UMA ABORDAGEM HÍBRIDA PARA RESOLVER PROBLEMAS COM DUAS RESTRIÇÕES NÃO LINEARES DE IGUALDADE

Carlos Henrique Fonseca
Centro Federal de Educação Tecnológica de Minas Gerais
Av. Amazonas 7675 - Nova Gameleira, Belo Horizonte - MG, 30510-000
fonseca.carlosh@gmail.com

Elizabeth Fialho Wanner
Departamento de Computação - DECOM
Centro Federal de Educação Tecnológica de Minas Gerais
Av. Amazonas 7675 - Nova Gameleira, Belo Horizonte - MG, 30510-000
efwanner@gmail.com

Flávio Vinícius Cruzeiro Martins
Departamento de Computação - DECOM
Centro Federal de Educação Tecnológica de Minas Gerais
Av. Amazonas 7675 - Nova Gameleira, Belo Horizonte - MG, 30510-000
flaviocruzeiro@decom.cefetmg.br

RESUMO

Este trabalho apresenta um mecanismo de busca local para algoritmos genéticos que melhoram a velocidade de convergência do algoritmo quando aplicado a problemas com duas restrições não lineares de igualdade. O operador de busca local proposto, denominado Multisseção, utiliza aproximações quadráticas para todas as funções do problema e emprega um método baseado no algoritmo da bisseção para encontrar a solução do problema. O operador é um método livre de derivada e não necessita de nenhuma avaliação extra de função além das avaliações que estão disponíveis a cada geração do algoritmo. Os algoritmos foram aplicados a um conjunto de problemas teste e os resultados obtidos mostram que o algoritmo genético híbrido melhora a velocidade de convergência quando comparado ao algoritmo tradicional.

PALAVRAS CHAVE. Otimização não-linear, Restrições de igualdade, Operadores de busca local

Área Principal: Otimização Mono-Objetivo

ABSTRACT

This paper introduces a local search tool for genetic algorithms that improve the convergency speed of the algorithm when applied to problems with two non linear equality constraints. The local search operator, named Multisseção, uses quadratic approximations for all problem's functions and uses a bisection-based method to find the problem solution. This derivative free operator doesn't need any extra evaluations of functions besides the ones avaiable at each genetic algorithms generation. The algorithms were applied to a set of test problems and the results obtained show that the hibrid algorithm increases the convergency speed in comparison to the traditional algorithm.

KEYWORDS. Non linear optimization. Equality constraints. Local search operators.

Main Area: Mono Objective Optimization

1. Introdução

A solução de problemas de otimização é de suma importância para as mais diversas áreas tais como engenharia de produção, economia, biologia, dentre outras. Muitos processos realizados atualmente podem ser otimizados de forma a diminuir custos ou economizar matéria-prima, fornecendo assim um benefício, seja financeiro ou de racionamento, para atingir determinadas metas.

Problemas não lineares com restrições de igualdade constituem um desafio dentro da área de otimização, uma vez que a região factível é bastante restrita, fazendo-se necessário que haja métodos específicos para a solução desses problemas. O método mais comum para tratamento de restrições utilizado o método de penalidade (RICHARDSON et. al, 1989), que consiste em atribuir uma penalidade a pontos que estejam fora da região factível do problema. As restrições são agregadas à função objetivo via um parâmetro de penalidade associado à violação das restrições. Apesar de simples, esta metodologia apresenta algumas dificuldades. Se o parâmetro é grande, uma maior ênfase é dada à factibilidade fazendo com que o método convirja prematuramente para um ponto factível. Por outro lado, se o parâmetro for pequeno, o método irá passar muito tempo explorando a região infactível e a solução final apresentará uma violação alta das restrições.

Devido aos problemas apresentados inerentes aos métodos de penalidade, alguns esforços têm sido feitos na tentativa de lidar com os problemas com restrições (COELLO, 2002). Entretanto, todos a maior parte dos métodos tratam apenas de restrições de desigualdade.

As restrições de igualdade constituem um problema para métodos de otimização, uma vez que esse tipo de restrição define um conjunto factível de dimensão menor que o espaço das variáveis do problema. Usualmente, as restrições de igualdade são, primeiramente, transformada em restrições de desigualdade através de uma relaxação e, depois, o problema com restrições de desigualdade é resolvido.

O trabalho proposto por Wanner (2005) propõe um operador de busca local para Algoritmos Genéticos aplicados a problemas não lineares com uma restrição de igualdade. Este operador usa as avaliações de função, objetivo e de restrição, disponíveis a cada geração do Algoritmo Genético para construir um problema quadrático associado. Este problema é, então, resolvido usando as condições de Karush-Kuhn-Tucker e um método de bisseção. A solução do problema associado é inserida, deterministicamente, na população do Algoritmo Genético. Os resultados mostraram que a metodologia proposta apresenta uma velocidade de convergência superior a do algoritmo padrão.

Apesar dos resultados do trabalho apresentado por Wanner (2005) serem promissores, a metodologia só poderia ser aplicada a problemas com uma única restrição de igualdade. Numa tentativa de resolver problemas com várias restrições de igualdade, Peconick (2007) propôs uma metodologia que construia aproximações quadráticas para as restrições lineares. Usando um algoritmo de projeções iterativo, as projeções das restrições de igualdade de alguns pontos sobre a superfície restrita aproximada. Os pontos obtidos, que atendiam às restrições de igualdade, eram inseridos na população do Algoritmo Genético. Entretanto, os autores identificaram alguns casos nos quais a metodologia não podia ser aplicada. Os resultados, nos casos aplicáveis, mostraram uma velocidade de convergência superior quando comparado ao algoritmo tradicional.

Este trabalho tem como objetivo estender a metodologia proposta em Wanner (2005) para a solução de problemas com duas restrições de igualdade. Um operador de busca local, baseado em aproximações quadráticas e em uma adaptação ao método de bisseção, será proposto. Este operador será inserido no ciclo evolutivo do Algoritmo Genético tradicional e, usando as avaliações disponíveis pelo algoritmo para as funções objetivo e de restrição, irá construir um problema quadrático associado. Este problema associado será resolvido usando as equações de Karush-Kuhn-Tucker e uma metodologia baseada no método de bisseção.

O operador de busca local foi acoplado a um Algoritmo Genético e aplicado em alguns problemas analíticos. Os resultados, comparados com o Algoritmo Genético padrão, mostram que a metodologia proposta pode ser eficientemente aplicada aos problemas não lineares com duas restrições não lineares de igualdade.

2. Aproximações Quadráticas

Considere o problema de otimização com duas restrições de igualdade descrito por (1)

$$x^* = \underset{x}{\operatorname{argmin}}\; f(x) \qquad s.a. \quad g_i(x) = 0, i = 1, 2 \tag{1}$$

sendo $x \in R^n$, $f(\cdot) : R^n \to R$ a função objetivo e $g_i(\cdot) : R^n \to R$ as restrições de igualdade.

Suponha que as funções possam ser aproximadas localmente por funções quadráticas da seguinte forma:

$$f(x) = (x - x_f)^T H(x - x_f) \tag{2}$$

e

$$g_i(x) = (x - x_{g_i})^T G(x - x_{g_i}) - 1 \tag{3}$$

sendo x_f o ponto ótimo da função aproximada e x_{g_i}, $i = 1, 2$ o centro de cada restrição de igualdade.

A construção das aproximações quadráticas para as funções envolvidas no problema é um método livre de derivadas e essa construção pode ser feita localmente de forma quadrática. No inicio do processo de busca, essas aproximações tendem a não ser precisas, porém, à medida que o processo vai convergindo, as aproximações locais tendem a ser mais acuradas e o ponto de solução do problema quadrático associado tende a coincidir com o problema real. As avaliações de funções, fornecidas pelo Algoritmo Genético (AG), serão usadas para calcular as aproximações quadráticas, não sendo necessário, assim, avaliar as derivadas das funções. A vantagem de se usar um método livre de derivadas consiste no fato de que algumas funções podem ser desconhecidas ou até mesmo não diferenciáveis, além de não ser necessária a avaliação extra de função para calcular o gradiente.

O método apresentado em Wanner (2005) consiste em encontrar uma aproximação quadrática da forma

$$h(z) = z^T Hz + r^T z + \gamma \tag{4}$$

utilizando pontos z_1, z_2,..., z_N fornecidos pela própria população utilizada no AG. Seja H uma matriz simétrica, r um vetor e γ um escalar, é possível encontrar

$$f(z_i) \approx h(z_i), i = 1, 2, ..., N \tag{5}$$

Observe que o problema da construção da aproximação quadrática transforma-se em um problema de encontrar H, r e γ com erro de aproximação dado por

$$E(i) = z_i^T Hz_i + r^T z_i + \gamma - h(z_i), i = 1, 2, ..., N \tag{6}$$

A solução desse problema se dá através da solução do sistema linear de N equações nas variáveis H, r e γ. O número total de variáveis do problema é dado por $\frac{(n+1)(n+2)}{2}$, em que n representa a dimensão do problema.

Se $N > \frac{(n+1)(n+2)}{2}$, o sistema é sobre-determinado e pode-se encontrar uma solução de norma mínima da forma

$$\min_{H,r,\gamma} ||E|| \tag{7}$$

Se a norma euclidiana é usada, então $h(z)$ é a aproximação obtida através dos métodos dos quadrados mínimos, mas a convexidade da função não é garantida. Sendo assim, pode-se obter $h(z)$ através de

$$min \left(t : \sqrt{\sum_i E_i^2} \leq t, H > 0 \right) \quad (8)$$

garantindo assim que a matriz H é positiva definida sendo a função, portanto, convexa. A equação (8) pode ser resolvida usando o SeDuMi (STURM, 1999).

Como demonstrado em Wanner (2005), é possível encontrar uma expressão para essa aproximação no formato

$$h(z) = (z - z_h)^T H(z - z_h) - c \quad (9)$$

sendo z_h o mínimo de h e $c = 0.25r^T H^{-1} r + \gamma$.

Para a utilização desse método de aproximação, portanto, é necessário ter pelo menos $\frac{(n+1)(n+2)}{2}$ pontos disponíveis.

Como o objetivo é uma aproximação local, o melhor indivíduo de um AG é selecionado e uma vizinhança deste indivíduo é escolhida. Os pontos no interior da vizinhança são utilizados para a aproximação quadrática. Caso não haja pontos suficientes, a aproximação quadrática não é construída. O tamanho da vizinhança é um parâmetro externo que pode ser escolhido pelo otimizador. No caso deste trabalho foi utilizada uma vizinhança gerada em torno o melhor ponto em um raio $r = 0.1$.

3. Multisseção

O operador de busca local, nomeado Multisseção, resolve problemas com duas restrições de igualdade, podendo ser considerado, na verdade, uma extensão do método apresentado em Wanner (2005), utilizando uma ideia de bisseção para mais dimensões.

A partir de um problema de minimização com restrições de igualdade, escrito de maneira geral na seguinte forma

$$\begin{aligned} min \quad & f(x) \\ s.a. \quad & h_j(x) = 0, j = 1, 2 \end{aligned} \quad (10)$$

suponha que as funções sejam quadráticas. Assim, depois de uma mudança de coordenadas apropriada, o problema se torna

$$x^* = \underset{x}{\operatorname{argmin}} \, (z - z_f)^T (z - z_f) \quad (11)$$

$$s.a. \begin{cases} (z - z_{g1})^T Q_1 (z - z_{g1}) - c_1 = 0 \\ (z - z_{g2})^T Q_2 (z - z_{g2}) - c_2 = 0 \end{cases}$$

Observe que

$$\nabla f(x) = 2(z - z_f) \quad (12)$$

$$\nabla h_1(x) = 2Q_1(z - z_{g_1}) \quad (13)$$

$$\nabla h_2(x) = 2Q_2(z - z_{g_2}) \quad (14)$$

Desta forma, a condição de Karash-Kuhn-Tucker, assumindo-se que as *constraint qualifications* são satisfeitas, pode ser obtida da seguinte forma:

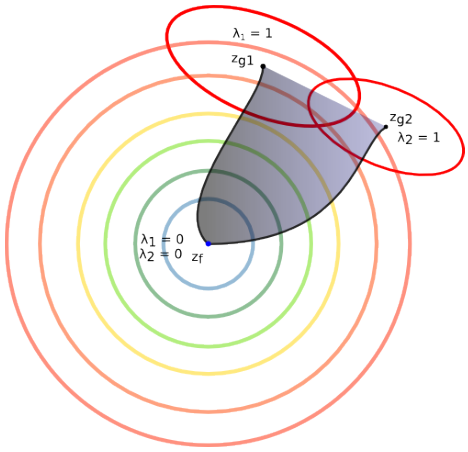
$$\nabla f(x) + \lambda_1 \nabla h_1 + \lambda_2 \nabla h_2 = 0$$

$$2(z - z_f) + 2\lambda_1 Q_1 (z - z_{g_1}) + 2\lambda_2 Q_2 (z - z_{g_2}) = 0$$

$$(I + \lambda_1 Q_1 + \lambda_2 Q_2) z = z_f + \lambda_1 Q_1 z_{g_1} + \lambda_2 Q_2 z_{g_2} \quad (15)$$

Observe que a equação (15) parametriza uma região delimitada por uma curva que une z_f a z_{g_1} e uma curva que une z_f a z_{g_2}. Pode-se notar que quando um dos $\lambda_i = 0$, a equação representa o caminho de z_f a z_{g_i} (o centro da outra restrição). Se $\lambda_i = 0, \forall i$, temos $z = z_f$ (centro da função irrestrita) e para outros valores de λ obtêm-se a região delimitadas pelos caminhos. A Figura 1 mostra essa região parametrizada pela equação (15) em um problema bidimensional.

Figura 1: Área delimitada pelos caminhos parametrizados entre o mínimo irrestrito e os mínimos das aproximações quadráticas

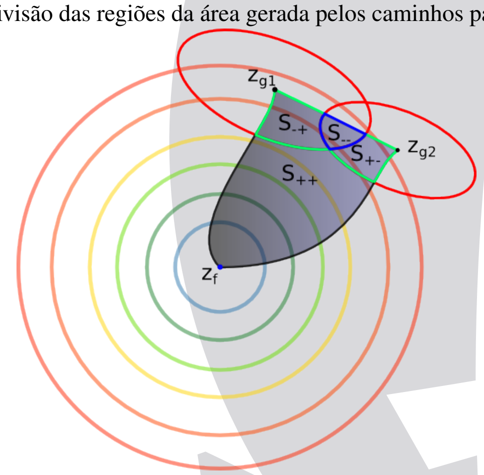


Essa área pode ser dividida em quatro regiões $\{S_{--}, S_{-+}, S_{+-}, S_{++}\}$. As regiões correspondem ao valor dos pontos internos a elas quando avaliados nas restrições e se definem da seguinte maneira:

- S_{--}: região que apresenta pontos x, tais que $g_1(x) < 0$ e $g_2(x) < 0$.
- S_{+-}: região que apresenta pontos x, tais que $g_1(x) > 0$ e $g_2(x) < 0$.
- S_{-+}: região que apresenta pontos x, tais que $g_1(x) < 0$ e $g_1(x) > 0$.
- S_{--}: região que apresenta pontos x, tais que $g_1(x) > 0$ e $g_1(x) > 0$.

A Figura 2 mostra as quatro regiões denominadas $\{S_{--}, S_{-+}, S_{+-}, S_{++}\}$.

Figura 2: Divisão das regiões da área gerada pelos caminhos parametrizados

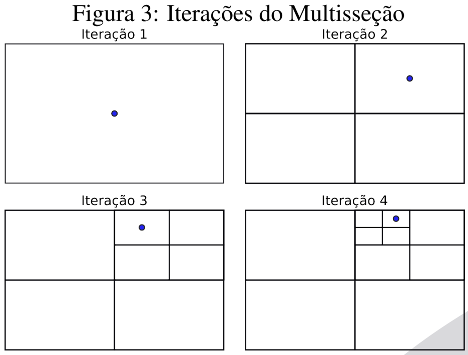


No caso de problemas com uma única restrição de igualdade, a ideia do mecanismo de bisseção é muito simples. Sobre algum intervalo, uma função atinge o zero se o sinal da função muda. Caminhando sobre o caminho parametrizado que une z_f a z_g, temos duas regiões distintas: $S_+ = \{z|g(x) > 0\}$ e $S_- = \{z|g(z) < 0\}$. O ponto procurado z^* é aquele que está sobre a restrição e portanto $g(z^*) = 0$. Portanto, dado um intervalo sobre esse caminho, é possível garantir a existência de tal ponto. Usando o algoritmo de bisseção, divide-se o intervalo até obter z^* com uma dada precisão ϵ.

A mesma ideia pode ser usada no caso com duas restrições de igualdade. Pode-se dividir o retângulo que contém as quatro regiões, $\{S_{--}$,S_{-+},S_{+-},$S_{++}\}$, com o objetivo de se obter um retângulo menor que ainda contenha z* (vide Figura 3).

Seguindo a ideia baseada na bisseção onde dividimos o valor de λ iterativamente de modo que sempre terminamos com um λ que retorne $h(x)$ positivo e outro que retorne $h(x)$ negativo, a ideia aqui proposta é sempre caminhar nas regiões S_{--} e S_{++}. Devemos, portanto, sempre selecionar uma dessas regiões e subdividi-la em quatro regiões de maneira iterativa, como mostrado na Figura 3, para o caso de duas restrições.

Figura 3: Iterações do Multisseção



O algoritmo estima em qual quadrante pode estar a região S_{--} com base na informação de cada restrição. D,essa maneira a região é dividida iterativamente com o objetivo de se obter uma área menor que ainda contenha o ponto de interseção. Os passos deste método são os apresentados no Algoritmo 1, onde através das equações das restrições e do mínimo irrestrito da função objetivo, obtém-se o ponto de interseção das duas restrições que está mais próximo do centro da função objetivo.

O método recebe como parâmetro de entrada uma constante ϵ utilizada como critério de parada. O método executa as suas iterações enquanto $|x_a - x_b| > \epsilon$, ou seja enquanto o ponto encontrado não for menor do que esse critério de tolerância.

4. Algoritmo Genético

O Algoritmo Genético utilizado nesse trabalho foi apresentado em Takahashi et al. (2003) e Ramos et al. (2003) e é denominado Real-Biased Genetic Algorithm (RBGA). Este algoritmo possui codificação real e um operador de cruzamento polarizado com tendência para o melhor dos pais. A seleção utilizada foi feita por Roleta e o melhor indivíduo da população é deterministicamente inserido na próxima população.

O operador de cruzamento tendencioso funciona da seguinte maneira: para cada par de indivíduos selecionados com probabilidade p_c, um dos filhos, x_g, é obtido da seguinte maneira:

$$x_g = \alpha x_1 + (1 - \alpha)x_2 \tag{16}$$

sendo x_1 o pai com a melhor aptidão, x_2 o outro pai e α um valor no intervalo $-\xi, 1 + \xi$, com ξ sendo um fator de extrapolação dentro do intervalo $[0, 1]$. O valor α é escolhido de acordo com a distribuição probabilística definida por

$$\alpha = (1 + 2\xi)\beta_1\beta_2 \tag{17}$$

Algoritmo 1: Método da Multisseção

```
Input: f(·), l_a, l_b, ε
Output: x_a
while não Criterio de Parada do
    x_a ← x(l_ia, ∀i)
    x_b ← x(l_ib, ∀i)
    g_a ← g(x_a)
    g_b ← g(x_b)
    for i ← 1 to m do
        M ← g_ib
        l_ik ← (l_ia + l_ib)/2 ∀i
        x_k ← x(l_ik,∀i)
        g_k ← f(x_k)
        for i ← 1 to m do
            if M * g_ik > 0 then
                l_ib ← l_ik
                x_b ← x_k
            else
                l_ia ← l_ik
                x_a ← x_k
            end
        end
    end
end
```

sendo β_1 e β_2 variáveis aleatórias com distribuição probabilística uniforme dentro do domínio $[-1, 1]$. Dessa maneira, α possui uma distribuição quadrática que faz com que o filho tenha uma maior probabilidade de estar mais próximo de x_1 (o melhor pai) do que de x_2 (o pior pai).

O outro filho escolhido de maneira não tendenciosa: α é escolhido no intervalo $[-\xi, 1+\xi]$ com probabilidade uniforme.

Para o tratamento de restrições foi utilizado o método de penalidade e o critério de parada utilizado foi um número máximo de iterações.

4.1. Algoritmo Genético Hibridizado

O AG Híbrido implementado neste trabalho tem características similares ao AG apresentado na Seção 4, mas ele apresenta um operador de busca local elitista. Este operador substitui o pior indivíduo da população pelo indivíduo encontrado durante a busca local. O algoritmo 2 apresenta o pseudo-código do AG acoplado ao operador de busca local.

Para hibridizar o AG com o operador de busca local, algumas definições foram estabelecidas:

- O operador de busca local executado sobre a população corrente do AG a cada cinco gerações.
- Após gerar uma vizinhança ao redor do melhor ponto da população corrente do AG, os indivíduos dentro desta vizinhança serão usados para gerar as aproximações quadráticas das funções do problema.
- Uma vez que as aproximações são geradas, o Multisseção é aplicado sobre elas a fim de encontrar o ponto de interseção entre as restrições. Caso a vizinhança não possua o número de indivíduos necessários para a construção das aproximações, o AG prossegue normalmente.

Algoritmo 2: Algoritmo Genético Hibridizado

Input: função $f(\cdot)$, taxa de cruzamento t_c, taxa de mutação t_m, tamanho da população n
Output: ponto ótimo x^*
$p_0 \leftarrow$ PopulaçãoInicial()
while *não* $CriteriodeParada$ **do**
  $pop \leftarrow$ Seleção()
  $pop \leftarrow$ Cruzamento(t_c)
  $pop \leftarrow$ Mutação(t_m)
  $pop(n) \leftarrow$ BuscaLocal()
end
$x^* \leftarrow pop(1)$

- Finalmente, o ponto obtido com o operador de busca local será deterministicamente inserido na população do AG.

5. Resultados

O AG padrão e o AG híbrido foram testados usando um conjunto de problemas analíticos que foram escolhidos com características distintas e graus de dificuldade diversos. Os problemas estão listados a seguir:

Problema 1:

$$x^* = \underset{x}{\operatorname{argmin}}\ x_1^2 + x_2^2 \tag{18}$$
$$s.a. \begin{cases} (x_1-2)^2 + (x_2-8)^2 - 4 = 0 \\ (x_1-5)^2 + (x_2-5)^2 - 9 = 0 \\ -10 \leq x_i \leq 10, i = 1, 2 \end{cases}$$

Problema 2:

$$x^* = \underset{x}{\operatorname{argmin}}\ (x_1^2 + x_2^2)cos(x_1) \tag{19}$$
$$s.a. \begin{cases} (x_1-2)^2 + (x_2-8)^2 - 4 = 0 \\ (x_1-5)^2 + (x_2-5)^2 - 9 = 0 \\ -10 \leq x_i \leq 10, i = 1, 2 \end{cases}$$

Problema 3:

$$\begin{aligned} x^* = \underset{x}{\operatorname{argmin}}\ & x_1^2 + x_2^2 + x_1x_2 - 14x_1 - 16x_2 + (x_3-10)^2 + 4(x_3-5)^2 \\ & + (x_5-3)^2 + 2(x_6-1)^2 + 5x_7^2 + 7(x_8-11)^2 \\ & + 2(x_9-10)^2 + (x_{10}-7)^2 + 45 \end{aligned} \tag{20}$$
$$s.a. \begin{cases} 3(x_1-2)^2 + 4(x_2-3)^2 + 2x_3^2 - 7x_4 - 120 = 0 \\ 5x_1^2 + 8x_2 + (x_3-6)^2 - 2x_4 - 40 = 0 \\ -10 \leq x_i \leq 10, i = 1, ..., 10 \end{cases}$$

O AG utilizado possui os seguintes parâmetros, escolhidos empiricamente:

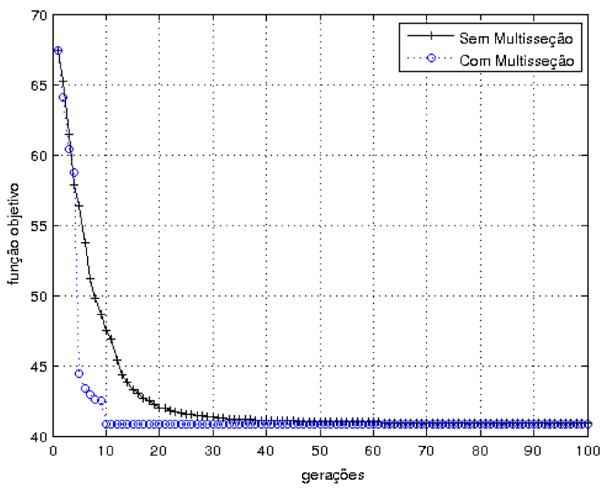
- tamanho da população: 100
- taxa de cruzamento: 0.6

- taxa de polarização: 0.3
- taxa de mutação: 0.02
- raio de mutação: 0.05
- fator de dispersão: 1.8
- fator de extrapolação: 0.2
- penalidade das restrições: 100
- número máximo de gerações: 100

Nos testes realizados neste trabalho, o único critério de parada utilizado foi o número máximo de gerações. Tanto o Algoritmo Genético puro quanto o método híbrido foram executados 30 vezes sobre a mesma população inicial. Ao final das 30 execuções,a curva de convergência média foi obtida. Essa curva corresponde à média dos melhores indivíduos dos algoritmos ao longo das 100 gerações e fornece uma forma simples de se verificar se houve uma melhora na velocidade de convergência do algoritmo proposto quando comparado ao original.

As Figuras 4, 5 e 6 mostram a curva de convergência média para o Problema 1, 2 e 3, respectivamente. Nos gráficos, o eixo x representa as gerações enquanto que o eixo y representa o valor do melhor indivíduo da geração correspondente.

Figura 4: Curva de Convergência Média para o Problema 1



Pode-se notar, na Figura 4, que na geração 10 o algoritmo híbrido já encontra a solução final do problema. Isso acontece devido às características quadráticas do problema. Ainda é possível notar que a inserção do operador de busca local trouxe um impacto positivo ao AG, chegando ao resultado ótimo, cerca de 25 gerações mais cedo.

Na Figura 5 percebe-se a ação do Multisseção nas gerações 5, 10, 15 e 20, quando o ponto final é encontrado. A curva média mostra ainda que a convergência do algoritmo híbrido acontece bem mais rapidamente que a do AG puro.

A Figura 6 mostra que o método híbrido converge mais rapidamente que o AG puro a partir da geração 50, fazendo com que ele chegue ao ponto final mais rapidamente.

Pelos resultados apresentados, pode-se notar que o algoritmo híbrido sempre tem uma curva de convergência média pelo menos tão boa quanto a do AG puro. Todos os resultados apresentados nesta seção e as discussões sobre os mesmos baseiam-se apenas na curva de convergência média dos dois métodos.

Figura 5: Curva de Convergência Média para o Problema 2

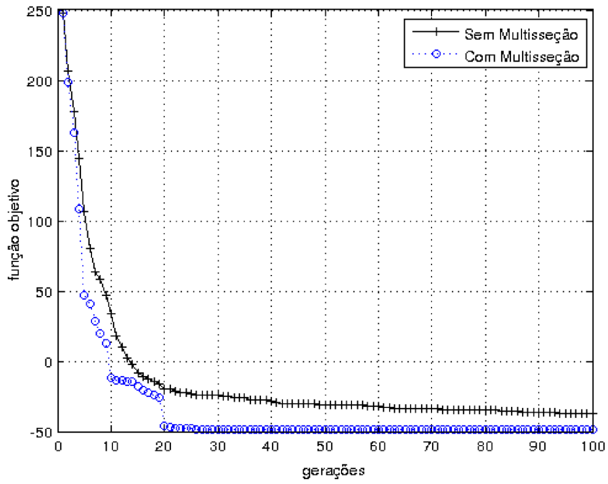
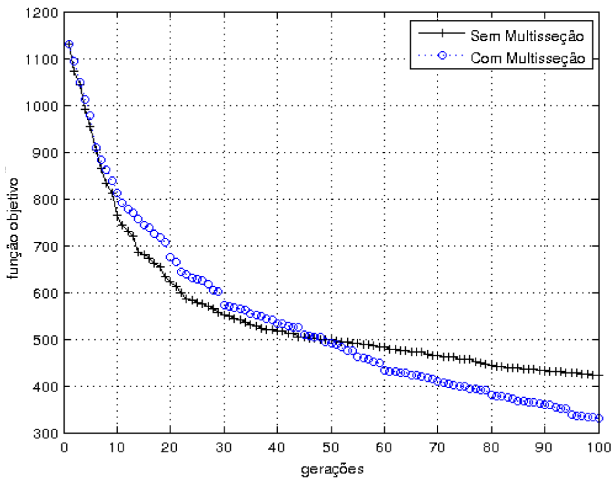


Figura 6: Curva de Convergência Média para o Problema 3



Como todo método de busca local, o tempo computacional gasto no método híbrido é superior ao tempo gasto no método puro. Neste caso, o cálculo das aproximações quadráticas impõe um custo computacional ao algoritmo. Entretanto, como mostrado em Wanner (2006), esse custo é desprezível quando a metodologia é aplicada a problemas cuja avaliação de funções é computacionalmente caro.

6. Conclusões

O presente trabalho apresentou uma metodologia de busca local para tratamento de problemas com duas restrições de igualdade. A metodologia utiliza aproximações quadráticas para as funções do problema de otimização. Esta metodologia obtém a solução do problema restrito via uma mecanismo baseado no método de bisseção e nas equações de Karush-Kuhn-Tucker.

O operador de busca local, denominado Multisseção, foi acoplado a um algoritmo genético com codificação real. Este algoritmo híbrido apresentou um bom desempenho quando aplicado a problemas com duas restrições não lineares de igualdade.

Uma vez que o operador, um método livre de derivadas que não exige nenhuma avaliação extra de função objetivo, pode ser acoplado a qualquer algoritmo evolutivo no tratamento de duas restrições de igualdade.

Para uma maior aplicabilidade do operador, a metodologia precisa ser ainda estendida para tratar problemas com qualquer número de restrições não lineares de igualdade.

7. Agradecimentos

Os autores gostariam de agradecer à FAPEMIG, à CAPES e ao CNPq.

Referências

Coello, C. A. C. (2002), Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: A survey of the state of the art, *Computer Methods in Applied Mechanics and Engineering*, vol 191, 1245-1287.

Peconick, G., Wanner, E. F., Takahashi, R. H. C. (2007), Projection-based local search operator for multiple equality constraints within genetic algorithms, *IEEE Congress on Evolutionary Computation*, Edinburgh: IEEE Press, 3043–3049.

Ramos, R.; Saldanha, R.; Takahashi, R.; Moreira, F. (2003), The real-biased multiobjective genetic algorithm and its application to the design of wire antennas, *IEEE Transactions on Magnetics*, v. 39, n. 3, p. 1329–1332.

Richardson, J.T., Palmer, M.R, Liepins, G., Hilliard, M. (1989), Some guidelines for genetic algorithm with penalty function, *Proceedings of the Third International Conference on Genetic Algorithms*, Morgan Kaufmann, p 191-197.

Sturm, J. F. (1999), Using sedumi 1.02, a matlab toolbox for optimization over symmetric cones, *Optimization Methods and Software*, v. 11, n. 1-4, p. 625–653.

Takahashi, R., Vasconcelos, J., Ramirez, J., Krahenbuhl, L (2003), A multiobjective methodology for evaluating genetic operators, *IEEE Transactions on Magnetics*, v. 39, n. 3, p. 1321–1324.

Wanner, E. F., Guimarães, F. G., Saldanha, R. R., Takahashi, R.H.C., Fleming, P.J. (2005), Constraint quadratic approximation operator for treating equality constraints with genetic algorithms, *IEEE Congress on Evolutionary Computation*, Edinburgh: IEEE Press.

Wanner, E. F. (2006), Operadores para algoritmos genéticos baseados em aproximações quadráticas de funções de variáveis contínuas, *Tese de Doutorado*, Universidade Federal de Minas Gerais, p.20.