Computational Complexity

Homework Sheet 3

Hand in via Canvas before March 1, 2021, at 13:00

Exercise 1 (2pt)**.** Prove that $\mathsf{coNP} \subseteq \mathsf{PSPACE}$.

Exercise 2 (2pt)**.** Prove that $\mathsf{NL} \subseteq \mathsf{P}$.

- *Hint:* consider the configuration graph of the nondeterministic Turing machine—see Section 4.1.1. in the book [1].

Definition 1. Remember the Slitherlink puzzle from the first homework assignment, and the notion of *Slitherlink instances* and *solutions*. Let SLITH-CHECK be the decision problem where the input consists of a Slitherlink instance I and a candidate solution S for I, and the question is to decide whether S is a valid solution for I.

For the sake of concreteness, suppose that Slitherlink instances I are described a list of triples (i, j, v), each denoting that the cell in the i-th column and the j-th row contains the value $v \in \{\square, 0, 1, 2, 3\}$—and this list contains a triple for each cell in the grid. Suppose, moreover, that solutions S are described by a list of triples (i, j, E), where $E \subseteq \{\mathrm{L}, \mathrm{R}, \mathrm{T}, \mathrm{B}\}$, each denoting that the cell in the i-th column and the j-th row is next to an edge on its left if $\mathrm{L} \in E$, an edge on its right if $\mathrm{R} \in E$, an edge on its top side if $\mathrm{T} \in E$, an edge on its bottom side if $\mathrm{B} \in E$.

For example, the Slitherlink instance I_1 in Figure 1a is represented by:

$$[(1,1,2),(2,1,\square),(3,1,2),(1,2,\square),(2,2,0),(3,2,\square),(1,3,\square),(2,3,2),(3,3,2)],$$

and the solution S_1 in Figure 1b is represented by:

$$[(1,1,\{\mathrm{L},\mathrm{T}\}),(2,1,\{\mathrm{R},\mathrm{T}\}),(3,1,\{\mathrm{B},\mathrm{L}\}),(1,2,\{\mathrm{B},\mathrm{L}\}),(2,2,\emptyset),(3,2,\{\mathrm{R},\mathrm{T}\}),(1,3,\{\mathrm{R},\mathrm{T}\}),(2,3,\{\mathrm{B},\mathrm{L}\}),(3,3,\{\mathrm{B},\mathrm{R}\})].$$

(a) A Slitherlink instance I_1. (b) A solution S_1 for I_1.

Figure 1: An example of a Slitherlink instance and a solution.

Exercise 3 (3pt)**.** Show that SLITH-CHECK is in L.

- Describe how a logarithmic-space algorithm for SLITH-CHECK would work. Clearly describe **(i)** what properties the algorithm checks, **(ii)** why these properties characterize valid solutions, **(iii)** what the algorithm keeps in memory, **(iv)** why the memory usage is logarithmic in the size of the input, and **(v)** how the algorithms checks the properties you described using only the input and the limited memory space.

Definition 2. Define the complexity class DP as follows:

$$\mathsf{DP} = \{ A \cap B \mid A \in \mathsf{NP}, B \in \mathsf{coNP} \}.$$

Definition 3. Let $G = (V, E)$ be an undirected graph. A subset $C \subseteq V$ of vertices is called a *clique* of G if every $v_1, v_2 \in C$ with $v_1 \neq v_2$ are connected by an edge in E.

Consider the decision problems Clique and Exact-Clique:

$$\mathsf{Clique} = \{ (G, k) \mid G \text{ is an undirected graph that has a clique of size } k \}.$$

$$\mathsf{Exact\text{-}Clique} = \{ (G, k) \mid G \text{ is an undirected graph that has a clique of size } k \text{ but has no clique of size } k+1 \}.$$

Exercise 4 (3pt)**.**

(a) Prove that if $\mathsf{DP} \subseteq \mathsf{NP}$, then the Polynomial Hierarchy collapses.

(b) Prove that Exact-Clique is DP-complete (under polynomial-time reductions).

- *Hint:* You may use the following. Let L be an arbitrary problem in NP. Then there exists a polynomial-time reduction R from L to Clique such that for every $x \in \{0,1\}^*$ the instance $(G, k) = R(x)$ has the following additional properties:[1]
  - G has no clique of size $k + 1$.
  - If G has no clique of size k, then it also has no clique of size $k - 1$.

Remark 1. Answers will be graded on two criteria: they should (1) be correct and intelligent, and also (2) concise and to the point.

Remark 2. If you find a solution to one of the exercises in a paper or book, you can use this to inform your solution. Make sure that you write down the solution in your own words, conveying that you understand what is going on.

References

[1] Sanjeev Arora and Boaz Barak. *Computational Complexity – A Modern Approach.* Cambridge University Press, 2009.

[1]Note that these properties are only guaranteed for the value of k given by the reduction—i.e., for the value of k in $R(x) = (G, k)$.