Tutorial 05 – PSPACE and time hierarchy

Exercise 1. NL = coNL

Let coNL be the class of languages whose complements belong to NL. We want to show that NL = coNL (Immerman–Szelepcsényi theorem).

1. Show that coPath $= \{(G, s, t)$: there is no path from s to t in $G\}$ is coNL-complete (by logspace reductions).

Given a graph G and its vertex s, we define S_i as the set of vertices of G that are accessible from s in at most i steps.

2. Propose an NDTM M_0 in NL that takes on input (G, s, u, i) and accepts iff $u \in S_i$.
3. Propose a NDTM M_1 in NL that has at least one accepting path on input $(G, s, u, i, |S_i|)$ and, when M_1 accepts this input, then it outputs 1 if $u \in S_i$ and 0 otherwise. (We suppose that i and $|S_i|$ are given in binary.)
4. Propose a NDTM M_2 in NL that has at least one accepting path on input $(G, s, i, |S_i|)$ and, when M_2 accepts this input, then $|S_{i+1}|$ is written on the output tape.
5. Construct a NDTM M in NL with input (G, s) such that $M(G, s)$ has at least one accepting path and, when if $M(G, s)$ accepts, the number of vertices accessible from s is written in the output tape.
6. Prove that coPath $\in$ NL.
7. Deduce that NL = coNL.

Exercise 2. Generalized Geography

Geography is a two player word game where the first player start by naming a town, say *Lyon*, then the second player name a town starting with the last letter of the previous named town, say *Nantes*, and so on without repeating any town. The first player that cannot name further cities loses the game.

1. How would you formalize this game in terms of a decision problem with input $\langle G, s\rangle$, where G is a graph and s is a vertex of G?
2. How can you explain TQBF as a two-player game?
3. Generalized geography (Geo) is a generalization of Geography to arbitrary graphs. Show that Geo is PSPACE-complete.
4. Show that Geo is PSPACE-complete even if we know that the underlying graph G is planar and every vertex has total degree bounded by 3.

Exercise 3. *Deterministic time hierarchy*

In this exercise we consider the *time hierarchy theorem* that states that allowing Turing machines more computation time strictly increases the set of languages that they can decide.

Theorem. *If f, g functions satisfying $f(n)\log f(n) = o(g(n))$ and g is time-constructible, then* $\mathrm{DTIME}(f(n)) \subsetneq \mathrm{DTIME}(g(n))$.

Let $\mathcal{U}$ be the universal Turing machine constructed on TD1 (i.e., a machine that halts in $C_\alpha T \log T$ steps if M_α halts in T steps.) Consider the language

$$\mathcal{L} := \{(\alpha, x) \colon \mathcal{U}\big(\alpha, (\alpha, x)\big) \text{ rejects in at most } g(|(\alpha, x)|) \text{ steps}\}.$$

1. Show that $\mathcal{L} \in \mathrm{DTIME}(g(n))$.
2. Show that if M_α is a machine that runs in time $cf(n)$ for some $c > 0$, then the language recognized by M_α is not equal to $\mathcal{L}$, i.e., $\mathcal{L} \notin \mathrm{DTIME}(f(n))$.

Exercise 4. *Little space is no space at all!*

We are going to show that a language that can be recognized in $o(\log\log n)$ space can in fact be recognized in $O(1)$ space. This implies, e.g., that $\mathsf{DSPACE}(\sqrt{\log\log n}) = \mathsf{DSPACE}(1)$.

1. Let $\mathcal{L} \notin \mathsf{DSPACE}(1)$ and suppose that M is a Turing machine that recognizes $\mathcal{L}$ in space $o(\log\log n)$. For every $k \in \mathbb{N}$ show that there exists $x \in \mathcal{L}$ such that $M(x)$ uses at least k cells on the work tape during the computation.
2. At every step of the computation of $M(x)$ we define the current *internal configuration* of M as the tuple (q, y), where q is the current state of M and y encodes the current contents of the work tapes of M and the positions of the work heads. Show that M has $o(\log|x|)$ different internal configurations during the computation of $M(x)$.
3. We define the ith *crossing sequence* $\mathcal{C}_i(x)$ of a M on an input x, $1 \leq i \leq |x| - 1$, as a vector $(c_1, c_2, \ldots, c_m)$, where c_1 is the internal configuration of M when the input head first crossed from a cell i to a cell $i+1$ on the input tape, c_2 is the internal configuration of M when the input head first crossed from a cell $i+1$ to i, etc. (so that the odd elements denote configurations of M after crossing ith cell left-to-right, and the even ones – after crossing right-to-left from $(i+1)$th cell to ith). Show that there are $o(|x|)$ different crossing-sequences that appear on an input x.
4. Show that the crossing sequences of M on x_k are pairwise different. Conclude.