
Algoritmos Genéticos

— Encontrando n termos de —
Somatória

Bruno Borges e Rafael Assis

Trabalho de ICC - UFU - 2018-1

Problema



“O problema consiste em achar uma sequência de n termos positivos e distintos que quando somados resultam em um valor desejado r .”

Metodologia e Compatibilidade



- Solução usando Algoritmos Genéticos (AG) em Python
- Foi utilizado AG, pois o espaço de total busca é grande dependendo de n e r ficando semelhante a um fatorial.

$$U = (r - 1)_1 (r - 2)_2 (r - 3)_3 \dots (r - n)_n$$

Exemplo: para $n = 5$ e $r = 500 \Rightarrow 30.629.362.512.000$

Devido a aleatoriedade, tendência a convergir de AG e haver várias respostas , consegue executar em segundos o exemplo acima.

Fluxograma do AG geral

1. Gerar população inicial com n indivíduos
2. Calcular fitness dos indivíduos
3. Avaliar a população:
 - a. Se um indivíduo ideal for encontrado ou atingiu o máximo de gerações
 - b. Apresenta melhor indivíduo encontrado (FIM)
 - c. Se não, vai para (4)
4. Seleção: escolher indivíduos que formarão pares para crossover
5. Crossover: Gerar Filhos com os pais escolhidos
6. Mutação: sobre uma taxa p , aplicar mutação nos filhos gerados
7. Reinserção: Colocar os novos filhos na população (Volta para 2)

Configuração do cromossomo

O cromossomo é representado por um array de 'n' posições, onde cada gene 'xi' é um termo da somatória, $x_1 + x_2 + \dots + x_n$.

x1	x2	...	xn
----	----	-----	----

Devido a restrição, cada $x_i < r$

Parâmetros e população inicial

- Taxa de crossover = 100%, ou seja, gera-se 100 indivíduos
- Taxa de mutação = 10%

O AG consiste numa população inicial de 100 indivíduos gerados aleatoriamente respeitando a regra de não igualdade entre os genes.

Avaliação da população

O fitness é a distância entre a soma dos x_i termos e o valor desejado.

$$FITNESS = \left| r - \sum_{i=1}^n x_i \right|, \text{ onde } r = \text{valor desejado}$$

O melhor indivíduo é aquele que apresentar um fitness igual ou próximo a 0.

Seleção

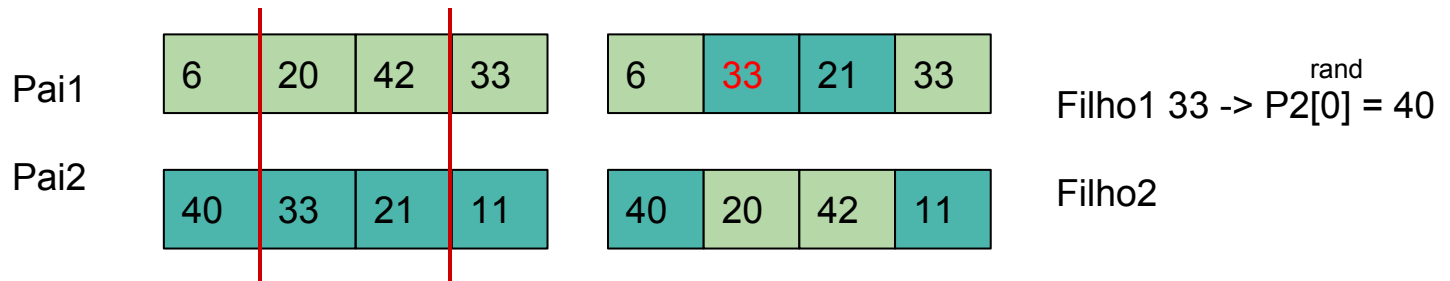
Foi escolhido o método de torneio

Escolhe-se 20% do tamanho da população de indivíduos de forma aleatória e seleciona o melhor deles para ser um dos pais no crossover

Crossover

Utilizado o método crossover múltiplo com a adição de troca aleatória quando houver algum repetido.

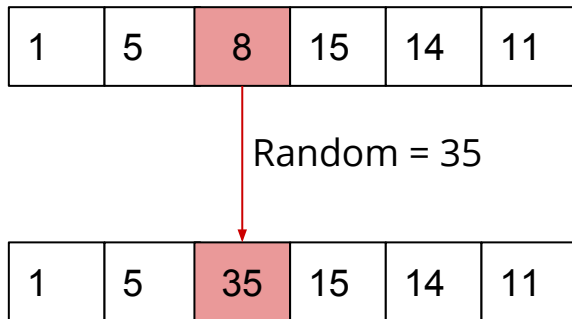
1. Seleciona faixa de troca de genes entre os pais e realiza a troca
2. Se existir elementos repetidos, irá fazer troca do elemento repetido com algum outro do outro pai até ficar sem repetição



Mutação

Para cada gene há 10% de chance de sofrer mutação.

Na mutação é gerado um número randômico entre 0 e o valor desejado - 1, então verifica-se a restrição de desigualdade entre os genes, em caso de repetição, gera-se outro valor até a restrição ser atendida, então o valor aleatório substituirá o gene marcado para mutação.

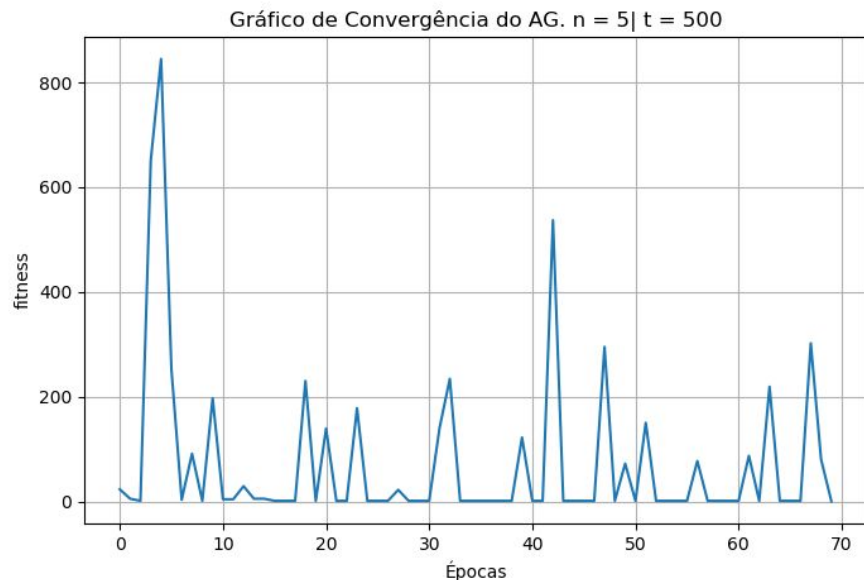


Reinserção

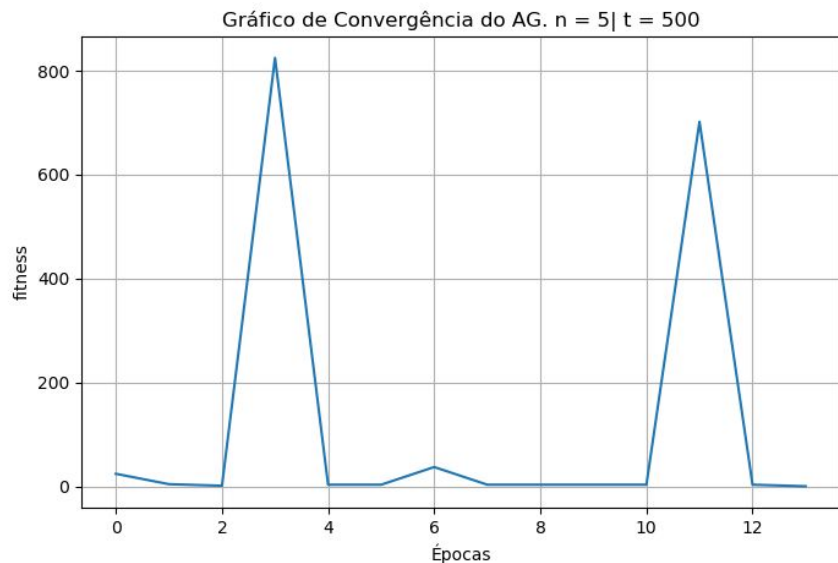
É feita a Reinserção pura, todos os filhos gerados substituem a população anterior

Exemplo de Resultados obtidos

Como AG é probabilístico, sua execução costuma dar resultados diferentes

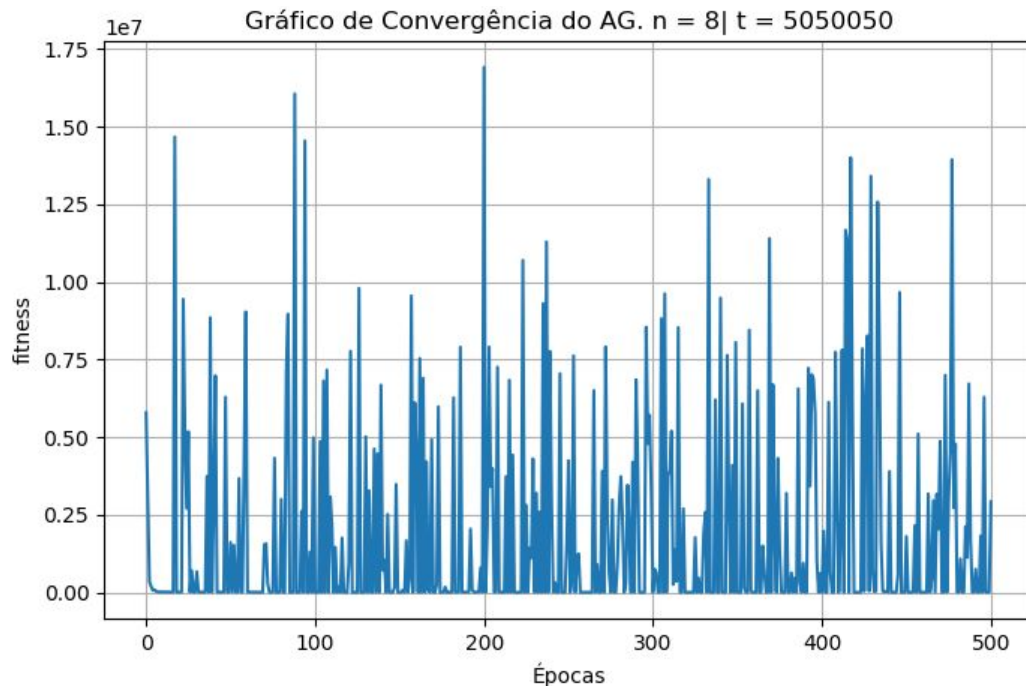


Geração 68: [104, 107, 13, 3, 273]
0.607 segundos



Geração 13 : [83, 37, 95, 51, 234]
0.119 segundos

Resultado para quando não converge



Geração: 218

Fitness: 84

Tempo: 5.19 Segundos

Cromossomo: [178899, 223083, 429427, 662305, 40650, 2648791, 692230, 174749]

Caso Particular

Ag - Somatoria - Buscando 20 numeros para 30

Chegou ao Limite de epocas: 500

A Melhor Resposta das Geracoes:

Geracao: 17| Fitness: 160

Cromossomo: [14, 17, 16, 6, 3, 4, 12, 19,
13, 2, 10, 11, 1, 9, 5, 0, 15, 8, 7, 18]

Sum: 190

Tempo de Treinamento : 9.189 segundos

Restrição Identificada pelos Resultados

Se a somatória de 0 até n para n ($sum(n)$) for maior que r , então, não há solução, pois precisaria que houvesse números repetidos para alcançar fitness 0 e esbarra na restrição.

Nesse caso, o melhor fitness possível para esse caso é $best(r)$

$$sum(n) = \sum_{x=0}^{n-1} = \frac{(n-1)n}{2} \quad best(n, r) = sum(n) - r$$

Prova do caso

Então, para o exemplo visto $n = 20$ e $r = 30$, temos que:

$$\text{sum}(20) = 190 > 30$$

$$\text{best}(20, 30) = 190 - 30 = 160$$

Ag - Somatoria - Buscando 20 numeros para 30

Chegou ao Limite de epocas: 500

A Melhor Resposta das Geracoes:

Geracao: 17| Fitness: 160

Cromossomo: [14, 17, 16, 6, 3, 4, 12, 19,
13, 2, 10, 11, 1, 9, 5, 0, 15, 8, 7, 18]

Sum: 190

Tempo de Treinamento : 9.189 segundos

Observando a análise matemática acima, temos que, o melhor fitness encontrado 160 no AG é mesmo o melhor possível, cujo cromossomo será de 0 a $n-1$ distintos

Referências

- Slides da Professora Rita