

# Aula 6

# Algoritmos Genéticos

**Prof. Dr. Peterson A. Belan**

# Algoritmos Genéticos (AG)

É um método de busca e otimização baseado no processo de seleção natural que simula a evolução das espécies (GOLDBERG, 1989; GOLDBARG; LUNA, 2000).

Os AGs geram os indivíduos (cadeias de bits) também chamados de cromossomo, para evoluírem em busca da solução de um dado problema. Cada indivíduo representa uma possível solução para o problema e são analisados conforme as necessidades.

Por meio de uma “competição”, os mais aptos são selecionados e em seguida cruzados para gerarem novos indivíduos mais aptos e melhores que os anteriores. Assim, a cada geração há uma maior probabilidade de um dos indivíduos estar próximo de uma solução para o problema.

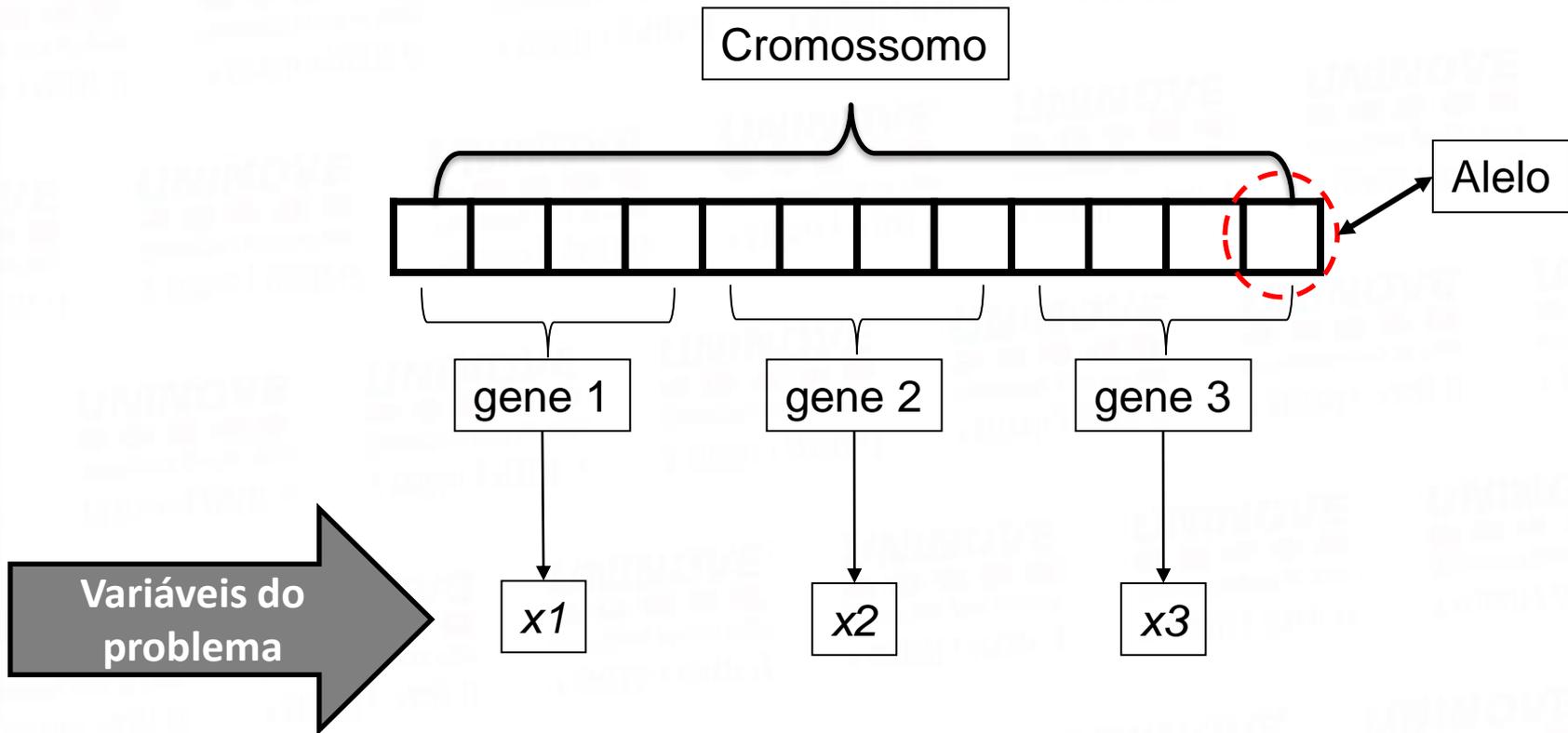
# Algumas características do AG

- 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.

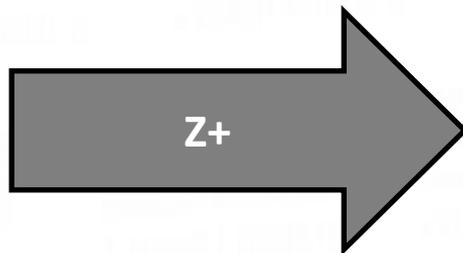
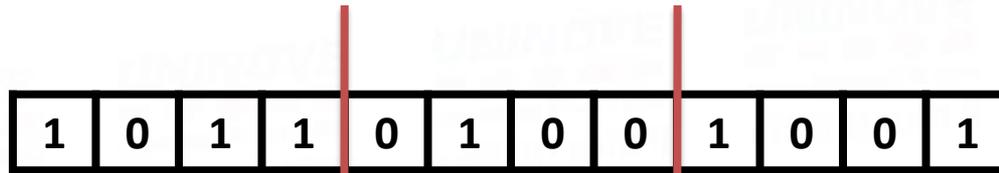
# Elementos do AG

<b>AG</b>	<b>Problema de Otimização</b>
Cromossomo	Representação de uma solução: string, número binário, número real, vetor, etc
Indivíduo	Solução de um problema
População	Conjunto de soluções
Gene	Parte da representação de uma solução; uma variável
Alelo	Valor que um gene pode assumir

# Estrutura de um Cromossomo Binário



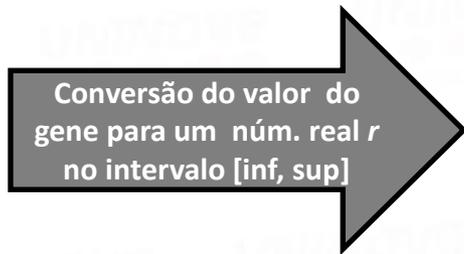
# Decodificação Cromossomo Binário



$$X1=11$$

$$X2=4$$

$$X3=9$$



$$r = \text{inf}_i + \frac{\text{sup}_i - \text{inf}_i}{2^k - 1} * v_i$$

Onde: **inf<sub>i</sub>** e **sup<sub>i</sub>** representam os valores limites da variável **i**; **v<sub>i</sub>** é o valor do gene **i** (ou variável **i**) na base 10; **k** é o comprimento do gene.

Ex. Se  $r = 1011$ ,  $\text{inf} = -5,0$  e  $\text{sup} = 5,0$ , então:

$$r = -5 + \frac{5 - (-5)}{2^4 - 1} * 11 = 2,3$$

• O comprimento do gene ( $k$ ) é dado por:  $2^k > (\text{sup}_i - \text{inf}_i) * 10^{p-1}$ , onde  $p$  é a precisão

# Cruzamento e Mutação

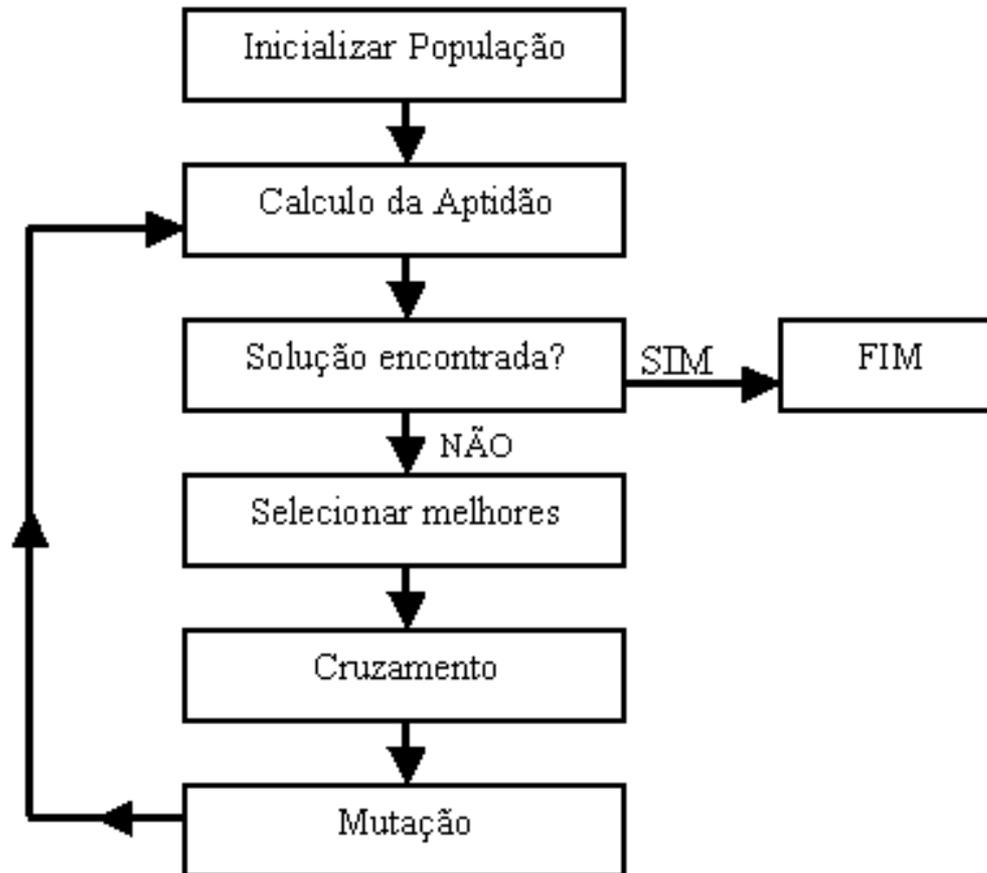
Pai 1 = 10001|10101

Pai 2 = 11101|11011

Filho 1 = 1000111011

Filho 2 = 1110110101  mutação → 1110111101

# Estrutura básica de um AG



# AG

Início

Seja  $s'$  a melhor solução da geração corrente,  $s^*$  a melhor solução obtida até então, Ger o contador do número de gerações e Max\_Ger o número máximo de gerações permitidas

Ger  $\leftarrow$  1

Gera população inicial

Avalia população {cálculo da aptidão dos indivíduos}

$s^* \leftarrow \text{MelhorIndivíduo}(\text{população inicial})$

Enquanto Ger  $\leq$  Max\_Ger ou algum outro critério de parada não for atingido

Faz seleção dos melhores indivíduos

Efetua o cruzamento entre os melhores indivíduos para recompor a população

Faz mutação de um percentual dos indivíduos da população

Avalia população {cálculo da aptidão dos indivíduos}

$s' \leftarrow \text{MelhorIndivíduo}(\text{população corrente})$

Se  $f(s') > f(s^*)$  então

$s^* \leftarrow s'$

Fim\_se

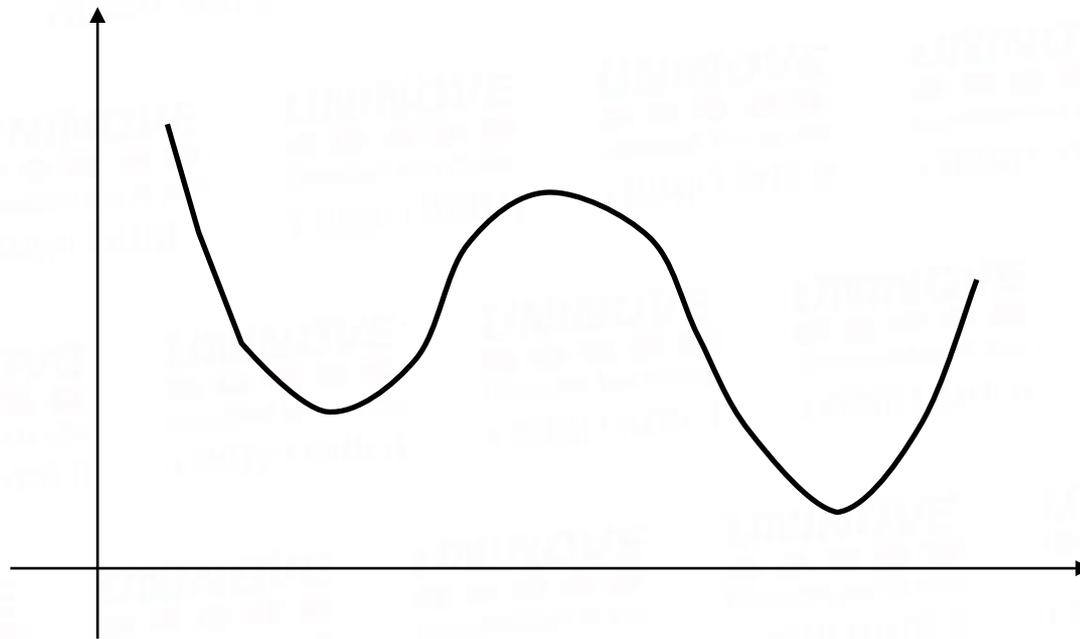
Ger  $\leftarrow$  Ger + 1

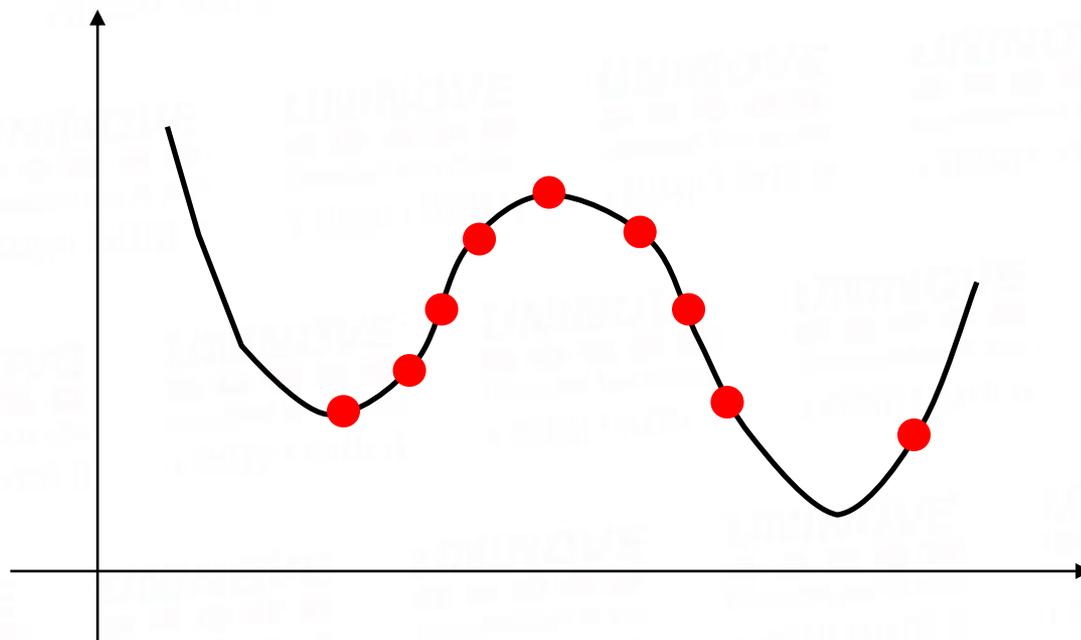
Fim\_enquanto

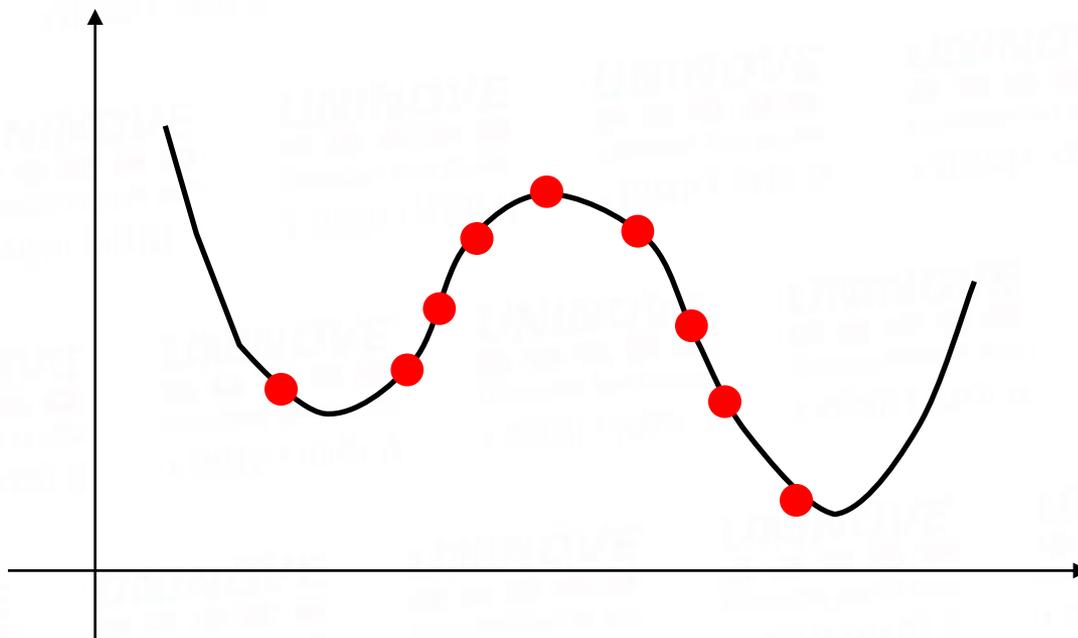
Retorna  $s^*$

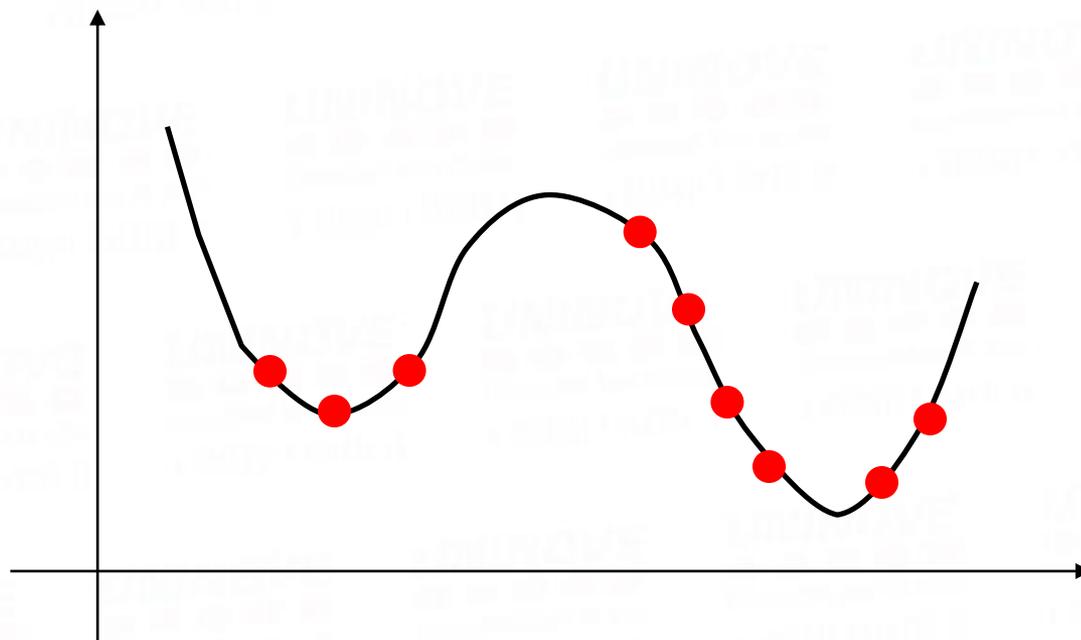
Fim

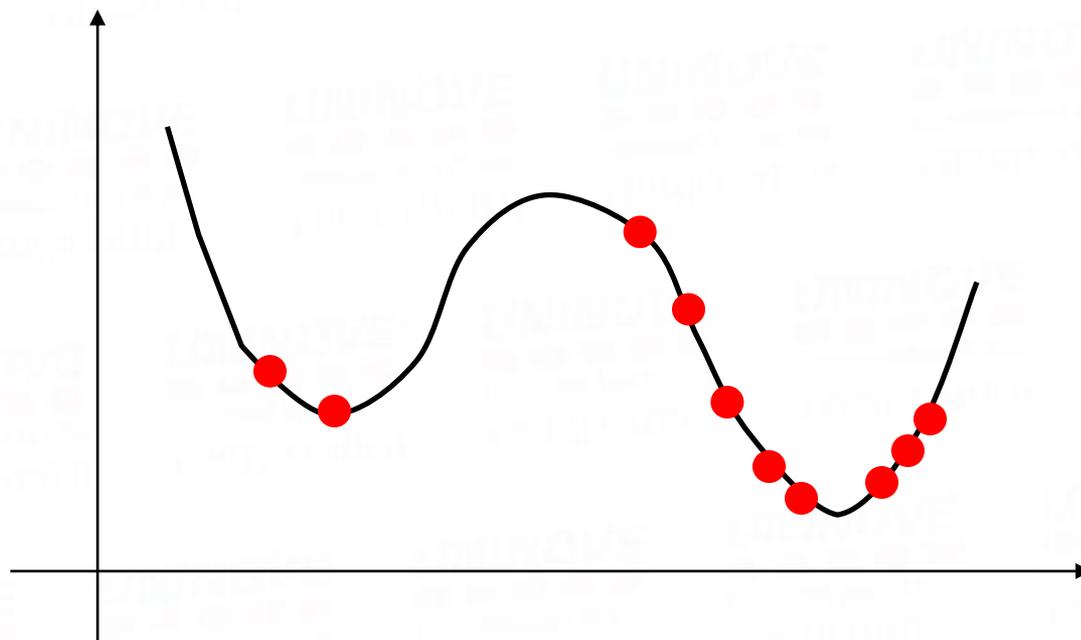
# Algoritmos Genéticos (AG) - Simulação

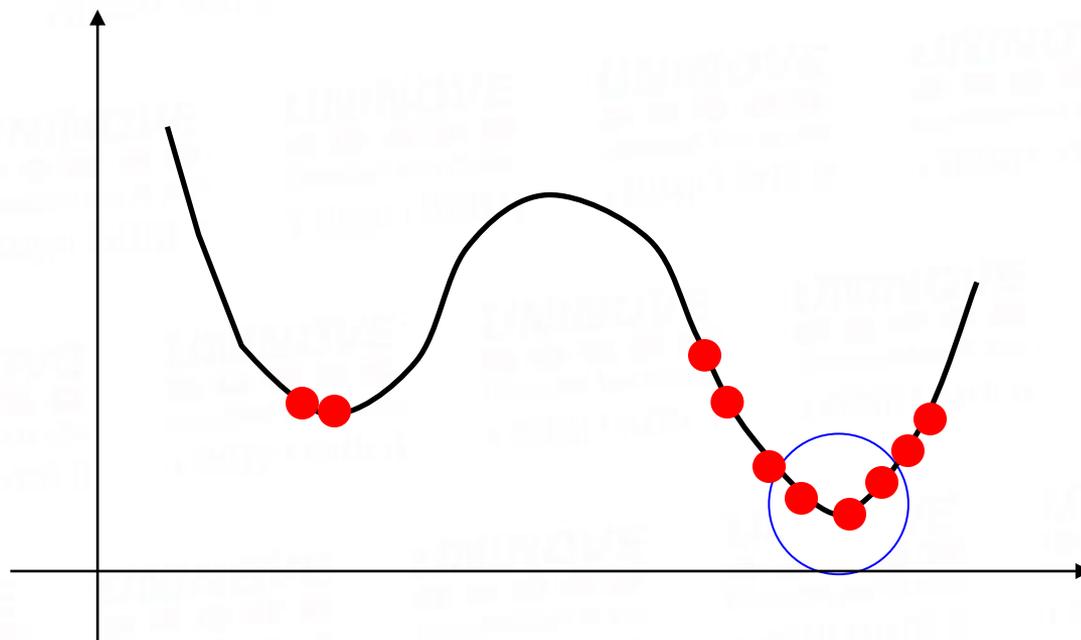












# Exemplo de Modelagem do AG

**Desenvolvimento de um AG para solucionar o seguinte problema de otimização<sup>1</sup>:**

Minimizar  $f(x) = |2X+10-18|$

sujeito a  $x \in \mathbb{N}, x \leq 15$

**Para isso é preciso definir:**

- Tamanho do cromossomo
- Quantidade de genes do cromossomo
- Ponto(s) de corte
- Tamanho da população
- Esquema de seleção
- Taxa de mutação

*1. Extraído da revista Revista PC Master Edição 64.*

# AG – Exercício 1 – Problema de otimização da produção

Considere uma microempresa que fabrica dois tipos de produto: rádio standard e rádio luxo. Com relação ao rádio standard temos as seguintes informações:

- (i) A linha de produção comporta um máximo de 24 pessoas;
- (ii) Cada rádio consome 1 homem/dia para ser produzido;
- (iii) Cada rádio fornece um lucro de R\$ 30,00.

Com relação ao rádio luxo:

- (i) A linha de produção comporta um máximo de 32 pessoas;
- (ii) Cada rádio consome 2 homens/dia para ser produzido;
- (iii) Cada rádio fornece um lucro de R\$ 40,00.

A fábrica possui 40 empregados a serem alocados nas duas linhas de produção. O objetivo é maximizar o lucro. Quantos rádios de cada tipo (standard e luxo) devem ser produzidos para maximizar o lucro?

Tal problema pode ser representado matematicamente da seguinte maneira:

$$\textit{Maximizar } f(x_1, x_2) = 30x_1 + 40x_2$$

$$\textit{Sujeito a: } x_1 + 2x_2 \leq 40$$

$$x_1 \leq 24$$

$$x_2 \leq 16$$

$$x_1 \geq 0, x_2 \geq 0,$$

# Algoritmos Genéticos

## Exercícios usando Matlab

# AG – Exercício prático 1

Desenvolvimento de um AG para solução do seguinte problema de otimização:

Minimizar  $f(x) = 2X^5 - 3X^3 * X^2$   
sujeito a  $x \in R; 0,001 \leq x \leq 10,000$

Tomando como base os problemas descritos, pede-se:

a) Modele AGs para resolvê-los

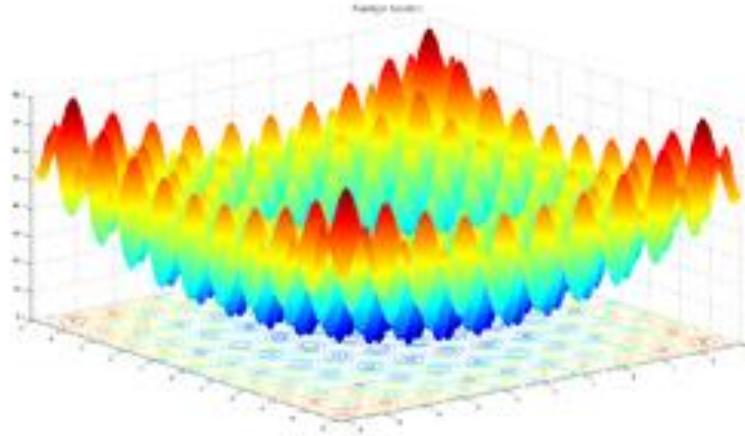
b) Apresente as simulações usando o MATLAB

Obs.: Vale ressaltar que este último exercício é válido apenas para fins didáticos, tendo em vista que sua solução poderia ser facilmente obtida usando métodos clássicos de Pesquisa Operacional como, por exemplo, o método gráfico.

# Alguns Algoritmos Metaheurísticos

- Método de Busca em Vizinhança Variável (VNS)
- Iterated Local Search (ILS)
- Greedy Randomized Adaptative Search Procedure (GRASP)
- Busca Tabu
- Simulated Annealing
- Algoritmos Genéticos (AG)
- Colônia e Formigas
- Otimização por Enxame de Partículas

# AG – Desafio (minimizar a função rastrigin)



$$f(\mathbf{x}) = An + \sum_{i=1}^n [x_i^2 - A \cos(2\pi x_i)]$$

$$A = 10$$

$$n = 2;$$

$$x_i \in [-5.12, 5.12]$$

Mínimo Global

$$\mathbf{x} = \mathbf{0} \quad f(\mathbf{x}) = 0$$

```
function y = rastrigin(x)
A = 10;
n = 2;
s = 0;
for j = 1:n
    s = s+(x(j)^2-A*cos(2*pi*x(j)));
end
y = A*n+s;
```

# Referências

GOLDBARG, M. C. & LUNA, H. P., Otimização combinatória e programação linear: modelos e algoritmos, 3ª Edição. Rio de Janeiro: Editora Campus, 2000.

GOLDBERG, D. E. Genetic algorithms in search, optimization, and machine learning. Addison-Wesley, 1989.

LINDEN, R. Algoritmos Genéticos. Brasport, 2006.

LUKE, S. Essentials of metaheuristics. 2. ed. Raleigh: Lulu, 2013.

MITCHELL, M. An Introduction to Genetic Algorithms. The MIT Press, 1998.

RUSSELL, S. J.; NORVIG, P. Inteligência artificial . 2. ed. Rio de Janeiro: Campus, 2004.

SOUZA, M. J. F. Inteligência Computacional para Otimização, Notas de aula, Departamento de Computação, Universidade Federal de Ouro Preto, disponível em: <http://www.decom.ufop.br/prof/marcone>.