Lesson 14: NLOG- and PSPACE-complete problems

Theme: NLOG- and PSPACE-complete problems.

1 NLOG-complete problems

Recall from Lesson 12 that a language $L$ is in $\text{Log}$, if there is a constant $c \geq 0$, a 2-tape DTM $M$ that decides $L$ such that on input word $w$:

- The first tape always contains only the input word $w$, i.e., $M$ never changes the content of the first tape.
- $M$ uses/visits $\leq c \cdot \log(|w|)$ cells of its second tape.

Likewise, we say that a language $L$ is in $\text{NLog}$, if there is a 2-tape NTM $M$ that decides $L$ such that the above two conditions are satisfied.

A DTM $M$ computes a function $F : \Sigma^* \to \Sigma^*$ in log-space, if $M$ is a 3-tape DTM that works as follows.

- The first tape always contains only the input word $w$, i.e., $M$ never changes the content of the first tape.
- $M$ uses/visits $\leq c \cdot \log(|w|)$ cells of its second tape.
- When $M$ enters the accepting state, the content of its third tape is $F(w)$.

A function is log-space computable, if there is a DTM that computes it in log-space.

**Definition 14.1** We say that a language $L_1$ is log-space reducible to another language $L_2$, denoted by $L_1 \leq_{\text{log}} L_2$, if there is a log-space computable function $F : \Sigma^* \to \Sigma^*$ such that for every $w \in \Sigma^*$, the following holds.

$$w \in L_1 \text{ if and only if } F(w) \in L_2.$$

**Definition 14.2** Let $L$ be a language.

- $L$ is NLOG-hard, if for every $L' \in \text{NLog}$, $L' \leq_{\text{log}} L$.
- $L$ is NLOG-complete, if $L \in \text{NLog}$ and $L$ is NLOG-hard.

Let $G$ be a (directed) graph and $s, t$ are two vertices. We denote by $\lfloor (G, s, t) \rfloor$ the string representation of $(G, s, t)$ Define the following language:

$$\text{REACH} := \{ \lfloor (G, s, t) \rfloor \mid \text{there is a path from } s \text{ to } t \text{ in } G \}$$

**Theorem 14.3** $\text{REACH}$ is NLOG-complete.

We can define the class $\text{coNLog} := \{ L \mid \Sigma^* - L \in \text{NLog} \}$.

**Theorem 14.4** $\text{NLog} = \text{coNLog}$. 

1


2 PSPACE-complete problems

Definition 14.5 Let $L$ be a language.

- $L$ is PSPACE-hard, if for every $L' \in \text{PSPACE}$, $L' \leq_p L$.
- $L$ is PSPACE-complete, if $L \in \text{PSPACE}$ and $L$ is PSPACE-hard.

Quantified Boolean formulas (QBF) are formulas of the form:

$$Q_1 x_1 \ Q_2 x_2 \ \cdots \ Q_n x_n \ \varphi(x_1, \ldots, x_n),$$

where each $Q_i \in \{\forall, \exists\}$ and $\varphi(x_1, \ldots, x_n)$ are Boolean formulas with variables $x_1, \ldots, x_n$.

The intuitive meaning of each $Q_i$ is as follows.

- $\forall x \ \psi$ means that for all $x \in \{\text{True}, \text{False}\}$, $\psi$ is true.
- $\exists x \ \psi$ means that there is $x \in \{\text{True}, \text{False}\}$ such that $\psi$ is true.

<table>
<thead>
<tr>
<th>TQBF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input: A QBF $\varphi$.</td>
</tr>
<tr>
<td>Task: Output True, if $\varphi$ is true. Otherwise, output False.</td>
</tr>
</tbody>
</table>

Note that the usual Boolean formula can be viewed as a QBF, where each $Q_i$ is $\exists$. Thus, TQBF is a more general problem than SAT.

Theorem 14.6 TQBF is PSPACE-complete.

Theorem 14.7 PSPACE = NPSPACE = coNPSPACE.