

**Instituto Tecnológico de Aeronáutica - ITA**  
**Inteligência Artificial para Robótica Móvel - CT-213**  
**Aluno: Danilo de Farias Matos**

**Relatório do Laboratório 5 - Estratégias Evolutivas**

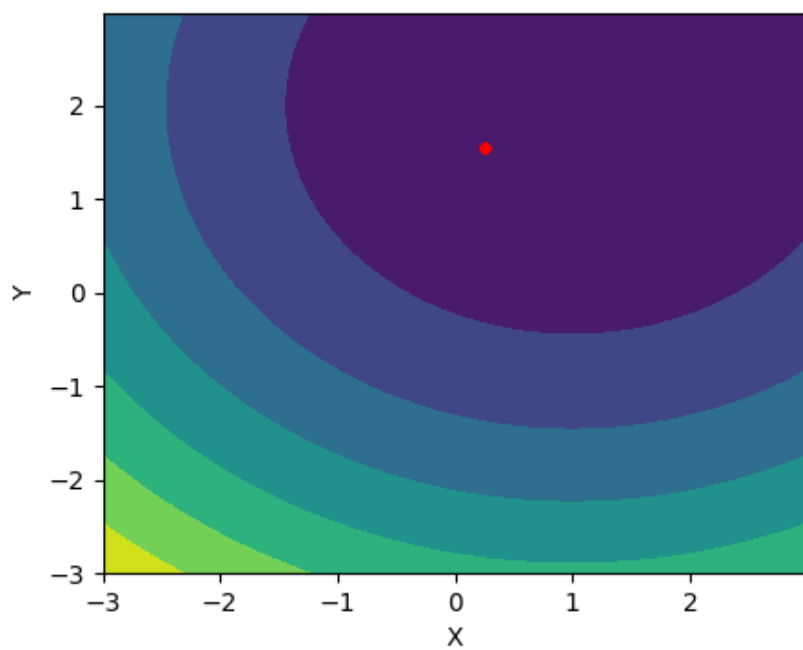
**1. Breve Explicação em Alto Nível da Implementação**

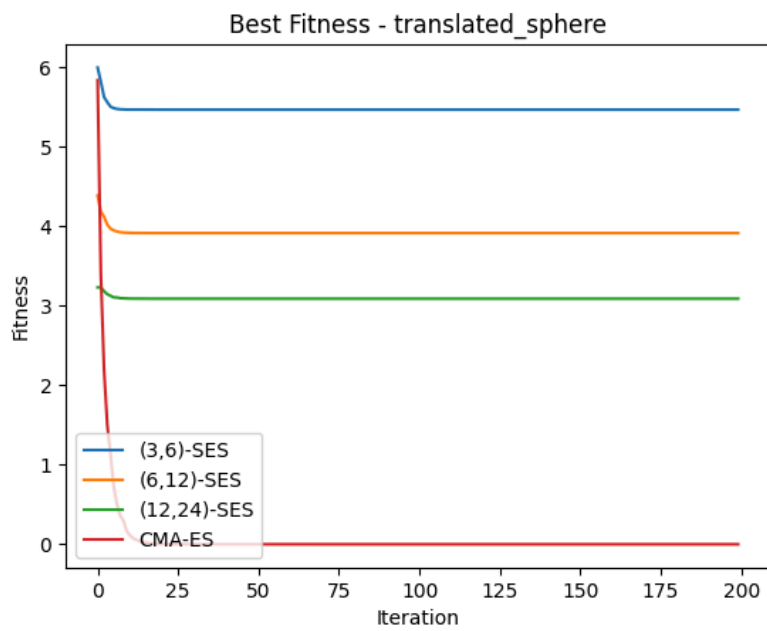
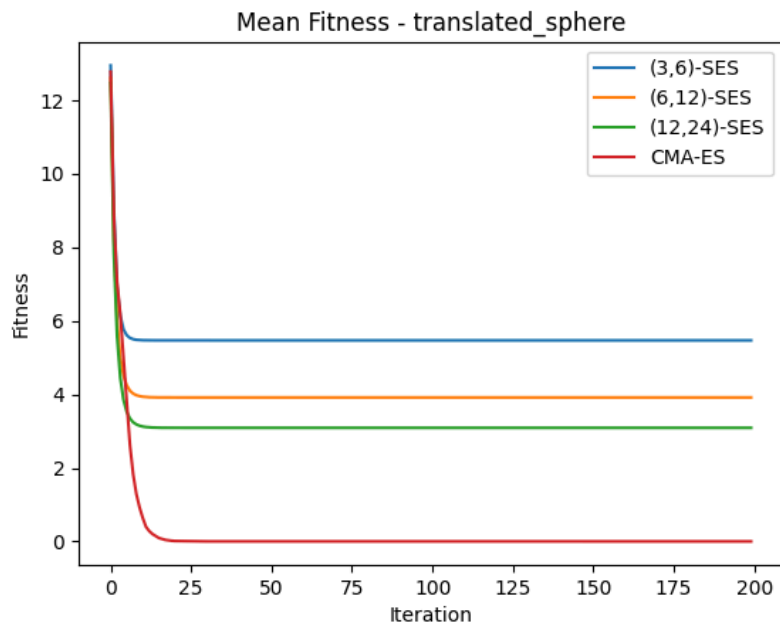
**1.1. Estratégia Evolutiva Simples**

Iniciei a implementação ordenando de maneira crescente os melhores valores (mais baixos) de cada um dos samples, depois disso, selecionei os pais para serem a representação dos 'n' melhores exemplos. Na sequência, calculei a média e a covariância novamente com base apenas nos pais selecionados e atualizei os samples com base nesses novos valores.

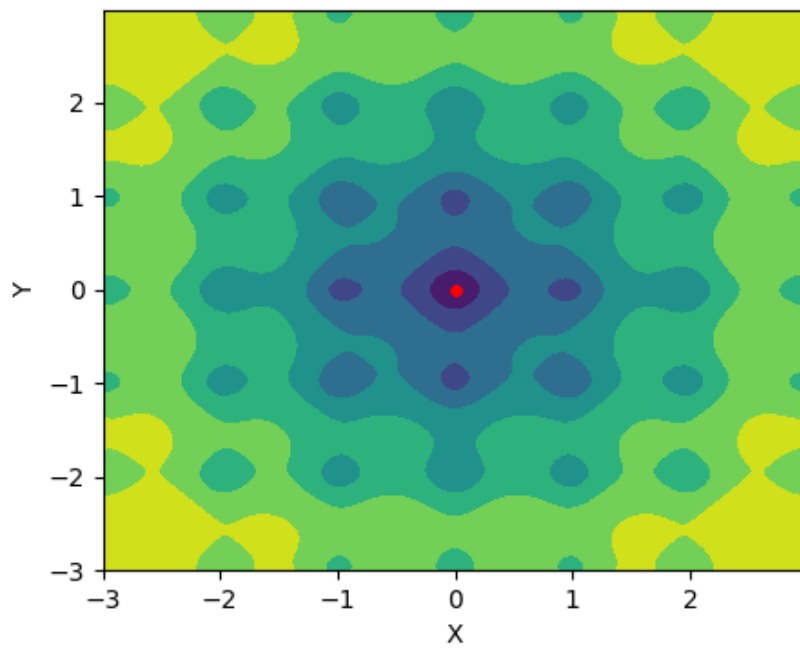
**2. Figuras Comprovando Funcionamento do Código**

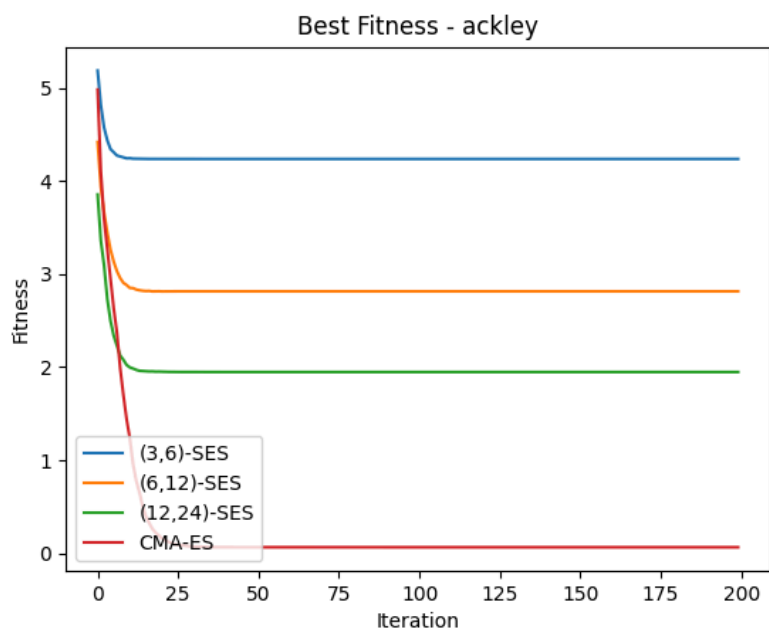
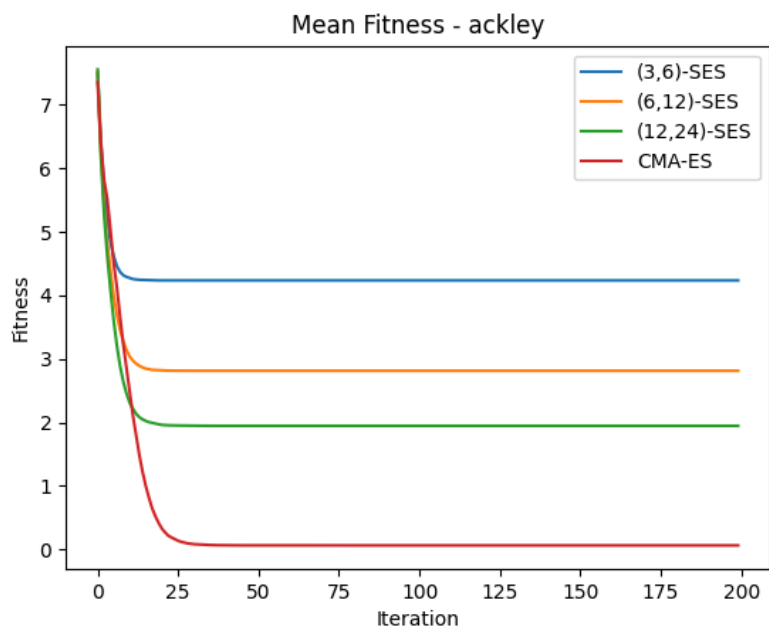
## 2.1. Função *Translated Sphere*



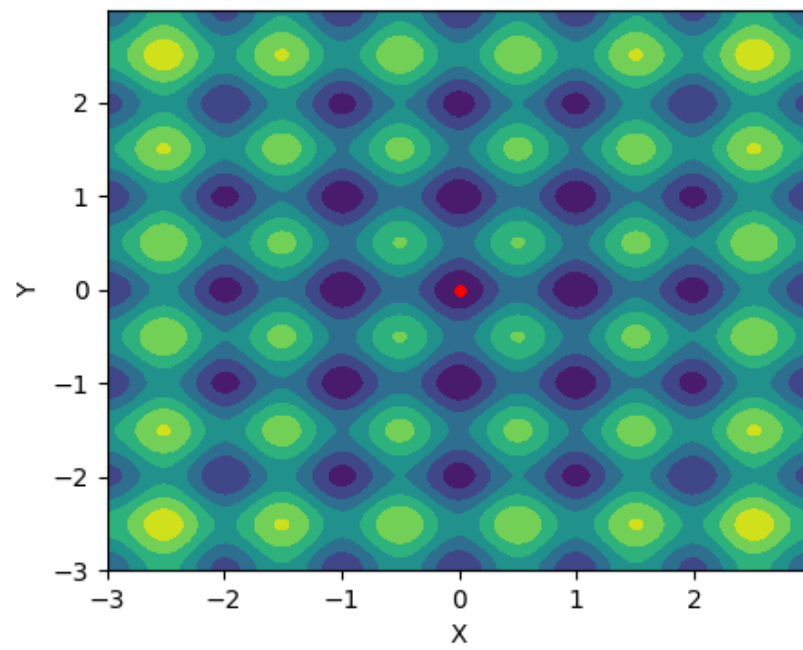


## 2.2. Função Ackley

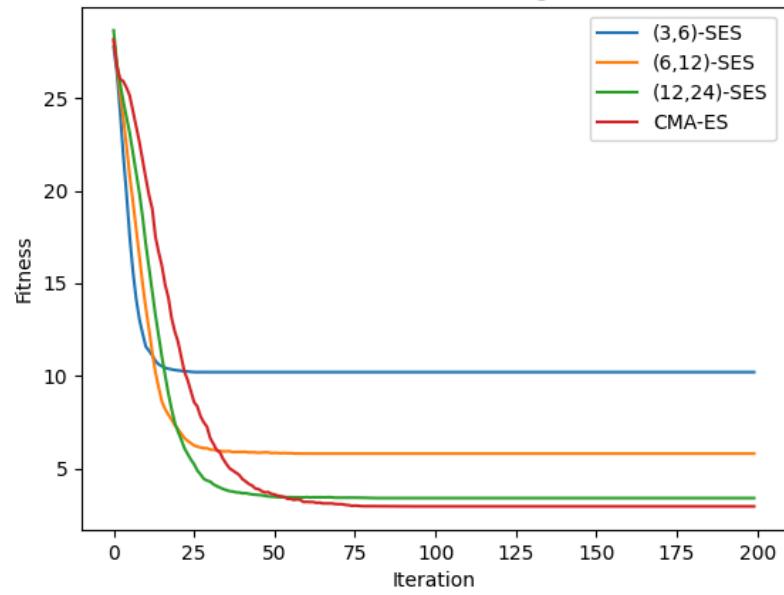




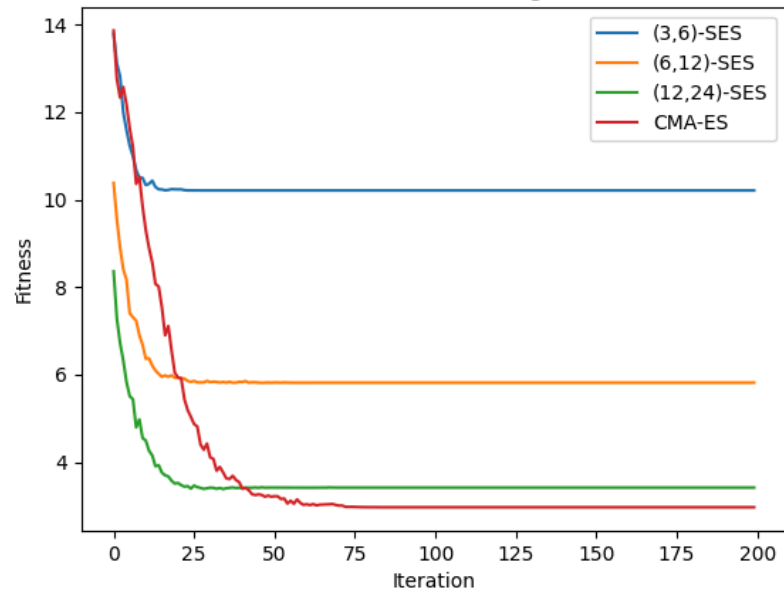
### 2.3. Função Rastrigin



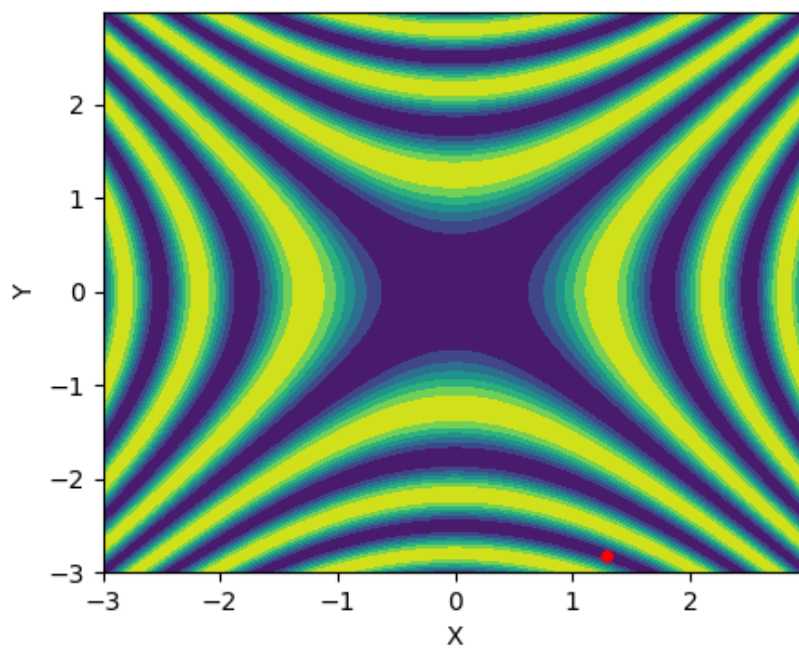
Mean Fitness - rastrigin



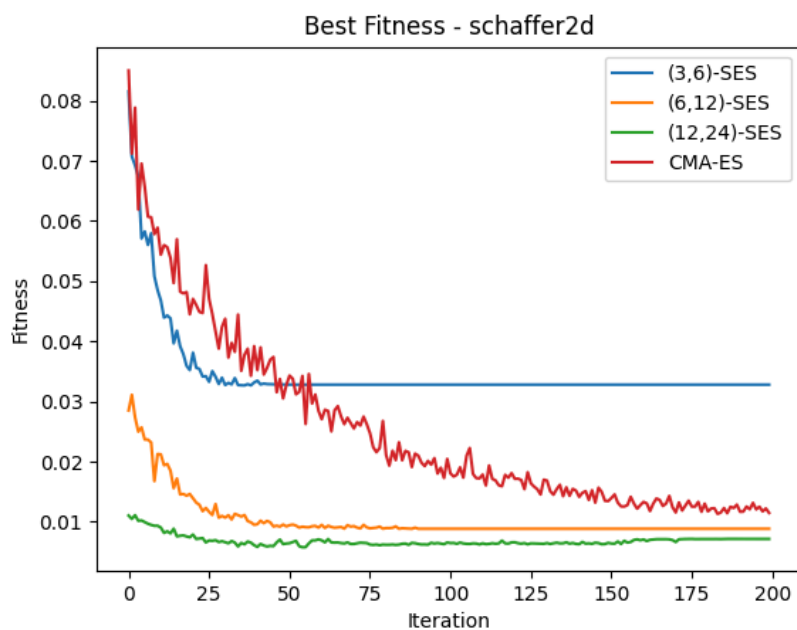
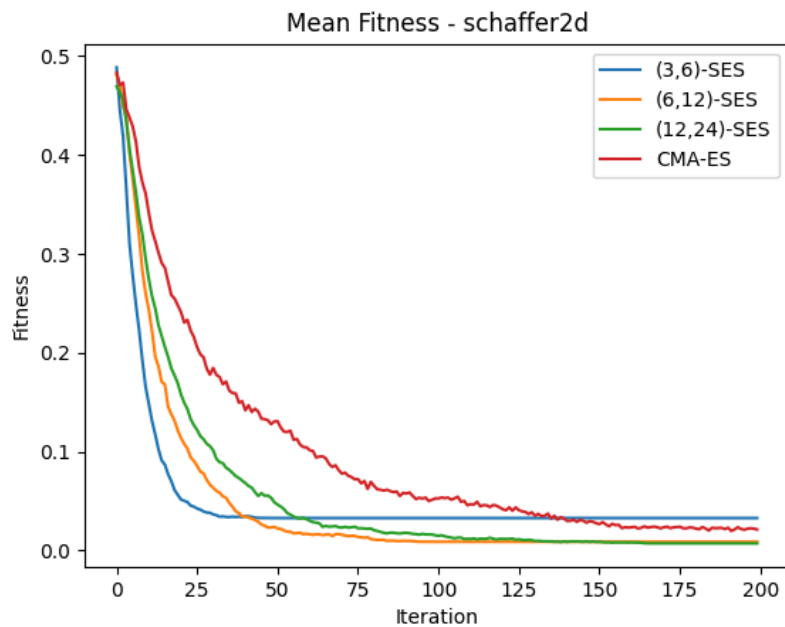
Best Fitness - rastrigin



## 2.4. Função Schaffer







### 3. Discussões e Conclusões

Sugestão: cerca de uma página.

- Por que você acha que os resultados são diferentes para cada função?
- Comente de forma sucinta sobre os resultados para cada um dos algoritmos e das funções, principalmente sobre questões como convergência, incluindo sobre convergência para mínimo local.

- A primeira coisa que pude observar é que a estratégia evolutiva simples sempre melhorou com o uso de uma população maior e “seleção de pais” maior, os gráficos demonstram que o aumento desses valores de hiperparâmetros, em todos os casos, foi benéfico para se aproximar mais do mínimo geral.
- Outra situação interessante de se mencionar é que a Estratégia evolutiva simples conseguia efetuar quase o triplo de iterações por segundo em comparação com a CMA-ES, indicando que ela é provavelmente mais eficiente do ponto de vista do custo computacional.
- Acredito que a diferença ocorre principalmente em virtude da forma como as funções fazem o custo mudar, por exemplo, a translated sphere deixa uma mudança mais sutil quando os valores vão se aproximando do mínimo global, o que, aparentemente, fez com que a estratégia evolutiva simples ficasse presa em um mínimo local mais facilmente. Já a função Schaffer fez com que o CMA-ES não encontrasse tão bem assim o mínimo global e demorasse um pouco mais de iterações para convergir, provavelmente porque ela não recompensa muito bem algoritmos que utilizam mais “exploration” em lugar de “exploitation” .
- Abaixo segue uma análise que fiz sobre cada função:
  - Translated Sphere
    - Os dois algoritmos precisaram de poucas iterações para convergir, mas somente o CMA-ES conseguiu alcançar, de fato, o mínimo global, ficando o SES preso em um mínimo local mais distante do global.
  - Ackley
    - Também fez com que fosse necessário poucas iterações para encontrar os mínimos e o SES se manteve mais distante que o CMA-ES do mínimo global.
  - Rastrigin
    - Já esta função demandou um pouco mais de iterações dos dois algoritmos para encontrar um ponto de mínimo, foi possível observar também que o SES com população 24 e 12 pais ficou bem próximo do valor de mínimo encontrado pelo CMA-ES e que ambos se aproximaram bastante do mínimo global.
  - Schaffer2d
    - Essa foi a função que foi mais interessante pra mim, porque demonstrou que o CMA-ES pode ter mais dificuldade em determinadas situações demandando um maior número de iterações para encontrar um mínimo e, ainda, ficar preso em um mínimo local pior que a SES.