

# **Algoritmos bio-inspirados e neuroevolução**

**por Diogo Trentini**

# Sumário

- Problemas de otimização
  - Otimização contínua
  - Otimização combinatorial
- Algoritmos bio-inspirados
  - Ant Colony Optimization
  - Algoritmos evolucionários
  - Exemplo (GA)
- Neuroevolução (EA + ML)

# Problemas de otimização

- **Objetivo:** Achar a melhor solução dentre todas as soluções possíveis.
- Duas categorias:
  - **Problemas de otimização contínua:** procura-se uma solução definida no domínio dos números reais ( $\mathbb{R}$ ).
  - **Problemas de otimização combinatorial:** procura-se uma solução representada por um número inteiro ( $\mathbb{I}$ ), grafo ou qualquer representação de um conjunto finito.

# Otimização contínua

- Definido por:

$$\begin{array}{ll} \text{minimizar}_{\vec{x}} & f(x) \\ \text{sujeito a} & g_i(\vec{x}) \leq 0, i = 1, \dots, l \\ & h_j(\vec{x}) = 0, j = 1, \dots, m \\ \text{onde} & f(x) : \mathbb{R}^n \rightarrow \mathbb{R} \end{array}$$

# Otimização contínua

- Exemplo: despacho econômico de energia elétrica.

$$\text{minimizar}_x \sum_{i=1}^6 f(x_i) \quad \text{sujeito a } \sum_{i=1}^6 x_i = 1263$$

$$\begin{array}{ll} f(x_1) = 0.007x_1^2 + 7x_1 + 240, & 100 \leq x_1 \leq 500 \\ f(x_2) = 0.0095x_2^2 + 10x_2 + 200, & 50 \leq x_2 \leq 200 \\ f(x_3) = 0.009x_3^2 + 8.5x_3 + 220, & 80 \leq x_3 \leq 300 \\ f(x_4) = 0.009x_4^2 + 11x_4 + 200, & 50 \leq x_4 \leq 150 \\ f(x_5) = 0.008x_5^2 + 10.5x_5 + 220, & 50 \leq x_5 \leq 200 \\ f(x_6) = 0.0075x_6^2 + 12x_6 + 190, & 50 \leq x_6 \leq 120 \end{array}$$

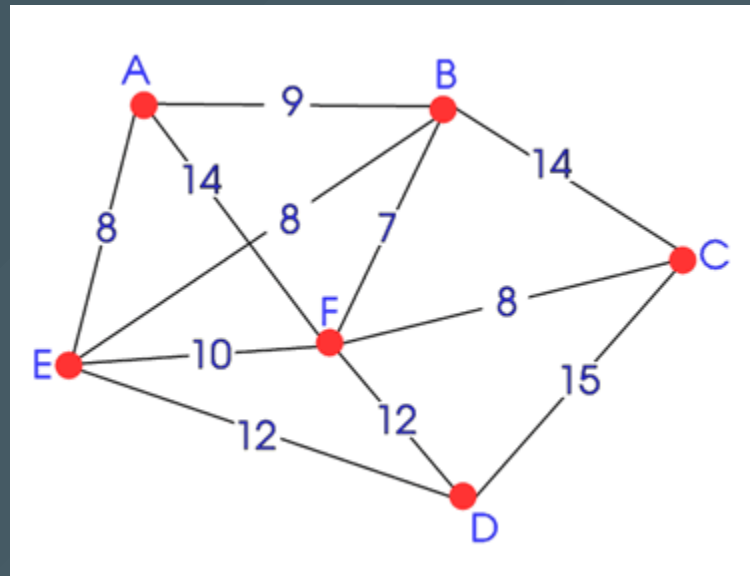
# Otimização combinatorial

- Definido por  $(I, f, m, g)$ , onde:
  - $I$  é um conjunto de instâncias  $x$  ( $x \in I$ ).
  - $f(x)$  é o conjunto de soluções possíveis.
  - Dado uma instância  $x$  com uma solução  $y$ ,  $m(x, y)$  é a medida de  $y$  (normalmente  $\in \mathbb{R}$ ).
  - $g$  é a função objetivo (min/max) onde devemos encontrar, para  $x$ , uma solução  $y$  com:

$$m(x, y) = g\{m(x, y') \mid y' \in f(x)\}, \quad x \in I$$

# Otimização combinatorial

- Exemplo: problema do caixeiro-viajante. Dado a lista de cidades abaixo (com as distâncias entre cada par de cidades), qual é a menor rota onde o caixeiro visita cada cidade exatamente uma vez e volta ao local de origem (no caso, cidade A)?



# Algoritmos bio-inspirados

- Algoritmos que simulam comportamentos inteligentes da natureza para resolverem problemas (normalmente de otimização).
- Normalmente não garantem encontrar a **melhor** solução, mas encontram boas soluções com menor custo computacional (se comparado a outros métodos).



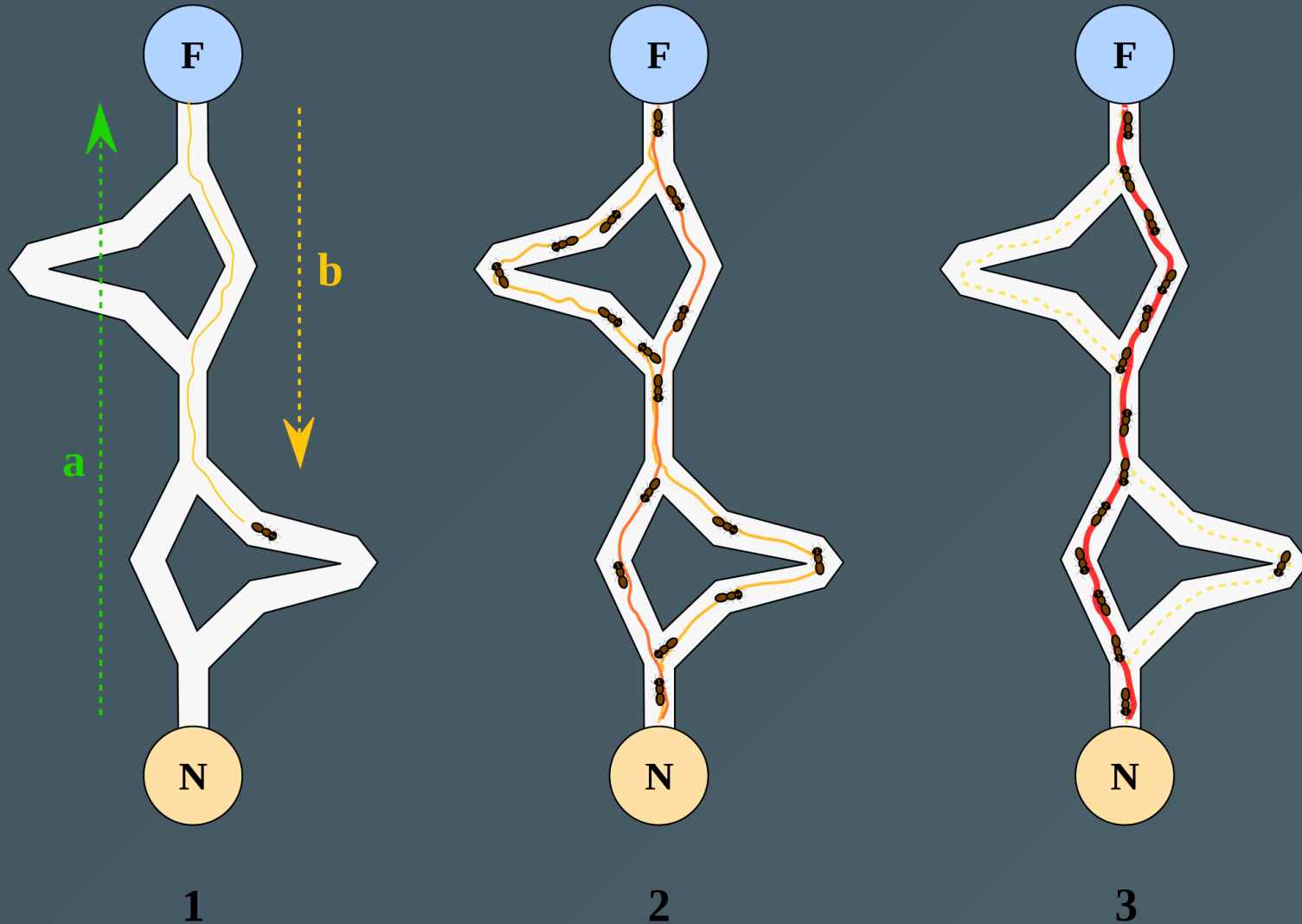
# Algoritmos bio-inspirados

- Duas principais categorias:
  - Inteligência de enxame:
    - ACO (**A**nt **C**olony **O**ptimization)
    - PSO (**P**article **S**warm **O**ptimization)
  - Evolução:
    - GA (**G**enetic **A**lgorithm)
    - DE (**D**ifferential **E**volution)

# Ant Colony Optimization



# Ant Colony Optimization



# Algoritmos evolucionários

- Algoritmos baseados no processo evolutivo.
- Alguns tipos:
  - Algoritmo genético
  - Evolução diferencial
  - Estratégia evolutiva
  - Programação genética
  - Neuroevolução

# Algoritmos evolucionários

- Pseudo-código (algoritmo genético):

```
Generate initial population randomly
Evaluate each individual's fitness

while (termination condition not met) do
    Select individuals (usually the best-fit) for \
        reproduction
    Generate new individuals with crossover and mutation \
        operations
    Evaluate each individual's fitness
    Replace population (usually the least-fit) with new \
        individuals
end
```

# Exemplo (GA)

$$\text{minimizar}_x \quad f(x) = x^2 - 2x, \quad -30 \leq x \leq 30$$

População inicial:

ID	Gene	Solução ( $x$ )	Fitness
A	010011	19	$19^2 - 2 * 19 = 323$
B	000111	7	$7^2 - 2 * 7 = 35$
C	001100	24	$24^2 - 2 * 24 = 528$
D	011000	12	$12^2 - 2 * 12 = 120$

# Exemplo (GA)

- Reprodução: B (fitness 35) e D (fitness 120).
- Crossover:

Pais	Gene	Filhos	Gene
B	000111	E	000000
D	011000	F	011111

- Mutação (apenas F):

Pais	Gene	Filhos	Gene
F	011111	F'	001111



# Exemplo (GA)

- Cálculo do fitness:

ID	Gene	Solução ( $x$ )	Fitness
E	000000	0	$0^2 - 2 * 0 = 0$
F	001111	15	$15^2 - 2 * 15 = 195$



# Exemplo (GA)

- Seleção natural (substituindo mais fracos):

ID	Gene	Solução ( $x$ )	Fitness
E	0000000	0	$0^2 - 2 * 0 = 0$
B	0001111	7	$7^2 - 2 * 7 = 35$
F	0011111	15	$15^2 - 2 * 15 = 195$
D	0110000	12	$12^2 - 2 * 12 = 120$

Média do fitness da população anterior: 251.5

Média do fitness da população atual: 87.5

# Neuroevolução

- Método que utiliza **algoritmos evolucionários** para evoluir/treinar **redes neurais**.
- Podem ser utilizadas para:
  - Evoluir os pesos das conexões de uma rede.
  - Evoluir a estrutura de uma rede.
- Podem evoluir as redes e parâmetros:
  - Simultaneamente (com algoritmos evolucionários normais)
  - Separadamente (com algoritmos meméticos)

# Neuroevolução

- Podem possuir diferentes tipos de codificação:
  - Direta: o genótipo (genes do algoritmos evolucionários) mapeia diretamente o fenótipo (topologia/parâmetros da rede neural). Ou seja, cada neurônio e conexão é definida no gene.
  - Indireta: o genótipo mapeia indiretamente o fenótipo. Ou seja, precisamos processar o genótipo para gerarmos o fenótipo.
    - Temos como vantagens a compressão do fenótipo (i.e., menor espaço de busca) e modularização da rede, por exemplo.

# Neuroevolução

- Exemplos:

Método	Codificação	Algoritmo	Aspectos
NEAT	Direta	GA	Estrutura e parâmetros
HyperNEAT	Indireta (com CPPN)	GA	Parâmetros
HyperNEAT	Indireta (com DPPN)	GA + MA (Memetic Algorithm)	Estrutura e parâmetros

# EXEMPLOS

- [Flexible Muscle-Based Locomotion for Bipedal Creatures](#)
- [MarI/O - Machine Learning for Video Games](#)
- [Inteligência Artificial com Dinossauro da Google](#)