

Computação Evolucionária

Evolução é um processo de otimização em que o objetivo é melhorar a habilidade de um organismo ou sistema em sobreviver em ambientes competitivos e que mudam dinamicamente.

No contexto da computação evolucionária, o foco está na evolução biológica, a partir das visões de Lamarck (1744-1829) e Darwin (1809-1882).

A teoria evolucionária Lamarckiana é baseada na hereditariedade, i.e. a herança de traços adquiridos, em que indivíduos se adaptam durante o tempo de vida e transmitem seus traços para os descendentes, que continuam a se adaptar.

O método de adaptação se baseia no conceito de uso e desuso.

Computação Evolucionária

A teoria evolucionária Darwiniana é baseada na seleção natural, em que em um mundo com recursos limitados e populações estáveis, cada indivíduo compete com outros para sobreviver.

Indivíduos com as melhores características ou traços tem maiores chances de sobreviver e reproduzir, sendo essas características passadas aos seus descendentes.

Essas características desejáveis são herdadas pelas gerações futuras e se tornam dominantes, ao longo do tempo, dentro da população.

Durante a geração de um descendente, eventos aleatórios causam mudanças aleatórias nas características do descendente: se forem benéficas, as chances de sobrevivência aumentam.

Computação Evolucionária

Computação evolucionária se refere a sistemas para a resolução de problemas, que usam modelos computacionais de processos evolucionários, tais como:

- seleção natural
- sobrevivência do mais apto
- reprodução

Evolução via seleção natural de uma população de indivíduos, escolhidos aleatoriamente, pode ser vista como uma busca pelo espaço de possíveis valores de cromossomas.

Computação Evolucionária

Um algoritmo evolucionário (*Evolutionary Algorithm* – EA) é uma busca estocástica por uma solução ótima para um dado problema.

O processo de busca evolucionário é influenciado pelos seguintes componentes de um EA:

- codificação das soluções do problema como um cromossoma
- função para avaliar a aptidão (*fitness*) ou capacidade de sobrevivência dos indivíduos
- inicialização da população inicial
- operadores de seleção
- operadores de reprodução

Computação Evolucionária

Algoritmo Evolucionário Genérico

Seja $t=0$ o contador de gerações;

Criar e inicializar uma população n_x -dimensional $C(0)$, consistindo de n_s indivíduos;

Enquanto condição de parada(s) for falsa faça

 Avaliar a aptidão $f(x_i(t))$ de cada indivíduo $x_i(t)$;

 Realizar reprodução para gerar descendentes;

 Selecionar a nova população $C(t+1)$;

 Avançar para a nova geração, i.e. $t=t+1$;

Fim

Computação Evolucionária

Cada geração de um EA é referida como uma geração.

As diferentes formas nas quais componentes do EA são implementados resultam em diferentes paradigmas de EC:

- Algoritmos genéticos (*Genetic Algorithms* – GAs), que modelam evolução genética.
- Programação genética (*Genetic Programming* – GP), que é baseada em algoritmos genéticos, mas indivíduos são programas.
- Programação evolucionária (*Evolutionary Programming* – EP), que é derivada da simulação do comportamento adaptativo na evolução (i.e. evolução fenotípica).

Computação Evolucionária

- Estratégias de evolução (*Evolution Strategies* – ESs), que são direcionadas para modelagem dos parâmetros que controlam a variação na evolução, i.e. a evolução da evolução.
- Evolução diferencial (*Differential Evolution* – DE), que é similar a algoritmos genéticos, diferindo na mecanismo de reprodução usado.
- Evolução cultural (*Cultural Evolution* – CE), que modela a evolução da cultura de uma população e como a cultura influencia a evolução genética e fenotípica dos indivíduos.
- Co-evolução (*Co-Evolution* – CoE), onde, inicialmente, indivíduos fracos evoluem via cooperação ou em competição entre si, adquirindo as características necessárias para sobreviver.

Computação Evolucionária

A teoria Darwiniana está embutida no algoritmo:

- Seleção natural ocorre via operador de reprodução, onde os melhores pais tem uma melhor chance de serem selecionados para produzir descendentes e para serem selecionados para a nova população.
- Mudanças aleatórias são aplicadas via operador de mutação.

As características dos organismos, que influenciam suas capacidades para sobreviver e reproduzir, são representadas por cadeias longas de informação contidas nos cromossomas desses organismos.

Computação Evolucionária

Cromossomas são estruturas de moléculas de DNA compactas e entrelaçadas, encontradas no núcleo das células orgânicas, em que cada um contém um grande número de genes, cada um, por sua vez, consistindo da unidade hereditária.

No contexto da EC, cada indivíduo representa uma solução candidata para um problema de otimização e as características de um indivíduo são representadas por um cromossoma, também referido como um genoma.

Essas características consistem das variáveis do problema de otimização.

Cada variável que precisa ser otimizada é referida como um gene, a menor unidade de informação, e a atribuição de um valor a uma variável é referida como um alelo.

Computação Evolucionária

A eficiência e complexidade do algoritmo de busca depende muito do esquema de representação.

Uma representação genética, também conhecida como codificação genética, descreve os elementos do genótipo e como estes são mapeados em fenótipos.

Em uma representação discreta, o indivíduo é descrito por uma sequência de l valores discretos de um alfabeto com cardinalidade k . Em GAs, normalmente é utilizado um alfabeto binário 0, 1 com cardinalidade $k=2$.

Genótipos binários podem ser mapeados em diferentes fenótipos, tais como números inteiros, números reais, cronograma, configuração de circuitos, etc.

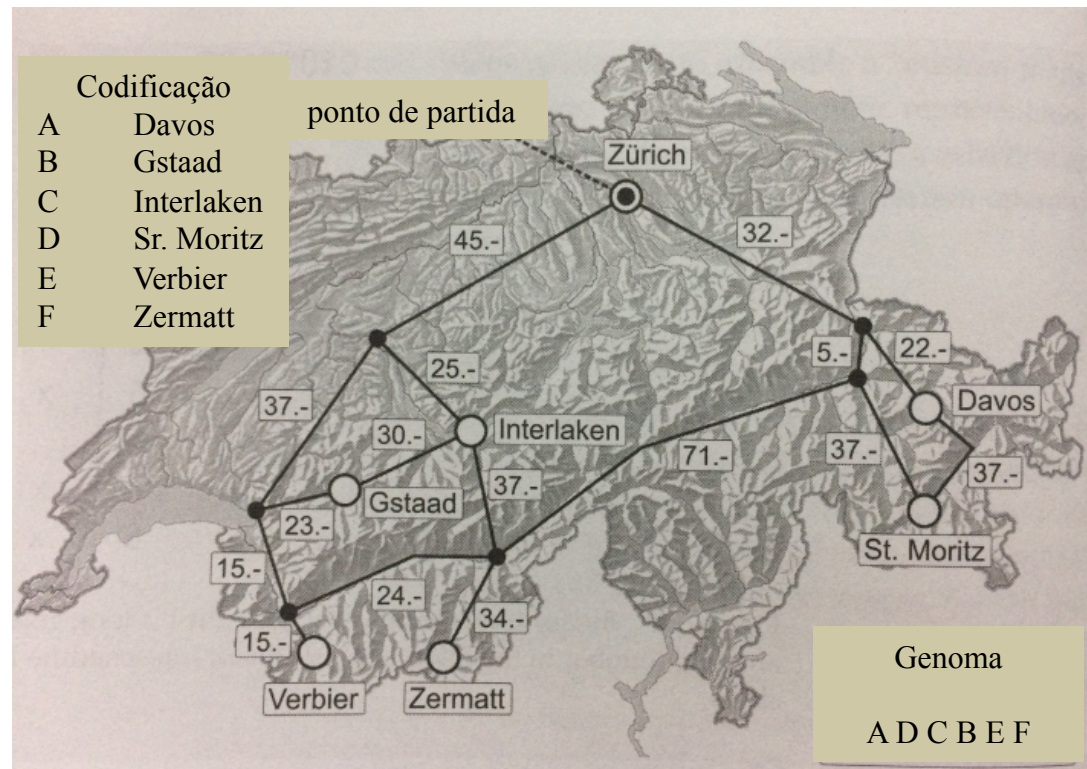
Computação Evolucionária

Ex: Considere o genótipo <01010100>.

- fenótipo inteiro $i = 0 \times 2^7 + 1 \times 2^6 + 0 \times 2^5 + 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 0 \times 2^0 = 84$
- fenótipo real r no intervalo $[min, max] = min + (i/255)(max - min) = 0,3294$
- cronograma: oito tarefas distribuídas em dois turnos, em que 0 denota turno da manhã e 1 denota turno da tarde

Computação Evolucionária

Ex: Considere o planejamento de uma viagem entre 6 cidades ($k=6$), a um custo mínimo de transporte.

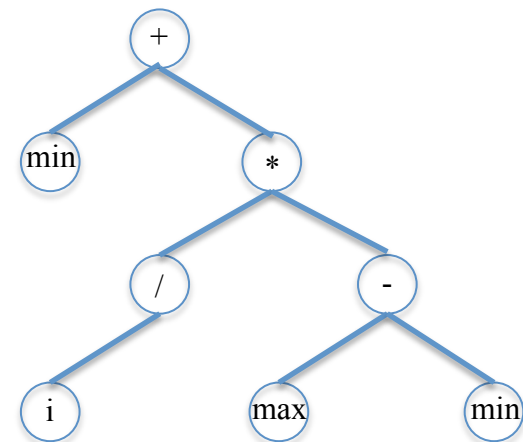


Computação Evolucionária

Em uma representação real, o genótipo consiste de um conjunto de n números do domínio dos números reais, tipicamente representados por valores em ponto flutuante, sendo adequada para soluções que requerem parâmetros de alta precisão.

Uma representação baseada em árvore é adequada para descrever estruturas hierárquicas, usadas em programação genética, onde cada indivíduo é um programa de computador.

Ex: $r = \min + (i/255)(\max - \min)$



Computação Evolucionária

A população inicial deveria ser suficientemente grande e diversificada para garantir que os indivíduos apresentem diferentes valores de aptidão, uma vez que se todos tiverem a mesma aptidão, a seleção não irá operar de forma adequada.

O tamanho da população deveria depender das propriedades do espaço de busca e do custo computacional em avaliar cada indivíduo por várias gerações.

No caso de representação binária, cada genótipo é criado através de uma sequência aleatória de 0s e 1s.

No caso de representação real, cada genótipo é criado através de uma sequência aleatória de valores reais dentro de um intervalo predefinido.

Computação Evolucionária

A função de aptidão associa uma pontuação para cada fenótipo na população.

Avaliar a aptidão dos indivíduos é normalmente a parte que consome mais tempo de um algoritmo evolucionário, em que a qualidade das soluções evoluídas depende de quão exaustiva for a avaliação dos indivíduos.

O papel da seleção é alocar uma grande quantidade de descendentes aos melhores indivíduos da população.

A pressão de seleção indica o percentual de indivíduos que irão criar descendentes para a próxima geração, para a qual um valor alto significa que uma percentagem pequena de indivíduos será selecionada para reprodução, comprometendo a diversidade.

Computação Evolucionária

É importante manter um equilíbrio entre a pressão de seleção e outros fatores que podem gerar diversidade, tal como mutação genética.

Na seleção proporcional, a probabilidade $p(i)$ que um indivíduo i faça uma cópia de seu próprio genoma é dada pela taxa entre o valor de sua própria aptidão $f(i)$ e a soma dos valores de aptidão de todos os indivíduos da população $\sum_i f(i)$:

$$p(i) = \frac{f(i)}{\sum_i^N f(i)}$$

Logo, o número esperado de descendentes de um indivíduo i é $Np(i)$, onde N é o tamanho da população.

Computação Evolucionária

A seleção pela roleta associa um indivíduo a cada setor e o tamanho do setor é diretamente proporcional à probabilidade de reprodução $p(i)$ do indivíduo: uma bolinha é lançada e o cromossoma onde ela para é selecionado.

A fim de recriar uma população de N indivíduos, a roleta é girada N vezes.

Algoritmo Seleção da Roleta

Seja $i=1$;

$soma=p(i)$;

Escolha um número aleatório r no intervalo $[0,1]$;

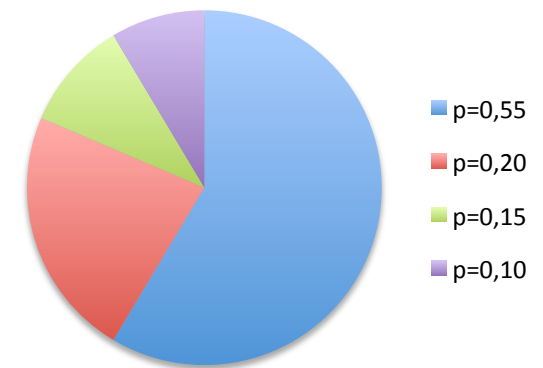
Enquanto $soma < r$ faça

$i=i+1$;

$soma=soma+p(i)$;

End;

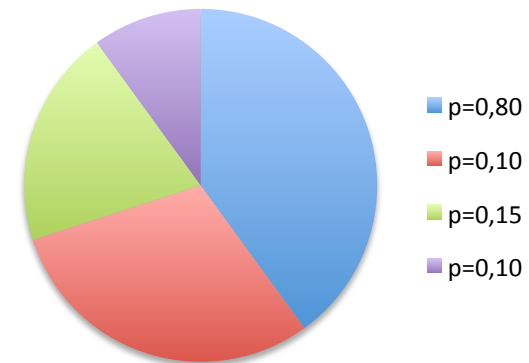
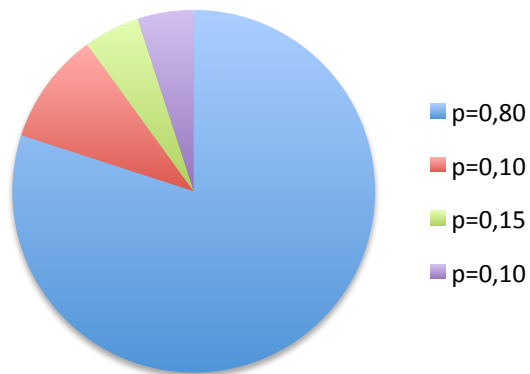
Retorna i .



Computação Evolucionária

Seleção por classificação consiste em classificar os indivíduos da população e alocar probabilidades de reprodução proporcional à classificação: menor probabilidade recebe 1 e maior probabilidade recebe N .

Não interessa o quão pequena seja a diferença entre dois indivíduos, o melhor deles terá sempre a maior probabilidade de gerar descendentes.



Computação Evolucionária

Seleção por classificação truncada consiste em pegar somente os n indivíduos na lista de classificação e gerar o mesmo número de descendentes para cada indivíduo selecionado.

Suponha uma população de 100 indivíduos, de onde selecionamos os 20 melhores e geramos 5 cópias de cada um a fim de criar a nova população.

Assumindo que n não seja muito pequeno, esse método garante que os indivíduos que obtiveram baixa pontuação de aptidão, mas ainda mais alta que os piores, gerarão a mesma quantidade de descendentes que os melhores indivíduos.

Computação Evolucionária

Seleção por torneio consiste em selecionar k indivíduos da população, onde k é um número aleatório associado ao tamanho do torneio.

O indivíduo com a melhor aptidão dentre os k gera então um descendente e os k indivíduos voltam para a população, podendo participar de novo torneio, que é organizado para cada descendente a ser gerado.

Seleção por torneio alcança um bom compromisso em manter tanto a pressão de seleção quanto a diversidade genética na população.

Computação Evolucionária

Na substituição geracional, os novos descendentes substituem toda a população antiga de indivíduos.

Para evitar que bons indivíduos sejam perdidos em gerações futuras, uma estratégia, denominada elitismo, consiste em manter o n melhores indivíduos da população anterior.

Também é possível relaxar um pouco na substituição geracional completa, inserindo somente uns poucos descendentes na população no lugar de indivíduos com pior aptidão.

Computação Evolucionária

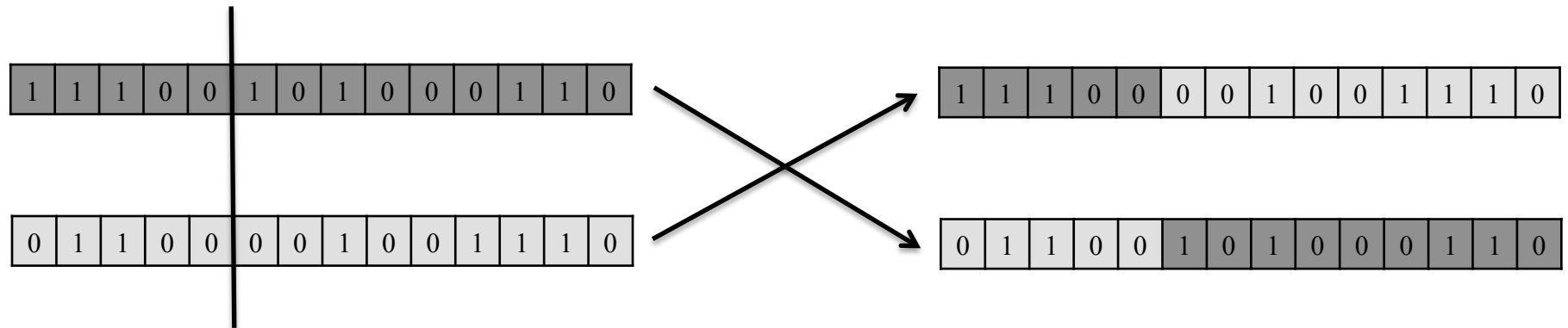
Operadores genéticos introduzem diversidade na população e permitem a exploração de novas soluções.

Cruzamento (*crossover*) ou recombinação garante que descendentes herdem características dos pais criando uma recombinação em pares do genoma dos indivíduos selecionados.

Os descendentes recentemente gerados são aleatoriamente emparelhados e partes de seus genótipos são trocadas pelo operador de cruzamento com uma probabilidade p_c .

Computação Evolucionária

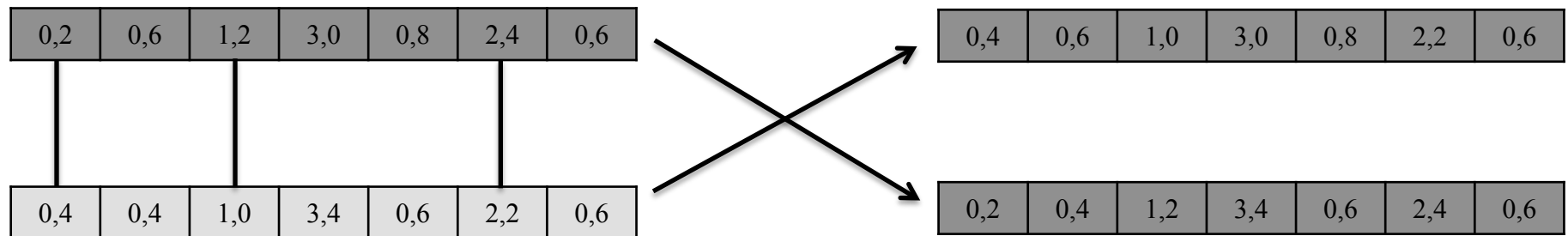
Cruzamento de um ponto consiste em selecionar aleatoriamente um ponto de cruzamento em cada uma das duas cadeias e trocar material genético entre os indivíduos em torno desse ponto.



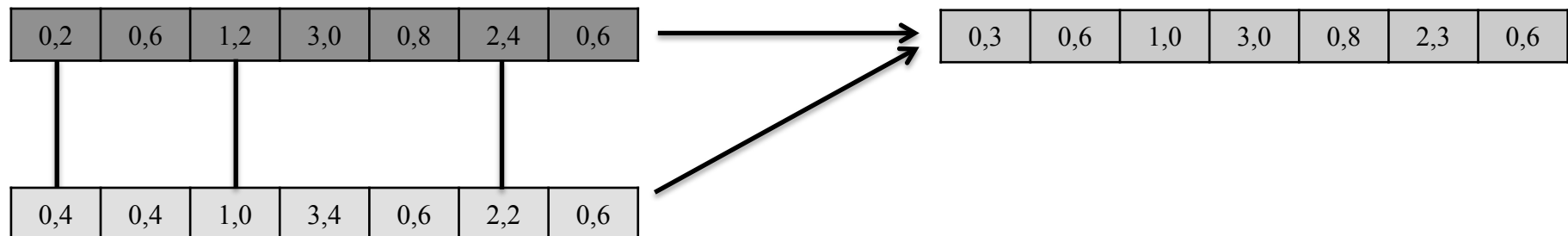
Cruzamento de vários pontos consiste em selecionar aleatoriamente n pontos de cruzamento nas duas cadeias e trocar material genético que recaiam entre esses pontos.

Computação Evolucionária

Cruzamento uniforme consiste em trocar material genético entre posições escolhidas aleatoriamente.



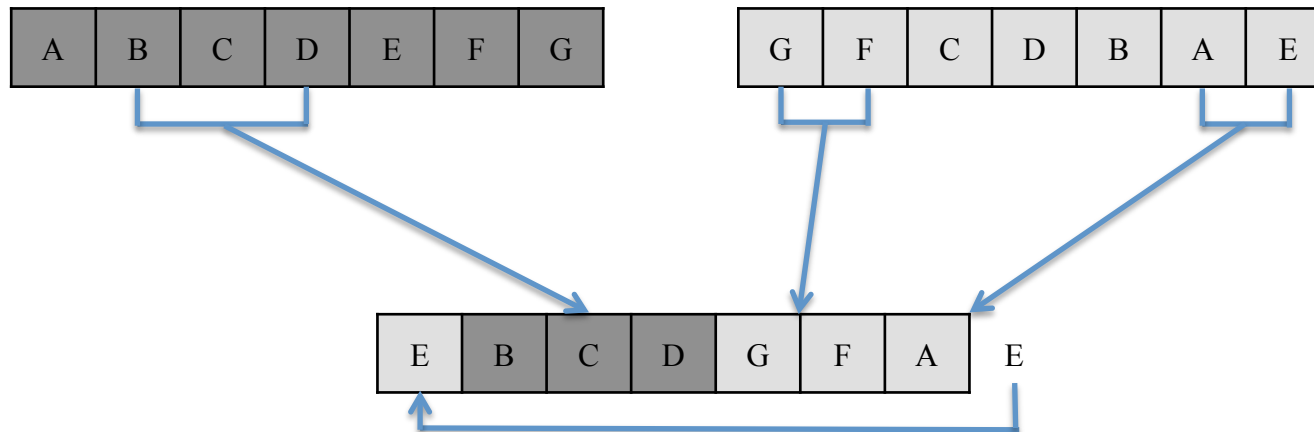
Cruzamento aritmético cria um único genótipo a partir da média de n posições aleatórias de duas cadeias genéticas.



Computação Evolucionária

Quando o genótipo representa uma sequência, o operador de cruzamento deve respeitar as restrições, como no problema do caixeiro viajante.

O operador de cruzamento cria um genótipo pegando aleatoriamente uma parte de uma cadeia e preenchendo os campos restantes com os alfabetos restantes na ordem em que aparecem na outra cadeia.



Computação Evolucionária

Mutação ocorre à nível do indivíduo, consistindo de pequenas modificações aleatórias do genótipo que permitem que a evolução explore variações de soluções existentes.

Uma mutação consiste em mudar o conteúdo de cada posição do genótipo com probabilidade p_m .

Na representação binária, mutação consiste em trocar o valor do bit selecionado.

1	1	1	0	0	1	0	1	0	0	0	1	1	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---

1	1	0	0	0	1	0	1	1	0	0	1	1	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---

Computação Evolucionária

Na representação real, uma posição selecionada é modificada adicionando um valor aleatório obtido de uma distribuição Gaussiana $N(0,\sigma)$, onde 0 é a média e σ é a variância, a fim de produzir poucas mutações grandes.

0,2	0,6	1,2	3,0	0,8	2,4	0,6
-----	-----	-----	-----	-----	-----	-----

0,2	0,7	1,2	3,0	0,8	2,2	0,6
-----	-----	-----	-----	-----	-----	-----

Na representação em sequência, mutação consiste em trocar os conteúdos de duas posições escolhidas aleatoriamente no genótipo do indivíduo.

G	F	C	D	B	A	E
---	---	---	---	---	---	---

G	F	C	E	B	A	D
---	---	---	---	---	---	---