

fig 4.7 → Versão final do nosso GA mais simples

É claro que alguém poderia inventar similares computacionais destas características naturais. Se isto for feito, provavelmente os GAs tornar-se-ão não só ferramentas de busca mais eficientes como também serão uma notável analogia das sociedades naturais, podendo até ser objeto de estudos sociológicos²⁹.

O processo também não é de todo empírico, como pode parecer até agora. Existe uma teoria razoavelmente sólida por trás dos GAs, teoria esta que omitimos até agora por questões de conveniência, mas que veremos no capítulo 5 a seguir.

4.8) Uma execução manual

Para que possamos entender melhor o funcionamento dos algoritmos genéticos, vamos tentar resolver, usando um GA, o problema de maximizar a função do exemplo 4.1, dada por $f(x, y) = \left| x * y * \text{sen}\left(\frac{y\pi}{4}\right) \right|$, com x e y pertencentes ao intervalo $[0,15]$ ³⁰. Como é possível que esta função retorne um valor igual a zero, usaremos uma função de avaliação $g(x, y) = 1 + f(x, y)$. Como discutiremos com detalhes na seção 4.9.a, funções que retornem valores negativos ou zero geram espaços da roleta que nunca serão sorteados e, por conseguinte, devem ser evitadas.

Dado o intervalo dos valores, são necessários 4 bits para cada variável, o que implica em um cromossomo de 8 bits, e vamos utilizar uma taxa de mutação de 1%. Para que o tamanho do exemplo seja adequado para um livro, vamos manter uma população de apenas 6 indivíduos.

A população inicial, sorteada aleatoriamente, consiste dos seguintes indivíduos:

Cromossomo	x	y	g(x,y)
01000011	4	3	9,5

²⁹ É claro que isto é apenas uma digressão, mas a idéia é bastante interessante para merecer uma menção aqui!

³⁰ Antes tínhamos considerado valores reais no intervalo $[-100,100]$. Entretanto, representar cromossomos de 44 bits iria tomar muito espaço. Para simplificar o exemplo e facilitar sua compreensão, vamos trabalhar com cromossomos menores e valores inteiros. Entretanto, o raciocínio aplicado aqui vale para qualquer tipo de cromossomo.

00101001	2	9	13,7
10011011	9	11	71,0
00001111	0	15	1,0
10011001	5	5	18,7
11100011	14	3	30,7
<i>Somatório das avaliações:</i>			144,6

A roleta para esta população é dada na figura 4.8.

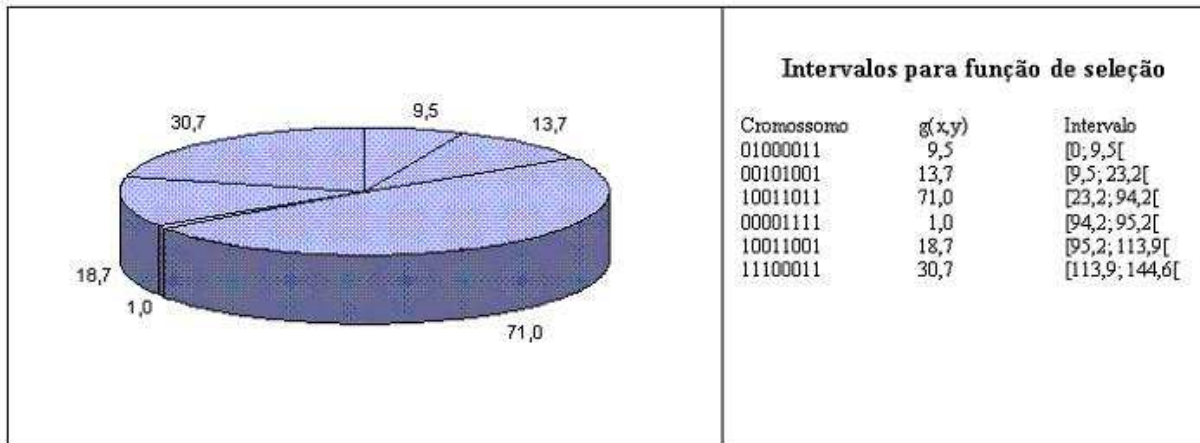


Fig. 4.8 → Roleta completa para a população da primeira geração do exemplo corrente

Sorteamos então os indivíduos que gerarão a próxima geração. Para tanto sorteamos 6 números entre 0 e a soma das avaliações (144,6). Os números sorteados e os indivíduos que são escolhidos a partir dele são os seguintes:

Número Sorteado	Cromossomo Escolhido
12,8	00101001
65,3	10011011
108,3	10011001
85,3	10011011
1,8	01000011
119,5	11100011

Como podemos ver pelo sorteio, não existe nenhuma obrigação de que todos os pais sejam selecionados pelo menos uma vez. O terceiro cromossomo (10011011) foi sorteado duas vezes, participando de dois crossovers. Poderíamos até ter uma situação em que houvesse um crossover no qual os dois pais são iguais. Isto é indesejável, pois, a não ser que o operador de mutação atuasse, necessariamente os dois filhos seriam iguais aos pais e estaríamos diminuindo a diversidade da população, mas é uma situação possível dentro do sorteio.

Fazendo o crossover em todos os pares de pais, temos os resultados demonstrados na figura 4.9.

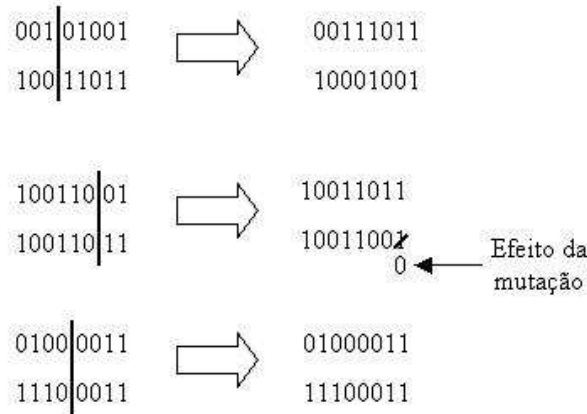


Fig. 4.9 → Operadores genéticos aplicados aos pais selecionados na primeira geração do exemplo corrente.

Os pontos de corte foram selecionados de forma aleatória e algo extremamente interessante nos filhos gerados é o fato de que a terceira reprodução gera, por coincidência, filhos iguais aos pais. Não existe nenhuma objeção de cunho teórico a esta situação que pode ocorrer no crossover de um ponto quando os pais têm um prefixo ou um sufixo em comum.

Para cada um dos bits foi sorteado um número entre 0 e 99, para a realização da mutação. Se o número sorteado fosse zero, o bit seria invertido. Caso contrário, o bit seria mantido. Este sorteio dá exatamente uma chance em cem do bit ser invertido, o que corresponde a uma taxa de mutação de 1%.

Um ponto interessante a ressaltar em relação a este operador de mutação é que muitos trabalhos, ao invés de inverter o bit ao decidir aplicar o operador de mutação, sorteiam um número zero ou um para colocar na posição em questão. Isto faz com que a taxa efetiva de mutação seja metade da taxa adotada, pois, em média, em metade dos sorteios o bit sorteado será igual àquele previamente existente na posição.

A nova geração e suas avaliações são então as seguintes:

Cromossomo	x	y	g(x,y)
00111011	3	3	7,4
10001001	8	9	51,9
10011011	9	11	71,0
10011000	9	8	1,0
01000011	4	3	9,5
11100011	14	3	30,7
<i>Somatório das avaliações:</i>			171,5

Um ponto interessante em relação a esta segunda geração é que, apesar do melhor cromossomo não ter uma avaliação melhor do que o melhor cromossomo da geração anterior, a avaliação média dos cromossomos subiu. Antes, a avaliação média era de $\frac{144,6}{6} = 24,1$, e agora, na segunda geração, a avaliação média subiu para $\frac{171,5}{6} = 28,6$.

A roleta desta nova população é dada na figura 4.10.

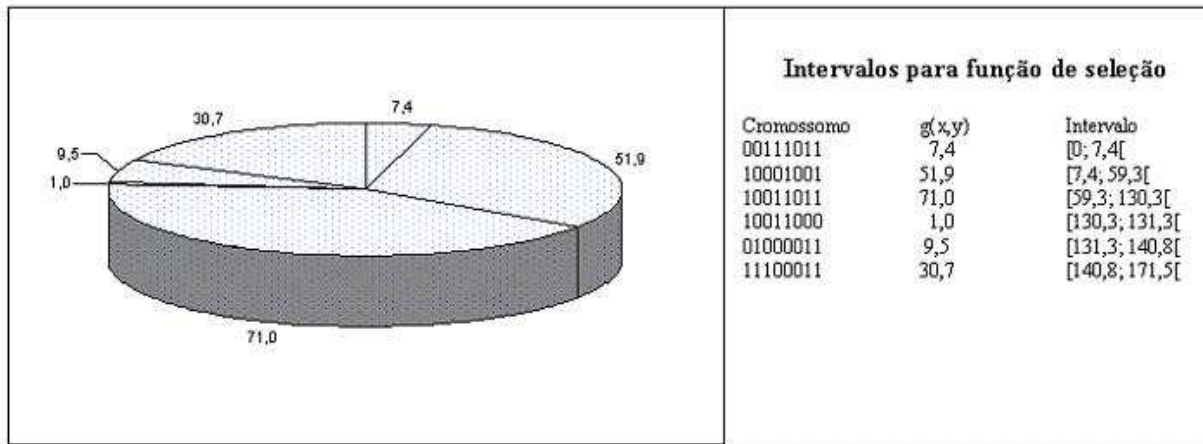


Fig. 4.10 → Roleta completa para a população da segunda geração do exemplo corrente. Note-se que o total da soma das avaliações de todos os indivíduos aumentou, indicando que, em média, esta população é mais adaptada ao problema do que a geração anterior.

Sorteamos então os indivíduos que produzirão a próxima geração. Para tanto sorteamos 6 números entre 0 e a soma das avaliações (171,5). Os números sorteados e os indivíduos que são escolhidos a partir dele são os seguintes:

Número Sorteado	Cromossomo Escolhido
10,4	10001001
132,5	01000011
61,2	10011011
148,6	11100011
129,7	10011011
75,2	10011011

Aconteceu de um dos elementos desta geração ser selecionado três vezes, sendo que duas destas foram dentro do mesmo processo reprodutivo. Isto pode acontecer porque o processo é aleatório. Se for desejado, pode-se estabelecer um controle para evitar esta repetição, pois estaremos perdendo diversidade na população e nos encaminhando para a ocorrência de convergência genética.

Fazendo o crossover em todos os pares de pais, temos os seguintes resultados:

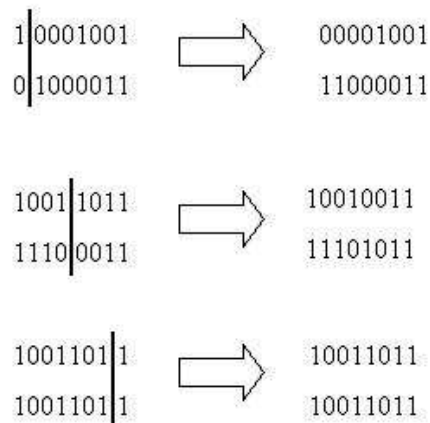


Fig. 4.11 → Operadores genéticos aplicados aos pais selecionados na segunda geração do exemplo corrente.

Mais uma vez os pontos de corte foram selecionados de forma aleatória. Algo que deve ser notado em relação aos resultados é que os pais da terceira reprodução são iguais, um ao outro. Logo, qualquer ponto de corte que seja selecionado para ambos gerará filhos iguais. Isto faz com que a população perca diversidade, pois agora temos apenas 5

indivíduos diferentes, contra seis da geração anterior. Este efeito, de convergência genética, e é muito comum em populações que realizam cruzamentos endógenos. Existem algumas maneiras de minimizar este efeito, modificando os operadores genéticos ou as técnicas de seleção de pais. Ao longo deste livro nós sempre procuraremos discutir, para todas as técnicas descritas, o efeito que elas causam sobre o fenômeno da convergência genética.

A mutação foi feita da mesma maneira que na geração anterior, só que para nenhum dos filhos gerados obtivemos o resultado igual a zero no sorteio, o que fez com que nenhum dos bits sofresse uma mutação. Criamos então a terceira geração de cromossomos, que tem as seguintes características:

Cromossomo	x	y	g(x,y)
00001001	0	9	1,0
11000011	3	3	7,4
10010011	9	3	20,1
11101011	14	11	109,9
10011011	9	11	71,0
10011011	9	11	71,0
<i>Somatório das avaliações:</i>			280,4

Mais uma vez a avaliação global da população aumentou e desta vez também tivemos uma melhora do indivíduo melhor avaliado. Poderíamos continuar o processo, mas já é possível ter um quadro bastante claro de como o processo acontece.

Neste momento você poderia achar que o algoritmo só funcionou porque o sorteio foi direcionado, isto é, que ao calcular o exemplo eu fui “malandro”, para fazer com que o GA desse certo. Esta é uma dúvida extremamente razoável neste ponto (e se jurarmos que não o fizemos, você pode não acreditar) e você só a apagará se executar os códigos deste capítulo e ver o que tudo que fizemos aqui realmente acontece.

4.9) Discussões Adicionais

4.9.a) Método da roleta viciada

Quando montamos uma roleta viciada para uma determinada população, somamos todas as avaliações e para cada indivíduo alocamos um espaço igual à avaliação deste indivíduo dividida pela soma das avaliações de todos os indivíduos. O que aconteceria então se tivéssemos um ou mais indivíduos com avaliação negativa? A resposta é: a soma total ainda seria 360°, mas a soma dos espaços alocados apenas para os de avaliação positiva excederia 360° (isto é, a soma dos pedaços seria maior que o círculo) e ainda teríamos que lidar com o problema de alocar um espaço negativo para o indivíduo em questão.

Exemplo 4.4: Imaginemos uma situação hipotética, com uma função de avaliação $f(x)=x$ e o domínio da função f é o intervalo $[-20,20]$.

<i>Indivíduo</i>	<i>Avaliação $f(x)=x$</i>	<i>Pedaço da roleta (%)</i>	<i>Pedaço da roleta (°)</i>
1	1	6,25	22,5
-5	-5	-31,25	-112,5
20	20	125	450
<i>Total</i>	<i>16</i>	<i>100.00</i>	<i>360.0</i>

Se olhássemos apenas para os valores totais (última linha da tabela) tudo pareceria estar OK. Afinal, a soma dos elementos em termos percentuais atinge 100% e em termos de graus atinge 360° (circunferência completa). Qual é o problema então? Repare que a soma dos valores positivos (marcados em negrito) excede 360° (na verdade eles chegam a 472,5°, que consiste na soma de 360° com o módulo do espaço ocupado pelo valor negativo). Como podemos

alocar este espaço para eles? Estaríamos dando mais de uma volta pra depois retroceder quando encontrássemos o valor negativo? Isto é impraticável em termos lógicos e de implementação.

Dado que escolhermos sempre um valor entre 0 e 16 para rodar a nossa roleta, o elemento com valores negativos nunca seria escolhido, uma vez que a soma dos valores de avaliação até ele são sempre menores que o valor da soma até seu antecessor. Logo, se o antecessor não foi escolhido, então não há jeito do indivíduo com avaliação negativa ser escolhido. Este conceito fica mais compreensível com um exemplo prático. Rodei a roleta e escolhi um valor igual a 3. A avaliação do primeiro é 1, logo ele não foi escolhido. A soma da avaliação do primeiro com o segundo é -4, que também é menor que 3, logo ele também não pode ser escolhido. Vocês podem testar com todos os números positivos no intervalo [2,16] e verão que não há jeito do indivíduo de avaliação negativa ser selecionado.

Vamos resolver isto então usando a técnica descrita na seção anterior (somar uma constante maior que zero ao valor da função de avaliação). Logo vamos usar a função de avaliação $f(x)=x+6$. O valor 6 foi escolhido porque seu módulo é maior do que o módulo do maior valor negativo da função de avaliação. Agora o quadro para roleta fica assim:

<i>Indivíduo</i>	<i>Avaliação $f(x)=x+6$</i>	<i>Pedaço da roleta (%)</i>	<i>Pedaço da roleta (°)</i>
1	7	20,6	74,1
-5	1	2,9	10,6
20	26	76,5	275,3
<i>Total</i>	<i>34</i>	<i>100.00</i>	<i>360.0</i>

Uma vez que nenhum valor é negativo, todos os pedaços são menores que a roleta como um todo e podemos aplicar sem problemas o nosso método.

Existem alguns casos particulares que devem ser considerados quando analisamos o método da roleta viciada, como por exemplo o fato de que ele tem um problema sério quando existe dentro da população um **superindivíduo**, que é definido como aquele que tem uma avaliação muito superior à média do resto da população.

Exemplo 4.5: Imagine que temos os seguintes indivíduos em nossa população:

Indivíduo	Avaliação
x_1	20
x_2	30
x_3	40
x_4	10
x_5	9900

O indivíduo x_5 é um superindivíduo. Sua avaliação é extremamente superior a todas as outras e a área que ele ocupará na roleta equivalerá a 356,4°, o que significa que, pela lei das probabilidades, o indivíduo x_5 será selecionado 99% das vezes, se usarmos o método da roleta, e pode ser que uma geração inteira de pais se resuma a este indivíduo após uma seleção, o que vai acabar com a diversidade da população, causando uma convergência genética prematura. Isto pode ser minimizado com algum dos métodos expostos no capítulo 8, que descreve modificações na função de avaliação.

Outro problema é o fato de que, como baseamos este método totalmente em sorteios aleatórios, é possível que os melhores indivíduos de uma população nunca sejam selecionados. Este tipo de situação não é contornável, mas, usando elitismo (veja capítulo 7) podemos fazer com que os melhores cromossomos não morram de uma geração para outra, o que dará a eles uma nova chance de se reproduzir.

Este método é, talvez, o mais usado na literatura, mas não necessariamente aquele que fornece os melhores resultados possíveis. No capítulo 9 discutiremos outros métodos de seleção de pais e como o uso destes métodos alternativos pode melhorar o desempenho global do GA.

4.9.b) Função de avaliação

Como discutimos na seção 4.9.a, é muito importante que a função de avaliação tenha um contradomínio estritamente positivo, isto é, que nenhum indivíduo da população tenha avaliação negativa ou zero. Isto faria com que a soma das avaliações diminuísse, impedindo que o módulo de seleção de pais tenha um desempenho adequado. A maneira de fazer isto é extremamente simples. Se a menor das funções $f(x)$ é $-c$, basta tentarmos maximizar a função $f'(x)=f(x)+c'$, onde $c'>c$. Se a menor avaliação for igual a zero, basta fazer com que a constante c' seja igual a 1. Isto garante que nenhuma avaliação pode ser igual ou menor a zero.

Exemplo 4.6: Imagine agora que nosso problema seja encontrar uma palavra de 5 letras procurando por todas as combinações possíveis de letras do alfabeto (o que nos rende um espaço de busca de tamanho 23^5). Seja a solução para nosso problema a palavra *FELIZ*. Uma função de avaliação que poderíamos usar seria tal que a palavra *FELIZ* tivesse avaliação igual a 1 e todas as outras palavras teriam avaliação igual a zero. O problema desta função é que ela não diferencia entre soluções sub-ótimas – por exemplo, *FELIX* teria avaliação igual a *PEQRT*, apesar da primeira ser muito mais próxima da solução ótima do que a segunda. Uma outra função de avaliação que poderíamos usar seria considerar o número de letras acertadas. Neste caso, *FELIX* teria avaliação igual a 4 e *PEQRT*, 1 (a segunda letra), demonstrando que a primeira palavra candidata é mais próxima da solução ideal que a segunda. Neste caso, palavras que não tenham nenhuma letra na posição correta teriam avaliação igual a zero, não sendo nunca selecionadas. Para permitir que sejam selecionadas, basta que somemos um à avaliação de todas as palavras, fazendo com que a avaliação de todas vá de 1 a 6.

Se ocorrerem avaliações negativas, existem outras técnicas além do deslocamento das avaliações. Estas outras técnicas serão descritas mais adiante, no capítulo 6.

É interessante observar que a função de avaliação não é necessariamente uma função real a coeficientes reais. Ela pode ser uma função discreta ou até mesmo uma função de inteiros. Veremos mais adiante, no capítulo 16, um GA aplicado a escalonamento de horários de salas de aula em faculdades, e a função de avaliação deste GA corresponde a uma função de quantos alunos foram deixados sem sala (um número inteiro com certeza, pois 0,5 alunos nada significa, já que não podemos deixar apenas a perna de um aluno de fora de uma sala).

Outro ponto a ser levado em consideração é o fato de que os algoritmos genéticos são técnicas de maximização. É difícil alterar os métodos de seleção de pais usados para selecionar com mais frequência aqueles indivíduos que possuem uma avaliação menor. A melhor maneira de fazer isto, caso desejemos encontrar um elemento que minimize uma função é invertendo a função de avaliação de interesse ($f(x)$) e maximizando a função $g(x)=1/f(x)$. Neste caso, nós temos que nos preocupar com o caso em que $f(x)=0$. Caso este ponto esteja presente no contradomínio da nossa função, usaremos a mesma técnica descrita acima: maximizar a função $h(x)=1/(f(x)+c)$, onde c é uma constante real positiva qualquer.

Exemplo 4.7: Imagine que temos o problema de resolver um sistema de equações não lineares de 4 variáveis, dado por:

$$\left\{ \begin{array}{l} x^2 + y^3 + z^4 - w^5 = 0 \\ x^2 + 3z^2 - w = 0 \\ z^5 - y = 0 \\ x^4 - z + yw = 0 \end{array} \right. . \text{ Qual função de avaliação podemos usar para avaliar a qualidade dos cromossomos?}$$

É preciso entender primeiro que existem várias possíveis funções que se adequam ao problema, cada uma delas com suas características positivas e negativas. Uma idéia é calcular o erro absoluto cometido pela solução codificada no cromossomo sendo avaliado. Por exemplo, imagine que o cromossomo contém a representação binária para $x=1$; $y=-1$; $z=1$; $w=1$. Substituindo os valores nas equações obtemos:

$$\left\{ \begin{array}{l} 1^2 + (-1)^3 + 1^4 - 1^5 = 0 \\ 1^2 + 3 * 1^2 - 1 = 3 \\ 1^5 - (-1) = 2 \\ 1^4 - 1 + -1 * 1 = -1 \end{array} \right. , \text{ e o erro absoluto é dado por}$$

$$\varepsilon = |0-0| + |3-0| + |2-0| + |-1-0| = 6.$$

O primeiro problema desta função é o fato de que quanto maior o erro pior o indivíduo - logo este é um problema de minimização. A sugestão que fizemos no texto acima pode ser usada, isto é, podemos usar a inversa da função determinada, o que faria com que nossa função de avaliação fosse dada por $1/\varepsilon$.

Temos então um novo problema: no caso da solução perfeita, o erro é igual a zero, o que nos obrigaria a calcular o valor $1/0$, causando um erro de execução em qualquer programa. Podemos resolver este problema modificando nossa função de avaliação para o seu valor final, dado por $f(x, y, z, w) = \frac{1}{\varepsilon + 1}$, onde ε é o erro absoluto cometido ao substituir as variáveis na equação. Esta não é a única solução possível, mas para efeitos didáticos, será a única explorada neste texto.

A função de avaliação ideal, além de todas as características citadas até aqui, deve ser computada rapidamente. Isto é importante devido ao fato de que um GA costuma avaliar milhares de cromossomos durante sua execução. Logo, se a computação da função de avaliação for muito lenta, o GA pode demorar excessivamente para obter o resultado final.

Neste caso, todos os números representados são parâmetros de uma única função que queremos maximizar. Entretanto, é possível também que uma função de avaliação tenha múltiplos objetivos a alcançar. No exemplo do circuito citado acima, o engenheiro pode estar preocupado não só com a melhoria do desempenho, mas com o calor dissipado pelo circuito, sua taxa de distorção e vários outros fatores que devem ser considerados. Assim, a função de avaliação deve aprender a priorizar entre os vários objetivos de forma a refletir os níveis de otimização/piora em cada critério que o engenheiro está disposto a aceitar. Uma abordagem comum é criar pesos de forma a gerar um único número real que reflita a qualidade geral da solução (Koza, 2003). Nós veremos mais detalhes em seguida, na seção 14.1.

Ao final desta seção, você pode ficar um pouco decepcionado pelo fato de não serem oferecidas “receitas de bolo” para a obtenção de uma função de avaliação adequada para cada problema a ser enfrentado. Esta falta de receitas decorre do fato de que a determinação da função de avaliação é uma ação complexa que depende totalmente do problema que precisa ser resolvido. O importante é manter em mente os princípios citados aqui e buscar uma função que demonstre quão boas são as soluções representadas pelos cromossomos. É possível que você precise de algumas tentativas antes de encontrar a função ideal para o seu problema.

4.10) Exercícios Resolvidos

- 1) Quantos pontos de corte tem um cromossomo de 9 bits?

Todo cromossomo de k bits tem exatamente k-1 pontos de corte. Logo, um cromossomo de 9 bits tem 8 pontos de corte. Pense nos pontos de corte como sendo o espaço entre dois genes. Se temos dois genes, temos um espaço entre os dois. Se temos três genes “enfileirados”, temos o espaço entre o primeiro e o segundo e o espaço entre o segundo e o terceiro, resultado em dois pontos de corte e assim por diante.

Alguns autores gostam de permitir k pontos de corte, o que faz com que seja possível que os filhos sejam idênticos aos pais, se o k-ésimo ponto de corte seja sorteado. Isto é uma alternativa possível, mas a população perde variedade ainda mais rápido, especialmente se os cromossomos forem curtos, pois a probabilidade de serem copiados exatamente para a próxima geração é de 1/k (probabilidade do k-ésimo ponto de corte ser selecionado).

2) É possível, usando o crossover de um ponto e tendo os pais 111111 e 000000, gerar o filho 001100?

Não. No crossover de um ponto, selecionamos um ponto de corte k , e cada filho é composto pelo prefixo de tamanho k de um pai e do sufixo de tamanho $n-k$ do outro pai, onde n é o tamanho dos pais.

O filho 001100 contém um prefixo e um sufixo de um pai entremeados por uma subpalavra do outro pai, logo, não segue a regra descrita no parágrafo acima. Para gerar este filho precisamos de um operador de crossover mais poderoso, que será descrito mais à frente, no capítulo 6.

3) Quero fazer um GA para otimizar um parâmetro real dentro do intervalo $[-1,1]$ com precisão de 0,001. Se escolher usar uma representação binária, quantos bits deve ter meu cromossomo?

Lembrando da fórmula de conversão de binário para real:

$$real = inf_i + \frac{sup_i - inf_i}{2^k - 1} * r_i$$

Nesta fórmula, r_i representa o número inteiro que é representado pelo cromossomo binário. Temos que $sup_i - inf_i = 1 - (-1) = 2$ e queremos que a diferença entre dois valores consecutivos seja igual a 0,001. Logo, temos:

$$inf_i + \frac{sup_i - inf_i}{2^k - 1} * (r_i + 1) - \left[inf_i + \frac{sup_i - inf_i}{2^k - 1} * (r_i) \right] = 0,001$$

Logo:

$$\frac{2}{2^k - 1} * (1) = 0,001 \Rightarrow \frac{2}{0,001} + 1 = 2^k \Rightarrow k = \lceil \log_2 2001 \rceil = 11$$

Porque foi feito o arredondamento para cima. A resposta é simples: o logaritmo vai retornar um número quebrado ($\approx 10,96$) e é impossível ter um número não inteiro de bits. Se optássemos por arredondar para baixo, não atingiríamos a precisão desejada (faça as contas!). Logo, optamos por arredondar o resultado para cima, e chegamos ao número necessário de 11 bits.

Este exemplo demonstra que o número de bits escolhidos para um cromossomo não é aleatório, mas sim função da precisão desejada. Entretanto, se você precisa otimizar parâmetros reais, seria interessante que você optasse por uma representação mais natural, que embutisse os conceitos associados a parâmetros reais de forma natural, ao invés de impor uma representação só porque você está acostumado com ela. Assim, seria mais interessante usar um cromossomo real, como descrito no capítulo 10.

4.11) Exercícios

- 1) Realize os seguintes crossovers de um ponto
 - a) 000111 e 101010 com ponto de corte=4
 - b) 11011110 e 00001010 com ponto de corte=1
 - c) 1010 e 0101 com ponto de corte=2