Sublogarithmic space

Wednesday, November 20, 2024 8:16 AM

In this lecture, we will prove the following theorem:

Theorem 1: For every function $s : \mathbb{N} \to \mathbb{N}$ with $s(n) = o(\log \log n)$, we have $\mathrm{SPACE}(s) = \mathrm{SPACE}(1)$.

Proof: To prove the above theorem, it is enough to show that any TM M with $s_M = \omega(1)$ which halts on all inputs must satisfy

$$s_M = \Omega(\log \log)$$

Let M be a TM with $s_M = \omega(1)$. Thus, for any $m \in \mathbb{N}$, there is a minimum $n \in \mathbb{N}$ s.t. $s_M(n) \geq m$. Let $x \in \{0,1\}^n$ be such that the space cost of $M(x)$ equals $s_M(n)$ ($\therefore$ space cost of $M(x)$ is $\geq m$).

For each $i \in [n]$ (think of i as the position of the input head), define the residual configuration of $M(x)$ at i as the tuple (q, h_w, w).

state of M

position of work tape head

content of the work tape

That is, the residual configuration is simply the content of temporary storage of TM M on x (when the input head position is at i).

Now, for each position $i \in [n]$ of the input head, let

Now, for each position $i \in [n]$ of the input head, let

$\vec{r} := (r_1, r_2, \dots, r_\ell)$ where each r_j is a residual configuration

at $i \in [n]$ be the crossing sequence of $M(x)$

when the input head is at i.

There are at most $t := \underbrace{\gamma}_{\text{\# states of } M} \cdot \underbrace{s_M(n)}_{\text{\# positions of work tape}} \cdot \underbrace{2^{s_M(n)}}_{\text{contents of work tape}}$ possible

residual configurations.

Now, note that (for given input head position i), we have that $\ell \leq t$, otherwise by pigeonhole some residual configuration would repeat itself and thus M would loop forever.

Thus, there are at most t^t possible sequences $\vec{r}$ of residual configurations.

Let $b \in [n]$ be an input tape index whose crossing sequence contains a residual configuration which uses $s_M(n)$ work-tape space (such an index must exist by the choice of x).

Now one of the sets $[b-1]$ or $[b+1, n]$ must have size $\geq \frac{n-1}{2}$. Let's assume w.l.o.g. that $n-b \geq \frac{n-1}{2}$ (i.e. the size of $[b+1, n]$ is $\geq \frac{n-1}{2}$).

Now, if $t^t < \frac{n-1}{4}$, by pigeonhole, there are

three indices $b < i < j < k \in [n]$ s.t. their residual configura three indices $b < i < j < k \in [n]$ s.t. their residual configuration sequences are the same. And since we have 3 head positions, at least two of the values x_i, x_j, x_k are the same, since $x_a \in \{0,1\}$ $\forall a \in [n]$.

Suppose $x_i = x_j$ (the other cases are analogous).

Thus, let $y := x_1 x_2 \cdots x_i x_{j+1} \cdots x_n$ (note $|y| < n$).

Since the residual configuration sequences of $M(x)$ at i and j are the same, and $x_i = x_j$ we have that the behaviour of M on x and y is the same, which implies that the space cost of $M(y)$ is the same as the space cost of $M(x)$, contradicting minimality of x. (The space cost of $M(y)$ is $s_M(n)$ since the crossing sequence of b in y is the same as the crossing sequence of b in x).

Hence we must have $t^t \geq \frac{n-1}{4}$. Since $t = \gamma \cdot s_M(n) \cdot 2^{s_M(n)} \leq$
$\leq 2^{3 s_M(n)}$, we have $\left(2^{3 s_M(n)}\right)^{2^{3 s_M(n)}} \geq \frac{n-1}{4} \Rightarrow$

$2^{3 s_M(n)} \cdot 3 s_M(n) \geq \log n - 3 \Rightarrow 3 s_M(n) + \log(3 s_M(n)) \geq \log\log n - 3$

$\Rightarrow s_M = \Omega(\log\log)$ as we wanted. $\boxed{M}$

Constant Space

Now let's investigate the question of what it means for a TM M to have constant space cost.

Note that if M has constant space cost, we might as well include all possible contents of the work tape (as well as the head positions) into the description of M itself! Thus having constant space cost is equivalent to not having any work space at all!

This should remind you of an automaton, which you may have seen in previous classes. More precisely we have:

Definition 1: a two-way deterministic finite automaton (2DFA) is a TM that has a read-only input tape and no extra tape.

The above discussion implies the following proposition:

Proposition 1: every language in SPACE(1) can be decided by a 2DFA.

Note that 2DFAs have this "two-way" adjective, to emphasize that the finite automaton can move the input head left or right. If we restrict the input tape emphasize that the finite automaton can move the input head left or right. If we restrict the input tape head to only move right, we get the deterministic finite automaton (DFA) model.

Proposition 2: any language that can be decided by a 2DFA can also be decided by a DFA.

Definition 2: the class REGULAR is the class of languages that can be decided by a DFA.

Hence the above propositions imply:

Theorem 2: REGULAR = SPACE(1).