

Aula 5

Algoritmos Genéticos

A ignorância gera confiança com mais freqüência do que o conhecimento: são aqueles que sabem pouco, e não aqueles que sabem muito, que tão positivamente afirmam que esse ou aquele problema jamais será resolvido pela ciência.

(Charles Darwin)

Objetivos

Esperamos que, ao final desta aula, você seja capaz de:

- compreender a metáfora biológica utilizada para criação dos Algoritmos Genéticos;
- entender a estrutura e o funcionamento lógico de um Algoritmo Genético.

Pré-requisitos

Mais uma vez vamos invadir os domínios da Biologia. Assim é bom relembrar alguma coisa sobre as teorias evolucionárias e a forma como nossos genes trocam informação para criação de uma nova vida. Uma boa leitura, para melhor compreensão dos temas tratados nesta aula, é o livro *A Origem das espécies*, de Charles Darwin.

Introdução

A Inteligência Artificial, tradicionalmente, trabalha com bases de dados e bases de regras, realizando inferências lógicas até convergir para uma conclusão. Esse processo é caracterizado como comportamento inteligente, quando a tarefa executada é classificada como inteligente, se realizada por seres humanos.

Devido à dificuldade existente para que sistemas inteligentes tradicionais resolvam determinadas classes de problemas, como reconhecimento de padrões ou sistemas reativos, intensificaram-se as pesquisas no campo da

inteligência computacional (IC). As técnicas mais conhecidas de IC, segundo Palazzo (1997), são:

- redes neurais artificiais
- lógica fuzzy
- computação evolutiva
- agentes autônomos

O objeto de estudo desta aula são os **Algoritmos Genéticos** (AG's), os quais estão inseridos na computação evolutiva, portanto, entre as técnicas de IC anteriormente citadas, realizaremos uma breve descrição sobre computação evolutiva para então começarmos a falar sobre os AG's.

5.1 O que são os Algoritmos Genéticos (AG's)?

Os AG's são técnicas de busca baseadas nas teorias da evolução, nos quais as variáveis são representadas como genes em um cromossomo (indivíduo). Combinam a sobrevivência dos mais aptos com a troca de informação de uma forma estruturada, mas aleatória. O AG apresenta um grupo de soluções candidatas (população) na região de soluções. Por seleção natural e operadores genéticos, mutação e cruzamento, os cromossomos com melhor aptidão são encontrados. A seleção natural garante que os cromossomos mais aptos gerem descendentes nas populações futuras. Usando um operador de cruzamento, o AG combina genes de dois cromossomos pais previamente selecionados para formar dois novos cromossomos, os quais têm uma grande possibilidade de serem mais aptos que os seus genitores (DAVIS, 1991).

Holland (1975) acreditava que a incorporação das características naturais de evolução em um computador poderia produzir uma técnica para solucionar problemas da mesma maneira como funcionam na natureza os processos de seleção e adaptação. A forma de codificação dos cromossomos biológicos ainda não é totalmente compreendida pela comunidade científica, apesar dos enormes avanços conseguidos até os dias de hoje. Mas existem algumas características gerais da teoria que são aceitas, como nos ensina Davis (1991):

- evolução
- seleção natural
- reprodução
- ausência de memória

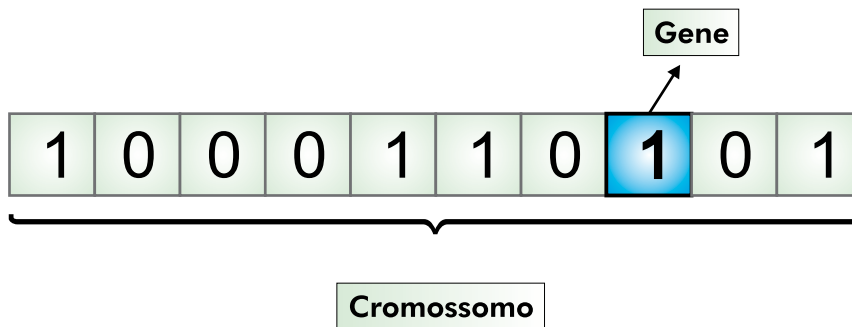
Serrada (1996, p. 23) define algoritmos genéticos como

[...] métodos estocásticos de busca cega de soluções quase ótimas. Neles se mantém uma população que representa um conjunto de possíveis soluções a qual é submetida a certas transformações

com as que se tenta obter novos candidatos e a um processo de seleção dirigido em favor dos melhores candidatos.

Na prática, nós podemos implementar facilmente um AG com o simples uso de *strings* de bits ou caracteres para representar os cromossomos e, com simples operações de manipulação de bits, podemos implementar cruzamento, mutação e outros operadores genéticos. Na figura seguinte, podemos ver a simplicidade de representação de um cromossomo composto por dez genes, por meio de uma *string* de valores binários.

Figura 1: Exemplo de representação de um cromossomo de genes binários



Pelo que foi apresentado até o momento, nota-se que a principal inovação dos AG's, no que se refere aos métodos de busca, é a implementação de um mecanismo de seleção de soluções no qual, em curto prazo, os melhores têm maior probabilidade de sobreviver e, em longo prazo, os melhores têm maior probabilidade de ter descendência. Dessa forma, o mecanismo de seleção se divide em dois segmentos: o primeiro escolhe os elementos que vão participar da transformação (operador de seleção) e o outro escolhe os elementos que vão sobreviver (operador de substituição).

Cabe ainda destacar que os AG's são, segundo Serrada (1996), métodos de busca. Conheça quais são esses métodos.

- **Cega:** não tem conhecimento específico do problema a ser resolvido, tendo como guia apenas a função objetivo.
- **Codificada:** não trabalham diretamente com o domínio do problema e sim com representações dos seus elementos.
- **Múltipla:** executa busca simultânea em um conjunto de candidatos.
- **Estocástica:** aleatória. Combinam em proporção variável regras probabilísticas e determinísticas. Esse conceito refere-se tanto às fases de seleção como às de transformação.

Apesar de aleatórios, eles não são **caminhadas aleatórias** não direcionadas, ou seja, buscas totalmente sem rumo, pois exploram informações históricas para encon-

trar novos pontos de busca em que são esperados melhores desempenhos. Isso é feito por meio de processos iterativos, em que cada iteração é chamada de geração.

5.2 Parâmetros Genéticos

É importante também analisar de que maneira alguns parâmetros influem no comportamento dos algoritmos genéticos, para que se possa estabelecê-los conforme as necessidades do problema e dos recursos disponíveis. Serão listados a seguir alguns parâmetros genéticos utilizados, segundo Mitchell (1996).

- **Tamanho da população:** o tamanho da população determina o número de cromossomos na população, afetando diretamente o desempenho global e a eficiência dos AG's. Com uma população pequena, o desempenho pode cair, pois desse modo a população fornece uma pequena cobertura do espaço de busca do problema. Uma grande população geralmente fornece uma cobertura representativa do domínio do problema, além de prevenir convergências prematuras para soluções locais ao invés de globais. No entanto, para se trabalhar com grandes populações, são necessários maiores recursos computacionais, ou que o algoritmo trabalhe por um período de tempo muito maior.
- **Taxa de cruzamento:** quanto maior for essa taxa, mais rapidamente novas estruturas serão introduzidas na população. Mas se essa for muito alta, a maior parte da população será substituída, e pode ocorrer perda de estruturas de alta aptidão. Com um valor baixo, o algoritmo pode tornar-se muito lento.
- **Tipo de cruzamento:** o tipo de cruzamento a ser utilizado determina a forma como se procederá a troca de segmentos de informação entre os casais de cromossomos selecionados para cruzamento.
- **Taxa de mutação:** determina a probabilidade em que uma mutação ocorrerá. Mutação é utilizada para dar nova informação para a população e também para prevenir que a população se sature com cromossomos semelhantes (convergência prematura). Uma baixa taxa de mutação previne que uma dada posição fique estagnada em um valor, além de possibilitar que se chegue em qualquer ponto do espaço de busca. Com uma taxa muito alta, a busca se torna essencialmente aleatória, além de aumentar muito a possibilidade de que uma boa solução seja destruída. DeJong sugeriu que a Taxa de Mutação seria inversamente proporcional ao tamanho da população, e Hessner e Manner sugeriram que a Taxa de Mutação ótima seria, aproximadamente (MICHALEWICZ, 1996, p. 87):

$$(n * \ell^{\frac{1}{2}})^{-1}$$

Onde n é o tamanho da população e ℓ é o número de genes do cromossomo.

5.3 Estrutura e Componentes Básicos

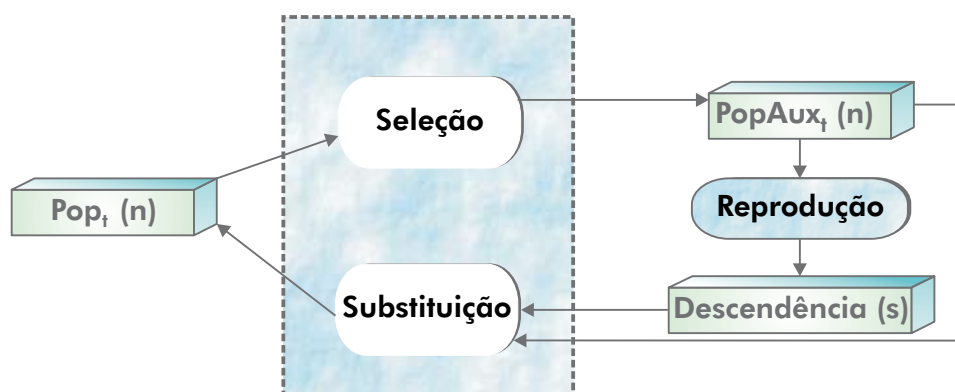
Como não há uma definição rigorosa para **algoritmo genético** que seja aceita por todos os componentes da comunidade da computação evolucionária, podemos dizer que a maioria dos métodos denominados **AG's** tem pelo menos os seguintes elementos em comum (HOLLAND, 1975):

- uma população de cromossomos;
- seleção de acordo com a aptidão;
- cruzamento (*crossover*) para gerar novas populações;
- mutação aleatória para produzir novos indivíduos.

Os cromossomos componentes da população de um AG assumem tipicamente a forma de *strings* binárias. Cada posição dentro dessa *string* tem dois alelos possíveis: 0 e 1. Cada cromossomo pode ser visto como um ponto no espaço de busca das soluções candidatas. O AG processa populações de cromossomos, efetuando substituições sucessivas de uma população por outra. Os AG's requerem uma função de aptidão que atribua um valor (valor de aptidão ou *fitness*) para cada cromossomo da população atual. O valor de aptidão de cada cromossomo depende de quão bem esse cromossomo resolve o problema dado.

Um esquema gráfico da estrutura de um AG básico seria como o apresentado na figura a seguir. Observe.

Figura 2: Estrutura básica de um AG



Como pode ser visto, uma população Pop_t que consta de n membros é submetida a um processo de seleção em que será gerada uma população auxiliar $PopAux_t$ de n criadores. Dessa população auxiliar será extraído um subgrupo de indivíduos denominados progenitores, que são os que efetivamente vão se reproduzir. Fazendo uso dos operadores genéticos, os progenitores passarão por uma série de transformações na fase de reprodução, por

meio das quais serão gerados os indivíduos componentes da descendência. Para formar a nova população $Pop_{[t+1]}$, devem ser selecionados n sobreviventes dos $n+s$ da população auxiliar e da descendência, o que é feito na fase de **substituição**. A fase de **seleção** é realizada em duas etapas (retângulo mais escuro) com finalidade de emular as duas vertentes do princípio de seleção natural: seleção de criadores ou seleção e seleção de sobreviventes para a próxima geração ou substituição.

5.4 Métodos e Critérios para Implantação de um AG

Para implantar um AG, é preciso definir de forma correta os seguintes métodos e critérios. Segundo Serrada (1996), esses métodos e critérios são os listados a seguir.

- a) **Critério de codificação:** tendo em vista que os AG's trabalham com manipulação de *strings* de determinados alfabetos (representação), preferencialmente com o uso de notação binária, pois quanto menor o alfabeto, maior a facilidade de implementação e manipulação, deve-se especificar a codificação com a qual se faz corresponder cada ponto do domínio do problema com um gene ou conjunto de genes do cromossomo.
- b) **Critério de tratamento dos indivíduos não factíveis:** nem sempre é possível estabelecer uma correspondência ponto-a-ponto entre o domínio do problema e o conjunto de *strings* binárias (ou de outro alfabeto utilizado) usado para resolvê-lo. Como consequência, nem todas as *strings* (indivíduos) codificam indivíduos válidos do espaço de busca e devem ser habilitados procedimentos úteis para distingui-las.
- c) **Critério de inicialização:** refere-se a como deve ser construída a população inicial com a qual se inicializará o AG.
- d) **Critério de parada:** devem ser determinadas as condições nas quais se considera que o AG encontrou uma solução aceitável ou tenha fracassado no processo de busca e não faça sentido continuar.
- e) **Funções de avaliação e aptidão:** deve ser determinada a função de avaliação mais apropriada para o problema, assim como a função de aptidão que utilizará o AG para resolvê-lo.
- f) **Operadores genéticos:** denominação dada aos operadores utilizados para levar a cabo a reprodução. Todo AG faz uso de pelo menos três operadores genéticos (levando em conta o AG básico): seleção, cruzamento e mutação. Fique claro que esses não são os únicos possíveis e, além de tudo, admitem variações.
- g) **Critérios de seleção:** a seleção deve dirigir o processo de busca em favor dos indivíduos mais aptos. Isso pode ser feito de várias maneiras como,

por exemplo, por amostragem direta, por amostragem aleatória simples ou por amostragem estocástica.

- h) **Critério de substituição:** os critérios com que se selecionam os criadores não necessariamente têm de ser os mesmos usados para selecionar os sobreviventes, logo a necessidade de especificá-los separadamente. Alguns exemplos são:

- substituição **imediate**: os s descendentes substituem seus progenitores;
- substituição **por fator cheio**: os s descendentes substituem aqueles membros da população mais parecidos com eles;
- substituição **por inserção**: são selecionados para serem eliminados s membros da população de criadores (geralmente os s piores), os quais serão substituídos pelos descendentes;
- substituição **por inclusão**: os s descendentes são somados aos n progenitores em uma única população e nesta são selecionados os n melhores.

- i) **Parâmetros de funcionamento:** um AG precisa que lhe proporcionem certos parâmetros de funcionamento (tamanho da população, probabilidades de aplicação dos operadores genéticos, precisão da codificação, etc.).

A seleção de um ou outro método ou critério anterior depende não apenas do tipo de problema a ser resolvido, mas também de que não sejam violados certos requisitos imprescindíveis ao bom funcionamento do AG.

5.5 Quando um AG pode ser Usado?

Em geral, os AG's têm sido mais usados para solução de funções de otimização, nas quais eles têm se mostrado bastante eficientes e confiáveis. Mas, como é de se esperar, nem todos os problemas podem ter resultados satisfatórios ou mesmo ser representados adequadamente para o uso de técnicas de AG. A recomendação geral é levar em conta as seguintes características relativas ao problema, antes de tentar o uso das referidas técnicas (GOLDBERG, 1989):

- a) o espaço de busca (possíveis soluções) do problema em questão deve estar delimitado dentro de uma certa faixa de valores;
- b) o mais recomendável é tentar resolver problemas que tenham espaços de busca discretos (mesmo que sejam muito grandes);
- c) deve ser possível definir uma função de aptidão (*fitness*) que nos indique quão boa ou ruim é uma determinada resposta. Essa não é nada além da função objetivo do problema foco. Um AG apenas maximiza, mas a minimização pode ser facilmente conseguida utilizando o recíproco da função

maximizante (tendo cuidado para não gerar divisão por zero). Uma das principais características dessa função é que deve ser capaz de dar prioridade às boas soluções de forma a que se propaguem com maior rapidez;

- d) as soluções devem poder ser codificadas de uma maneira que resulte relativamente fácil a sua implementação no computador;
- e) a forma mais comum de codificação das soluções é por meio de *strings* binárias, mas têm sido utilizados também outros alfabetos como números reais e letras. A preferência recai, como já foi explicado anteriormente, sobre a codificação com números binários, pois foram os inicialmente propostos por Holland (1975) e porque são bastante fáceis de implementar.

5.6 Mecanismos de Amostragem de Populações

Após decidir como será realizada a codificação de um AG (forma de representação do problema com números binários ou outro alfabeto), o passo seguinte é decidir como o AG procederá para seleção dos indivíduos (cromossomos), ou seja, como escolher os indivíduos na população que criarão os descendentes para a próxima geração.

Os mecanismos de amostragem são bastante variados. Entre eles, destacam-se três grupos principais, segundo o grau de influência da aleatoriedade no processo (DAVIS, 1991; GOLDBERG, 1989). Observe.

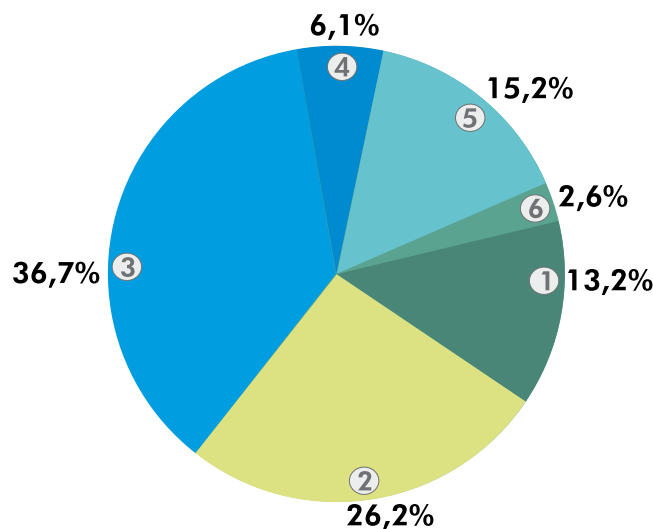
- a) **Amostragem direta:** seleção de um subconjunto de indivíduos da população mediante um critério fixo, no estilo de “os *n* melhores”, “os *n* piores”, “a dedo”, etc.
- b) **Amostragem aleatória simples ou equiprovável:** é determinada a mesma probabilidade para todos os elementos da população, de serem selecionados para a amostra.
- c) **Amostragem estocástica:** são atribuídas probabilidades de seleção ou pontuações aos elementos da população com base na sua função de aptidão (fitness).

Existem muitos mecanismos de amostragem estocástica, de acordo com a sua aplicação, entre os quais: seleção por sorteio, por restos, por roleta e por torneio. Apresentaremos o método de amostragem por roleta, uma vez que é geralmente o mais utilizado. Inicialmente proposto por Goldberg (1989), a **amostragem por roleta** é um método bastante simples que consiste em criar uma roleta na qual cada cromossomo possui um segmento proporcional à sua aptidão. Suponhamos uma população de seis cromossomos cuja aptidão é dada por uma função qualquer (nesse caso, é simplesmente a conversão de binário para decimal) conforme mostrado na tabela a seguir.

Tabela 1: Valores de exemplo para ilustrar seleção por roleta

CROMOSSOMO Nº	STRING	APTIDÃO	% DO TOTAL
1	0101101	45	13,2
2	1011001	89	26,2
3	1111101	125	36,7
4	0010101	21	6,1
5	0110100	52	15,2
6	0001001	9	2,6
Total		341	100,0

Com os valores percentuais constantes na quarta coluna da tabela anterior, iremos elaborar a roleta constante da figura seguinte. Essa roleta irá ser girada seis vezes para efetuar a seleção da população auxiliar (amostra) levando em conta que os indivíduos com maior área na roleta têm, conseqüentemente, maiores chances de serem selecionados mais vezes do que os indivíduos menos aptos.

Figura 3: Representação gráfica da roleta

5.7 Operadores Genéticos

São os operadores genéticos que transformam a população por meio de sucessivas gerações, estendendo a busca até chegar a um resultado satisfatório. Um algoritmo genético padrão evolui, em suas sucessivas gerações, mediante o uso de três operadores básicos:

- **seleção:** realiza o processo de adaptabilidade e sobrevivência;
- **cruzamento:** representa o acasalamento entre os indivíduos;
- **mutação:** introduz modificações aleatórias.

a) **Seleção**

A idéia principal do operador de seleção em um algoritmo genético é oferecer aos melhores indivíduos da população corrente (aqueles que estão mais adaptados ao meio), preferência para o processo de reprodução, permitindo que esses indivíduos possam passar as suas características às próximas gerações. Isso funciona como na natureza, em que os indivíduos altamente adaptados ao seu ambiente possuem naturalmente mais oportunidades para reproduzir do que aqueles indivíduos considerados mais fracos (SCHNEIDER, 1998).

b) **Cruzamento**

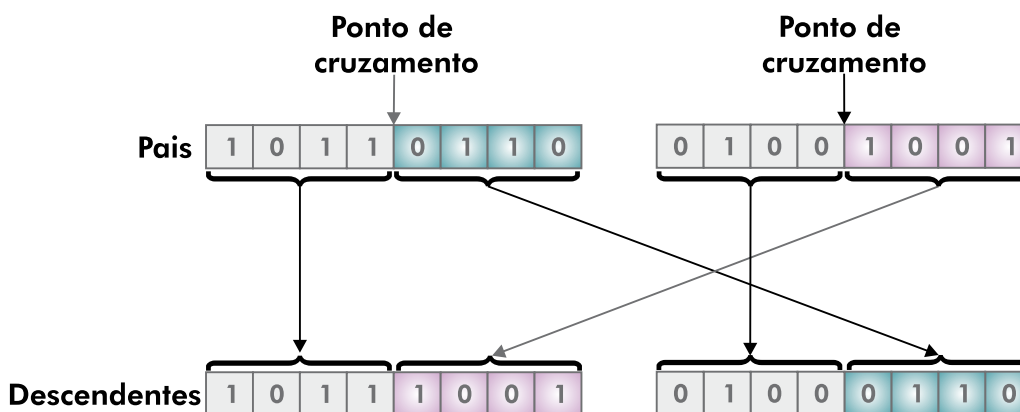
Poderia ser dito que a principal característica de distinção do AG em relação a outras técnicas é o uso do cruzamento (MITCHELL, 1996).

O operador cruzamento é utilizado após o de seleção. Essa fase é marcada pela troca de segmentos entre **casais** de cromossomos selecionados para dar origem a novos indivíduos que formarão a população da próxima geração. A idéia central do cruzamento é a propagação das características dos indivíduos mais aptos da população por meio de troca de segmentos informações entre eles, o que dará origem a novos indivíduos.

As duas formas mais comuns de reprodução sexual em AG's são de um ponto de cruzamento e de dois pontos de cruzamento.

- **Um ponto de cruzamento** (*single-point crossover*): o ponto de quebra do cromossomo é escolhido de forma aleatória sobre a longitude da string que o representa, e é a partir desse ponto que se realiza a troca de material cromossômico entre os dois indivíduos, como é representado na figura seguinte, na qual a termo de representação foi escolhido o ponto de cruzamento 4.

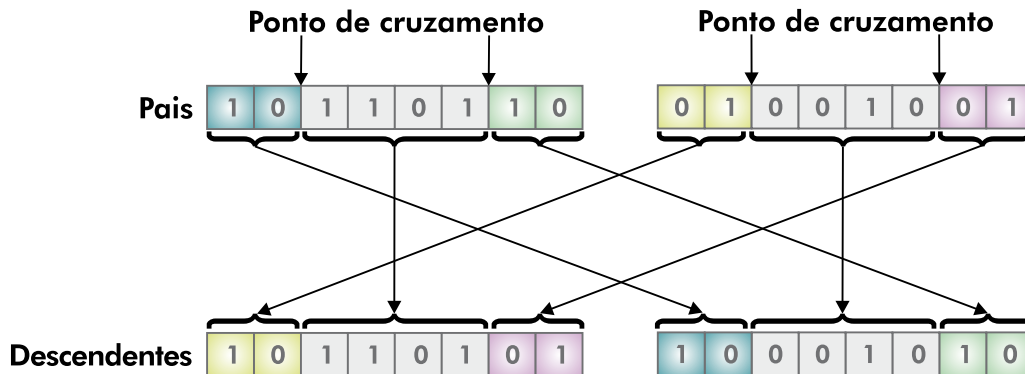
Figura 4: Esquema gráfico do cruzamento de um ponto



- **Dois pontos de cruzamento** (*two-point crossover*): procede-se de maneira similar ao cruzamento de um ponto, mas a troca de segmentos do gene

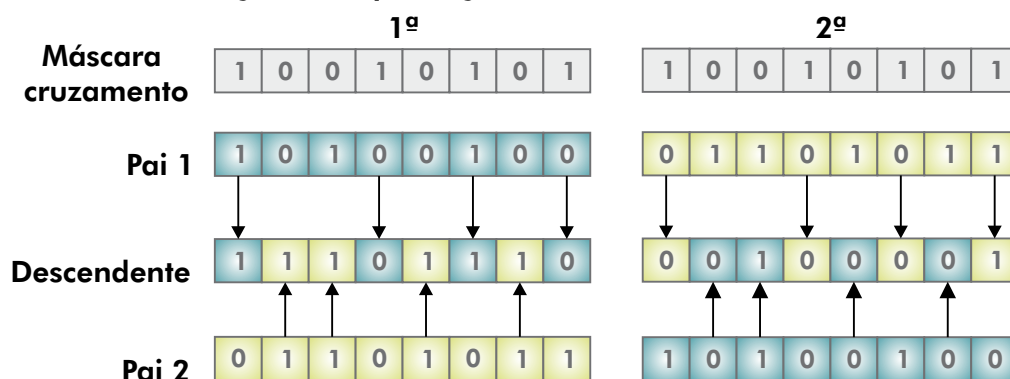
ocorre como mostrado na figura, na qual a termo de representação foram escolhidos os pontos 2 e 6.

Figura 5: Esquema gráfico do cruzamento de dois pontos



- **Cruzamento uniforme** (*uniform crossover*): é radicalmente diferente do cruzamento de um ponto. Cada gene do descendente é criado copiando o gene correspondente de um dos pais, escolhido de acordo com uma **máscara de cruzamento** gerada aleatoriamente. Onde houver 1 na máscara de cruzamento, o gene correspondente será copiado do primeiro pai e, onde houver 0, será copiado do segundo. O processo é repetido com os pais trocados para produzir o segundo descendente. Uma nova máscara de cruzamento é criada para cada par de pais. O número de pontos de troca de informação não é fixo, mas em geral é usado $\ell/2$ (onde ℓ é o comprimento do cromossomo). A figura demonstra graficamente o processo.

Figura 6: Esquema gráfico do cruzamento uniforme

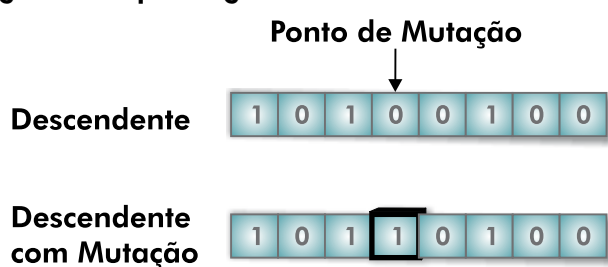


c) Mutação

A mutação é geralmente vista como um operador de *background*, responsável pela introdução e manutenção da diversidade genética na população (HOLLAND, 1975). Ela trabalha alterando arbitrariamente um ou mais

componentes de uma estrutura escolhida entre a descendência, logo após o cruzamento, fornecendo dessa forma meios para a introdução de novos elementos na população. Assim a mutação assegura que a probabilidade de se chegar a qualquer ponto do espaço de busca nunca será zero. O operador de mutação é aplicado aos indivíduos com uma probabilidade dada pela taxa de mutação r_m . Geralmente se utiliza uma taxa de mutação pequena, justamente por ser um operador genético secundário. A figura seguinte ilustra o processo de mutação em um indivíduo da descendência (BÄCK, 1996).

Figura 7: Esquema gráfico de ocorrência de mutação



5.8 O Algoritmo Genético Simples (AGS)

O algoritmo genético simples (AGS) foi proposto por Goldberg em seu texto clássico (GOLDBERG, 1989) e é o modelo que, com pequenas alterações, é usado geralmente para ilustrar o funcionamento dos AG's.

Os métodos e critérios utilizados pelo AGS são os listados a seguir.

- a) **Critério de codificação:** depende de cada problema a forma como vai ser feita a representação, desde que cada ponto do domínio do problema corresponda a um elemento do espaço de busca, o qual deve ser composto por inteiros binários simples (*strings* binárias simples).
- b) **Critério de tratamento dos indivíduos não factíveis:** não usado, pois considera que todas as soluções possíveis de representação pelos indivíduos da população são factíveis.
- c) **Critério de inicialização:** inicialização realizada por cadeias binárias constituídas aleatoriamente.
- d) **Funções de avaliação e aptidão:** a função de aptidão e a de avaliação são iguais.
- e) **Operadores genéticos:** usa o cruzamento simples (um ponto de cruzamento) e a mutação é realizada bit a bit na população de indivíduos.
- f) **Critério de seleção:** por roleta.

- g) **Critério de substituição:** imediato.
- h) **Critério de parada:** é limitado pela estipulação de um número máximo de iterações.
- i) **Parâmetros de funcionamento:** discretos. Um conjunto de valores tipicamente utilizados, segundo Grefenstette (1990), seria:
 - tamanho da população = 50
 - número máximo de iterações = 1000
 - taxa de cruzamento = 0,60
 - taxa de mutação = 0,001

Os passos seguidos pelo AGS seriam, basicamente, os a seguir, descritos mais detalhadamente. Vejamos.

Primeiro: inicializar um contador de gerações t com valor 0.

Segundo: iniciar com uma população par de n cromossomos com ℓ bits cada um, gerada randomicamente (soluções candidatas do problema).

Terceiro: repetir os passos a seguir até que t iterações tenham sido efetuadas:

- a) calcular a aptidão (fitness) $f(X)$ de cada cromossomo X da população atual P ;
- b) com base nos valores acumulados das aptidões dos cromossomos, criar a roleta de seleção;
- c) repetir os passos a seguir até que n descendentes tenham sido criados:
 - efetuar duas vezes o **giro** da roleta de forma a selecionar o par de cromossomos pais da população corrente;
 - realizar o cruzamento simples (um ponto de cruzamento), como foi demonstrado na seção 2.5.5, entre os pares de pais selecionados em **a**, o que irá gerar dois novos descendentes;
 - sobre a descendência, aplicar o processo de mutação (2.5.5) segundo a taxa dada, bit a bit. Ou seja, calcular a probabilidade de mutação para cada bit da descendência, alterando o valor do gene quando for estipulado.
- d) substituir a população atual pela descendência.

Como pôde ser observado, a construção de um AGS é bastante simples e tem obtido bons resultados em muitos problemas práticos. Com base em um AGS, devido à extrema flexibilidade por ele apresentada e aos inúmeros métodos e operadores que podem ser utilizados (e criados), podemos obter uma infinidade de variações que vão se adaptar melhor a determinadas situações. Isso, é claro, fica por conta da análise e da criatividade do pesquisador ao tentar resolver um determinado problema.

5.9 Limitações do AGS

Apesar de sua simplicidade e poderio, o AGS, na hora de ser posto em prática, apresenta-nos algumas características por vezes bastante frustrantes. Portanto convém, antes de usá-lo na solução prática de algum problema, estarmos cientes de suas limitações, as quais estão diretamente ligadas a quatro características fundamentais, conforme mostra o quadro.

Quadro: Características que podem limitar o AGS

CARACTERÍSTICA	DESCRIÇÃO
Representação	O espaço de busca deve ser representado unicamente mediante alfabeto binário, o que dificulta por vezes por em prática a solução de um problema por meio de AGS.
Irrestrição	O mecanismo do AGS não considera as possíveis restrições ou ligações que possam ser impostas a uma busca.
Opacidade	O AGS é um algoritmo de busca cega, ou seja, é guiado apenas pela aptidão dos indivíduos, sem incorporar nenhum outro conhecimento específico do problema em questão.
Finitude	Para efeitos práticos, o AGS só pode trabalhar com populações finitas, de tamanho não muito grande.

Fonte: Serrada (1996, p. 42).

5.10 Representação e Codificação dos AG's

As partes que relacionam um AG a um problema dado são a **codificação** e a **função de avaliação**.

Se um problema pode ser representado por um conjunto de parâmetros (genes), os parâmetros podem ser unidos para formar uma cadeia de valores (cromossomo). Esse processo se chama **codificação**. Em genética, esse conjunto representado por um cromossomo em particular é referenciado como **genótipo**, contendo a informação necessária para construir um organismo, conhecido como **fenótipo**. Esses mesmos termos se aplicam em AG. Por exemplo, para se desenhar um prédio, um conjunto de parâmetros especificando o desenho da planta é o genótipo, e a construção final é o fenótipo. A adaptação de cada indivíduo depende de seu fenótipo, no qual se pode inferir seu genótipo.

Por exemplo, para um problema de maximizar uma função de três variáveis, $f(X, Y, Z)$, poderíamos representar cada variável por um número binário de 10 bits, obtendo-se um cromossomo de 30 bits de longitude e 3 genes.

Existem vários aspectos relacionados com a codificação de um problema a serem tomados em conta no momento de sua realização:

- deve ser utilizado um alfabeto o menor possível para representar os parâmetros (normalmente utilizam-se dígitos binários);

- as variáveis que representam os parâmetros do problema devem ser discretizadas para poder representar as *strings* de bits;
- a maior parte dos problemas tratados com AG's são não lineares e muitas vezes existem relações **ocultas** entre as variáveis que formam a solução;
- o tratamento dos genótipos inválidos deve ser tomado em conta para o desenho da codificação.

5.11 Seleção das Funções de Avaliação e Aptidão

As funções de avaliação e de aptidão devem ser elaboradas tendo em vista o tipo de problema a ser resolvido, ou seja, para cada problema teremos, na grande maioria dos casos, que criar novas funções para atribuição dos valores de aptidão aos indivíduos componentes da população.

Torna-se necessário diferenciar função de aptidão e função de avaliação para fazer que as avaliações assumam valores positivos e para controlar a densidade da população. Inicialmente, a função de avaliação se faz coincidir com o objetivo a maximizar, isso sem esquecer que nem sempre é fácil especificar numericamente o objetivo.

Uma vez que a função deve ser de fácil manipulação prática, pode ocorrer, em determinadas circunstâncias, que exista um objetivo claramente definido, mas a determinação explícita de seu valor para cada indivíduo é complicada. Em tal situação, não se pode tomar esse objetivo como função de avaliação, visto que seria inoperante.

Dado que os AG's maximizam atitudes, admite-se por padrão que se maximiza o objetivo. Em certos casos, pode ser mais prático determinar o problema como sendo de minimização de certo objetivo.

Finalmente, uma vez selecionada a função de avaliação, é construída a função de aptidão deslocando-a e escalonando-a convenientemente. O deslocamento tem por finalidade principal fazer com que a função de aptidão retorne valores positivos.

Pois é... Chegamos ao fim desta aula com uma boa base sobre computação evolucionária e algoritmos genéticos. Resta agora saber como, quando e onde utilizar essas técnicas para solução de problemas. É aí que entra a criatividade e o bom senso, e isso depende de você!

Síntese da aula

Nesta aula, conhecemos o **carro-chefe** da computação evolucionária: os algoritmos genéticos. Vimos um pouco sobre suas bases e fonte de inspiração biológica, entendendo como foi utilizada a teoria da evolução para gerar

técnicas computacionais capazes de adaptação e de solucionar problemas que, com as técnicas tradicionais, não eram resolvidos de forma satisfatória no tempo desejado. Estudamos também suas aplicações e sua estrutura de funcionamento, conversando sobre os parâmetros genéticos, os componentes básicos de um AG, os mecanismos de amostragem e as funções de seleção e aptidão. Concluindo, demos uma boa olhada na estrutura de um AGS (Algoritmo Genético Simples), seu poderio na solução de diversos problemas e também suas limitações.

Atividades

1. Explique como surgiu a inspiração para o desenvolvimento dos Algoritmos Genéticos.
2. Com relação aos AG's, é **incorreto** afirmar que:
 - a) na prática, nós podemos implementar facilmente um AG com o simples uso de *strings* de bits ou caracteres para representar os cromossomos e, com simples operações de manipulação de bits, podemos implementar cruzamento, mutação e outros operadores genéticos.
 - b) a principal inovação dos AG's, no que se refere aos métodos de busca, é a implementação de um mecanismo de seleção de soluções no qual, em curto prazo, os melhores têm maior probabilidade de sobreviver e, em longo prazo, os melhores têm maior probabilidade de ter descendência.
 - c) são caminhadas aleatórias não direcionadas, ou seja, buscas totalmente sem rumo, pois exploram informações históricas para encontrar novos pontos de busca em que são esperados melhores desempenhos.
 - d) o tamanho da população determina o número de cromossomos na população, afetando diretamente o desempenho global e a eficiência dos AG's.
3. Quanto aos Parâmetros Genéticos, podemos afirmar que:
 - a) quanto maior for a taxa de cruzamento, mais rapidamente novas estruturas serão introduzidas na população. Mas se esta for muito alta, acarretará uma convergência indesejável em virtude de crescer também a taxa de mutação.
 - b) o tipo de cruzamento a ser utilizado determina a forma como se procederá a troca de segmentos de informação entre os **casais** de cromossomos selecionados para cruzamento.
 - c) mutação é utilizada para dar nova informação para a população e também para garantir a convergência prematura, a qual demoraria demais sem a alteração desse parâmetro.

- d) uma alta taxa de mutação previne que uma dada posição fique estagnada em um valor, além de possibilitar que se chegue em qualquer ponto do espaço de busca.
4. Quando podemos utilizar um AG na solução de um determinado problema? Escreva um texto e discuta com seus colegas!

Comentário das atividades

As atividades desta aula compreendem os objetivos inicialmente propostos, que são compreender a metáfora biológica utilizada para criação dos algoritmos genéticos (atividade um e dois) e entender a estrutura e o funcionamento lógico de um algoritmo genético (atividades três e quatro).

Como resposta à **atividade um**, podemos afirmar que os AG's inspiram-se nas teorias da evolução. A inspiração para sua criação surgiu pela observação dos processos naturais de evolução e seleção natural, e sua adequação para a solução de problemas de otimização.

A resposta da **atividade dois** é a letra **(c)**, pois é incorreto afirmar que os AG's são "caminhadas aleatórias não direcionadas". As demais sentenças estão corretas, pois, como descrito na primeira, a representação nos AGs costuma ser mediante *strings* de *bits* ou caracteres, e as operações sobre essas estruturas são simples manipulações de *bits*. A segunda sentença reforça o fato dos AG's se inspirarem no processo de seleção natural para encontrar soluções para um determinado problema; e, finalmente, a quarta sentença diz que o tamanho da população determina o número de cromossomos do algoritmo, conseqüentemente, seu custo computacional (quanto mais cromossomos, mais necessidade de processamento).

Na **atividade três**, deve ser assinalada a alternativa **(b)**, uma vez que as demais alternativas incorrem em erro que contradiz as funções dos respectivos parâmetros genéticos, conforme segue: a alternativa **(a)** erra ao associar o crescimento da taxa de mutação à taxa de cruzamento. A alternativa **(c)** erra em afirmar que mutação garante a convergência prematura, fator que na verdade é indesejado em um AG; e, por fim, a alternativa **(d)** erra em colocar como correto o uso de uma alta taxa de mutação que, na verdade, acabaria tornando a busca totalmente aleatória.

Finalmente, na **atividade quatro**, devemos dizer que os AG's tendem a ser mais bem aproveitados para solucionar problemas de otimização matemática com espaços de busca limitados e discretos, que permitam a criação de uma função de aptidão. Mas nem sempre serão a melhor alternativa à solução de todos os problemas desse tipo.

Referências

- BÄCK, Thomas. **Evolutionary algorithms in theory and practice**: evolution strategies, evolutionary programming, genetic algorithms. New York: Oxford University Press, 1996.
- DAVIS, Laurence. **Handbook of genetics algorithms**. New York: Van Nostrand Reinhold, 1991.
- GOLDBERG, David E. **Genetic algorithms in search, optimization and machine learning**. Massachusetts: Addison-Wesley: Co, 1989.
- GREFENSTETTE, John **A User's guide to GENESIS**. Nashville: Vanderbilt University, 1990.
- HOLLAND, John H. **Adaptation in natural and Artificial systems**. Ann Arbor: University of Michigan Press, 1975.
- MICHALEWICZ, Z. **Genetic algorithms + data structures = evolution programs**. New York: Springer-Verlag Berlin Heidelberg, 1996.
- MITCHELL, Melanie. **An introduction to genetic algorithms**. Massachusetts: MIT Press, 1996.
- PALAZZO, Luiz A. M. **Algoritmos para computação evolutiva**. Relatório Técnico – Pelotas: Grupo de Pesquisa em Inteligência Artificial – Universidade Católica de Pelotas, 1997.
- SCHNEIDER, André M. **Algoritmo adaptativo genético para acompanhamento da trajetória de alvos móveis**. Dissertação (Mestrado) – Porto Alegre: Instituto de Informática, Universidade Federal do Rio Grande do Sul, 1998.
- SERRADA, Anselmo P. **Una introducción a la computación evolutiva**. Valladolid, Espanha: Universidad de Valladolid, 1996.

Na próxima aula

O tema da nossa próxima aula já muda um pouco o enfoque do que estudamos até agora. Iremos, nas próximas páginas, aprender sobre a teoria de agentes, o que, por um lado acaba saindo da biologia e, por outro, se formos aprofundar os estudos nessa tecnologia, acaba entrando na sociologia, no comportamento social inteligente mais especificamente.

Anotações
