CS340: Theory of Computation	Sem I 2017-18

Lecture Notes 21: Hierarchy Theorems

Raghunath Tewari IIT Kanpur

Definition 0.1. A function $f : \mathbb{N} \to \mathbb{N}$ is said to be *space constructible* if there exists a TM M that on input 1^n marks $f(n)$ cells on the output tape and halts using $O(f(n))$ amount of work space.

Theorem 1 (Space Hierarchy Theorem)**.** *For every space constructible function $s(n) \geq \log n$ there exists a language that is in $\mathsf{SPACE}(s(n))$ but not in $\mathsf{SPACE}(s'(n))$, where $s'(n) \in o(s(n))$.*

Proof. We will show that there is a TM D that runs in space $O(s(n))$ and for every TM M that runs in space $o(s(n))$, D disagrees with M on at least one input. Hence $L(D)$ does not have any TM that runs in space $o(s(n))$.

Tentative design of D

Input: $\langle N \rangle$

1. Verify whether the input is a correct encoding of a TM. If not, *reject*.
2. Compute $s(n)$.
3. Simulate N on the string $\langle N \rangle$ using the work space of D. If at any point the space bound exceeds $s(n)$ then *reject*
4. If N halts then halt. *Accept* iff N rejects.

Clearly D runs in space $O(s(n))$. Moreover if N runs in space asymptotically smaller than $s(n)$ then by the definition of D, N and D differs on the input $\langle N \rangle$.
However there are two problems with the above construction, that can fortunately be taken care of.

- *Problem 1*: Even if $t(n) \in o(s(n))$ it can happen that for some small values of n, $t(n) \geq s(n)$. Hence for some TMs N' (with a small description) it can happen that N' runs in asymptotically lesser space but N' and D do not differ on any input.

  Solution: Pad the input with some redundant characters at the end. The input will be $\langle N \rangle 1^k$. This artificially increases the input length. If N runs in asymptotically lesser space than D then for some k the output of D and N will differ.

- *Problem 2*: What if N loops forever on the input $\langle N \rangle$ and does not halt?

  Solution: Here we use the fact that a TM using space $s(n)$ has at most $2^{k_1 s(n)}$ distinct configurations for some constant k_1. We simulate N for at most $2^{k_1 s(n)}$ number of steps using a counter and halt and reject if it does not accept by then.

Modified design of D

Input: $\langle N \rangle 1^k$

1. Verify whether $\langle N \rangle$ is a correct encoding of a TM. If not, *reject*.
2. Compute $s(n)$.
3. Simulate N on the string $\langle N \rangle$ using the work space of D. If at any point the space bound exceeds $s(n)$ then *reject*
4. If N runs for more than $2^{k_1 s(n)}$ number of steps then halt and *reject*.
5. If N halts then halt. *Accept* iff N rejects.

$\square$

Similarly there is a hierarchy theorem for time as well. It is slightly weaker than its space analogue.

Definition 0.2. A function $f : \mathbb{N} \to \mathbb{N}$ is said to be *time constructible* if there exists a TM M that on input 1^n outputs the binary encoding of $f(n)$ on the output tape and halts in $O(f(n))$ time.

Theorem 2 (Time Hierarchy Theorem)**.** *For every time constructible function $t(n) \geq n$ there exists a language that is in $\mathsf{TIME}(t(n))$ but not in $\mathsf{TIME}(t'(n))$, where $t'(n) \in o(t(n)/\log t(n))$.*