

USO DO ALGORITMO GENÉTICO PARA SOLUÇÃO DO PROBLEMA DE CORTE E EMPACOTAMENTO BIDIMENSIONAL

Wanderlyn Fernandes Duarte

Facci/Funcesi

wanderfd@gmail.com

Fabricio Roulin Bittencout

Facci/Funcesi

fabriciorb@gmail.com

Thaís Cotta Barbosa da Silva

Facci/Funcesi

thais_cotta@yahoo.com.br

Rafael Diego Barbosa

Facci/Funcesi

dba.rafael@gmail.com

Joe Jackson Sangalli Vidor

Facci/Funcesi

joesangalli@gmail.com

Resumo

O planejamento de um problema de corte e empacotamento sem o auxílio de um método inteligente depende tempo e muitas vezes não possui um resultado satisfatório, ocasionando perda de tempo e prejuízos nas indústrias. Neste trabalho, é apresentado o uso de Algoritmos Genéticos (AG) para a solução do problema de corte e empacotamento bidimensional. Esse problema consiste em distribuir um número variado de itens, de diferentes dimensões, em um recipiente de tamanho fixo. Os problemas de corte e empacotamento possuem diversas aplicações práticas nas empresas como: como alocação de programas em disco ou fita magnética, problemas de programação de alocação de veículos, corte de bobinas, corte de vigas, e vários outros. Uma solução otimizada desses problemas pode trazer ganhos significativos nas concorrências. Esse trabalho apresenta uma revisão literária do assunto seguida de uma proposta de solução com AG e das simulações e análises. Neste trabalho foi implementado um AG para solução do problema de corte de chapas de vidro, tratado como um problema de corte bidimensional. Os resultados obtidos mostraram a eficiência dos AG na solução essa classe de problemas e pode-se concluir que é satisfatória a sua utilização.

Palavras-Chaves: Corte 2D; Empacotamento 2D; Algoritmos Genéticos.

Abstract

Planning the solution for a problem of cutting and packing, without any help of an intelligent method, takes time and sometimes it doesn't have a satisfied result, generating time and money losses. In this paper, it's presented using Genetic Algorithm, the solution for the problem of cutting and packing bi dimensional. This problem consists in distribute a variety number of items, of different dimensions, in a

company, like: allocating programs in disk or magnetic tape, problems of programming vehicle allocation, coils cut, beams cut, among others. One optimized solution for these problems can bring significant gains over the competitors. This paper presents one literary review from this subject followed by one proposed solution with GA, simulations and analysis. It was implemented in this paper, one GA for the solution to the problem of cutting glass sheets, treated as a problem of bi dimensional cut. The results show us the efficiency of GA in the solution of this class of problems and we may conclude that your utilization is satisfactory.

Keywords: Bin packing problem; Genetic algorithms.

• INTRODUÇÃO

É grande a diversidade de situações do mundo real onde o problema de corte e empacotamento está presente, tal problema apesar de sua aparente simplicidade possui grande complexidade e exige grande esforço computacional. A definição de um problema de corte consiste basicamente em cortar peças menores a partir de uma peça de dimensões maiores de maneira otimizada, e no caso do problema de empacotamento consiste em alocar de maneira otimizada peças menores em um recipiente de dimensões maiores (Cintra, 1998; Neto, 2005).

Neto (2005) afirma que Kantorovich (1960) foi pioneiro em pesquisas com corte unidimensional realizando suas pesquisas desde 1939, mas apenas em 1960 que publicou seu primeiro trabalho que tinha como objetivo minimizar as perdas (*trim loss*) com o corte unidimensional. Durante as décadas de 50 e 60 outros trabalhos trataram problema similares.

Entretanto, em todos eles foram apresentados métodos adequados apenas para problemas pequenos (Neto, 2005; Dowsland; Dowsland, 1992).

Nos problemas bidimensionais são consideradas duas dimensões na sua solução, sendo o problema caracterizado pelo primeiro símbolo da quádrupla na representação de Dyckhoff (1990). Problemas bidimensionais são comuns em empresas como vidraçarias, onde é necessário o corte de placas de vidro menores em grandes placas, sendo que essas placas maiores são o material de consumo da empresa. Objetiva-se nesse caso o menor consumo e o menor desperdício possível do material (Andrade, 2006). Problemas de corte e empacotamento, em geral, pertence à classe de problemas NP-difícil (Dowsland e Dowsland, 1992; Dyckhoff e Finke; 1992) e por isso necessita de métodos eficientes para sua solução. Sendo alguns deles comentados a seguir.

Poldi e Arenales (2006) afirmam que os pioneiros na pesquisa de problemas de corte foram Gilmore e Gomory (1961), propondo um método de geração de colunas para obtenção de uma solução contínua.

Coffman, Garey e Johnson (1978) propuseram o algoritmo Multifit para uma classe de problemas de sequenciamento de atividades conhecida como P||Cmax, segundo a classificação apresentada por Lawler et al. (1989). Este algoritmo transforma o problema de sequenciamento de tarefas em um problema de empacotamento unidimensional, também conhecido como “*bin packing problem*”. Neste algoritmo, implementado para o sequenciamento de tarefas a serem processadas por um processador, considera-se que cada processador representa uma caixa e as tarefas a serem processadas são os itens a serem armazenados. No Multifit o algoritmo utilizado para empacotamento foi o FFD (*First Fit Decreasing*) que inicialmente pré-ordena os itens a serem alocados em ordem decrescente, e no

processo de alocação, o item é alojado na primeira caixa disponível para alocação, sempre respeitando o limite de alocação suportado pelas caixas.

Graham et al. (1979) apresentaram a heurística conhecida como LPT (*Longest Processing Time*) alocando as tarefas para processamento obedecendo uma ordem decrescente pelos tempos de processamento, assim como em Muller e Limberger (2000) que implementaram o algoritmo 3-Fases.

Em Constantino e Gomes (2002) são apresentados conceitos dos algoritmos genéticos mesclados com uma heurística própria aplicados ao problema de corte industrial bidimensional, seus resultados são importantes e encorajam a utilização dos algoritmos genéticos.

Canto et al. (2010) propuseram um algoritmo genético para resolver o problema de corte bidimensional, que consiste basicamente em determinar um arranjo das peças a serem cortadas, maximizando a utilização de lâminas de vidro, respeitando as restrições impostas pelo fluxo de produção e prazos de entrega.

Hifi et al. (2011) apresentaram um algoritmo paralelo peer-to-peer para resolver o problema de corte bidimensional. Já Candido et al. (2011) propuseram uma estratégia para a geração de padrões de corte bidimensionais do tipo guilhotinado, considerado padrões de corte não-estagiados e padrões de corte em dois estágios, com a possibilidade de rotação dos itens para ambos os casos. A implementação apresentada pelos autores divide-se em duas fases: na primeira, os itens são selecionados e agrupados em subconjuntos através de algoritmos genéticos; e na segunda, empregam-se técnicas heurísticas para realizar o encaixe dos itens e determinar arranjo geométrico do padrão de corte.

Neste trabalho, busca-se resolver o problema de corte e empacotamento bidimensional usando Algoritmos Genéticos, visto que as pesquisas referentes ao problema mostram uma tendência no uso de métodos heurísticos de otimização e os algoritmos genéticos têm se mostrado eficientes nessas aplicações. Foi implementado um AG para solução do problema de corte de chapas de vidro, tratado como um problema de corte bidimensional.

• ALGORITMOS GENÉTICOS

Os Algoritmos Genéticos (AG's) são ferramentas poderosas de busca e otimização baseadas na Teoria da Evolução de Darwin (1859). As variáveis do problema são representadas através de genes em um cromossomo, também denominados indivíduos. Cada ponto no espaço de busca e otimização representa um indivíduo e as coordenadas do ponto representam os genes deste cromossomo (indivíduo). Assim, através de um conjunto de pontos, constituímos uma população que serão as soluções candidatas do problema.

Eles utilizam a seleção natural para escolher, em uma população de soluções candidatas, os melhores indivíduos que gerarão descendentes. Através de um operador chamado *crossover* (cruzamento) o AG combina genes de dois indivíduos, previamente selecionados, para formar dois novos descendentes, os quais possuem grande probabilidade de serem mais aptos que seus respectivos pais, e além disto podem sofrer mutação (Carvalho et al., 2005).

AG's pertencem a um grupo de técnicas computacionais conhecido como Computação Evolutiva ou Evolucionária. Tais técnicas fundamentam-se no desenvolvimento de modelos computacionais que apresentam uma representação genética e simulam a evolução natural dos seres vivos. Isto é possível através de uma representação matemática e algorítmica de alguns conceitos apresentados por Darwin

(1859) e de genética.

Os AG's são, dentre os ramos da Computação Evolucionária, os mais conhecidos, e tiveram origem no trabalho de John Holland ainda na década de 60 e, futuramente com a publicação de seu livro *Adaptation in Natural and Artificial Systems* em 1975 (Holland, 1992). Neste livro, o autor introduz o assunto no âmbito da genética, economia, teoria de jogos, reconhecimento de padrões e inferência estatística, controle e otimização de funções. Desde então, os AG's apresentavam um conceito mais amplo do que simples otimização, eram apresentados como um modelo de aprendizagem de máquina (Heitkötter e Beasley, 2005).

Futuramente, com a publicação do livro *Genetic Algorithms in Search, Optimization and Machine Learning* de David Goldberg (Goldberg, 1989) a idéia de otimização ocupou lugar central nas pesquisas e teoria dos AG's. Desde as primeiras pesquisas de Holland (1992), os AG's vêm se aprimorando, e cada vez mais, sendo aplicadas a problemas de busca e otimização em diversas áreas. Apresentando grande aplicabilidade em áreas como engenharia, robótica, finanças empresariais, logística, gerenciamento de redes, otimização multicritério, ciências biológicas, telecomunicações, etc (Miranda, 2005).

O funcionamento básico de um AG é apresentado na Figura 1. Um processo iterativo onde são analisadas, em cada etapa, um grupo de possíveis soluções do problema em questão (Koza, 1992).

Inicialmente uma população aleatória é gerada, em seguida os indivíduos serão classificados através de uma função de avaliação e logo após, selecionados os melhores. Os que não foram selecionados são descartados. Dentre os selecionados, eles irão reproduzir (crossover) gerando descendentes. Todos os indivíduos da nova população poderão sofrer mutação, alterando assim suas características e diversificando a população, exceto os que foram selecionados por elitismo, estes são adicionados sem alteração (Mitchell, 1999).

Os indivíduos que compõem a nova população serão avaliados e o processo se repete até que uma solução ótima, ou próxima de ótima, seja encontrada dentre os membros da população.

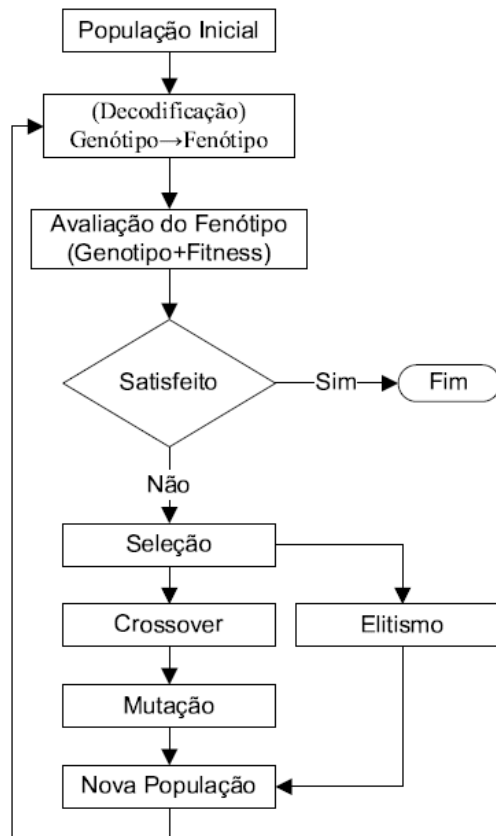


Figura 1 - Funcionamento de um Algoritmo Genético

• DESCRIÇÃO E MODELAGEM DO PROBLEMA

Suponha que uma empresa possua uma quantidade de chapas de vidro, de dimensão $C \times L$, sendo $C = 8$ metros e $L = 4$ metros, e esta mesma empresa possua um grande número de peças (n) que devem ser cortadas nessas chapas de vidro. Essas n peças possuem dimensões $c_i \times l_i$ e todas elas devem ser produzidos usando o menor número de chapas de vidro possível.

O problema será maximizar o corte das peças (n) produzidas pela empresa, ou seja, reduzir o número de chapas de vidro necessárias para o corte de todas as peças.

Este tipo de problema pode ser tratado como um problema de corte de estoque bidimensional (*cutting stock problem*), quando se considera a chapa de vidro como um recipiente fixo, no qual a soma das áreas das peças cortadas nessa chapa não deve ultrapassar a área da chapa de vidro (material de consumo).

Para solucionar este problema de corte bidimensional, serão considerados os seguintes pontos:

1. As chapas de vidro possuem as dimensões de 8 metros x 4 metros;
2. Cada solução de corte representa a utilização de uma chapa;
3. Os recipientes serão as chapas de vidro;
4. Os itens serão as peças a serem cortadas;
5. A perda de material no corte da chapa é desprezível (espaço entre peças).

Modelamento do problema:

n : número de tipos de item;
 $a_i = c_i \times l_i$: dimensão de cada item tipo i ($i > 0$), $i = 1, \dots, n$;
 $A = C \times L$: restrição do problema (dimensão da chapa de vidro);
 $c_i \in (0, C)$;
 $l_i \in (0, L)$.

Variáveis de decisão:

a_i : item i que faz parte da solução, $i = 1, \dots, n$.

$$\max z = \sum_{i=1}^n a_i$$

s.a

$$\sum_{i=1}^n a_i \leq A$$

$$a_i \geq 0$$

Com base neste modelo foi desenvolvido um algoritmo para apresentar as melhores soluções deste corte de peças em chapas de vidro.

• ABORDAGEM GENÉTICA PARA SOLUÇÃO DO PROBLEMA

A estratégia de modelagem do problema para uso dos Algoritmos Genéticos foi representar cada gene dos indivíduos por inteiros (0, 1 ou 2) que indicam o corte ou não corte das peças na chapa e orientação (0 ou 90 graus). Em seguida, busca-se cortar uma chapa de cada vez, até encontrar a solução, ou seja, a cada chapa cortada (parte da solução é encontrada), as peças já cortadas são retiradas do universo e conseqüentemente o tamanho do cromossomo dos indivíduos diminui e, novamente tenta-se cortar uma outra chapa de vidro. Desta forma, de chapa em chapa cortada, o algoritmo genético vai refinando e reduzindo seu espaço de busca e encontrando pequenas novas soluções que, ao final, unidas, formam a solução final para o problema. Esta estratégia se mostrou eficiente, veja Tabela 1.

Tabela 1- Simulações com diferentes quantidades de peças para corte

Peças	Tempo (seg.)	Área Total para Corte	Chapas Cortadas	Aproveitamento
20	0,40	114	4	100%
50	2	292	10	98,61%
100	19	622	20	99,34%
200	55	1264	36	98,48%
400	767	2563	83	96,79%

Aleatoriamente 20 (vinte) indivíduos são gerados para compor a população inicial, que será o ponto de partida para o processo de busca da solução. Cada indivíduo é formado por um vetor de inteiros 0, 1 ou 2, onde o inteiro 1 representa o corte da peça na sua posição original ($c_i \times l_i$) na chapa de vidro (material de consumo). O inteiro 2 também representa o corte da peça na chapa, porém com orientação

alterada em 90 graus, ou seja, na posição $l_i \times c_i$. O último valor possível para um gene do indivíduo seria o inteiro 0, que representa o não corte da peça. Desta forma, cada indivíduo possui uma sequência de inteiros diretamente relacionados com um vetor principal que possui todas as peças com suas respectivas dimensões. Em todas as gerações seguintes, trabalhou-se com 20 (vinte) indivíduos na população.

Cada indivíduo da população é avaliado por uma função objetivo que visa maximizar o corte de peças nas chapas de dimensões 8 metros x 4 metros, minimizando o desperdício do material de consumo. Ao final da avaliação dos indivíduos, todos terão uma nota (*fitness*) que os classificarão em um ranking de aptidões. Os indivíduos mais aptos ou qualificados são os que apresentarem menor desperdício da chapa de vidro. Os indivíduos menos aptos são os que apresentarem maior desperdício da chapa de vidro, juntamente com os que apresentam como solução uma área de corte com dimensões superiores a 8 metros por 4 metros. Dessa forma, não são cortadas peças onde a soma das suas áreas seja superior a 32 metros quadrados, ou 8 metros x 4 metros.

Após a avaliação, os mais aptos são selecionados para reproduzirem, gerando descendentes que irão compor a população de indivíduos da nova geração. Neste trabalho é usado o elitismo, que consiste na transferência dos dois melhores indivíduos de uma geração para a outra sem modificação. Isto é feito para que não se percam as melhores soluções encontradas para o problema até então.

Os indivíduos selecionados na etapa anterior iniciam a fase de reprodução (*crossover*) e mutação. A reprodução usada neste trabalho foi o de *crossover* um ponto, sendo este ponto definido aleatoriamente, trabalhando com uma taxa de cruzamento de 100%. A taxa de mutação considerada foi de 2%, aplicada a cada um dos genes dos indivíduos. Na abordagem genética utilizada nesse trabalho a mutação pode ocasionar três mudanças na solução candidata: mudando o inteiro 0 para 1 ou 2, coloca-se uma peça na lista de corte que antes não estava sendo cortada, a outra mudança seria mudando o inteiro 1 ou 2 para 0, retirando-se assim aquela respectiva peça da lista de corte e por último mudando o inteiro 1 para 2, ou vice-versa, a peça teria sua orientação no corte modificada em 90 graus, proporcionando uma possibilidade diferente de arranjo no corte.

A partir de então, uma nova população é gerada e após ser avaliada e apresentar uma solução satisfatória, esta será usada como solução de corte de determinadas peças na chapa de vidro, caso contrário, entrará em um *loop* até que se encontre uma solução.

Na avaliação de cada indivíduo, a abordagem genética utiliza uma matriz de booleanos para simular o corte das peças indicadas naquele cromossomo. Isso é necessário, pois as áreas somadas das peças pode não exceder a área da chapa de vidro, porém devido às suas dimensões $c_i \times l_i$ serem variadas, a combinação dessas peças pode exceder as dimensões $C \times L$ da chapa de vidro, neste caso 8 metros x 4 metros. Desse modo o algoritmo proposto nesse trabalho respeita uma regra para o posicionamento das peças que serão cortadas. A regra determinada, posiciona a primeira peça no canto superior esquerdo da chapa de vidro, e as outras peças são posicionadas sempre no primeiro ponto superior esquerdo disponível. Para o controle da área já utilizada da chapa, o algoritmo utiliza uma matriz de booleanos, indicando *true* para a área utilizada. Quando o algoritmo tenta posicionar uma peça na chapa e essa não tem espaço para esse posicionamento, o *fitness* daquele indivíduo se torna negativo e recebe a soma das áreas de todas as peças que não puderam ser posicionadas.

Os indivíduos com *fitness* negativo são considerados pelo algoritmo as piores soluções, e não são aceitos como resultado válido porque representam a existência de uma quantidade superior de peças à suportada pela chapa de vidro.

Os critérios de parada definidos para este algoritmo foram:

1. Parar caso encontre as peças que preencham completamente uma chapa de vidro de 8 metros x 4 metros;
2. Parar caso a melhor solução de corte não varie por um período de iterações sucessivas.

• SIMULAÇÕES E RESULTADOS

Para validar a abordagem genética usada na solução do problema em questão, foram geradas aleatoriamente peças com dimensões dentro do intervalo de 1 a 4 metros para as variáveis c_i e l_i . Estas peças devem ser cortadas em chapas de vidro de 8 metros x 4 metros. O computador usado nas simulações desta pesquisa foi um PC com Processador *Celeron 1.5 Ghz*, com 1 GB de *RAM*. O algoritmo de simulação foi implementado em Java.

Para que fosse possível apresentar todas as peças graficamente, trabalhou-se inicialmente com um universo de 20 (vinte) peças para serem cortadas em chapas de vidro de 8 metros x 4 metros. Em seguida o algoritmo genético posicionou estas peças com um aproveitamento de 100%. O cálculo do aproveitamento é feito com base na área de sobra, caso ocorra, até a penúltima chapa cortada, visto que na última chapa serão posicionadas as peças que restarem, não existindo outras para serem cortadas e otimizadas.

Nas Figuras 2, 3, 4 e 5, são apresentadas todas as peças cortadas em um total de 4 (quatro) chapas de vidro. Como pode ser observado, foram posicionadas 3 (três) chapas com aproveitamento de 100%, apenas na última chapa o algoritmo posicionou as 04 (quatro) peças restantes. Tratando-se de um problema que visa minimizar o número de chapas conseguindo maximizar o corte de peças por chapa, esta abordagem apresentou resultados significativos.

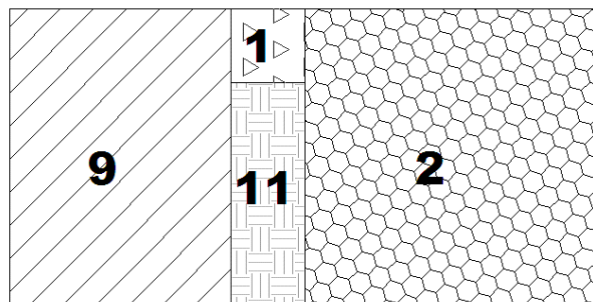


Figura 2- Peças cortadas na primeira chapa de vidro

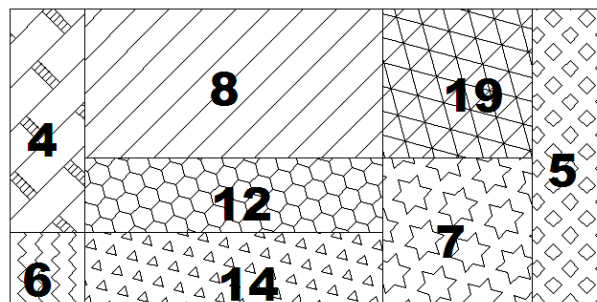


Figura 3- Peças cortadas na segunda chapa de vidro

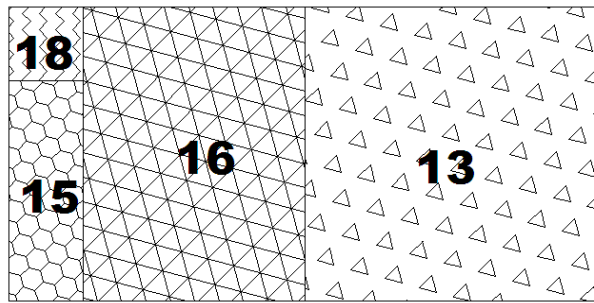


Figura 4- Peças cortadas na terceira chapa de vidro

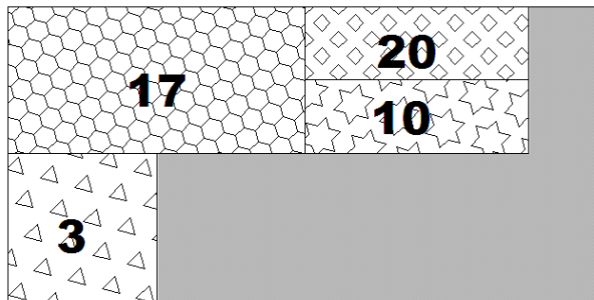


Figura 5- Peças cortadas na quarta chapa de vidro

Foram realizadas outras simulações para teste da eficiência do algoritmo proposto e como pode ser visto pela Tabela 1, ao aumentar a quantidade de peças para serem cortadas em chapas, logicamente aumenta-se a área total de corte. O tempo gasto pelo algoritmo para se obter a resposta, varia de menos de 1 segundo para cortar 20 peças até 767 segundos (12,7 minutos), maior tempo gasto, que foi necessário para cortar 400 peças em 83 chapas de vidro de 8 metros x 4 metros, com um aproveitamento de 96,79%.

Na Figura 6, observa-se que à medida que a quantidade de peças aumenta logicamente aumenta-se a área total de corte, mas o algoritmo usado conseguiu posicionar as peças de forma otimizada e em uma proporção de crescimento de chapas cortadas muito mais baixa que a do aumento de peças.

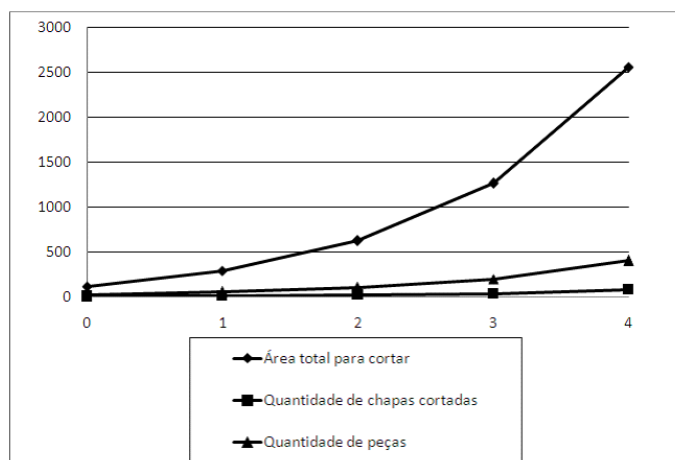


Figura 6- Representação gráfica das simulações de corte de peças

• CONCLUSÕES

A solução proposta através do uso de Algoritmos Genéticos se mostrou eficiente e perfeitamente viável para a solução do problema de corte de chapa de vidro, que foi tratado como um problema de corte de estoque bidimensional (*cutting stock problem*).

Esta proposta de algoritmo permitiu encontrar a solução dos problemas tanto em situações simples, com poucos itens, quanto extremas, com maior número de itens. Porém, como todo algoritmo, ele é um suporte às tomadas de decisão. Facilitando e agilizando os cálculos necessários para solução dos problemas relacionados à alocação de tarefas e corte de chapas de vidro. O que não isenta a necessidade da decisão gerencial de validar e verificar se as soluções apresentadas são realmente viáveis.

Muitos aperfeiçoamentos podem ser feitos com o intuito de aprimorar o algoritmo implementado, como: o aprimoramento da abordagem bidimensional para soluções com itens de formatos complexos (formas diferentes das retangulares), a opção na abordagem bidimensional de manter um espaço entre os itens, permitindo a solução de problemas onde há perda de material durante o corte e a modelagem de uma abordagem tridimensional, utilizada por exemplo em carregamento de veículos e contêiner.

Referências

ANDRADE, C. E. *Um Algoritmo Exato para o Problema de Empacotamento Bidimensional em Faixas*. Dissertação (Instituto de Computação) — Universidade Estadual de Campinas, Campinas, 2006.

CANDIDO, L., GUIMARÃES, T. ; SOUZA, L. *Geração de Padrões de Corte em Guillotina Bidimensional utilizando Algoritmos Genéticos e Heurísticas de Encaixe*. Anais do XIV Simpósio de Pesquisa Operacional e Logística da Marinha, Rio de Janeiro, 2011.

CANTO, N.C.F.; COSTA, F.; SASSI, R.J.; , *A genetic algorithm for solving the two-dimensional cutting in glass sheets problem*, Information Systems and Technologies (CISTI), 2010 5th Iberian Conference on , vol., no., pp.1-4, 16-19, 2010

CARVALHO, A. C. P. L. F.; BRAGA, A. P.; LUDEMIR, T. B. *Sistemas Inteligentes Fundamentos e Aplicações*. Barueri: Manole, 2005. 525 p.

CINTRA, G. F. *Algoritmo Híbridos para Problemas de Corte Unidimensional*. 103 p. Dissertação (Mestrado Matemática Aplicada) — USP - Universidade de São Paulo, São Paulo, 1998.

COFFMAN, J. E. G.; GAREY, M. R.; JOHNSON, D. S. An application of bin-packing to multiprocessor scheduling. *SIAM Journal Computing*, n. 7, p. 1–17, 1978.

CONSTANTINO, A. A.; GOMES, A. M. J. *Um algoritmo genético híbrido para o problema de corte industrial bidimensional*. *Acta Scientiarum*, n. 6, p. 1727–1731, 2002.

DARWIN, C. *Origin of Species*. [S.l.: s.n.], 1859.

DOWSLAND, K.; DOWSLAND, W. *Packing problems*. *European Journal of Operational Research*, n. 56, p. 2–14, 1992.

DYCKHOFF, H. *A typology of cutting and packing problems*. *European Journal of Operation Reserch*, n. 444, p. 145–159, 1990.

DYCKHOFF, H.; FINKE, U. *Cutting and packing in production and distribution: Typology and bibliography*. Heidelberg: Springer-Verlag Co., 1992.

- GILMORE, P. C.; GOMORY, R. E.** *A linear programming approach to the cutting stock problem.* Operations Research, n. 9, p. 849–859, 1961.
- GOLDBERG, D. E.** *Genetic Algorithms in Search, Optimization, and Machine Learning.* Reading: Addison-Wesley Publishing Company, Inc., 1989.
- GRAHAM, R. et al.** *Optimization and approximation in determinist sequencing and scheduling: a survey.* Annals of Discrete Mathematics, n. 5, p. 287–326, 1979.
- HEITKÖTTER, J. & BEASLEY, D.** *The Hitch-Hiker's Guide to Evolutionary Computation*, <ftp://ftp.cs.bham.ac.uk/pub/Mirrors/ftp.de.uu.net/EC/FAQ/www/index.html>, 2000, Ultimo acesso: Outubro 2005
- HIFI, M.; SAADI, T.; HADDADOU, N.;** , *High Performance Peer-to-Peer Distributed Computing with Application to Constrained Two-Dimensional Guillotine Cutting Problem, Parallel, Distributed and Network-Based Processing (PDP)*, 2011 19th Euromicro International Conference on , vol., no., pp.552-559, 9-11 Feb. 2011 doi: 10.1109/PDP.2011.56
- HOLLAND, J. H.** *Adaptation in Natural and Artificial Systems.* [S.l.]: MIT Press, 1975.
- KANTOROVICH, L. V.** *Mathematical methods of organizing and planning production.* Management Science, n. 6, p. 366–422, 1960.
- KOZA, J.R.** *Genetic Programming: On the Programming of Computers by means of Natural Selection*, MIT Press, 1992.
- LAWLER, E. et al.** *Sequencing and scheduling: algorithms and complexity.* Report BS-R8909 — Center of Mathematics and Computer Science, Amsterdam, 1989.
- MIRANDA, M. N.** *Algoritmos genéticos: Fundamentos e aplicações.* Rio de Janeiro, 2005. Disponível em: <<http://www.gta.ufrj.br/marcio/genetic.html>>. Acesso em: agosto 2007.
- MITCHELL, M.** *An Introduction to Genetic Algorithms.* MIT Press, 1999.
- MULLER, F. M.; LIMBERGER, S. J.** *Uma heurística de trocas para o problema de sequenciamento de tarefas em processadores uniformes.* Pesquisa Operacional, n. 1, p. 31–42, 2000.
- NETO, L. S.** *Modelo não-linear para minimizar o número de objetos e o setup num problema de corte unidimensional.* Tese — UNICAMP Universidade Estadual de Campinas, Campinas, 2005.
- POLDI, K. C.; ARENALES, M. N.** *Heurística para o problema de corte de estoque unidimensional inteiro.* Pesquisa Operacioanl, n. 26, p. 473–492, 2006.