Time and Space Complexity- II

A non-deterministic TM M is said to be $t(n)$ time bounded if on every input of length n, **every** computation takes at most $t(n)$ steps. We then define NTIME($t(n)$) to be the set of languages accepted by $O(t(n))$ time bounded multi-tape NDTM's.

Hierarchy theorems still hold for non-deterministic time classes. The simulation of all NDTM's by one NDTM (universal machines) still proceeds the same way as for TM's. But now we cannot just "do the opposite". So, hierarchy theorems are much more complicated to prove.

We define space bounds for NDTM's in a similar manner to time bounds.

Savitch's Theorem If $s(n) \geq \log n$ is a nice space bound, then an $s(n)$-space bounded multi-tape NDTM can be simulated by a $O((s(n))^2)$ space bounded (deterministic) TM.

Proof There are at most $2^{O(s(n))}$ ID's of a $s(n)$-space bounded NDTM M. (Why ?) We construct a graph with one vertex for each possible ID (we only imagine this graph, it is not explicitely written down) and a directed edge from u to v if we can go from u to v in one step of the computation, if $u \vdash_M v$ in notation introduced earlier. Then the problem reduces to determining whether there is a directed path from the starting ID to some accepting one in $l = 2^{O(s(n))}$ steps. We solve recursively the more general problem of - given two ID's u, v and an integer l, determine whether we can go from u to v in at most l steps (denoted $u \models^l v$). To do this, we just check whether there exists a w such that $u \models^{l/2} w$ and $w \models^{l/2} v$. We cycle through all w 's; the current candidate w is written down in a register. We recursively first check whether $u \models^{l/2} w$; if yes, we then check whether $w \models^{l/2} v$. But note that for the second recursive call, we may reuse the space originally used by the first call. We only need to remember w. So, total space requirement $S(l)$ satisfies : $S(l) \leq s(n) + S(l/2)$, which solves to $S(l) \leq (\log l)s(n)$.

Theorem A $t(n)$ time bounded NDTM can be simulated by a $2^{O(t(n))}$ time bounded DTM.

Since the NDTM is $O(t(n)$ time bounded, it has at most $2^{O(t(n))}$ ID's and so now we may explicitely write down the graph discussed above and see if there is a path from the starting to an accepting computation.

We define the class P of languages accepted by polynomial time bounded TM 's and NP, the class of languages accepted by polynomial time bounded

NDTM's as

$$P = \cup_{k=1}^{\infty} \mathrm{DTIME}(n^k) \qquad\qquad NP = \cup_{k=1}^{\infty} \mathrm{NTIME}(n^k).$$

Clearly,, $P \subseteq NP$. Are they equal ?