Classe P e NP

Esdras Lins Bispo Jr.
bispojr@ufg.br

Teoria da Computação
Bacharelado em Ciência da Computação

14 de agosto de 2017

UFG
Regional Jataí

Plano de Aula

UFG
Regional Jataí

Sumário

Complexidade de Tempo

Definição 7.7

Seja $t : \mathbb{N} \rightarrow \mathbb{R}^+$ uma função. Defina a **classe de complexidade de tempo**, $\textbf{TIME}(t(n))$, como sendo a coleção de todas as linguagens que são decidíveis por uma máquina de Turing de tempo $O(t(n))$.

Exemplo

- $A = \{0^k 1^k \mid k \geq 0\}$
- $A \in \textbf{TIME}(n^2)$, pois
- M_1 decide A em tempo $O(n^2)$

Complexidade de Tempo

Problema

Seja a linguagem $A = \{0^k 1^k \mid k \geq 0\}$. Quanto tempo uma máquina de Turing simples precisa para decidir A?

Descrição de uma possível MT simples

M_1 = "Sobre a cadeia de entrada ω:

1. Faça uma varredura na fita e *rejeite* se um 0 for encontrado à direita de um 1.
2. Repita se ambos 0s e 1s permanecem sobre a fita:
    1. Faça uma varredura na fita, cortando um único 0 e um único 1.
3. Se 0s ainda permanecerem após todos os 1s tiverem sido cortados, ou se 1s ainda permanecerem após todos os 0s tiverem sido cortados, *rejeite*. Caso contrário, se nem 0s nem 1s permanecerem sobre a fita, *aceite*.

Complexidade de Tempo

Problema

Existe uma máquina que decide assintoticamente a linguagem A mais rapidamente?

Com outras palavras...

$A \in \mathbf{TIME}(t(n))$, para algum $t(n) = o(n^2)$?

Complexidade de Tempo

Descrição de uma outra MT simples

M_2 = "Sobre a cadeia de entrada ω:

1. Faça uma varredura na fita e *rejeite* se um 0 for encontrado à direita de um 1.
2. Repita enquanto alguns 0s e alguns 1s permanecem sobre a fita:
    1. Faça uma varredura na fita, verificando se o número total de 0s e 1s remanescentes é par ou ímpar. Se for ímpar, *rejeite*.
    2. Faça uma varredura novamente na fita,
       cortando alternadamente um 0 não e outro sim
       (começando com o primeiro 0)
       e então cortando alternadamente um 1 não e outro sim
       (começando com o primeiro 1).
3. Se nenhum 0 e nenhum 1 permanecer sobre a fita, *aceite*. Caso contrário, *rejeite*.

Complexidade de Tempo

Problema

Podemos decidir a linguagem A em tempo $O(n)$ (também chamado **tempo linear**)?

Sim... é possível!

Se utilizarmos uma máquina de Turing com duas fitas!

Complexidade de Tempo

Descrição de uma outra MT simples

M_3 = "Sobre a cadeia de entrada ω:

1. Faça uma varredura na fita e *rejeite* se um 0 for encontrado à direita de um 1.
2. Faça uma varredura nos 0s sobre a fita 1 até o primeiro 1. Ao mesmo tempo, copie os 0s para a fita 2.
3. Faça uma varredura nos 1s sobre a fita 1 até o final da entrada. Para cada 1 lido sobre a fita 1, corte um 0 sobre a fita 2. Se todos os 0s estiverem cortados antes que todos os 1s sejam lidos, *rejeite*.
4. Se todos os 0s tiverem agora sido cortados, *aceite*. Se algum 0 permanecer, *rejeite*.

Relacionamentos de Complexidade entre Modelos

Teorema 7.8

Seja $t(n)$ uma função, em que $t(n) \geq n$. Então toda máquina de Turing multifita de tempo $t(n)$ tem uma máquina de Turing de um única fita equivalente de tempo $O(t^2(n))$.

Teorema 7.11

Seja $t(n)$ uma função, em que $t(n) \geq n$. Então toda máquina de Turing não-determinística de uma única fita de tempo $t(n)$ tem uma máquina de Turing de um única fita equivalente de tempo $2^{O(t(n))}$.

A Classe P

Diferenças de complexidade de tempo

- MT simples *x* MT multi-fita:
  potência quadrática (ou *polinomial*)
- MT simples *x* MT não-determinística:
  no máximo *exponencial*.

A Classe P

Diferenças entre as taxas de crescimento

Exemplo: n^3 e 2^n

- Admita $n = 1000$;
- Logo, $n^3 = 1$ bilhão;
- Mas, 2^n é maior que o número de átomos do universo.

Sumário

A Classe P

Definição 7.12

P é a classe de linguagens que são decidíveis em tempo polinomial sobre uma máquina de Turing determinística de uma única fita.

Em outras palavras

$$\mathbf{P} = \bigcup_k \mathbf{TIME}\left(n^k\right).$$

A Classe P

Definição 7.12

P é a classe de linguagens que são decidíveis em tempo polinomial sobre uma máquina de Turing determinística de uma única fita.

Em outras palavras

$$\mathbf{P} = \bigcup_k \mathbf{TIME}\,(n^k).$$

P é importante porque...

A Classe P

Definição 7.12

P é a classe de linguagens que são decidíveis em tempo polinomial sobre uma máquina de Turing determinística de uma única fita. Em outras palavras

$$\mathbf{P} = \bigcup_k \mathbf{TIME}\ (n^k).$$

P é importante porque...

- **P** é invariante para todos os modelos de computação polinomialmente equivalentes à máquina de Turing determinística de uma única fita;

A Classe P

Definição 7.12

P é a classe de linguagens que são decidíveis em tempo polinomial sobre uma máquina de Turing determinística de uma única fita. Em outras palavras

$$\mathbf{P} = \bigcup_k \mathbf{TIME}\,(n^k).$$

P é importante porque...

- **P** é invariante para todos os modelos de computação polinomialmente equivalentes à máquina de Turing determinística de uma única fita;
- **P** corresponde aproximadamente à classe de problemas que são realisticamente solúveis em um computador.

A Classe P

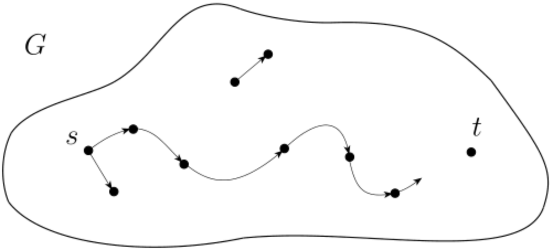
Problema do caminho em um grafo

$CAM = \{\langle G, s, t\rangle \mid G$ é um grafo direcionado que tem um caminho direcionado de s para $t\}$.

A Classe P

Problema do caminho em um grafo

$CAM = \{\langle G, s, t\rangle \mid G$ é um grafo direcionado que tem um caminho direcionado de s para $t\}$.



A Classe P

Teorema 7.14

$CAM \in \mathbf{P}$

A Classe P

Teorema 7.14

$CAM \in \mathbf{P}$

Prova

$M =$ "Sobre a cadeia de entrada $\langle G, s, t \rangle$ em que G é um grafo direcionado com nós s e t:

1. Ponha uma marca sobre o nó s.
2. Repita o seguinte até que nenhum nó adicional seja marcado:
   1. Faça uma varredura em todas as arestas de G. Se uma aresta (a, b) for encontrada indo de um nó marcado a para um nó não marcado b, marque o nó b.
3. Se t estiver marcado, *aceite*. Caso contrário, *rejeite*.

A Classe P

Problema dos números primos entre si

PRIM-ES $= \{\langle x, y\rangle \mid x$ e y são primos entre si $\}$.

A Classe P

Problema dos números primos entre si

PRIM-ES $= \{\langle x, y \rangle \mid x$ e y são primos entre si $\}$.

Teorema 7.15

PRIM-ES $\in$ **P**

A Classe P

Prova (Parte 1): Algoritmo de Euclides (E)

E = "Sobre a cadeia de entrada $\langle x, y \rangle$ em que x e y são números naturais em binário:

1. Repita até que $y = 0$:
   1. Atribua $x \leftarrow x \bmod y$.
   2. Intercambie x e y.
2. Dê como saída x".

A Classe P

Prova (Parte 1): Algoritmo de Euclides (E)

E = "Sobre a cadeia de entrada $\langle x, y \rangle$ em que x e y são números naturais em binário:

1. Repita até que $y = 0$:
   1. Atribua $x \leftarrow x \bmod y$.
   2. Intercambie x e y.
2. Dê como saída x".

Prova (Parte 2): Máquina de Turing que decide PRIM-ES

R = "Sobre a cadeia de entrada $\langle x, y \rangle$ em que x e y são números naturais em binário:

1. Rode E sobre $\langle x, y \rangle$.
2. Se o resultado for 1, *aceite*. Caso contrário, *rejeite*".

Sumário

1. Revisão
   - Classe P
   - Complexidade entre Modelos
2. Classe P
3. Classe NP

Classe NP

Questão

- Tentativas de evitar a força bruta em alguns problemas não têm sido bem sucedidas.

Classe NP

Questão

- Tentativas de evitar a força bruta em alguns problemas não têm sido bem sucedidas.
- Não se sabe se existem algoritmos de tempo polinomial que resolvem determinados problemas.

Classe NP

Questão

- Tentativas de evitar a força bruta em alguns problemas não têm sido bem sucedidas.
- Não se sabe se existem algoritmos de tempo polinomial que resolvem determinados problemas.

Possíveis soluções...

- Os algoritmos polinomiais para estes problemas utilizam técnicas desconhecidas; *ou*

Classe NP

Questão

- Tentativas de evitar a força bruta em alguns problemas não têm sido bem sucedidas.
- Não se sabe se existem algoritmos de tempo polinomial que resolvem determinados problemas.

Possíveis soluções...

- Os algoritmos polinomiais para estes problemas utilizam técnicas desconhecidas; *ou*
- Os algoritmos polinomiais para estes problemas simplesmente **não existem**.

UFG Regional Jataí

Classe NP

Questão

- Tentativas de evitar a força bruta em alguns problemas não têm sido bem sucedidas.
- Não se sabe se existem algoritmos de tempo polinomial que resolvem determinados problemas.

Possíveis soluções...

- Os algoritmos polinomiais para estes problemas utilizam técnicas desconhecidas; *ou*
- Os algoritmos polinomiais para estes problemas simplesmente **não existem**.

Curioso...

Existe um grupo de problemas deste tipo que, existindo um algoritmo polinomial que resolve um destes problemas, é possível resolver todos os problemas do grupo.

A Classe NP

Problema do **caminho hamiltoniano** em um grafo

$CAMHAM = \{\langle G, s, t\rangle \mid G$ é um grafo direcionado com um caminho hamiltoniano de s para $t\}$.

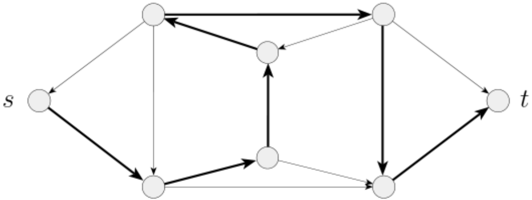


FIGURA 7.17
Um caminho hamiltoniano passa por todo nó exatamente uma vez

Classe NP

Característica importante

O problema *CAMHAM* tem **verificabilidade polinomial**

Classe NP

Característica importante

O problema *CAMHAM* tem **verificabilidade polinomial**

Outro problema polinomialmente verificável...

$COMPOSTOS = \{x \mid x = pq, \text{ para inteiros } p, q > 1\}$

Classe NP

Característica importante

O problema *CAMHAM* tem **verificabilidade polinomial**

Outro problema polinomialmente verificável...

$COMPOSTOS = \{x \mid x = pq, \text{ para inteiros } p, q > 1\}$

Exemplo

$33 \in COMPOSTOS \therefore$

Classe NP

Característica importante

O problema *CAMHAM* tem **verificabilidade polinomial**

Outro problema polinomialmente verificável...

$COMPOSTOS = \{x \mid x = pq, \text{ para inteiros } p, q > 1\}$

Exemplo

$33 \in COMPOSTOS \therefore$

- $3 \times 11 = 33$
- $3, 11 \in \mathbb{Z}$

UFG
Regional Jataí

Classe NP

Característica importante

O problema *CAMHAM* tem **verificabilidade polinomial**

Outro problema polinomialmente verificável...

$COMPOSTOS = \{x \mid x = pq, \text{ para inteiros } p, q > 1\}$

Exemplo

$33 \in COMPOSTOS \therefore$

- $3 \times 11 = 33$
- $3, 11 \in \mathbb{Z}$

Porém...

Existem problemas que não podem ser verificados em tempo polinomial. Exemplo: $\overline{CAMHAM}$.

Classe NP

Definição 7.18

Um **verificador** para uma linguagem A é um algoritmo V, em que

$$A = \{\omega \mid V \text{ aceita } \langle \omega, c \rangle \text{ para alguma cadeia } c\}.$$

Classe NP

Definição 7.18

Um **verificador** para uma linguagem A é um algoritmo V, em que

$$A = \{\omega \mid V \text{ aceita } \langle \omega, c \rangle \text{ para alguma cadeia } c\}.$$

Detalhes

Medimos o tempo de um verificador somente em termos do comprimento de ω, portanto um **verificador de tempo polinomial** roda em tempo polinomial no comprimento de ω.

UFG
Regional Jataí

Classe NP

Definição 7.18

Um **verificador** para uma linguagem A é um algoritmo V, em que

$$A = \{\omega \mid V \text{ aceita } \langle \omega, c \rangle \text{ para alguma cadeia } c\}.$$

Detalhes

Medimos o tempo de um verificador somente em termos do comprimento de ω, portanto um **verificador de tempo polinomial** roda em tempo polinomial no comprimento de ω.

Nomenclaturas...

Uma linguagem A é **polinomialmente verificável** se ela tem um verificador de tempo polinomial.

Classe NP

Certificado (Prova)

- A informação adicional, representada por c, utilizada por um verificador é chamada de **certificado** (ou prova) da pertinência a uma dada linguagem.

UFG
Regional Jataí

Classe NP

Certificado (Prova)

- A informação adicional, representada por c, utilizada por um verificador é chamada de **certificado** (ou prova) da pertinência a uma dada linguagem.
- Para verificadores polinomiais, o certificado tem comprimento polinomial (no comprimento de ω.

Classe NP

Certificado (Prova)

- A informação adicional, representada por c, utilizada por um verificador é chamada de **certificado** (ou prova) da pertinência a uma dada linguagem.
- Para verificadores polinomiais, o certificado tem comprimento polinomial (no comprimento de ω.

Exemplo

- Um certificado para uma cadeia $\langle G, s, t \rangle \in CAMHAM$ é um caminho hamiltoniano de s a t.

Classe NP

Certificado (Prova)

- A informação adicional, representada por c, utilizada por um verificador é chamada de **certificado** (ou prova) da pertinência a uma dada linguagem.
- Para verificadores polinomiais, o certificado tem comprimento polinomial (no comprimento de ω.

Exemplo

- Um certificado para uma cadeia $\langle G, s, t\rangle \in CAMHAM$ é um caminho hamiltoniano de s a t.
- Um certificado para um número composto $x \in COMPOSTOS$ é um dos seus divisores.

Classe NP

Definição 7.19

NP é a classe das linguagens que têm verificadores de tempo polinomial.

Classe NP

Definição 7.19

NP é a classe das linguagens que têm verificadores de tempo polinomial.

Teorema 7.20

Uma linguagem está em **NP** sse ela é decidida por alguma máquina de Turing não-determinística de tempo polinomial.

Classe NP

Definição 7.19

NP é a classe das linguagens que têm verificadores de tempo polinomial.

Teorema 7.20

Uma linguagem está em **NP** sse ela é decidida por alguma máquina de Turing não-determinística de tempo polinomial.

Definição 7.21

$\mathbf{NTIME(t(n))} = \{L \mid L$ é uma linguagem decidida por uma MT não-determinística de tempo $O(t(n))\}$.

Classe NP

Definição 7.19

NP é a classe das linguagens que têm verificadores de tempo polinomial.

Teorema 7.20

Uma linguagem está em **NP** sse ela é decidida por alguma máquina de Turing não-determinística de tempo polinomial.

Definição 7.21

$\textbf{NTIME(t(n))} = \{L \mid L$ é uma linguagem decidida por uma MT não-determinística de tempo $O(t(n))\}$.

Corolário

$\textbf{NP} = \bigcup_k \textbf{NTIME}(n^k)$

Classe NP

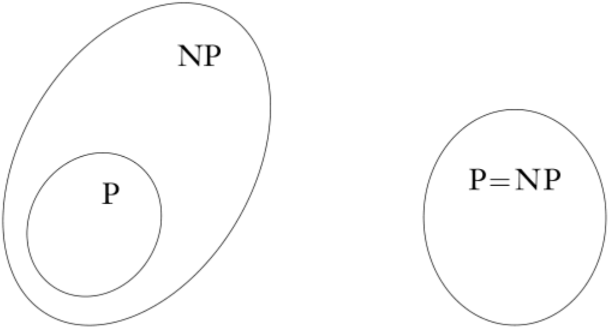


FIGURA 7.26
Uma dessas possibilidades é correta

Classe P e NP

Esdras Lins Bispo Jr.
bispojr@ufg.br

Teoria da Computação
Bacharelado em Ciência da Computação

14 de agosto de 2017

UFG
Regional Jataí