Computational Complexity

Lecture 7: the Polynomial Hierarchy

Ronald de Haan
me@ronalddehaan.eu

University of Amsterdam

February 22, 2021
Recap

What we saw last time..

- Space-bounded computation
- Limits on memory space
- L, NL, PSPACE
- Logspace reductions
- NL-completeness
What will we do today?

- The Polynomial Hierarchy
- Bounded quantifier alternation
- Alternating Turing machines
We saw that 3COL is NP-complete, but how about the following problem?

3COL-Extension = \{ (G, V_0) \mid G = (V, E) is an undirected graph, V_0 \subseteq V, and each 3-coloring of the vertices in V_0 can be extended to a proper 3-coloring of the entire graph G \}

There seems to be no single (polynomial-size) certificate for yes-inputs

It is a "\(\forall \exists\)-type" question

We need a different complexity class to capture the complexity of 3COL-Extension
The complexity class $\Sigma_2^p$

**Definition (NP)**

A language $L \subseteq \{0, 1\}^*$ is in the class NP if there is a polynomial $q : \mathbb{N} \to \mathbb{N}$ and a polynomial-time Turing machine $M$ such that for every $x \in \{0, 1\}^*$:

$$x \in L \text{ if and only if } \text{there exists some } u \in \{0, 1\}^{q(|x|)} \text{ such that } M(x, u) = 1.$$ 

**Definition (coNP)**

A language $L \subseteq \{0, 1\}^*$ is in the class coNP if there is a polynomial $q : \mathbb{N} \to \mathbb{N}$ and a polynomial-time Turing machine $M$ such that for every $x \in \{0, 1\}^*$:

$$x \in L \text{ if and only if } \text{for all } u \in \{0, 1\}^{q(|x|)} \text{ it holds that } M(x, u) = 1.$$
The complexity class $\Sigma^p_2$

Definition ($\Sigma^p_2$)

A language $L \subseteq \{0, 1\}^*$ is in the class $\Sigma^p_2$ if there is a polynomial $q : \mathbb{N} \to \mathbb{N}$ and a polynomial-time Turing machine $M$ such that for every $x \in \{0, 1\}^*$:

$$x \in L \text{ if and only if } \exists u_1 \in \{0, 1\}^{q(|x|)} \text{ such that for all } u_2 \in \{0, 1\}^{q(|x|)} \text{ it holds that } M(x, u_1, u_2) = 1.$$
The complexity class $\Pi^p_2$

**Definition ($\Pi^p_2$)**

A language $L \subseteq \{0, 1\}^*$ is in the class $\Sigma^p_2$ if there is a polynomial $q : \mathbb{N} \to \mathbb{N}$ and a polynomial-time Turing machine $M$ such that for every $x \in \{0, 1\}^*$:

$$x \in L \text{ if and only if } \forall u_1 \in \{0, 1\}^{q(|x|)} \exists u_2 \in \{0, 1\}^{q(|x|)} \text{ such that } M(x, u_1, u_2) = 1.$$  

It turns out that 3COL-Extension is $\Pi^p_2$-complete.
The complexity classes $\Sigma^p_i$

**Definition ($\Sigma^p_i$)**

Let $i \geq 1$. A language $L \subseteq \{0, 1\}^*$ is in the class $\Sigma^p_i$ if there is a polynomial $q : \mathbb{N} \to \mathbb{N}$ and a polynomial-time Turing machine $M$ such that for every $x \in \{0, 1\}^*$:

$$x \in L \text{ if and only if } \exists u_1 \in \{0, 1\}^{|x|} q(|x|) \text{ such that }$$

$$\text{for all } u_2 \in \{0, 1\}^{|x|} q(|x|)$$

$$\vdots$$

$$\text{for all } u_i \in \{0, 1\}^{|x|} q(|x|)$$

$$\text{it holds that } M(x, u_1, \ldots, u_i) = 1. \text{ if } i \text{ is even},$$

$$\vdots$$

$$\text{there exists } u_i \in \{0, 1\}^{|x|} q(|x|)$$

$$\text{such that } M(x, u_1, \ldots, u_i) = 1. \text{ if } i \text{ is odd}.$$
The complexity classes $\Pi^p_i$

Definition ($\Pi^p_i$)

Let $i \geq 1$. A language $L \subseteq \{0,1\}^*$ is in the class $\Pi^p_i$ if there is a polynomial $q : \mathbb{N} \rightarrow \mathbb{N}$ and a polynomial-time Turing machine $M$ such that for every $x \in \{0,1\}^*$:

$x \in L$ if and only if

for all $u_1 \in \{0,1\}^{q(|x|)}$

there exists $u_2 \in \{0,1\}^{q(|x|)}$ such that

$\vdots$

for all $u_i \in \{0,1\}^{q(|x|)}$

it holds that $M(x, u_1, \ldots, u_i) = 1$. if $i$ is odd,

$\vdots$

there exists $u_i \in \{0,1\}^{q(|x|)}$

such that $M(x, u_1, \ldots, u_i) = 1$. if $i$ is even.
The Polynomial Hierarchy (PH)

**Definition (Σ₀, Π₀, PH)**

\[ Σ₀ = Π₀ = P \]

\[ PH = \bigcup_{i \geq 0} Σ_i^p. \]

- Some relations:
  - \( Π_i^p = \{ \overline{L} | L \in Σ_i^p \} \)
  - \( Σ_1^p = NP, Π_1^p = coNP \)
  - \( Σ_i^p \subseteq Π_{i+1}^p \subseteq Σ_{i+2}^p, \)
    \( \Pi_i^p \subseteq \Sigma_{i+1}^p \subseteq \Pi_{i+2}^p \)
  - \( Σ_i^p \subseteq Σ_{i+1}^p, Π_i^p \subseteq Π_{i+1}^p \)
  - \( Σ_i^p \cup Π_i^p \subseteq \text{PSPACE} \)
  - \( PH \subseteq \text{PSPACE} \)
“Collapse” of the hierarchy

- Statements like “P ≠ NP” and “NP ≠ coNP” are widely believed conjectures
- We can use these as assumptions to show some results
  - E.g., assuming that P ≠ NP, NP-complete problems are not in P.
- For some results, stronger conjectures seem necessary
- Another conjecture: “the PH does not collapse”
  - “the PH collapses to P” \( \text{PH} = \text{P} \)
  - “the PH collapses to the \( i \)th level” \( \text{PH} = \Sigma^p_i \)

**Theorem**

Let \( i \geq 1 \). If \( \Sigma^p_i = \Pi^p_i \), then \( \text{PH} = \Sigma^p_i \).

If \( P = \text{NP} \), then \( \text{PH} = \text{P} \).
QBF problems complete for $\Sigma^p_i$ and $\Pi^p_i$

- $\Sigma_i\text{SAT} = \{ \varphi = \exists u_1 \forall u_2 \ldots Q_i u_i \psi(u_1, \ldots, u_i) : \varphi \text{ is a true QBF} \},$
  where each $u_j = (u_{j,1}, \ldots, u_{j,\ell})$ is a sequence of propositional variables,
  $\exists u_j$ stands for $\exists u_{j,1} \exists u_{j,2} \ldots \exists u_{j,\ell}$, and $\forall u_j$ for $\forall u_{j,1} \forall u_{j,2} \ldots \forall u_{j,\ell}$

- $\Pi_i\text{SAT} = \{ \varphi = \forall u_1 \exists u_2 \ldots Q_i u_i \psi(u_1, \ldots, u_i) : \varphi \text{ is a true QBF} \},$

**Theorem**

Let $i \geq 1$. Then $\Sigma_i\text{SAT}$ is $\Sigma^p_i$-complete and $\Pi_i\text{SAT}$ is $\Pi^p_i$-complete (both under polynomial-time reductions).
Oracle characterizations of $\Sigma^p_i$ and $\Pi^p_i$

**Theorem**

Let $i \geq 2$. Then $\Sigma^p_i = \text{NP}^{\Sigma^p_{i-1}\text{SAT}}$ and $\Pi^p_i = \text{coNP}^{\Sigma^p_{i-1}\text{SAT}}$.

- (Or replace $\Sigma^p_{i-1}\text{SAT}$ by any $\Sigma^p_{i-1}$-complete or $\Pi^p_{i-1}$-complete problem.)

- This is often written as: $\Sigma^p_i = \text{NP}^{\Sigma^p_{i-1}}$ and $\Pi^p_i = \text{coNP}^{\Sigma^p_{i-1}}$
Configurations $C$ consist of:
(1) tape contents
(2) tape head positions
(3) state $q \in Q$

Configuration graph of a TM $M$ on some input $x$:

- Nodes are all the configurations that are reachable from the initial configuration $C_0$
- Edge from $C$ to $C'$ if applying one of the transition functions in $C$ results in $C'$
Alternating Turing machines

Definition (Alternating Turing machines; ATMs)

- Instead of a single transition function $\delta$, there are two transition functions $\delta_1, \delta_2$.
- The set $Q \setminus \{q_{\text{acc}}, q_{\text{rej}}\}$ is partitioned into $Q_\exists$ and $Q_\forall$.
- Executions of alternating TMs are defined using a labeling procedure on the configuration graph. Repeatedly apply, until a fixpoint is reached:
  - Label each configuration with $q_{\text{acc}}$ with “accept.”
  - If a configuration $c$ with $q \in Q_\exists$ has an edge to a configuration $c'$ that is labeled with “accept,” then label $c$ with “accept.”
  - If a configuration $c$ has a state $q \in Q_\forall$ and both configurations $c', c''$ that are reachable from it in the graph are labeled with “accept,” then label $c$ with “accept.”
- The TM accepts the input if the starting configuration is labeled with “accept.”
- The TM runs in time $T(n)$ if for every input $x$ and for every possible sequence of transition function choices, the machine halts after at most $T(|x|)$ steps.
Alternating Turing machines (ct’d)

$C_0 \exists \leadsto$ so the ATM accepts the input

- $\circ = \text{reject}$
- $\bullet = \text{accept}$
**ATIME, $\Sigma_i$TIME, and $\Pi_i$TIME**

**Definition (ATIME)**

Let $T : \mathbb{N} \to \mathbb{N}$ be a function. A decision problem $L \subseteq \{0, 1\}^*$ is in $\text{ATIME}(T(n))$ if there exists an ATM that decides $L$ and that runs in time $O(T(n))$.

**Definition ($\Sigma_i$TIME)**

Let $T : \mathbb{N} \to \mathbb{N}$ be a function. A decision problem $L \subseteq \{0, 1\}^*$ is in $\Sigma_i\text{TIME}(T(n))$ if there exists an ATM that decides $L$, that runs in time $O(T(n))$, whose initial state is in $Q_\exists$, and that on every input and on every path in the configuration graph alternates at most $i - 1$ times between $Q_\exists$ and $Q_\forall$.

$\Pi_i\text{TIME}$ is defined similarly to $\Sigma_i\text{TIME}$, with the difference that the initial state of the ATM is in $Q_\forall$. 
Theorem

\[ \text{PSPACE} = \bigcup_{c \geq 0} \text{ATIME}(n^c). \]

Theorem

Let \( i \geq 1 \). Then:

\[ \Sigma_i^P = \bigcup_{c \geq 0} \Sigma_i \text{TIME}(n^c) \]
\[ \Pi_i^P = \bigcup_{c \geq 0} \Pi_i \text{TIME}(n^c). \]
An overview of complexity classes

- $L \subseteq NL \subseteq P \subseteq coNP \subseteq \Sigma_2^p \subseteq \Pi_2^p \subseteq \Sigma_3^p \subseteq \Pi_3^p \subseteq PH \subseteq PSPACE \subseteq EXP$

- $P \cap coNP \subseteq \Sigma_2^p \cap \Pi_2^p \subseteq \Sigma_3^p \cap \Pi_3^p \subseteq PH$

- $P \cap coNP \cap \Sigma_2^p \cap \Pi_2^p \cap \Sigma_3^p \cap \Pi_3^p \subseteq PH \cap PSPACE \cap EXP$

- $PH \cap PSPACE \cap EXP$
The classes $\Sigma^p_i$ and $\Pi^p_i$.

The Polynomial Hierarchy.

$\Sigma^p_i$-complete and $\Pi^p_i$-complete QBF problems.

Characterizations using oracles and ATMs.
Next time

- A “breather”
- Time to reflect on what we’ve done so far
- Requests for things to recap?