

Trabalho 02 - Ag simple

Cecília Carneiro e Silva

23/08/2017

1 Ag - Simple Example

Implementação de um exemplo de Algoritmo Genético. Seguindo a seção 3 do artigo do Tomassini. A linguagem utilizada foi Racket Language. Repositório da aplicação: <https://github.com/ceciliacsilva/agSimple>. Minimização da função $f(x)$, deseja-se encontrar o valor de x para $f(x)$ mínimo.

1.1 Considerações

- Avaliação dos indivíduos: baseada na função de aptidão (fitness), indica a qualidade do indivíduo na população.
- Para o resultado da função aptidão ser positivo, sendo $f(x)$ negativa para todo x , criou-se $g(x) = -f(x)$.
- Na verdade, foi encontrado o valor máximo da função.
- Critérios de parada: número máximo de gerações, repetição do melhor indivíduo por n gerações.

1.2 Estruturação

- Cromossomo = string, "0011011001"
- Indivíduo = pair, '(cromossomo . fitness)
- População = list, '((cromossomo1 . fitness1) (cromossomo2 . fitness2) ...)

1.3 Parametros do AG

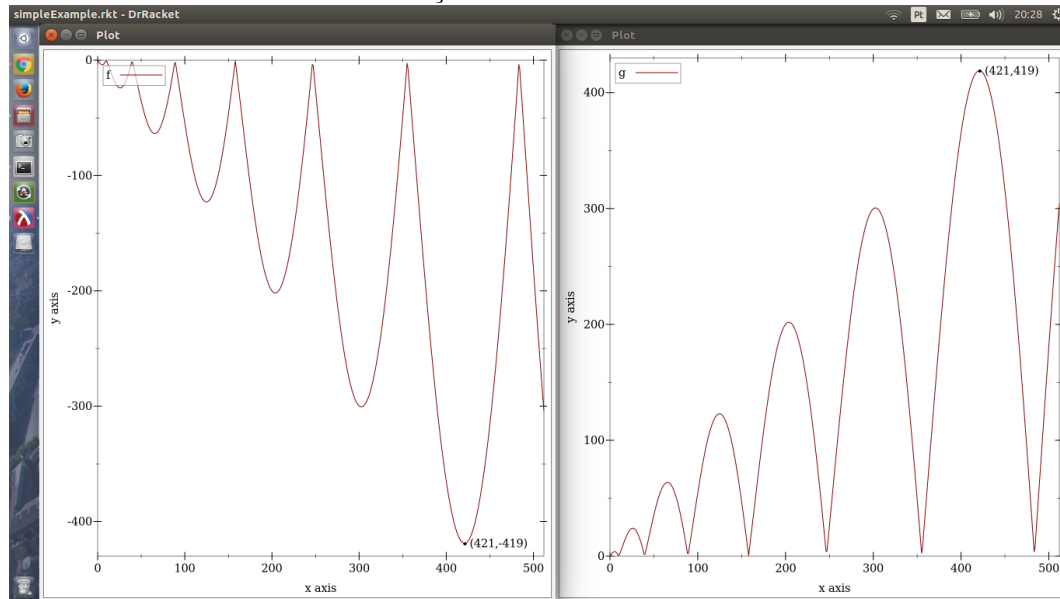
- pc: 0,6 (artigo)
- pm: 0,01 (artigo)
- tamanho da população: 50 (artigo)
- x máximo: 512 (artigo)
- cromossomo máximo: 1024 (artigo)

- número máximo de gerações: 200
- repetição máxima do melhor indivíduo: 5

1.4 Resultados

```
> ;;(find-min <nome_simulacao> <parametros-ag>)
> (find-min "simul3" *ag*)
x para f(x) mínimo: 421 valor: 418.9827640161443
'("1101001010" . 418.9827640161443)
```

Gráficos exibidos ao fim da execução:



Vídeo com a evolução da população: <https://github.com/ceciliacsilva/agSimple/blob/master/simul3/output.mp4>

- OBS: para gerar o vídeo é necessário entrar o diretório <nome_simulacao> e executar o script "toVideo.sh".

```
$ cd <nome_simulacao>
$ sh toVideo.sh
```

1.4.1 Outros testes

- simul4:
 - cromossomo máximo: 2048
- simul5:
 - repetição máxima do melhor indivíduo: 20
- simul6:
 - x máximo: 256