a ele é dedicado a seção 2.6 deste trabalho.

2.5.1 Heurística First Fit Decreasing (FFD)

A heurística FFD, já em sua forma original, resolve o problema do corte para múltiplos objetos em estoque, isto é, resolve o Problema de Corte e Empacotamento para diversos objetos. O funcionamento deste algoritmo baseia-se na idéia de que há uma fila de itens a serem encaixados nos objetos, se o primeiro item não utilizado puder ser encaixado na solução corrente, deverá ser utilizado por ela (CINTRA, 04). O algoritmo FFD utiliza apenas a restrição física do problema, não fazendo mais nenhuma restrição aos itens que serão utilizados na solução. Por exemplo, suponha cinco itens, cuja unidade é metros, ordenados da seguinte forma:

I = [1,50; 1,25; 1,00; 0,75; 0,50]

e, este itens devem ser colocados em barras de 2 metros, o algoritmo funcionada seguinte forma:

- Inicia-se uma barra B1 como solução.
- Coloca-se o primeiro item, 1,50 metros em, B1.

Solução parcial:
B1 = [1,50]

- Verifica se o próximo item, 1,25 metros, pode ser colocado em B1, como não pode, é inciada uma nova solução.

Solução parcial:
B1 = [1,50]
B2 = [1,25]

- Verifica se o próximo item, 1,00 metro, pode ser encaixado em uma das soluções correntes, como não cabe em B1 e também não cabe em B2, é aberta uma nova solução.

Solução parcial:
B1 = [1,50]
B2 = [1,25]
B3 = [1,00]

- Verifica se o próximo item, 0,75 metro, pode ser encaixado em uma das soluções, não cabe em B1, mas cabe em B2, então é colocado em B2.

Solução parcial:
B1 = [1,50]
B2 = [1,25; 0,75]
B3 = [1,00]

Seguindo o raciocínio chegasse a seguinte solução:

Solução Final:
B1 = [1,50; 0,50]
B2 = [1,25; 0,75]
B3 = [1,00]

Verificasse que o único trabalho do algoritmo é o de verificar se o item pode ser utilizado em alguma das soluções já existentes. Caso não possa ser utilizado, é aberta uma nova solução.

2.5.2 Heurística First Fit (FF)

A Heurística FF é do tipo *on-line,* ou seja, os itens a serem empacotados não são conhecidos previamente, tem se aqui a diferença entre o FF e o FFD. Ou seja, as duas heurísticas trabalham tentando inserir um a um estes itens nos *bins* já existentes na solução, testando cada *bin* na ordem em que foram criados, verificando se estes possuem espaço para armazená-los. Caso ainda não tenha sido criado nenhum *bin* ou todos os *bins* criados não possuam espaço suficiente para armazenar um item será criado um novo *bin* onde será empacotado este item.

2.6 Algoritmos Genéticos

Desenvolvidos por *John Holland*, seus colegas e alunos na Universidade de *Michigan*; os Algoritmos Genéticos são algoritmos de busca baseados em mecanismos de Seleção Natural (Genética), dentro do conceito de "sobrevivência dos mais adaptados".

Durante os últimos anos, tem-se observado uma expansão na utilização de Algoritmos Evolutivos (Deb, 2001), especialmente os Algoritmos Genéticos, em problemas de otimização. Uma das características mais importantes dos Algoritmos Evolutivos é que possibilitam encontrar soluções otimizadas ou adequadas para um problema sem usar informação extra, como cálculo de derivadas de funções (Goldberg, 1989) citado por (TICONA, 2003).

Esta seção tem como objetivo apresentar os Algoritmos Genéticos como técnica de otimização. Os tópicos tratados abordam os conceitos básicos para entender o funcionamento, e as variações possíveis dos Algoritmos Genéticos, e as principais vantagens deste método em relação às técnicas tradicionais de otimização.

2.6.1 Definição

Os algoritmos genéticos são algoritmos de buscas baseados nos mecanismos de seleção natural e na genética. Esses algoritmos implementam estratégias de buscas paralelas e aleatórias para solucionar problemas de otimização (HOLLAND, 75).

Vários problemas que a computação evolutiva tenta resolver envolvem busca de uma solução num espaço vasto de candidatos, sujeitos a restrição do tempo. Neste contexto, é que surgem os algoritmos genéticos, como uma maneira de solucionar problemas de busca adaptativa, nos quais o conhecimento para controlar a busca é obtido dinamicamente.

Assim, os algoritmos genéticos são técnicas de busca inspiradas em mecanismos de seleção e genética natural. Dada uma população inicial de soluções, esta evolui até convergir para uma solução, por meio da aplicação de operadores genéticos de seleção, cruzamento e mutação.

Considerando um problema de otimização qualquer, os algoritmos genéticos iniciam a busca de melhores soluções a partir de um conjunto inicial de soluções aleatórias. Cada elemento do conjunto inicial de soluções é denominado indivíduo ou cromossomo. Um indivíduo pode ser representado por uma cadeia de símbolos, por exemplo, uma cadeia binária. É importante que cada indivíduo da população seja capaz de representar completamente uma possível solução do problema tratado.

Em seguida, uma nova população é gerada a partir da população inicial. Para criar os indivíduos da nova população, são utilizados operadores genéticos de cruzamento e mutação. Uma população é obtida a partir da anterior, aplicando-se o cruzamento para aqueles indivíduos com um maior valor de aptidão tenham maior chance de se reproduzir, simulando o processo de seleção natural. O valor de aptidão é calculado para cada indivíduo mediante uma função chamada função de aptidão. Assim, os algoritmos genéticos são métodos de busca guiados pela função de aptidão. Aleatórios, não executam buscas sem rumo, pois através de processos iterativos (gerações) eles exploram informações históricas de cada geração para encontrar novos pontos de busca onde são esperados melhores desempenhos (YEPES 2006).

Os indivíduos de uma população possuem um conjunto de características, também denominado de fenótipo. Estas características são transmitidas dos pais para os filhos entre as novas populações pela hereditariedade durante a reprodução. Segundo (DEVILLERS, 96), a adaptação é medida através da evolução do fenótipo no ambiente do problema. O processo de seleção natural possibilita aos indivíduos mais aptos à sobrevivência e a reprodução em maior escala em relação aos indivíduos que não conseguiram se adaptar, propiciando a evolução da espécie.

Do mesmo modo que a seleção natural, os algoritmos genéticos selecionam as soluções para gerar indivíduos cada vez mais aptos, através da recombinação do material genético (cromossomos) de seus antecessores. Possuem uma população com determinado número de possíveis soluções para o problema (cromossomos), e para cada possível solução está associado um valor de aptidão (capacidade de adaptação). Estes indivíduos sofrerão um processo de seleção, o qual irá determinar quais deles serão escolhidos para a reprodução. Realizada a seleção, os indivíduos irão reproduzir e gerar novos descendentes.

Após realizada a reprodução e geração de novos indivíduos, os pais são descartados ("morrem") dando lugar a seus descendentes que irão substituí-los na nova população. A nova geração passará pelo mesmo processo de seus pais, até que gerem uma nova população. Este processo ocorre sucessivamente até as que melhores soluções dominem uma população.

Segundo (FURTADO, 98), conforme a adaptação dos indivíduos (solução) ao ambiente (problema), a população com o passar das gerações apresenta seus melhores descendentes. Conforme estes indivíduos vão gerando novas populações, começam demonstrar poucas diferenças em relação a seus pais, fazendo com que o algoritmo convirja a um conjunto de soluções ótimas ou próximas do ótimo para a resolução do problema em questão. Assim, o processo de geração de novas populações é repetido iterativamente até que o algoritmo genético chegue a uma solução aceitável, ou satisfaça alguma condição de parada.

E embora possam parecer simplistas do ponto de vista biológico, estes algoritmos são suficientemente complexos para fornecer mecanismos de busca adaptativa, poderosos e robustos (CANTÚ, 95).

2.6.2 Representação das Soluções

Baseado nos sistemas naturais, os algoritmos genéticos codificam e representam as soluções dos problemas na forma de indivíduos. Os indivíduos podem ser constituídos por um ou mais cromossomos (DEVILLERS, 96).

Num espaço de busca ou espaço de possíveis soluções para o problema, os algoritmos genéticos mantêm uma população de indivíduos. Estes representam possíveis soluções para o problema a ser resolvido. Cada indivíduo da população é representado por um vetor de tamanho finito em cima de algum alfabeto finito. Tradicionalmente utilizam vetores binários {0,1}, onde cada elemento simboliza a presença {1} ou ausência {0} de uma determinada característica, o seu genótipo.

Um exemplo de um cromossomo pode ser visualizado na Figura 2.6.1:

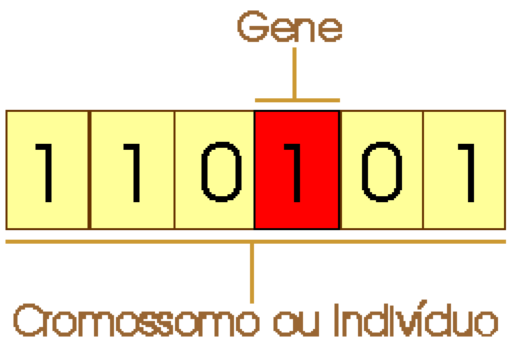

Figura 2.6.1 - Exemplo de representação de um cromossomo de genes binários.

Um cromossomo é formado por um conjunto de genes. Cada gene ou uma combinação de genes representa ou controla uma característica específica de um indivíduo (solução). A forma de representação pode ser posicional ou não posicional.

A técnica posicional implica em que a posição de cada gene dentro de um cromossomo influência diretamente na obtenção do fenótipo, que é determinado a partir do somatório binário. A Figura 2.6.2 mostra um exemplo onde o primeiro cromossomo possui um fenótipo igual a 1379. No segundo cromossomo foi alterado apenas o terceiro gene da esquerda para a direita, modificando o fenótipo para 1381.

Posicional											
Cromossomo											Fenótipo
1	0	1	0	1	1	0	0	0	1	1	1379
1	0	1	0	1	1	0	0	1	1	1	1381

Figura 2.6.2 - Representação cromossômica posicional

Já utilizando a técnica não posicional, a posição dos genes dentro do cromossomo é irrelevante, visto que se pode obter o mesmo fenótipo a partir de vários genótipos diferentes. A Figura 2.6.3 mostra o primeiro cromossomo com fenótipo igual a 7, no segundo cromossomo foi modificando o terceiro gene da direita para a esquerda e o fenótipo ficou igual a 8. Enquanto no terceiro cromossomo foram modificados da direita para a esquerda o quarto e o oitavo gene, o fenótipo passou a ser igual 7, o mesmo fenótipo do primeiro cromossomo.

Assim, foi possível observar que com genótipos diferentes, pode haver um ou mais fenótipos iguais. Para obter o fenótipo apenas são somados os genes.

Não Posicional											
Cromossomo											Fenótipo
1	0	0	1	1	1	1	0	1	0	1	7
1	0	1	1	1	1	1	0	1	0	1	8
1	0	0	0	1	1	1	1	1	0	1	7

Figura 2.6.3 - Representação cromossômica não posicional

A representação aplicada neste trabalho é do tipo não posicional, pois o que procuramos é um maior aproveitamento dos objetos, assim, o que nos interessa é somente o número de barras utilizadas nas soluções e seus respectivos aproveitamentos, não diferenciando o aproveitamento do primeiro ou o *n-ésimo* objeto da solução encontrada.

A seguir serão apresentados os principais operadores genéticos encontrados na literatura e suas variações mais utilizadas.

2.6.3 Seleção

Seleção é o processo onde os indivíduos com melhor valor de aptidão têm a maior probabilidade de gerar um ou mais descendentes para a geração seguinte. Este operador é a versão artificial da seleção das espécies do Darwinismo, que estabelece que os seres mais aptos tenham maiores chances de sobreviver, ou seja, os mais fortes e menos vulneráveis aos predadores e doenças.

O objetivo principal do operador de seleção é copiar boas soluções, eliminando soluções de baixa aptidão, enquanto o tamanho da população é constante (DEB, 2001). Isto é realizado seguindo os seguintes passos:

- Identificar boas soluções na população;
- Realizar múltiplas cópias das boas soluções;
- Eliminar soluções de baixa aptidão da população o que permite que várias cópias de boas soluções passem para as próximas gerações.

As melhores soluções são guardadas em uma lista de soluções escolhidas ou lista de soluções, que será utilizada para realizar as operações de cruzamento e mutação.

O propósito de seleção de pais nos algoritmos genéticos é possibilitar maior probabilidade de reprodução para indivíduos com maior valor de aptidão. No entanto, não exclusivamente, em virtude da manutenção da diversidade genética da população.

Existe um grande número de estratégias de seleção. As mais comuns são seleção pelo torneio, roleta, seleção proporcional, e seleção por ranking.

Na seleção por torneio, são realizadas várias competições entre duas soluções, e a melhor solução é copiada na lista de soluções que serviram para que novos indivíduos sejam gerados. Este processo é repetido até preencher a lista. Goldberg e Deb mostraram que este método possui uma convergência igual, ou melhor, que outras estratégias de seleção, além de possuir uma complexidade computacional menor (DEB, 2001). Este foi o tipo de seleção aplicada ao algoritmo genético implementado neste trabalho, com pequenas adaptações, as quais serão detalhadas e explicadas na seção 3.2.3.

Na seleção pelo método da roleta, inicialmente proposto por Goldberg (GOLBERG, 89), cada indivíduo da população é representado na roleta conforme seu valor de aptidão. Desta forma, os indivíduos com elevada aptidão receberão um intervalo maior na roleta, enquanto aqueles que têm mais baixa aptidão receberão menor intervalo na roleta. Após a distribuição na roleta, são gerados aleatoriamente números no intervalo entre 0 e o total do somatório da aptidão de todos os indivíduos da população. É gerado um determinado número de vezes, dependendo do tamanho da população. O indivíduo que possuir em seu intervalo o número gerado será selecionado para o cruzamento.

Para exemplificar o método da Roleta, suponhamos uma população de 6 cromossomos cuja aptidão é dada por uma função qualquer (neste caso é simplesmente a conversão de binário para decimal) conforme mostrado na Tabela 2.6.1 (YEPES, 2006).

Tabela 2.6.1 - Valores de exemplo para ilustrar seleção por roleta

Cromossomo nº	String	Aptidão	% do total
1	0101101	45	13,2
2	1011001	89	26,2
3	1111101	125	36,7
4	0010101	21	6,1
5	0110100	52	15,2
6	0001001	9	2,6
	Total	**341**	**100,0**

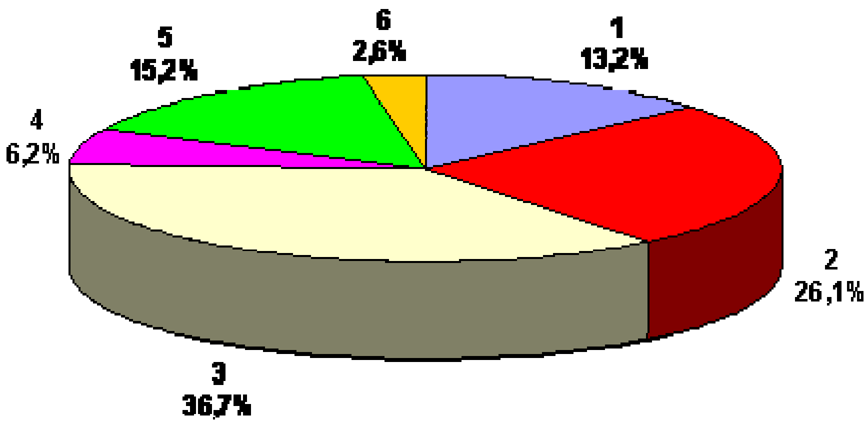

Figura 2.6.4 - Representação gráfica da roleta.

Com os valores percentuais constantes na quarta coluna da Tabela 2.6.1, foi elaborada a roleta constante da Figura 2.6.4 (YEPES, 2006). Esta roleta será girada para efetuar a seleção dos indivíduos que irão fazer o cruzamento, levando em conta que, os indivíduos com maior área na roleta tem, consequentemente, maiores chances de serem selecionados mais vezes que os indivíduos menos aptos.

De forma simplificada, a roleta compreende três passos:

- Somar o valor de aptidão de todos os membros da população, chamado o resultado total da aptidão.
- Gerar *N* números de forma aleatória no intervalo entre 0 e o total da aptidão.
- Retornar o primeiro membro da população cuja aptidão somada a aptidão do membro precedente da população, seja maior ou igual a *N*.

Na estratégia de seleção proporcional, o número de cópias de uma solução na lista de soluções escolhidas é proporcional ao seu valor de aptidão. Para calcular o número de cópias esperado, é necessário obter a probabilidade de cada solução:

$$p_i = \frac{F_i}{\sum_{i=1}^{N} F_i} \qquad (2.6)$$

Sendo que F_i é a aptidão da solução i. N é o tamanho da população. O número de cópias na lista de soluções é calculado por $C_i = p_i \times N$. Ou seja, as soluções com melhor valor de aptidão terão mais copias na lista de soluções. O escalonamento é um problema associado com esta estratégia. Quando existe uma solução com um valor de aptidão muito maior comparado com o resto da população, esta super-solução terá uma probabilidade de escolha perto de um, e terá muitas cópias na lista de soluções. Caso todas as soluções possuam valores similares de aptidão, terão a mesma probabilidade de serem escolhidas, e cada uma será copiada na lista de soluções. Isto é equivalente a não realizar operação de seleção.

A estratégia de seleção pelo ranking ordena as soluções da população conforme ao seu valor de aptidão, desde a pior solução (*ranking* 1) até a melhor (*ranking N*). Depois, o número de copias das soluções é proporcional ao valor de *ranking*, (TICONA, 2003).

2.6.4 Cruzamento

Poderia ser dito que a principal característica de distinção do Algoritmos Genéticos em relação as outras técnicas é o uso do cruzamento (MITCHELL 96).

O operador Cruzamento é utilizado após o de seleção. Esta fase é marcada pela troca de segmentos entre "casais" de cromossomos selecionados para dar origem a novos indivíduos que formarão a população da próxima geração. Ele é considerado o operador genético predominante, por isso é aplicado com probabilidade dada pela taxa de *crossover*, que deve ser maior que a taxa de mutação.

A idéia central do cruzamento é a propagação das características dos indivíduos mais aptos da população por meio de troca de segmentos de informações entre os mesmos, o que dará origem a novos indivíduos.

As duas formas mais comuns de reprodução sexual em Algoritmos Genéticos são de um ponto de cruzamento e de dois pontos de cruzamento.

Um ponto de cruzamento (*single-point crossover*): o ponto de quebra do cromossomo é escolhido de forma aleatória sobre a longitude da string que o representa, e é a partir desse ponto que se realiza a troca de material cromossômico entre os dois indivíduos, como é representado na Figura 2.6.5, na qual a termo de representação foi escolhido o ponto de cruzamento 4.

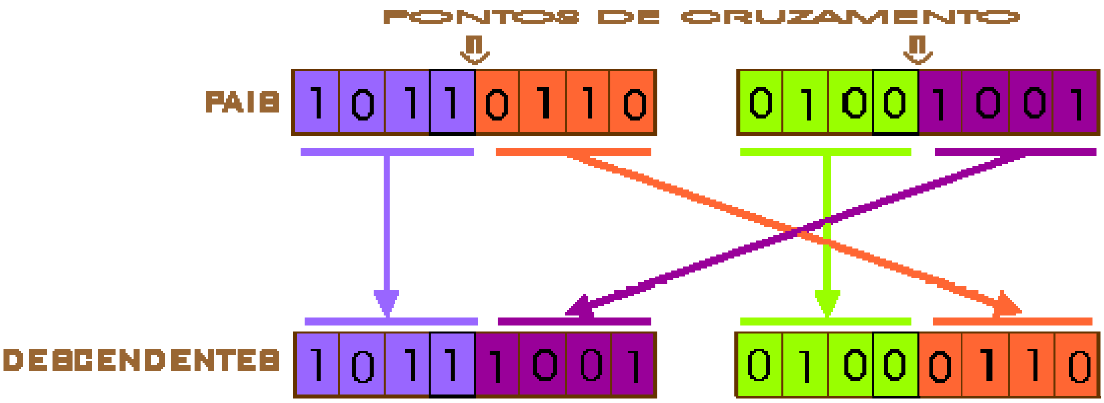

Figura 2.6.5 - Esquema gráfico do cruzamento de um ponto.

Dois pontos de cruzamento (*two-point crossover*): procede-se de maneira similar ao cruzamento de um ponto, mas a troca de segmentos do gene ocorre como mostrado na Figura 2.6.6, na qual a termo de representação foram escolhidos os pontos 2 e 6:

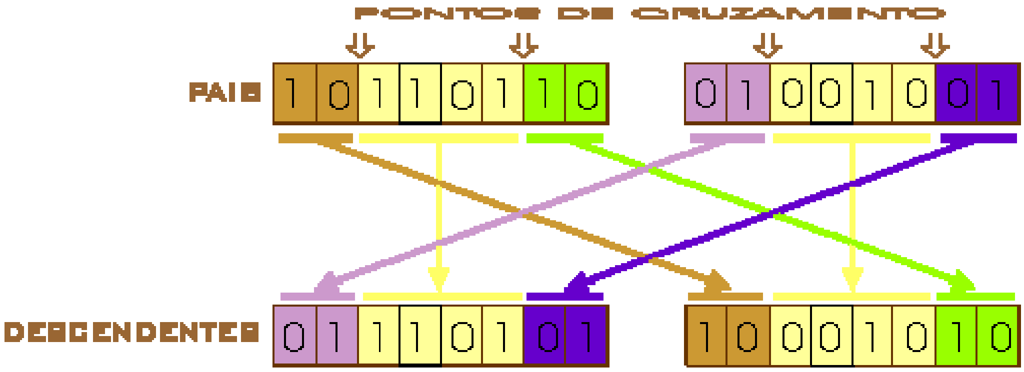

Figura 2.6.6 - Esquema gráfico do cruzamento de dois pontos.

Cruzamento Uniforme (*uniform crossover*): É radicalmente diferente do cruzamento de um ponto. Cada gene do descendente é criado copiando o gene correspondente de um dos pais, escolhido de acordo com uma máscara de cruzamento gerada aleatoriamente. Onde houver 1 na máscara de cruzamento, o gene correspondente será copiado do primeiro pai e, onde houver 0 será copiado do segundo. O processo é repetido com os pais trocados para produzir o segundo descendente. Uma nova máscara de cruzamento é criada para cada par de pais. O número de pontos de troca de informação não é fixo, mas em geral é usado $l/2$ (onde l é o comprimento do cromossomo). Este processo é mostrado graficamente na Figura 2.6.7.

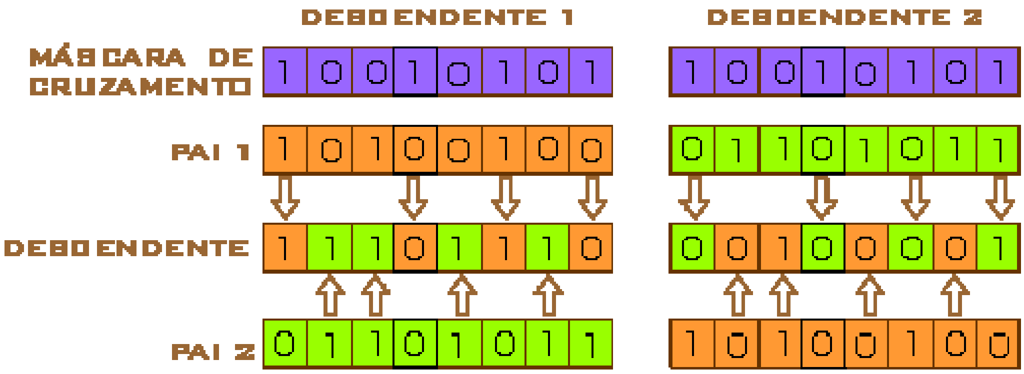

Figura 2.6.7 - Esquema gráfico do cruzamento uniforme.

Apesar das operações de seleção e cruzamento parecerem muito simples, é nelas que se concentra o poder dos algoritmos genéticos (Goldberg, 1989).

2.6.5 Mutação

A mutação é um operador que produz uma alteração aleatória em uma posição de um pequeno número de indivíduos, isto para possibilitar uma maior variabilidade de material genético dentro da população.

A mutação é a segunda maneira dos algoritmos genéticos explorarem o espaço de busca. Esta pequena alteração causada é para tentar impedir que o algoritmo tenha convergência muito rápida, evitando sua estabilização em regiões de mínimos locais, (TICONA, 2003).

2.6.6 Elitismo

Pelo menos uma cópia sem alterações da melhor solução da geração anterior é passada para a nova população, assim, o operador de elitismo mantém as melhores soluções encontradas previamente nas gerações seguintes. Isto implica que a melhor solução não se deteriora nas sucessivas gerações.

Uma forma de implementar elitismo é copiar diretamente uma porcentagem das soluções da população atual à população seguinte. O restante das soluções é gerada usando os operadores genéticos usuais sobre a população atual. Desta forma, as melhores soluções passam diretamente para população seguinte, além de participar da criação do resto de soluções da população seguinte.

Outra forma de elitismo consiste em criar a população seguinte a partir da população atual usando os operadores genéticos usuais, e escolher as melhores N soluções de ambas populações.

Este pode ser um modelo melhor do que ocorre na natureza (pelo menos em espécies superiores).

2.6.7 Parâmetros Genéticos

É importante também, analisar de que maneira alguns parâmetros influem no comportamento dos algoritmos genéticos, para que se possa estabelecê-los conforme as necessidades do problema e dos recursos disponíveis.

- **Tamanho da População**: O tamanho da população afeta o desempenho global e a eficiência dos algoritmos genéticos. Com uma população pequena o desempenho pode cair, pois deste modo a população fornece uma pequena cobertura do espaço de busca do problema. Uma grande população geralmente fornece uma cobertura representativa do domínio do problema, além de prevenir convergências prematuras para soluções locais ao invés de globais. No entanto, para se trabalhar com grandes populações, são necessários maiores recursos computacionais, ou que o algoritmo trabalhe por um período de tempo muito maior.
- **Taxa de Cruzamento**: Quanto maior for esta probabilidade, mais rapidamente novas estruturas serão introduzidas na população. Mas se esta for muito alta as estruturas com boas aptidões poderão ser retiradas e a maior parte da população será substituída. Com um valor baixo, o algoritmo pode tornar-se muito lento (DE JONG, 93).
- **Taxa de Mutação**: Determina a probabilidade em que uma mutação ocorrerá. Mutação é utilizada para dar nova informação para a população e também para prevenir que a população se sature com cromossomos semelhantes (Convergência Prematura). Uma baixa taxa de mutação previne que uma dada posição fique estagnada em um valor, além de possibilitar que se chegue em qualquer ponto do espaço de busca. Com uma taxa muito alta a busca se torna essencialmente aleatória além de aumentar muito a possibilidade de que uma boa solução seja destruída. A melhor Taxa de Mutação é dependente da aplicação mas, para a maioria dos casos é entre 0,001 e 0,1. Com uma taxa muito alta a busca se torna essencialmente aleatória.
- **Intervalo de Geração**: Controla a percentagem da população que será substituída durante a próxima geração. Com um valor alto, a maior parte da população será substituída, mas com valores muito altos pode ocorrer perda de estruturas de alta aptidão. Com um valor baixo, o algoritmo pode tornar-se muito lento.

2.6.8 Diferenças entre Algoritmos Genéticos e outras estratégias de otimização

Vista definição de Algoritmos Genéticos, pode-se apontar algumas diferenças em relação a técnicas tradicionais: (Goldberg, 1989)

- Os Algoritmos Genéticos trabalham com uma população de soluções em cada iteração ao invés de uma única solução. Em cada iteração os Algoritmos Genéticos processam um conjunto de soluções, esta característica é denominada de paralelismo implícito.
- Os Algoritmos Genéticos não precisam de informação adicional a não ser o valor de aptidão das soluções. Isto faz que os Algoritmos Genéticos sejam aplicáveis a uma grande variedade de problemas, alguns dos quais não se tem informações a priori.
- Os Algoritmos Genéticos empregam regras probabilísticas para guiar a sua busca. Por exemplo, o processo de seleção é baseado na aleatoriedade de duas soluções (seleção pelo torneio), ou na probabilidade de escolha (seleção proporcional) dessas soluções. O operador de mutação permite que os Algoritmos Genéticos não recaiam nos ótimos locais, mudando a busca para outra região do espaço. Além de que as soluções da população inicial são escolhidas arbitrariamente. Em contrapartida, uma técnica de regras fixas não terá como escapar de ótimos locais, em caso de uma má decisão sobre a direção da busca.

A possibilidade de se trabalhar com várias soluções simultaneamente, de não precisar de informações adicionais e poder escapar de ótimos locais fazem dos Algoritmos Genéticos uma técnica que pode ser bastante satisfatória para Problemas de Corte e Empacotamento.

3 PROPOSTA DESTE TRABALHO

Esta seção consiste em mostrar a função de adaptabilidade para o problema de corte e empacotamento e também tudo o que foi desenvolvido neste trabalho, através de detalhes da implementação e exemplos.

3.1 Função de Adaptabilidade para Problema de Corte e Empacotamento Unidimensional

Cada indivíduo de uma população tem associado a si um valor de aptidão, identificando seu grau de adaptabilidade ao ambiente. Sendo assim, o indivíduo que após essas gerações obter um ótimo valor de aptidão (ou considerado satisfatório, próximo do ótimo) é selecionado para resolver o problema, através da aplicação da solução nele codificada.

A determinação do valor de aptidão de cada indivíduo é obtido através da função aptidão. A função aptidão mostra o quanto uma solução é boa se comparada com outra solução da população. Desta maneira, os elementos da população corrente devem ser avaliados individualmente pela função aptidão, para verificar se estão ou não aptos para solucionar o problema (ou perto da solução). Esta função apresenta o grau de adaptação do indivíduo ao meio ambiente, para saber se possui ou não condição de sobreviver e criar descendentes. Assim, quanto mais alto for o valor de aptidão de uma solução, maior é a sua capacidade de sobrevivência e reprodução e, consequentemente, maior é sua representação na próxima geração.

A função aptidão para o problema de corte unidimensional é representada pela Equação (VINK, 97):

$$f = \sum_{i=0}^{B} \frac{(F_i / C)^2}{B}$$

Onde:

- B = Número barras
- C = Capacidade da barra
- F_i = Ocupação, aproveitamento da barra.

3.2 Algoritmo Genético

Para resolver o problema do objeto em estudo, o Problema de Corte e Empacotamento Unidimensional (*Bin Packing Problem*), utilizou-se o Algoritmo Genético, já caracterizado na seção 2.6 deste trabalho, pois sabe-se que o uso de métodos exatos para este problema se limita a pequenas instâncias, devido ao grande tempo de execução para se chegar ao resultado final.

Assim, este trabalho iniciou com o intuito de produzir um Algoritmo Genético eficiente, propondo variações nos operadores clássicos da literatura, mas ainda dentro de seu conceito padrão, estipulado por Holland em (Holland (1975)), sendo representado de maneira sucinta:

Algoritmo Genético Clássico

1: P ← População Inicial;

2: **enquanto** condição de parada não satisfeita **faça**

3: P^t ← Seleção(P);

4: P ← Cruzamentos (P^t);

5: P ← Mutações(P);

6: **fim enquanto**

7: Solução Melhor indivíduo(P);

A condição de parada para os algoritmos genéticos podem ser número de iterações que se deseja que o algoritmo seja executado, um intervalo de tempo ou a convergência, está última considerada quando há um certo número de gerações sem melhora na solução. No caso deste trabalho estabelece-se um determinado número de iterações.

3.2.1 Caracterização do Problema

Supondo-se um problema de minimização do número de barras (*bins*), estas representadas pela Figura 3.2.1, onde se utilizou a representação por números inteiros, e que a capacidade (c) de cada uma seja 100 metros.

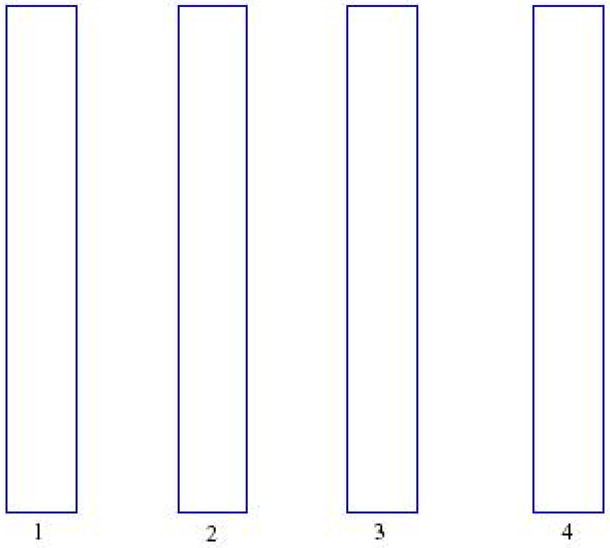

Figura 3.2.1 - Exemplo de barras

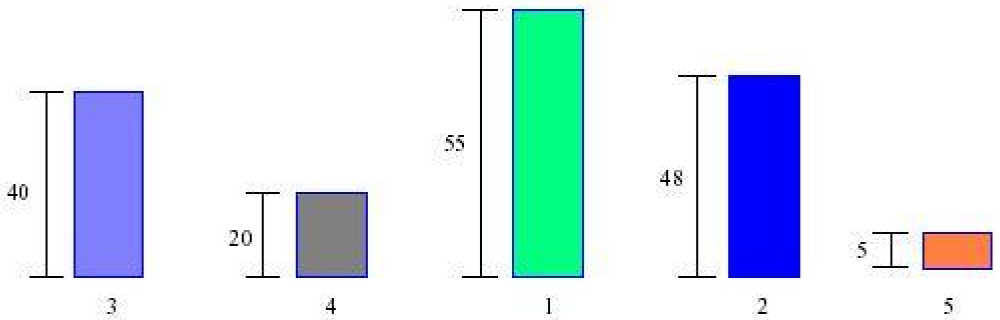

Figura 3.2.2 - Itens a serem inseridos nas barras

A disposição da Figura 3.2.2 indica que: Existem 3 itens de 40 m, 4 itens de 20 m, 1 item de 55 m, 2 itens de 48 m e 5 itens de 5 m.

Estes itens serão os dados de entrada para que o algoritmo genético faça a busca pelo menor número de barras possíveis que suporte os itens. No algoritmo estes dados de entrada são representados por um vetor **V** cujo tamanho é igual ao número total de itens, e cada posição V[i] do vetor é caracterizada pelo tamanho do item que se encontra nesta posição. Para o exemplo acima teríamos a representação de **V** da seguinte forma:

V = [40, 40, 40, 20, 20, 20, 20, 55, 48, 48, 5, 5, 5, 5, 5]

Note que se têm três itens de 40 m a serem inseridos nas barras, no vetor **V** têm-se os três itens de 40m. E assim, o mesmo ocorre também para os demais itens.

Para a representação das soluções do problema, que são os indivíduos do algoritmo genético, também se tem um vetor, por exemplo, o vetor **S**, de modo que na posição S[i] do vetor encontra-se o número da barra a qual o item V[i] foi colocado. Na próxima seção a um detalhamento maior sobre a representação do vetor **S.**

Assim, dado o vetor inicial de itens o propósito do algoritmo é encontrar uma solução que se utilize do menor número de barras possíveis, e todos os itens estejam dentro de uma barra, e nenhuma barra exceda sua capacidade.

3.2.2 População Inicial

Para que o algoritmo comece a ser executado precisa-se de uma população inicial, que nada mais é do que uma geração aleatória de possíveis soluções para o problema, população a qual o algoritmo aplicará os operadores genéticos em busca de soluções melhores.

Para que esta população inicial seja gerada os únicos dados de entrada que se precisa é o vetor de itens, como o mostrado anteriormente, e a capacidade c das barras. Para esta geração, assim como para todo o algoritmo, considera-se que o número de barras é ilimitado, ou seja, serão utilizadas quantas barras forem necessárias para que se coloquem todos os itens. Cada indivíduo da população inicial é gerado da seguinte forma:

- É gerado um vetor **A** de números aleatórios, com os números variando entre 1 e o tamanho do vetor de entrada, este vetor não contém repetições para que se cubra todas as posições do vetor de entrada.
- O vetor de números aleatórios gerado será a ordem na qual os itens do vetor de entrada serão colocados nas barras, ou seja, suponha que A[1] = 4, e tomando o vetor **V** citado anteriormente como vetor de entrada, o primeiro

item a ser inserido na primeira barra seria o item V[4] que é um item de20m.

- Para o vetor **S,** que representa cada indivíduo, na posição onde se encontra o item no vetor inicial é inserido o número da barra a qual o item foi inserido, neste caso, como o item foi inserido na primeira barra: S[4] = 1.
- Os itens são inseridos na barra atual enquanto esta suportar o item que está sendo inserido, caso a barra atual não suporte o item, é iniciada uma nova barra para que seja colocado este item. Este processo é repetido até o fim do vetor **A,** ou seja, como **A** cobre todas as posições do vetor de entrada, todos os itens foram inseridos em alguma barra.

Desse modo, uma população de soluções iniciais para o exemplo de barras e itens citados acima, poderia ser:

S1 (3; 4; 2; 1; 1; 2; 2; 3; 1; 4; 1; 1; 2; 2; 3)
S2 (1; 1; 2; 2; 2; 2; 3; 3; 4; 4; 1; 1; 1; 1; 3)
S3 (1; 2; 3; 4; 3; 3; 4; 1; 2; 4; 1; 2; 2; 3; 3)
S4 (1; 1; 2; 1; 2; 2; 2; 3; 4; 4; 3; 3; 3; 3; 3)
S5 (3; 3; 2; 3; 2; 2; 2; 1; 4; 4; 1; 1; 1; 1; 1)

A solução **S1** é representada na Figura 3.2.3 **- Representação da Solução S1** e possui o seguinte significado:

- Os itens de 40 m estão alocados nas barras de números 2, 3 e 4.
- Os itens de 20 m estão alocados nas barras de números 1 e 2.
- O item de 55 m está alocado na barra de número 3.
- Os itens de 48 m estão alocados nas barras de números 1 e 4.
- Os itens de 5 m estão alocados nas barras de números 1, 2 e 3.

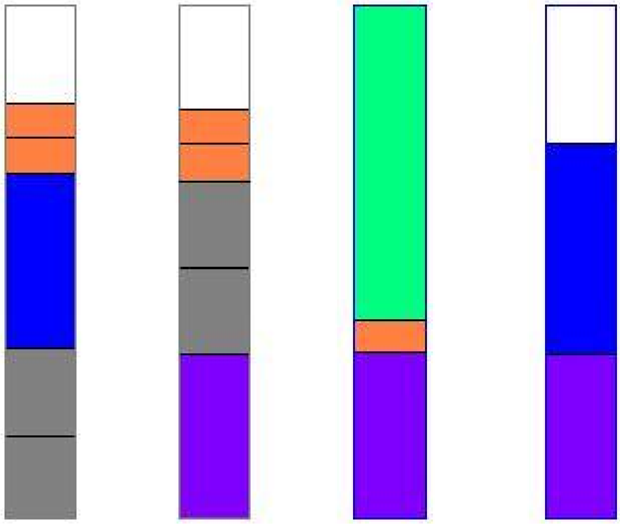

Figura 3.2.3 - Representação da Solução S1.

Com a população inicial gerada calcula-se a aptidão, também conhecida como *fitness,* para cada indivíduo, para que se possa ser possível iniciar efetivamente a busca pela melhor solução, ou uma solução próxima de dela. Como exemplo, para as soluções apresentadas acima, a Tabela 3.2.1 mostra o valor de seus *fitness*.

Tabela 3.2.1 - *Fitness* das soluções exemplos

Solução Si	**Fitness f(Si)**
S1	**0,84**
S2	**0,89**
S3	**0,93**
S4	**0,89**
S5	**0,89**

Depois de gerada a população inicial e calculado o *fitness* para cada indivíduo passamos para o processo iterativo do algoritmo, passo dois do algoritmo genético clássico.

3.2.3 Seleção

O operador de seleção, como próprio nome sugere, é o responsável por selecionar os indivíduos pais na população atual, para que destes sejam gerados novos indivíduos para a próxima geração.

O método de seleção implementado foi o torneio, cuja definição encontra-se na seção 2.6.3. O que este operador faz é pegar aleatoriamente na população atual dois indivíduos, comparar seus *fitness* e guardar em um indivíduo pai uma copia do que apresentar o maior *fitness*, depois repete-se a operação novamente guardando a melhor solução em um indivíduo mãe. Nestas duas soluções selecionadas, indivíduos pai e mãe, é aplicado o operador de cruzamento e os novos indivíduos gerados, indivíduos filhos, são inseridos na nova população.

Este processo de seleção e cruzamento é repetido até que a nova população esteja completa.

3.2.4 Cruzamento

Após a seleção dos indivíduos pais é feito o cruzamento para a geração dos novos indivíduos. O operador de cruzamento implementado com o método um ponto de cruzamento, que consiste em trocar as informações dos pais em um determinado ponto dos seus genes, este ponto de cruzamento recebe o nome de *pcross*, referente ao ponto de *crossover*, e é gerado aleatoriamente a cada novo cruzamento feito.

Por exemplo, tome como soluções pais fornecidos pelo operador de seleção:

S1(1; 1; 2; 1; 2; 2; 2; 3; 4; 4; 3; 3; 3; 3; 3) e

S2(1; 1; 2; 2; 2; 2; 3; 3; 4; 4; 1; 1; 1; 1; 3) e o *pcross* = 10, a Figura 3.2.4 representa o cruzamento entre os dois indivíduos:

S1 (1; 1; 2; 1; 2; 2; 2; 3; 4; 4; 3; 3; 3; 3; 3)

S2 (1; 1; 2; 2; 2; 2; 3; 3; 4; 4; 1; 1; 1; 1; 3)

$$\Downarrow$$

S1’ (1; 1; 2; 1; 2; 2; 2; 3; 4; 4; 1; 1; 1; 1; 3)

S2’ (1; 1; 2; 2; 2; 2; 3; 3; 4; 4; 3; 3; 3; 3; 3)

Figura 3.2.4 - Cruzamento em um ponto com *pcrros* = 10.

Os indivíduos **S1’** e **S2’** é o resultado do cruzamento entre **S1** e **S2**. Note que para alguns casos o cruzamento pode gerar indivíduos inviáveis, pois com a troca de genes pode haver barras que excedam sua capacidade, para isto foi implementado um operador que faz a correção de indivíduos, assim, quando o cruzamento gerar indivíduos inviáveis estes são passados para o operador citado e ele é retornado como uma solução viável para o problema. O operador que faz a correção dos indivíduos inviáveis esta descritos a seguir, na seção 3.2.6 deste trabalho.

É importante ressaltar também que após o cruzamento pode haver barras cujo aproveitamento é 0, ou seja, não está sendo utilizada, para isto logo após o cruzamento é feita uma verificação se há alguma barra com aproveitamento nulo, e caso exista, esta é retirada do indivíduo e o número de barras é ajustado. Note que é justamente quando esta situação ocorre que o número de barras diminui, e ainda implicando que o aproveitamento geral das outras barras foi aumentado.

Na literatura, após a aplicação do operador de cruzamento, as duas soluções do cruzamento são inseridas na nova população, mas como um diferencial do trabalho resolveu-se inserir apenas a melhor das duas soluções geradas. Com esta escolha no operador de cruzamento atente-se para o fato de que a cada cruzamento só um descendente vai para a nova população, com isto o operador de seleção foi chamado o dobro de vezes do que o algoritmo genético tradicional. E o que se pode observar foi que como o operador de seleção é chamado mais vezes houve uma maior abragência do número de indivíduos da população atual envolvida nos torneios do operador, e em alguns casos até totalidade dos indivíduos da população haviam participado de ao menos um torneio, o que ainda garante a diversidade biológica para a próxima geração.

Assim como na biologia, o operador de cruzamento é o grande responsável por gerar a nova população, ficando por sua conta 90% dos indivíduos da nova população, o restante é gerado pelos operadores de mutação e elitismo citados a seguir.

3.2.5 Mutação

Com o intuito de se garantir uma maior diversidade de soluções para evitar que o algoritmo não fique preso a mínimos locais, parte da nova população é gerada pelo operador de mutação.

O operador de mutação consiste em selecionar aleatoriamente um indivíduo na população atual, aplicar a mutação a ele e inseri-lo na nova população. O operador de mutação consiste em escolher um ponto aleatório dentro do indivíduo e, também aleatoriamente, alterá-lo. Como os indivíduos são representados por um vetor que indica a barra onde cada item está alocado, quando a mutação é aplicada o que ele está fazendo é trocar um item de barra, assim o número gerado para alteração da barra é entre 1 e o número de barras do indivíduo.

Por exemplo, dada à solução **S1 (1; 2; 3; 4; 3; 3; 4; 1; 2; 4; 1; 2; 2; 3; 3)** selecionada para sofrer mutação, o ponto de mutação gerado como 9, e a barra sorteada para receber o item a barra 4, o indivíduo gerado pela mutação é representado na Figura 3.2.5.

S1 (1; 2; 3; 4; 3; 3; 4; 1; 2; 4; 1; 2; 2; 3; 3)

$$\Downarrow$$

S1’ (1; 2; 3; 4; 3; 3; 4; 1; 4; 4; 1; 2; 2; 3; 3)

Figura 3.2.5 - Mutação no ponto 9

Assim como no cruzamento, a mutação também pode gerar indivíduos inviáveis, pois com a troca de genes a barra que recebe um novo item pode exceder sua capacidade. Para isto foi implementado o operador que faz a correção de indivíduos, assim, quando a mutação gerar indivíduos inviáveis. Note que se este indivíduo **S1’** for uma solução do problema citado no inicio desta seção a barra 4 excedeu sua capacidade, e a este indivíduo terá de ser aplicado o operador de correção para que possa ser incluído na população atual.

O operador de mutação é responsável pela geração 5% dos indivíduos da nova população.

3.2.6 Correção de Indivíduos

Este operador foi implementado para que, dado um indivíduo inviável, ele o torne viável. Sempre após os operadores de mutação e cruzamento é chamado um método para verificar se o indivíduo é viável, caso não seja este operador é chamado recebendo o indivíduo em questão e retornando-o viável.

O primeiro passo feito é calcular o aproveitamento de cada uma das barras da solução e guardar quais excederam a capacidade. A partir deste ponto a estratégia implica em retirar da barras que excederam a capacidade o menor item que deixe a solução viável. Por exemplo, a capacidade das barras do problema é 50 e foi localizada uma barra excedida que estava com os seguintes itens [17, 25, 7, 9], ou seja, sua capacidade está em 58. Dentre os itens procura se o menor que seja maior ou igual ao excesso da barra, ou seja, o menor dos itens que seja maior ou igual a 8, o que nos da que item a ser retirado é o item 9. Isto é feito a fim de se deixar a barra com o maior aproveitamento possível. Depois de verificado qual o item deve ser retirado da barra verifica-se dentre todas as barras quais têm capacidade disponível para suportar o item retirado. Dentre as barras que suportem o item, este é colocado na de menor capacidade disponível. Suponha que dentre as barras que suportem a o item 9 retirado, estejam barras com capacidade livre de 10, 15 e 20, o item é colocado na barra com capacidade livre igual a 10, isto para que a barra fique mais perto de um aproveitamento total.

Para esta estratégia também foi implementado o caso onde o item retirado não se encaixe em nenhuma das barras da solução, para que se coloque o item uma nova barra.

3.2.7 Elitismo

Como descrito na seção 2.6.6, elitismo consiste em pegar a melhor, ou certa porcentagem das melhores soluções da população atual e passá-las diretamente para a nova população. Neste caso optou-se por pegar uma porcentagem da população atual e passá-la para a nova população, e com exaustivos testes e variações de porcentagens observou-se que as melhores soluções encontradas foram geradas quando o algoritmo estava gerando a nova população com uma porcentagem entre 3% e 5%. Isto quer dizer, por exemplo, que os 5% melhores indivíduos da população atual são copiados diretamente na nova população. Os resultados apresentados na seção 4 foram obtidos com uma taxa de elitismo de 5%.

Lembrando que, na geração dos demais indivíduos da nova população, seja por cruzamento ou mutação, os indivíduos que foram passados diretamente para a nova população também estavam presentes.

3.3 Diferenças entre o Algoritmo Implementado e os Algoritmos Genéticos Convencionais

Este trabalho implementou o Algoritmo Genético descrito na seção 3.1, e este apresenta algumas diferenças com relação aos algoritmos genéticos convencionais. Uma diferença básica foi no cruzamento dos indivíduos, que na forma convencional dos dois pais selecionados surgem dois filhos para a nova população, no cruzamento implementado apenas o filho com maior aptidão vai para nova população, assim precisando chamar o operador de seleção o dobro de vezes do que o convencional.

Outra diferença básica foi que a mutação para estes tipos de problemas pode gerar indivíduos inviáveis e ao invés de descartar estes indivíduos “defeituosos” como se é feito convencionalmente, aplicou a eles uma correção, conforme descrita anteriormente. Este procedimento torna estes indivíduos novamente viáveis, e os insere na nova população. Neste caso, assim como no convencional, o novo indivíduo pode ter aptidão menor do que o indivíduo do qual foi gerado, o que é interessante para variação do espaço de busca.

3.4 Tecnologias e Padrões

No desenvolvimento deste trabalhou foi utilizada a linguagem Java, sobre a máquina virtual da empresa *Sun microsystems* (J2SDK 1.5.0) (J2SDE). A escolha desta linguagem se baseou no fato da Máquina Virtual Java oferecer um *framework* muito vasto já implementado, facilitando a codificação. Além disto, Java tem uma forte convicção na orientação a objetos, que torna a modelagem do sistema mais natural e fácil de se realizar. Outra vantagem de Java é que facilita a implementação em hardwares não-padrões como PDAs, celulares e plataformas proprietárias.

O sistema foi baseado na Convenção de Codificação Java (JCC), o que implica em uma facilidade de entendimento de outras pessoas. Além disso, todo o algoritmo foi implementado afim de garantir uma documentação do código fonte, sendo possível através do sistema (JavaDoc).

Os testes foram efetuados no sistema operacional: *Windows XP Professional* (Windows XP). O computador utilizado para a coleta dos resultados foi um Intel Pentium IV, 2.8 GHz e com 480 MB de memória RAM.

4 RESULTADOS

A melhor compreensão do Algoritmo Genético foi possível de maneira experimental. Parte do programa destinada à implementação do Algoritmo Genético foi feita de forma bem flexível, de maneira que pudessem ser testados diferentes valores dos parâmetros relacionados aos operadores genéticos.

A maior dificuldade encontrada em relação ao método foi a calibragem de seus parâmetros, esta calibragem foi feita através de uma bateria de testes. Assim, antes da coleta dos resultados foram feitos exaustivos testes para verificar a integridade e eficiência das soluções obtidas pelo sistema. Depois foram montadas algumas instâncias para a análise e demonstração dos resultados.

Visando um detalhamento para análise dos resultados procurou-se resolver instâncias que poderiam perfeitamente ser encontradas no mundo real, dentro da realidade de indústria que convivem com este problema.

Assim, de acordo com os dados dos problemas mais encontrados nas pesquisas feitas, foram criadas três classes de instâncias, **C1**, **C2**, e **C3** que são detalhadas na Tabela 3.4.1. Para cada uma destas classes de instância foram gerados problemas com 120, 500 e 1000 itens.

Tabela 3.4.1 - Dados das Classes de Instâncias.

Classe C*i*	Tamanho da barra	Tamanho máximo para um item	Tamanho mínimo para um item
C1	150	100	20
C2	350	240	45
C3	500	335	65

Assim para cada classe de instância e para cada quantidade de itens estabelecidas foram geradas instâncias aleatórias. Os resultados serão apresentados de acordo com o número de itens para cada classe de problemas.

Os dados apresentados para cada solução são o *fitness* (valor da função objetivo) do melhor indivíduo, número de barras utilizadas e a quantidade de barras com aproveitamento total.

Para uma maior organização do trabalho todas as instâncias geradas e utilizadas estão listadas no Apêndice A deste trabalho e estão separadas de acordo o número de itens de cada problema e a classe o qual pertence. No apêndice também tem uma descrição de como são organizadas as instâncias e na seção A.1 estão as instâncias geradas com 120 itens, na seção A.3 as instâncias com 500 itens e na seção A.4 as instâncias com 1000 itens.

Na Tabela 3.4.2 abaixo são mostrados os dados dos melhores resultados encontrados para cada uma das classes de instâncias com o problema sendo gerado com 120 itens:

Tabela 3.4.2 - Resultados para Instâncias com 120 Objetos.

Classe C*i*	**Melhor *Fitness* f(Si)**	**Número de barras utilizadas**	**Número de barras com aproveitamento Total**
C1	0.96396	47	34
C2	0.96534	50	21
C3	0.95667	52	19

A solução para instância **C3** com 120 objetos é apresentada em detalhes na seção B.2 deste trabalho, mostrando que mesmo a solução que obteve um *fitness* mais baixo para esta quantidade de itens ainda teve um bom aproveitamento e uma boa distribuição entre as barras utilizadas na solução.

Na Tabela 3.4.3 abaixo são mostrados os dados dos melhores resultados encontrados para cada uma das classes de instâncias com o problema sendo gerado com 500 itens:

Tabela 3.4.3 - Resultados para Instâncias com 500 Objetos.

Classe C*i*	**Melhor *Fitness* f(Si)**	**Número de barras utilizadas**	**Número de barras com aproveitamento Total**
C1	0.96423	205	94
C2	0.91772	211	29
C3	0.94726	206	49

Na Tabela 3.4.4 abaixo são mostrados os dados dos melhores resultados encontrados para cada uma das classes de instâncias com o problema sendo gerado com 1000 itens:

Tabela 3.4.4 - Resultados para Instâncias com 1000 Objetos.

Classe C*i*	**Melhor *Fitness* f(Si)**	**Número de barras utilizadas**	**Número de barras com aproveitamento Total**
C1	0.91663	421	66
C2	0.90647	427	35
C3	0.91273	418	29

Um detalhamento para a solução da instância **C1** é apresentada na seção B.3 deste trabalho.

5 CONCLUSÃO

O trabalho foi desenvolvido com o propósito de implementar um sistema computacional para minimizar os custos e o desperdício de material envolvido em determinado problema, onde se verificou que os resultados determinam uma boa maneira em que o conjunto de objetos é aproveitado, atingindo as especificações do projeto e aos objetivos propostos. Com isto o algoritmo genético mostrou-se eficiente para resolução dos problemas de corte e empacotamento unidimensional.

De acordo com as comparações dos resultados com outros trabalhos com mesmas classes de instâncias e estudo do detalhamento das soluções tiveram-se indícios de bons resultados para o algoritmo. Todos os resultados encontrados ficaram no intervalo de soluções encontradas em outros trabalhos.

Pôde-se observar que quanto maior a quantidade de itens e quanto maior a capacidade da barra as combinações que seja um somatório igual ao tamanho exato da barra fica mais difícil, o que faz com que uma proporção menor das barras sejam totalmente aproveitadas, e ainda, seu *fitness* fica menor. Ainda assim o algoritmo conseguiu achar uma boa distribuição de itens dentro das barras, assim quando o aproveitamento da barra não é total fica próximo disto, o que pode ser observado nos detalhamentos das soluções na seção B.2 e B.3 deste trabalho.

Após o desenvolvimento do mesmo, foram observadas algumas peculiaridades do método utilizado para implementação. Os Algoritmos Genéticos ajudam em termos de velocidade de processamento de soluções viáveis e freqüentemente, se não obtém a solução ótima, obtém uma solução próxima a ela. Com isto este algoritmo pode ser inserido no contexto de produção de varias indústrias para ajudar maximizar recursos implicando em maiores lucros a estas.

Convém também ressaltar que é inviável o uso deste sistema quando se trabalha com itens de tamanhos muito próximos ao tamanho da barra, pois este leva um tempo elevado para o processamento da solução, devido à dificuldade de obter soluções viáveis.

6 PROPOSTA PARA TRABALHOS FUTUROS

Como uma primeira proposta, pode-se implementar se outras heurísticas para que se possa ter uma comparação de resultados, e também de tempos da geração das soluções. Pode se também fazer uma análise mais rigorosa dos resultados.

Uma outra proposta de trabalho é a avaliação experimental rigorosa para determinar qual é a verdadeira influência de cada operador dentro do algoritmo genético, buscando novas probabilidades de aplicação destes operadores para alcançar melhores resultados.

Como este trabalho pode ser inserido no contexto de varias indústrias, uma proposta interessante é a de se desenvolver uma interface gráfica de uso amigável, e tentar inseri-lo em uma aplicação industrial real, assim ainda surge a opção e aprimorá-lo com detalhes, como armazenamento das soluções geradas e indicando se foram usadas, assim tendo também a possibilidade de gerar relatórios.

7 REFERÊNCIAS BIBLIOGRÁFICAS

ANDRADE, C.E.; BATISTA, F.L.C.; TOSO, R.F. Modelo de Otimização para Transporte de Cargas em Ambientes Reduzidos. 2004. Monografia Projeto Orientado (Monografia apresentada para obtenção do título em Bacharel em Ciência da Computação) – Universidade Federal de Lavras – Lavras.

ARENALES, M. N. (1993). Uma teoria para o problema de corte. Tese de Livre Docência, ICMSC-USP, São Carlos, S.P., Brasil.

CAMPOS, D. F. Introdução à pesquisa operacional. Apostila / Módulo I para o curso de Engenharia Mecânica da UFRN.

CANTÚ, P. E.(1995), A summary of research on Parallel Genetic Algorithms, IlliGAL Report, 95007, University of Illinois at Urbana- Champaign.

CINTRA, G.; WAKABAYASHI, Y. Dynamic Programming and Column Generation Based Approaches for Two-Dimensional Guillotine Cutting Problems, Lecture Notes in Computer Science, v.3059, p. 175 – 190, 2004.

CECILIO, F.O. (2003) Refinamentos na heurística de George e Robinson para o problema do carregamento de caixas dentro de contêineres. Departamento de Engenharia de Produção, Universidade Federal de São Carlos, São Carlos, SP, Brasil.

CORMEN 2002, T. H.; Leiserson, C. E.; Rivest, R. L; Stein, C. Algoritmos: Teoria e Prática. Tradução da Segunda Edição Americana por Vandenberd D. de Souza.Rio de Janeiro: Campus, 2002, p.763-807.

DE JONG, K. A. (Ed.) (1993), Evolutionary Computation, MIT Press.

DEB, K. [2001], Multi-Objective Optimization using Evolutionary Algorithms, John Wiley & Sons, New York.

DEVILLERS, J.(ed.) (1996), Genetic Algorithms in Molecular Modeling - Principes of Qsar and Drug Design 1, Academic Press.

DOWSLAND, K.A. e DOWSLAND, W.B. (1992) Packing Problems. European Journal of perational Research, v.56, p.2-14.

DYCKHOFF, H. e FINKE, U. (1992), "Cutting and Packing in Production and Distribution: Typology and Bibliography", Springer-Verlag Co, Heidelberg.

FURTADO, J. C. (1998). Algoritmo Genético Construtivo para Resolução de Problema de Agrupamentos, Tese de Doutorado, INPE.

GRAMANI, M. C. Minimizing the Number of Different Cutting patterns – a minimum path approach. In: EURO/INFORMS JOINT INTERNATIONAL MEETING. 2003, Istanbul, Turkey.

GEORGE, J.A. e ROBINSON, D.F. (1980) A Heuristic for Packing Boxes into a Container. *Computers & Operations Research*, v.7, p.147-156.

GOLDBERG, D. E. [1989], Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley Publishing Company, Inc., Reading, MA.

GOLDEN, B. L. (1976). Approaches to the cutting-stock problem. AIIE Transactions, 8(2), 265-274.

HOLLAND 1975. J. H. Holland; Adaptation in natural and artificial systems, (1975).

J2SE, Sun microsystems - Java Virtal Machine 5.0, URL: http://java.sun.com, 25 de Janeiro de 2006.

JavaDoc, How to Write Doc Comments for the Javadoc Tool, URL:http://java.sun.com/j2se/javadoc/writingdoccomments/, 25 de Janerio de 2006.

JCC, Code Conventions for the Java Programming Language, URL:http://java.sun.com/docs/codeconv/, 25 de Janeiro de 2006.

JOHNSON, 1979. Garey, M. R. e Johnson, D. S. "Computers and Intractability: A Guide to the Theory of NPCompleteness", W.H. Freeman and Company, São Francisco, 1979.

MARTINS, G.H. (2002) Packing in two and three dimensions. Monterey, CA, EUA. PhD Dissertation. Naval Postgraduate School.

MEDEIROS, O. M. (1998). Apostila Pesquisa Operacional. São Paulo, S.P., Brasil.

MITCHELL 96, Melanie. An Introduction to Genetic Algorithms. Massachusets: MIT Press, 1996.

MORABITO, R. e ARENALES, M. (1994) An And/Or-graph Approach to the Container Loading Problem. International Transactions in Operations Research, v.1, n.1, p.59-73.

PAPADIMITRIOU, 1982. PAPADIMITRIOU, C. H.; STEIGLITZ, K. Combinatorial Optimization: Algorithms and Complexity. USA, Dover Publications Inc., 1982.

RAO, S. S. "Optimization - Theory and Applications - Second Edition Wiley Eastern Limited, 1978.

RAUPP, M.P. Introdução à Otimização Linear, LNCC, Rio de Janeiro. Notas de Aulas, Curso de Verão LNCC, 2003.

SILVA, A. S. N. (2005). Estudo e Implementação, Mediante Têmpera Simulada do Problema de Alocação de Salas. Departamento de Ciência da Computação, Universidade Federal de Lavras, UFLA, Lavras, MG, Brasil

SILVA, J.L.C., SOMA, N.Y., MACULAN, N. (2003) A Greedy Search for the Three-Dimensional Bin Packing Problem: the Packing Static Stability Case. International Transactions in Operational Research, v.10, p.1-13.

TICONA, W. G. C. (2003) Aplicação de Algoritmos Genéticos Multi-Objetivo para Alinhamento de Seqüências Biológicas. Instituto de Ciências Matemáticas e de Computação - ICMC-USP, Universidade de São Paulo, São Paulo, SP, Brasil.

VIEIRA, V.G. (1999). Problema da Mochila. Trabalho de Graduação UFSM. Santa Maria, R.S., Brasil.

VINK, Marco. (1997). Solving Combinatorial Problems Using Evolutionary Algorithms.

WINDOWS XP Microsoft Windows XP Professional, URL:http://www.microsoft.com/windowsxp/pro/default.mspx/, 25 de Janeiro de 2006.

YANASSE, H.H.; FURTADO, J.C. at al. O Problema de Corte e Empacotamento e Aplicações Industriais. In: 2ª OFICINA NACIONAL DE PCE. XX CONGRESSO NACIONAL DE MATEMÁTICA APLICADA E COMPUTACIONAL. 1997, Gramado. Mini Curso... Gramado: 1997. 145p.

YEPES 2006, I. Uma incursão aos algoritmos genéticos URL: http://www.geocities.com/igoryepes/, 2006.

Apêndice A

Instâncias

A.1 Descrição do formato das instâncias

As instâncias estão separadas de acordo com a quantidade de itens para classe. Assim nas próximas seções serão apresentadas as instâncias utilizadas para as três classes **C1**, **C2** e **C3** com o problema tendo 120, 500 e 1000 itens. Na seção A.2 estão as instâncias com 120 itens, na A.3 as instâncias com 500 itens e na A.4 as instâncias com 1000 itens.

Para representação das instâncias será usada uma tabela com duas colunas, uma para indicar o tamanho do item e a outra para indicar a quantidade deste itens contido no problema. Na ultima linha da tabela há um total de número de itens de tamanho diferentes contido em cada instância.

A.2 Instâncias com 120 itens

As Tabelas Tabela A 1, Tabela A 2 e Tabela A 3 - Instância para Classe **C3** com 120 itens. representam os problemas gerados aleatoriamente para as classes de instâncias **C1**, **C2** e **C3**, respectivamente, com 500 itens.

Tabela A 1 - Instância para Classe **C1** com 120 itens.

Tamanho do Item	Quantidade de itens
84	4
80	1
70	1
25	2
87	2
36	2
61	2
75	5
23	3
62	3
68	2
32	2
74	3
65	4

64	3
26	5
54	2
40	2
52	3
93	3
59	3
89	2
71	1
27	1
63	2
67	2
81	1
69	2
24	4
96	2
41	1
35	2
82	3
29	1
38	4
43	1
48	1
33	2
34	2
95	2
50	4
20	2
53	1
91	2
37	2
46	1
22	1
79	1
92	1
30	1
66	1
39	1
58	1
99	1
60	1
28	1
97	1
55	1
86	1
77	1
49	1
Total 61 itens de tamanhos diferentes	**Total de 120 itens**

Tabela A 2 - Instância para Classe **C2** com 120 itens.

Tamanho do Item	Quantidade de itens
162	2
143	1
94	1
226	1
174	1
113	1
159	2
128	1
222	1
97	3
167	1
227	3
139	1
190	2
67	2
211	2
85	2
181	4
201	4
118	2
66	1
111	3
185	1
120	3
140	2
48	1
214	1
168	1
208	2
150	1
141	2
151	1
87	2
61	4
176	1
117	1
103	1
219	2
70	1
192	2
75	1
63	1
122	1
126	1
55	2
223	1
207	1
221	1
62	1
123	1
213	1
71	1
161	1
217	1

91	1
56	1
179	2
206	1
178	1
170	1
52	1
240	1
72	1
112	2
152	1
51	1
90	2
132	1
196	1
228	1
153	1
50	1
57	1
164	1
101	1
182	1
186	1
203	2
230	1
98	1
229	1
156	1
47	1
107	1
135	1
Total 85 itens de tamanhos diferentes	**Total de 120 itens**

Tabela A 3 - Instância para Classe **C3** com 120 itens.

Tamanho do Item	Quantidade de itens
229	1
211	1
151	1
123	2
136	2
279	1
271	1
296	3
305	1
324	2
330	1
246	1
145	3
291	1
266	2
164	1
220	1
153	1
113	1
313	2
150	3
174	1
178	1
195	2
205	1
250	1
130	1
244	2
96	1
170	1
295	1
262	1
184	1
95	1
260	1
199	1
256	5
297	1
79	1
189	4
331	1
323	1
322	1
292	1
127	1
121	3
165	1
65	3
333	1
131	1
328	1
118	1
175	2
314	1

135	1
193	1
190	1
143	1
219	1
299	1
325	1
284	1
304	1
188	1
154	2
311	2
255	1
142	1
126	1
285	1
281	2
222	1
265	1
86	1
257	1
272	1
210	1
287	1
321	1
332	1
270	1
187	1
105	1
124	1
158	2
74	1
303	1
134	1
107	1
267	1
104	1
Total 91 itens de tamanhos diferentes	**Total de 120 itens**

A.3 Instâncias com 500 itens

As Tabelas Tabela A 4, Tabela A 5 e Tabela A 6 representam os problemas gerados aleatoriamente para as classes de instâncias **C1**, **C2** e **C3**, respectivamente, com 500 itens.

Tabela A 4 - Instância para Classe **C1** com 500 itens.

Tamanho do Item	Quantidade de itens
59	9
49	9
74	8
28	5
91	11
84	5
26	6
70	6
37	8
71	10
78	5
51	8
46	15
92	2
33	12
23	6
47	9
66	5
52	9
53	3
82	5
69	8
85	2
97	6
73	6
24	5
68	4
21	3
58	5
48	6
67	7
61	4
72	4
57	9
95	11
56	5
60	9
44	7
99	3
20	7
63	8
87	6
90	8
86	5
77	7
98	7

36	2
75	2
81	12
65	11
25	6
50	12
55	5
41	3
35	4
34	5
45	10
89	3
62	8
93	5
32	5
94	3
27	8
76	3
79	9
80	7
64	8
31	4
96	11
22	4
88	4
29	7
40	2
83	5
39	3
42	3
100	3
38	6
43	3
30	4
54	2
Total 81 itens de tamanhos diferentes	**Total de 500 itens**

Tabela A 5 - Instância para Classe **C2** com 500 itens.

Tamanho do Item	Quantidade de itens
222	4
218	5
210	3
230	6
221	3
79	2
151	6
189	4
134	5
183	2
196	4
70	6
229	4
66	5
78	5
102	3
144	2
208	3
89	3
67	5
74	2
127	2
204	4
174	2
225	6
212	5
147	4
135	1
211	3
205	2
198	2
104	3
165	4
203	2
106	4
237	3
207	2
100	3
121	1
49	3
162	2
226	1
150	5
141	3
119	7
164	3
166	4
63	3
60	1
188	4
130	2
153	2
68	4
122	3
142	3

192	3
240	3
76	3
179	7
236	2
158	5
219	3
87	5
88	5
200	3
190	4
175	3
85	4
117	4
191	2
112	3
184	5
167	2
118	4
216	2
96	3
161	2
126	1
149	2
129	3
56	4
159	1
59	1
93	2
110	2
131	7
139	3
217	3
91	3
157	1
125	2
115	2
90	5
199	3
58	2
154	1
133	3
137	4
206	2
187	6
233	2
128	2
223	2
81	3
195	2
202	4
136	4
160	2
220	1
109	4
92	2
107	2
194	2
55	3

83	2
103	1
177	2
72	1
215	2
145	3
50	2
51	3
168	2
61	4
64	2
169	1
53	2
82	4
163	1
201	3
140	2
235	4
47	4
80	2
197	1
132	4
101	4
97	3
152	2
120	2
224	1
95	2
111	1
116	5
148	1
114	4
98	1
124	1
239	2
238	2
99	6
48	3
69	2
170	1
54	1
77	2
73	3
181	2
65	2
52	3
155	4
182	2
228	3
178	2
185	1
84	1
146	1
46	1
45	1
193	2
62	1
213	1
231	1

123	2
156	2
176	2
172	1
75	1
143	1
86	1
94	1
Total 181 itens de tamanhos diferentes	**Total de 500 itens**

Tabela A 6 - Instância para Classe **C3** com 500 itens.

Tamanho do Item	Quantidade de itens
277	2
200	1
95	2
223	3
312	6
228	2
98	4
166	2
140	1
317	5
116	4
253	6
103	4
225	2
230	1
77	2
300	4
208	3
291	3
172	3
194	6
143	3
335	2
105	2
250	3
299	1
197	4
297	6
163	4
138	2
269	3
180	3
212	3
117	2
125	4
66	4
86	2
72	3
213	2
146	1
241	2
301	1
278	3
318	2
254	5
136	3
122	1
177	6
333	3
187	2
332	2
270	4
129	3
88	4
319	2

110	4
226	2
97	3
115	4
74	1
272	1
167	3
242	1
175	2
150	2
181	2
322	3
211	2
144	4
135	1
231	2
276	2
119	2
206	1
182	1
152	5
124	3
274	3
261	2
109	1
265	1
324	3
153	3
259	2
202	2
90	2
173	2
94	2
263	2
161	4
326	3
100	1
237	1
198	1
234	1
102	2
148	2
246	2
69	3
137	2
314	3
170	2
114	4
289	1
256	2
147	3
310	2
104	3
204	2
248	3
171	1
295	1
267	1
235	3

145	3
193	4
67	2
271	2
131	1
134	1
121	1
128	1
327	1
247	2
304	3
96	3
308	1
229	1
286	5
273	3
168	2
99	1
266	1
222	4
283	3
255	3
107	2
139	1
309	2
155	3
76	5
323	2
158	1
87	2
290	1
133	1
219	3
216	2
240	3
302	2
195	2
92	3
294	5
238	3
260	1
81	1
142	2
78	1
85	2
79	1
282	1
296	1
179	1
84	6
210	2
298	1
209	2
232	1
108	1
118	1
279	2
91	2
316	1

264	2
71	2
68	1
130	1
329	3
192	2
159	1
123	3
249	1
162	1
321	2
220	2
243	2
127	1
75	1
275	2
80	1
215	2
65	1
331	2
113	1
151	2
268	1
257	1
311	2
239	1
293	2
196	1
244	1
236	1
176	1
154	1
178	1
186	2
288	1
183	5
201	1
245	1
101	2
132	1
258	1
306	1
189	1
305	1
285	1
224	1
252	2
160	1
214	1
126	1
185	2
221	1
112	4
313	1
227	1
203	1
303	1
330	1
169	1

141	1
Total 233 itens de tamanhos diferentes	**Total de 500 itens**

A.3 Instâncias com 1000 itens

As Tabelas Tabela A 4, Tabela A 5 e Tabela A 6 representam os problemas gerados aleatoriamente para as classes de instâncias **C1**, **C2** e **C3**, respectivamente, com 1000 itens.

Tabela A 7 - Instância para Classe **C1** com 1000 itens.

Tamanho do Item	Quantidade de itens
62	12
92	16
57	15
76	13
87	20
67	14
70	17
69	9
20	13
59	15
37	16
78	14
31	19
79	17
71	10
21	11
61	11
26	15
100	13
42	18
93	13
90	11
72	8
85	14
98	12
95	12
63	15
53	10
83	18
88	15
73	8
54	11
39	14
41	9
44	12
64	12
68	15
33	14
99	14
97	15
84	13
24	11
60	8
89	7
77	9
52	14

75	13
96	18
36	11
35	15
48	10
27	17
58	11
22	13
66	9
45	11
32	11
28	11
40	7
50	12
25	15
65	10
49	11
91	8
55	17
38	11
94	13
80	9
34	11
74	9
51	9
82	11
29	11
47	11
81	12
43	6
23	10
56	8
86	12
30	11
46	14
Total 81 itens de tamanhos diferentes	**Total de 1000 itens**

Tabela A 8 - Instância para Classe **C2** com 1000 itens.

Tamanho do Item	Quantidade de itens
150	8
208	5
133	8
151	4
131	7
71	2
54	11
155	7
46	6
80	6
87	6
170	6
230	8
69	5
123	4
165	8
184	6
172	4
237	3
78	3
90	6
211	4
227	4
216	9
58	5
56	6
198	6
204	5
115	4
195	5
116	3
120	11
207	6
138	3
194	7
60	6
72	2
103	10
157	2
108	7
148	5
163	2
96	7
222	7
110	5
147	4
67	4
64	7
139	13
239	3
185	8
121	5
73	9
191	5
130	4

236	2
89	7
171	6
161	6
218	6
124	4
109	6
101	4
234	8
197	6
77	12
49	6
182	8
135	3
233	5
100	7
142	3
166	5
136	6
231	8
61	5
51	4
212	5
190	5
97	6
158	7
196	4
102	6
205	7
223	8
92	6
173	4
47	3
99	4
159	4
127	8
220	9
52	6
193	6
145	7
232	5
79	6
177	5
93	8
63	4
83	6
70	5
178	9
188	5
45	8
132	5
152	3
164	3
75	9
74	6
225	6
134	4
105	8
156	5

224	5
228	10
106	5
179	5
215	4
126	2
85	5
238	8
86	10
240	5
210	5
76	3
199	8
169	3
82	5
203	7
65	3
226	8
235	4
187	8
209	4
113	3
48	5
214	7
137	4
192	4
57	5
94	9
168	5
183	5
140	5
143	2
162	3
66	5
53	4
146	5
91	7
107	2
149	1
125	3
202	7
154	4
213	4
50	5
174	5
144	3
160	3
112	3
175	4
129	5
55	6
84	3
98	3
81	6
119	6
200	4
104	3
153	3
114	4

180	4
111	8
128	2
186	2
181	2
189	3
62	4
141	5
201	6
59	4
122	3
68	2
95	1
118	2
229	2
206	2
221	1
176	3
117	3
217	1
Total 193 itens de tamanhos diferentes	**Total de 1000 itens**

Tabela A 9 - Instância para Classe **C3** com 1000 itens.

Tamanho do Item	Quantidade de itens
87	6
297	4
324	9
84	8
214	3
107	5
207	7
106	7
316	4
194	5
261	5
135	4
226	5
298	4
120	3
332	3
269	5
243	2
331	2
67	6
219	4
88	3
94	1
220	4
313	1
233	3
174	3
182	3
154	3
305	2
293	6
96	4
165	4
270	6
291	3
129	3
311	7
223	1
179	1
199	4
159	3
255	5
318	1
296	7
310	7
105	5
253	5
184	3
285	3
329	6
290	5
202	3
278	4
286	3

280	1
109	4
142	5
320	5
208	3
215	3
295	1
153	3
268	3
213	5
190	1
71	3
244	5
99	3
289	5
241	5
131	3
111	4
178	2
168	7
169	6
205	5
266	3
175	5
251	11
185	2
117	6
249	6
334	5
155	7
147	3
240	3
148	4
196	3
252	4
254	3
300	5
161	1
77	10
257	8
260	4
70	3
114	5
170	6
164	2
309	5
265	7
211	5
204	3
126	6
212	3
89	4
288	4
104	3
103	3
192	3
221	5
128	6
98	4

282	6
307	5
134	8
163	3
156	6
250	5
326	3
264	1
132	4
210	3
299	6
314	3
306	2
274	1
187	3
140	4
267	4
72	2
166	5
124	3
276	3
97	7
287	4
146	3
256	5
319	3
130	5
315	3
302	5
152	6
78	4
197	3
225	2
141	7
92	2
101	6
245	4
328	5
281	2
312	4
271	3
284	1
277	5
239	2
275	2
108	5
263	4
112	3
224	5
222	2
80	5
127	5
206	6
177	1
231	5
136	1
69	4
100	2
188	5

143	4
102	5
242	8
122	3
304	2
217	6
139	4
123	2
145	3
273	2
333	4
85	6
133	2
74	3
246	1
323	3
150	3
95	5
83	6
186	5
116	1
171	4
321	4
73	7
236	3
238	3
75	3
180	4
119	2
138	1
76	5
66	2
230	4
149	4
237	5
121	3
294	5
283	5
247	7
189	4
292	5
176	3
65	3
195	4
172	3
91	5
228	5
162	5
144	3
330	3
248	3
158	3
218	3
82	2
86	4
198	1
262	1
327	2
308	2

325	1
93	5
259	2
193	3
183	2
181	5
167	2
229	1
227	2
209	1
157	5
200	1
216	1
173	4
279	3
113	4
118	1
258	3
272	3
301	3
335	3
203	1
191	3
110	3
68	1
90	1
201	2
232	3
79	3
81	2
234	3
322	2
235	2
125	1
303	1
317	3
Total 267 itens de tamanhos diferentes	**Total de 1000 itens**

Apêndice B

Melhores Indivíduos

B.1 Descrição do formato dos indivíduos

No Apêndice B serão mostrados detalhes de duas das soluções encontradas no trabalho, são a instância da classe C3 com 120 itens e a instância da classe C1 com 1000 itens. Estes detalhes são a respeito da distribuição de itens nas barras.

Os dados serão mostrados como saem do programa, são mostradas como ficaram as ocupações para as barras utilizadas na solução. Os dados serão mostrados em colunas para uma maior organização do trabalho.

B.2 Detalhamento da solução da Classe C3 com 120 itens

```
-------------------------
Melhor Indivíduo
Fitness: 0.95667
-------------------------
Barra 1: 244  256
Soma da Barra 1: 500
-------------------------
Barra 2: 222  267
Soma da Barra 2: 489
-------------------------
Barra 3: 195  297
Soma da Barra 3: 492
-------------------------
Barra 4: 219  281
Soma da Barra 4: 500
-------------------------
Barra 5: 313  187
Soma da Barra 5: 500
-------------------------
Barra 6: 244  256
Soma da Barra 6: 500
-------------------------
Barra 7: 220  265
Soma da Barra 7: 485
-------------------------
Barra 8: 211  285
Soma da Barra 8: 496
-------------------------
Barra 9: 130  127  104
Soma da Barra 9: 361
-------------------------
Barra 10: 313  184
Soma da Barra 10: 497
-------------------------
Barra 11: 118  255  126
Soma da Barra 11: 499
-------------------------
Barra 12: 123  256  121
Soma da Barra 12: 500
-------------------------
Barra 13: 266  150
Soma da Barra 13: 416
-------------------------
Barra 14: 266  113  121
Soma da Barra 14: 500
-------------------------
Barra 15: 178  322
Soma da Barra 15: 500
-------------------------
Barra 16: 260  134
Soma da Barra 16: 394
-------------------------
Barra 17: 142  86  270
Soma da Barra 17: 498
-------------------------
-------------------------
Barra 18: 136  79  281
Soma da Barra 18: 496
-------------------------
Barra 19: 150  314
Soma da Barra 19: 464
-------------------------
Barra 20: 195  292
Soma da Barra 20: 487
-------------------------
Barra 21: 246  250
Soma da Barra 21: 496
-------------------------
Barra 22: 205  295
Soma da Barra 22: 500
-------------------------
Barra 23: 296  188
Soma da Barra 23: 484
-------------------------
Barra 24: 256  135  105
Soma da Barra 24: 496
-------------------------
Barra 25: 190  303
Soma da Barra 25: 493
-------------------------
Barra 26: 296  189
Soma da Barra 26: 485
-------------------------
Barra 27: 151  65  284
Soma da Barra 27: 500
-------------------------
Barra 28: 210  287
Soma da Barra 28: 497
-------------------------
Barra 29: 153  321
Soma da Barra 29: 474
-------------------------
Barra 30: 136  257  107
Soma da Barra 30: 500
-------------------------
Barra 31: 164  332
Soma da Barra 31: 496
-------------------------
Barra 32: 174  323
Soma da Barra 32: 497
-------------------------
Barra 33: 291  150
Soma da Barra 33: 441
-------------------------
Barra 34: 262  95  143
Soma da Barra 34: 500
-------------------------
```

```
--------------------------
Barra 35: 123  256  121
Soma da Barra 35: 500
--------------------------
Barra 36: 229  271
Soma da Barra 36: 500
--------------------------
Barra 37: 305  193
Soma da Barra 37: 498
--------------------------
Barra 38: 296  189
Soma da Barra 38: 485
--------------------------
Barra 39: 324  175
Soma da Barra 39: 499
--------------------------
Barra 40: 324  154
Soma da Barra 40: 478
--------------------------
Barra 41: 189  311
Soma da Barra 41: 500
--------------------------
Barra 42: 333  158
Soma da Barra 42: 491
--------------------------
Barra 43: 189  311
Soma da Barra 43: 500
--------------------------
Barra 44: 175  325
Soma da Barra 44: 500
--------------------------
Barra 45: 170  328
Soma da Barra 45: 498
--------------------------
Barra 46: 65  131  304
Soma da Barra 46: 500
--------------------------
Barra 47: 331  165
Soma da Barra 47: 496
--------------------------
Barra 48: 154  272  74
Soma da Barra 48: 500
--------------------------
Barra 49: 199  299
Soma da Barra 49: 498
--------------------------
Barra 50: 145  145  145  65
Soma da Barra 50: 500
--------------------------
Barra 51: 279  96  124
Soma da Barra 51: 499
--------------------------
Barra 52: 330  158
Soma da Barra 52: 488
---------------------------
```

B.3 Detalhamento da solução da Classe C1 com 1000 itens

```
--------------------------
Melhor Indivíduo
Fitness: 0.91663
--------------------------
Barra 1: 83  52
Soma da Barra 1: 135
--------------------------
Barra 2: 31  100
Soma da Barra 2: 131
--------------------------
Barra 3: 93  43
Soma da Barra 3: 136
--------------------------
Barra 4: 44  82
Soma da Barra 4: 126
--------------------------
Barra 5: 70  39
Soma da Barra 5: 109
--------------------------
Barra 6: 49  94
Soma da Barra 6: 143
--------------------------
Barra 7: 37  71  30
Soma da Barra 7: 138
--------------------------
Barra 8: 36  94
Soma da Barra 8: 130
--------------------------
Barra 9: 42  53  47
Soma da Barra 9: 142
--------------------------
Barra 10: 31  28  86
Soma da Barra 10: 145
--------------------------
Barra 11: 36  27  82
Soma da Barra 11: 145
--------------------------
Barra 12: 57  31  29
Soma da Barra 12: 117
--------------------------
Barra 13: 44  55  46
Soma da Barra 13: 145
--------------------------
Barra 14: 52  81
Soma da Barra 14: 133
--------------------------
Barra 15: 96  29
Soma da Barra 15: 125
--------------------------
Barra 16: 83  55
Soma da Barra 16: 138
--------------------------
Barra 17: 93  43
Soma da Barra 17: 136
--------------------------
Barra 18: 37  60  45
Soma da Barra 18: 142
--------------------------
Barra 19: 64  75
Soma da Barra 19: 139
--------------------------
Barra 20: 98  51
Soma da Barra 20: 149
--------------------------
Barra 21: 92  34
Soma da Barra 21: 126
--------------------------
Barra 22: 75  56
Soma da Barra 22: 131
--------------------------
Barra 23: 95
Soma da Barra 23: 95
--------------------------
Barra 24: 54  94
Soma da Barra 24: 148
--------------------------
Barra 25: 92  47
Soma da Barra 25: 139
--------------------------
Barra 26: 71  68
Soma da Barra 26: 139
--------------------------
Barra 27: 85  64
Soma da Barra 27: 149
--------------------------
Barra 28: 100  50
Soma da Barra 28: 150
--------------------------
Barra 29: 87  61
Soma da Barra 29: 148
--------------------------
Barra 30: 52  94
Soma da Barra 30: 146
--------------------------
Barra 31: 79  55
Soma da Barra 31: 134
--------------------------
Barra 32: 63  65
Soma da Barra 32: 128
--------------------------
Barra 33: 37  33  80
Soma da Barra 33: 150
--------------------------
Barra 34: 85  64
Soma da Barra 34: 149
--------------------------
Barra 35: 57  89
Soma da Barra 35: 146
```

```
--------------------------
Barra 36: 57  93
Soma da Barra 36: 150
--------------------------
Barra 37: 84  66
Soma da Barra 37: 150
--------------------------
Barra 38: 83  64
Soma da Barra 38: 147
--------------------------
Barra 39: 58  91
Soma da Barra 39: 149
--------------------------
Barra 40: 79  63
Soma da Barra 40: 142
--------------------------
Barra 41: 83  66
Soma da Barra 41: 149
--------------------------
Barra 42: 57  88
Soma da Barra 42: 145
--------------------------
Barra 43: 71  77
Soma da Barra 43: 148
--------------------------
Barra 44: 95  32
Soma da Barra 44: 127
--------------------------
Barra 45: 28  29  81
Soma da Barra 45: 138
--------------------------
Barra 46: 39  52  32  23
Soma da Barra 46: 146
--------------------------
Barra 47: 97  36
Soma da Barra 47: 133
--------------------------
Barra 48: 53  41  49
Soma da Barra 48: 143
--------------------------
Barra 49: 24  75  48
Soma da Barra 49: 147
--------------------------
Barra 50: 61  73
Soma da Barra 50: 134
--------------------------
Barra 51: 99  40
Soma da Barra 51: 139
--------------------------
Barra 52: 62  55  23
Soma da Barra 52: 140
--------------------------
Barra 53: 92  58
Soma da Barra 53: 150
--------------------------
Barra 54: 96  45
Soma da Barra 54: 141
--------------------------
Barra 55: 61  48  23
Soma da Barra 55: 132
--------------------------
Barra 56: 62  80
Soma da Barra 56: 142
--------------------------
Barra 57: 99  45
Soma da Barra 57: 144
--------------------------
Barra 58: 60  22  51
Soma da Barra 58: 133
--------------------------
Barra 59: 26  96  27
Soma da Barra 59: 149
--------------------------
Barra 60: 52  55  38
Soma da Barra 60: 145
--------------------------
Barra 61: 79  48  23
Soma da Barra 61: 150
--------------------------
Barra 62: 92  48
Soma da Barra 62: 140
--------------------------
Barra 63: 62  80
Soma da Barra 63: 142
--------------------------
Barra 64: 70  72
Soma da Barra 64: 142
--------------------------
Barra 65: 37  94
Soma da Barra 65: 131
--------------------------
Barra 66: 54  77
Soma da Barra 66: 131
--------------------------
Barra 67: 95  27  23
Soma da Barra 67: 145
--------------------------
Barra 68: 59  83
Soma da Barra 68: 142
--------------------------
Barra 69: 35  94
Soma da Barra 69: 129
--------------------------
Barra 70: 79  25  38
Soma da Barra 70: 142
--------------------------
Barra 71: 54  80
Soma da Barra 71: 134
--------------------------
Barra 72: 96  51
Soma da Barra 72: 147
--------------------------
Barra 73: 35  28  81
Soma da Barra 73: 144
--------------------------
Barra 74: 87  37  24
Soma da Barra 74: 148
--------------------------
Barra 75: 87  31  28
Soma da Barra 75: 146
```

```
--------------------------
Barra 76: 66  34  34
Soma da Barra 76: 134
--------------------------
Barra 77: 92  22  30
Soma da Barra 77: 144
--------------------------
Barra 78: 92  55
Soma da Barra 78: 147
--------------------------
Barra 79: 85  33  25
Soma da Barra 79: 143
--------------------------
Barra 80: 73  74
Soma da Barra 80: 147
--------------------------
Barra 81: 97  51
Soma da Barra 81: 148
--------------------------
Barra 82: 76  74
Soma da Barra 82: 150
--------------------------
Barra 83: 96  49
Soma da Barra 83: 145
--------------------------
Barra 84: 41  91
Soma da Barra 84: 132
--------------------------
Barra 85: 58  74
Soma da Barra 85: 132
--------------------------
Barra 86: 62  83
Soma da Barra 86: 145
--------------------------
Barra 87: 32  29  86
Soma da Barra 87: 147
--------------------------
Barra 88: 96  50
Soma da Barra 88: 146
--------------------------
Barra 89: 83  64
Soma da Barra 89: 147
--------------------------
Barra 90: 37  63  38
Soma da Barra 90: 138
--------------------------
Barra 91: 64  74
Soma da Barra 91: 138
--------------------------
Barra 92: 85  52
Soma da Barra 92: 137
--------------------------
Barra 93: 64  82
Soma da Barra 93: 146
--------------------------
Barra 94: 68  75
Soma da Barra 94: 143
--------------------------
Barra 95: 95  54
Soma da Barra 95: 149
```

```
--------------------------
Barra 96: 56  86
Soma da Barra 96: 142
--------------------------
Barra 97: 65  74
Soma da Barra 97: 139
--------------------------
Barra 98: 100  47
Soma da Barra 98: 147
--------------------------
Barra 99: 67  78
Soma da Barra 99: 145
--------------------------
Barra 100: 96  50
Soma da Barra 100: 146
--------------------------
Barra 101: 69  26  27  28
Soma da Barra 101: 150
--------------------------
Barra 102: 76  70
Soma da Barra 102: 146
--------------------------
Barra 103: 90  39
Soma da Barra 103: 129
--------------------------
Barra 104: 100  50
Soma da Barra 104: 150
--------------------------
Barra 105: 93  46
Soma da Barra 105: 139
--------------------------
Barra 106: 92  42
Soma da Barra 106: 134
--------------------------
Barra 107: 92  55
Soma da Barra 107: 147
--------------------------
Barra 108: 87  49
Soma da Barra 108: 136
--------------------------
Barra 109: 66  82
Soma da Barra 109: 148
--------------------------
Barra 110: 79  68
Soma da Barra 110: 147
--------------------------
Barra 111: 70  78
Soma da Barra 111: 148
--------------------------
Barra 112: 93  55
Soma da Barra 112: 148
--------------------------
Barra 113: 90  54
Soma da Barra 113: 144
--------------------------
Barra 114: 38  74  30
Soma da Barra 114: 142
--------------------------
Barra 115: 26  88  35
Soma da Barra 115: 149
```

```
--------------------------
Barra 116: 26  90  33
Soma da Barra 116: 149
--------------------------
Barra 117: 100  41
Soma da Barra 117: 141
--------------------------
Barra 118: 59  88
Soma da Barra 118: 147
--------------------------
Barra 119: 99  45
Soma da Barra 119: 144
--------------------------
Barra 120: 83  47
Soma da Barra 120: 130
--------------------------
Barra 121: 63  82
Soma da Barra 121: 145
--------------------------
Barra 122: 83  66
Soma da Barra 122: 149
--------------------------
Barra 123: 59  26  58
Soma da Barra 123: 143
--------------------------
Barra 124: 88  41
Soma da Barra 124: 129
--------------------------
Barra 125: 96  23  30
Soma da Barra 125: 149
--------------------------
Barra 126: 62  78
Soma da Barra 126: 140
--------------------------
Barra 127: 79  65
Soma da Barra 127: 144
--------------------------
Barra 128: 96  40
Soma da Barra 128: 136
--------------------------
Barra 129: 42  33  75
Soma da Barra 129: 150
--------------------------
Barra 130: 76  37  31
Soma da Barra 130: 144
--------------------------
Barra 131: 26  32  82
Soma da Barra 131: 140
--------------------------
Barra 132: 61  86
Soma da Barra 132: 147
--------------------------
Barra 133: 76  68
Soma da Barra 133: 144
--------------------------
Barra 134: 88  44
Soma da Barra 134: 132
--------------------------
Barra 135: 62  78
Soma da Barra 135: 140
--------------------------
Barra 136: 31  85  27
Soma da Barra 136: 143
--------------------------
Barra 137: 76  61
Soma da Barra 137: 137
--------------------------
Barra 138: 99  47
Soma da Barra 138: 146
--------------------------
Barra 139: 68  77
Soma da Barra 139: 145
--------------------------
Barra 140: 31  90  27
Soma da Barra 140: 148
--------------------------
Barra 141: 92  40
Soma da Barra 141: 132
--------------------------
Barra 142: 78  33  36
Soma da Barra 142: 147
--------------------------
Barra 143: 67  79
Soma da Barra 143: 146
--------------------------
Barra 144: 20  59  71
Soma da Barra 144: 150
--------------------------
Barra 145: 87  63
Soma da Barra 145: 150
--------------------------
Barra 146: 55  81
Soma da Barra 146: 136
--------------------------
Barra 147: 100  46
Soma da Barra 147: 146
--------------------------
Barra 148: 78  21  24  27
Soma da Barra 148: 150
--------------------------
Barra 149: 70  42  27
Soma da Barra 149: 139
--------------------------
Barra 150: 69  78
Soma da Barra 150: 147
--------------------------
Barra 151: 21  90  38
Soma da Barra 151: 149
--------------------------
Barra 152: 59  86
Soma da Barra 152: 145
--------------------------
Barra 153: 26  95  29
Soma da Barra 153: 150
--------------------------
Barra 154: 37  37  33  35
Soma da Barra 154: 142
--------------------------
Barra 155: 97  24  28
Soma da Barra 155: 149
```

```
--------------------------
Barra 156: 87  21  42
Soma da Barra 156: 150
--------------------------
Barra 157: 67  79
Soma da Barra 157: 146
--------------------------
Barra 158: 90  55
Soma da Barra 158: 145
--------------------------
Barra 159: 69  77
Soma da Barra 159: 146
--------------------------
Barra 160: 20  68  60
Soma da Barra 160: 148
--------------------------
Barra 161: 67  82
Soma da Barra 161: 149
--------------------------
Barra 162: 76  70
Soma da Barra 162: 146
--------------------------
Barra 163: 44  96
Soma da Barra 163: 140
--------------------------
Barra 164: 36  22  45  47
Soma da Barra 164: 150
--------------------------
Barra 165: 64  84
Soma da Barra 165: 148
--------------------------
Barra 166: 92  20  37
Soma da Barra 166: 149
--------------------------
Barra 167: 92  57
Soma da Barra 167: 149
--------------------------
Barra 168: 67  77
Soma da Barra 168: 144
--------------------------
Barra 169: 57  84
Soma da Barra 169: 141
--------------------------
Barra 170: 53  97
Soma da Barra 170: 150
--------------------------
Barra 171: 84  24  40
Soma da Barra 171: 148
--------------------------
Barra 172: 37  33  36  35
Soma da Barra 172: 141
--------------------------
Barra 173: 69  79
Soma da Barra 173: 148
--------------------------
Barra 174: 58  91
Soma da Barra 174: 149
--------------------------
Barra 175: 100  38
Soma da Barra 175: 138
--------------------------
Barra 176: 76  55
Soma da Barra 176: 131
--------------------------
Barra 177: 39  96
Soma da Barra 177: 135
--------------------------
Barra 178: 42  55  46
Soma da Barra 178: 143
--------------------------
Barra 179: 25  91  34
Soma da Barra 179: 150
--------------------------
Barra 180: 82  56
Soma da Barra 180: 138
--------------------------
Barra 181: 75  66
Soma da Barra 181: 141
--------------------------
Barra 182: 97  47
Soma da Barra 182: 144
--------------------------
Barra 183: 67  83
Soma da Barra 183: 150
--------------------------
Barra 184: 88  43
Soma da Barra 184: 131
--------------------------
Barra 185: 45  94
Soma da Barra 185: 139
--------------------------
Barra 186: 66  80
Soma da Barra 186: 146
--------------------------
Barra 187: 69  81
Soma da Barra 187: 150
--------------------------
Barra 188: 97  46
Soma da Barra 188: 143
--------------------------
Barra 189: 26  44  74
Soma da Barra 189: 144
--------------------------
Barra 190: 52  86
Soma da Barra 190: 138
--------------------------
Barra 191: 59  80
Soma da Barra 191: 139
--------------------------
Barra 192: 87  21  42
Soma da Barra 192: 150
--------------------------
Barra 193: 21  93  36
Soma da Barra 193: 150
--------------------------
Barra 194: 71  77
Soma da Barra 194: 148
--------------------------
Barra 195: 56  86
Soma da Barra 195: 142
```

```
-------------------------
Barra 196: 73  77
Soma da Barra 196: 150
-------------------------
Barra 197: 99  38
Soma da Barra 197: 137
-------------------------
Barra 198: 63  84
Soma da Barra 198: 147
-------------------------
Barra 199: 88  44
Soma da Barra 199: 132
-------------------------
Barra 200: 72  75
Soma da Barra 200: 147
-------------------------
Barra 201: 32  25  81
Soma da Barra 201: 138
-------------------------
Barra 202: 87  33  25
Soma da Barra 202: 145
-------------------------
Barra 203: 42  96
Soma da Barra 203: 138
-------------------------
Barra 204: 27  65  51
Soma da Barra 204: 143
-------------------------
Barra 205: 61  81
Soma da Barra 205: 142
-------------------------
Barra 206: 70  77
Soma da Barra 206: 147
-------------------------
Barra 207: 92  55
Soma da Barra 207: 147
-------------------------
Barra 208: 100  42
Soma da Barra 208: 142
-------------------------
Barra 209: 85  33  27
Soma da Barra 209: 145
-------------------------
Barra 210: 59  91
Soma da Barra 210: 150
-------------------------
Barra 211: 57  86
Soma da Barra 211: 143
-------------------------
Barra 212: 85  65
Soma da Barra 212: 150
-------------------------
Barra 213: 59  21  68
Soma da Barra 213: 148
-------------------------
Barra 214: 70  79
Soma da Barra 214: 149
-------------------------
Barra 215: 63  83
Soma da Barra 215: 146
-------------------------
Barra 216: 88  58
Soma da Barra 216: 146
-------------------------
Barra 217: 69  75
Soma da Barra 217: 144
-------------------------
Barra 218: 98  47
Soma da Barra 218: 145
-------------------------
Barra 219: 93  22  30
Soma da Barra 219: 145
-------------------------
Barra 220: 39  99
Soma da Barra 220: 138
-------------------------
Barra 221: 90  35  23
Soma da Barra 221: 148
-------------------------
Barra 222: 57  39  52
Soma da Barra 222: 148
-------------------------
Barra 223: 78  26  43
Soma da Barra 223: 147
-------------------------
Barra 224: 33  33  84
Soma da Barra 224: 150
-------------------------
Barra 225: 67  81
Soma da Barra 225: 148
-------------------------
Barra 226: 57  87
Soma da Barra 226: 144
-------------------------
Barra 227: 88  56
Soma da Barra 227: 144
-------------------------
Barra 228: 98  48
Soma da Barra 228: 146
-------------------------
Barra 229: 69  75
Soma da Barra 229: 144
-------------------------
Barra 230: 75  75
Soma da Barra 230: 150
-------------------------
Barra 231: 77  36  29
Soma da Barra 231: 142
-------------------------
Barra 232: 42  42  40  25
Soma da Barra 232: 149
-------------------------
Barra 233: 70  20  60
Soma da Barra 233: 150
-------------------------
Barra 234: 61  89
Soma da Barra 234: 150
-------------------------
Barra 235: 53  97
Soma da Barra 235: 150
```

```
--------------------------
Barra 236: 76  69
Soma da Barra 236: 145
--------------------------
Barra 237: 92  41
Soma da Barra 237: 133
--------------------------
Barra 238: 64  84
Soma da Barra 238: 148
--------------------------
Barra 239: 58  81
Soma da Barra 239: 139
--------------------------
Barra 240: 96  45
Soma da Barra 240: 141
--------------------------
Barra 241: 31  22  91
Soma da Barra 241: 144
--------------------------
Barra 242: 67  66
Soma da Barra 242: 133
--------------------------
Barra 243: 92  33  24
Soma da Barra 243: 149
--------------------------
Barra 244: 88  54
Soma da Barra 244: 142
--------------------------
Barra 245: 97  52
Soma da Barra 245: 149
--------------------------
Barra 246: 63  84
Soma da Barra 246: 147
--------------------------
Barra 247: 97  43
Soma da Barra 247: 140
--------------------------
Barra 248: 84  65
Soma da Barra 248: 149
--------------------------
Barra 249: 42  63  36
Soma da Barra 249: 141
--------------------------
Barra 250: 83  33  29
Soma da Barra 250: 145
--------------------------
Barra 251: 68  25  55
Soma da Barra 251: 148
--------------------------
Barra 252: 83  64
Soma da Barra 252: 147
--------------------------
Barra 253: 57  93
Soma da Barra 253: 150
--------------------------
Barra 254: 31  21  97
Soma da Barra 254: 149
--------------------------
Barra 255: 59  63  25
Soma da Barra 255: 147
```

```
--------------------------
Barra 256: 33  24  82
Soma da Barra 256: 139
--------------------------
Barra 257: 76  72
Soma da Barra 257: 148
--------------------------
Barra 258: 60  89
Soma da Barra 258: 149
--------------------------
Barra 259: 95  50
Soma da Barra 259: 145
--------------------------
Barra 260: 98  50
Soma da Barra 260: 148
--------------------------
Barra 261: 73  73
Soma da Barra 261: 146
--------------------------
Barra 262: 76  72
Soma da Barra 262: 148
--------------------------
Barra 263: 26  98  24
Soma da Barra 263: 148
--------------------------
Barra 264: 39  44  64
Soma da Barra 264: 147
--------------------------
Barra 265: 98  22  28
Soma da Barra 265: 148
--------------------------
Barra 266: 98  34
Soma da Barra 266: 132
--------------------------
Barra 267: 97  46
Soma da Barra 267: 143
--------------------------
Barra 268: 70  78
Soma da Barra 268: 148
--------------------------
Barra 269: 76  26  48
Soma da Barra 269: 150
--------------------------
Barra 270: 63  35  50
Soma da Barra 270: 148
--------------------------
Barra 271: 70  63
Soma da Barra 271: 133
--------------------------
Barra 272: 59  89
Soma da Barra 272: 148
--------------------------
Barra 273: 63  83
Soma da Barra 273: 146
--------------------------
Barra 274: 41  27  81
Soma da Barra 274: 149
--------------------------
Barra 275: 85  63
Soma da Barra 275: 148
```

```
--------------------------
Barra 276: 96  47
Soma da Barra 276: 143
--------------------------
Barra 277: 53  25  47  23
Soma da Barra 277: 148
--------------------------
Barra 278: 87  34  29
Soma da Barra 278: 150
--------------------------
Barra 279: 98  49
Soma da Barra 279: 147
--------------------------
Barra 280: 45  86
Soma da Barra 280: 131
--------------------------
Barra 281: 97  35
Soma da Barra 281: 132
--------------------------
Barra 282: 42  84  24
Soma da Barra 282: 150
--------------------------
Barra 283: 99  46
Soma da Barra 283: 145
--------------------------
Barra 284: 76  68
Soma da Barra 284: 144
--------------------------
Barra 285: 99  46
Soma da Barra 285: 145
--------------------------
Barra 286: 73  38  30
Soma da Barra 286: 141
--------------------------
Barra 287: 59  80
Soma da Barra 287: 139
--------------------------
Barra 288: 39  96
Soma da Barra 288: 135
--------------------------
Barra 289: 61  85
Soma da Barra 289: 146
--------------------------
Barra 290: 52  94
Soma da Barra 290: 146
--------------------------
Barra 291: 69  31  48
Soma da Barra 291: 148
--------------------------
Barra 292: 63  82
Soma da Barra 292: 145
--------------------------
Barra 293: 31  26  93
Soma da Barra 293: 150
--------------------------
Barra 294: 54  68  27
Soma da Barra 294: 149
--------------------------
Barra 295: 84  58
Soma da Barra 295: 142
--------------------------
Barra 296: 85  46
Soma da Barra 296: 131
--------------------------
Barra 297: 39  97
Soma da Barra 297: 136
--------------------------
Barra 298: 20  78  50
Soma da Barra 298: 148
--------------------------
Barra 299: 79  22  49
Soma da Barra 299: 150
--------------------------
Barra 300: 70  68
Soma da Barra 300: 138
--------------------------
Barra 301: 67  74
Soma da Barra 301: 141
--------------------------
Barra 302: 87  61
Soma da Barra 302: 148
--------------------------
Barra 303: 78  68
Soma da Barra 303: 146
--------------------------
Barra 304: 87  59
Soma da Barra 304: 146
--------------------------
Barra 305: 37  78  35
Soma da Barra 305: 150
--------------------------
Barra 306: 67  58  25
Soma da Barra 306: 150
--------------------------
Barra 307: 93  49
Soma da Barra 307: 142
--------------------------
Barra 308: 67  65
Soma da Barra 308: 132
--------------------------
Barra 309: 31  95  22
Soma da Barra 309: 148
--------------------------
Barra 310: 54  96
Soma da Barra 310: 150
--------------------------
Barra 311: 79  68
Soma da Barra 311: 147
--------------------------
Barra 312: 99  50
Soma da Barra 312: 149
--------------------------
Barra 313: 53  96
Soma da Barra 313: 149
--------------------------
Barra 314: 89  45
Soma da Barra 314: 134
--------------------------
Barra 315: 67  83
Soma da Barra 315: 150
```

```
--------------------------
Barra 316: 89  55
Soma da Barra 316: 144
--------------------------
Barra 317: 59  90
Soma da Barra 317: 149
--------------------------
Barra 318: 85  65
Soma da Barra 318: 150
--------------------------
Barra 319: 66  74
Soma da Barra 319: 140
--------------------------
Barra 320: 20  79  51
Soma da Barra 320: 150
--------------------------
Barra 321: 56  86
Soma da Barra 321: 142
--------------------------
Barra 322: 87  61
Soma da Barra 322: 148
--------------------------
Barra 323: 70  79
Soma da Barra 323: 149
--------------------------
Barra 324: 59  88
Soma da Barra 324: 147
--------------------------
Barra 325: 100  42
Soma da Barra 325: 142
--------------------------
Barra 326: 98  49
Soma da Barra 326: 147
--------------------------
Barra 327: 68  28  50
Soma da Barra 327: 146
--------------------------
Barra 328: 71  26  28  23
Soma da Barra 328: 148
--------------------------
Barra 329: 93  35  22
Soma da Barra 329: 150
--------------------------
Barra 330: 79  71
Soma da Barra 330: 150
--------------------------
Barra 331: 93  46
Soma da Barra 331: 139
--------------------------
Barra 332: 20  39  91
Soma da Barra 332: 150
--------------------------
Barra 333: 64  80
Soma da Barra 333: 144
--------------------------
Barra 334: 88  35  25
Soma da Barra 334: 148
--------------------------
Barra 335: 37  32  80
Soma da Barra 335: 149
--------------------------
Barra 336: 99  46
Soma da Barra 336: 145
--------------------------
Barra 337: 62  82
Soma da Barra 337: 144
--------------------------
Barra 338: 52  94
Soma da Barra 338: 146
--------------------------
Barra 339: 24  27  94
Soma da Barra 339: 145
--------------------------
Barra 340: 94  56
Soma da Barra 340: 150
--------------------------
Barra 341: 57  31  27  34
Soma da Barra 341: 149
--------------------------
Barra 342: 70  73
Soma da Barra 342: 143
--------------------------
Barra 343: 99  45
Soma da Barra 343: 144
--------------------------
Barra 344: 42  39  32  25
Soma da Barra 344: 138
--------------------------
Barra 345: 21  60  40  29
Soma da Barra 345: 150
--------------------------
Barra 346: 31  25  91
Soma da Barra 346: 147
--------------------------
Barra 347: 95  44
Soma da Barra 347: 139
--------------------------
Barra 348: 59  84
Soma da Barra 348: 143
--------------------------
Barra 349: 90  52
Soma da Barra 349: 142
--------------------------
Barra 350: 67  71
Soma da Barra 350: 138
--------------------------
Barra 351: 99  49
Soma da Barra 351: 148
--------------------------
Barra 352: 20  31  53  41
Soma da Barra 352: 145
--------------------------
Barra 353: 99  48
Soma da Barra 353: 147
--------------------------
Barra 354: 37  100
Soma da Barra 354: 137
--------------------------
Barra 355: 67  42  34
Soma da Barra 355: 143
```

```
--------------------------
Barra 356: 92  46
Soma da Barra 356: 138
--------------------------
Barra 357: 57  87
Soma da Barra 357: 144
--------------------------
Barra 358: 78  71
Soma da Barra 358: 149
--------------------------
Barra 359: 70  42  35
Soma da Barra 359: 147
--------------------------
Barra 360: 85  46
Soma da Barra 360: 131
--------------------------
Barra 361: 87  39  22
Soma da Barra 361: 148
--------------------------
Barra 362: 87  50
Soma da Barra 362: 137
--------------------------
Barra 363: 96  49
Soma da Barra 363: 145
--------------------------
Barra 364: 26  28  94
Soma da Barra 364: 148
--------------------------
Barra 365: 90  47
Soma da Barra 365: 137
--------------------------
Barra 366: 39  27  38  43
Soma da Barra 366: 147
--------------------------
Barra 367: 51  86
Soma da Barra 367: 137
--------------------------
Barra 368: 20  78  52
Soma da Barra 368: 150
--------------------------
Barra 369: 54  81
Soma da Barra 369: 135
--------------------------
Barra 370: 62  20  31  32
Soma da Barra 370: 145
--------------------------
Barra 371: 21  97  32
Soma da Barra 371: 150
--------------------------
Barra 372: 53  81
Soma da Barra 372: 134
--------------------------
Barra 373: 75  75
Soma da Barra 373: 150
--------------------------
Barra 374: 84  48
Soma da Barra 374: 132
--------------------------
Barra 375: 99  51
Soma da Barra 375: 150
--------------------------
Barra 376: 79  35  32
Soma da Barra 376: 146
--------------------------
Barra 377: 62  88
Soma da Barra 377: 150
--------------------------
Barra 378: 21  95  34
Soma da Barra 378: 150
--------------------------
Barra 379: 20  72  58
Soma da Barra 379: 150
--------------------------
Barra 380: 20  95  35
Soma da Barra 380: 150
--------------------------
Barra 381: 79  54
Soma da Barra 381: 133
--------------------------
Barra 382: 95  54
Soma da Barra 382: 149
--------------------------
Barra 383: 94  56
Soma da Barra 383: 150
--------------------------
Barra 384: 61  83
Soma da Barra 384: 144
--------------------------
Barra 385: 87  60
Soma da Barra 385: 147
--------------------------
Barra 386: 98  52
Soma da Barra 386: 150
--------------------------
Barra 387: 42  97
Soma da Barra 387: 139
--------------------------
Barra 388: 62  83
Soma da Barra 388: 145
--------------------------
Barra 389: 85  24  36
Soma da Barra 389: 145
--------------------------
Barra 390: 62  21  65
Soma da Barra 390: 148
--------------------------
Barra 391: 88  22  40
Soma da Barra 391: 150
--------------------------
Barra 392: 95  55
Soma da Barra 392: 150
--------------------------
Barra 393: 72  41  30
Soma da Barra 393: 143
--------------------------
Barra 394: 98  34
Soma da Barra 394: 132
--------------------------
Barra 395: 37  44  68
Soma da Barra 395: 149
```

```
--------------------------
Barra 396: 20  100  30
Soma da Barra 396: 150
--------------------------
Barra 397: 72  49  29
Soma da Barra 397: 150
--------------------------
Barra 398: 100  36
Soma da Barra 398: 136
--------------------------
Barra 399: 87  25  38
Soma da Barra 399: 150
--------------------------
Barra 400: 71  27  46
Soma da Barra 400: 144
--------------------------
Barra 401: 70  26  53
Soma da Barra 401: 149
--------------------------
Barra 402: 87  53
Soma da Barra 402: 140
--------------------------
Barra 403: 41  32  65
Soma da Barra 403: 138
--------------------------
Barra 404: 100  44
Soma da Barra 404: 144
--------------------------
Barra 405: 90  50
Soma da Barra 405: 140
--------------------------
Barra 406: 89  55
Soma da Barra 406: 144
--------------------------
Barra 407: 87  35  27
Soma da Barra 407: 149
--------------------------
Barra 408: 98  29  23
Soma da Barra 408: 150
--------------------------
Barra 409: 76  72
Soma da Barra 409: 148
--------------------------
Barra 410: 92  58
Soma da Barra 410: 150
--------------------------
Barra 411: 62  31  45
Soma da Barra 411: 138
--------------------------
Barra 412: 57  48  34
Soma da Barra 412: 139
--------------------------
Barra 413: 57  93
Soma da Barra 413: 150
--------------------------
Barra 414: 73  44  30
Soma da Barra 414: 147
--------------------------
Barra 415: 31  39  44  28
Soma da Barra 415: 142
--------------------------
Barra 416: 60  22  38  30
Soma da Barra 416: 150
--------------------------
Barra 417: 37  83  30
Soma da Barra 417: 150
--------------------------
Barra 418: 31  88  22
Soma da Barra 418: 141
--------------------------
Barra 419: 57  86
Soma da Barra 419: 143
--------------------------
Barra 420: 70  25  51
Soma da Barra 420: 146
--------------------------
Barra 421: 85
Soma da Barra 421: 85
```