

IMPLEMENTAÇÃO DE UM ALGORITMO HÍBRIDO PARA OTIMIZAÇÃO DE FUNÇÕES BIDIMENSIONAIS

IMPLEMENTATION OF A HYBRID ALGORITHM FOR OPTIMIZATION OF BIDIMENSIONAL FUNCTIONS

Fernando S. Freitas^a, Gisely L. Ströher^a, Gylles R. Ströher^a

^a*Departamento de Engenharia Química, Universidade Tecnológica Federal do Paraná, Campus Apucarana, Brasil, fernandofreitas@alunos.utfpr.edu.br, gisely@utfpr.edu.br, gyllesrs@utfpr.edu.br*

Palavras-chave: Algoritmo híbrido, Otimização global.

Resumo. O presente trabalho tem como objetivo propor um método híbrido para otimização de funções bidimensionais a partir do sequenciamento de dois tipos de metodologias, não determinísticas e subsequentemente determinística. Na primeira fase o método proposto utiliza Algoritmo Genético e na segunda fase o algoritmo Simplex de *Nelder Mead*. O Algoritmo Genético provê uma área promissora em que se encontra o mínimo global e na sequência o Simplex é iniciado a fim de identificar o ponto exato, ou o mais preciso possível, em que o mínimo global se encontra. O algoritmo híbrido proposto foi extensivamente testado usando funções *Benchmark*. Os testes foram realizados variando diversos parâmetros do AG com o objetivo de identificar qual combinação de parâmetros provê os melhores resultados de otimização global em um menor tempo de processamento. Por fim, comparou-se os resultados obtidos neste trabalho com o de outros algoritmos propostos por outros autores disponíveis na literatura.

Keywords: Hybrid algorithm, Global optimization.

Abstract. The present work aims to propose a hybrid method for the optimization of two-dimensional functions from the sequencing of two types of methodologies, nondeterministic and subsequently deterministic. The first time the option of using the Genetic Algorithm and the second phase of the algorithm Simplex Nelder Mead. The Genetic Algorithm provides an area that is at least in the sequence Simplex is a process of identifying the exact point, or the most accurate, in which the global minimum is found. The hybrid algorithm was extensively tested using Benchmark functions. The tests were performed by varying the GA parameter with the set of qualification parameters provided the results of the overall optimization process in a shorter processing time. Finally, it was compared with the results of the work of other algorithms proposed by other authors available in the literature.

1 INTRODUÇÃO

Otimizar globalmente uma função em um determinado domínio de pesquisa consiste em encontrar seus mínimos globais sem ficar preso a um de seus mínimos locais (Chelouah, 2003). Em diversas áreas, como na engenharia e na ciência, existem inúmeros problemas que podem ser considerados como de otimização global. Para uma grande parte desses problemas, têm sido estudados algoritmos probabilísticos, heurísticas, meta-heurísticas, dentre outros, devido à dificuldade de solucionar tais problemas de maneira computacionalmente eficiente para instâncias relativamente grandes (Melo, 2009).

Métodos baseados em computação evolutiva constituem uma classe de algoritmos de busca e otimização estocástica inspirados na teoria da evolução natural de Darwin. Estes algoritmos têm recebido atenção especial nos últimos anos por se tratarem de métodos robustos, capazes de fornecer soluções de alta qualidade para problemas considerados intratáveis por métodos tradicionais de otimização, os quais foram concebidos para problemas lineares, contínuos e diferenciáveis (Vianna, 2009). Conforme comenta Schwefel (1994), o mundo real é não-linear e dinâmico, cheio de fenômenos como descontinuidade, instabilidade estrutural e formas geométricas fractais. Em problemas em que é necessário levar em conta tais fenômenos, os métodos tradicionais certamente não apresentarão desempenho satisfatório, assim, os métodos evolutivos são uma alternativa para tentar superar as limitações apresentadas por métodos tradicionais, embora não garantam a obtenção da solução exata (Vianna, 2009).

Um das principais abordagens para sistemas baseados na evolução natural são os algoritmos genéticos, ou AG. Os AGs foram introduzidos por Holland (1992) com o objetivo de formalizar matematicamente e explicar processos de adaptação em sistemas naturais e desenvolver sistemas artificiais (simulados em computador) que retenham os mecanismos originais encontrados em sistemas naturais. Os algoritmos genéticos têm sido intensamente aplicados em problemas de otimização (Drummond et al., 2001), apesar de não ter sido este o propósito original que levou ao seu desenvolvimento.

Os Algoritmos Genéticos são uma classe de algoritmos que utilizam técnicas inspiradas na biologia evolutiva como hereditariedade, mutação, seleção e recombinação. AGs se diferenciam de métodos tradicionais de otimização por se basearem em uma população de possíveis soluções e não apenas na otimização de parâmetros do problema. Além disso, AGs podem ser implementados sem a necessidade de conhecimento prévio do problema. Necessita-se apenas de uma forma adequada de representação e avaliação das possíveis soluções do problema a ser tratado (Lima, 2008).

A lenta convergência de AGs antes de fornecer uma solução precisa é um inconveniente bem conhecido. Esta limitação, muitas vezes impede que os AGs sejam realmente de interesse prático para muitas aplicações. Para contornar tal limitação, vários métodos híbridos têm sido propostos na literatura, combinando um AG com algum algoritmo clássico de *hill-climbing*, basicamente o papel dedicado ao AG ainda é localizar alguma região promissora dentro do amplo espaço de solução, enquanto o algoritmo de otimização clássico permite alcançar, com precisão e rapidez, a melhor solução situada naquela região (Chelouah, 2003).

Segundo Chelouah, a maioria dos algoritmos de otimização local existentes podem ser escolhida para cooperar dessa forma com AG, no presente trabalho propõem-se um método híbrido combinando o AG com o Algoritmo Simplex de Nelder-Mead. O algoritmo híbrido proposto visa determinar o mínimo global de funções bidimensionais multimodais na qual o AG tem a função de encontrar uma área promissora e o Simplex na sequência visa a determinar o ponto exato, ou o mais próximo possível em que o mínimo se encontra.

2 METODOLOGIA

2.1 Descrição do método híbrido de otimização

Se acrônimos forem usados, você deve defini-los antes de sua primeira ocorrência.

O algoritmo necessita de uma população inicial de indivíduos na qual contém coordenadas reais associadas a função objetivo. Essa população inicial passa por operadores inspirados na genética de reprodução que simulam acontecimentos biológicos reais aos seres vivos. Descendentes são gerados a partir dos pais, criando assim uma nova população promissora, que é escolhida na seleção dos melhores indivíduos. O mecanismo de reprodução usado foi o crossover, que consiste em reordenar coordenadas de dois indivíduos e mutação que substitui uma coordenada de um indivíduo. Após o AG definir uma área promissora onde se encontra o mínimo global, o Simplex é iniciado afim de identificar o ponto mínimo no qual ele não encontraria sem o auxílio do AG.

As principais características dos métodos utilizados no AG, assim como alguns parâmetros foram os seguintes: geração da população inicial, operadores de seleção que basicamente seleciona os indivíduos com melhores avaliações, operadores de recombinação (*crossover*) que gera descendentes a partir dos melhores indivíduos escolhidos, operador de mutação que agrega diversidade a população e elitismo onde garante que os melhores indivíduos de cada geração não morram.

O algoritmo Simplex pertence a uma classe mais geral de algoritmos de pesquisa direta e que não faz uso de derivadas, portanto trata-se de uma técnica de otimização não linear, baseada no método de otimização livre de derivada, *Derivative-Free Optimization* (DFO). O Simplex busca, a partir de uma primeira solução básica viável, percorrer de forma iterativa os vértices de um polígono até alcançar uma solução considerada ótima para o problema. Constitui-se de um método que minimiza uma função de M variáveis que, por sua vez, depende da comparação da função nos seus $(M + 1)$ vértices de um simplex geral seguido pela recolocação do vértice com o valor mais alto por outro ponto de menor valor na função objetivo. De um ponto de vista computacional, esse método mostra-se ser compacto e efetivo (Gomes, 2017).

2.2 Critério de parada

O AG funciona basicamente por meio de iterações onde a sistemática pode ser visualizada na Figura 1. Em resumo, o algoritmo gera uma população inicial de indivíduos que contemple todo o domínio de busca estipulado inicialmente, na sequência é calculado os respectivos valores da função objetivo para cada indivíduo. A nova população é gerada a partir dessa por meio das etapas genéticas que já foram descritas: seleção, recombinação e mutação. A nova população criada a partir da antiga passa para a próxima geração onde o processo é reiterado. O processo iterativo só é interrompido quando atingir o número máximo de gerações definido inicialmente ou uma precisão relacionada as coordenadas dos indivíduos, ρ_{abs} , dada pela Equação (1).

$$\frac{1}{N_{ind}} \sum_i^{N_{ind}} \|Ind_i - MelhorInd\| \leq \rho_{abs} \quad (1)$$

Para exemplificar, é calculado a distância entre um os indivíduos da geração atual, *ind*, com o melhor ponto encontrado até o momento, *MelhorInd*, sendo que, a iteração é interrompida caso esse valor dê menor que ρ_{abs} , sendo também definido inicialmente.

Assim que um dos critérios de parada do AG é atingido o algoritmo Simplex é então acionado. O Simplex recebe a coordenada do ponto mínimo a partir do melhor indivíduo gerado pelo AG, e esse ponto mínimo passa a ser solução inicial, ou seja, um dos vértices da forma geométrica, em geral ortogonal, do Simplex.

A iteração realizada pelo Simplex já descrita acima só é interrompida quando pelo menos um dos dois critérios de parada são atingidos. O primeiro critério de parada é satisfeito quando o algoritmo atingir o número de iterações máximas estipulado inicialmente, k , o que em geral acaba não acontecendo. O outro critério de parada, normalmente o que é satisfeito primeiro, está relacionado ao cálculo do desvio padrão dos vértices $n + 1$ de um Simplex avaliados na função objetivo. As interações são interrompidas se o cálculo do desvio padrão, Equação 2, for menor que um erro pré-definido, Er . O próprio simplex adapta-se à paisagem local e contrai-se a um mínimo local final.

$$\sqrt{\frac{\sum_{i=1}^{n+1} (f_i - \bar{f})^2}{n+1}} \leq Er \quad (2)$$

Em que f_i é o valor dos $n + 1$ vértices de um Simplex avaliados na função objetivo, n é o número de variáveis da função objetivo e \bar{f} é a média aritmética dos vértices.

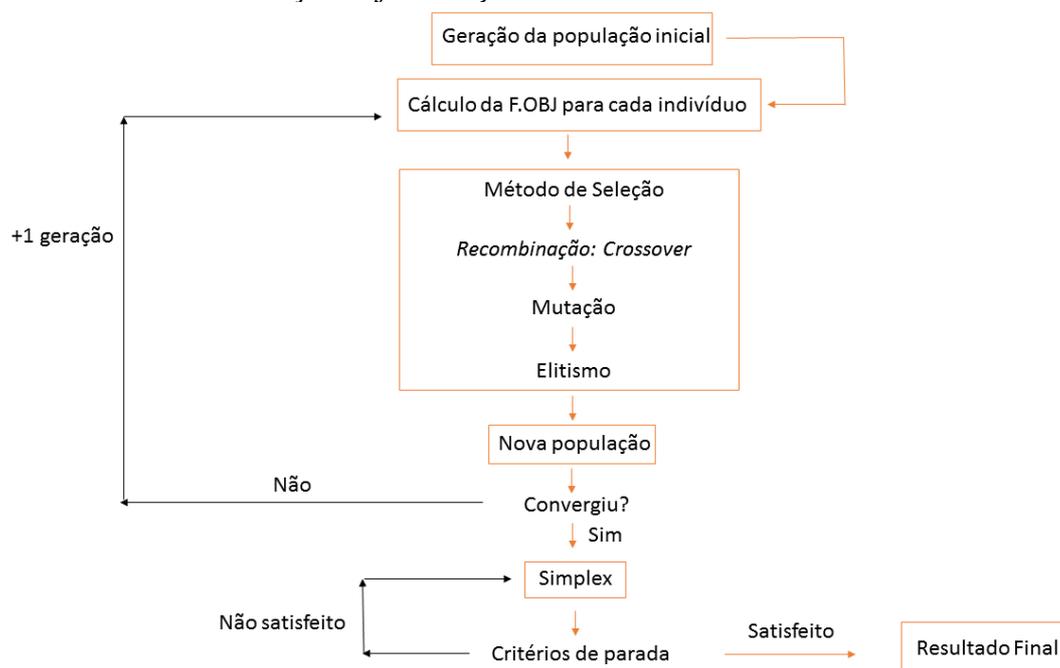


Figura 1 – Pseudocódigo do AH

3 RESULTADOS E DISCUSSÕES

A fim de determinar a eficiência do algoritmo híbrido, várias funções bidimensionais *Benchmark* foram testadas. Uma vez que o algoritmo genético não é método determinístico, cada vez que o programa for executado um novo histórico de gerações pode ser obtido, assim seguindo a recomendação de *Chelouah; Siarry, 2003*, cada função foi testada cem vezes. Os testes foram realizados variando diversos parâmetros do AG com o objetivo de identificar qual combinação de parâmetros provê os melhores resultados de otimização global em um menor tempo de processamento.

Foram utilizadas seis funções *Benchmark* variando os seguintes operadores genéticos: *crossover*, mutação e método de seleção. Para avaliar a eficiência do Algoritmo Híbrido, AH, foram estipulados os seguintes critérios obtidos a partir de 100 minimizações por função: taxa de minimizações bem-sucedidas (T.M.B.), média do número de avaliações da função objetivo (M.FOBJ), média entre o melhor ponto encontrado com o ótimo global conhecido (M.GL) e o tempo médio de processamento (T.M.P).

Quando os critérios de paradas do AG e na sequência do Simplex são atingidos o AH fornece

as coordenadas de um ponto e o valor da função objetivo avaliado nesse ponto $FOBJ_{AH}$. O resultado é então comparado com o mínimo analítico conhecido da função $FOBJ_{ANAL}$, para o julgamento do resultado obtido ser dado como bem-sucedido optou-se em utilizar o requisito apresentado na literatura, mencionada anteriormente, dada pela Equação 3.

$$|FOBJ_{AH} - FOBJ_{ANAL}| < \varepsilon_{rel}|FOBJ_{INIT}| + \varepsilon_{abs} \quad (3)$$

em que $\varepsilon_{rel} = 10^{-4}$, $\varepsilon_{abs} = 10^{-6}$ e $FOBJ_{INIT}$ é a média experimental do valor da função objetivo calculada em cem pontos selecionados aleatoriamente antes que o algoritmo seja executado.

Os resultados obtidos variando os operadores genéticos já mencionados são apresentados nas Tabelas 1 a 8. Todos os resultados apresentados nesta seção foram obtidos utilizando um microcomputador com a configuração Windows 7 Ultimate, processador Intel® Core™ i3-2310M CPU @ 2.10 GHz, memória (RAM) de 4 GB, sistema operacional de 64 Bits.

Função teste	T.M.B. (%)	M.OBJ.	M.GL.	T.M.P. (s)
RC	100	37	3.7191e-07	0.09
ES	90	388	9.6047e-09	4.02
GP	100	75	9.4281e-09	0.31
B2	98	422	8.9496e-09	5.95
R2	100	66	1.1844e-08	0.13
Z2	100	38	1.4429e-08	0.14

Tabela 1 – Torneio/ Dois pontos/ Mutação de 2,5%

Função teste	TMB (%)	M.OBJ	M.GL	T.M.P (s)
RC	100	42	3.7199e-07	0.004
ES	96	496	1.5547e-08	4.93
GP	96	53	1.5037e-08	0.19
B2	92	498	1.2512e-08	7.23
R2	100	176	1.5113e-08	2.35
Z2	100	317	8.8484e-09	4.16

Tabela 2 – Roleta/ Dois pontos/ Mutação de 2,5%

Função teste	TMB (%)	M.OBJ	M.GL	T.M.P (s)
RC	100	40	3.7160e-07	0.004
ES	96	463	1.1670e-08	3.50
GP	98	62	1.1134e-08	0.22
B2	81	492	1.1537e-08	3.77
R2	100	94	1.5956e-08	0.48
Z2	100	267	1.2879e-08	2.00

Tabela 3 – Amostragem Estocástica Uniforme/ Dois pontos/ Mutação de 2,5%

Função teste	TMB (%)	M.OBJ	M.GL	T.M.P (s)
RC	100	39	3.7127e-07	0.085
ES	95	383	1.4582e-08	3.43
GP	99	70	9.4616e-09	0.27
B2	98	444	1.2547e-08	6.57
R2	99	64	1.1700e-08	0.19
Z2	100	37	1.2717e-08	0.17

Tabela 4 – Torneio/ Um ponto/ Mutação de 2,5%.

Função teste	TMB (%)	M.OBJ	M.GL	T.M.P (s)
RC	98	40	3.6806e-07	0.1
ES	91	425	1.0615e-08	5.01
GP	100	80	9.7534e-09	0.8
B2	93	436	1.0269e-08	6.64
R2	100	68	1.0423e-08	0.08
Z2	100	38	1.1902e-08	0.066

Tabela 5 – Torneio/ Uniforme/ Mutação de 2,5%

Função teste	TMB (%)	M.OBJ	M.GL	T.M.P (s)
RC	99	40	3.6986e-07	0.11
ES	92	470	1.2615e-08	9.89
GP	94	46	1.2941e-08	0.12
B2	92	423	1.0828e-08	7.97
R2	100	68	1.3403e-08	0.12
Z2	100	38	1.0105e-08	0.12

Tabela 6 – Torneio/ baseado em Maioria/ Mutação de 2,5%

Função teste	TMB (%)	M.OBJ	M.G.L	T.M.P (s)
RC	100	40	3.6991e-07	0.11
ES	78	208	1.3576e-08	2.3
GP	97	49	1.1849e-08	0.18
B2	80	267	1.2256e-08	2.5
R2	100	68	1.3053e-08	0.16
Z2	100	37	1.0655e-08	0.06

Tabela 7 – Torneio/ Dois pontos/ Mutação de 5%

Função teste	TMB (%)	M.OBJ	M.G.L	T.M.P (s)
RC	100	40	3.7185e-07	0.09
ES	70	126	1.0347e-08	1.8
GP	88	127	1.0759e-08	0.1
B2	76	54	1.4743e-08	0.33
R2	100	68	1.7287e-08	0.1
Z2	100	44	1.1922e-08	0.24

Tabela 8 – Torneio/ Dois pontos/ Mutação de 0%

As Tabelas 1 a 3 apresentam resultados obtidos variando-se o método de seleção onde observa-se que os valores obtidos entre elas tiveram pequenas variações, todavia, ao analisar algumas funções específicas, como por exemplo a função B2 observa-se uma pequena variação nos valores da T.M.B. em que o melhor resultado esteve presente no método de seleção torneio, Tabela 1. Outro exemplo é a função Z2 que apresentou uma M.OBJ. muito menor no torneio, promovendo assim um menor tempo de processamento, uma vez que o número de avaliações da função objetivo foi menor, e por fim, o M.G.L. para o torneio, ao contrário dos outros dois métodos, esteve na faixa de 10^{-9} para a maioria das funções avaliadas o que reflete diretamente na precisão dos resultados obtidos.

Fixando-se o torneio, pelos motivos já comentados, para o método de seleção dos indivíduos, avaliou-se, na sequência, os operadores genéticos de crossover, um ponto, dois pontos, uniforme e baseado em maioria cujos os resultados são sumarizados nas Tabelas 1, 4, 5 e 6, respectivamente. Os resultados obtidos pelos diferentes métodos sinalizam que os operadores de crossover avaliados não influenciaram significativamente o desempenho do A.H., sendo que a influência mais aparente foi com relação as T.M.B. que teve maiores taxas para o operador de dois pontos, Tabela 1. A T.M.P., associado ao erro do valor da função objetivo, foi menor para o operador de dois pontos também, estando na ordem de 10^{-9} para a maioria das funções analisadas.

Por fim, variou-se a probabilidade de mutação, que é fundamental para todo AG pois é ela que garante a continuidade da existência de diversidade genética na população. Os testes realizados até então foram com 2,5% de mutação, e nas Tabelas 7 e 8, utilizou-se 5% e 0% de mutação respectivamente. Como já era esperado, quando não houve mutação os resultados de T.M.B. foram os piores já obtidos, isso se deve ao fato de que, segundo Linden (2012), quando a probabilidade de mutação for baixo demais, o AG agirá de forma extremamente parcimoniosa e a população não terá diversidade depois de um certo número de gerações, estagnando bem rápido devido à convergência genética.

Enquanto que para a probabilidade de mutação de 5%, os resultados também não foram tão satisfatórios quanto para a mutação de 2,5%, sendo a melhor para todas as funções analisadas. Uma taxa probabilidade de mutação muito alta o AG apresenta um comportamento similar a um algoritmo aleatório (*random walk*) e perderá suas características interessantes.

É importante ressaltar que em geral os resultados obtidos para todas as combinações de operadores não apresentaram grande variação, entretanto, uma sensível melhora do desempenho do AH pode ser obtida com uma certa combinação de operadores e de valores dos parâmetros do AG, no presente trabalho as condições que proveram os melhores resultados foram : Método de seleção torneio e o crossover de dois pontos com uma probabilidade de mutação de 2,5%.

Por fim, comparou-se os melhores resultados das minimizações para cinco diferentes funções de duas variáveis apresentada pelo AH com o de outras versões de algoritmos genéticos disponíveis na literatura como dados pela Tabela 9.

Algoritmo	Método	REFERÊNCIA
Algoritmo Híbrido (A.H.)	Híbrido: AG e Busca Simplex	<i>Presente trabalho</i>
<i>Continuous hybrid algorithm (CHA)</i>	Híbrido: AG e Busca Simplex	<i>Chelouah and Siarry, 2003</i>
<i>Continuous Genetic Algorithm (CGA)</i>	AG	<i>Chelouah and Siarry, 2000</i>
<i>Enhanced Continuous Tabu Search (ECTS)</i>	<i>Busca Tabu</i>	<i>Chelouah and Siarry, 2000</i>
<i>Enhanced Simulated Annealing (ESA)</i>	<i>Simulated Annealing</i>	<i>Siarry et al., 1997</i>
<i>Continuous Reactive Tabu Search (CRTS min.) minimum</i>	<i>Busca Tabu</i>	<i>Battiti and Tecchiolli, 1996</i>
<i>Continuous Reactive Tabu Search (CRTS ave.) average</i>	<i>Busca Tabu</i>	<i>Battiti and Tecchiolli, 1996</i>
<i>Taboo Search (TS)</i>	<i>Busca Tabu</i>	<i>Cvijovic and Klinowski, 1995</i>
<i>INTEROPT</i>	<i>Simulated Annealing</i>	<i>Bilbro and Snyder, 1991</i>

Tabela 9 – Outros métodos presentes na literatura utilizados para comparação.

Os resultados disponíveis para a média da função objetivo, M. OBJ., para os diferentes métodos foram apresentados na Tabela 10. A diferença entre o CRTS min. e o CRTS ave. é que para o primeiro o algoritmo faz uso do valor mínimo da FOBJ. entre os vizinhos considerados, já o CRTS ave. o algoritmo usa o valor médio da FOBJ.

Função	Método								
	A.H.	CHA	CGA	ECTS	ESA	CTRS min.	CTRS ave.	TS	INTEROPT
RC	37	295	620	245	-	41	38	492	4172
GP	75	259	410	231	783	171	248	486	6375
B2	422	132	320	210	-	-	-	-	-
R2	66	459	960	480	796	-	-	-	-
Z2	38	215	620	195	15820	-	-	-	-

Tabela 10 – Média do número de avaliações da função objetivo obtidos para diversos métodos para funções de duas variáveis.

Como mostrado na Tabela 10, para as funções de duas variáveis avaliadas, os resultados obtidos para a média do número de avaliações da função objetivo, M.OBJ., superaram as expectativas sendo melhores que os presentes na literatura para a maioria dos casos, exceto para a função B2, mas ainda assim os resultados eram esperados devido ao fato de que, para esta função, o mínimo global se encontra em um vale muito estreito com relação ao domínio de busca, o mesmo ocorre para a função Easom. Uma possível solução para esse problema seria aumentar o tamanho da população inicial ou aumentar o número de gerações.

O algoritmo CHA proposto por *Chelouah e Siarry* está baseado em métodos similares aos deste trabalho, entretanto precisou no geral de mais iterações para encontrar o mínimo global, se faz importante frisar que o CHA não é restrito a funções de duas variáveis independentes, podendo ser aplicado para funções n dimensionais. Contudo, o AH proposto no presente trabalho apresentou resultados mais satisfatórios mostrando-se na maioria das vezes semelhantes ou melhores que alguns outros métodos computacionais disponíveis na literatura, convergindo para o resultado final com uma maior eficiência, ou seja, com uma menor passagem pela função objetivo em um menor tempo de processamento.

4 CONCLUSÕES

No presente trabalho desenvolveu-se um algoritmo híbrido, AH, para otimização global de funções bidimensionais em que o mesmo foi testado com algumas funções *Benchmark* e na sequência comparado com resultados de outros autores presentes na literatura. As várias combinações de parâmetros do algoritmo genético foram testadas e a combinação que apresentou os melhores resultados foi o método de seleção torneio e o *crossover* de dois pontos com uma probabilidade de mutação de 2,5%. Após a identificação da área promissora pelo AG, o Simplex de Nelder e Mead não apresentou grande dificuldade em encontrar o ponto exato do mínimo global. O AH apresentou resultados satisfatórios e melhores que os disponíveis na literatura para a maioria das funções testadas, com exceção da função B2 devido ao seu domínio de busca ser muito grande e o A.H. precisar de mais iterações para encontrar o mínimo global.

REFERÊNCIAS

- VIANNA, Dalessandro Soares; GOMES, Roberta Claudino Barreto Pessanha. Um algoritmo evolutivo para o problema de roteamento em arcos capacitados. *Sistemas & Gestão*, v. 1, n. 2, p. 116-131, 2009.
- LIMA, Ednaldo Oliveira. Algoritmo genético híbrido aplicado à otimização de funções. Trabalho de conclusão de curso, UFES, 2008.
- CHELOUAH, Rachid; SIARRY, Patrick. *Genetic and Nelder–Mead algorithms hybridized for a more accurate global optimization of continuous multimodal functions*. *European Journal of Operational Research*, v. 148, n. 2, p. 335-348, 2003.
- GOMES, F. I. F. N. et al. Estratégia híbrida de otimização sem derivadas para a inversão completa da forma da onda. *Holos*, v. 33, n. 2, p. 2-34, 2017.
- DRUMMOND, L. M. A.; OCHI, L. S.; VIANNA, D. S. (2001). *An asynchronous parallel metaheuristic for the period vehicle routing problem*. *Future Generation Computer Systems*, 17:379-386.
- MELO, Vinícius Veloso de. Técnicas de Aumento de Eficiência para metaheurísticas aplicadas a otimização global contínua e discreta. 2009. Tese de Doutorado. Universidade de São Paulo.
- LINDEN, Ricardo. Algoritmos genéticos. 3 Ed. Rio de Janeiro: Editora Ciência Moderna Ltda., 2012.
- R. CHELOUAH, P. SIARRY, *A continuous genetic algorithm designed for the global optimization of multimodal functions*, *Journal of Heuristics* 6 (2000) 191–213
- R. Chelouah, P. Siarry, *Tabu search applied to global optimization*, *European Journal of Operational Research* 123/2 (2000) 30–44, *Special issue on combinatorial optimization*.