Theory of Computer Science

G. Röger
F. Pommerening
Spring Term 2020

University of Basel
Computer Science

Exercise Sheet 12 — Solutions

Exercise 12.1 (Polynomial Reduction; 3 marks)

Consider the following decision problems:

HittingSet:

- *Given:* finite set M, set of sets $\mathcal{S} = \{S_1, \dots, S_n\}$ with $S_i \subseteq M$ for all $i \in \{1, \dots, n\}$, natural number $k \in \mathbb{N}_0$
- *Question:* Is there a set H with at most k elements, which contains at least one element from each set in $\mathcal{S}$.

  Formally: Is there a set H with $|H| \leq k$ and $H \cap S_i \neq \emptyset$ for all $i \in \{1, \dots, n\}$?

VertexCover:

- *Given:* undirected graph $G = \langle V, E \rangle$, natural number $k \in \mathbb{N}_0$
- *Question:* Does G have a vertex cover of size at most k, i.e., a set of vertices $C \subseteq V$ with $|C| \leq k$ and $\{u, v\} \cap C \neq \emptyset$ for all $\{u, v\} \in E$?

Prove that HittingSet is NP-hard. You may use that VertexCover is NP-complete.

Solution:

We have to show VertexCover $\leq_\mathrm{p}$ HittingSet. To do so, we define $f(\langle \langle V, E \rangle, k \rangle) = \langle V, E, k \rangle$. The function f is obviously total and computable in polynomial time.
We still have to show: $G = \langle V, E \rangle$ has a vertex cover of size $\leq k$ if and only if there is a hitting set of size $\leq k$ hitting all sets in E:

- A vertex cover C of G is a hitting set for E because it contains an element from each set in E (= an end point of each edge). Moreover, if it satisfies the size bound of the VertexCover instance, it trivially satisfies (the same) size bound of the HittingSet instance.
- Let H be a hitting set of E. As we do not explicitly require that $H \subseteq M$, we remove all elements that are not in $M = V$, which only makes the set smaller. H contains an element from each set in E, which means it covers each vertex of the graph and is therefore a vertex cover of G. Hence, if there is such a hitting set of size at most k, there is a vertex cover of the required size.

In summary, f is total, polynomially computable, and satisfies the reduction property and thus shows that VertexCover $\leq_\mathrm{p}$ HittingSet. Since VertexCover is NP-complete and thus NP-hard, HittingSet also has to be NP-hard.

Exercise 12.2 (Polynomial Reduction; 4 marks)

Consider the following decision problems:

IndSet:

- *Given:* undirected graph $G = \langle V, E \rangle$, natural number $k \in \mathbb{N}_0$
- *Question:* Does G contain an independent set of size k or larger, i.e., is there a set $I \subseteq V$ with $|I| \geq k$ and $\{u, v\} \notin E$ for all $u, v \in I$?

Hence, if there is such a hitting set of size at most k, there is a vertex cover of the required size.
SetPacking:

- *Given:* finite set M, set of sets $\mathcal{S} = \{S_1, \dots, S_n\}$ with $S_i \subseteq M$ for all $i \in \{1, \dots, n\}$, natural number $k \in \mathbb{N}_0$
- *Question:* Is there a set $\mathcal{S}' \subseteq \mathcal{S}$ with $|\mathcal{S}'| \geq k$, such that all sets in $\mathcal{S}'$ are pairwise disjoint, i.e., for all $S_i, S_j \in \mathcal{S}'$ with $S_i \neq S_j$ it holds that $S_i \cap S_j = \emptyset$?

Prove that SetPacking is NP-hard. You may use that IndSet is NP-complete.

Solution:

We have to show IndSet $\leq_\mathrm{p}$ SetPacking. To do so, we define $f(\langle V, E\rangle, k) = \langle E \cup V, \mathcal{S}, k\rangle$ with $\mathcal{S} = \{S_v \mid v \in V\}$, where $S_v = \{e \in E \mid v \in e\} \cup \{v\}$. The function f is obviously total and computable in polynomial time.
We still have to show: $\langle V, E\rangle$ contains an independent set of size $\geq k$ if and only if $\mathcal{S}$ contains at least k pairwise disjoint sets:

- For an independent set $I \subseteq V$ we know $\{u, v\} \notin E$ for all $u, v \in I$. Consider the set $\mathcal{S}'_I = \{S_u \mid u \in I\}$. Since every $v \in V$ only occurs exactly in the set S_v, $\mathcal{S}'_I$ consists of $|I|$ different sets. We show by contradiction that these are additionally pairwise disjoint:

  Assume there are $S_u, S_v \in \mathcal{S}'_I$ with $S_u \neq S_v$ and there exist $e \in S_u \cap S_v$. We have $e \in E$ (and thus $|e| = 2$), since every $w \in V$ only occurs in one set. From $e \in S_u$ we get $u \in e$ and from $e \in S_v$ we get $v \in e$. Together this means that $\{u, v\} \in E$. This is a contradiction to I being an independent set.

- Let $\mathcal{S}' \subseteq \mathcal{S}$ be a set of pairwise disjoint sets. For all $S_u, S_v \in \mathcal{S}'$ with $S_u \neq S_v$ (and thus $u \neq v$) it then holds that there can be no e with $u \in e$ and $v \in e$, and therefore $\{u, v\} \notin E$. We can conclude that $\{v \mid S_v \in \mathcal{S}'\}$ is an independent set of size $|\mathcal{S}'|$ in $\langle V, E\rangle$.

In summary, f is total, polynomially computable, and satisfies the reduction property and thus shows that IndSet $\leq_\mathrm{p}$ SetPacking. Since IndSet is NP-complete and thus NP-hard, SetPacking also has to be NP-hard.

Exercise 12.3 (Decidability and NP; 3 marks)

Prove that no undecidable language can be in NP.

Solution:

Consider a language L in NP. By definition of NP there is an NTM M_L that accepts L in polynomial (and thus finite) time. This NTM can be simulated by a DTM M'_L in exponential (but still finite) time (see slides E2.10). This means we can specify a DTM that computes the characteristic function of L: we simulate M'_L on the input w and output 1 if M'_L accepts w and 0 otherwise. The critical step with this is that the simulation of M'_L always terminates (after exponential but finite time). Since the characteristic function of L is computable, L is decidable.