# Tópicos em Complexidade Computacional

## P, NP, EXP e completude de prolemas

**Professor Murilo V. G. da Silva**

Departamento de Informática

Universidade Federal do Paraná

09/06/2022

# A classe $\mathrm{P}$

## Tempo determinístico: $\mathrm{DTIME}$

Dada $T : \mathbb{N} \to \mathbb{N}$, a classe $\mathrm{DTIME}(T(n))$ é o conjunto de toda linguagem $L$, tal que

- Existe uma MT $M$ que decide $L$
- A máquina $M$ tem complexidade de tempo $c \cdot T(n)$, para algum $c > 0$

Exemplos: $\mathrm{DTIME}(n^2)$, $\mathrm{DTIME}(n^3 \log n)$, $\mathrm{DTIME}(2^n)$, etc

## A classe $\mathrm{P}$

$$\mathrm{P} = \bigcup_{c \geq 1} \mathrm{DTIME}(n^c)$$

Exemplos de problemas em $\mathrm{P}$: Conectividade de grafos, multiplicação (versão decisão), teste de primalidade, etc.

Pergunta: A classe $\mathrm{P}$ mudaria se usássemos outro modelo de computação?

## A classe NP

Uma linguagem $L \subseteq \{0,1\}^*$ está em $\mathrm{NP}$ se existe um polinômio $p : \mathbb{N} \to \mathbb{N}$ e existe uma MT polinomial $M$ tal que $\forall x \in \{0,1\}^*$

$$x \in L \Leftrightarrow \exists u \in \{0,1\}^{p(|x|)} : M(x,u) = 1$$

- A máquina $M$ acima é chamada de verificador de $L$
- A string $u$ é chamada de certificado de $x$ (com relação à $L$ e $M$).

**Note:** A partir da definição, obtemos diretamente $\mathrm{P} \subseteq \mathrm{NP}$.

**Exercício 3.1:** Mostre que os seguintes problemas estão em $\mathrm{NP}$:

- conectividade de grafos, isomorfismo de grafos, conjunto independente
- satisfatibilidade
- testar se um número é composto
- fatoração (versão decisão)

## A classe EXP

$$\mathrm{EXP} = \bigcup_{c \geq 1} \mathrm{DTIME}(2^{n^c})$$

Teorema 3.2: $\mathrm{P} \subseteq \mathrm{NP} \subseteq \mathrm{EXP}$

Prova:

- $\mathrm{P} \subseteq \mathrm{NP}$ (slide anterior)
- $\mathrm{NP} \subseteq \mathrm{EXP}$:
  - Seja $L \in NP \quad \Rightarrow \quad \exists$ polinômio $p$ e $\exists$ verificador polinomial $M$
  - Mostraremos: Existe $M'$ exponencial para decidir se $x \in L$. Segue $M'$:
    - enumera cada $u$ e roda $M(x, u)$ (são $2^{p(|x|)}$ escolhas de $u$)
    - aceita string $x \Leftrightarrow M(x, u) = 1$
  - Complexidade de $M'$: para alguma constante $c$, tempo de $M'$ é $2^{O(n^c)}$
  - $L \in \mathrm{EXP}$

# Máquinas de Turing não determinísticas

Considere a função $f : \{0,1\}^* \to \{0,1\}^*$ e a função tempo construtível $T : \mathbb{N} \to \mathbb{N}$.

## Máquinas de Turing não determinística (MTN)

$M = (\Gamma, Q, (\delta_1, \delta_2))$ tal que

- $\Gamma = \{\triangleright, \square, 0, 1\}$
- $Q = \{q_{\text{START}}, q_{\text{HALT}}, q_{\text{ACCEPT}}\} \cup \{q_1, ..., q_n\}$
- $\delta : Q \times \Gamma^3 \to \Gamma^2 \times \{L, S, R\}^3$
- Aceita se existe computação levando a $q_{\text{ACCEPT}}$. (obs: estado explícito de aceitação)

Complexidade de tempo $T(n)$: $\forall x$, $\exists$ uma sequência de escolhas que leve a $q_{\text{ACCEPT}}$ ou $q_{\text{HALT}}$ em $T(|x|)$ passos.

## Tempo não determinístico: NTIME

Dada $T : \mathbb{N} \to \mathbb{N}$, a classe $\text{NTIME}(T(n))$ é o conjunto de toda linguagem $L$, tal que

- Existe uma MTN $M$ que decide $L$, e
- A complexidade de tempo de $M$ é $c \cdot T(n)$, para algum $c > 0$

# A classe $\mathrm{NP}$, reduções e completude

## Teorema 3.3

$$\mathrm{NP} = \bigcup_{c \geq 0} \mathrm{NTIME}(n^c)$$

## Reduções e completude

$L$ é (*Karp*) Redutível em tempo polinomial para $L'$ se $\exists$ MT polinomial $M$ tal que

- $x \in L \Leftrightarrow M(x) \in L'$

Notação: $L \leq_p L'$

- $L'$ é $\mathrm{NP}$-difícil: se $\forall L \in \mathrm{NP}$, $L \leq_p L'$.
- $L'$ é $\mathrm{NP}$-completa: se for $\mathrm{NP}$-difícil e estiver em $\mathrm{NP}$.

# A classe NP, reduções e completude

Redução de $L$ para $L'$:

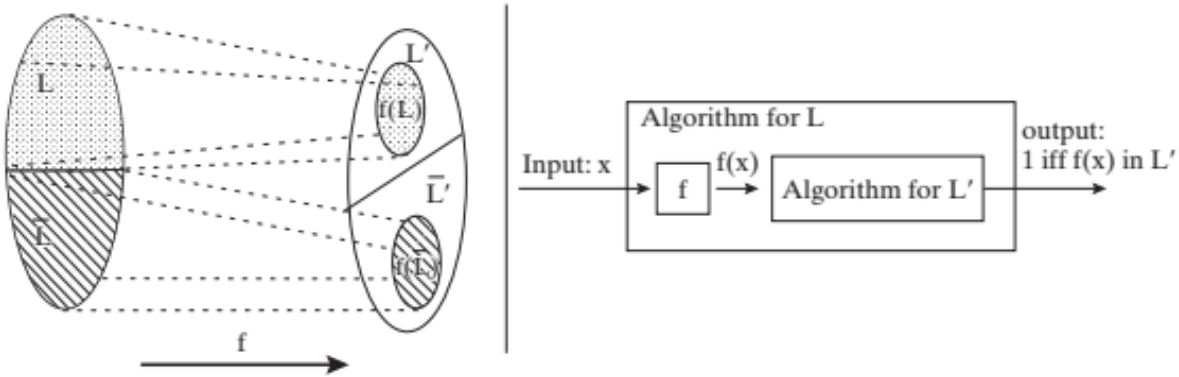


Figura: Barak & Arora, pág. 43

## Teorema 3.4

1. $L \leq_p L'$ e $L' \leq_p L'' \quad \Rightarrow \quad L \leq_p L''$.
2. $L$ é NP-dificíl e $L \in \mathrm{P} \quad \Rightarrow \quad \mathrm{P} = \mathrm{NP}$.
3. Seja $L$ é NP-completa. Então $L \in \mathrm{P} \Leftrightarrow \mathrm{P} = \mathrm{NP}$.

Prova (principal argumento): Dados polinômios $p(n)$ e $q(q)$, a função composta $p(q(n))$ também é um polinômio.

# Existem problemas NP-completos?

## Teorema 3.5

A seguinte linguagem é NP-completa:
$\text{TMSAT} = \{\langle \alpha, x, 1^n, 1^t \rangle \ : \ \exists u \in \{0,1\}^n, \ M_\alpha(\langle x, u \rangle) = 1 \text{ em } t \text{ passos}\}$

Prova:

- (1) $\text{TMSAT} \in \text{NP}$: Mostraremos verificador $V$ para a linguagem:
  - Dada instância $\langle \alpha, x, 1^n, 1^t \rangle$
  - $V(\langle \alpha, x, 1^n, 1^t \rangle, u)$ faz o seguinte:
    - Simula $M_\alpha(x, u)$ por $t$ passos (via MT universal)
    - $V$ Aceita $\Leftrightarrow M(x, u) = 1$
- (2) $\text{TMSAT} \in \text{NP}$-difícil:
  - Seja $L \in \text{NP}$. $\exists p(n)$ e $\exists$ verificador $V_L$ de tempo $q(n)$.
  - Redução: $f(x) = \langle V_L, x, 1^{p(|x|)}, 1^{q(|x|+p|x|)} \rangle$
  - Basta mostrar $\forall x \in \{0,1\}^* \ x \in L \Leftrightarrow f(x) \in \text{TMSAT}$
    - ($\Rightarrow$): Por def., se $\exists u$ e $V_L(x, u) = 1$ então $f(x) \in \text{TMSAT}$
    - ($\Leftarrow$): Por def., se $f(x) = \langle V_L, x, 1^{p(|x|)}, 1^{q(|x|+p|x|)} \rangle \in \text{TMSAT}$ então $\exists u$ e $V_L(x, u) = 1$ □

Pergunta: Esse problema é útil?

# Existem problemas NP-completos?

Um problema menos artificial:

**Teorema 3.6 [Teorema de Cook-Levin]**

O problema SAT é NP-completo

Prova: Daqui algumas aulas