



Algoritmos Genéticos

Luis Martí

LIRA/DEE/PUC-Rio




Baseado nas transparências dos professores:

Teresa B. Ludermir (UFPE)

Ricardo Linden (CEPEL)

Marco Aurélio Pacheco (PUC-Rio)

Conteúdo

- 
- Introdução
 - O Algoritmo Genético Binário
 - Noções de Otimização
 - O Algoritmo Genético com Parâmetros Contínuos
 - Aspectos Práticos e Avançados
 - Aplicações



Introdução

Algoritmos Genéticos



- São técnicas de busca e otimização.
- É a metáfora da teoria da evolução das espécies iniciada pelo Fisiologista e Naturalista inglês Charles Darwin.
- Desenvolvido por John Holland (1975) e seus alunos.
- Popularizado por David Goldberg (1989).

Teoria da Evolução

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



Charles
Darwin

“As espécies evoluem pelo principio da seleção natural e sobrevivência do mais apto.”

Teoria da Evolução




Gregor
Mendel




- 1865- Gregor Mendel apresenta experimentos do cruzamento genético de ervilhas.
 - ◆ Pai da genética.
- A Teoria da Evolução começou a partir da conceituação integrada da seleção natural com a Genética.


Otimização

- 
- É a busca da melhor solução para um dado problema.
 - ◆ Consiste em tentar várias soluções e usar a informação obtida para conseguir soluções cada vez melhores.
 - Exemplo de otimização:
 - ◆ Telespectador através de ajuste na antena da televisão otimiza a imagem buscando várias soluções até alcançar uma boa imagem.


Otimização

- 
- As técnicas de otimização, geralmente, apresentam:
 - **Espaço de busca:** onde estão todas as possíveis soluções do problema;
 - ◆ **Função objetivo:** utilizada para avaliar as soluções produzidas, associando a cada uma delas uma nota.

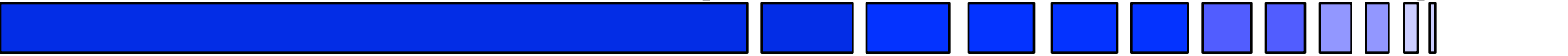
Características dos Algoritmos Genéticos

- 
- É um algoritmo estocástico (não é determinístico).
 - Trabalha com uma população de soluções simultaneamente.
 - Utiliza apenas informações de custo e recompensa. Não requer nenhuma outra informação auxiliar (como por exemplo o gradiente).


Características dos Algoritmos Genéticos (II)

- 
- São fáceis de serem implementados em computadores.
 - Adaptam-se bem a computadores paralelos.
 - São facilmente hibridizados com outras técnicas.
 - Funcionam com parâmetros contínuos ou discretos.

Algoritmos Genéticos (Conceitos Básicos)

- 
- AG manipula uma população de indivíduos.
 - Indivíduos são possíveis soluções do problema.
 - Os indivíduos são combinados (crossover) uns com os outros, produzindo filhos que podem sofrer ou não mutação.
 - As populações evoluem através de sucessivas gerações até encontrar a solução ótima.


Aplicações

- 
- Em problemas difíceis de otimização, quando não existe nenhuma outra técnica específica para resolver o problema.
 - Otimização de funções numéricas em geral
 - Otimização combinatória
 - ◆ Problema do caixeiro viajante
 - ◆ Problema de empacotamento
 - ◆ Alocação de recursos (*job shop scheduling*)
 - Aprendizado de Máquina
 - Projetos



O Algoritmo Genético Binário

Algoritmo Genético Tradicional

- 
1. Gerar a população inicial.
 2. Avaliar cada indivíduo da população.
 3. Enquanto critério de parada não for satisfeito faça
 - 3.1 Selecionar os indivíduos mais aptos.
 - 3.2 Criar novos indivíduos aplicando os operadores crossover e mutação.
 - 3.3 Armazenar os novos indivíduos em uma nova população.
 - 3.4 Avaliar cada cromossomo da nova população.

Problema 1

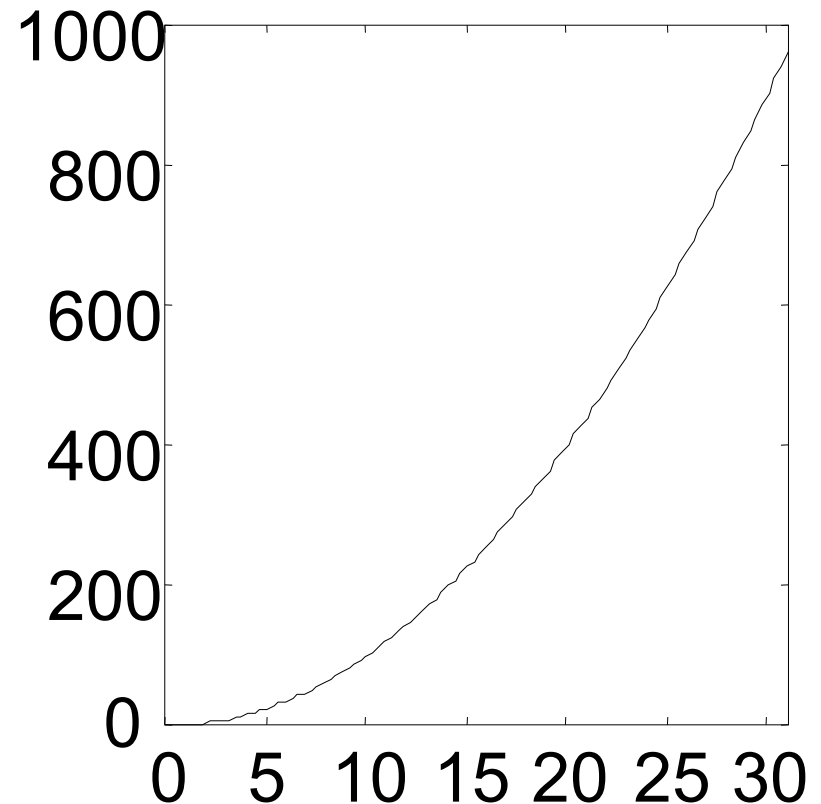
Problema: Use um AG
para encontrar o ponto
máximo da função:

$$f(x) = x^2$$

com x sujeito as seguintes
restrições:

$$0 \leq x \leq 31$$

x é inteiro



Indivíduo



■ Cromossomo

- ◆ Estrutura de dados que representa uma possível solução para o problema.
- ◆ Os parâmetros do problema de otimização são representados por cadeias de valores.
- ◆ Exemplos:
 - Vetores de reais, (2.345, 4.3454, 5.1, 3.4)
 - Cadeias de bits, (111011011)
 - Vetores de inteiros, (1,4,2,5,2,8)
 - ou outra estrutura de dados.

Indivíduo (II)



■ Aptidão

- ◆ Nota associada ao indivíduo que avalia quão boa é a solução por ele representada.

■ Aptidão pode ser:

- ◆ Igual a função objetivo (raramente usado na prática).
- ◆ Resultado do **escalonamento** da função objetivo.
- ◆ Baseado no **ranking** do indivíduo da população.

Cromossomo do Problema 1



- Cromossomos binários com 5 bits:
 - $0 = 00000$
 - $31 = 11111$
- Aptidão
 - ◆ Neste problema, a aptidão pode ser a própria função objetivo.
 - ◆ Exemplo:

$$\text{aptidão}(00011) = f(3) = 9$$

Seleção



■ Seleção

- ◆ Imitação da seleção natural.
- ◆ Os melhores indivíduos (maior aptidão) são selecionados para gerar filhos através de crossover e mutação.
- ◆ Dirige o AG para as melhores regiões do espaço de busca.

■ Tipos mais comuns de seleção

- ◆ Proporcional a aptidão.
- ◆ Torneio.

População Inicial do Problema 1

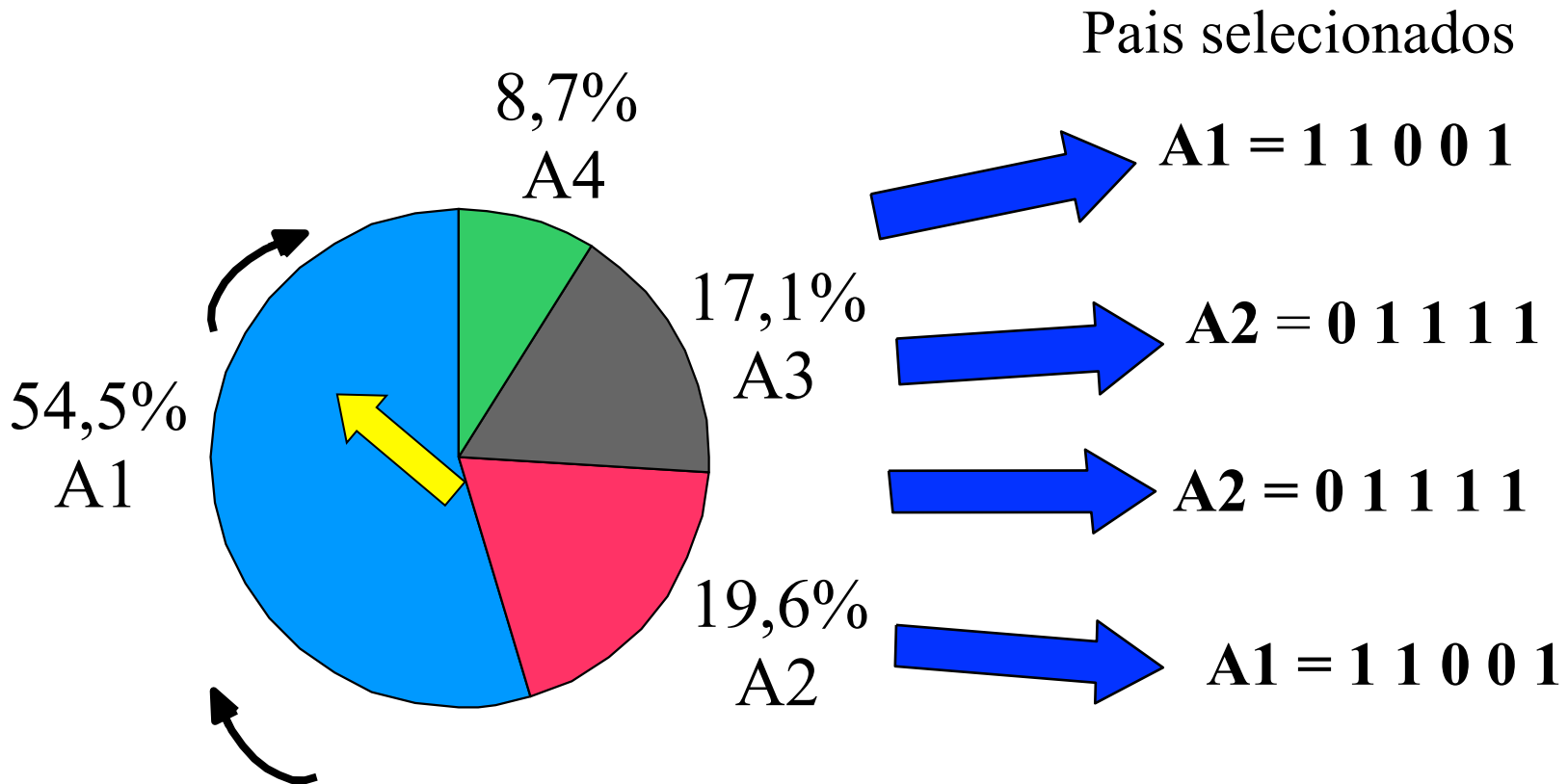
É aleatória (mas quando possível, o conhecimento da aplicação pode ser utilizado para definir população inicial)

	cromossomos	x	$f(x)$	Prob. de seleção
Pop. inicial	$A_1 = 1\ 1\ 0\ 0\ 1$	25	625	54,5%
	$A_2 = 0\ 1\ 1\ 1\ 1$	15	225	19,6%
	$A_3 = 0\ 1\ 1\ 1\ 0$	14	196	17,1%
	$A_4 = 0\ 1\ 0\ 1\ 0$	10	100	8,7%

Probabilidade de seleção
proporcional a aptidão

$$p_i = \frac{f(x_i)}{\sum_{k=1}^N f(x_k)}$$

Seleção proporcional a aptidão (Roleta)




Seleção por Torneio



- Escolhe-se n (tipicamente 2) indivíduos aleatoriamente da população e o melhor é selecionado.

Seleção por Torneio

- 
- A seleção por torneio (GOLDBERG, 1989) consiste em escolher aleatoriamente certo número de indivíduos da população (designado por dimensão do torneio) e fazer um torneio entre eles.
 - Cada torneio consiste em **comparar** os valores de **aptidão dos indivíduos envolvidos**, sendo o **vencedor** (e o selecionado) aquele com **melhor valor de aptidão**.
 - O número de torneios realizados é igual ao número de indivíduos a serem selecionados, ou seja, igual ao tamanho da população.

Seleção por Torneio



- Esta técnica:
 - ◆ não conduz à convergência prematura (desde que a dimensão dos torneios seja pequena),
 - ◆ combate a estagnação da população,
 - ◆ é simples de implementar
 - ◆ não requer grande esforço computacional.
- Este é talvez o mecanismo de seleção mais utilizado na resolução de problemas de otimização.

Crossover e Mutação

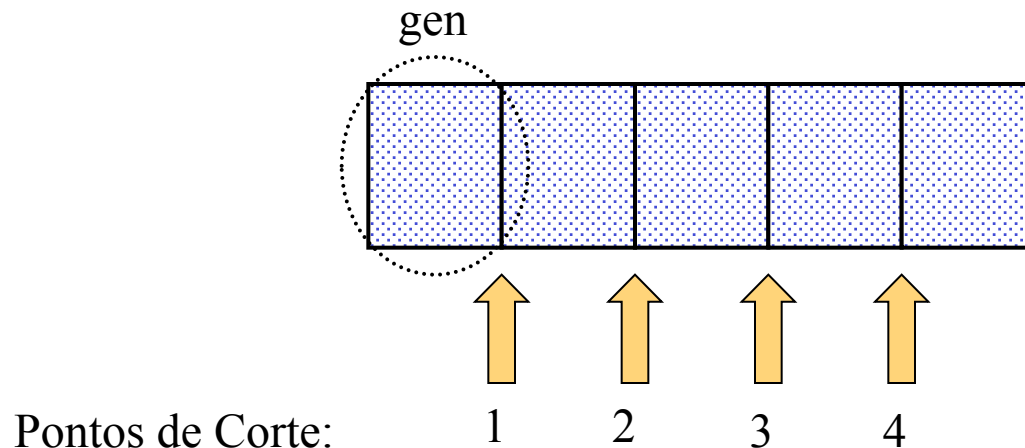


- Combinam pais selecionados para produção de filhos.
- Principais mecanismos de busca do AG.
- Permite explorar áreas desconhecidas do espaço de busca.

Operador de Crossover



- Vamos começar com o operador de crossover mais simples, chamado de operador de crossover de um ponto.
- Depois de selecionados dois pais pelo módulo de seleção de pais, um ponto de corte é selecionado.
- Um ponto de corte constitui uma posição entre dois genes de um cromossomo.
- Cada indivíduo de n genes contém $n-1$ pontos de corte.



Operador de Crossover




- Depois de sorteado o ponto de corte, nós separamos os pais em duas partes: uma à esquerda do ponto de corte e outra à direita.
- É importante notar que não necessariamente estas duas partes têm o mesmo tamanho.
- O primeiro filho é composto através da concatenação da parte esquerda do primeiro pai com a parte direita do segundo pai.
- O segundo filho é composto através da concatenação das partes que sobraram (a metade esquerda do segundo pai com a metade à direita do primeiro pai).

Operador de Mutação

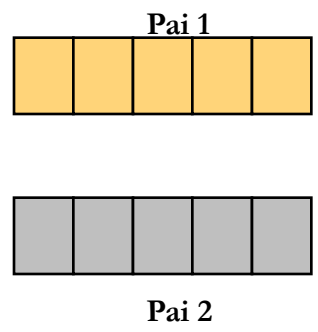
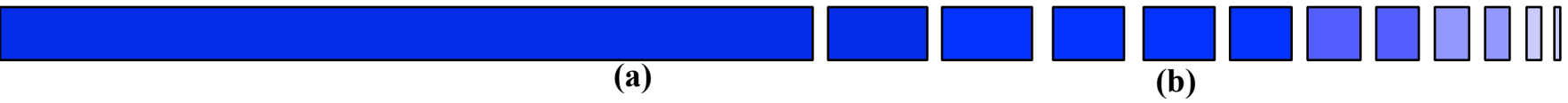


- Depois de compostos os filhos, entra em ação o operador de mutação.
- Este opera da seguinte forma:
 - ◆ Ele tem associada uma probabilidade extremamente baixa (da ordem de 0,5%);
 - ◆ Nós sorteamos um número entre 0 e 1.
 - ◆ Se ele for menor que a probabilidade pré-determinada então o operador atua sobre o gene em questão, alterando-lhe o valor aleatoriamente.
 - ◆ Repete-se então o processo para todos os gens componentes dos dois filhos.

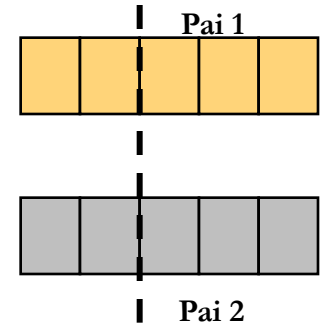
Comentários

- 
- Valor da probabilidade deve ser baixo.
 - ◆ Se ele for muito alto, o algoritmo genético se parecerá muito com uma técnica chamada “random walk”
 - Alguns textos preferem que o operador de mutação não aja de forma aleatória, mas sim, alterando o valor do gene para outro valor válido do nosso alfabeto genético.
 - ◆ Corresponde em multiplicar a probabilidade do operador de mutação por $n/(n-1)$, onde n é a cardinalidade do alfabeto genético.

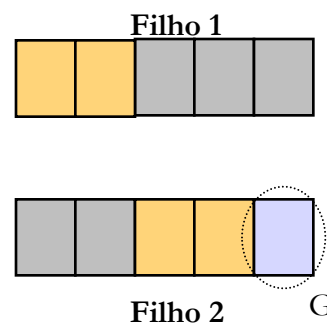
Juntando os operadores



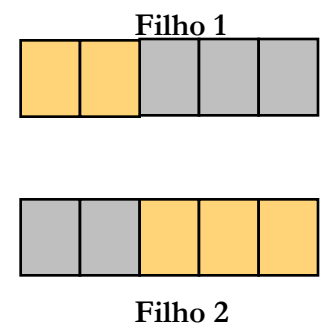
Selecioneamos um ponto de corte



Depois do operador de crossover



Depois do operador de mutação



(d)

(c)

Gen alterado pela mutação

Módulo de População



- O módulo de população é responsável pelo controle da nossa população.
- Por simplicidade, população não pode crescer
 - ◆ permite que armazenemos a população em um vetor de tamanho constante.
- Pais têm que ser substituídos conforme os filhos vão nascendo
 - ◆ Pode parecer estranho, visto que estamos acostumados a ver a população humana sempre crescendo.
 - ◆ Quando nasce um bebê, não é obrigatório que alguém de alguma geração anterior caia fulminado!
 - ◆ Entretanto, simula bem ambientes de recursos limitados

Outros Crossover's

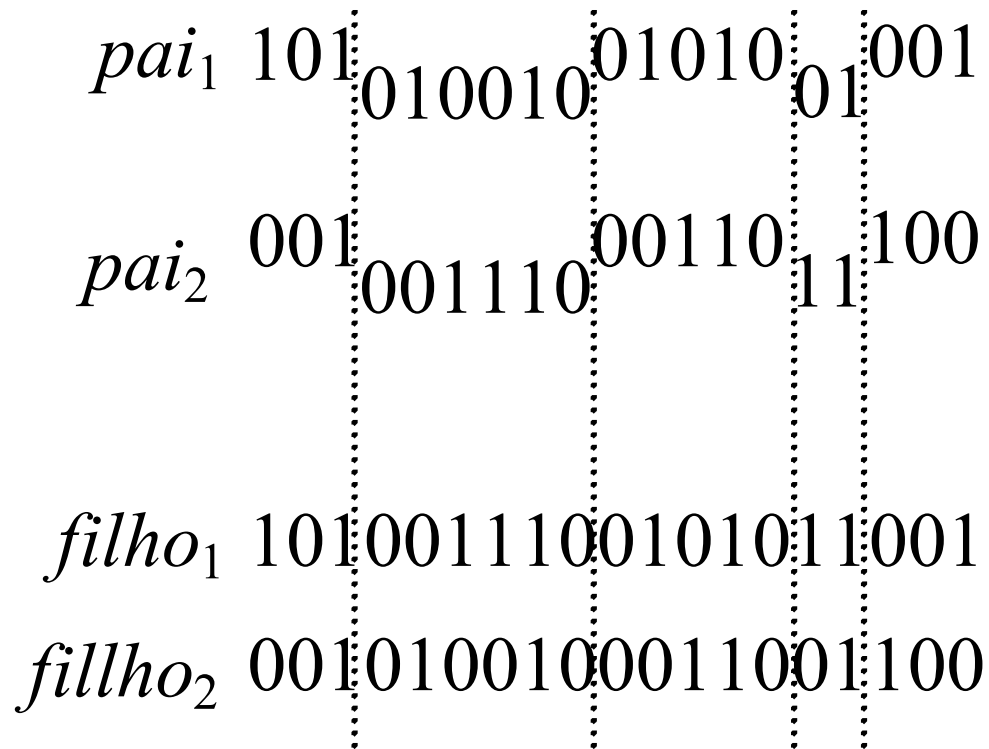


- Crossover de 2-pontos

<i>pai</i> ₁	010	011000	101011
<i>pai</i> ₂	001	001110	001101
<i>filho</i> ₁	010	001110	101011
<i>filho</i> ₂	001	011000	001101

Considerado melhor que o crossover de 1 ponto.

Crossover de n-Pontos



Crossover de 4-pontos

Crossover Uniforme



Máscara de bits aleatória
1 1 0 1 0 1 1 0 1 0

<i>pai</i> ₁	1	1	1	0	1	1	0	1	1	0
	↓	↓		↓		↓	↓		↓	
<i>filho</i> ₁	1	1	1	0	0	1	0	1	1	0
			↑		↑			↑		↑
<i>pai</i> ₂	0	1	1	0	0	0	1	1	0	0

O *filho*₁ tem 50% de chance de levar um bit do *pai*₁ e 50% de chance de levar um bit de *pai*₂

O *filho*₂ leva o que sobra de *pai*₁ e *pai*₂

Problema 2

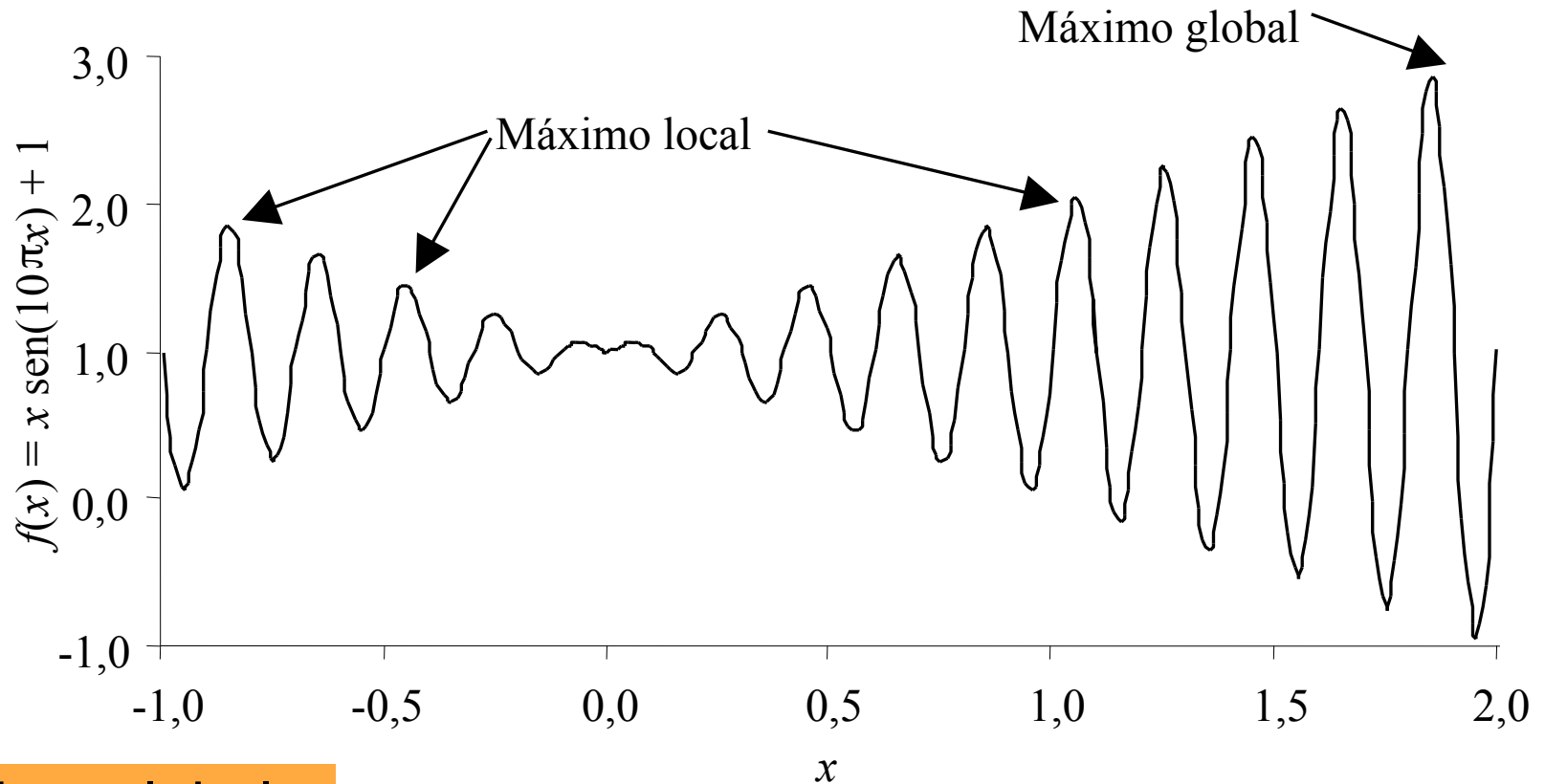
Achar o máximo da função utilizando
Algoritmos Genéticos,

$$f(x) = x \operatorname{seno}(10\pi x) + 1,0$$

Restrita ao intervalo:

$$-1,0 \leq x \leq 2,0$$

Problema 2 (II)




Máximo global:

$$x = 1,85055$$

$$f(x) = 2,85027$$

Problema 2 (III)

- 
- Função multimodal com vários pontos de máximo.
 - É um problema de otimização global (encontrar o máximo global)
 - Não pode ser resolvido pela grande maioria dos métodos de otimização convencional.
 - Há muitos métodos de otimização local, mas para otimização global são poucos.

O Cromossomo Problema 2



- Representar o único parâmetro deste problema (a variável x) na forma de um cromossomo:
 - ◆ Quantos bits deverá ter o cromossomo?
 - ◆ Quanto Mais bits melhor precisão numérica.
 - ◆ Longos cromossomos são difíceis de manipular.
 - ◆ Para cada decimal é necessário 3,3 bits
 - ◆ Cromossomo com 22 bits

1000101110110101000111

O Cromossomo Problema 2 (II)



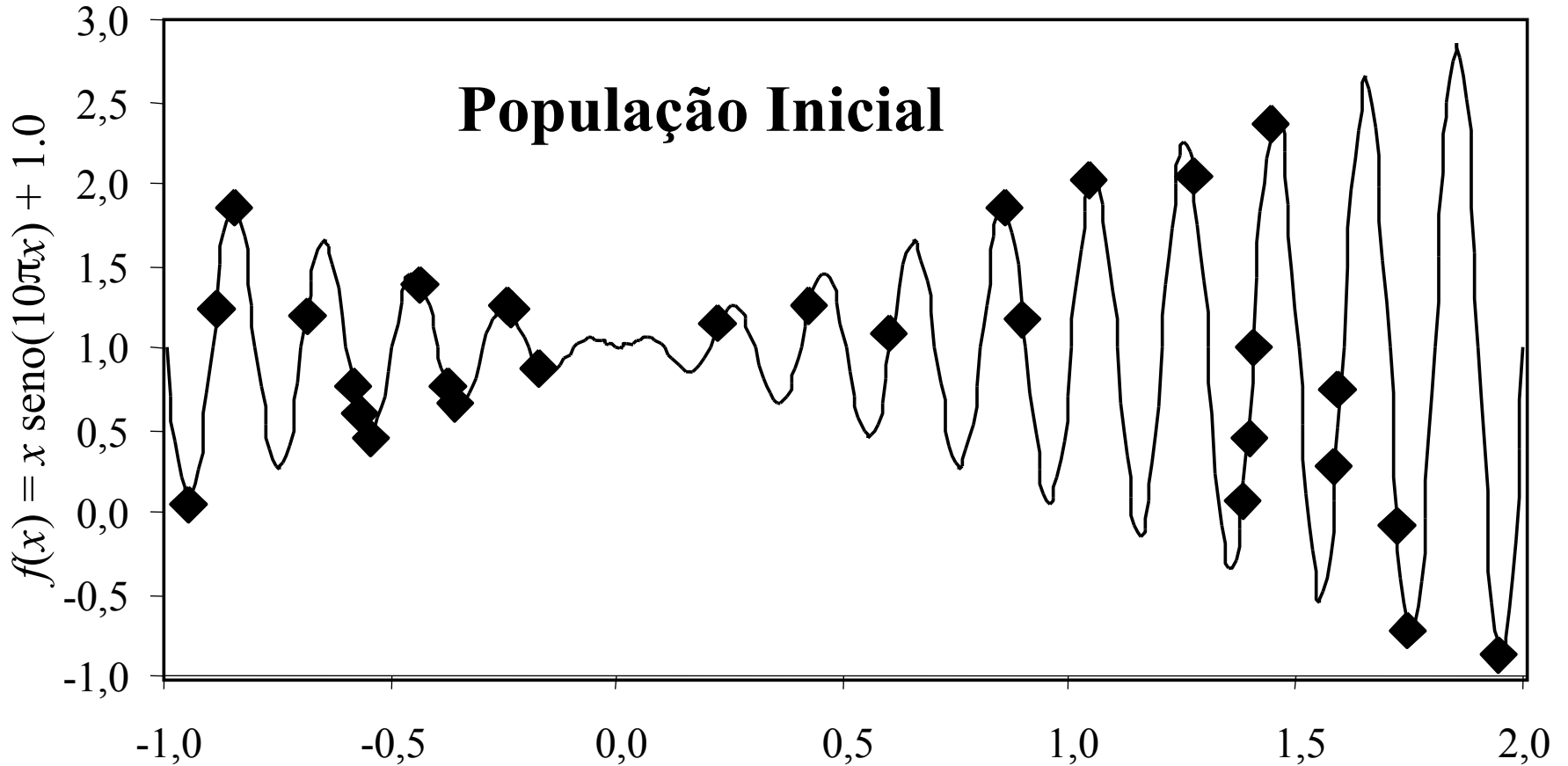
■ Decodificação

- ◆ cromossomo = 1000101110110101000111
- ◆ $b_{10} = (1000101110110101000111)_2 = 2288967$
- ◆ Valor de x precisa estar no intervalo $[-1,0; 2,0]$

$$x = \min + (\max - \min) \frac{b_{10}}{2^l - 1}$$

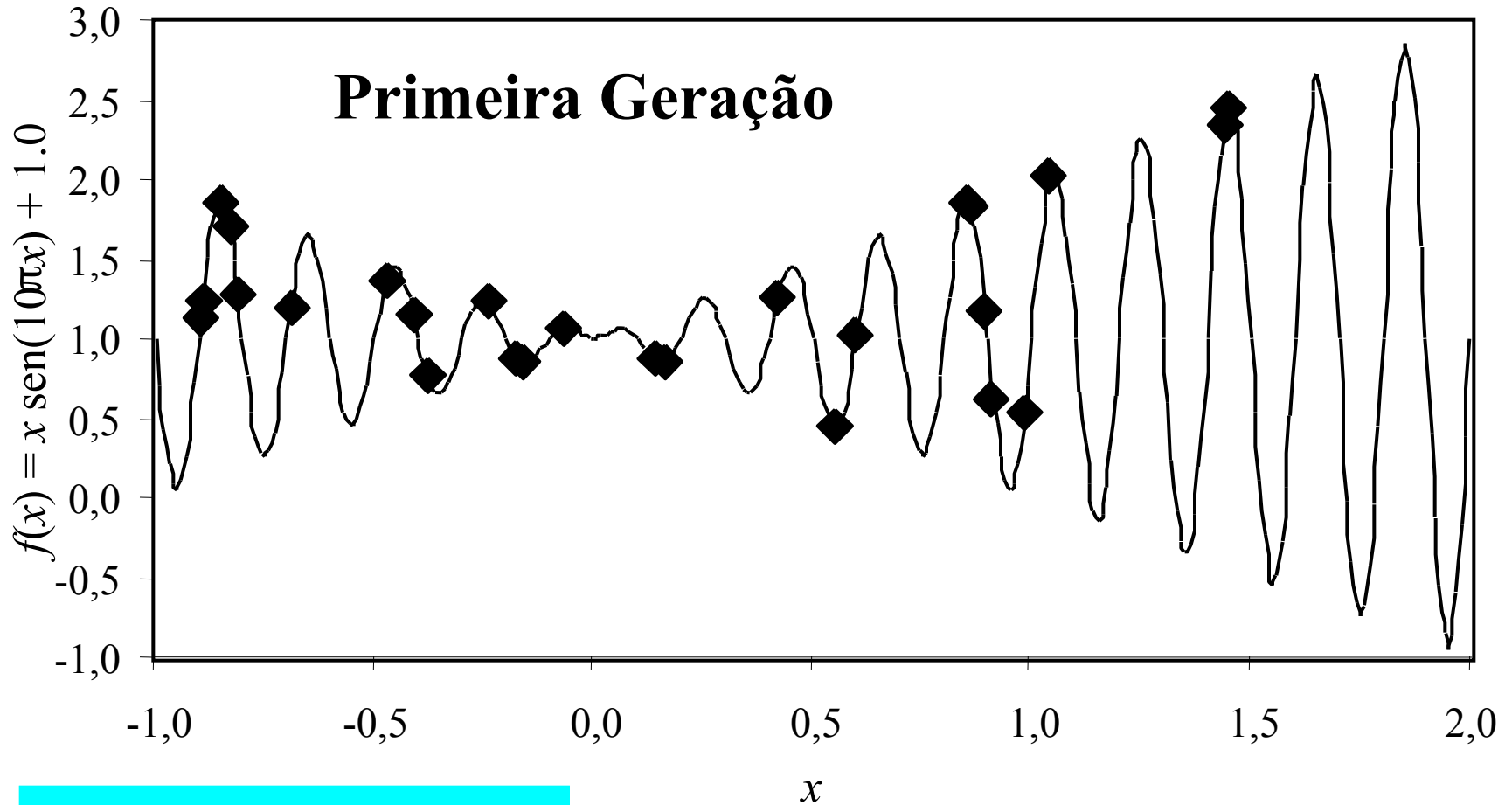
$$x = -1 + (2 + 1) \frac{2.288.967}{2^{22} - 1} = 0,637197$$

As Gerações do Problema 2



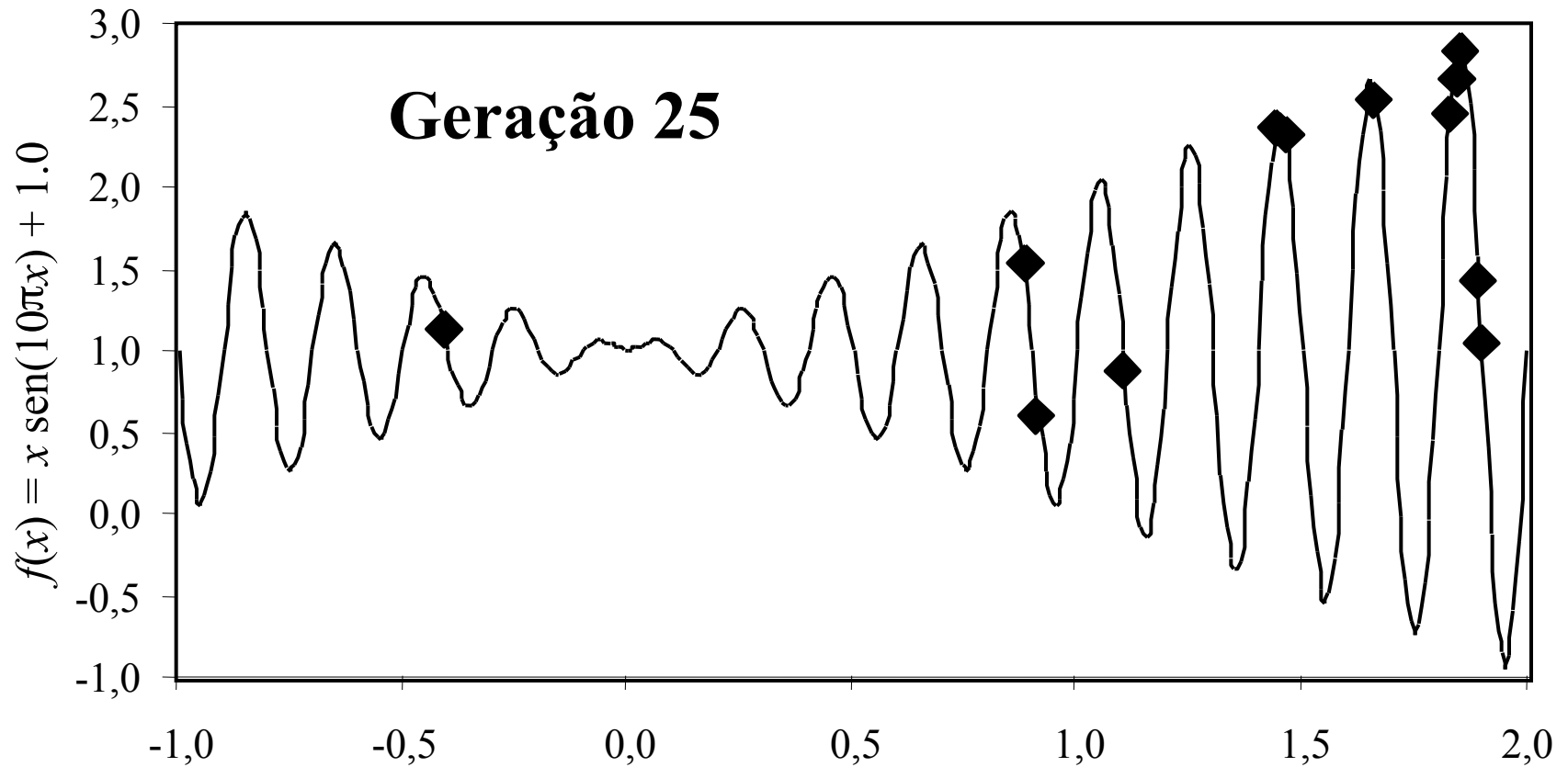
População gerada aleatoriamente

As Gerações do Problema 2 (II)



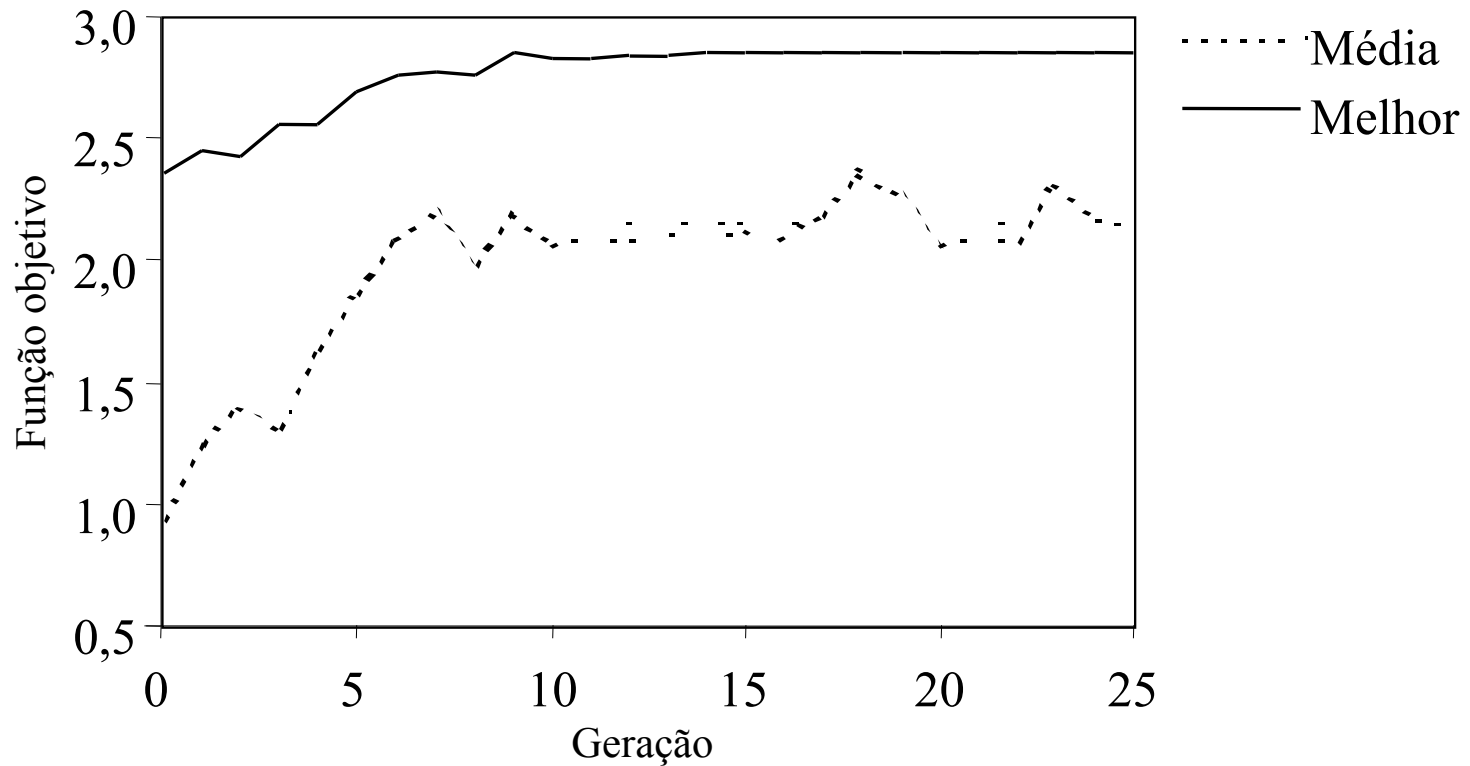
Pouca melhoria

As Gerações do Problema 2 (III)




A maioria dos indivíduos encontraram o máximo global

As Gerações do Problema 2 (IV)

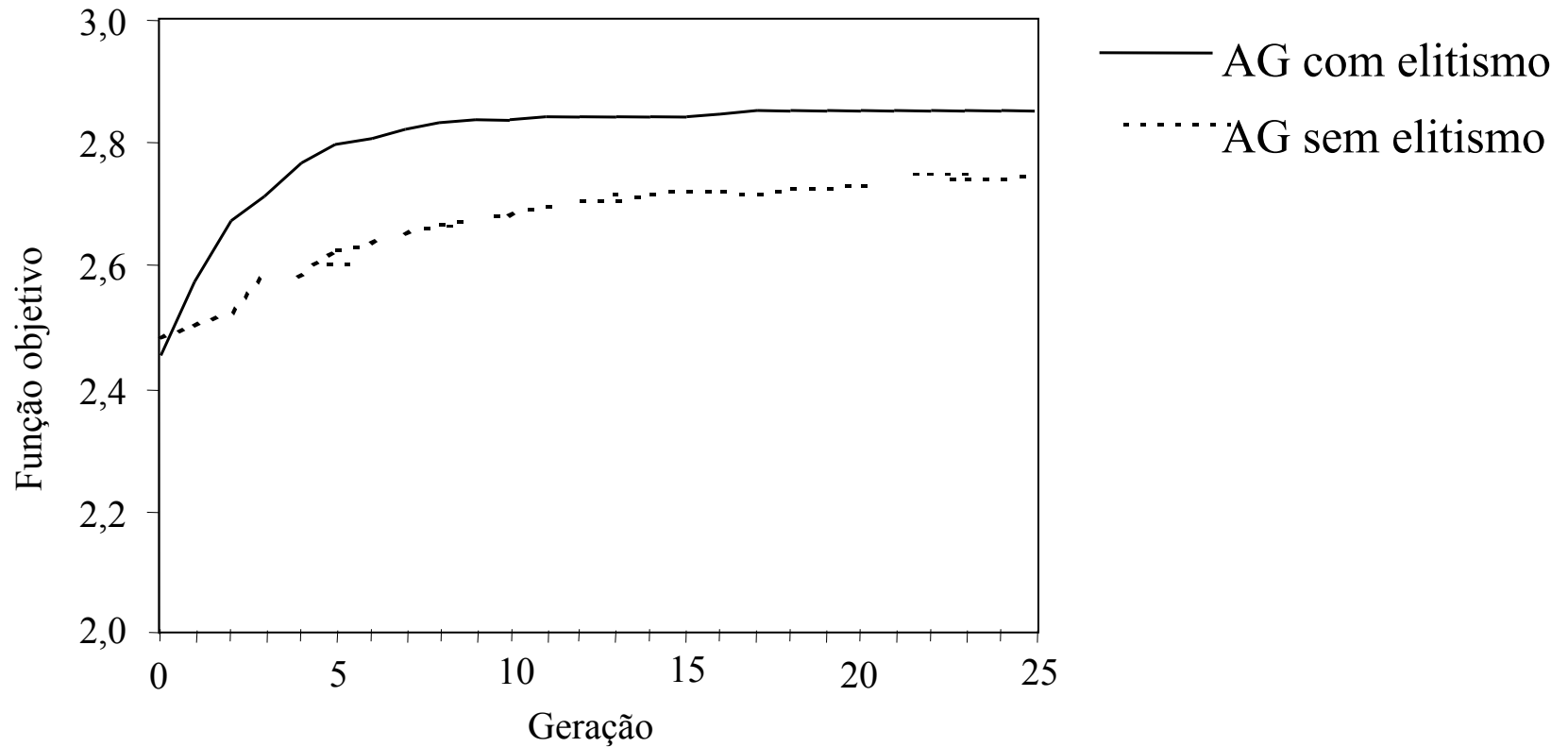


Na geração 15 o AG já encontrou o ponto máximo

Elitismo


- 
- O crossover ou mutação podem destruir a melhor indivíduo.
 - Por que perder a melhor solução encontrada?
 - Elitismo transfere a cópia do melhor indivíduo para a geração seguinte.

Elitismo no Problema 2



AG com elitismo é melhor ?

Critérios de Parada

- 
- Número de gerações.
 - Encontrou a solução (quando esta é conhecida).
 - Perda de diversidade.
 - Convergência
 - ◆ nas últimas k gerações não houve melhora da na aptidão
 - Média
 - Máxima

Terminologia

■ Indivíduo

- ◆ Simples membro da população.

■ Cromossomo e Genoma e :

- ◆ Coleção de genes
- ◆ Estrutura de dados que codifica a solução de uma problema.

■ Genótipo

- Na biologia, representa a composição genética contida no Genoma. Nos AGs, representa a informação contida no cromossomo ou genoma.

Terminologia




■ Fenótipo:

- ◆ Objeto ou estrutura construída a partir das informações do genótipo.
- ◆ É o cromossomo decodificado.
 - Exemplo: Se o cromossomo codifica as dimensões de um edifício, então o fenótipo é o edifício construído.

■ Gene:

- ◆ Codifica um simples parâmetro do problema


Exercício

- 
- Encontrar de x para o qual a função $f(x) = x^2 - 3x + 4$ assume o valor mínimo.
 - ◆ Assumir que $x \in [-10, +10]$
 - ◆ Codificar X como vetor binário
 - ◆ Criar uma população inicial com 4 indivíduos
 - ◆ Aplicar Mutação com taxa de 1%
 - ◆ Aplicar Crossover com taxa de 60%
 - ◆ Usar seleção por torneio.
 - ◆ Usar 5 gerações.



Aspectos Práticos

Principais Tópicos

- 
- ◆ População Inicial
 - ◆ Funções Objetivo de Alto Custo
 - ◆ Critérios de Parada
 - ◆ Convergência Prematura
 - ◆ Diversidade
 - ◆ Tipos de Substituição
 - ◆ Problemas na Aptidão

População Inicial (1/3)



- Gerada Aleatoriamente.
- Gerada uniformemente em uma grade.
- Gerada com tendenciosidade para regiões promissoras do espaço de busca

População Inicial (2/3)



- Para garantir que toda posição da cadeia tem 0 e 1 na população:

1) Gera a primeira metade da população aleatoriamente.

2) Inverte todos os bits da primeira metade: tem-se a segunda metade.

1a. metade	2 ^a metade
1011010	0100101
0111011	1000100
0001101	1110010
1100110	0011001

População Inicial (3/3)

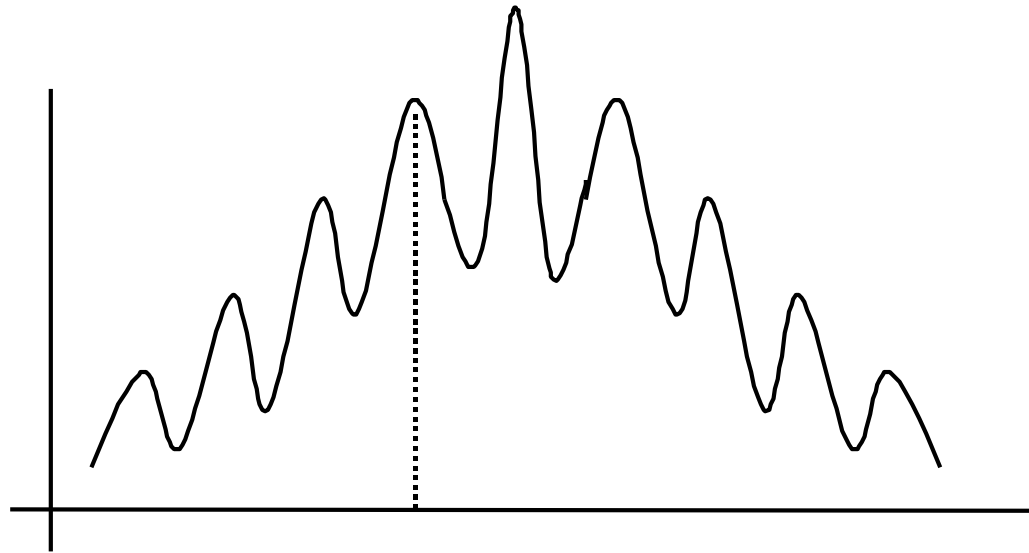


- *Seeding*: insere a solução obtida por outro método de otimização na população inicial (garante que AG não fará pior do que o outro método)
- Iniciar com uma larga população inicial e depois reduzir o tamanho.

Convergência Prematura (1/2)



- O AG converge para um mínimo/máximo local.



Convergência Prematura (2/2)

■ Causas:

- ◆ Excessivo números de filhos de um mesmo indivíduo (o superindivíduo)
- ◆ Perda de diversidade.
- ◆ *Genetic Drift*
 - Desaparecimento de um determinado gene na população.
 - Ocorre principalmente em pequenas populações.
- ◆ Alta pressão de seleção
 - Poder que faz com que os indivíduos com maior aptidão tenham mais descendentes.

Diversidade (1/2)

■ Combatendo a perda de diversidade

- ◆ Aumentar a taxa de mutação.
- ◆ Evitar cromossomos duplicatas na população.
- ◆ Diminuir a pressão da seleção.

Diversidade (2/2)

■ Combatendo a perda de diversidade

- ◆ Controlar o número de filhos do superdividuo (indivíduo com alta aptidão, mas não com aptidão ótima) usando:
 - Ranking.
 - Escalamiento.
 - Seleção por torneio.

Tipos de Substituição



- Substituição Geracional
- Substituição Geracional com Elitismo
- Substituição de Estado Uniforme

Substituição Geracional



Seja N o tamanho da população:


- ◆ Os N pais são substituídos pelos N filhos em cada geração.
- ◆ Os N pais são substituídos por N indivíduos do conjunto união de pais e filhos.
- Comentário: o segundo caso aumenta a pressão de seleção.

Substituição Geracional com Elitismo




- Os $k < N$ melhores pais nunca são substituídos.
- Tipicamente $k = 1$
- Aumentando k aumenta a pressão de seleção (risco de convergência prematura).

Substituição de Estado Uniforme (1/2)

- 
- Em cada “geração” apenas 2 (ou 1) filhos são gerados e substituem:
 - ◆ Os 2 piores indivíduos da população.
 - ◆ Os pais.
 - ◆ Os 2 indivíduos mais velhos (i.e. que estão a mais tempo da população), pois já transmitiram os seus genes.
 - Taxa de crossover é geralmente alta (~ 1)

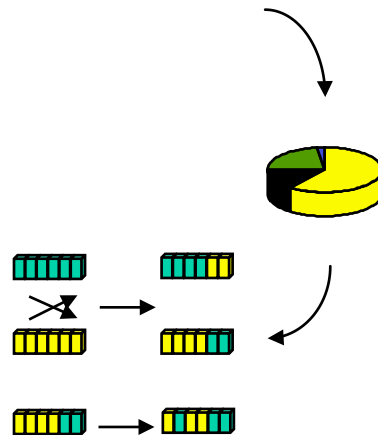
Substituição de Estado Uniforme (2/2)

- 
- Alternativamente, $k < N$ filhos são gerados e substituem os k piores indivíduos.
 - Evitar inserir um filho na população quando já existe uma duplicata dele na população.

Exemplo de Steady State

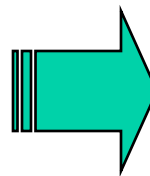
C19	120
C18	110
C17	100
C16	99
C15	95
C14	81
C13	76
C12	67
C11	58
C10	44
C9	42
C8	36
C7	22
C6	20
C5	19
C4	17
C3	10
C2	8
C1	5

avaliações de P(t)



38
6
121
88
58
17

crie n novos



120
110
100
99
95
81
76
67
58
44
42
36
22
38
6
121
88
58
17

substitua os n piores

ordena



121
120
110
100
99
95
88
81
76
67
58
58
44
42
38
36
22
17
6

avaliações de P(t+1)

Steady State sem Duplicados

- Substituição parcial de indivíduos com **exclusão de duplicados**
- Evita os **duplicados que são mais frequentes** com steady state (populações mais estáticas)
- Maior **eficiência do paralelismo** de busca, garantindo `pop_size` indivíduos diferentes
- Descendentes duplicados são desprezados
- Maior **overhead para teste** de igualdade

Problemas na Aptidão (1/3)



- Aptidão negativa não funciona com a roleta
- Aptidão excessivamente alta
 - Poucos indivíduos ocupando larga fatia da roleta
 - Muitos indivíduos ocupando pequena fatia da roleta
 - Causa convergência prematura
 - ◆ Solução: controlar o número de filhos do superindivíduo.

.

Problemas na Aptidão (2/3)



- Resolução insuficiente para diferenciar os melhores dos piores indivíduos.
 - ◆ A seleção torna-se aleatória (Passeio ao Acaso).
 - ◆ Convergência lenta

Problemas na Aptidão (3/3)

■ Exemplo:

Cromossomo	Função objetivo	Probabilidade de seleção
A	2000,999588	20,004%
B	2000,826877	20,002%
C	2000,655533	20,001%
D	2000,400148	19,998%
E	2000,102002	19,995%

■ Soluções

- ◆ Expandir o intervalo da aptidão (usando ranking)
- ◆ Seleção por torneio

Técnicas de Aptidão

■ Aptidão é a Avaliação

$$A_i = f_i$$

Exemplo: $A_i = 999,979$

■ Windowing

- ◆ subtrair uma constante dos valores de f_i

■ Normalização Linear

- ◆ atribuir valores a A_i baseados no rank do cromossoma

Windowing

- Obtenha a avaliação mínima na população.
- Atribua a cada cromossoma i uma aptidão igual a:

$$A_i = (f_i - A_{\text{mín}})$$

- Opcionalmente, atribua uma **aptidão mínima de “sobrevivência”**, maior que a aptidão mínima calculada, como garantia de reprodução para os cromossomas menos aptos.

- Exemplo:

$$A_i = (999,979 - 999,066) = 0,913$$

Normalização Linear

- Coloque os `pop_size` cromossomas em ordem decrescente de avaliação ($i=1$ é o menos apto).
- Crie aptidões, partindo de um valor mín e crescendo linearmente até o valor máx.
- Os valores de máx e mín (ou a constante de incremento) são parâmetros da técnica.

$$A_i = \text{mín} + \frac{(\text{máx} - \text{mín})}{\text{pop_size} - 1} \times (i - 1)$$

- Quanto maior a constante de incremento, maior a pressão seletiva sobre os melhores.


Exemplo Comparativo




Rank dos cromossomas	6	5	4	3	2	1
Avaliação original	200	9	8	7	4	1
Aptidão é avaliação	200	9	8	7	4	1
Normalização Linear, taxa=10	60	50	40	30	20	10
Normalização Linear, taxa=20	101	81	61	41	21	1
Windowing	199	8	7	6	3	0

- **SUPER INDIVÍDUO:** cromossoma 6
 - poucas chance de recombinação com outros indivíduos; elimina competidores em poucas gerações; rápida convergência.
- **COMPETIÇÃO PRÓXIMA:** entre cromossomas 3, 4 e 5
 - é preciso aumentar a pressão seletiva sobre os melhores


Critérios de Parada

- 
- Atingiu um dado número de gerações ou avaliações.
 - Encontrou a solução (quando esta é conhecida).
 - Perda de diversidade.
 - Convergência: não ocorre melhora significativa na solução durante um dado número de gerações.


Funções Objetivo de Alto Custo (1/3)

- 
- Em muitos problemas do mundo real o custo computacional do AG está concentrado na avaliação do individuo.
 - Exemplo:
 - ◆ Simulação completa de um processo.
 - ◆ Um treinamento de uma rede neural.

Funções Objetivo de Alto Custo (2/3)

- 
- Dicas para reduzir o número de reavaliações do indivíduo:
 - ◆ Evitar cromossomos iguais na população inicial.
 - ◆ Verificar se o filho já existe nas populações passadas e na atual.
 - ◆ Verificar se filho = pai (e.g. checar se crossover e mutação foi aplicado).
 - ◆ Manter a população com cromossomos distintos.

Funções Objetivo de Alto Custo (3/3)

- 
- Simplificar a função objetivo (pelo menos nas gerações iniciais)
 - Usar um método de subida de encosta quando o AG já encontrou as regiões promissoras do espaço de busca (nas gerações finais).

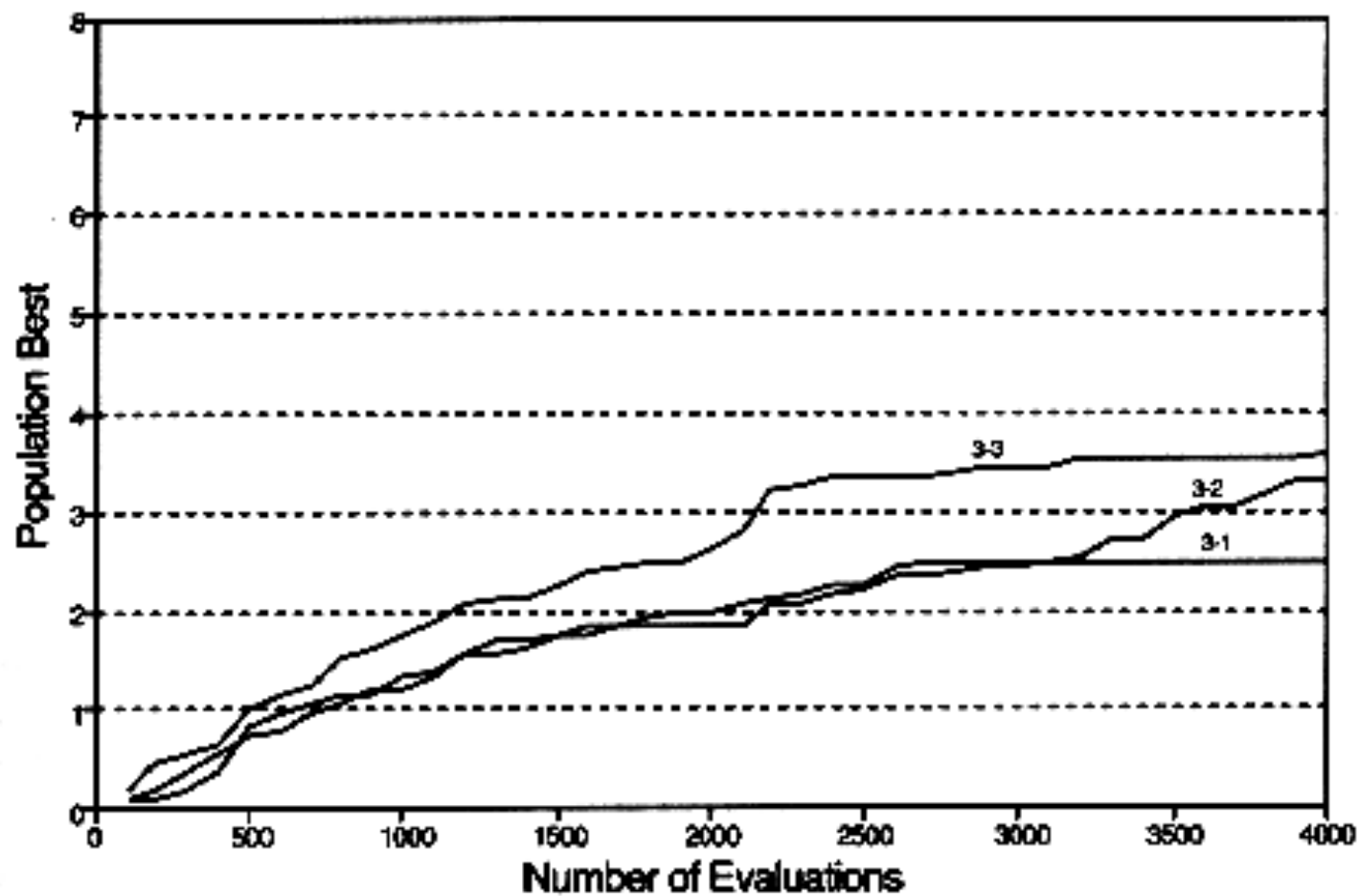




Figure 3.1: Performance curves for GA 3-1, GA 3-2, and GA 3-3 on binary f6


Parâmetros

- 
- O sucesso de um AG depende, em grande parte, da escolha dos seus parâmetros de configuração.
 - Entre os parâmetros a ser definidos temos as taxas de aplicação dos operadores de mutação e cruzamento, tamanho da população, número de gerações.
 - Muitos pesquisadores procuram descobrir qual seria o número de parâmetros mais adequado para resolver um problema específico.
 - O ajuste dos parâmetros em quase todas as pesquisas é feita numa etapa chamada de *tunning* dos parâmetros

Parâmetros

- 
- É difícil que um conjunto de parâmetros possa ser adequado para resolver um problema em todos os estágios da evolução sendo os AGs um processo dinâmico.

Parâmetros

- 
- As técnicas de adaptação dos parâmetros podem ser classificadas assim:
 - ◆ Determinística: Quando a mudança nos valores dos parâmetros ocorre seguindo alguma regra determinística.
 - ◆ Adaptativa: Ocorre quando existe um *feedback* por parte do AG que é usada para determinar o valor do parâmetro na próxima geração.