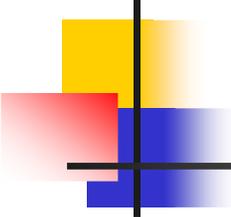


Computação Evolutiva

Aula 8 – Algoritmos Genéticos
(Parte IV)

Prof. Tiago A. E. Ferreira



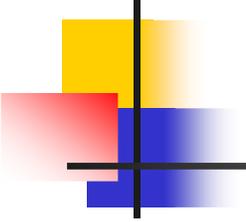
Roteiro

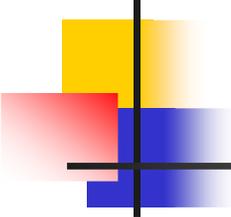
- Possibilidade de Recombinação de Múltiplos Pais
- Modelos de População
- Processos de seleção
- Exemplo de um AG

Recombinação de Múltiplos Pais

- Relembrar que não estamos totalmente vinculados às particularidades naturais
- Notando que a mutação usa 1 cromossomo, e o cruzamento “tradicional” usa 2, a extensão para cruzamentos com no. de cromossomos > 2 pode se examinado
- Examinado desde 1960's, ainda é de uso raro. Porém estudos indicam usabilidade

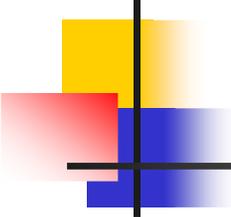
Recombinação de Múltiplos Pais

- 
- Três Tipos Principais:
 - Baseado na frequência dos genes, e.g. generalização do cruzamento uniforme por voto
 - Baseado na segmentação e recombinação dos pais
 - Baseado em operações numéricas em genes de valores reais, e.g., cruzamento de centro de massa, operadores generalizados de cruzamento aritmético.



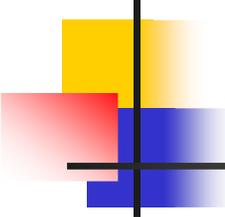
Modelos de População

- Os AGS usam modelos genéricos
 - Cada indivíduo sobrevive por exatamente uma geração
 - O conjunto inteiro dos pais é substituído por sua prole
- Sobre outra vertente, existem os modelos de estado-ligado
 - Uma prole é criada por geração
 - Um membro da população é substituído



Modelos de População

- Geração de *Gaps*
 - Seja a quantidade de pais selecionados - μ
 - Seja a quantidade da prole gerada - λ
 - Neste caso, λ ($< \mu$) indivíduos velhos são recolocados por λ indivíduos novos
 - A razão λ/μ define os *Gaps* Procriacionais – percentagem da população que é substituída
 - Modelo: GENITOR GENETIC ALGORITHM (GGA)
 - Usualmente $\lambda = 1$, implicando em um *gap* de $1/\mu$



Competição Baseada em Fitness

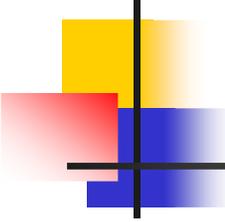
- A seleção pode ocorrer em dois lugares
 - Seleção na geração atual
 - Seleção a partir dos pais + prole para a geração da próxima geração (seleção por sobrevivência)
- Os operadores de seleção trabalham sobre a totalidade do indivíduo
 - São independentes da representação
- Distinção entre seleção
 - Operadores: define as probabilidades de seleção
 - Algoritmos: Define como as probabilidades são implementadas

Exemplo de Implementação: AGS

- Número esperado de cópias de um indivíduo i

$$E(n_i) = \mu \cdot f(i) / \langle f \rangle$$

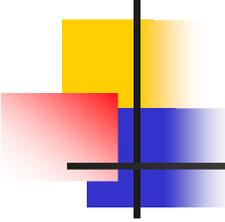
- Onde μ = tamanho da Pop., $f(i)$ = é o fitness de i , e $\langle f \rangle$ é o fitness médio da pop.
- Algoritmo da Roleta
 - Dada a distribuição de probabilidade, gire a roleta n vezes para selecionar n indivíduos
 - Não há nenhuma garantia sobre o valor de n_i
- Stochastic Universal Sampling (SUS) Algorithm
 - Baker's SUS Algorithm
 - N ponteiros da roleta igualmente espaçados e uma vez girado
 - Garante: $\text{Piso}(E(n_i)) \leq n_i \leq \text{Teto}(E(n_i))$



Seleção Proporcional ao Fitness

■ Problemas Incluem

- Um membro com alto *fitness* pode rapidamente dominar os demais membros da população com baixo *fitness* – Convergência Prematura
- Se *fitness* são semelhantes há a perda de pressão da seleção Natural
- Altamente susceptível a transposição de funções



Seleção Proporcional ao Fitness

- Escalonamento pode resolver os dois últimos problemas

- Janelas:

$$f(i) = f(i) - \beta^t$$

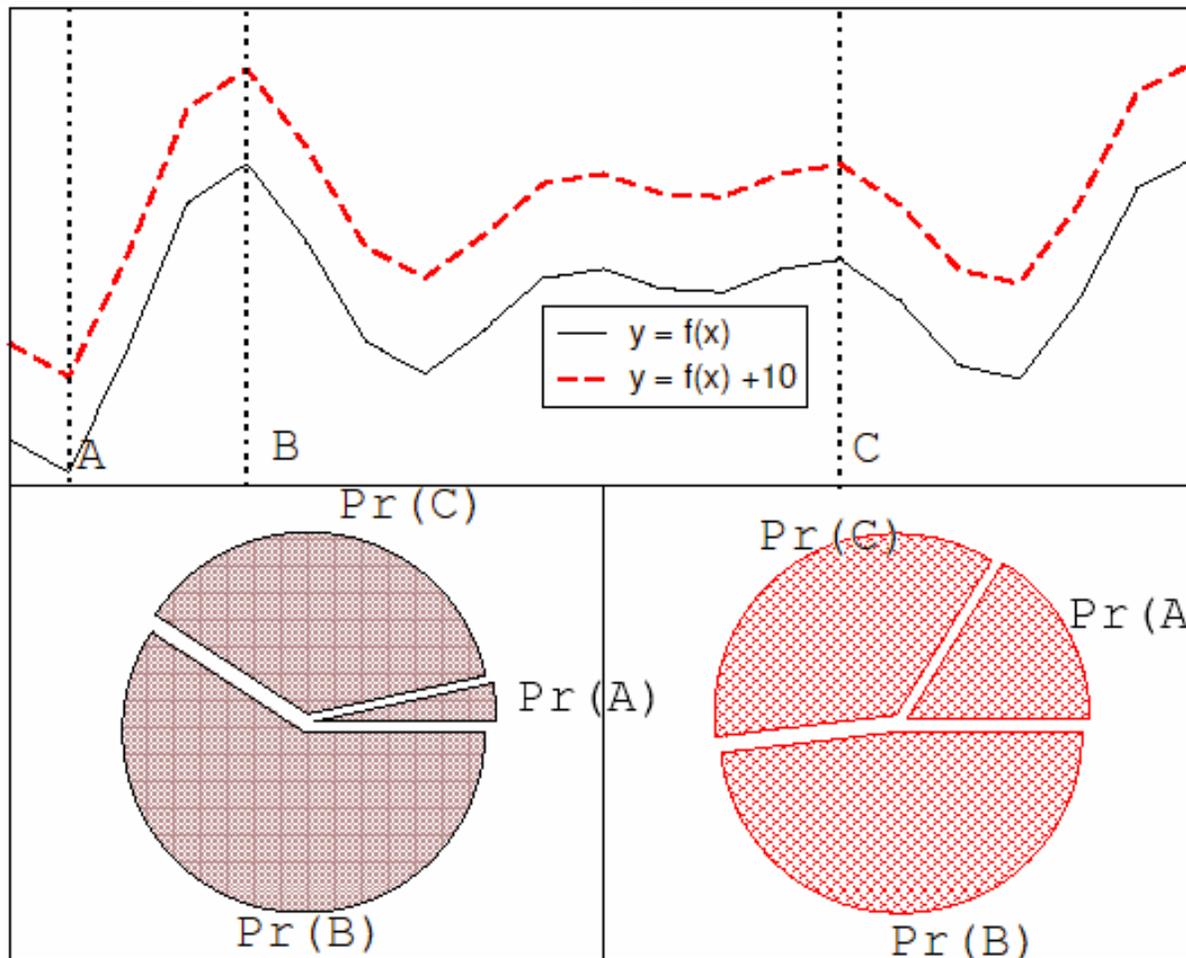
- Onde β é o pior fitness nas ultimas n gerações

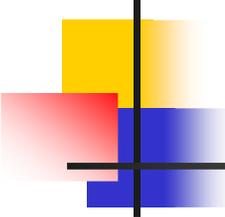
- Escalonamento Sigma:

$$f(i) = \max\left(f(i) - (\langle f \rangle - c \cdot \sigma_t), 0.0\right)$$

- Onde c é uma constante, usualmente 2.0

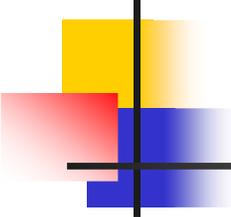
Transposição de Funções





Seleção Baseada em Rank

- Tenta remover os problemas gerados pela seleção baseada em fitness
 - Seleção Baseada em probabilidades relativas
- Rank da população é de acordo com o fitness e a seleção é baseada no rank, onde o rank μ é pior que o rank 1
- Este impõe um *overhead* em ordenamento, porém é usualmente desprezível quando comparável ao tempo de avaliação de fitness.

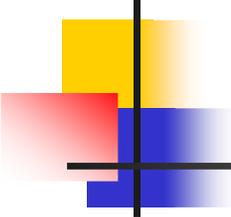


Rankeamento Linear

$$P_{lin-rank}(i) = \frac{(2-s)}{\mu} + \frac{2i(s-1)}{\mu(\mu-1)}$$

- Parametrizado pelo fator s : $1.0 \leq s \leq 2.0$
 - Mede o advento do melhor indivíduo
- Exemplo com 3 membros

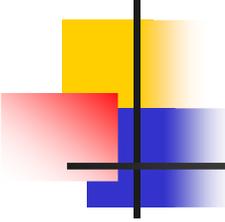
	Fitness	Rank	P_{selFP}	$P_{selLR} (s = 2)$	$P_{selLR} (s = 1.5)$
A	1	1	0.1	0	0.167
B	5	2	0.5	0.67	0.5
C	4	2	0.4	0.33	0.33
Sum	10		1.0	1.0	1.0



Rankeamento Exponencial

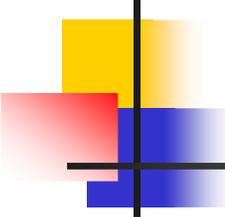
$$P_{exp-rank}(i) = \frac{1 - e^{-i}}{c}$$

- O Rankeamento linear é limitado pela pressão de seleção
- Rankeamento exponencial pode alocar mais do que duas cópias do indivíduo de melhor fitness
- O fator de normalização c é dependente do tamanho da população



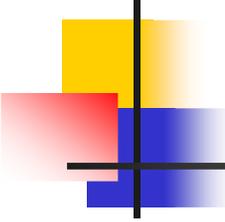
Seleção por Torneio

- Os métodos citados atuam sobre uma estatística global da população
 - Poderia ser um gargalo sobre máquina paralelas
- Procedimento informal
 - Pegue k membros de forma aleatória e então seleciona o melhor dentre eles
 - Repita o processo para a seleção de mais indivíduos



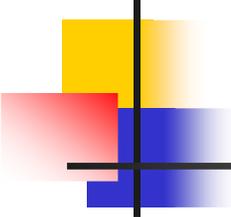
Seleção por Torneio

- A Probabilidade de seleção do indivíduo i irá depender:
 - Rank de i
 - Tamanho da amostra k
 - Quanto maior k maior a pressão de seleção
 - Competidores com recolocação
 - Se competidores sem recolocação – aumento da pressão de seleção
 - Se o mais alto fitness sempre ganha (determinístico) ou se este acontece com probabilidade p
- Para $k=2$, o tempo para o indivíduo de mais alto fitness assumir a população é o mesmo que o rankeamento linear com $s=2 \cdot p$



Seleção por Sobrevivência

- A maior parte dos métodos citados são usados para a seleção dos pais
- A seleção por sobrevivência pode ser dividida em duas classes
 - Seleção baseada em era
 - AGS
 - Seleção baseada em fitness



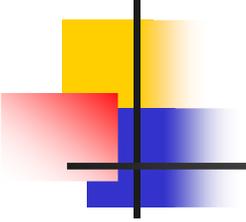
Duas Classes Especiais

- **Elitismo**

- Inteiramente utilizado
- Mantêm pelo menos uma cópia da melhor solução

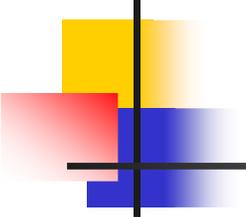
- **GENITOR**

- Processo de deletar o pior indivíduo
- Rápido domínio: é usado com grandes populações (~ centenas de indivíduos) ou com uma política de não duplicatas de indivíduos



Exemplo: AG Baseado em Ordem

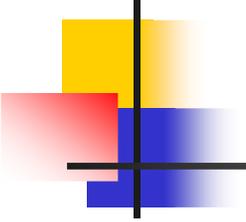
- Problema de escalonamento de tarefas vinculadas
 - J é um conjunto de tarefas
 - O é um conjunto de operadores
 - M é um conjunto de máquinas
 - $ABLE \subseteq O \times M$ define quais máquinas podem desempenhar quais operadores
 - $PRE \subseteq O \times O$ define quais operadores podem preceder outros operadores
 - $DUR: \subseteq O \times M \rightarrow \mathbb{IR}$ define a duração de $o \in O$ em $m \in M$



Exemplo: AG Baseado em Ordem

- A meta:
 - Encontrar o escalonamento tal que
 - Completo: Todas as tarefas sejam escalonadas
 - Correto: Todas as condições definidas por ABLE e por PRE sejam satisfeitas
 - Ótima: O Custo total do escalonamento seja o mínimo

Exemplo: AG Baseado em Ordem

- 
- Representação: Os indivíduos são permutações
 - As Permutações são decodificadas para os escalonamentos
 - O fitness de uma permutação é o inverso de seu custo
 - Use quaisquer operadores adequados de mutação e cruzamento
 - Use uma seleção de sobrevivência
 - Use inicialização aleatória

Exemplo: Escalonamento de Máquinas - Métodos

