1 Basic complexity results

Can you see how the time hierarchy theorem is related to the undecidability of the language of self-rejecting machines?

Question 1. *Compare the hard language constructed in the proof of the time hierarchy theorem to the language $DIAG$ of self-rejecting machines. Convince yourself that the time-hierarchy theorem is a time-bounded version of the undecidability of $DIAG$.*

The following question rehashes the proof idea of the time hierarchy theorem, to make sure you internalized it. It asks to show that using a "fancier" machine model we can get a tighter time hierarchy.

Question 2. *Suppose, for the sake of this question, that we're working in a Turing Machine model for which there is a Universal Machine U with almost no time overhead; that is, given input $\langle M \rangle, x$ such that $M(x)$ halts after T steps, $U(\langle M \rangle, x)$ halts after $O(T \cdot \log T)$ steps.*[1] *For this model, prove a time hierarchy theorem that is tighter than the one shown in class. What is the least T' such that you can show $\mathsf{TIME}[T'] \not\subseteq \mathsf{TIME}[T]$?*

Recall that we defined $\mathsf{TIME}[T]$ as the class of languages decidable in time $O(T)$, where this O-notation represents some slackness. Why didn't we define this class using time strictly T? The following question asks you to think about this.

Question 3. *Can we speed-up computation by using a larger alphabet and set of states (i.e., "stronger hardware")? By how much can we speed-up computation using this trick? How is this related to the slackness in the definition of* $\mathsf{TIME}[T]$*?*

[1] The model of multitape TMs has a universal TM U with overhead similar to this. Specifically, U runs in time $C_{|\langle M \rangle|} \cdot T \cdot \log(T)$, where $C_{|\langle M \rangle|}$ is a number that depends on the description size of M.