



Algoritmos Genéticos - Capítulo 10

Desvantagens da Representação Binária

Prof. Ricardo Linden

Representação

- ✴ Representação cromossomial é fundamental para um algoritmo genético;
 - ✴ É a maneira básica de traduzir a informação do nosso problema em uma maneira viável de tratamento pelo computador ;
 - ✴ É completamente arbitrária;
 - ✴ Maioria dos pesquisadores usa representação binária pois ela é a mais simples e tem sido a mais freqüentemente usada

Representação binária

- ✦ Representação binária tem dificuldades com múltiplas dimensões de variáveis contínuas, especialmente quando uma grande precisão é requerida.
 - Grande número de bits será necessário para atingir tal precisão;
 - Cromossomos se tornarão extremamente grandes, dificultando a operação do GA;
 - Há uma discretização inerente nos valores reais quando cromossomos binários são usados;
 - Podemos ignorá-la quando usamos bits suficientes;
 - Esta quantidade pode fazer com que nossos cromossomos se tornem grandes demais;

Representação binária

- ✱ Se número de estados distintos para um parâmetro não é um múltiplo de dois, teremos que usar um número de bits igual a:

$$\lceil \log_2 n \rceil$$

, onde n é o número de parâmetros;

- ✱ Isto gera um excesso de estados igual a:

$$2^{\lceil \log_2 n \rceil} - n$$

, que:

- ✱ ou se tornarão inválidos, necessitando tratamento especial,
- ✱ ou serão redundantes aumentando a probabilidade de um determinado parâmetro, tornando a roleta ainda mais viciada;
- ✱ Este efeito é normalmente ignorado, mas pode ser problemático.

Outras questões da representação binária

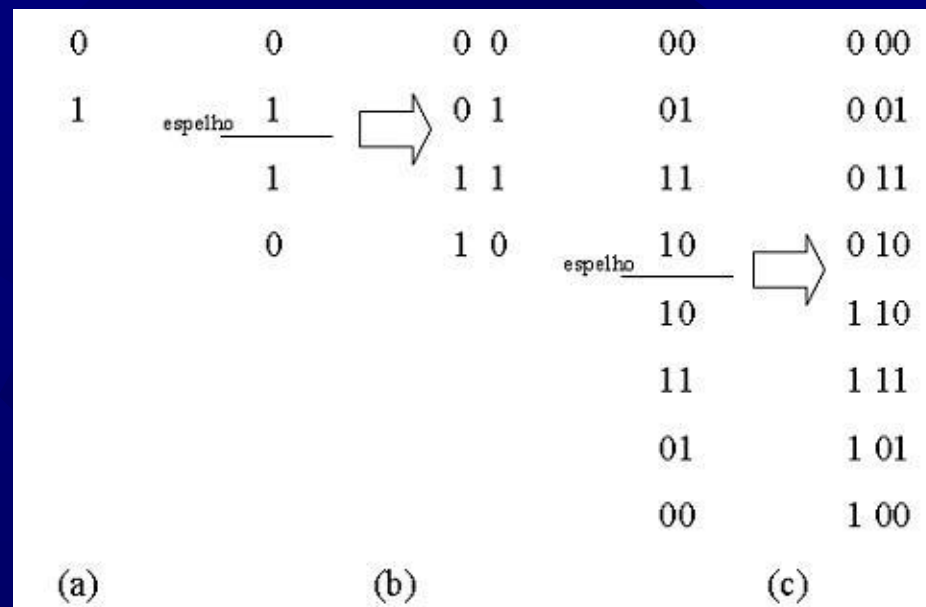
- ☀ Alfabetos de baixa ordem (com poucos símbolos) são mais eficientes para representar esquemas do que alfabetos de alta ordem;
 - Como nenhum alfabeto pode ter menos do que dois símbolos, então a codificação binária é a melhor para manipular esquemas de forma eficiente!
- ☀ Existem argumentos favoráveis a alfabetos de alta ordem:
 - Menos gerações para conformidade da população;
 - Redução de oportunidades para ocorrência de problemas enganadores;
 - Evitar abismos de Hamming.

Abismo de Hamming

- ✱ Para efetuar uma mudança de valor unitário, nós às vezes necessitamos mudar todos os bits de um número binário;
- ✱ Efeito chamado de Abismo de Hamming;
- ✱ Por exemplo:
 - ✱ para mudar do número 7 (0111) para o número 8 (1000), precisamos alterar todos os bits de uma única vez
 - ✱ para mudar do número 8 para o número 9 (1001), precisamos mudar apenas um único bit;
 - ✱ duas mudanças com o mesmo efeito final requerem alterações díspares nos cromossomos;
 - ✱ problema para o operador de mutação!

Abismo de Hamming

- ☀ Para evitá-lo, pode-se usar representação em código de Gray, também chamado de código espelhado;
- ☀ Código de Gray de n bits é formado “espelhando-se” o código de Gray de $n-1$ bits, colocando 0 na frente dos números acima do “espelho” e 1 na frente dos números abaixo do “espelho”.



Código de Gray

- ✴ Usando o código de Gray, o abismo de Hamming não é mais um problema;
- ✴ Ainda enfrentamos uma condição inexorável dos números binários: **existem bits mais significativos**;
- ✴ Exemplo: número binário 1000
 - ✴ se trocarmos o primeiro bit, faremos uma mudança de 8 para 0 (valor absoluto 8),
 - ✴ se trocarmos o último, faremos uma mudança de 8 para 9 (valor absoluto 1).

Bits mais significativos

- ☀ Idéia:

- ☀ aplicar a mutação sobre o número que é representado e não sobre sua representação binária. Passos:

- converter o número de volta para o seu valor,
- realizar uma mutação de magnitude pequena,
- voltar para a forma binária.

- ☀ Mutações no bit mais significativo isto é equivalente a “sacudir” o cromossomo;

- Vai levá-lo para uma área do espaço de soluções potencialmente não explorada;
- ☀ Não necessariamente este efeito é totalmente indesejado.

- ☀ Se variação for indesejada, pode-se evitar tal divergência de duas maneiras:

- ☀ fazendo a probabilidade de mutação aumentar com a diminuição da ordem do bit;
- ☀ fazer o mais simples, que é usar a codificação real diretamente.

Regras gerais para escolha de representação

1. A representação deve ser a mais simples possível;
2. Se houver soluções proibidas ao problema, então elas não devem ter uma representação;
3. Se o problema impuser condições de algum tipo, estas devem estar implícitas dentro da nossa representação;

Conclusão Óbvia:

Nem sempre podemos recorrer de forma natural à representação binária

Outras representações

- ✴ Existem várias alternativas à representação binária;
- ✴ Principais:
 - Números reais: (43.2 -33.1 ... 0.0 89.2);
 - Permutações de elementos (como no caso do GA baseado em ordem): (E4 E3 E7 E2 E6 E1 E5);
 - Listas de regras: (R1 R2 R3 ... R22 R23)
 - Qualquer estrutura de dados que pudermos imaginar!