Exercises on Oracles, Relativization, and the Polynomial Hierarchy
CSCI 6114 Fall 2021

Joshua A. Grochow

September 9, 2021

An *oracle Turing machine* is a TM equipped with an additional tape, the oracle tape, and three additional states: Q, Y, N (for "query", "yes", "no"). If it enters the Q state, then it queries the oracle about the string x on the oracle tape. The oracle answers the query in the next time step with either YES or NO: if the oracle says YES, the TM enters state Y, and if the oracle says NO then enters state N. When an oracle machine is *instantiated* with a particular language L for the oracle, the oracle's answers are correctly answering whether x (the string on the oracle tape) is in L. In this case we speak of machine M with oracle L, sometimes denoted M^L.

Given a class of oracle TMs $\mathcal{M}$ and a class $\mathcal{C}$ of languages, we define $\mathcal{M}^{\mathcal{C}}$ to be the class of languages L such that there exists $O \in \mathcal{C}$ (for "oracle") and a machine $M \in \mathcal{M}$ such that M^O decides L correctly: $L = L(M^O)$. Many standard complexity classes such as $\mathsf{P}, \mathsf{NP}, \mathsf{PSPACE}, \mathsf{EXP}$ have such canonical corresponding classes of oracle TMs that we often write, e.g., $\mathsf{P}^{\mathcal{C}}$ for the class of languages decided by polynomial-time oracle Turing machines with some oracle from $\mathcal{C}$ (rather than giving a different notation for the class of polynomial-time oracle TMs). Such classes are colloquially called *relativizable*, because it is "clear" what it means to relativize them to an oracle.

1. Show that $\mathsf{P}^{\mathsf{P}} = \mathsf{P}$ and $\mathsf{NP}^{\mathsf{P}} = \mathsf{NP}$.

2. Show that $\mathsf{P}^{\mathsf{NP}} \neq \mathsf{NP}$ unless $\mathsf{NP} = \mathsf{coNP}$ (and thus PH collapses).

3. Show that $\mathsf{P}^{\mathsf{NP}} = \mathsf{P}^{\mathsf{coNP}}$, and more generally $\mathsf{P}^{\mathcal{C}} = \mathsf{P}^{\mathsf{co}\mathcal{C}}$.

4. Show that $\mathsf{NP} \cup \mathsf{coNP} \subseteq \mathsf{P}^{\mathsf{NP}} \subseteq \Sigma_2\mathsf{P} \cap \Pi_2\mathsf{P}$.

5. (a) Show that $\Sigma_2\mathsf{P} = \mathsf{NP}^{\mathsf{NP}}$.
   (b) More generally, show that $\Sigma_k\mathsf{P} = \mathsf{NP}^{\Sigma_{k-1}\mathsf{P}} = \Sigma_{k-1}\mathsf{P}^{\mathsf{NP}}$ and $\Pi_k\mathsf{P} = \mathsf{coNP}^{\Pi_{k-1}\mathsf{P}}$.

6. We say that a statement *relativizes* if it remains true in the presence of any oracle.

   (a) Show that $\mathsf{P} \subseteq \mathsf{NP} \subseteq \mathsf{PSPACE}$ relativizes, that is, for any oracle O, $\mathsf{P}^O \subseteq \mathsf{NP}^O \subseteq \mathsf{PSPACE}^O$.
   (b) What happens when we relativize the statement $\mathsf{P} \subseteq \mathsf{NP} \subseteq \mathsf{PSPACE}$ to a PSPACE-complete oracle?

7. Use the oracle characterization of PH to give an alternative, simpler proof that if $\Sigma_k\mathsf{P} = \Sigma_{k+1}\mathsf{P}$, then $\mathsf{PH} = \Sigma_k\mathsf{P}$.

8. Use the oracle characterization of PH to give a simple proof that Exercise 4 relativizes to give: $\Sigma_k\mathsf{P} \cup \Pi_k\mathsf{P} \subseteq \mathsf{P}^{\Sigma_k\mathsf{P}} \subseteq \Sigma_{k+1}\mathsf{P} \cap \Pi_{k+1}\mathsf{P}$.

9. Show that $\mathsf{NP}^{\mathsf{NP}\cap\mathsf{coNP}} = \mathsf{NP}$. This is an example of lowness:

   Definition 1. Given a relativizable complexity class $\mathcal{C}$, a language L is *low for* $\mathcal{C}$ if $\mathcal{C}^L = \mathcal{C}$. $\mathsf{Low}(\mathcal{C})$ is the class of all such languages: $\mathsf{Low}(\mathcal{C}) = \{L | \mathcal{C}^L = \mathcal{C}\}$.

10. The previous exercise showed that $\mathsf{NP} \cap \mathsf{coNP} \subseteq \mathsf{Low}(\mathsf{NP})$. Show that this is an equality.

11. The (relativized) Karp–Lipton Theorem says that for any oracle X, if $\mathsf{NP}^X \subseteq \mathsf{P}^X/poly$ then $\mathsf{PH}^X = \Sigma_2\mathsf{P}^X$. Use the fact that this theorem relativizes, together with what we know about the relationship between sparse sets and $\mathsf{P/poly}$ to show that PH collapses if and only if there exists a sparse set S such that PH^S collapses.

Resources

- There is also an oracle X relative to which $\mathsf{P}^X \neq \mathsf{NP}^X$ (Baker, Gill, & Solovay, *SIAM J. Comput.*, 1975). It's worth thinking about how you would construct such a thing! Hint: diagonalize against poly-time Turing machines.

  Combined with exercise 6(b), this shows that any proof resolving the P versus NP question must be non-relativizing.

The proof is covered in detail in Sipser §9.2, Du & Ko §4.3–4.8, Arora & Barak §3.5.

- I believe it is an open question whether there exists an oracle X relative to which PH "looks like" the arithmetic hierarchy, in the sense that: (a) PH^X is infinite, but (b) $\mathsf{P}^{\Sigma_k \mathsf{P}^X} = \Sigma_{k+1}\mathsf{P}^X \cap \Pi_{k+1}\mathsf{P}^X$ for all k.

- PH defined in terms of oracles: Homer & Selman §7.4, Du & Ko Ch. 3, Arora & Barak §5.5.

- General introduction to oracles: Homer & Selman §3.9 (in the context of computability, no poly-time bounds), Du & Ko §3.1 for nondeterministic poly-time oracle TMs, Arora & Barak §3.5

- Du & Ko §4.3–4.8 talk about other relativizations of NP and §9.6 talks about relativized PH.

- High-level discussions of relativization and its role in complexity: Moore & Mertens §9.4, Wigderson §5.1