

Comparação de algoritmos genéticos e programação quadrática seqüencial para otimização de problemas em engenharia

Genetic algorithms and sequential quadratic programming comparisons for engineering problems optimization

Mateus Lembi Teles¹, Herbert Martins Gomes²

Universidade Federal do Rio Grande do Sul, UFRGS, Porto Alegre, RS,

¹e-mail: mateus@victum.com.br ²e-mail: herbert@mecanica.ufrgs.br

RESUMO: Um dos objetivos do presente trabalho é a comparação do Algoritmo Genético, utilizado como método de otimização paramétrica de estruturas de engenharia, com a solução da literatura e com o método de Programação Quadrática Seqüencial (*Sequential Quadratic Programming - SQP*) para determinados casos utilizados como *benchmarks*, como, por exemplo, treliças planas e espaciais. Também se estuda o comportamento e desempenho do referido algoritmo para variações em seus parâmetros de entrada nestes mesmos exemplos analisados. Conforme os resultados apresentados, em termos de precisão, os métodos tiveram desempenho equivalente. Com o que é apresentado, pode-se concluir que a utilização do Algoritmo Genético como método de otimização paramétrica de estruturas de engenharia é viável devido à eficiência e robustez do mesmo.

ABSTRACT: One of the aims of this study is to compare the genetic algorithm used as a method of parametric optimization of engineering structures, with the solution of literature and the method of Sequential Quadratic Programming (Sequential Quadratic Programming: SQP) for specific cases as benchmarks, such as planar and spatial trusses. It is also studied the behavior and performance of this algorithm to variations in their input parameters in these same samples analyzed. According to results presented in terms of accuracy, the methods have equivalent performance. By the presented results, it can be concluded that the use of genetic algorithm as a method of parametric optimization of engineering structures is feasible due to its efficiency and robustness.

1. INTRODUÇÃO

Um aspecto importante dos processos construtivos de estruturas na engenharia como, por exemplo, molas e treliças, é a minimização dos custos de produção, onde estão envolvidos os custos de material, de fabricação e de montagem, entre outros. Neste sentido, os métodos de otimização são considerados de fundamental importância. Portanto, este trabalho mostra-se relevante, pois tem como foco o uso de um método de otimização aplicado a estruturas de engenharia. Além disso, o método utilizado, o Algoritmo Genético (AG), é considerado pouco tradicional e suas adaptações são bastante robustas e fornece uma resposta razoável para vários problemas de otimização, o que causou interesse quanto à utilização do mesmo. Sendo assim, um dos objetivos deste trabalho é a comparação do Algoritmo Genético com a solução da literatura e com outro método de otimização, o método de Programação Quadrática Seqüencial (*Sequential*

Quadratic Programming: SQP), para determinados casos utilizados como *benchmarks* (casos de referência). Outro objetivo é estudar o comportamento e desempenho do referido algoritmo para variações em seus parâmetros de entrada/utilização nestes mesmos exemplos analisados.

2. REVISÃO BIBLIOGRÁFICA

Segundo Braga (Souza Jr. [16]), ferramentas computacionais de análise de estruturas utilizando técnicas numéricas (elementos finitos, elementos de contorno, etc) contribuíram muito para o avanço dos métodos de otimização. Devido ao grande número de incógnitas com que estas ferramentas trabalham, poderia-se contentar apenas com a análise através de processos de "tentativa e erro". Para diminuir o esforço computacional nos procedimentos de programas computacionais de elementos finitos, os códigos comerciais foram equipados com

otimizadores, conforme se verifica hoje em dia nos programas comerciais de grande desempenho como o Ansys®, Nastran®, entre outros. Como se percebe, a otimização é uma das ferramentas de aplicação imediata mais importante da atualidade, empregada na resolução dos mais variados problemas nas diversas áreas da engenharia. Como alguns exemplos recentes, o método dos algoritmos genéticos foi utilizado por Souza Jr. [16] na solução de problemas de dimensionamento de estruturas tubulares metálicas espaciais com barras cruzadas para coberturas. Malaquias [8] aplicou os AGs em alguns problemas de otimização de rotas de distribuição de medicamentos. Romão [14] empregaram os AGs no controle de um processo térmico. O método dos AGs foi aplicado por Konzen [5] para otimização de métodos de controle de qualidade de fármacos. De modo geral, todos os métodos de otimização experimentam, em maior ou menor intensidade, dificuldades relacionadas à não convergência e à existência de múltiplos pontos ótimos locais da função objetivo. Independente do método utilizado, como vantagens no uso de procedimentos de otimização podem ser citados os seguintes aspectos:

- Diminuição do tempo dedicado ao projeto;
- Possibilidade de tratamento simultâneo de uma grande quantidade de variáveis e de restrições de difícil visualização gráfica ou tabular;
- Possibilidade de obtenção de algo melhor (geralmente um projeto mais barato), mesmo que não seja um ótimo global;
- Redução de custo em relação aos modelos experimentais.

Como desvantagens no uso da otimização podem ser consideradas algumas dificuldades para a sua aplicação, tais como:

- Aumento do tempo computacional;
- Quando as funções envolvidas são descontínuas (ou de gradientes descontínuos), pode haver convergência lenta;
- Presença de muitos pontos ótimos locais (ponto ótimo global raramente é obtido);

Para superar alguns dos problemas no uso da otimização foram desenvolvidos diversos métodos de busca pela solução ótima. A escolha de um ou outro método de busca deve estar estritamente relacionada ao problema a ser otimizado. Dessa maneira, o uso de AGs deve ser limitado apenas àquelas situações em que não há um algoritmo exato capaz de resolver o problema em um tempo razoável, dentro de uma memória finita. Por outro lado, apesar de a execução de um algoritmo genético ser relativamente lenta, ela produz a solução para problemas difíceis com múltiplos

ótimos e funções não suaves ou contínuas, o que é compensador, pois nesses casos os “algoritmos tradicionais” baseados em gradientes muitas vezes falham.

Os algoritmos genéticos (AGs) são uma classe de algoritmos evolutivos que tiveram sua origem no início da década de 70 através dos trabalhos desenvolvidos na Universidade de Michigan pelo professor John Holland [4] e seus alunos. A partir dos trabalhos publicados por Holland, principalmente o livro “*Adaptation in Natural and Artificial Systems*” de 1975, a evolução pôde ser encarada como um processo de otimização e os algoritmos genéticos foram utilizados com sucesso em diversos problemas de otimização (Coello *et al* [1], Fonseca *et al* [3], Miasaki [10], Sandrini [15], Souza jr.[16]). O algoritmo genético (AG) constitui um método de busca aleatória direcionada capaz de obter a solução ótima global num espaço de busca complexo (com ou sem restrições). O AG é considerado um método robusto, utilizado basicamente para resolver problemas em pesquisas numéricas, otimização de funções e aprendizagem de máquina, dentre outras áreas. Basicamente, os AGs operam num conjunto de soluções aplicando o princípio da sobrevivência dos indivíduos mais aptos para produzir uma solução cada vez melhor, diferentemente de outros métodos que partem de um ponto do espaço de busca para encontrar a solução ótima. Dessa maneira, geração após geração, um novo conjunto de indivíduos é criado pelo processo de seleção dos indivíduos progenitores, de acordo com o nível de aptidão associado ao problema. A procriação é feita entre os progenitores através de operadores extraídos da genética natural. Este processo leva à evolução de indivíduos que se adaptam melhor ao problema, de modo semelhante à adaptação natural. Os AGs, de acordo com Goldberg [17], modelam os processos genéticos naturais básicos, tais como: seleção, cruzamento e mutação, através de operadores matemáticos.

2.1 Algoritmos Genéticos

Os algoritmos genéticos básicos funcionam da seguinte forma:

1. Geração da população inicial que consiste de um conjunto de possíveis soluções para o problema a ser resolvido. Essa população é geralmente gerada de forma aleatória;
2. A população é avaliada (de acordo com uma função chamada função de custo) e cada cromossomo recebe um valor que reflete sua qualidade para resolução do problema;

3. Depois de avaliados, os indivíduos passam por um processo de seleção onde os indivíduos mais aptos são selecionados e os menos aptos são descartados;
4. São aplicados os operadores genéticos nos cromossomos selecionados. Os operadores genéticos mais conhecidos são o de cruzamento e o de mutação;
5. Uma nova geração de soluções é obtida contendo os descendentes gerados pelas modificações realizadas na etapa 4;
6. As etapas de 2 a 5 são repetidas até que seja encontrada uma solução satisfatória. As etapas mencionadas podem ser melhor entendidas através de um algoritmo, mostrado na Figura 1 segundo Rodrigues [13]:

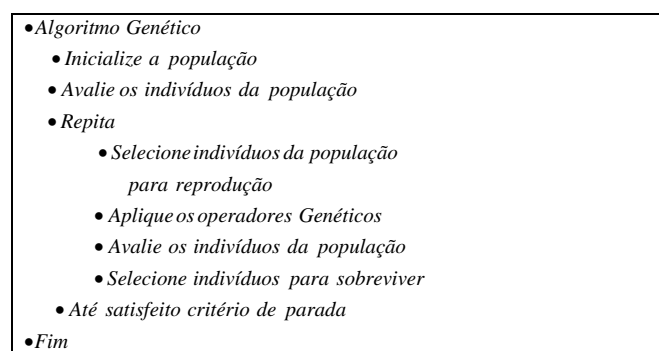


Fig. 1 - O Pseudocódigo Algoritmo Genético.

População inicial - Gerada de forma aleatória, a população inicial é o ponto de partida para a evolução do algoritmo genético. Cada um dos indivíduos da população representa uma possível solução para o problema. Uma vez gerada, a população é avaliada pela função de custo. A população de um AG é formada por indivíduos. A representação desses indivíduos define como a estrutura será manipulada. Essa representação depende do tipo de problema a ser resolvido. Os principais tipos de representação são a binária, real e simbólica. A representação binária é utilizada para problemas inteiros e numéricos, sendo que os problemas numéricos também podem fazer uso de representação real. A desvantagem da utilização da representação binária está na limitação da resolução que esta abordagem induz nos valores assumidos pelas variáveis de projeto. Já a representação de permutação de símbolos é recomendada para ser utilizada em problemas baseados em ordem. Neste trabalho foi utilizada a codificação real para a representação dos indivíduos.

Avaliação - A Função de Custo, que é uma função matemática responsável por realizar o processo de avaliação dos indivíduos, faz o papel do meio ambiente e o valor retornado (medida de sua aptidão)

representa a adaptabilidade do indivíduo ao seu habitat e a sua capacidade de sobrevivência. Em conjunto com a Função de Custo, pode-se criar uma Função de Penalização para que os indivíduos que violarem alguma das restrições do problema venham a ter uma avaliação baixa a fim de que não possam ser escolhidos como solução ótima. Neste trabalho adotou-se uma função de penalização que pune os indivíduos de maneira proporcional à sua violação da restrição do problema.

Seleção - A idéia básica do processo de seleção em um algoritmo genético é permitir que os indivíduos mais adaptados (melhor aptidão) tenham maiores chances de reprodução. Ou seja, o processo de seleção escolhe os melhores pais para o processo de reprodução, garantindo com isto a evolução dos indivíduos (soluções).

A escolha dos indivíduos reprodutores pode ser feita de inúmeras formas. Dentre os métodos mais importantes encontrados na literatura, destacam-se: Seleção por Método da Roleta e Seleção por Torneio. A seleção pelo método da roleta consiste em colocar todos os indivíduos em uma roleta, onde a posição de cada indivíduo é proporcional à sua aptidão. A roleta é rodada N vezes, sendo que N é o número de indivíduos que serão necessários para se realizar os operadores genéticos. Quanto maior a aptidão do indivíduo, mais chances de ele ser escolhido. Já a seleção por torneio é uma das formas mais utilizadas devido à sua eficiência e simplicidade na implementação em problemas de otimização. Essa seleção preserva melhor a diversidade da população porque a escolha é realizada apenas comparando o valor da função de custo entre os indivíduos que participam do torneio. Neste método, existe um parâmetro denominado tamanho do torneio (K) que define quantos indivíduos são selecionados aleatoriamente dentro da população para competir. Uma vez definidos os competidores, aquele dentre eles que possui a melhor avaliação é selecionado para a aplicação do operador. Quanto maior o tamanho do torneio, maior também a perda da diversidade: em um torneio de tamanho cinco, cerca de metade da população é perdida a cada geração. Por isso, o valor usual é $K = 2$ de acordo com Linden [7]. Neste trabalho adotou-se a Seleção por Torneio de tamanho dois por ser considerado um dos métodos de seleção mais eficientes Linden [7].

Reprodução - Segundo a teoria Darwinista, o principal mecanismo responsável pela evolução dos seres vivos é a reprodução, pois os seres vivos vão modificando-se e adaptando-se ao meio à medida que vão se reproduzindo. Nos algoritmos

genéticos a reprodução se dá pela aplicação dos operadores genéticos, num processo que pode ser visto da seguinte forma:

Geracional - Sua principal característica é a substituição total da população a cada geração, tendo como conseqüência a possibilidade de perda de material genético de boa qualidade. Uma tentativa de solucionar este problema é adotar um processo de seleção elitista, onde os melhores indivíduos de uma geração passam para a próxima sem serem alterados. Neste trabalho garantiu-se a presença de dois indivíduos melhores adaptados nas gerações futuras.

2.2 Operadores genéticos

O princípio básico dos operadores genéticos é transformar a população através de sucessivas gerações. Os operadores genéticos são necessários para que a população se diversifique e mantenha as características de adaptação adquiridas nas gerações anteriores, de acordo com Goldberg [17]. Os algoritmos genéticos sem os operadores genéticos perderiam o sentido, pois eles são os responsáveis por manter a diversidade de indivíduos em uma população (propagação de cromossomos com potenciais de gerar indivíduos aptos e possibilidade de aleatoriamente mudar as características genéticas fugindo de regiões de mínimos locais). Os principais operadores genéticos são o de Cruzamento e o de Mutação. A seguir, esses dois tipos de operadores são melhor descritos.

Cruzamento - O cruzamento é o operador responsável pela propagação das características dos indivíduos mais aptos da população, por meio da troca de material genético entre os pais, permitindo que os filhos herdem essas características. Este operador deve ser aplicado com maior frequência devido à sua grande eficiência. Existem vários tipos de cruzamentos, dentre eles: cruzamento simples e heurístico.

- Cruzamento simples - O mais simples dos operadores de cruzamento consiste em selecionar um ponto de forma aleatória nos dois pais e então, as informações anteriores ao ponto de cruzamento num pai são combinadas com as informações posteriores ao ponto de cruzamento no outro pai, resultando assim, em um filho, segundo Rodrigues [13]. De forma similar, se obtém o segundo filho, conforme a Figura 2. Um número gerado aleatoriamente, comparado com uma taxa de

cruzamento, determina se duas configurações selecionadas serão submetidas ao cruzamento.

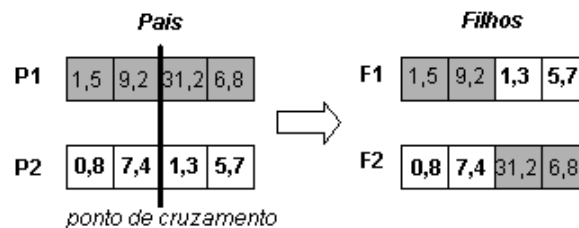


Fig. 2- Cruzamento simples em representação real.

- Cruzamento heurístico - Consiste em estabelecer um intervalo fechado para cada par de valores de cromossomos, de uma distância $\mu \in [0,2]$ abaixo do menor valor armazenado na posição até uma distância μ acima do maior valor armazenado na posição e escolher um valor aleatório pertencente a este intervalo, segundo Linden [7]. O valor de μ adotado neste trabalho foi igual a dois. Pode-se ver um exemplo do funcionamento deste cruzamento na Figura 3.



Fig. 3 - Cruzamento heurístico em representação real para $\mu = 0,5$.

Para este trabalho adotou-se o cruzamento heurístico, pois ele permite que o gene do cromossomo assuma um valor dentro de uma faixa mais ampla em relação ao cruzamento simples, aumentando, assim, o espaço de busca da solução ótima.

Mutação - O operador genético de mutação é aplicado com uma pequena probabilidade após a realização do cruzamento. É responsável pela introdução e manutenção da diversidade do material genético da população alterando aleatoriamente o valor de um gene de um cromossomo, como é mostrado na Figura 4. Os limites para o sorteio devem ser conhecidos *a priori* e representam o espaço de busca para aquela posição.



Fig. 4 - Mutação aleatória em representação real.

A diversidade introduzida pela mutação garante a probabilidade de se chegar a qualquer ponto do

espaço de busca, além de contornar o problema dos máximos locais alterando levemente o espaço de busca. Uma taxa de mutação indica a probabilidade do indivíduo sofrer mudanças em suas características hereditárias.

2.3 Parâmetros de Controle

A qualidade de um algoritmo genético está relacionada com a maneira de se escolher a população inicial, com o critério de parada e com a definição dos parâmetros de controle. A seguir, alguns desses tipos de parâmetros:

- Tamanho da população: diz quantos indivíduos farão parte da população a cada geração.
- Taxa de Cruzamento: probabilidade dos indivíduos sofrerem a ação desse operador.
- Taxa de Mutação: é a probabilidade do conteúdo de um indivíduo ser modificado.
- Tipo de Cruzamento: Parâmetro qualitativo. Nele se define qual tipo de cruzamento é utilizado (simples e heurístico).
- Tipo de Seleção: também é um parâmetro qualitativo em que se tem o tipo de seleção utilizado no desenvolvimento do AG.

2.4 Critério de Parada

Como critério de parada para os algoritmos genéticos utiliza-se, normalmente, o número máximo de gerações, o tempo limite de processamento ou o critério da estagnação, ou seja, quando não se observa melhoria da população depois de várias gerações consecutivas. No algoritmo implementado foram utilizados os critérios da estagnação (50) e do número máximo de gerações (100) cujos valores foram os valores padrão para o *toolbox* do Matlab.

3. PROGRAMAÇÃO QUADRÁTICA SEQUENCIAL

Problemas de otimização, como citados no item 2, podem ser resolvidos através de métodos de programação não-linear baseados em gradientes. Um dos métodos mais promissores é o de Programação Quadrática Sequencial (*Sequential Quadratic Programming- SQP*) o qual, basicamente, consiste na aproximação sequencial do problema de Programação Não-Linear como um problema de Programação Quadrática. Como todo método baseado em gradientes, no SQP existe

a necessidade de calcular, durante o processo de busca da solução ótima, as derivadas tanto da função objetivo (massa de uma estrutura, por exemplo) como dos vínculos ou restrições (restrição de tensões, por exemplo). Estas derivadas podem ser estimadas numericamente, mas a convergência pode não ser boa. A melhor opção é a determinação analítica dessas derivadas.

$$\begin{aligned} & \text{Minimizar } f(x) \\ & \text{Sujeito a } h(x) = 0 \\ & \quad \quad \quad g(x) \leq 0 \end{aligned} \quad (1)$$

Ao contrário de outros métodos que tentam converter o problema em uma seqüência de subproblemas de otimização sem restrições, o SQP tenta resolver o problema de otimização iterativamente, onde a solução em cada passo é obtida pela solução de uma aproximação do problema não-linear onde o objetivo ($f(x)$) é substituído por uma aproximação quadrática e as restrições não-lineares ($h(x)$ e $g(x)$) são substituídas por aproximações lineares. O método SQP, a cada iteração, resolve o seguinte problema de programação quadrática:

$$\begin{aligned} & \min \nabla^T f(x^k)d + \frac{1}{2}d^T H(x^k, \lambda^k, \mu^k)d \\ & \text{Sujeito a } \begin{cases} h(x^k) + \nabla^T h(x^k)d = 0 \\ g(x^k) + \nabla^T g(x^k)d \leq 0 \end{cases} \end{aligned} \quad (2)$$

para determinar a melhor direção de busca a partir do ponto (x^k) e então proceder a atualização para o próximo ponto $x^{k+1} = x^k + a_k d^k$. Assim como no método de Newton, apesar de o valor ótimo do passo ser $a_k = 1$ valores de $a_k \in (0,1]$ são determinados para garantir as propriedades de convergência global do método. Este procedimento geralmente é feito com o uso de técnicas de *linesearch* (minimização unidirecional de uma função de mérito) ou pela limitação do problema quadrático indicado na região de confiança. A matriz $H(x^k, \lambda^k, \mu^k)$ é uma aproximação positiva definida da matriz Hessiana da função de Lagrange, que pode ser atualizada por qualquer método da métrica variável (DFP – David – Fletcher – Powell, BFGS – Broyden – Fletcher – Goldfarb – Shanno, etc.) de acordo com Sandrini[15]. Para resolver os problemas deste

trabalho, foi utilizada a função *fmincon* do pacote *optimization* do software MATLAB®. Para a atualização da matriz Hessiana foi utilizado o método BFGS. Para maiores detalhes do funcionamento do método SQP consultar Sandrini[15].

4. APRESENTAÇÃO DOS PROBLEMAS

4.1 Metodologia

Neste trabalho serão tratados três problemas considerados *benchmarks* (casos de referência) por terem sido muito explorados e estudados por parte de diversos autores da literatura para avaliar o desempenho dos algoritmos genéticos aplicados à otimização paramétrica de estruturas de engenharia. Dessa maneira, este trabalho será realizado em duas etapas. A primeira etapa é a comparação do Algoritmo Genético com o método SQP para os casos utilizados como *benchmarks*. A segunda etapa é estudar o comportamento e desempenho do referido algoritmo para variação em seus parâmetros de entrada/utilização nestes mesmos exemplos analisados.

Para realizar as duas etapas, pretende-se fazer o uso de um programa de computador específico para Algoritmo Genético. Este programa é o GATOOOL, o *toolbox* de algoritmos genéticos do MATLAB®, *software* consagrado como ferramenta poderosa, de qualidade e de grande utilização pela literatura. O GATOOOL permite a execução das rotinas relacionadas à otimização dos três casos *benchmarks* que são escritas e editadas no próprio MATLAB®. A Tabela 3.1 mostra em resumo os parâmetros de entrada do AG.

Tabela 1 – Parâmetros de entrada do AG.

Tamanho de População	Específico para cada problema
Taxa de Cruzamento	Específica para cada problema
Taxa de Mutação	Específica para cada problema
Método de Seleção	Torneio de tamanho dois
Tipo de Cruzamento	Heurístico
Elitismo	Dois indiv. para cada geração

4.2. Problema da Mola – 3 variáveis a serem otimizadas

O Problema da Mola, segundo Montes [11], consiste em projetar uma mola do tipo helicoidal para cargas de tração e compressão como mostrada na Figura 5. O problema é determinar o diâmetro médio “d” do fio de que é feita, o diâmetro externo

“D” da mola e o número de espiras ativas “n” que resultam no mínimo volume para a mola de forma a suportar a carga com as restrições de projeto do material, frequências naturais e deformações máximas.

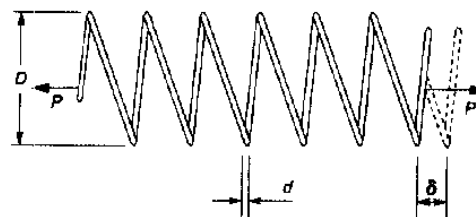


Fig. 5 – Mola do tipo helicoidal para cargas de tração e compressão.

Volume da Mola = Função de Custo = $(n+2) Dd^2$,
ou em termos das variáveis de projeto:

$$f(\vec{x}) = (x_3 + 2)x_2x_1^2 \quad (3)$$

onde $(x_1, x_2, x_3) = (d, D, n)$

Restrição de Deformação Máxima:

$$g_1(\vec{x}) = 1 - \frac{x_2^3x_3}{71.785x_1^4} \leq 0 \quad (4)$$

Restrição da Tensão de Cisalhamento:

$$g_2(\vec{x}) = \frac{x_2(4x_2 - x_1)}{12.566x_1^3(x_2 - x_1)} + \frac{1}{5.108x_1^2} - 1 \leq 0 \quad (5)$$

Restrição de Frequência Natural:

$$g_3(\vec{x}) = 1 - \frac{140,45x_1}{x_2^2x_3} \leq 0 \quad (6)$$

Restrição do Diâmetro Externo:

$$g_4(\vec{x}) = \frac{x_2 + x_1}{1,5} - 1 \leq 0 \quad (7)$$

Limites das variáveis de projeto: $0,05 < x_1 < 0,2$; $0,25 < x_2 < 1,3$ e $2 < x_3 < 15$ (com as mesmas unidades que o trabalho original).

Este problema foi resolvido pelo método SQP e pelo método dos AGs. O método SQP apresenta uma única solução por se tratar de um método determinístico. Já o método dos Algoritmos Genéticos apresenta diferentes soluções para cada combinação de dados de entrada (tamanho de população, taxa de cruzamento e taxa de mutação). Dessa maneira, o método SQP chegou a uma solução que viola ligeiramente a restrição da tensão de cisalhamento ($g_2(x) = 0,00031 > 0$). Já o método dos AGs teve em sua melhor solução o cumprimento estrito de todas as restrições do problema. Para se chegar a melhor solução pelo método dos AGs, o programa deste problema foi executado várias vezes para o ajuste dos parâmetros de entrada. Inicialmente, os valores dos parâmetros utilizados foram os recomendados pela

literatura, segundo Fonseca [3], conforme a Tabela 2. As soluções encontradas para ambos os métodos podem ser vistas na Tabela 3 onde também há um comparativo com a solução encontrada pela literatura, segundo Leite [6] (Na tabela 3 estão os melhores resultados dos AG).

Tabela 2 – Valores para os parâmetros do AG.

Tamanho de População	100, 200 e 400
Taxa de Cruzamento	60%, 80% e 100%
Taxa de Mutação	1%, 5% e 10%

Tabela 3 – Comparação entre as soluções dos métodos SQP e AG com a solução da literatura.

Método	Volume(cm ³)	d(cm)	D(cm)	n
Literatura	0,20779	0,1373	0,9068	11,29
SQP	0,20910	0,1373	1,0580	8,48
Diferença(%)	0,63	5,46	16,68	-24,88
AG	0,20992	0,1310	0,8927	11,72
Diferença(%)	1,03	-0,27	-1,56	3,78

A Figura 6 mostra as 27 combinações possíveis de dados de entrada para o AG com o percentual de sucesso de execução de cada combinação num total de 100%. Cabe salientar que se considerou “execução de sucesso” a execução que obteve o valor ótimo das variáveis sem violar nenhuma restrição do problema. Para cada combinação, executou-se dez vezes o programa de AG (total de 270 execuções) para realizar a comparação das execuções de sucesso. Pode-se perceber que a combinação que obteve maior êxito foi a seguinte: Tamanho de População = 400, Taxa de Cruzamento = 80% e Taxa de Mutação = 1%. Também é notável que, para as combinações Taxa de Cruzamento = 80%, Taxa de Mutação = 10% e Tamanho de População = 100, 200 e 400, não houve execuções de sucesso bem como para as combinações Taxa de Cruzamento = 100%, Taxa de Mutação = 10% e Tamanho de População = 100 e 200.

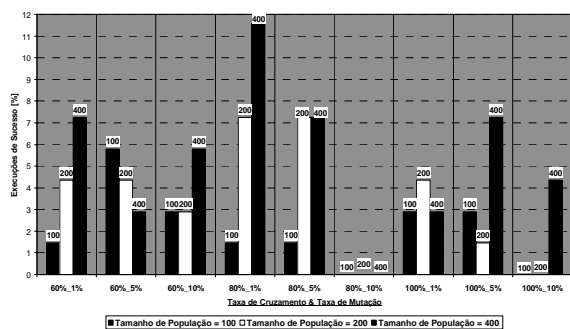


Figura 6 – Taxa de cruzamento & mutação versus percentual de execuções de sucesso.

A Figura 7 exibe o percentual de execuções de sucesso para cada tipo de dado de entrada analisado de maneira isolada, sendo que a Figura 7(a) mostra o “Tamanho de População”, a Figura 7(b) exibe a “Taxa de Cruzamento” e a Figura 7(c) ilustra a “Taxa de Mutação” versus Percentual de execuções de sucesso.

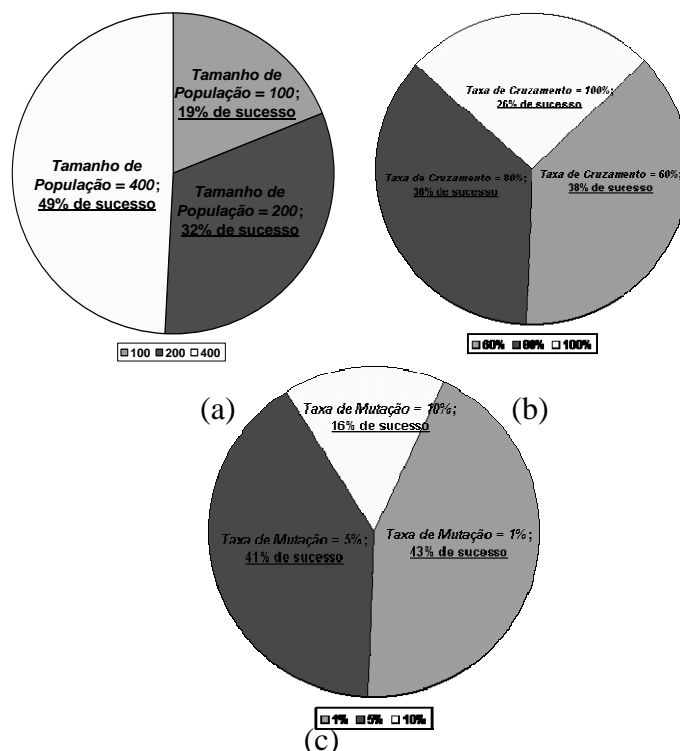


Fig. 7 – (a) Tamanho de população x execuções de sucesso [%]; (b) Taxa de cruzamento [%] e (c) Taxa de mutação [%].

Vistos de modo isolado, pode-se concluir que os dados de entrada com melhores resultados são: Tamanho de População = 400, Taxa de Cruzamento = 60% e Taxa de Mutação = 1%. Entretanto, isto diverge do que foi observado na análise da Figura 7 onde a combinação de melhor êxito encontrada foi com a Taxa de Cruzamento de 80% em vez de 60%. Isto demonstra muito bem o fato de que os parâmetros de entrada do AG interagem de forma complexa e dependente da função que está sendo otimizada.

4.3. Problema da Treliça Espacial de 25 Barras – 8 variáveis a serem otimizadas

O Problema da Treliça Espacial de 25 Barras, segundo Pyrz [12], consiste em minimizar a massa da estrutura mostrada na Figura 8 através de métodos de otimização (AG e SQP), com os esforços calculados através de um programa de

análise de treliça especial que considera um comportamento linear-elástico para as barras da estrutura. O problema é determinar as áreas ótimas das barras que resultarão na menor massa da estrutura sujeitas aos condicionantes abaixo indicados. Dados do material: Módulo de Elasticidade = $6,89 \times 10^4$ MPa e Densidade = $2,770$ $[\text{kg}/\text{m}^3]$. Restrições de deslocamentos nos nós 1 e 2: $|x_1| < 8,89$ mm, $|x_2| < 8,89$ mm, $|y_1| < 8,89$ mm, $|y_2| < 8,89$ mm, $|z_1| < 8,89$ mm e $|z_2| < 8,89$ mm. Restrições de tensões nas barras da estrutura: $|\sigma_{\text{admissível}}| < 275,79$ MPa. Dados geométricos da estrutura estão nas Tabelas 4, 5 e 6.

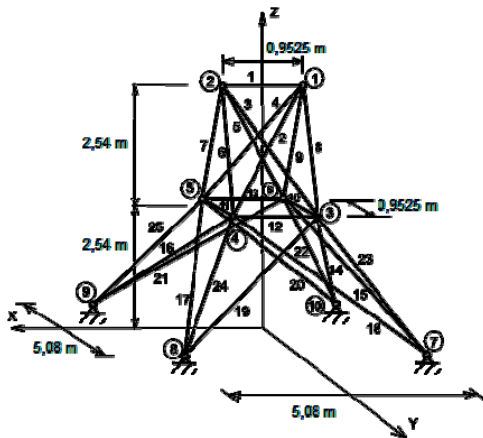


Fig. 8 – Treliça espacial de 25 barras.

Tabela 4- Cargas na estrutura.

Nó	F_x [kN]	F_y [kN]	F_z [kN]
1	4,450	-44,5	-44,5
2	0	-44,5	-44,5
3	2,225	0	0
6	2,268	0	0

Obs.: Cada barra pode ter sua área modificada entre $32,26$ e $2.580,64$ mm^2 .

Tabela 5 – Grupos de barras com mesma área.

Grupo	A1	A2	A3	A4	A5	A6	A7	A8
Conectividades das barras	1-2	1-4, 2-3, 1-5, 2-6	2-4, 2-5, 1-3, 1-6	3-6, 4-5	3-4, 5-6	3-10, 6-7, 4-9, 5-8	4-7, 3-8, 5-10, 6-9	6-10, 3-7, 4-8, 5-9

Tabela 6 - Coordenadas dos nós.

Nó	1	2	3	4	5	6	7	8	9	10
x	-0,95	0,95	-0,95	0,95	0,95	-0,95	-2,54	2,54	2,54	-2,54
y	0	0	0,95	0,95	-0,95	-0,95	2,54	2,54	-2,54	-2,54
z	5,08	5,08	2,54	2,54	2,54	2,54	0	0	0	0

Novamente, este problema foi resolvido pelo método SQP e pelo método dos AGs. Para se chegar à melhor solução pelo método dos AGs, o programa deste problema foi executado várias vezes para o ajuste dos parâmetros de entrada. Inicialmente, os valores dos parâmetros utilizados foram os recomendados pela literatura, segundo Fonseca [3]. O método SQP chegou a uma solução que não transgredir nenhuma restrição do problema. Já o método dos AGs teve, em todas as suas soluções, o descumprimento das restrições de deslocamentos na direção y nos nós um e dois do problema na faixa de 3% a 7%. As soluções para ambos os métodos podem ser vistas na Tabela 7 onde também há um comparativo com a solução encontrada pela literatura segundo Coello e Christiansen [1] usando uma variação de algoritmos genéticos.

Tabela 7 – Comparação entre as soluções da literatura, do método SQP e do método do AG.

Método	Literatura	SQP	Dif.	AG	Dif.
Massa(Kg)	224,05	211,22	-5,73	204,04	-8,93
A1(cm^2)	0,645	0,323	-50,0	0,399	-91,7
A2(cm^2)	4,516	0,323	-92,9	0,374	-91,7
A3(cm^2)	20,645	23,432	13,5	21,503	4,2
A4(cm^2)	0,645	0,323	-50,0	0,673	4,3
A5(cm^2)	9,032	12,850	42,3	12,226	35,4
A6(cm^2)	7,097	5,015	-29,3	4,744	-33,2
A7(cm^2)	3,226	1,021	-68,4	0,466	-85,5
A8(cm^2)	21,935	25,270	15,2	24,679	12,5

Apesar de apresentar descumprimento em algumas restrições do problema, a solução via método dos AGs obteve redução de massa na faixa de 2,16% e 8,93%. Sendo que o valor de 8,93% corresponde à melhor solução. Além disso, a combinação de parâmetros do AG correspondente à melhor solução foi a seguinte: Tamanho de População = 400, Taxa de Cruzamento = 80% e Taxa de Mutação = 10%.

4.4. Problema da Treliça Espacial de 72 Barras – 16 variáveis a serem otimizadas

O Problema da Treliça Espacial de 72 Barras, segundo Erbaturo [2], consiste em minimizar a massa da estrutura mostrada na Figura 9 através de métodos de otimização (AG e SQP) programa de análise de treliça especial que considera um comportamento linear-elástico para as barras da estrutura. O problema é determinar as áreas ótimas das barras que resultarão na menor massa da

estrutura sujeitas aos condicionantes indicados. Dados do material: Módulo de Elasticidade = $6,89 \times 10^4$ MPa e Densidade = 2.770 [kg/m³]. Restrições de deslocamentos em todos os nós nas direções x e y : $|\text{deslocamento}| < 6,35$ mm. Restrições de tensões nas barras da estrutura: $|\sigma_{\text{admissível}}| < 172,37$ MPa. Como cargas na estrutura tem-se no nó 1 $F_x=22,24$ kN, $F_y=22,24$ kN e $F_z=22,24$ kN. Obs.: Cada barra pode ter sua área modificada entre $64,52$ e $1.290,32$ mm².

Tabela 8 – Grupos de barras com mesma área.

Grupo	A1	A2	A3	A4	A5	A6	A7	A8
Barras	1-4	6-12	13-16	17-18	19-22	23-30	31-34	35-36
Grupo	A9	A10	A11	A12	A13	A14	A15	A16
Barras	37-40	42-48	49-52	53-54	55-58	59-66	67-70	71-72

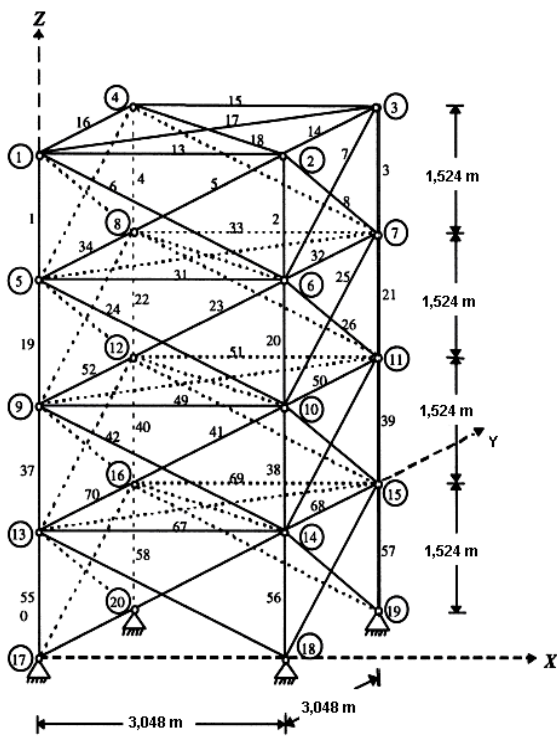


Fig. 9 – Treliça espacial de 25 barras.

Este problema foi resolvido pelo método SQP e pelo método dos AGs. Para se chegar a melhor solução pelo método dos AGs, o programa deste problema foi executado várias vezes para o ajuste dos parâmetros de entrada. Inicialmente, os valores dos parâmetros utilizados foram os recomendados pela literatura, segundo Fonseca e Neves (2004), conforme a Tabela 9. O método SQP chegou a uma solução que não transgride nenhuma restrição do problema assim como o método dos AGs.

As soluções para ambos os métodos podem ser vistas na Tabela 10 onde também há um

comparativo com a solução encontrada pela literatura segundo Erbaturo [2] usando um algoritmo genético modificado.

Tabela 9 – Valores para os parâmetros do AG.

Tamanho de População	25, 50 e 100
Taxa de Cruzamento	60%, 80% e 100%
Taxa de Mutação	1%, 5% e 10%

Tabela 10 – Comparação entre as soluções da literatura, do método SQP e do método do AG.

Método	Literatura	SQP	Dif.(%)	AGs	Dif.(%)
Massa(kg)	174,98	260,62	40,98	166,37	-4,92
A1(cm ²)	1,0	6,7	565,6	0,7	-29,4
A2(cm ²)	3,5	3,9	13,6	3,7	6,2
A3(cm ²)	3,1	3,9	26,6	3,2	1,8
A4(cm ²)	3,4	5,7	68,4	3,5	3,6
A5(cm ²)	3,0	6,7	124,3	3,5	16,6
A6(cm ²)	3,4	3,9	14,7	3,4	-0,1
A7(cm ²)	0,8	3,9	406,5	0,6	-16,1
A8(cm ²)	1,1	3,9	268,4	0,7	-37,5
A9(cm ²)	7,5	6,7	-10,7	7,1	-4,3
A10(cm ²)	3,8	3,9	3,9	3,4	-10,3
A11(cm ²)	0,6	3,9	507,8	0,9	35,1
A12(cm ²)	0,6	3,9	507,8	0,8	17,8
A13(cm ²)	11,3	6,7	-41,2	10,2	-9,5
A14(cm ²)	3,3	4,2	29,7	3,2	-1,1
A15(cm ²)	0,7	3,9	478,9	0,7	-2,6
A16(cm ²)	1,0	4,4	336,4	0,7	-30,6

Apesar de cumprir todas as restrições do problema, a solução via método SQP não obteve redução de massa, pelo contrário, a massa obtida ficou 48,94% maior em relação à da literatura. Por outro lado, as soluções via método dos AGs conseguiram redução de massa de até 4,92%, sendo que o valor de 4,92% corresponde à melhor solução. Além disso, a combinação de parâmetros do AG correspondente à melhor solução foi: Tamanho de População = 100, Taxa de Cruzamento = 60% e Taxa de Mutação = 5%. Dentre as 27 soluções diferentes obtidas via método dos algoritmos genéticos, nem todas conseguiram redução de massa da estrutura em relação à resposta da literatura sendo notável que o pior desempenho se deu para as nove combinações de parâmetros que possuíam Taxa de Cruzamento = 100%, onde as respostas destas combinações tiveram aumento de massa na faixa de 25% a 65% em relação à da literatura. Em contrapartida, sete das 27 combinações possíveis resultaram em redução de massa na faixa de 0,3% a 4,92%. Estas sete combinações estão mostradas na Tabela 11

juntamente com a redução de massa correspondente.

Tabela 11 – Combinações dos parâmetros de entrada dos AGs com redução de massa.

Combinação	A	B	C	D	E	F	G
Tam. Pop.	50	50	50	100	100	100	100
Taxa Cruz.(%)	60	60	80	60	70	60	80
Taxa Mut.(%)	5	10	10	5	5	10	10
Redução Massa(%)	2,37	0,58	3,22	4,92	1,99	2,41	2,75

Observando-se a Tabela 11 é possível notar que ocorreu redução da massa da estrutura para as combinações que possuíam os parâmetros Tamanho de população = 50 ou 100, Taxa de cruzamento = 60% ou 80% e Taxa de mutação = 5% ou 10%, ou seja, para os parâmetros Tamanho de população = 25, Taxa de cruzamento = 100% e Taxa de mutação = 1% não houve redução de massa em suas soluções, mas em muitas delas a massa da estrutura praticamente se igualou ao valor alcançado pela literatura.

4. CONCLUSÃO

Com o que foi apresentado, pode-se concluir que a utilização do Algoritmo Genético como método de otimização paramétrica de estruturas de engenharia é viável, pois, para os três problemas propostos, o método atingiu respostas relativamente melhores que às encontradas pela literatura sem desrespeitar alguma restrição, exceto para o Problema da Treliça Espacial de 25 Barras onde houve descumprimento na faixa de 3% a 7% das restrições de deslocamentos na direção y nos nós um e dois do problema. Isto pode ter ocorrido devido à Função de Penalização não ter sido suficiente para fazer com que os indivíduos que violaram alguma das restrições do problema viessem a ter uma avaliação baixa a fim de que não pudessem ser escolhidos como solução ótima. Além disso, o Algoritmo Genético mostrou melhor desempenho que o método SQP em dois dos três problemas propostos, quais sejam, o Problema da Mola e o Problema da Treliça Espacial de 72 Barras sendo que, no Problema da Mola, o método SQP violou ligeiramente a restrição da tensão de cisalhamento da mola e no Problema da Treliça Espacial de 72 Barras chegou a um valor “ótimo” da massa da estrutura 48,94% maior em relação ao da literatura quando, em contrapartida, o método

dos AGs obteve um valor ótimo 4,92% menor em relação ao da literatura.

Também se pode perceber que o desempenho do método dos AGs, por ser um método probabilístico, é extremamente sensível aos parâmetros de entrada escolhidos para sua execução, pois para cada um dos três problemas encontrou-se uma combinação de parâmetros distinta que fez com que o algoritmo alcançasse o seu melhor desempenho.

Para o problema da mola, pôde-se concluir que os dados de entrada com melhores resultados são: Tamanho de População = 400, Taxa de Cruzamento = 60% e Taxa de Mutação = 1%. Entretanto, isto divergiu do que foi observado na análise da Figura 7 onde a combinação de melhor êxito encontrada foi com a Taxa de Cruzamento de 80% em vez de 60%. Pôde-se perceber, também, que, para as combinações Taxa de Cruzamento = 80%, Taxa de Mutação = 10% e Tamanho de População = 100, 200 e 400, não houve execuções de sucesso bem como para as combinações Taxa de Cruzamento = 100%, Taxa de Mutação = 10% e Tamanho de População = 100 e 200. Isto pode ter ocorrido, pois uma alta taxa de cruzamento permite uma exploração maior do espaço de busca e reduz as chances de convergência para um ótimo local. Entretanto, se essa taxa for muito alta pode resultar na perda de tempo computacional devido à exploração de regiões não promissoras dentro do espaço de busca (Souza Jr. [16]).

Para o Problema da Treliça Espacial de 25 Barras, as soluções de todas as combinações de parâmetros do AG obtiveram redução de massa, sendo que a combinação correspondente à melhor solução foi a seguinte: Tamanho de População = 400, Taxa de Cruzamento = 80% e Taxa de Mutação = 10%. Entretanto, isto ocorreu com descumprimento das restrições de deslocamentos na direção y nos nós um e dois do problema.

No Problema da Treliça Espacial de 72 Barras, foi possível notar que ocorreu redução da massa da estrutura para as combinações que possuíam os parâmetros Tamanho de População = 50 ou 100, Taxa de Cruzamento = 60% ou 80% e Taxa de Mutação = 5% ou 10%, ou seja, para os parâmetros Tamanho de População = 25, Taxa de Cruzamento = 100% e Taxa de Mutação = 1% não houve redução de massa em suas soluções, mas em muitas delas a massa da estrutura praticamente se igualou ao valor alcançado pela literatura. Os resultados mostram que o algoritmo genético

torna-se mais eficiente quando há uma maior população. Isto se explica pelo fato de o tamanho da população influenciar o número de pontos do espaço de busca analisados a cada iteração, contudo, caso este número seja grande demais, o algoritmo demorará demais e pode estar se aproximando de uma busca exaustiva. Além disso, se o valor da probabilidade atribuída ao operador de mutação for baixo demais, ele agirá de forma extremamente moderada e a população não terá diversidade depois de certo número de gerações, estagnando rapidamente devido à convergência genética.

Como proposta de trabalhos futuros sugere-se a determinação dos parâmetros de entrada dos algoritmos genéticos de forma evolutiva e automática, criando-se, assim, um algoritmo onde os parâmetros são variáveis a fim de melhorar o desempenho dos resultados referentes à função de custo e uma maior estabilidade destes resultados em iterações distintas evitando o uso de parâmetros iniciais fixos e constantes ao longo das iterações.

REFERÊNCIAS

1. Coello, C. A.; Christiansen, A. D., *Multiobjective optimization of trusses using genetic algorithms*. Computers and structures, Cambridge, n. 75, p. 647-660, 2000.
2. Erbatur, F.; Hasağebi, O.; Tütüncü, I.; Kiliç, H., *Optimal design of planar and space structures with genetic algorithms*. Computers and Structures, Cambridge, n. 75, p. 209-224, 2000.
3. Fonseca, M.; Neves, F.A., *Algoritmos genéticos aplicados à otimização discreta de estruturas treliçadas espaciais*. Universidade Federal de Ouro Preto, 2004. 15f.
4. Holland, J.H., *Adaptation in natural and artificial systems*. Michigan: University of Michigan Press, Ann Harbor, 1975.
5. Konzen, P.H.A.; Furtado, J.C.; Carvalho, C.W.; Ferrão, M.F.; Molz, R.F.; Bassani, I.A.; Hüning, S.L., *Otimização de métodos de controle de qualidade de fármacos usando algoritmo genético e busca tabu*. Universidade de Santa Cruz do Sul, 2002. 19 f.
6. Leite, J.P.B.; Topping, B.H.V., *Improved genetic operators for structural engineering optimization*. Advances in engineering software, Cambridge, v. 29, n. 8, p. 529-562, 1998.
7. Linden, R., *Algoritmos genéticos: uma importante ferramenta da inteligência computacional*. Rio de Janeiro: Brasport, 2006.
8. Malaquias, N.G.L., *Uso dos algoritmos genéticos para a otimização de rotas de distribuição*. Dissertação de Mestrado – Universidade Federal de Uberlândia, 2006. 113 f.
9. Matlab, *Optimization toolbox user's guide*. Natick: Mathworks, 2007.
10. Miasaki, C.T., *Planejamento da expansão do sistema de transmissão de energia elétrica utilizando controladores facts*. Tese de Doutorado – Universidade Estadual Paulista, Ilha Solteira, 2006. 158 f.
11. Montes, E.M.; Coello, A.A.C.; Becerra, R. L., *Engineering optimization using a simple evolutionary algorithm*, in: International conference on tools with artificial intelligence, 15, 2003, California.
12. Pyrz, M.; Zawidska, J., *Optimal discrete truss design using improved sequential and genetic algorithm*. Engineering computations, Bingley, v. 18, n. 8, p. 1078-1090, 2001.
13. Rodrigues, N.M., *Um algoritmo cultural para problemas de despacho de energia elétrica*. Dissertação de Mestrado – Universidade Estadual de Maringá, 2007. 99 f.
14. Romão, W.; Niederauer, C.A.P.; Martins, A.; Tcholakian, A.; Pacheco, R.C.S., *Algoritmos genéticos e conjuntos difusos aplicados ao controle de um processo térmico*. Universidade Federal de Santa Catarina, Florianópolis, 1997. 14 f.
15. Sandrini, V.S., *Um estudo da otimização da geometria de um pára-quadras simplificado*. Dissertação de Mestrado – Universidade Federal do Rio Grande do Sul, Porto Alegre, 2005. 93 f.
16. Souza jr., D.A., *Otimização pelo método dos algoritmos genéticos e dimensionamento de estruturas tubulares metálicas espaciais com barras cruzadas para coberturas*. Tese de doutorado – Universidade Federal de Uberlândia, 2005. 145 f.
17. Goldberg, D.E., *Genetic Algorithms in search, optimization and machine learning*. Reading Addison-Wesley Publishing, MA, 1989.