

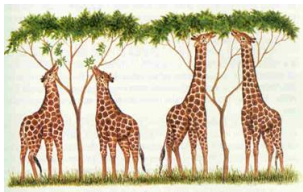
Algoritmos Genéticos (Computação Evolutiva)

Danilo Sipoli Sanches

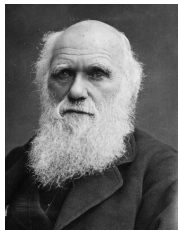
Departamento Acadêmico de Computação
Universidade Tecnológica Federal do Paraná
Cornélio Procópio



- Engloba métodos e técnicas computacionais inspirados
 - na teoria da evolução das espécies, de seleção natural (Darwin)
 - na Genética iniciada por Mendel
- Bases da evolução:
 - seres mais adaptados aos seus ambientes sobrevivem
 - características genéticas de tais seres são herdadas pelas próximas gerações
 - diversidade é gerada por cruzamento e mutações



- 1859 - Charles Darwin publica o livro “A Origem das Espécies”



As espécies evoluem pelo princípio da seleção natural e sobrevivência do mais apto

- 1865 - Gregor Mendel, pai da genética, apresenta experimentos do cruzamento genético de ervilhas



John Henry Holland: Father of Genetic Algorithms

- Nos anos 1960, John Holland e seus alunos propuseram a construção de um algoritmo de busca e otimização: os algoritmos genéticos;
- Professor of Electrical Engineering and Computer Science at the University of Michigan;
- Ann Arbor –Book ‘Adaptation in Natural and Artificial Systems’ (1975)



Algoritmo Genético Canônico

Fonte: <https://icaroagostino.github.io/post/sbo/>



Figura: Passos de um GA.

- Possuem as seguintes características:
 - Trabalham com uma codificação baseado em um conjunto de parâmetros;
 - A codificação pode ter diferentes representações (Estrutura de Dados);
 - Trabalham com uma população de soluções;
 - Utilizam informações de custo ou recompensa.
 - Utilizam regras de transição não determinísticas;
 - São baseados na técnica gerar-e-testar

Terminologia:

- Indivíduo: Simples membro da população.
- Cromossomo e Genoma: Coleção de genes;
- Estrutura de dados que codifica a solução de uma problema

- A população de um algoritmo genético é o conjunto de indivíduos que estão sendo cogitados como solução

Cada indivíduo é uma possível solução do problema

Algoritmos Genéticos - Indivíduo

- Um indivíduo no AG é um **cromossomo**
- Ou seja, um indivíduo é um conjunto de atributos da solução
- Geralmente é uma cadeia de bits que representa uma solução possível para o problema

Exemplo

População de tamanho $n=4$

Geração de indivíduos, com seus cromossomos

Cada elemento do vetor é um gene, um atributo da solução

Indivíduo 1 = [1 1 1 0 1]

Indivíduo 2 = [0 1 1 0 1]

Indivíduo 3 = [0 0 1 1 0]

Indivíduo 4 = [1 0 0 1 1]

Algoritmos Genéticos - Indivíduo

- Outras representações são possíveis
- Boa representação depende do problema

Exemplo - Problema da Mochila

- Dada uma lista de coisas com preços e tamanhos
- É fornecido o valor da capacidade da mochila
- Escolha as coisas de forma a maximizar o valor daquilo que cabe dentro da mochila, sem ultrapassar sua capacidade

Cada bit é usado para dizer se a coisa correspondente está ou não na mochila

- Crom: A = 1 0 1 1 0 0 1 0 1 1
- Crom: B = 1 1 1 1 1 1 0 0 0 0

Codificação Binária

Cada cromossomo é uma string de bits - 0 ou 1

Algoritmos Genéticos - Indivíduo

- Outras representações são possíveis
- Boa representação depende do problema

Exemplo - Problema do caixeiro viajante

- São dadas cidades e as distâncias entre elas
- O caixeiro viajante tem que visitar todas as cidades
- Encontrar a sequência de cidades em que as viagens devem ser feitas de forma que a distância percorrida seja a mínima possível

Cromossomos descrevem ordem de visita das cidades

- Crom: A = 1 5 3 2 6 4 7 9 8
- Crom: B = 8 5 6 7 2 3 1 4 9

Codificação por Permutação

Cada cromossomo é uma string de números que representa uma posição em uma sequência

Algoritmos Genéticos - Função de Aptidão

- Função de *fitness*, função de custo \rightarrow determina uma nota a cada indivíduo
- Nota avalia quão boa é a solução que este indivíduo representa

Se indivíduo I_1 representa uma solução melhor do que $I_2 \rightarrow$ avaliação de I_1 deve ser maior do que de I_2

O objetivo de um AG pode ser maximizar o número de 1s

Indivíduos	Função de aptidão (fitness)
[11101]	4
[01101]	3
[00110]	2
[10011]	3
Aptidão média	3

- Função de Aptidão Padrão
 - utiliza apenas informação sobre “qualidade do cromossomo”

$$f_i = \frac{q_i}{\sum_j q_j}$$

q = aptidão do cromossomo

- Função de Aptidão Padrão

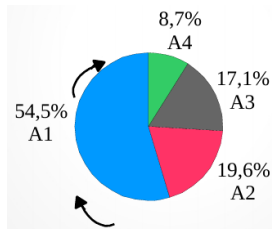
Cromossomo	Grau	Aptidão Padrão
1 4	4	0.4
3 1	3	0.3
1 2	2	0.2
1 1	1	0.1

- De acordo com a teoria de Darwin, o melhor sobrevivente para criar a descendência é selecionado
- Privilegiar indivíduos com função de **avaliação alta**
 - não desprezar completamente indivíduos com função de **avaliação extremamente baixa**
 - indivíduos com péssima avaliação podem ter características genéticas favoráveis à criação de um indivíduo ótimo
 - características podem não estar presentes em nenhum outro cromossomo

- Há muitos métodos para selecionar o melhor cromossomo
 - seleção por **roleta**
 - seleção por **torneio**
 - seleção por **amostragem universal estocástica**
 - ...

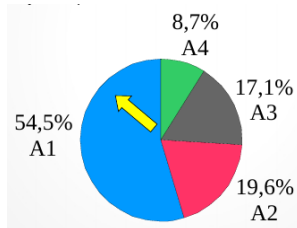
A seleção dirige o AG para as melhores regiões do espaço de busca

- Para visualizar este método considere um círculo dividido em N regiões (tamanho da população)
 - onde a área de cada região é proporcional à aptidão do indivíduo



Algoritmos Genéticos - Seleção por Roleta

- Coloca-se sobre este círculo uma “roleta”
- A roleta é girada um determinado número de vezes, dependendo do tamanho da população
- São escolhidos como indivíduos que participarão da próxima geração, aqueles sorteados na roleta

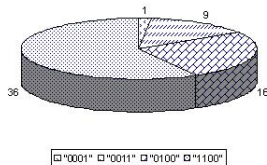


Algoritmos Genéticos - Seleção por Roleta

Exemplo

Indivíduo	Avaliação	Aptidão Relativa (parte da roleta)
0001	1	1.61
0011	9	14.51
0100	16	25.81
0110	36	58.07
Total	62	100.0

Roleta Viciada para População Exemplo



Algoritmos Genéticos - Seleção por Torneio

- Escolhe n indivíduos (e.g., $n=3$) aleatoriamente, com mesma probabilidade
- Cromossomo com maior aptidão é selecionado para a população intermediária
- Processo se repete até que a população intermediária seja preenchida
- Parâmetro n , tamanho do torneio, define pressão seletiva
 - \uparrow número de indivíduos que participam do torneio $\rightarrow \uparrow$ pressão seletiva
 - indivíduo tem que ser melhor do que uma quantidade maior de competidores

Indivíduo	Aptidão	Candidatos \rightarrow Vencedor
10110	2.23	$S_1, S_2, S_5 \rightarrow S_2$
11000	7.27	$S_2, S_4, S_5 \rightarrow S_2$
11110	1.05	$S_5, S_1, S_3 \rightarrow S_1$
01001	3.35	$S_4, S_5, S_3 \rightarrow S_4$
00110	1.69	$S_3, S_1, S_5 \rightarrow S_1$

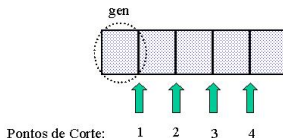
- Um conjunto de operações é necessário para que, dada uma população, seja possível gerar populações sucessivas que (espera-se) melhorem sua aptidão com o tempo
- Estas operações são os **operadores genéticos**
 - **cruzamento**
 - **mutação**
 - **elitismo**

Os operadores genéticos permitem explorar áreas desconhecidas do espaço de busca

- O operador **cruzamento** (*crossover* ou recombinação) cria novos indivíduos, misturando características de **dois indivíduos pais**
- O resultado desta operação é **um indivíduo** que potencialmente combine as melhores características dos indivíduos usados como base
- Alguns tipos de cruzamento são:
 - cruzamento em um ponto
 - cruzamento em dois pontos
 - cruzamento multi-pontos
 - uniforme

Algoritmos Genéticos - Cruzamento

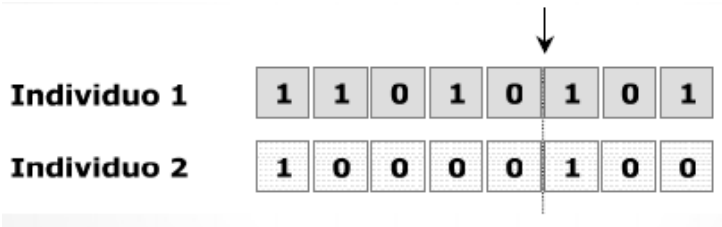
- Um ponto de corte deve ser selecionado
- Constitui uma posição entre dois genes de um cromossomo
- Cada indivíduo de n genes contém $n - 1$ pontos de corte possíveis



- Separação do pai em duas partes: esquerda e direita do ponto de corte
 - partes não necessariamente têm o mesmo tamanho
- Primeiro filho: concatenação da parte esquerda do primeiro pai + parte direita do segundo pai
- Segundo filho: concatenação da parte esquerda do segundo pai + parte direita do primeiro pai

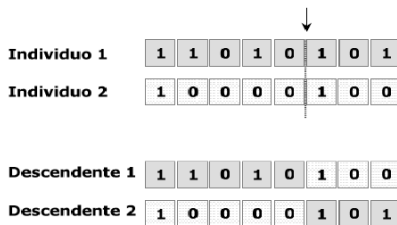
Algoritmos Genéticos - Cruzamento de um Ponto

- No cruzamento de um ponto, divide-se cada progenitor em duas partes, em uma localidade k (escolhida aleatoriamente)

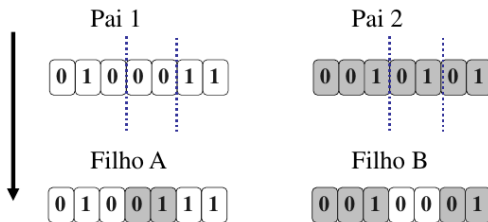


Algoritmos Genéticos - Cruzamento de um Ponto

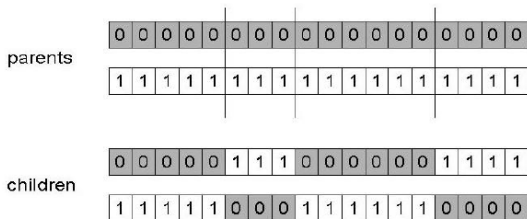
- O descendente 1 consiste em genes 1 a k do progenitor 1, e genes $k + 1$ a $n - 1$ do progenitor 2
- O descendente 2 é “reverso”



Algoritmos Genéticos - Cruzamento de Dois Pontos

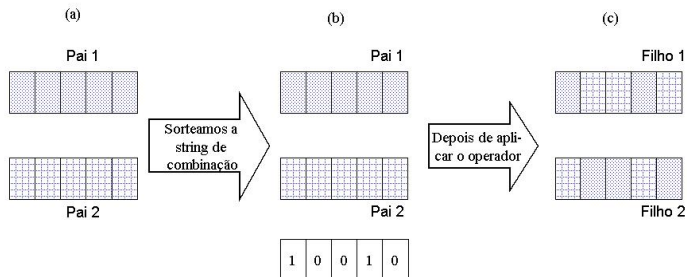


Algoritmos Genéticos - Cruzamento de n Pontos

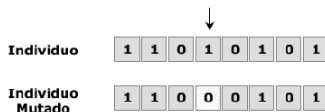


Algoritmos Genéticos - Cruzamento Uniforme

- Para cada gene é sorteado um número zero ou um
- Se o sorteado for 1, um filho recebe o gene do primeiro pai e o segundo filho o gene do segundo pai
- Se o sorteado for 0, o primeiro filho recebe o gene do segundo pai e o segundo filho recebe o gene do primeiro pai



- A mutação modifica aleatoriamente alguma característica do indivíduo, sobre o qual é aplicada

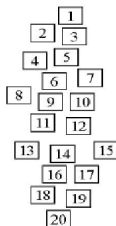


- O operador de mutação é necessário para a introdução e manutenção da diversidade genética da população
- Desta forma, a mutação assegura que a probabilidade de se chegar a qualquer ponto do espaço de busca, possivelmente, não será zero
 - reduz chance de ficar preso em mínimos locais
 - taxa de mutação pequena \rightarrow 0.5% ou 1%

- Conjunto de indivíduos mais adaptados é automaticamente selecionado para a próxima geração
- Evita modificações deste(s) indivíduo(s) pelos operadores genéticos
 - utilizado para que os melhores indivíduos não desapareçam da população

Algoritmos Genéticos - Gerações

- Algoritmo é iterado até algum critério de parada
 - tempo de execução, número de gerações, valores de aptidão mínimo ou médio
- A cada passo, um novo conjunto de indivíduos é gerado a partir da população anterior
- A este novo conjunto dá-se o nome de geração
- Com a criação de uma grande quantidade de gerações que é possível obter resultados dos AGs



Algoritmos Genéticos - Algoritmo

Algoritmo_Genetico

- p = tamanho da população
- r = taxa de cruzamento
- m = taxa de mutação
- 1 $P \leftarrow$ gerar aleatoriamente p indivíduos
- 2 **Para cada** i em P , computar Aptidão(i)
- 3 **Enquanto** critério_parada não é atingido
 - 1 selecionar p membros de P para reprodução
 - 2 aplicar cruzamento a pares de indivíduos selecionados segundo taxa r , adicionando filhos em PS
 - 3 realizar mutação em membros PS , segundo taxa m
 - 4 $P \leftarrow PS$
 - 5 **Para cada** i em P , computar Aptidão(i)
- 4 **Retornar** o indivíduo de P com maior aptidão

Codificação e avaliação de aptidão são pontos chave

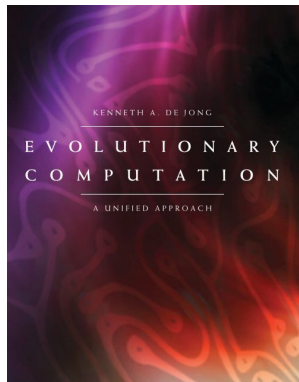
- O desempenho dos algoritmos genéticos é fortemente influenciado pela definição dos seus parâmetros
- **Tamanho da população**
 - populações pequenas: cobrem pouco o espaço de busca
 - populações grandes: apesar de evitar mínimos locais, requer mais recursos computacionais e tempo
- **Intervalo de geração:**
 - porcentagem da população que será substituída
 - grande (comum): filhos substituem pais
 - pequena: “pais e filhos convivem”

- O desempenho dos algoritmos genéticos é fortemente influenciado pela definição dos seus parâmetros
- **Taxa de cruzamento**
 - se for muito baixa: busca pode estagnar
 - se for muito alta: boas estruturas podem ser perdidas
- **Taxa de mutação:**
 - possibilita que qualquer ponto de espaço de busca seja atingido
 - se for muito alta: busca aleatória

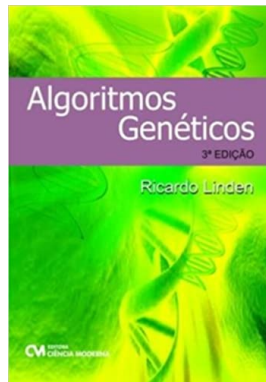
Questões Importantes

- Como criar cromossomos?
- Qual tipo de codificação usar?
 - primeiras perguntas que devem ser feitas ao resolver um problema com AG
 - codificação dependerá fortemente do problema
- Como escolher os pais para a realização do crossover?
- A geração de uma população a partir de duas soluções pode causar a perda da melhor solução. O que fazer?

Livros de apoio



Kenneth De Jong



Ricardo Linden

Figura: Opções de livros sobre AGs