APLICAÇÃO DA TÉCNICA DE CONDENSAÇÃO AO ALGORITMO DE OTIMIZAÇÃO GLOBAL

Rúbia Mara de Oliveira Santos
Universidade Federal de Mato Grosso do Sul
Instituto de Matemática - INMA
rubia.oliveira@ufms.br

Shih Ting Ju
Universidade Federal de Mato Grosso do Sul
Faculdade de Computação - FACOM
shih.marg@gmail.com

Wellington de Oliveira dos Santos
Universidade Federal de Mato Grosso do Sul
Faculdade de Computação - FACOM
wellington.oliveira@indext.com.br

RESUMO

A escolha dos pontos iniciais é um critério importante para a execução de algoritmos de otimização global. Este artigo tem por objetivo incorporar ao algoritmo Branch-and-Bound com busca local via técnica da Condensação um método para encontrar pontos iniciais com vistas à obtenção de soluções ótimas globais para os problemas geométricos signomiais. O desempenho computacional do algoritmo proposto foi avaliado por meio de problemas clássicos e as soluções ótimas obtidas foram comparadas com as soluções existentes na literatura.

PALAVRAS CHAVE. Programação Geométrica, Otimização Global, Programação Matemática.

ABSTRACT

The choice of starting points is an important criterion for the execution of global optimization algorithms. The purpose of this paper is to incorporate a method to find initial points to obtain global optimal solutions for signomial geometric problems to the Branch-and-Bound algorithm with local search via Condensation technique. The computational performance of the algorithm was evaluated using classical problems and the optimal solutions were compared to existing solutions found in the literature.

KEYWORDS. Geometric Programming, Global Optimization, Mathematical Programming.

Introdução

Otimização Global é uma especialização dentro da área de otimização dedicada à caracterização e à obtenção de mínimos (ou máximos) globais de problemas não-lineares. Soluções baseadas em otimização global são encontradas em diversos e importantes problemas práticos como: problemas de economia de escala, problemas de alocação e de localização, problemas de transporte e em diversos problemas de projeto presentes em Engenharia. Do ponto de vista computacional, problemas de otimização global pertencem à classe dos problemas NP-Completos [Vavasis 1995], que são problemas em que o tempo computacional cresce exponencialmente com o tamanho da entrada de dados para todo método conhecido.

Em [Maranas e Floudas 1997], os autores propõem um algoritmo que utiliza a técnica *Branch-and-Bound* para otimização global de problemas de otimização não-linear. Um aspecto importante neste trabalho é a definição de um excelente limitante inferior para a solução global do problema. No trabalho de [Oliveira e Ferreira 2005] foi incorporado ao algoritmo clássico proposto em [Maranas e Floudas 1997] um limitante superior obtido por meio da técnica de Condensação. A Condensação é usada como estratégia de busca local no algoritmo *Branch-and-Bound*, visando refinar o limitante superior, acelerando desta forma a convergência do algoritmo. Uma proposta futura apresentada por [Oliveira e Ferreira 2006] para o melhor desempenho do método proposto diz respeito à quantidade de pontos factíveis encontrados durante a execução do algoritmo. Este artigo tem por objetivo incorporar ao algoritmo o método proposto por [Ruckaert e Martens 1978] para encontrar pontos iniciais. A escolha dos pontos iniciais é um critério importante para a execução do algoritmo de otimização global. O ponto de partida influencia na divisão dos limitantes e pode alterar o resultado completamente. O algoritmo *Branch-and-Bound* com condensação foi aplicado aos problemas clássicos da literatura, considerados problemas difíceis de resolver, proposto por [Dembo 1976] e [Ruckaert e Martens 1978] e encontrou a solução ótima para todos os problemas.

O artigo está organizado da seguinte maneira: o problema de Programação Geométrica Posinomial e alguns conceitos básicos são descritos na Seção 1. Na Seção 2, os conceitos sobre Programação Geométrica Signomial são apresentados detalhadamente. O algoritmo *Branch- and- Bound* encontra-se na Seção 3. A técnica da Condensação é apresentada na Seção 4. Uma breve descrição do método para encontrar pontos iniciais encontra-se na Seção 5. Os resultados numéricos do algoritmo *Branch-and-Bound* com a busca local via condensação aplicado aos problemas clássicos da literatura encontram-se na Seção 6. Por fim, conclusões e considerações finais são apresentadas na Seção 6.

1. Programação Geométrica Posinomial

Programação Geométrica é uma técnica desenvolvida para resolver problemas algébricos de programação não-linear. Os algoritmos de programação geométrica têm sido constantemente melhorados e atualmente são ferramentas poderosas para resolver problemas importantes em Engenharia. A Programação Geométrica surgiu em 1961. O matemático Richard Duffin, comprometido com o desenvolvimento da teoria de dualidade, solidificou a proposta de [Zener 1961] aplicando-a em suas teorias. Veja [Beightler e Phillps 1976].

O desenvolvimento matemático da Programação Geométrica está baseado na relação de desigualdade entre somatórios e produtórios de números positivos. Este desenvolvimento teórico chamado Programação Geométrica é também conhecido como *Programação Posinomial*. Um problema de Programação Geométrica (PG) apresenta a seguinte forma geral:

$$
(\text{PG}) \left|
\begin{array}{ll}
\text{Minimizar} & f(x) \\
\text{s.a} & g_i(x) \le 1,\ i = 1,2,\ldots,m. \\
 & x_j > 0,\ j = 1,2,\ldots,n.
\end{array}
\right.
$$

Um posinômio $g_i(x)$ é uma função composta por monômios T_k do seguinte tipo:

$$T_k = \alpha_k \prod_{j=1}^{n} x_j^{a_{kj}}$$

no qual $\alpha_k > 0$ e a_{kj}, $j = 1,2,\ldots,n$ são números reais. Como $x > 0$, obtém-se $T_k > 0$. Assim, a função objetivo e as restrições podem ser escritas como:

$$f(x) := \sum_{k \in J_0} T_k,$$
$$g_i(x) := \sum_{k \in J_i} T_k, \; i = 1,2,\ldots,m,$$
$$J_0 \cup J_1 \cup \ldots \cup J_m = \{1,2,\ldots,M\}.$$

O conjunto J_0 descreve os termos (monômios) da função objetivo e os conjuntos J_i, com i representando uma restrição do problema, descrevem os termos de cada restrição. Esses conjuntos são mutualmente disjuntos; M representa o número total de termos, m o número total de restrições e n o número total de variáveis do problema. Os resultados, desenvolvimento teórico e conclusões sobre Programação Geométrica podem ser visto em detalhe em [Oliveira 2005].

2. Programação Geométrica Signomial

Um problema de *Programação Geométrica Signomial* (PGS) é caracterizado por uma função objetivo e/ou restrições que são a diferença de dois posinômios, ou seja, o problema geométrico signomial contém um ou mais termos com coeficientes negativos. Cada monômio do problema é um produto de variáveis positivas, sendo que cada uma delas pode estar elevada a uma potência real, multiplicado por uma constante α_k real:

$$T_k = \alpha_k \prod_{j=1}^{n} x_j^{a_{jk}}.$$

Assim, um posinômio $g_i(x)$ é um somatório de monômios $T_k(x)$. Agrupando-se os monômios de mesmo sinal, o problema (PGS) pode ser formulado como o seguinte problema de otimização nao-linear:

$$\text{(PGS)} \left| \begin{array}{ll} \text{minimizar} & g_0(x) = g_0^+(x) - g_0^-(x) \\ \text{s.a} & g_i(x) = g_i^+(x) - g_i^-(x) \le 0, \; i = 1,\ldots,m. \end{array} \right. \qquad (2)$$

sendo

$$g_i^+(x) = \sum_{k \in K_i^+} \alpha_{ik} \prod_{j=1}^{n} x_j^{a_{jki}},$$

$$g_i^-(x) = \sum_{k \in K_i^-} \alpha_{ik} \prod_{j=1}^{n} x_j^{a_{jki}}, \; i = 1,\ldots,m.$$

Em (2), $x = (x_1,\ldots,x_n)$ é o vetor variável de variáveis positivas; g_i^+ e g_i^-, para $i = 1,\ldots,m$ são funções posinomiais em x; a_{jki} são constantes reais arbitrárias e α_k são coeficientes positivos. Os conjuntos $k_i^+ \; e \; k_i^-$ indicam que monômios positivos e negativos formam os posinômios g_i^+ e g_i^-, respectivamente. Veja [Maranas e Floudas 1997].

2.1 Transformação em diferença de duas funções convexas

A função objetivo e as restrições da formulação original (PGS) são geralmente funções não-convexas. Aplicando-se em (2) a transformação $x_j = \exp(y_j)$, para $j = 1,\ldots,n,$ obtém-se o seguinte problema denotado por (DC), cuja função objetivo e restrições são a diferença de duas funções convexas.

$$(\mathrm{DC})\left|\begin{array}{ll} \text{minimizar} & G_0(y) = G_0^+(y) - G_0^-(y) \\ \text{s.a} & G_i(y) = G_i^+(y) - G_i^-(y) \le 0,\ i = 1,\ldots,m \\ & y_j^L \le y_j \le y_j^U,\ j = 1,\ldots,n. \end{array}\right.$$

sendo

$$G_i^+(y) = \sum_{k\in K_i^+} \alpha_{ik} \exp\left\{\sum_{j=1}^n a_{jki} y_j\right\},$$

$$G_i^-(y) = \sum_{k\in K_i^-} \alpha_{ik} \exp\left\{\sum_{j=1}^n a_{jki} y_j\right\},\ i = 1,\ldots,m.$$

Dado que $y_j^L = \ln(x_j^L)$, é necessário que $x_j^L > 0$, para $j = 1,\ldots,n.$

2.2 Limitante Inferior

Um limitante inferior para o problema (DC) pode ser obtido através da minimização de uma relaxação convexa. Tal relaxação convexa é conseguida subestimando cada função côncava $-G_i^-$, por uma função linear $-L_i^-$, para todo $i = 1,\ldots,m.$ Assim, obtém-se um problema Relaxado Convexo, denotado por (RC), que proporciona um limitante inferior para o problema (DC).

$$(\mathrm{RC})\left|\begin{array}{ll} \text{minimizar} & G_0^{Conv}(y) = G_0^+(y) - L_0^-(y) \\ \text{s.a} & G_i^{Conv}(y) = G_i^+(y) - L_i^-(y) \le 0,\ i = 1,\ldots,m. \\ & y_j^L \le y_j \le y_j^U,\ j = 1,\ldots,n. \end{array}\right.$$

sendo

$$G_i^+(y) = \sum_{k\in K_i^+} \alpha_{ik} \exp\left\{\sum_{j=1}^n a_{jki} y_j\right\};$$

$$L_i^-(y) = \sum_{k\in K_i^-} \alpha_{ik} \left\{A_{jk} + B_{jk}\left\{\sum_{i=1}^N a_{jki} y_i\right\}\right\};\ j = 1,\ldots,M;$$

$$A_{jk} = \frac{Y_{jk}^U \exp(Y_{jk}^L) - Y_{jk}^L \exp(Y_{jk}^U)}{Y_{jk}^U - Y_{jk}^L};$$

$$B_{jk} = \frac{\exp(Y_{jk}^U) - \exp(Y_{jk}^L)}{Y_{jk}^U - Y_{jk}^L};$$

$$Y_{jk}^L = \sum_{j=1}^n \min(a_{jki} y_j^L, a_{jki} y_j^U);$$

$$Y_{jk}^U = \sum_{j=1}^n \max(a_{jki} y_j^L, a_{jki} y_j^U).$$

Em [Oliveira 2005] encontram-se os detalhes sobre a obtenção das funções lineares $L_i^-(y)$. Observe que o limitante inferior de $-G_i^-, -L_i^-(y)$, é composto da soma de funções lineares. Claramente, quanto menor a diferença entre a função original $G_i^-(y)$ e a linearização $L_i^-(y)$ mais próxima à solução do problema (RC) estará da solução do problema (DC). A qualidade desses limitantes inferiores está relacionada à forma com que termos do tipo $-\exp(y)$ são aproximadas por funções lineares dentro do mesmo intervalo $[Y^L, Y^U]$.

3. Algoritmo *Branch-and-Bound* (BB)

O algoritmo *Branch-and-Bound* é uma técnica de otimização global que pode ser aplicada em problemas de complexidade NP-Completo, encontrando o mínimo global de uma função não-convexa f sobre um hiper-retângulo $Q_{ini} \subset R^n$. O hiper-retângulo é iterativamente dividido até que o ótimo global seja obtido com uma dada precisão $\varepsilon > 0$ Apesar de algumas vezes demandar um elevado esforço computacional, o algoritmo encontra a solução $\varepsilon -$ótima global em tempo finito.

O método *Branch-and-Bound* pode ser interpretado como uma maneira conveniente de gerar subdivisões no espaço de busca, utilizando limites superiores e inferiores para refinar progressivamente as áreas de interesse, evitando que todo o espaço de busca seja investigado. Este procedimento termina quando a diferença entre o limite superior e o inferior é menor que a tolerância ε.

No algoritmo BB aplica-se uma heurística para o particionamento do retângulo Q_{ini} que contribui para a redução do número de buscas necessárias para encontrar a solução do problema. Para um retângulo $Q \subset Q_{ini}$, define-se $\phi_{\min}(Q) = \min_{x \in Q} f(x)$. O algoritmo encontra $\phi_{\min}(Q_{ini})$ com precisão $\varepsilon > 0$, usando duas funções $\phi_{lb}(Q)\, e\, \phi_{ub}(Q)$ e definidas em $\{Q : Q \subset Q_{ini}\}$, tais que $\phi_{lb}(Q) \le \phi_{\min}(Q) \le \phi_{ub}(Q)$.

No processo de *ramificação* (participação de retângulos), o número de retângulos é igual ao número de iterações. Retângulos podem ser eliminados caso o ponto de mínimo não pertence ao retângulo considerado. O algoritmo proposto neste trabalho segue os passos do algoritmo descrito em [Maranas e Floudas 1997].

4. Busca Local Via Condensação

Um tipo particular de transformação baseado na desigualdade geométrica-aritmética é chamada de *Condensação*. Esta técnica foi proposta inicialmente por [Duffin *et al.* 1967]. O princípio básico da condensação é aproximar uma função posinomial multi-termos por uma função de um único termo, isto é, por uma função monomial. Define-se o problema geométrico condensado como um problema geométrico na qual todas as funções posinomiais tenham sido reduzidas a funções de um único termo. Veja [Beightler e Phillps 1976] para maiores detalhes.

Tomando-se uma desigualdade genérica na forma $G_i^+(x) - G_i^-(x) \le 0$, a desigualdade pode ser reescrita como $\dfrac{G_i^+(x)}{G_i^-(x)} \le 1$, na qual $G_i^+(x)$ e $G_i^-(x)$ são posinômios. O passo seguinte é condensar o posinômio $G_i^-(x)$ num monômio $G_i^-(x, \bar{x})$ no ponto $x = \bar{x}$, a partir da desigualdade da média geométrica-aritmética: se $x_j > 0,\ j = 1,\ldots,n,$ então

$$\sum_{j=1}^{n} w_j x_j \le \prod_{j=1}^{n} x_j^{w_j} \text{ , com } w \in W .$$

Observe que a divisão de $G_i^+(x)$ por $G_i^-(x, \bar{x})$ leva a um posinômio. Veja [Oliveira 2005] para maiores detalhes. Um procedimento sub-ótimo para resolução de problemas signomiais via condensação é resumido a seguir.

Algoritmo de Condensação

Passo 1. Reescrever o problema de programação geométrica signomial na forma de função objetivo linear, na forma de uma variável adicional x_0, e restrições não-lineares como razões de funções posinomiais: $\dfrac{G_i^+(x)}{G_i^-(x)} \leq 1, \quad i = 1,2,..m;$

Passo 2. Condensar cada função posinomial $G_i^-(x), \ i = 1,2,\ldots,m$ num monômio $G_i^-(x,\bar{x})$ (função condensada), num dado ponto factível $\bar{x}$;

Passo 3. Resolver o problema posinomial resultante através da técnica adequada [Oliveira 2005];

Passo 4. O algoritmo termina quando a solução corrente for igual ao ponto de condensação. Não havendo convergência, retorna-se ao **Passo 2**, com a solução atual como o novo ponto da condensação.

A técnica da condensação foi usada como técnica de busca local no algoritmo *BB* com a finalidade de refinar o limitante superior. O Passo 2 do algoritmo *BB* foi adaptado para explorar esta técnica. A ideia fundamental consiste em criar uma restrição convexa do problema original sempre que o ponto corrente for factível. Com isso, resolvendo-se o problema convexo implicado, a melhoria do limitante superior seria acelerada. Os detalhes envolvidos na adaptação da técnica de condensação no Passo 2 do algoritmo *BB* são apresentados em [Oliveira 2005]. Assim, por simplicidade, apresenta-se apenas o resultado final do Passo 2 a seguir.

Passo 2: Se $\max G_i^+(x) \leq \varepsilon_f$ para $i = 1,2,\ldots,m$ então, o conjunto de restrições é ε_f -factível em $y^{c,Iter}$. Resolve-se então, o Problema Condensado Convexo.

$$\begin{aligned} &\text{Minimizar} \quad y_{n+1} \\ &\text{s.a} \qquad \frac{\mathrm{G}_i^+(y)}{G_i^-(y, y^{c,Iter})} \leq 0 \quad i = 1,2,\ldots,m \end{aligned}$$

com

$$\frac{\mathrm{G}_i^+(y)}{G_i^-(y, y^{c,Iter})} = \sum_{k \in K_i^+} \frac{\alpha_{ik}}{d_i} \exp\left\{\sum_{j=1}^{n+1} \beta_{ji} y_j^{c,Iter}\right\} \exp\left\{\sum_{j=1}^{n+1} a_{jki} - \beta_{ji} y_j\right\} \text{ e}$$

$$\beta_{ji} = \frac{\sum_{k \in K_i^-} \alpha_{ik} \prod_{j=1}^{n+1} (x_j^{c,Iter})^{a_{jki}}}{G_i^-(x^{c,Iter})} \qquad \text{e} \qquad d_i = \sum_{k \in K_i^-} \alpha_{ik} \prod_{j=1}^{n+1} (x_j^{c,Iter})^{a_{jki}}$$

Com a solução $y_j^*, \ j = 1,2,\ldots,n,$ atualiza-se o limitante superior:

$$G_0^{UBD} = \min(G_0^{UBD}, G_0(y^*)) .$$

5. Desenvolvimento do método

Uma proposta futura apresentada por [Oliveira e Ferreira 2006] para o melhor desempenho do método proposto diz respeito à quantidade de pontos factíveis encontrados durante a execução do algoritmo. Neste trabalho foi implementado, na linguagem C, um método para encontrar a solução inicial. A escolha dos pontos iniciais é um critério importante para a execução do algoritmo de otimização global BB. O ponto de partida influencia na divisão dos limitantes e pode alterar o resultado completamente. Para encontrar pontos iniciais, foi incorporado ao algoritmo o método proposto por [Ruckaert e Martens 1978]. Foi utilizado o método de geração de pontos baseado na solução ótima e nos limitantes superiores e inferiores de cada problema. Cria-se uma esfera com centro na solução ótima, abrangendo 90% do espaço de busca; este espaço interno à esfera é utilizado para a escolha do ponto inicial de acordo com uma entrada aleatória em uma matriz pré-definida.

Cálculo de Pontos Iniciais

---

a = 0.1 × valor ótimo da variável,
b = 1.9 × valor ótimo da variável,
l.b = limitante inferior da variável,
u.b = limitante superior da variável,
k = número da variável,
n = Matriz [k % 7] [k % 10],
u = a + (b – a) × n,
Cálculo do ponto inicial:
    Se u < l.b., então valor inicial da variável = l.b.,
    Se l.b. ≤ u ≤ u.b. então valor inicial of variável = u,
    Se u.b. < u então valor inicial of variável = u.b.

Os números aleatórios são gerados com base na Tabela 1, apresentada em [Ruckaert e Martens 1978]. O número aleatório é gerado pela posição da matriz [k % 7] [k % 10], em que % é o operador de resto da divisão; e k é o número da variável x_k. Esses números são usados para gerar diferentes pontos iniciais.

Tabela 1: Tabela de números aleatórios.

0.53	0.47	0.98	0.11	0.15	0.98	0.03	0.61	0.22	0.17
0.59	0.52	0.64	0.02	0.38	0.4	0.57	0.73	0.93	0.51
0.43	0.21	0.16	0.92	0.55	0.97	0.34	0.47	0.03	0.28
0.58	0.11	0.69	0.19	0.64	0.26	0.24	0.04	0.46	0.43
0.83	0.28	0.79	0.73	0.91	0.92	0.82	0.37	0.75	0.78
0.66	0.02	0.33	0.82	0.77	0.74	0.52	0.37	0.11	0.18
0.84	0.89	0.21	0.39	0.56	0.98	0.89	0.99	0.23	0.15

6. Experimentos e Resultados

Foram testados todos os problemas clássicos propostos em [Dembo 1976] e [Ruckaert e Martens 1978]. Mostrando-se eficiente em 95% dos problemas testados. Nas Tabelas 2 e 3 encontram-se os resultados após a execução do programa, comparando a solução obtida pelo algoritmo de otimização global BB com a busca local utilizando a Técnica da Condensação, com a solução ótima encontrada respectivamente pelos autores. O algoritmo foi executado em um computador com a seguinte configuração: Intel Core i7 1.66GHZ, 4 cores, 8 threads, 8GB de memória RAM. A primeira coluna refere-se ao número do problema, a segunda as soluções ótimas encontradas das variáveis do artigo, a terceira representando o número de iterações necessárias, a quarta representam as soluções encontradas pelo algoritmo de otimização global BB com busca local via Técnica Condensação e na quinta o tempo de execução em segundos.

Tabela 2. Comparação com os problemas clássicos propostos em [Dembo 1976].

Ex	Solução Dembo (1978)	It.	Solução BB com Condensação	Tempo (s)
1	t1=2.5229712E-6 t2=2.5288262E-5 t3=7.6566135E-6 t4=1.1853913E-9 t5=7.6971796E-9 t6=1.2919459E-3 t7=4.2615451E-3 t8=2.7863431E-3 t9=1.7093851E-5 t10=2.038954E-4 t11=6.628593E-4 t12=6.558187E-1 g0=4.890462E+9	40	t1=0.0000019 t2=0.0001439 t3=0.0000054 t4=0.0000000 t5=0.0000000 t6=0.0018245 t7=0.0033213 t8=0.0024768 t9=0.0000266 t10=0.0002436 t11=0.0004352 t12=0.9199106 g0=4.8983900E+9	0.332536
2	t1=78.0000 t2=33.0000 t3=29.99551065 t4=45.000000 t5=36.77517397 g0=−30665.49848	2	t1=78.0000005 t2=33.0000000 t3=29.9955194 t4=44.9999366 t5=36.7751831 g0=-30669.70704	0.0081
3	t1=1698.5276698 t2=53.5257212 t3=3031.5798057 t4=90.0909228 t5=95.000000 t6=10.5192394 t7=153.5353546 **g**0=1227.1831610	1	t1=1698.2321010 t2=53.6442191 t3=3031.3378083 t4=90.1068905 t5=95.0000000 t6=10.5025347 t7=153.5353545 g0=1227.2262351	0.009645
4ª	t1=6.3450905 t2=2.3427973 t3=0.6701581 t4=0.5966619 t5=5.9528907 t6=5.5291597 t7=1.0441714 t8=0.4036023 g0=3.9516982	1	t1=6.4720451 t2=2.2289267 t3=0.6689455 t4=0.5959830 t5=5.9315269 t6=5.5266183 t7=1.0115313 t8=0.3997444 g0=3.9511709	0.006793
4b	t1=6.1016207 t2=2.5741491 t3=0.6765178 t4=0.5959532 t5=5.9935248 t6=5.5315385 t7=1.1061818 t8=0.4071500 g0=3.9561968	1	t1=6.4720451 t2=2.2289267 t3=0.6689455 t4=0.5959830 t5=5.9315269 t6=5.5266183 t7=1.0115313 t8=0.3997444 g0=3.9511709	0.008235
5	t1=572.852745 t2=1361.497867 t3=5114.973425 t4=181.476181 t5=295.402778 t6=218.524281 t7=286.075426 t8=395.402602 g0=7049.324305	40	t1=578.9143317 t2=1361.7830016 t3=5108.5536923 t4=181.9849744 t5=295.6580841 t6=218.0154217 t7=286.3267569 t8=395.6580821 g0=7049.2510256	0.153376
6	t1=0.803773 t2=0.900000 t3=0.900000 t4=0.100000 t5=0.190837 t6=0.900000 t7=574.099615 t8=74.099636 t9=500.00000 t10=0.10000 t11=20.239117 t12=77.336450 t13=0.015467 g0=97.591034	40	t1=0.8037746 t2=0.8999866 t3=0.9950898 t4=0.1000000 t5=0.1908111 t6=0.8415869 t7=574.0791468 t8=74.0762970 t9=500.1616744 t10=0.1000000 t11=20.2324621 t12=77.6591761 t13=0.0069477 g0=97.8985859	0.607485
7	t1 =0.8037724 t2=0.8175130 t3=0.90000 t4=0.90000 t5=0.90000 t6=0.0999996 t7=0.1078842 t8=0.1908369 t9=0.1908369 t10=0.1908369 t11=505.664787 t12=5.6650580 t13=72.475185 t14=500.0 t15=500.000 t16=0.000001 g0=174.788807	1	t1=0.8030778 t2=0.8150248 t3=0.8999931 t4=0.9000000 t5=0.9001383 t6=0.0996264 t7=0.1063804 t8=0.1910004 t9=0.1910020 t10=0.1910664 t11=504.7229649 t12=4.7236069 t13=73.0756786 t14=500.1459123 t15=500.0943502 t16=0.0011331 g0=174.9549490	0.077101
8	t1=2.8566276 t2=0.61083257 t3=2.1503944 t4=4.7171337 t5=1.0002048 t6=1.3487370 t7=0.03160686 g0=1809.7615	40	t1=2.6847316 t2=0.5920301 t3=2.1316505 t4=4.1942178 t5=0.9461566 t6=1.2346387 t7=0.0330079 g0=1824.7007654	0.145035
8b	t1=3.8955214 t2=0.80868472 t3=2.6626285 t4=4.2983005 t5=0.85357785 t6=1.0953123 t7=0.02730898 g0=911.87957	40	t1=3.8770754 t2=0.8262856 t3=2.7118740 t4=3.9299748 t5=0.8002605 t6=1.0000000 t7=0.0277816 g0=918.6015563	0.136676

Tabela 3. Comparação com os problemas clássicos propostos em [Ruckaert e Martens 1978]

Ex	**Solução** Rijckaert e Martens 1978	It.	**Solução BB com Condensação**	**Tempo (s)**
1	t1=82.847 t2=87.924 t3=8.293 t4=1.364 g0=0.01208	1	t1=83.0000000 t2=89.0000000 t3=9.0000000 t4=0.1311430 g0=0.0120482	0.004789
2	t1=107.4 t2=84.9 t3=204.5 g0=6300	1	t1=108.7347038 t2=85.1262133 t3=204.3245959 g0=6299.8424279	0.003464
3	t1=749.89 t2=0.11114 t3=1.46193 t4=3.42481 g0=126344	1	t1=749.7119770 t2=0.1084133 t3=1.4609369 t4=3.4248955 g0=126303.359455	0.004466
4	t1=2.5180708 t2=2.5100911 t3=7.6625640 t4=1.1741216 t5=7.7709114 t6=1.3019503 t7=4.2801718 t8=2.7894894 t9=1.7389956 t10=1.973219 t11=6.638984 g0=3.168198	40	t1=0.4395445 t2=0.6762935 t3=9.2200101 t4=5.7433037 t5=1.5915477 t6=0.6324555 t7=3.1366042 t8=1.8860753 t9=1.0000000 t10=3.1622777 t11=5.6234133 g0=3.1962545	0.577111
5	t1=43.02 t2=44.85 t3=66.39 t4=1.11 g0=623015	40	t1=43.3787246 t2=45.8257569 t3=65.8611017 t4=1.3989797 g0=631140.54107	0.171006
6	t1=0.9701 t2=0.1985 t3=1.1216 t4=0.7841 t5=1.0040 t6=0.6948 t7=1.1157 t8=0.9993 g0=29.5985	14	t1=0.9688752 t2=0.1989576 t3=1.1212671 t4=0.7844100 t5=1.0022437 t6=0.7009703 t7=1.0940096 t8=0.9723097 g0=29.2294875	0.054293
7	t1=0.96856 t2=0.19355 t3=1.1332 t4=0.78624 t5=1.0001 t6=0.69479 t7=1.1163 t8=0.99689 g0=29.595	1	t1=0.9669249 t2=0.1997328 t3=1.1207749 t4=0.7830019 t5=1.0097518 t6=0.7022421 t7=1.0966427 t8=0.9750143 g0=29.2264580	0.012789
8	t1=1.34186 t2=0.99325 t3=0.87050 t4=0.92359 t5=3.14643 t6=0.40408 t7=1.54767 g0=178.478	40	t1=1.3435377 t2=0.9934900 t3=0.8707324 t4=0.9233628 t5=3.1508296 t6=0.4043360 t7=1.5408670 g0=178.4782851	0.490142
9	t1=0.819 t2=446 g0=11.91	1	t1=0.8112874 t2=442.6648656 g0=11.9643371	0.001936
10	t1=88.310 t2=7.454 t3=1.311 g0=-83.21	1	t1=88.3559877 t2=7.6725429 t3=1.3179237 g0=-83.2497284	0.002385
11	t1=8.1301 t2=0.6154 t3=0.5640 t4=5.6362 g0=-5.7398	1	t1=8.1300102 t2=0.6153483 t3=0.5640613 t4=5.6362199 g0=-5.7398203	0.005787
12	t1=6.4650 t2=0.6674 t3=1.0130 t4=5.9327 t5=2.2326 t6=0.5958 t7=0.4006 t8=5.5273 g0=-6.0482	1	t1=6.4464603 t2=0.6689276 t3=1.0177822 t4=5.9358387 t5=2.2480932 t6=0.5969044 t7=0.4015006 t8=5.5278734 g0=-6.0482393	0.013222
13	t1=579.307 t2=1359.97 t3=5109.97 t4=182.018 t5=295.601 t6=217.982 t7=286.416 t8=395.601 g0=7049.247	40	t1=578.9862234 t2=1359.7423779 t3=5110.5193819 t4=181.9909514 t5=295.5792261 t6=218.0090532 t7=286.4117260 t8=395.5792260 g0=7049.2479832	0.162727
14	t1=2.0953 t2=12.0953 t3=7.9047 t4=0.4594 t5=0.3579 t6=0.4548 t7=10.4548 t8=1.6405 t9=1.1975 t10=0.1000 g0=1.1436	40	t1=2.0832042 t2=12.0831948 t3=7.9168107 t4=0.4666959 t5=0.3516148 t6=0.4539132 t7=10.4539127 t8=1.6292930 t9=1.1909678 t10=0.1000000 g0=1.1436591	0.264206

15	t1=0.7240 t2=0.7240 t3=0.7240 t4=0.2576 t5=0.1771 t6=0.1218 t7=0.2053 t8=0.1411 t9=0.0970 t10=0.2985 g0=0.2015	1	t1=0.7259852 t2=0.7190747 t3=0.7244108 t4=0.2574511 t5=0.1774036 t6=0.1219631 t7=0.2051396 t8=0.1413569 t9=0.0971814 t10=0.2985905 g0=0.2056549	0.023972
16	t1=0.7295 t2=0.7133 t3=0.7030 t4=0.2653 t5=0.1821 t6=0.1241 t7=0.1979 t8=0.1329 t9=0.0893 t10=0.2947 g0=0.1966	40	t1=0.9235839 t2=0.9357974 t3=0.2374814 t4=0.2474391 t5=0.1539429 t6=0.1328563 t7=0.1836724 t8=0.1116473 t9=0.0958722 t10=0.2979358 g0=0.2007153	0.320675
17	t1=7.004 t2=7.646 t3=7.112 t4=0.0125 t5=0.8120 t6=0.9558 t7=0.382 t8=0.358 t9=0.353 t10=2.077 t11=0.453 g0=0.1406	40	t1=7.0037206 t2=7.6474292 t3=7.1120387 t4=0.0124632 t5=0.8118778 t6=0.9558132 t7=0.3816059 t8=0.3578206 t9=0.3535374 t10=2.0763874 t11=0.4535374 g0=0.1406067	0.437561
18	t1=0.3917 t2=0.0937 t3=0.7435E-5 t4=0.4809 t5=0.6431 t6=0.0251 t7=0.0066 t8=0.0220 t9=0.5500 t10=1.3823 t11=23831 t12=0.1234 t13=0.3348 g0=1.81830	40	t1=0.1446269 t2=0.1219378 t3=0.0000059 t4=0.5463723 t5=0.5192214 t6=0.0202928 t7=0.0107414 t8=0.0094957 t9=0.4356398 t10=5.0000000 t11=0.1226784 t12=0.1828325 t13=0.0153567 g0=2.2954745	0.60391
19	t1=5153.58 t2=6.64945 t3=169413 t4=743.39 t5=87998.7 t6=187.542 t7=0.124094 t8=29.2653 g0=17486	1	t1=5151.4887492 t2=6.6481417 t3=169007.9414960 t4=744.5843973 t5=88106.9042930 t6=187.6290469 t7=0.1242030 t8=29.3416889 g0=17485.9867055	0.02525
20	t1=13505 t2=36312 t3=3212 t4=110.78 t5=370411 t6=30.75 t7=1.12E-8 t8=47399 t9=147039 t10=7793 t_{11}=19322 t_{12}=362436 t13=14559 g0=-121.54	40	t1=13305.7054249 t2=36307.4225153 t3=3498.2253616 t4=129.3631175 t5=381044.6608593 t6=29.0390066 t7=0.0000000 t8=45000.8802953 t9=153274.816719 t10=7092.4183265 t11=19982.311488 t12=372783.43744 t13=15219.312593 g0=-113.7245075	0.619345
21	t1=1766 t2=18664 t3=95.12 t4=3087 t5=2000 t6=91.50 t7=94.83 t8=11.70 t9=2.19 t10=151.48 g0=-1237.55	1	t1=1770.1128938 t2=18832.0773185 t3=93.3697542 t4=3090.2591442 t5=2000.0000000 t6=91.8075234 t7=94.9434246 t8=11.7687841 t9=2.0800994 t10=151.9823794 g0=-1269.9972924	0.019282
22	t1=11.7446 t2=0.36756 t3=0.3474 t4=12.277 t5=14.166 t6=0.4 t7=38.2906 t8=0.73187 t9=0.131110 g0=-375.784	40	t1=10.9735462 t2=0.2912689 t3=0.0000100 t4=12.2734019 t5=9.0209041 t6=0.4000000 t7=33.3661958 t8=0.7190527 t9=0.7190497 g0=-279.0746582	0.621935
23	t1=78 t2=33 t3=29.998 t4=45 t5=36.7673 g0=10127.13	40	t1=78.0000000 t2=33.0400051 t3=30.6779229 t4=45.0000000 t5=35.0886886 g0=10234.3125824	0.288478
24	t1=0.8037 t2=0.9000 t3=0.9000 t4=0.1000 t5=0.1908 t6=0.1908 t7=574.099 t8=74.099 t9=500 t10=0.1 g0=97.591034	1	t1=0.8028064 t2=0.8998842 t3=0.9894390 t4=0.0994858 t5=0.1906441 t6=0.7070297 t7=571.2024922 t8=74.0345825 t9=500.2987968 t10=0.6479644 g0=97.7923507	0.022071

6. Conclusão

Neste artigo discutiu-se a possibilidade de melhoria de um algoritmo do tipo *Branch-and-Bound* voltado para problemas geométricos signomiais. A tentativa de melhoria consistiu em incorporar, ao algoritmo BB, um procedimento de busca local baseado no conceito de Condensação. O algoritmo foi implementado na Linguagem C, aplicado aos problemas clássicos da literatura [Dembo 1976] e [Ruckaert e Martens 1978] e apresentou excelente desempenho nos problemas testados. Trabalhos futuros pretende-se substituir a biblioteca de otimização NLOP, utilizada neste artigo, por uma biblioteca mais eficiente como a MINOS 5.5, um pacote especializado para programação não linear, eficiente, porém não é gratuita.

Agradecimentos

Os autores agradecem a FUNDECT "Fundação de Apoio ao Desenvolvimento do Ensino, Ciência e Tecnologia do Estado de Mato Grosso do Sul" e ao CNPq "Conselho Nacional de Desenvolvimento Científico e Tecnológico" pelo apoio financeiro.

REFERÊNCIAS BIBLIOGRÁFICAS

Beightler, C. e Phillps, D. T. (1976). *Applied Geometric Programming*. John Wiley & Sons.

Duffin, R. J., Peterson, E. L. e Zener, C. (1967). *Geometric Programming*. Wiley, New York.

Dembo, R. S. A. (1976). A set of Geometric Programming Test Problems and Their Solutions. University of Waterloo, Waterloo. Canadá.

Floudas, C. A., e Visweswaran, V. (1993). Primal – Relaxed Dual Global Optimization Approach. *Journal of Optimization Theory and Applications*, **78**:187-225.

Kachian, L. G. (1979). A Polynomial Algorithm in Linear Programming. *Soviet Mathematica Doklady.*, **20:** 191-194.

Maranas, C.D. e Floudas, C.A. (1997). Global Optimization in Generalized Geometric Programming. Computers e Chemical. **21**: 351-367.

Oliveira, R. M. (2005) *Algoritmos de Busca Global para Problemas Geométricos e Multiplicativos*. Tese de Doutorado da FEEC- UNICAMP.

Oliveira, R. M. e Ferreira, P. A. V. (2006). Um Algoritmo Branch and Bound com Buscal Local para Programação Geométrica Signomial. *In Anais do XXXVIII Simpósio Brasileiro de Pesquisa Operacional*, Goiânia-Brasil.

Pardalos, P. M., e Thoai, N. V. (1995). *Introduction to Global Optimization*. Kluwer Publishers.

Ruckaert. M. J. e Martens, X. M. (1978). Comparison of Generalized Programming Algorithms. *Journal of Optimization Theory and Applications*, **26** (2), 205-242.

Vavasis, S. A. (1995). *Complexity Issues in Global Optimization: A Survey*, Handbook of Global Optimization Problems, 27-40. Kluwer Academic Publishers, Netherlands.

Zener, C. (1961). A Mathematical aid in Optimizing Engineering Desing. *Proc. Natl. Acad. Sci.*, **47** (4), 537-539.