

# Revisão de artigos 2

Jean Carlo Machado e Renato Bustamante

# Artigos e Temas

- ▶ Iztok Fister Jr., Xin-She Yang - A Brief Review of Nature-Inspired Algorithms for Optimization, University of Maribor, 2013 [1]
  - Classificação de algoritmos pela característica da fonte de inspiração na natureza.
- ▶ R. S. Parpinelli, H. S. Lopes, 2011, New inspirations in swarm intelligence: a survey, Int. J. Bio-Inspired Computation [2]
  - métodos, características e aplicações de algoritmos baseados em meta-heurísticas de enxame.
- ▶ Arpan Kumar Kar, 2016, Bio inspired computing - A review of algorithms and scope of applications, Indian Institute of Technology [3]
  - Comparativos entre algoritmos de otimização baseados na natureza.

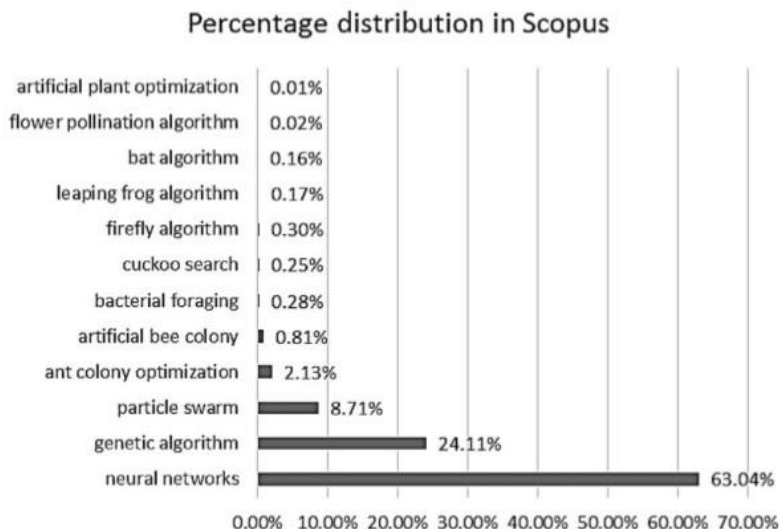
# Popularidade de algoritmos

"A good balance between exploitation and exploration may lead to the global optimally achievement."

Dentre os que tem bom balanço pode-se citar: PSO, evolução diferencial, busca do cuco, algoritmos de vaga-lume.

Alguns bons algoritmos não foram adotados pela comunidade: algoritmo de vespa(1991), algoritmo do tubarão(1998) ...

# Utilização de algoritmos na comunidade científica



**Fig. 2.** Search results in Scopus with algorithm names in title.

Figure 3: Utilização de algoritmos na comunidade científica

# Classificação

## Classificação pela fonte de inspiração:

- ▶ Baseados em inteligência de enxame:
  - ACO, PSO, FA (Firefly), CS (Cuckoo) ...
- ▶ Bio-inspirados, mas não baseados em enxame:
  - Differential Evolution, Flower Pollination ...
- ▶ Baseados em Física e Química:
  - Eletricidade, Gravidade, sistemas de Rios ...
- ▶ Outras fontes da Natureza
  - Social, Emocional ...

## Outras Formas de Classificação de Algoritmos :

- ▶ Trajetória vs população
- ▶ Baseados em atração vs não baseados em atração
- ▶ Baseados em regras vs baseados em equações

# Complexidades dos problemas pelo tempo

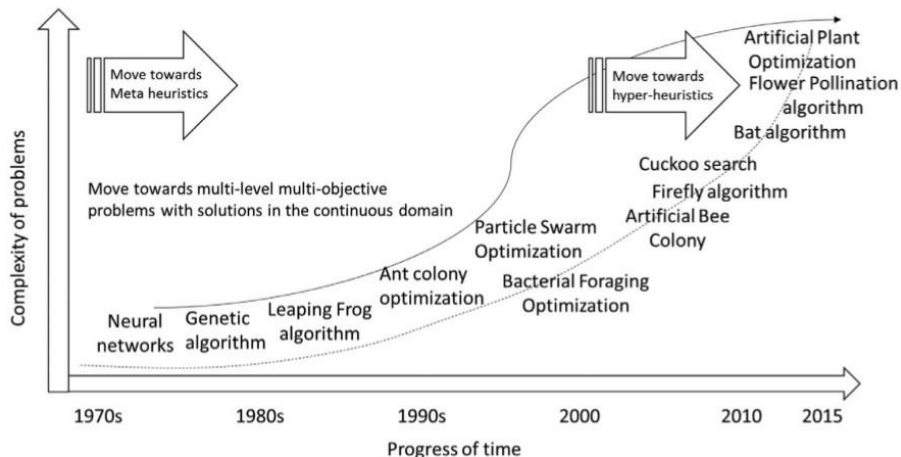


Figure 1:Complexidades dos problemas pelo tempo

# Nível de desenvolvimento de algoritmos

Categorization of algorithms based on scope for researchers.

Quadrant 1: Zone of theory development	Quadrant 2: Zone of applications
<ul style="list-style-type: none"><li>• Amoeba (Zhang et al., 2013)</li><li>• Artificial plant optimization (Cui &amp; Cai, 2013)</li><li>• Bean optimization (Zhang et al., 2010)</li><li>• Dove (Su et al., 2009)</li><li>• Eagle (Yang &amp; Deb, 2010)</li><li>• Fruit fly (Pan, 2012)</li><li>• Glow-worm (Krishnanand &amp; Ghose, 2005)</li><li>• Grey wolf algorithm (Mirjalili, Mirjalili, &amp; Yang, 2014)</li><li>• Krill-herd (Gandomi &amp; Alavi, 2012)</li><li>• Lion (Yazdani &amp; Jolai, 2015)</li><li>• Monkey (Mucherino &amp; Seref, 2007)</li><li>• Wolf (Liu et al., 2011)</li></ul>	<ul style="list-style-type: none"><li>• Bacterial foraging (Passino, 2002)</li><li>• Bat algorithm (Yang, 2010)</li><li>• Bee colony (Karaboga, 2005)</li><li>• Cuckoo search (Yang &amp; Deb, 2009)</li><li>• Firefly algorithm (Yang, 2009)</li><li>• Flower pollination (Yang, 2012)</li></ul>
Quadrant 3: Zone of rediscovery	Quadrant 4: Zone of commercialization
<ul style="list-style-type: none"><li>• Leaping Frog (Snyman, 1982)</li><li>• Shark (Hersovici et al., 1998)</li><li>• Wasp (Theraulaz et al., 1991)</li></ul>	<ul style="list-style-type: none"><li>• Ant colony optimization (Dorigo et al., 2006)</li><li>• Genetic algorithm (Holland, 1975)</li><li>• Neural networks (Grossberg, 1988)</li><li>• Particle swarm (Shi &amp; Eberhart, 1999)</li></ul>

Figure 2:Nível de desenvolvimento de algoritmos

# Algoritmos

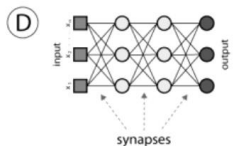
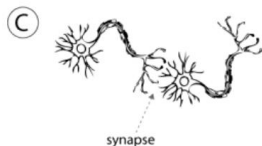
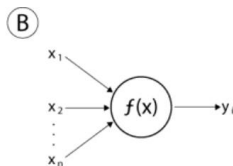
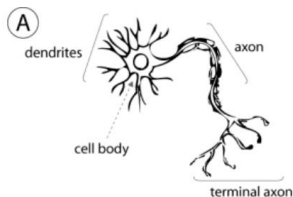


# Algoritmos

- ▶ Redes Neurais
- ▶ Genéticos
- ▶ Formiga
- ▶ Partículas
- ▶ Abelhas
- ▶ Bactérias
- ▶ Morcego
- ▶ Cuco
- ▶ Flores
- ▶ Sapo
- ▶ Slime mould
- ▶ Vaga-lume

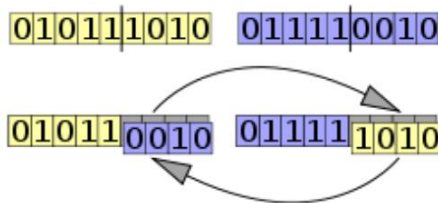
# Redes Neurais

- ▶ A rede de perceptron é a implementação mais simples.
- ▶ Pode ser utilizado para problemas lineares e não lineares.
- ▶ Black box
- ▶ <http://playground.tensorflow.org/>



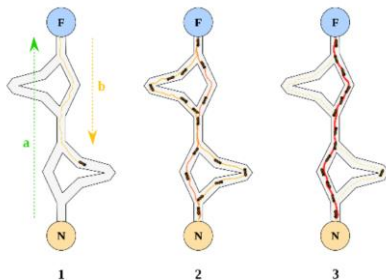
# Algoritmos Genéticos

- ▶ São bons para problemas combinatórios e não determinísticos.
- ▶ A eficiência para problemas com muitas dimensões tende a ser ruim.
- ▶ Já foi aplicado em agendamento de trabalhos, compressão de dados, gerenciamento econômico, teoria dos jogos, controle de satélite, etc.



# Algoritmos de formiga

- ▶ O tempo de resposta é razoável.
- ▶ Performance cai com o aumento da dimensionalidade.



- ▶ Aplicações: Escalonamento de Rotinas, Caixeiro Viajante, Rastreamento de objeto, Roteamento de Pacotes.

# Enxame de partículas (PSO)

- ▶ Relativamente simples de implementar.
- ▶ Foi utilizado em problemas de agendamento, e de maximização/minimização.



# Algoritmos de Abelha (2005)

- ▶ Bom para resolver problemas evolucionários.
- ▶ A intensificação é controlada por meios estocásticos e ávidos.
- ▶ Tem melhor performance do que algoritmos de Monte Carlo, Genéticos e ACO.
- ▶ Já foram utilizados em alocação de tarefas e distribuição de energia.
- ▶ Aplicações:
  - tree-SAT problem optimization.
  - utilizado em programação estocástica e dinamica.

## Forrageamento de abelhas

- Local: alocação de abelhas vizinhas para boa fonte de comida
- Global: abelhas scout buscam novas fontes de alimento

## Acasalamento de abelhas

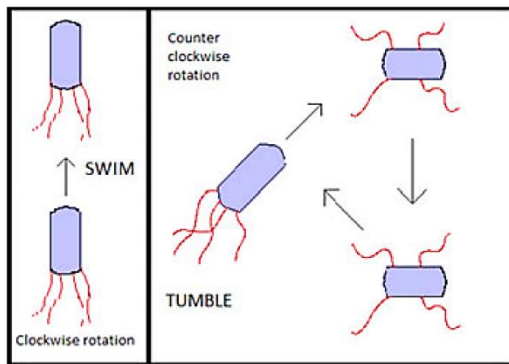
- Local: ninhadas de abelhas procriadas pela rainha
- Global: Criacao da Spermatheca

# Busca de alimentos bacterial

As bactérias se movem rotacionando-se e tombando.

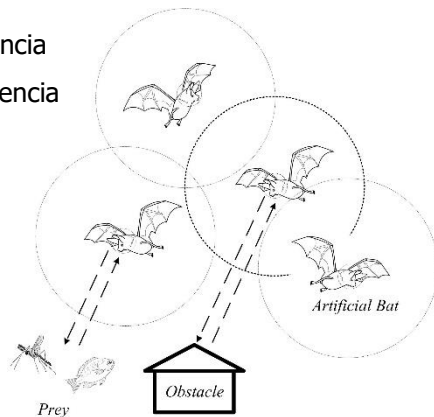
O objetivo da busca de alimentos pode ser racionalizado como: maximizar a entrada de energia por unidade de tempo procurando alimento.

O algoritmo apresenta baixa convergência em tarefas complexas.



# Algoritmos de Morcego (2010)

- ▶ Serve para problemas contínuos. Bom para otimizações com restrições
  - ▶ Utilizado apenas uma vez, para o problema do caixeiro viajante.
- 
- Local: baixo ruído e alta frequência
  - Global: alto ruído e baixa frequência





# Busca do Cuco

Bom para problemas com restrições não lineares complexas.

Já aplicado em:

- ▶ Procura de heurística
- ▶ Maximização de problemas opostos
- ▶ Problemas de alocação multi objetivo

# Polinização de flores

- ▶ Considerado como não SI (Swarm Intelligence) based
  - imita a polinização das flores e a consistência da flor associada a alguns insetos polinizadores
- ▶ Bom para problemas de otimização global.

## Salto do Sapo (2000)

- ▶ Combina os benefícios de algoritmos sociais e meméticos.
  - ▶ Bom para encontrar ótimos locais predominantes.
- Especialmente bom quando a função local está afetada por ruídos locais.

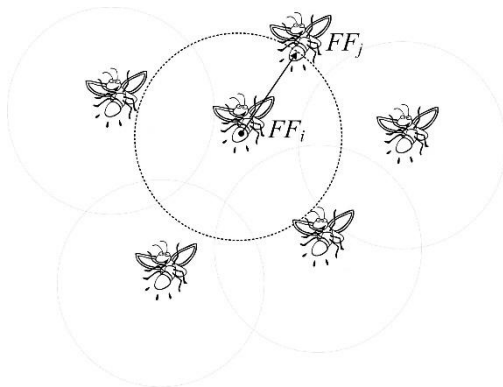
## Slime mould

- ▶ Foi aplicado apenas no artigo original

# Vaga-lume

- ▶ Vaga-lume é atraído por vaga-lumes
- ▶ A atração é proporcional a luminosidade e a distancia entre eles
- ▶ a luminosidade é determinada pela função objetivo

Lida com funções multi modais mais eficientemente do que outros algoritmos de enxame.



# Infestação de Baratas

## RIO

- ▶ Procuram locais escuros
- ▶ Socializam com baratas próximas
- ▶ periodicamente saem para procurar comida

# Mosquito

## MHSA

- ▶ Procuram por CO<sup>2</sup> ou aromas
- ▶ Se aproximam de locais com alta concentração de aroma (CO<sup>2</sup>)
- ▶ Pousam quando sentem o calor do hospedeiro

**Table 1** Meta-heuristics summary

<i>Algorithm</i>	<i>Inspiration</i>	<i>First applied to...</i>	<i>Mechanism of exploitation</i>	<i>Mechanism of exploration</i>	<i>Communication model</i>
BA	Bee foraging	Continuous optimisation	Neighbourhood search in good food sources	Random search of scout bees	Broadcast-like
ABC	Bee foraging	Continuous optimisation	Neighbourhood search carried by employed and onlooker bees	Random search of scout bees	Broadcast-like
MBO	Bee mating	Discrete optimisation	Neighbourhood search in queens and broods carried by workers	Spermatheca creation	Direct
BFO	Bacterial foraging	Continuous optimisation	Chemotaxis and reproduction steps	Elimination-dispersal step	Direct
GSO	Firefly bioluminescence	Continuous optimisation	Glow-worm position update	Find neighbour phase dictated by sensor range	Broadcast-like
FA	Firefly bioluminescence	Continuous optimisation	Firefly movement according to attractiveness	Random move of the best firefly	Broadcast-like
SMOA	Amoebae foraging	Continuous optimisation	Vegetative state	Dispersal state	Stigmergic
RIO	Cockroaches infestation	Continuous optimisation	Find friend phase	Find food phase	Broadcast-like
MHSA	Mosquito foraging	Discrete optimisation	Host attraction	Mosquitoes interaction	Broadcast-like
BA	Bat echolocation	Continuous Optimisation	Low loudness and high pulse rate values	High loudness and low pulse rate values	Broadcast-like

# Futuras pesquisas da CN

- ▶ Redução de parâmetros.
- ▶ Co evolução
- ▶ Novas inspirações

# Conclusão

O artigo [1] trouxe uma classificação interessante dos algoritmos pela característica da fonte de inspiração na natureza, no entanto esta classificação pode se tornar ultrapassada a medida que novos algoritmos são criados seguindo hiper-heurística e associando mais de uma meta-heurística.

O artigo [3] pareceu ter sido feito por várias pessoas, com pontos chave sendo feitos repetidamente, como se recortados. Mas aplicou uma metodologia interessante, selecionando algoritmos de otimização com pelo menos 15 referências.



Obrigado