

IA013 (2s2017) – EFC 2

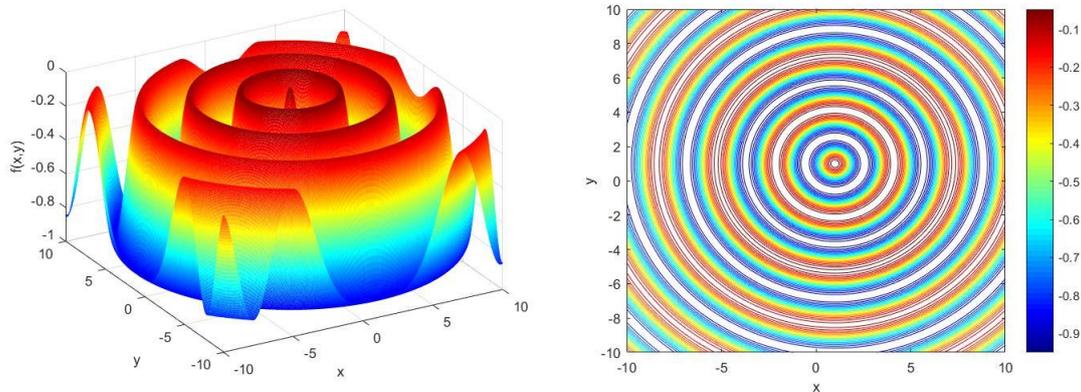
Realizar em grupos de 2 ou 3 alunos(as)
Data de entrega e pontuação junto a cada questão

Questão 1) Otimização em Espaços Contínuos com Enxame de Partículas e Algoritmo Imuno-Inspirado – Entrega: 02/11/2017 (1,0 ponto)

Considere a função de Schaffer modificada:

$$f(x, y) = -0,5 - \left(\frac{\sin^2(\sqrt{(x-1)^2 + (y-1)^2}) - 0,5}{(1 + 0.001[(x-1)^2 + (y-1)^2])^2} \right),$$

com x e y pertencentes ao intervalo $[-10,10]$. O máximo global desta função se encontra no ponto $\mathbf{x}^* = (1,1)$, no qual a função assume o valor $f(\mathbf{x}^*) = 0$. A figura abaixo apresenta a superfície da função juntamente com as curvas de nível.



O objetivo deste exercício é realizar uma análise comparativa do comportamento e dos resultados de dois algoritmos de otimização: (1) PSO (*particle swarm optimization*) e (2) um sistema imunológico artificial (e.g., o CLONALG ou a *opt-aiNet*).

No caso do PSO, deve-se trabalhar com **três** configurações diferentes de vizinhança: (1) totalmente conectada, (2) anel e (3) baseada em uma rede complexa.

Sugestão: no caso da rede complexa, adote um dos procedimentos discutidos em aula para a construção de uma rede *scale-free*, como, e.g., o método de Barabási & Albert baseado na ideia de *preferential attachment*.

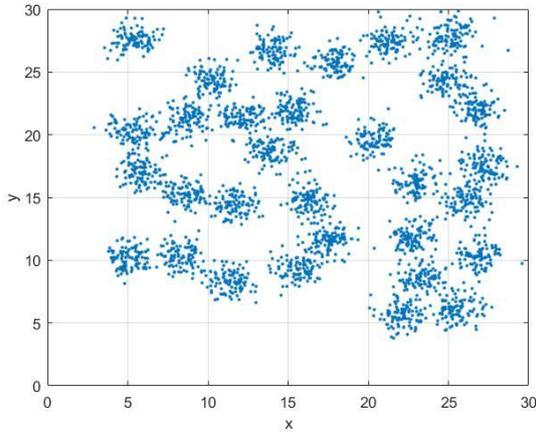
Para ambos os algoritmos, analise o desempenho em termos: (1) da melhor solução encontrada, e (2) da distribuição da população final sobre a superfície/curvas de nível da função objetivo, apresentando os resultados obtidos em 10 execuções independentes de cada método. Mostre, também, para uma única execução de cada algoritmo (e versão), as curvas de *fitness* médio e do melhor indivíduo presente na população, comentando o que foi observado.

Não deixe de apresentar a estrutura de cada algoritmo desenvolvido, assim como o modo pelo qual cada operação foi implementada (e.g., mutação por afinidade). Justifique as escolhas feitas para os parâmetros de cada algoritmo.

Questão 2) Agrupamento de dados – Entrega: 09/11/2017 (1,0 ponto)

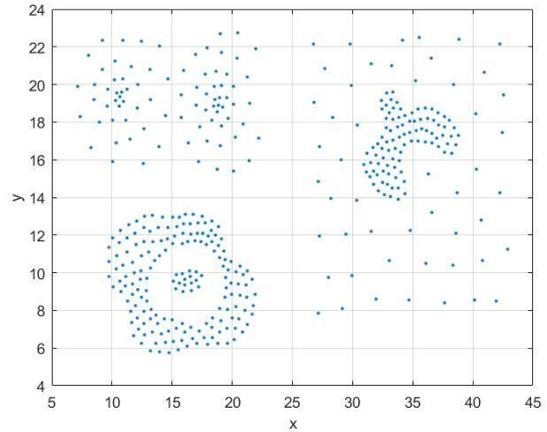
Objetivo: aplicar dois algoritmos imuno-inspirados, a saber, *aiNet* e ARIA, para realizar tarefas de agrupamento de dados. Dois conjuntos de dados, ilustrados na figura abaixo, serão abordados.

d31



(a) Extraído de C.J. Veenman, M.J.T. Reinders, E. Backer, “A maximum variance cluster algorithm”. *IEEE Trans. Pattern Analysis and Machine Intelligence* 2002. 24(9): p. 1273-1280.

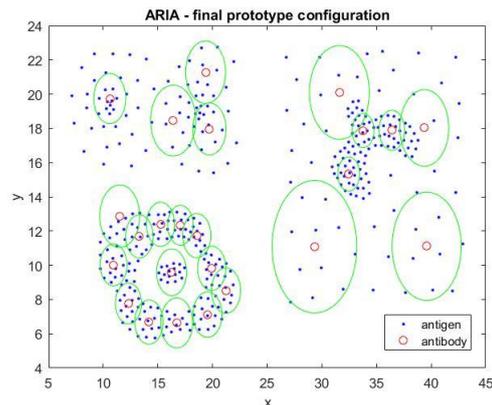
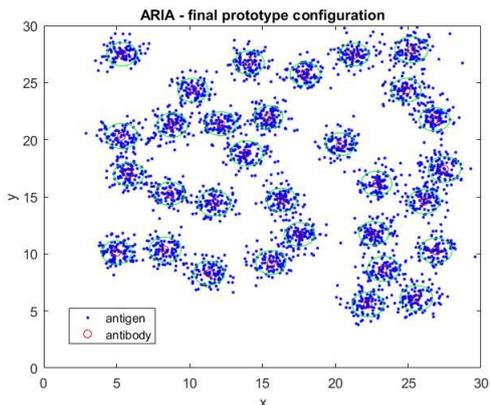
Zhan’s compound



(b) Extraído de C.T. Zahn, “Graph-theoretical methods for detecting and describing gestalt clusters”. *IEEE Transactions on Computers*, 1971. 100(1): p. 68-86.

Implemente os algoritmos *aiNet* e ARIA, apresentando uma breve descrição das operações envolvidas em cada passo. Indique e justifique os valores escolhidos para os principais parâmetros destes algoritmos. Exiba a distribuição dos anticorpos, juntamente com o respectivo raio, em diferentes momentos (iterações) de cada algoritmo.

Exemplos de resultados tipicamente obtidos com o algoritmo ARIA são mostrados a seguir:



Utilize, então, o número de anticorpos presentes na população final dos algoritmos *aiNet* e ARIA como valor de k (número de centros) do algoritmo k -means. Mostre o resultado obtido por este método e compare-o com aqueles associados aos algoritmos imuno-inspirados.

Finalmente, obtenha a árvore geradora mínima (*minimum spanning tree*) para as populações finais dos três algoritmos. Comente os resultados.

Leituras de apoio:

- [Texto introdutório ao algoritmo ARIA](#)
- F. O. de França, G. P. Coelho, P. A. D. Castro, F. J. Von Zuben, “Conceptual and Practical Aspects of the aiNet Family of Algorithms”, International Journal of Natural Computing Research, vol. 1, no. 1, 2010 (disponível na página do curso).

Questão 3) Colônia de Formigas – Entrega: 09/11/2017 (1,0 ponto)

Implemente uma versão do algoritmo de colônia de formigas (*Max-Min*) para o problema do caixeiro viajante. Apresente a estrutura básica do algoritmo, junto com os detalhes de cada operação.

Utilize as duas instâncias de 100 cidades fornecidas junto com o *toolbox* de mapa de Kohonen, disponibilizado para o EFC1, para testar o algoritmo implementado. Faça, também, uma comparação entre os resultados obtidos com o ACO e com o mapa de Kohonen, considerando **cinco execuções** independentes de cada técnica.

Leitura sugerida: [T. Stützle, M. Dorigo, “ACO Algorithms for the Traveling Salesman Problem”, 1999.](#)