Programa de Pós-graduação em Engenharia Mecatrônica - Universidade de Brasília
Disciplina: Sistemas Bioinspirados Aplicados a Engenharia (período 2017.2)
Professores: Daniel M. Muñoz Arboleda; Carlos H. Llanos, e-mail: damuz@unb.br; llanos@unb.br
Monitor: Carlos Eduardo da Silva Santos, e-mail: carlosedu@ifto.edu.br

**Segunda Lista de Exercícios**
**Algoritmos Firefly e Differential Evolution e modificações**
**Comparação de Algoritmos Bioinspirados**
**Data de entrega (27 de Outubro de 2017)**

**Nota**: A lista é individual. Usar template LaTeX da IEEE conference. Enviar via moodle os arquivos Matlab e o relatório em PDF.

**Primeira Questão:** Implementar no Matlab os algoritmos FA e DE clássicos para minimizar as funções *benchmark Griewank, Rastrigin, Rosenbrock, Ackley, Schwefel e Michalewicz*. No segunte link pode-se conferir a posição do mínimo global de cada função e o respectivo valor da função custo:
http://www-optima.amp.i.kyoto-u.ac.jp/member/student/hedar/Hedar_files/TestGO_files/Page364.htm

$$Griewank: f_1(\vec{x}) = 1 + \frac{1}{4000}\sum_{i=1}^{N} x_1^2 - \prod_{i=1}^{N}\cos\left(\frac{x_i}{\sqrt{i}}\right); x_i \in [-512, 512]$$

$$Rastrigin: f_2(\vec{x}) = \sum_{i=1}^{N}\left(x_i^2 - 10\cos(2\pi x_i) + 10\right); x_i \in [-8.0, 8.0]$$

$$Rosenbrock: f_3(\vec{x}) = \sum_{i=1}^{N/2} 100\cdot\left(x_{2i} - x_{2i-1}^2\right)^2 + \left(1 - x_{2i-1}\right)^2; x_i \in [-8.0, 8.0]$$

$$Ackley: f_4(\vec{x}) = -20\exp\left(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{N} x_i^2}\right) - \exp\left(\frac{1}{n}\sum_{i=1}^{N}\cos(2\pi x_i)\right) + 20 + e; x_i \in [-32.768, 32.768]$$

$$Schwefel: f_5(\vec{x}) = 418.9829N - \sum_{i=1}^{N} x_i \sin\left(\sqrt{\left(x_i\right)}\right); x_i \in [-500, 500]$$

$$Michalewicz: f_6(\vec{x}) = -\sum_{i=1}^{N}\sin(x_i)\sin\left(\frac{ix_i^2}{\pi}\right)^{2m}; x_i \in [0, \pi]; m = 10$$

Configure os algoritmos segundo as seguintes instruções:
1) Usar uma população do enxame *S*=20.
2) Usar as mesmas dimensionalidades usadas nos algoritmos da lista 1 (três dimensionalidades)
3) Repetir cada experimento 32 vezes, inicializando o enxame em diferentes posições aleatórias.
4) Configure os parâmetros dos algoritmos da seguinte maneira:

- Usar como critério de parada o número máximo de iterações.
- Ajuste o número máximo de iterações conforme a complexidade do problema.
- Ajuste o parâmetro *gamma*=0.85 (coeficiente de absorção do FA).
- Ajuste o parâmetro *alpha_0* (coeficiente de aleatoriedade inicial do FA) entre os valores [0.6 e 0.9]
- Ajuste o parâmetro *delta*=0.99 (coeficiente de redução aleatória do FA)
- Ajuste o parâmetro *F* (fator de mutação do DE) entre os valores [1.0 e 1.35]
- Ajuste o parâmetro *CR*=0.95 (taxa de *crossover* do DE)
- *threshold* = 0.01 para todas as funções *benchmark*, exceto para a função *Michalewicz* cujo *threshold* deve ser achado mediante uma regra de três simples. Neste primeiro exercício não vamos usar o *threshold* como critério de parada. Apenas será usado para verificar se o algoritmo alcançou um valor mínimo desejado para a função custo (*goal*).

5) Para cada experimento vamos usar o melhor valor da função custo (valor da função custo após as 1000 iterações) para calcular a média, mediana, desvio padrão, valor mínimo e número de acertos (*goals*) entre os 32 experimentos. Apresentar a solução do problema de otimização (posição do ponto mínimo encontrado),

entre os 32 experimentos. Observe-se que dos 32 experimentos, aquele que apresente o valor mínimo da função custo representa a melhor solução do problema. Para cada função *benchmark* deve-se apresentar a seguinte tabela contendo os resultados estatísticos. Não é necessário apresentar as curvas de convergência.

Tabela 1. Algoritmo ____. Resultados de convergência para a função ________ (32 runs).

| | | **Média** | **Mediana** | **Mínimo** | **Desvio Padrão** | ***goals*/32** |
|---|---|---|---|---|---|---|
| *S*=20 | *N*= | | | | | |
| | *N*= | | | | | |
| | *N*= | | | | | |

6) Realize as comparações de desempenho entre os algoritmos PSO, ABC, FA e DE usando testes não paramétricos para *S*=20 e o problema de maior dimensionalidade. Para isto aplique a seguinte metodologia:

A) Verifique se os resultados de cada algoritmo seguem uma distribuição normal usando o teste de Kolmogorov-Smirnov.
B) Aplique o teste de Kruskal-Wallis para determinar se os resultados dos algoritmos seguem distribuições de probabilidade com medianas iguais.
C) Aplique o teste de Wilconxon entre o algoritmo com melhor mediana e os outros algoritmos, verificando se os mesmos seguem distribuições de probabilidade diferentes.
D) É possível afirmar qual é o melhor algoritmo? Apresente conclusões.
E) **Para todos os testes use um nível de confiança de 95%.**

**Segunda Questão:** Implementar no Matlab os algoritmos PSO e DE com as variações de Aprendizagem por Oposição (OBL) e atrativo-repulsivo (AR) para minimizar as funções *benchmark Griewank, Rastrigin, Rosenbrock, Ackley, Schwefel e Michalewicz*

Configure os algoritmos usando *S*=20 (20 partículas) e *N* sendo o caso de maior dimensionalidade usado na primeira questão. Para o PSO utilize os mesmos parâmetros de configuração usados na lista de exercícios 1 e para o DE use os mesmos parâmetros de configuração da primeira questão da presente lista de exercícios. Adicionalmente, utilize os seguintes parâmetros:

1) Na estratégia OBL use um *limit*=40 (aplica-se o OBL após *limit* iterações sem melhorar o valor do *fitness*).
2) Na estratégia OBL calcule o número oposto da seguinte maneira: $x$=-$x$+0.1*rand(), isto permitirá fazer uma busca local aleatória entorno da solução oposta.
3) Na estratégia OBL aplique o número oposto para dimensões aleatórias.
4) Na estratégia atrativo-repulsivo deve ser aplicada em função da diversidade do enxame, a qual é calculada através da distância euclidiana entre as partículas.
5) Estabeleça valores de *threshold* mínimo e máximo para a diversidade do enxame.
6) Repetir cada experimento 32 vezes, inicializando o enxame em diferentes posições aleatórias.
6) Deve-se apresentar as seguintes tabelas contendo os resultados estatísticos. Apresentar a solução do problema de otimização (posição do ponto mínimo encontrado), entre os 32 experimentos. Observe-se que dos 32 experimentos, aquele que apresente o valor mínimo da função custo representa a melhor solução do problema. Não é necessário apresentar as curvas de convergência.

Tabela 2. Resultados de convergência para a função ________ (32 runs). *S*=20, *N*=

| **Algoritmo** | **Média** | **Mediana** | **Mínimo** | **Desvio Padrão** | ***goals*/32** |
|---|---|---|---|---|---|
| O-PSO | | | | | |
| AR-PSO | | | | | |
| O-DE | | | | | |
| AR-DE | | | | | |

7) Realize as comparações de desempenho entre os algoritmos O-PSO, AR-PSO, O-DE e AR-DE usando testes não paramétricos para *S*=20 e o problema de maior dimensionalidade. Para isto aplique a seguinte metodologia:

A) Verifique se os resultados de cada algoritmo seguem uma distribuição normal usando o teste de Kolmogorov-Smirnov.
B) Aplique o teste de Kruskal-Wallis para determinar se os resultados dos algoritmos seguem distribuições de probabilidade com medianas iguais.
C) Aplique o teste de Wilconxon entre o algoritmo com melhor mediana e os outros algoritmos, verificando se os mesmos seguem distribuições de probabilidade diferentes.
D) É possível afirmar qual é o melhor algoritmo? Apresente conclusões.
E) **Para todos os testes use um nível de confiança de 95%.**

Bom trabalho !