



Um Estudo Sobre a Utilização de Codificação Real no Algoritmo de Otimização Extrema Generalizada Adaptativo

Leonardo Becker da Luz¹, Fabiano Luis de Sousa², Ronan Arraes Jardim Chagas³

¹Instituto Nacional de Pesquisas Espaciais, São José dos Campos, SP, Brasil
Aluno de Mestrado do curso de Engenharia e Gerenciamento de Sistemas Espaciais - CSE.

²Instituto Nacional de Pesquisas Espaciais, São José dos Campos, SP, Brasil
Divisão de Mecânica Espacial e Controle - DIMEC.

³Instituto Nacional de Pesquisas Espaciais, São José dos Campos, SP, Brasil
Divisão de Sistemas Espaciais - DISEP.

leobeckerdaluz@gmail.com

Resumo. O GEO é um algoritmo evolutivo que recentemente teve uma versão adaptativa (A-GEO) desenvolvida. No presente trabalho, é apresentado um estudo que avalia o efeito na performance do algoritmo quando a codificação das variáveis de projeto é modificada de binária para real. O estudo foi feito utilizando um conjunto de 5 funções teste. Observou-se que as implementações com codificação real apresentaram performance bem superior às com codificação binária para a maioria das funções. Todavia, a melhora dos resultados por meio da utilização da codificação real parece ser bastante dependente de como as variáveis de projeto são modificadas. Ademais, no presente estudo, a codificação real levou a introdução de um novo parâmetro de controle, que precisa ser ajustado.

Palavras-chave: Algoritmos Evolutivos; Controle de Parâmetros; Sistemas Espaciais;

1. Introdução

A fase conceitual de desenvolvimento de uma nova espaçonave é onde busca-se encontrar a arquitetura que mais se adapta a missão [Lau et al., 2014]. Portanto, qualquer automação numérica desse processo que possa criar e avaliar as melhores soluções possíveis [Cuco, De Sousa e Neto, 2015], reduzir os custos e aumentar a qualidade do conceito de missão é altamente desejável [Hinckley, Englander e Hitt, 2017], como por exemplo a otimização numérica, que calcula a solução de melhor desempenho de acordo com objetivos e restrições formulados matematicamente [Mueller e Ochsendorf, 2015].

Um dos algoritmos de otimização baseados na seleção natural é o algoritmo de Otimização Extrema Generalizada (GEO) proposto por [De Sousa et al., 2003]. O GEO já foi aplicado a diversos problemas de engenharia como em [De Sousa et al. 2007], [De Albuquerque, De Sousa e Montes, 2016] e [Muraoka et al., 2006], e pode ser aplicado a problemas com espaços de projeto não convexos ou desconexos e com variáveis contínuas, discretas ou inteiras. Uma vantagem do GEO em relação aos outros algoritmos evolutivos é o fato dele



possuir apenas um parâmetro livre a ser ajustado. A fim de alterar automaticamente os valores deste parâmetro de forma que este se adapte em função dos resultados da otimização ao longo da busca, um mecanismo de controle de parâmetros foi implementado no GEO no trabalho de [Barroca, 2019], dando origem ao algoritmo A-GEO. Embora essa implementação tenha apresentado um desempenho superior ao GEO para um conjunto de testes numéricos, quando comparado a outros algoritmos evolutivos recentes, o A-GEO possui uma performance inferior para um conjunto de funções teste com variáveis contínuas.

Ambas as versões canônica e adaptativa do GEO possuem a codificação binária das variáveis de projeto, o que apresenta saltos no espaço de projeto a cada mutação de bit em cada uma das variáveis de projeto de um problema, além de ser necessária a definição da quantidade de bits que codifica cada variável. Em estudo recente, [Da Luz, De Sousa e Chagas, 2021] verificaram que a performance do A-GEO pode ser melhorada quando se aplica a abordagem adaptativa na versão GEOvar, mas esta ainda usa codificação binária para as variáveis. Em seu trabalho, [Mainenti-Lopes, Souza e De Sousa, 2012] apresentaram uma implementação do GEO como a codificação real das variáveis de projeto, que obteve um desempenho muito superior para o conjunto de funções estudado quando comparado com as implementações com codificação binária. Desta forma, torna-se interessante o desenvolvimento de uma versão do A-GEO utilizando a codificação real, que permitiria a variação de cada variável de forma contínua.

No presente trabalho é realizado um estudo preliminar da adaptação do A-GEO para operar diretamente com a codificação real das variáveis de projeto. Levando em consideração que diversos problemas de engenharia possuem variáveis contínuas, este passo para o aumento de performance do A-GEO visa o desenvolvimento de uma versão futura adaptativa no estado da arte e competitiva com algoritmos de otimização de alta performance.

2. Metodologia

A codificação real das variáveis de projeto codifica cada variável com um número real (Figura 1). Portanto, esta pode variar continuamente, eliminando os saltos ocorridos na codificação binária.

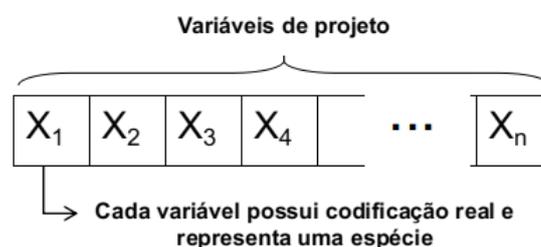


Figura 1. N variáveis de projeto com codificação real.

Uma implementação do GEO com codificação real, GEOreal1, introduzida por [Mainenti-Lopes, Souza e De Sousa, 2012], possui o mesmo funcionamento do GEO canônico, com exceção da forma como as variáveis de projeto são codificadas e perturbadas. Ao invés de



mutar um bit, a perturbação aplicada em cada variável de projeto é realizada diretamente no número real. Para tanto, o valor da variável é atualizado utilizando:

$$x'_i = x_i + N(0, \sigma)x_i, \quad (1)$$

onde x_i representa o valor atual da variável e $N(0, \sigma)$ representa uma distribuição normal com média 0 e desvio padrão σ . Neste trabalho, esta codificação e forma de perturbar as variáveis reais foram aplicadas ao A-GEO, gerando uma nova implementação denominada aqui de A-GEOreal1.

Originalmente, [Barroca, 2019] desenvolveu duas versões para o A-GEO, denominadas A-GEO1 e A-GEO2. Estas implementações diferem na população de referência utilizada no cálculo da métrica CoI (Chance-of-Improvement). Essa métrica possibilita saber durante a busca se as soluções estão melhorando com relação a uma solução de referência, a fim de modificar o parâmetro de controle do GEO (τ) para tornar a busca mais determinística ou mais estocástica. Enquanto no A-GEO1 é utilizada a melhor população encontrada até a iteração corrente, no A-GEO2 é utilizada a população corrente. A implementação A-GEO2 teve um desempenho superior em relação ao A-GEO1 para um conjunto de funções teste, aparentemente resultado da forma como ambos algoritmos equilibram a razão entre exploração e aprimoramento (*exploration vs. exploitation*) durante a busca da solução ótima no espaço de projeto. Desta forma, este presente trabalho implementou somente a versão A-GEO2 com codificação real, denominada aqui de A-GEO2real1.

A avaliação de desempenho dos algoritmos foi realizada através de um conjunto de 5 funções teste de minimização amplamente utilizadas para fins de benchmarking em algoritmos de otimização, conforme mostrado na Tabela 1.

Tabela 1. Conjunto de funções teste

Função	N	Bits por variável	Equação	Restrições Laterais	x^*	fx^*
Griewangk	10	14	$1 + \sum_{i=1}^N \frac{X_i^2}{4000} - \prod_{i=1}^N \cos\left(\frac{X_i}{\sqrt{i}}\right)$	$-600,0 \leq X_i \leq 600,0$	(0, 0, ...)	0
Rastringin	20	16	$3,0N + \sum_{i=1}^N [X_i^2 - 3,0 \cos(2\pi X_i)]$	$-5,12 \leq X_i \leq 5,12$	(0, 0, ...)	0
Rosenbrock	2	13	$\sum_{i=1}^{N-1} [100(X_i^2 - X_{i+1})^2 + (1 - X_i)^2]$	$-2,048 \leq X_i \leq 2,048$	(1, 1, ...)	0
Schwefel	10	16	$418,989N - \sum_{i=1}^N X_i \sin(\sqrt{ X_i })$	$-500,0 \leq X_i \leq 500,0$	(420.9, 420.9, ...)	0
Ackley	30	16	$20 + \exp\left(-0,2 \sqrt{\frac{1}{N} \sum_{i=1}^N X_i^2}\right) - \exp\left(\frac{1}{N} \sum_{i=1}^N \cos(2\pi X_i)\right)$	$-30,0 \leq X_i \leq 30,0$	(0, 0, ...)	0



Os algoritmos GEO, A-GEO2, GEOreal1 e A-GEO2real1 foram executados em cada uma dessas funções a fim de comparar os seus desempenhos. Para isso, primeiramente foi realizado o ajuste do parâmetro τ do GEO e GEOreal1 e do parâmetro σ para o GEOreal1 e A-GEO2real1. Estes ajustes permitem encontrar os valores destes parâmetros para cada algoritmo que resultam no melhor valor da função objetivo em cada problema. Com esses parâmetros configurados, cada algoritmo foi executado de forma independente 50 vezes com um critério de parada definido como cem mil (100000) avaliações da função objetivo (NFE). A fim de observar a performance de cada algoritmo durante a execução, o melhor valor da função objetivo foi armazenado a cada 500 NFEs e a média das 50 execuções para cada um desses valores armazenados foi obtida a fim de gerar os gráficos de desempenho dos algoritmos para cada uma das funções.

3. Resultados e Discussão

A etapa de ajuste de parâmetros gerou os valores de cada parâmetro que combinados resultam no melhor desempenho de cada algoritmo para cada função. Essa informação é mostrada na Tabela 2.

Tabela 2. Parâmetros obtidos através do ajuste de parâmetros.

Algoritmo	Parametro Livre	Griewangk	Rastringin	Rosenbrock	Schwefel	Ackley
GEO	τ	1,25	1,25	1,00	0,75	3,25
GEOreal1	τ	1,5	1,5	4,0	5,0	1,0
	σ	0,8	1,2	2,2	1,8	0,8
A-GEO2real1	σ	1,0	1,0	1,8	2,4	0,8

Estes parâmetros foram configurados em cada algoritmo para cada problema e as execuções foram realizadas, gerando os gráficos de desempenho apresentados na Figura 2. Cada ponto de cada curva dos gráficos representa o valor médio da função objetivo em 50 execuções, em função do NFE durante a busca. É importante observar que para o algoritmo A-GEO2 não há a necessidade de ajuste de parâmetros.

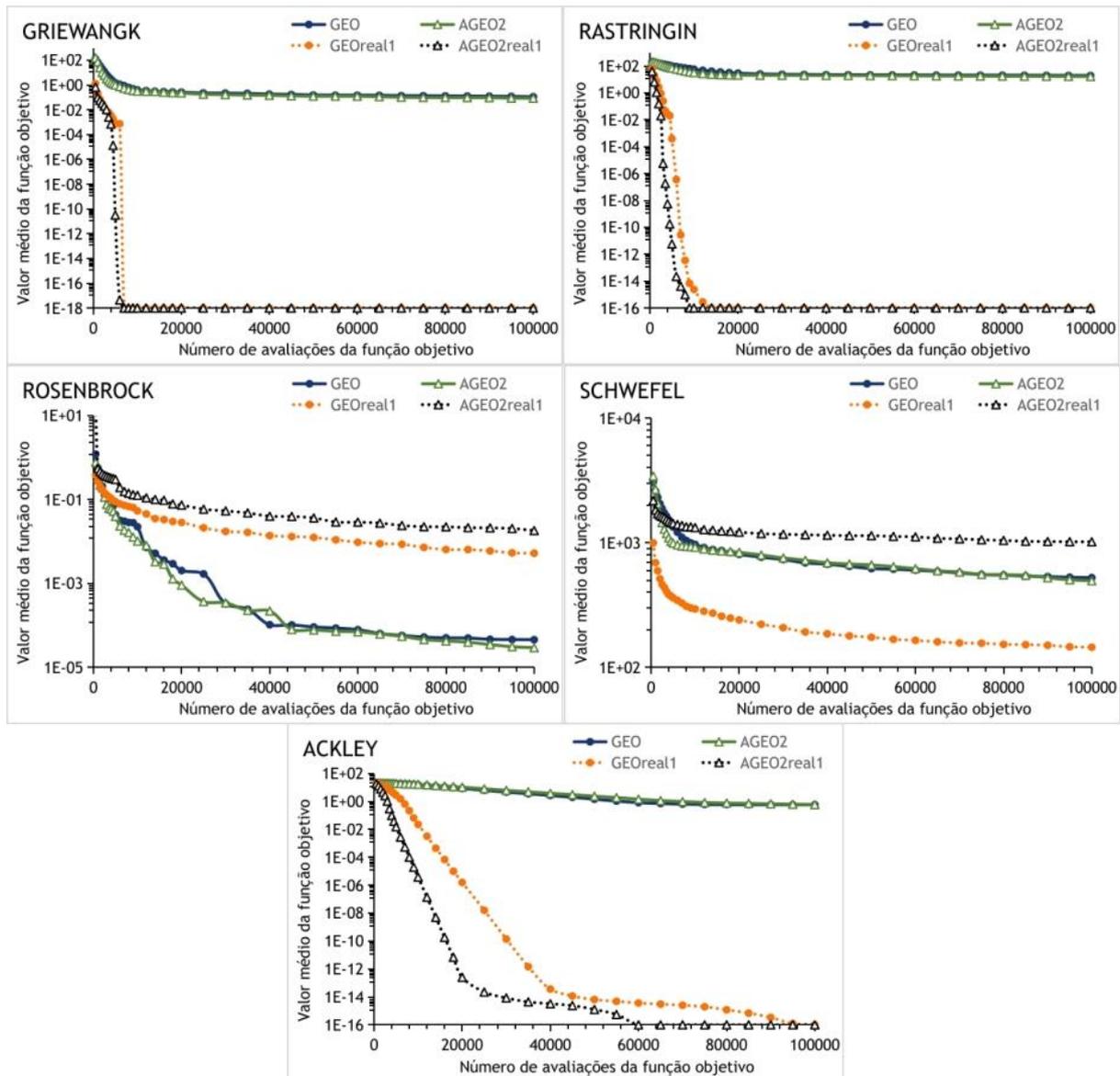


Figura 2. Comparação de desempenho entre os GEO, A-GEO2, GEOreal1 e A-GEO2real1.

De forma geral, é possível observar que as implementações com codificação real possuem um desempenho muito superior em relação aos resultados obtidos por meio das implementações com codificação binária na maioria das funções. De fato, a codificação binária só foi melhor que a codificação real para a função Rosenbrock. Observa-se também que os algoritmos GEOreal1 e A-GEO2real1 atingiram o ótimo global ($f_x^* = 0$) para as funções Griewangk, Rastrigin e Ackley.

Em complemento, observa-se que as duas funções onde os algoritmos com codificação real não atingiram o ótimo (Rosenbrock e Schwefel) foram as duas cujo valor das variáveis no ótimo global (X^*) não é zero. Isto pode estar associado ao processo de perturbação das



variáveis de projeto, conforme descrito na equação 1, já que quanto mais próximo do mínimo global, para um dado valor de σ , maior a perturbação nos valores das variáveis. Isso indica que a abordagem utilizada na equação 1 não é adequada para problemas onde o mínimo global está localizado em $x^* \neq 0$. É interessante observar que, comparando-se apenas as versões onde a diferença de implementação se dá somente em como as variáveis são codificadas, A-GEO2 e A-GEO2real1, o desempenho daquela com codificação real é menor apenas para as funções teste que possuem $x^* \neq 0$.

4. Conclusão

Os resultados confirmam que utilizar codificação real em implementações do GEO para problemas com variáveis contínuas tende a levar a ganhos significativos no desempenho do algoritmo, inclusive para suas versões adaptativas. Entretanto, isto parece ser dependente de como é feita a perturbação nas variáveis de projeto. A codificação real requer a adoção de uma estratégia para tornar adaptativa a perturbação das variáveis, o que não foi objeto do presente trabalho, mas que se encontra atualmente em estudo no intuito de se obter uma versão completamente adaptativa do GEO com codificação real.

Finalmente, é importante ressaltar que, embora a utilização de uma abordagem adaptativa para o GEO venha se mostrando vantajosa, ela ainda está em uma fase embrionária de desenvolvimento. Os avanços no uso de algoritmos evolutivos e meta-heurísticas inspiradas na natureza em aplicações de engenharia trazem uma oportunidade e um desafio para os pesquisadores avançarem na otimização de produtos, sistemas e serviços. De fato, o presente trabalho é mais um passo na introdução do controle adaptativo de parâmetros no GEO, de forma a torná-lo cada vez mais eficiente e amigável ao usuário em problemas de otimização.

Agradecimentos: O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Código de Financiamento 001.

Referências

- BARROCA, E. A New Adaptive Evolutionary Algorithm For Design Optimization. Master's thesis, Instituto Nacional de Pesquisas Espaciais (INPE), São Jose dos Campos, SP, Brazil. 2019.
- CUCO, A. P. C., DE SOUSA, F. L., & NETO, A. J. S. A multi-objective methodology for spacecraft equipment layouts. *Optimization and Engineering*, 16(1), 165-181. 2015.
- DA LUZ, L. B., DE SOUSA, F. L., & CHAGAS, R. A. Implementação Adaptativa de Variante do Algoritmo de Otimização Extrema Generalizada (GEO). In *Anais do XXXIII Seminário Integrado de Software e Hardware* (pp. 271-278). SBC. 2021.
- DE ALBUQUERQUE, B. F. C., DE SOUSA, F. L., & MONTES, A. S. Multi-objective approach for the automatic design of optical systems. *Optics express*, 24(6), 6619-6643. 2016.
- DE SOUSA et al. New stochastic algorithm for design optimization. *AIAA journal*, 41(9), 1808-1818. 2003.
- DE SOUSA, F. L., et al. Application of the generalized extremal optimization algorithm to an inverse radiative transfer problem. *Inverse Problems in Science and Engineering*, 15(7), 699-714. 2007.



HINCKLEY Jr, D., ENGLANDER, J., & HITT, D. Multi-Objective Optimization of Spacecraft Trajectories for Small-Body Coverage Missions. 2017.

LAU, V., et al. A multidisciplinary design optimization tool for spacecraft equipment layout conception. *Journal of Aerospace Technology and Management*, 6, 431-446. 2014.

MAINENTI-LOPES, I., SOUZA, L. C. G., & DE SOUSA, F. L. Design of a nonlinear controller for a rigid-flexible satellite using multi-objective Generalized Extremal Optimization with real codification. *Shock and vibration*, 19(5), 947-956. 2012.

MUELLER, C. T., & OCHSENDORF, J. A. Combining structural performance and designer preferences in evolutionary design space exploration. *Automation in Construction*, 52, 70-82. 2015.

MURAOKA, I. et al. Stochastic spacecraft thermal design optimization with low computational cost. *Journal of Spacecraft and Rockets*, 43(6), 1248-1257. 2006.