



Resumo Expandido

Título da Pesquisa (Português): Análise e implementações de metaheurísticas bio-inspiradas		
Título da Pesquisa (Inglês): Analysis and implementation of bio-inspired metaheuristics		
Palavras-chave: Otimização, Colônia de Vagalumes, Inteligência computacional		
Keywords: Optimization, Firefly algorithm, Computational intelligence		
Campus: IFMG-Sabará	Tipo de Bolsa: PIBIC	Financiador: FAPEMIG
Bolsista(s): Deylon Carlo Fidelis Couro		
Professor Orientador: Carlos Alexandre Silva		
Área de Conhecimento: Computação		Edital: 129/2014

Resumo: O tempo computacional para gerar soluções ganhou muita relevância, por este motivo, métodos exatos para resolução de problemas computacionais vêm sendo descartados e os métodos de heurísticas e meta-heurísticas estão ganhando cada vez mais espaço, principalmente quando se fala de problemas de otimização. A maioria destes problemas de otimização apresentam características de NPs, o que explica o fato de algoritmos heurísticos serem largamente utilizados pois, além de apresentar um tempo computacional de resolução aceitável, a solução proposta é dita como boa. Neste artigo foi proposta a utilização do algoritmo de Colônia de Vagalumes (CV), que usa de técnicas de heurística para otimização de funções. Para inferir se este algoritmo possui a capacidade de convergir as funções utilizadas para seus ótimos, foram utilizadas clássicas funções de 2 e 3 dimensões para realizar os testes, incluindo algumas que não foram utilizadas e testadas nos trabalhos da literatura. Os testes realizados permitiram concluir que o CV consegue atingir o ótimo para todas as funções testadas em um tempo computacional razoável, além de superar alguns resultados da literatura.

Abstract: The computational time to generate solutions has gained much importance, therefore, exact methods for solving computational problems have been discarded and methods of heuristics and meta-heuristics are gaining more and more space, especially when it comes to optimization problems. Most of these optimization problems have NPs features, which explains the fact that heuristic algorithms are widely used because, in addition to a computational time acceptable resolution, the proposed solution is said to be good. In this paper we propose the use of firefly colony algorithm (CV) that uses heuristic techniques for optimization functions. To infer whether this algorithm has the ability to converge the functions used for its fine, we used classical functions of 2:03 dimensions to perform the tests, including some that have not been used and tested in the work of literature. The tests showed that the CV can achieve optimal for all functions tested in a reasonable computational time, and overcome some results in the literature.

INTRODUÇÃO:

Problemas de otimização são comumente encontrados em aplicações práticas de grande valor econômico e/ou social, como, quando se deseja determinar a quantidade mínima de leitos de um hospital, o maior nível de produção de uma indústria, entre outros problemas. Por meio da inteligência computacional (IC) tem sido possível resolver diversos problemas práticos reais, e por isso tem surgido muitos algoritmos de IC nos últimos anos. Dentre os algoritmos mais recentes é possível citar: colônia de vaga-lume (Yang, 2008), colônia de bactérias (Niu & Wang, 2013), otimizador da formiga-leão (Mirjalili, 2015), entre outros. Os algoritmos citados acima baseiam-se em inteligência por enxames.

Funções bidimensional (2D) e tridimensionais (3D) são frequentemente utilizadas na construção de problemas de otimização. Mesmo que a dimensionalidade da função seja pequena a complexidade do problema a qual a função está associada pode ser elevada. Funções multimodais podem contribuir com o incremento do grau de dificuldade para a resolução do problema, visto que uma função multimodal é aquela que possui dois ou mais ótimos locais. Por esses motivos foram selecionadas funções que se encaixam nas características citadas, como, Alpine02, Bird.

As configurações destas funções são apresentadas na Fig. (1), respectivamente por 1 e 2, sendo descritas sua formulação matemática, a região de factibilidade, o ponto de otimalidade e o valor ótimo da função, sempre considerando a otimalidade como um problema de minimização.

1	$f_1(x) = \prod_{i=1}^N \sqrt{x_i} \text{sen}(x_i)$ $0 \leq x_1, x_2 \leq 10$ $x^* = (7.917, 7.917)$ $f(x^*) = -6.1295$	2	$f_2(x) = (x_1 - x_2)^2 + e^{(1-\text{sen}(x_1))^2} \cos(x_2) + e^{(1-\cos(x_1))^2} \text{sen}(x_2)$ $-2\pi \leq x_1, x_2 \leq 2\pi$ $x^* = (4.701055751981055, 3.15294601960139)$ $f(x^*) = -106.7645367198034$
---	---	---	--

Figura 1: Configurações das funções 2D

A visualização das funções 2D podem ser vistas na Fig. (2)

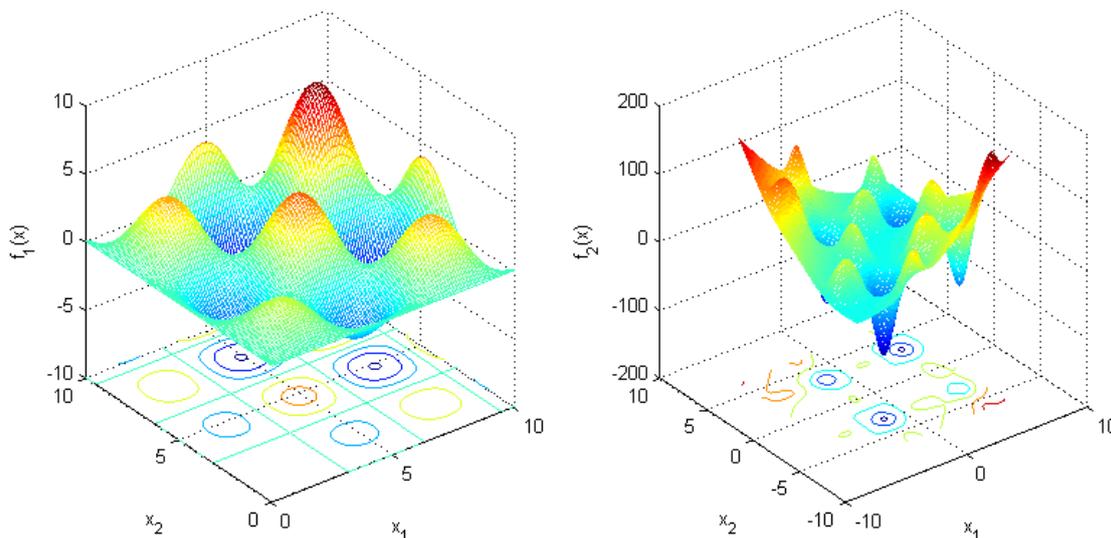


Figura 2: Representação das funções *Alpine02* e *Bird*

Os algoritmos citados baseiam-se em inteligência por enxames que consiste em um conjunto de agentes trabalhando simultaneamente para solucionar um problema. Se fosse pego um indivíduo, separadamente, para resolver um determinado problema, certamente não seria encontrada a solução pois, sozinhos estes agentes possuem capacidade de solução limitada. O algoritmo colônia de vagalumes (ACV) foi estruturado de acordo com o comportamento social de vagalumes ou pirlampos. Cada espécie produz um padrão único de pequenos lampejos rítmicos luminosos. A bioluminescência, emissão de luz, gerada por estes insetos

influenciam na atração entre as espécies para fins de reprodução e na atração de presas ou prevenção de predadores. Em relação a um ponto fixo, a intensidade da luz emitida por um vaga-lume diminui a medida que este se afasta do ponto fixo, ou seja, a intensidade da luz é inversamente proporcional à distância. Devido a absorção da luz pelo ar, costuma-se considerar que

Equação 1

onde k é a intensidade luminosa de um vaga-lume e r é a distância entre dois vaga-lumes em relação ao ponto fixo. No ACV, a intensidade luminosa é associada à função objetivo a ser otimizada.

O algoritmo proposto por (Yang, 2008) considera três regras fundamentais:

- Todos os vagalumes são unissex, ou seja, qualquer vagalume pode ser atraído por outro.
- A atratividade é diretamente proporcional a luminosidade, ou seja, o vagalume que produz menor intensidade de “brilho” será atraído pelo vagalume que produz uma intensidade maior. Sendo que esta luminosidade decai com o aumento da distância. Se não houver brilho o vagalume move-se aleatoriamente.
- A luminosidade de um vagalume é determinada pela função objetivo

O movimento do vagalume depende de três fatores: a atratividade (β), a absorção da luz pelo meio (λ) e a aleatoriedade (α). A variação da atratividade β entre os vagalumes a uma distância r é definida como:

Equação 2

O movimento do vagalume i no tempo discreto t , atraído pelo vagalume j , é dado pela equação:

Equação 3

Sendo β o parâmetro de atratividade e α um parâmetro de aleatoriedade.

A complexidade do algoritmo segue a análise apresentada em (Yang & He, 2013). No Algoritmo 1 existem dois loops internos, um fazendo a comparação entre a intensidade luminosa dos pares e variando todos vaga-lumes e outro loop movendo os vaga-lumes de acordo com a Equação (3). Com uma população de n vaga-lumes, teríamos uma complexidade $O(n^2)$. É possível alterar a implementação do algoritmo para obter uma complexidade $O(n \log(n))$, mas para isso precisaríamos usar um único loop para enumerar a atratividade e a luminosidade dos vaga-lumes. Segundo (Yang & He, 2013) a eficiência do ACV se dá pela subdivisão automática do método e a capacidade de tratar multimodalidade, permitindo que os vaga-lumes

possam encontrar todos os “ótimos” simultaneamente. Aliado a estas vantagens, os parâmetros do ACV podem ser ajustados para controlar a aleatoriedade de modo a acelerar a convergência do método.

Algoritmo 1: ALGORITMO COLÔNIA DE VAGA-LUMES

Entrada: $n, MaxGen, \beta, \alpha, \gamma$ { número de vaga-lumes, número máximo de gerações, parâmetros de atratividade, aleatoriedade e absorção de luz }

Saída: g^* { melhor solução obtida pelos vaga-lumes }

início

Determine função objetivo $f(x)$ para $x = (x_1, \dots, x_d)^T$

Gere população inicial de vaga-lumes $x^i, i = 1, 2, \dots, n$

Defina a intensidade de luz como: $I_i = f(x^i)$

enquanto $gerações \leq MaxGen$ **faça**

para todos os vaga-lumes x_i e x_j **faça**

se $I_i < I_j$ **então**

 | Mova o vaga-lume x_i para o x_j de acordo com Eq. (2)

fim

 Varie a atratividade com a distância r via $e^{-\gamma r^2}$

 Calcule novas soluções e atualize a intensidade luminosa

fim

 Ordene os vaga-lumes e encontre a melhor solução g^*

fim

fim

retorna g^*

Figura 3: Representação do Algoritmo Colônia de Vagalumes

METODOLOGIA:

No contexto de pesquisa atual apresentam-se diversas técnicas para solucionar problemas de otimização, como, heurísticas e meta-heurísticas. Por este motivo devem ser feitas análises para a verificação da eficiência dessas técnicas. O modelo e análise utilizado para avaliar o desempenho do algoritmo de CV foi a através de cálculos de mínimos locais (possivelmente globais) de funções multimodais.

Na literatura existem diversos trabalhos que exibem benchmarks de métodos computacionais aplicados a funções multimodais e multidimensionais, como em (Cuevas et al., 2013), (Li et al., 2013), (Jamil & Yang, 2013), (Im et al., 2004) e (Dieterich & Hartke, 2012). O trabalho de (Krishnanand e Ghose, 2009) apresenta um benchmark de algoritmos baseados em inteligência computacional aplicados a funções multimodais e multidimensionais, incluindo uma versão de algoritmo também baseado no comportamento de vaga-lumes, denominado de Glow-Worm, porém apresentando significativas diferenças com o ACV, especialmente no modo da “movimentação” das soluções nas vizinhanças.

Para as simulações computacionais foi utilizado o software MatLab RR2009a, 64-bit, versão 7.8.0.347, executado em sistema operacional Windows 8.1 de 64 bit, na arquitetura de processador Intel ® Core™ i5-4200M CPU 2.50GHz com memória de 4 GB. O benchmark é constituído de funções multimodais sendo duas

funções de duas dimensões (2D), quatro funções de três dimensões (3D). Para cada função foram realizadas 30 simulações com a configuração de 500 gerações e 50 vaga-lumes. Para a construção do benchmark foram considerados 125 variações dos parâmetros α , β e γ conforme apresentado na Tabela 1. Note que apenas o valor inicial de γ

Parâmetros	Configurações				
α	0.1	0.25	0.50	0.75	1.00
β	0.1	0.25	0.50	0.75	1.00
γ	0.01	0.25	0.50	0.75	1.00

Tabela 1: Configuração dos parâmetros

difere do padrão dos demais parâmetros, pois queríamos representar uma configuração de uma “quase nula” absorção da luz pelo meio. Estas divisões das configurações representariam casos extremos, bem como medianos e intermediários.

RESULTADOS E DISCUSSÕES:

Nas tabelas 2, 3 são apresentadas as informações para inferir sobre a eficiência do método computacional utilizado para construir o benchmark. Na primeira coluna (“N”) está indicada a dimensão da função. Na segunda coluna (“Função”) é descrito o nome da função utilizada. Na terceira coluna (“ $f_{MIN} \leq f_{LIT}^*(em\%)$ ”) é apresentada a porcentagem das simulações em que o ACV, através do valor ótimo médio obtido pelo algoritmo, atingiu o melhor valor da literatura ou até mesmo foi melhor do que este valor, considerando uma aproximação de 10^{-3} . A quarta coluna (“ $f_{MIN} < f_{LIT}^*(em\%)$ ”) apresenta a porcentagem das simulações em que o ACV, através do valor mínimo obtido pelo algoritmo, atingiu o melhor valor da literatura ou até mesmo foi melhor do que este valor, considerando uma aproximação de 10^{-3} . As informações sobre possíveis resultados que superaram os resultados da literatura estão presentes na quinta coluna (“ $f_{MIN} < f_{LIT}^*$ ”), sendo que “Sim” indica que houve superação de resultado, “Não” não ocorreu superação e “Inconclusivo” indica que não foi possível afirmar com exatidão se houve superação ou não. A maioria das funções tem como fonte o trabalho de (Jamil & Yang, 2013), sendo as exceções comentadas e apresentadas no decorrer da análise dos resultados.

N	Função	$\overline{f^*} \leq f_{LIT}^*$ (em %)	$f_{MIN} \leq f_{LIT}^*$ (em %)	$f_{MIN} < f_{LIT}^*$
2	<i>Alpine02</i>	96.00	100.00	Inconclusivo
2	<i>Bird</i>	61.60	100.00	Sim

Tabela 2: Eficiência do algoritmo CV para funções 2D

N	Função	$\overline{f^*} \leq f_{LIT}^*$ (em %)	$f_{MIN} \leq f_{LIT}^*$ (em %)	$f_{MIN} < f_{LIT}^*$
3	<i>SchmidtVetters</i>	100.00	100.00	Sim
3	<i>Mishra09</i>	94.40	100.00	Não
3	<i>Gulf</i>	100.00	100.00	Sim
3	<i>BoxBetts</i>	100.00	76.80	Não

Tabela 3: Eficiência do algoritmo CV para funções 3D

CONCLUSÕES:

Neste trabalho foi apresentado um benchmark de funções 2D e 3D com o intuito de avaliar a capacidade de otimização do algoritmo. Os parâmetros de aleatoriedade, absorção da luz pelo meio e atração entre os vagalumes do algoritmo sofreram variações. Por se tratar de um algoritmo de tempo polinomial, pode-se dizer que o tempo de resolução do ACV é baixo, além de apresentar uma configuração de 50 vagalumes e o número máximo de gerações ser 500. Outro aspecto interessante é que além de ser um bom otimizador o algoritmo conseguiu superar alguns resultados registrados na literatura, o que não era esperado. Posteriormente o número de funções para o benchmark será ampliada com funções N-Dimensionais multimodais, o que torna mais complexa a otimização.

REFERÊNCIA BIBLIOGRÁFICA:

- [1] Yang, X. S., 2008. Nature-Inspired Metaheuristic Algorithms. Addison-Wesley, Luniver Press, UK.
- [2] Niu, B. & Wang, H., 2013. Bacterial colony optimization. Discrete Dynamics in Nature and Society, vol. 2012, pp. 1-28.
- [3] Mirjalili, S., 2015. The ant lion optimizer. Advances in Engineering Software, vol. 83, pp. 80-98.
- [4] Yang, X. S., He, X., 2013. Firefly algorithm: recent advances and applications. International Journal of Swarm Intelligence, vol. 1, n. 1, pp.36-50.
- [5] Cuevas, E., Zaldívar, D. & Cisneros, M. P., 2013. A swarm optimization algorithm for multimodal functions and its application in multicircle detection. Mathematical Problems in Engineering, vol. 2013, pp. 1-22.

[6] Li,X.,Engelbrecht,A.&Epitropakis,M.G.,2013. BenchmarkfunctionsforCEC'2013special session and competition on niching methods for multimodal function optimization. Technical report, RMIT University, Evolutionary Computation and Machine Learning Group, Australia.

[7] Jamil, M. & Yang, X-S., 2013. A literature survey of benchmark functions for global optimization problems. International Journal of Mathematical Modelling and Numerical Optimisation, vol. 4, n. 2, pp. 150-194.

[8] Im,C.H.,Kim,H.K.,Jung,H.K.&Choi,K.,2004. Anovelalgorithmformultimodalfunction optimization based on evolution strategy. IEEE Transaction on Magnetics, vol. 40, n. 2, pp. 1224-1227.

[9] Dieterich, J. M. & Hartke, B., 2012. Empirical review of standard benchmark functions using evolutionary global optimization. Applied Mathematics, vol. 3 n. 10A, pp. 1552-1564.

[10] Krishnanand, K. N. & D. G. Ghose, 2009. Swarm optimization for simultaneous capture of multiple local optima of multimodal functions. Swarm Intelligence, vol. 3, pp. 87-124.

Participação em Congressos, publicações e/ou pedidos de proteção intelectual:

Participação no CMAC e SBPO