

ELE083 Computação Evolucionária

Trabalho Prático - Makespan

Davi Pinheiro Viana - 2013029912

Rafael Carneiro de Castro - 2013030210

Minas Gerais, Brasil

Keywords: Makespan, Computação Evolucionária, Evolução Diferencial

1. Introdução

O trabalho final da disciplina de Computação Evolucionária consiste no emprego de uma das técnicas estudadas em sala de aula no decorrer do semestre para a solução de um problema de Engenharia. Tal problema é *Job Shop Scheduling Problem* (JSSP).

O *Job Shop Scheduling Problem* é um problema clássico de otimização combinatória e possui diversas aplicações nas indústrias e empresas. Seu objetivo é obter uma sequência de tarefas a serem executadas de forma a maximizar a utilização dos recursos disponíveis. Por recursos, entende-se como máquinas, pessoas, ou ambos.

A nível de contexto, imagina-se uma linha de produção com X etapas, que devem ser executadas em ordem. Tem-se como entrada um arquivo texto contendo Y linhas, que representam Y pedidos, com X números inteiros em cada linha. Estes números representam o tempo que o pedido daquela linha vai gastar em cada etapa. O objetivo é retornar uma sequência de produção dos pedidos em tempo ótimo, de forma que o conjunto de pedidos será entrega o mais rápido possível.

Para este trabalho, considera-se que a linha de produção tem três etapas. Sendo assim, os arquivos de entrada contêm três colunas.

2. O Algoritmo da Evolução Diferencial

A seguir, na Figura 1, pode ser visto o pseudo-código do algoritmo de evolução diferencial implementado neste trabalho.

Este algoritmo foi implementado no MatLAB, e pode ser visto nos arquivos *lab3_peaks* e *lab3_rastrigin* anexos junto a este relatório.

3. Testando com as funções *peaks* e *rastrigin*

As funções *peaks* e *rastrigin* foram disponibilizadas e foram utilizadas para se testar o algoritmo da Evolução Diferencial. Para a função *peaks*, as variáveis x_1 e x_2 estão ambas no intervalo $[-3; 3]$, e tem mínimo global em $f(x^*) = -6.5511$, onde $x^* = [0.228; -1.625]$. Já para a função *rastrigin*, as variáveis x_1 e x_2 estão ambas no intervalo $[-2; 2]$, e tem mínimo global em $f(x^*) = -20$, onde $x^* = [0; 0]$. O algoritmo foi executado para a otimização de ambas as funções, e algumas iterações podem ser vistas nas Figuras 2 e 3. O mínimo global da função *peaks* foi bem aproximado pelo algoritmo em estudo a partir da geração (iteração) 40. Já para a função *rastrigin*, o mínimo global foi bem aproximado a partir da geração 50.



Figura 1: Pseudo-código implementado

4. Conclusão

A partir dos resultados obtidos, pode-se concluir que para ambas as funções testadas o algoritmo da Evolução Diferencial obteve resultados bem precisos, e com relativamente poucas iterações do código. O algoritmo é bem intuitivo e foi de simples implementação. Conclui-se assim que os objetivos propostos pela tarefa foram cumpridos.

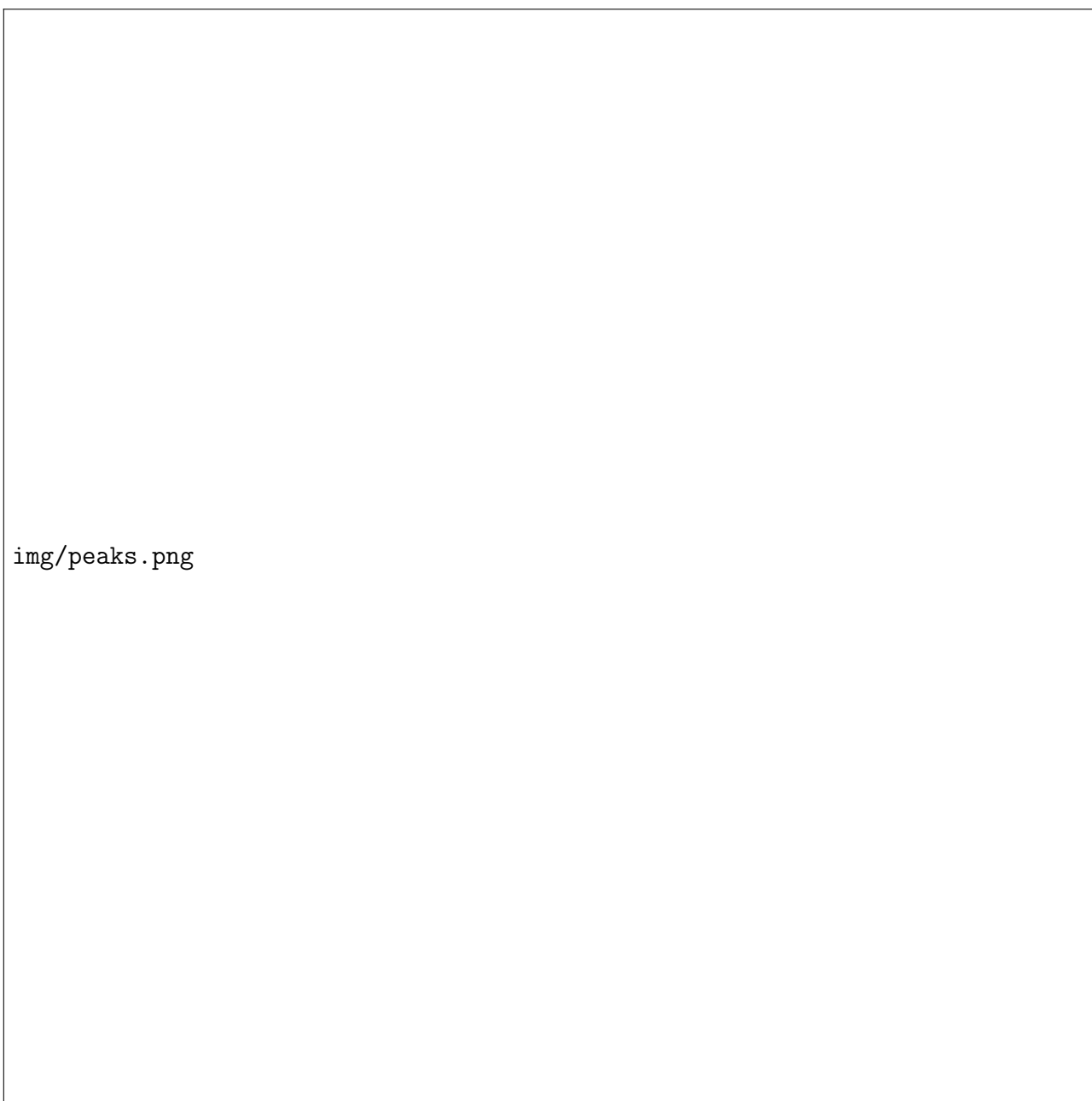
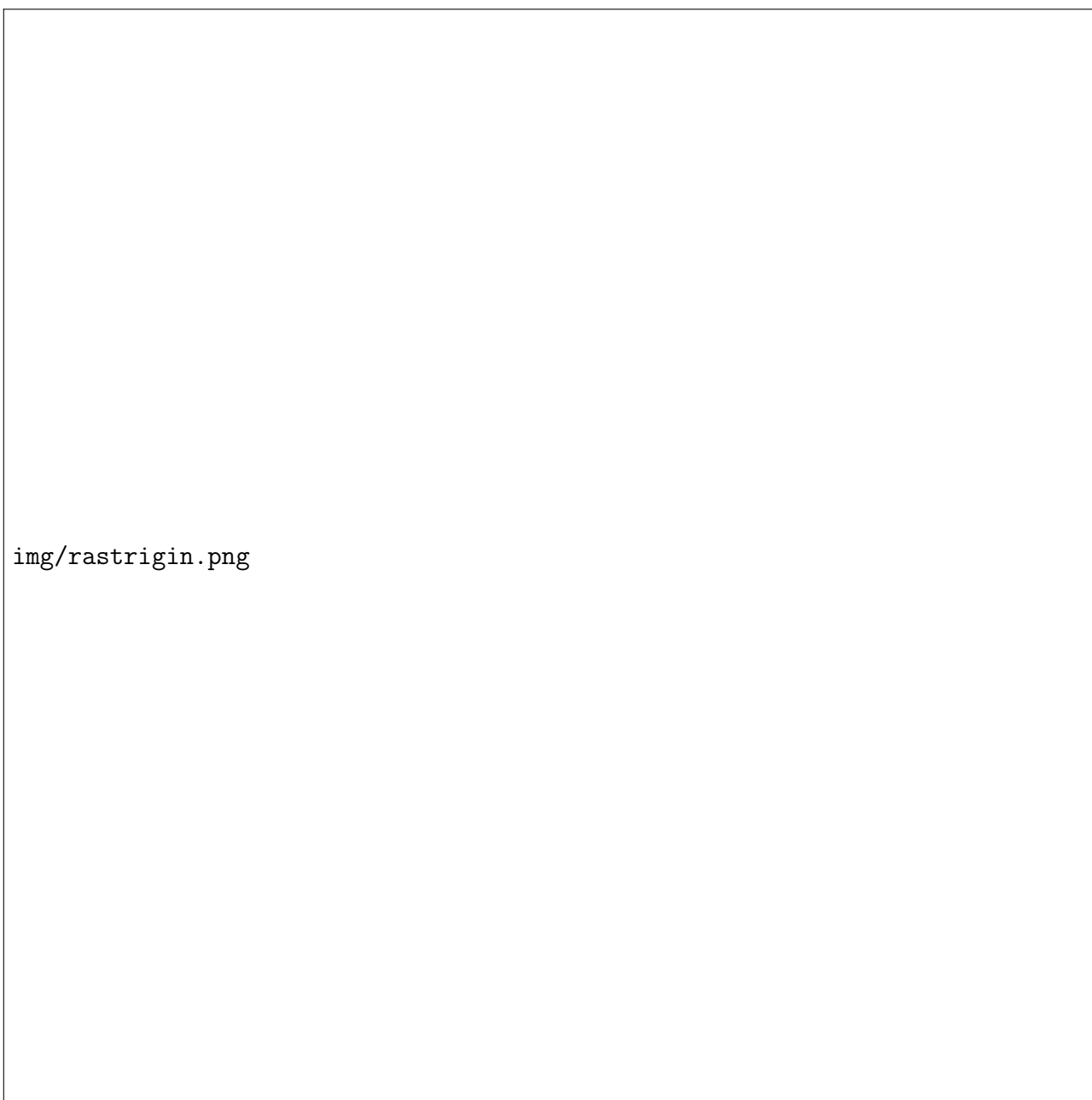


Figura 2: Interações do algoritmo para a função *peaks* nas gerações 1, 20 e 40



img/rastrigin.png

Figura 3: Interações do algoritmo para a função *rastrigin* nas gerações 1, 25 e 50