

1. Computação Evolutiva

Prof. Renato Tinós

Programa de Pós-Graduação Em
Computação Aplicada

Depto. de Computação e Matemática
(FFCLRP/USP)

1.5. Aspectos de Projeto

- 1.5.1. Uso de Algoritmos Evolutivos (AEs)
- 1.5.2. Testes de AEs
- 1.5.3. Projeto de AEs
- 1.5.4. Desempenho de Aes
- 1.5.5. Controle de Parâmetros

1.5.1. Uso de Algoritmos Evolutivos (AEs)

- Experimentos com AEs
 - Qual o objetivo?
 - Desenvolvimento e/ou estudo de AEs?
 - Para problema específico?
 - Aplicação
 - Como avaliar?

1.5.1. Uso de Algoritmos Evolutivos (AEs)

Exemplo 1.5.1. Otimizar, por parte da empresa, entregas de compras feitas pela Internet

- Diferente destinos a cada dia
- Tempo limitado para executar o algoritmo

Deve achar uma solução razoável no tempo disponível

1.5.1. Uso de Algoritmos Evolutivos (AEs)

Exemplo 1.5.2. Otimizar o sistema nacional de tráfego em rodovias

- Custo total muito alto
 - Custo da computação é negligenciável
- Disponibilidade grande de
 - Tempo (meses) para executar os algoritmos
 - Recursos (Ex.: máquinas)

Deve produzir um resultado (uma única vez) muito bom (de preferência, o melhor possível)

1.5.1. Uso de Algoritmos Evolutivos (AEs)

- Perspectivas

- Design

- Ache uma solução muito boa uma única vez

- Produção

- Ache uma solução razoável várias vezes
 - As instâncias do problema mudam

- Academia

- Desenvolvimento de investigações científicas e novos AEs

1.5.2. Testes de Algoritmos Evolutivos

- No caso de comparações entre algoritmos, pode-se:
 - Usar instâncias de problemas a partir de repositório público ou já utilizados no meio acadêmico
 - Pode-se escolher problemas com diferentes características
 - Será que são problemas semelhantes aos problemas reais?
 - Usar instâncias de problemas aleatórios
 - Usar instâncias de problemas práticos
 - Dificuldade de comparar com os resultados obtidos por outros pesquisadores

1.5.2. Testes de Algoritmos Evolutivos

- Exemplo Ruim:
 - Desenvolver um novo AG
 - Comparar o AG padrão (?) com o novo AG
 - Testar em 10 problemas da literatura
 - Resultado: novo AG é melhor em 7, igual em 1, pior em 2 problemas
 - Conclui-se que....
- Será que essa é uma boa metodologia?
 - Deve-se responder às seguintes questões
 - Quão relevante são os resultados?
 - Deve-se levar em conta os resultados e as características de cada problema
 - Deve-se comparar com algoritmos que têm características semelhantes ou que têm bom desempenho nos problemas em que se espera que o novo algoritmo seja bom
 - Podem ser generalizados os resultados?

1.5.3. Projeto de Algoritmos Evolutivos

- Escolha de uma representação
- Escolha uma forma de mapear o genótipo no fenótipo
- Escolha uma forma de avaliar os indivíduos
- Escolha operadores de reprodução
 - Exemplo: recombinação e mutação
- Escolha operadores de seleção
 - Quem serão os pais?
 - Quem será escolhido para compor a próxima população
- Decida como inicializar a população e os parâmetros e se estes último serão controlados (e como)
- Decida o critério de parada

1.5.3. Projeto de Algoritmos Evolutivos

- **AEs são estocásticos**
 - Não se pode tirar conclusões a partir de uma única execução
 - Deve-se executar independentemente o AE várias vezes
 - com diferentes sementes aleatórias
 - Usar medidas estatísticas
 - Médias, medianas, desvios
 - Usar testes estatísticos para gerar as conclusões
- **Experimentos com AEs envolvem comparação**
 - **Competição deve ser justa**
 - Uso da mesma quantidade de recursos
 - Exemplo: mesmo número de avaliações de solução
 - Uso de diferentes parâmetros
 - Uso das mesmas medidas de desempenho

1.5.4. Desempenho de Algoritmos Evolutivos

- Várias medidas de desempenho são utilizadas
 - Exemplos
 - Resultado médio em um dado tempo
 - Tempo médio para um dado resultado
 - Proporção de execuções que resultaram em sucesso (Ex.: achar o ótimo global)
 - Melhor resultado em n execuções
 - ...

1.5.4. Desempenho de Algoritmos Evolutivos

- Unidades de tempo
 - Tempo decorrido?
 - Depende da máquina
 - Tempo de CPU?
 - Depende de características de implementação
 - Gerações?
 - E se o tamanho da população muda durante a execução?
 - Avaliações da função de fitness?
 - Geralmente usada
 - Deve-se tomar cuidado quando representações diferentes são usadas

1.5.4. Desempenho de Algoritmos Evolutivos

- **Medidas de Desempenho**
 - *off-line*
 - **Eficiência** (velocidade do alg.)
 - Tempo de CPU
 - Números de pontos gerados no espaço de busca
 - **Evetividade** (qualidade do alg.)
 - Taxa de sucesso
 - Qualidade da solução no final
 - *on-line*
 - Distribuição da população (genótipo)
 - Distribuição do fitness (fenótipo)
 - Melhorias no fitness do melhor indivíduo por unidade de tempo ou por operação de reprodução
 - ...

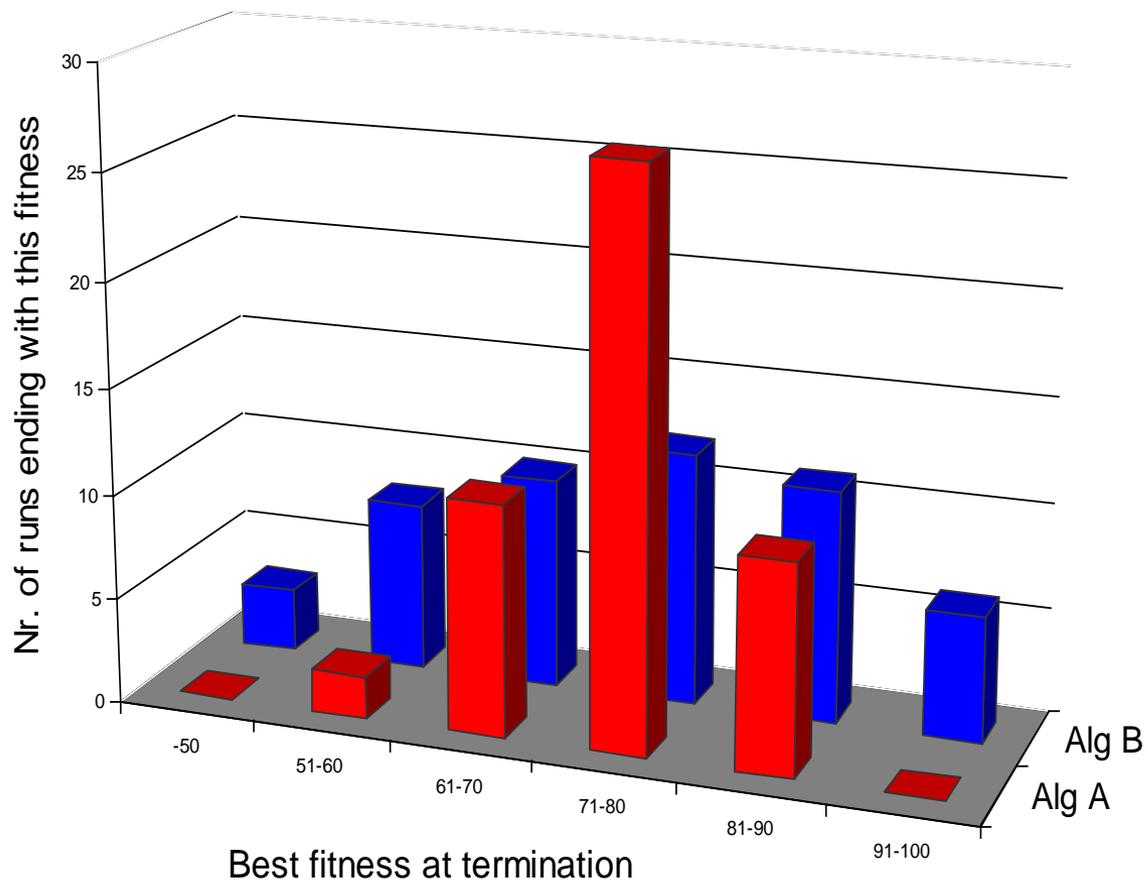
1.5.4. Desempenho de Algoritmos Evolutivos

- Medidas de Desempenho
 - **AES**: *average no. of evaluations to solution*
 - **SR**: *success rate*
 - % das execuções que acharam a solução (ótimo global ou igual a uma qualidade pré-definida)
 - **MBF**: *mean best fitness*
 - Média dos melhor fitness encontrado em cada execução
 - As vezes, é melhor usar outras medidas
 - Mediana
 - Melhor e pior
 - Quartis

1.5.4. Desempenho de Algoritmos Evolutivos

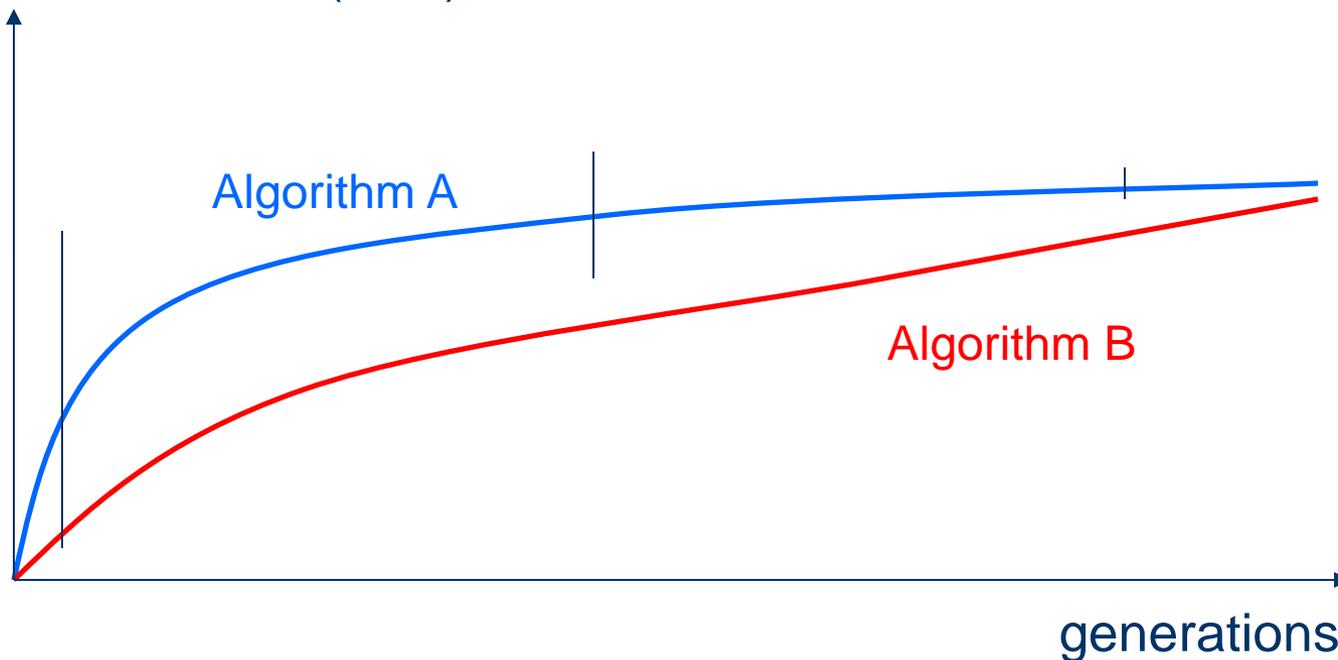
Exemplo:

Qual algoritmo é melhor?
Por que?
Quando?



1.5.4. Desempenho de Algoritmos Evolutivos

Populations mean (best) fitness



Qual algoritmo é melhor? Por que? Quando?

1.5.4. Desempenho de Algoritmos Evolutivos

- Comparação estatística e significância
 - Necessária quando diferentes algoritmos (ou com diferentes parâmetros) são comparados

1.5.4. Desempenho de Algoritmos Evolutivos

- Comparação estatística e significância
 - Exemplo:

| Trial | Old Method | New Method |
|---------|------------|------------|
| 1 | 500 | 657 |
| 2 | 600 | 543 |
| 3 | 556 | 654 |
| 4 | 573 | 565 |
| 5 | 420 | 654 |
| 6 | 590 | 712 |
| 7 | 700 | 456 |
| 8 | 472 | 564 |
| 9 | 534 | 675 |
| 10 | 512 | 643 |
| Average | 545.7 | 612.3 |

1.5.4. Desempenho de Algoritmos Evolutivos

- Comparação estatística e significância
 - Exemplo:

| Trial | Old Method | New Method |
|---------|-------------------|------------|
| 1 | 500 | 657 |
| 2 | 600 | 543 |
| 3 | 556 | 654 |
| 4 | 573 | 565 |
| 5 | 420 | 654 |
| 6 | 590 | 712 |
| 7 | 700 | 456 |
| 8 | 472 | 564 |
| 9 | 534 | 675 |
| 10 | 512 | 643 |
| Average | 545.7 | 612.3 |
| SD | 73.5962635 | 73.5473317 |
| T-test | 0.07080798 | |

- Desvio-padrão fornece informação adicional
- Teste t (ou similares) indica a chance de os valores serem provenientes da mesma distribuição (ex.: 7% de chance neste exemplo).

1.5.4. Desempenho de Algoritmos Evolutivos

- Teste t assume:
 - Distribuição normal
 - Deve-se checar através de métodos apropriados
 - Variância similar quando existem muito pouco dados
 - Grupos de dados com a mesmo tamanho
- Outros teste:
 - *Wilcoxon signed-rank test*
 - Não paramétrico

1.5.5. Controle de Parâmetros

- Motivação
 - Algoritmos Evolutivos (AEs) têm diversos parâmetros que definem a estratégia de busca
 - Exemplos
 - Operador de mutação e taxa de mutação
 - Operador de recombinação (ou crossover) e respectiva taxa
 - Mecanismo de seleção e pressão seletiva
 - Tamanho da população
 - Ajuste dos parâmetros influencia diretamente o desempenho do processo evolutivo
 - Parâmetros de um AE são geralmente fixos durante uma execução
 - Questão 1: Como encontrar bons valores dos parâmetros de um AE?

1.5.5. Controle de Parâmetros

- **Ajuste de Parâmetros**
 - Método tradicional
 - Teste e comparação através de execuções do AE com diferentes valores dos parâmetros
 - Deve-se alterar um parâmetro de cada vez
 - Problemas
 - Requer tempo
 - Erros do ajuste provocam desempenho ruim
 - Parâmetros são relacionados
 - Busca exaustiva não é viável
 - Bons valores podem se tornar ruins durante a execução

1.5.5. Controle de Parâmetros

- Motivação
 - AE são sistemas dinâmicos
 - Assim, valores dos parâmetros ótimos podem variar durante uma execução
 - Questão 2: Como variar os valores dos parâmetros?

1.5.5. Controle de Parâmetros

- **Controle de Parâmetros**
 - Valores dos parâmetros são ajustados on-line (durante a execução)
 - Métodos
 - **Determinista**: regras pré-definidas
 - **Adaptativo**: usando feedback do processo de busca
 - **Auto-adaptativo**: codificando os parâmetros no cromossomo e permitindo que eles sejam modificados pelo processo evolutivo

1.5.5.1. Exemplos

- Tarefa:

$$\min f(x_1, \dots, x_n)$$

$$L_i \leq x_i \leq U_i \quad , \quad i = 1, \dots, n \quad // \text{ limites}$$

$$g_i(\mathbf{x}) \leq 0 \quad , \quad i = 1, \dots, q \quad // \text{ rest. de desigualdade}$$

$$h_i(\mathbf{x}) = 0 \quad , \quad i = q+1, \dots, m \quad // \text{ rest. de igualdade}$$

- Estratégia

- AE com representação real (x_1, \dots, x_n)

- Recombinação intermediária (média)

- Mutação Gaussiana:

$$x'_i = x_i + N(0, \sigma)$$

desvio-padrão σ é chamado de tamanho do passo de mutação

1.5.5.1. Exemplos

- Controle determinista do tamanho do passo de mutação
 - Substitua σ constante por função $\sigma(t)$ determinista
 - Exemplo

$$\sigma(t) = 1 - 0,9 \frac{t}{T}$$

sendo $0 \leq t \leq T$ o índice da geração

- Características
 - Mudança no parâmetro é independente do processo evolutivo
 - Controle do parâmetro
 - É ditado pelo usuário
 - É previsível
 - Age sob todos os indivíduos da população

1.5.5.1. Exemplos

- *Controle adaptativo do tamanho do passo de mutação*
 - *Substitua σ constante por função $\sigma(t)$ adaptativa*
 - *Depende da evolução do processo explicitamente*
 - *Exemplo: Regra do 1/5 de sucesso (ES)*
 - *Mude $\sigma(t)$ a cada k gerações de acordo com:*

$$\sigma(t) = \begin{cases} \sigma(t-1) / c & \text{se } p_s > 1/5 \\ \sigma(t-1) c & \text{se } p_s < 1/5 \\ \sigma(t-1) & \text{caso contrário} \end{cases}$$

Sendo: t o índice da geração; $0,817 \leq c < 1$; p_s a taxa de mutações benéficas durante as últimas k gerações

- *Características*
 - *Mudança no parâmetro é dependente do progresso do processo evolutivo*
 - *Controle do parâmetro*
 - *Não é previsível*
 - *Age sob todos os indivíduos da população*

1.5.5.1. Exemplos

- *Controle auto-adaptativo do tamanho do passo de mutação*
 - *Cada indivíduo é associado a um σ*
 - *Pode associar um σ_i a cada dimensão do problema*
 - *Cromossomo: $(x_1, \dots, x_n, \sigma)$*
 - *Cada σ individual é sujeito a seleção e mutação (e recombinação)*
 - *Exemplo*

$$\sigma' = \sigma \times e^{N(0, \tau)}$$

$$x'_i = x_i + N(0, \sigma')$$

- *Características*
 - *Mudança no parâmetro é dependente de processo evolutivo*
 - *Controle do parâmetro*
 - *Não é previsível*
 - *Age separadamente em cada indivíduo da população*

1.5.5.1. Exemplos

- Restrições

$$g_i(\mathbf{x}) \leq 0 \quad , i = 1, \dots, q \quad // \text{rest. de desigualdade}$$

$$h_i(\mathbf{x}) = 0 \quad , i = q+1, \dots, m \quad // \text{rest. de igualdade}$$

- Possível solução

- Incorporar penalidades na função de avaliação

$$\text{fitness}(\mathbf{x}) = f(\mathbf{x}) + W \mathbf{p}(\mathbf{x})$$

sendo W uma cte. positiva (peso) e

$$p(\mathbf{x}) = \sum_{i=1}^m \begin{cases} 1 & \text{se restrição } i \text{ é violada} \\ 0 & \text{se restrição } i \text{ não é violada} \end{cases}$$

1.5.5.1. Exemplos

- Controle determinista das penalidades
 - Substitua a constante W pela função $W(t)$

$$W(t) = (C t)^\alpha$$

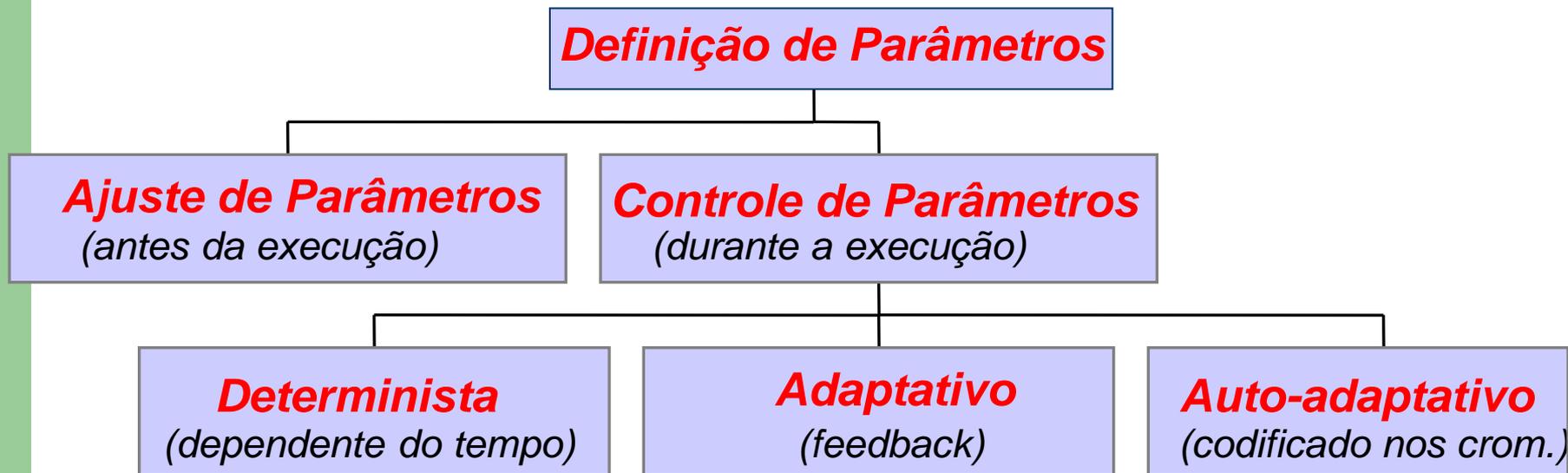
Sendo $0 \leq t \leq T$ o índice da geração, C e α valores positivos maiores ou iguais a 1

- Controle adaptativo e auto-adaptativo também podem ser empregados para o controle das penalidades

1.5.5.2. Aspectos do Controle de Parâmetros

- Praticamente qualquer componente de um AE pode ser parametrizado e controlado
 - Exemplos:
 - Representação
 - População
 - Exemplo: tamanho da população
 - Operadores de reprodução
 - Funções de avaliação
 - Operadores de seleção
 - Exemplo.: pressão seletiva

1.5.5.2. Aspectos do Controle de Parâmetros



1.5.5.2. Aspectos do Controle de Parâmetros

- As mudanças nos parâmetros podem ser baseadas em:
 - **Tempo ou número de avaliações** (Controle determinista)
 - **Estatísticas sobre população** (Controle adaptativo)
 - Progresso durante execução
 - Diversidade da População
 - Distribuição dos genes
 - Etc.
 - **Relativa ao fitness** dos indivíduos criados com determinados parâmetros (Controle adaptativo e controle auto-adaptativo)

Comentários

- Referências
 - Eiben, A. E. & Smith, J. E. *Introduction to Evolutionary Computation*. Springer, 2003
 - Capítulo 8
 - Capítulo 14
- Agradecimentos
 - Parte do material desta apresentação foi obtida através de
 - Material de apoio do livro Eiben, A. E. & Smith, J. E. *Introduction to Evolutionary Computation*. Springer, 2003.