Time & Space

Monday, October 28, 2019 6:09 AM

plan

- Password Game
- Turing Machines

"anything computable is computable by a TM"

Nondeterminism does not extend the class of languages accepted by TMs.

Use BFS on computation tree of

Configurations = ⟨TAPE CONTENT, STATE, HEAD POS.⟩

Thm. $NTIME(t(n)) \subseteq DTIME(2^{O(t(n))})$.

Pf. $NTIME(t(n)) \Rightarrow \exists$ a computation tree of depth $t(n)$ that accepts L.

What is max size of this tree? $(|\Gamma| \cdot |Q| \cdot 2)^{t(n)}$

since $|\Gamma|, |Q|$ are constants, $= 2^{O(t(n))}$.

BFS.

What about space $s(n)$?

Thm. (EASY) $NSPACE(s(n)) \subseteq DTIME(2^{O(s(n))})$
$\subseteq DSPACE(2^{O(s(n))})$.

Pf.

Total number of possible configurations with $s(n)$ space $\leq$ $|\Gamma|^{s(n)} \times |Q| \times (s(n)+n)$

$$= 2^{O(s(n) + \log(s(n)+n))}.$$

$$= 2^{O(s(n))}$$

$s(n)$ is a space-constructible function:

$s: \mathbb{N} \to \mathbb{N}$, $s(n) \geq \log_2 n$, $\exists$ TM s.t. on input 1^n outputs $s(n)$ in binary and uses $O(s(n))$ space.

It is possible for a TM to mark off $s(n)$ squares on its tape.

Thm. $NSPACE(s(n)) \subseteq DSPACE(O(s(n)^2))$.

Pf. with space $s(n)$, # configurations $= 2^{O(s(n))}$.

Imagine a **large** directed graph $G = (V, E)$

V = configurations

E = transitions of TM.

$\exists$ directed path from starting configuration C_0 to accepting configuration?

DFS?

└ needs depth # configurations in memory

BFS?

└ needs width # configurations in memory

[illegible] with nondeterminism: guess the

Easy with "nondeterminism": guess the next vertex!

space $= s(n)$.

Idea: try all possible "middle" vertices.

PATH(u, v, K): $\exists$ path of length $\leq K$ between u and v?

If $u = v$ OR $((u,v) \in E$ AND $K \geq 1)$
return YES

Else for $w \in V \setminus \{u, v\}$:
if PATH$(u, w, \lfloor \frac{K}{2} \rfloor)$ AND PATH$(w, v, \lceil \frac{K}{2} \rceil)$:
RETURN YES

RETURN NO.

Total space used?
$s(n) \times$ Depth of recursion

$d(\lceil K \rceil)$

$$d(k) \leq 1 + d\left(\left\lceil \frac{k}{2} \right\rceil\right)$$

$$\leq \lceil \log_2 k \rceil$$

What is k? $|V|$

$$\log_2 k = O(s(n))$$

$$\Rightarrow \text{total space} = O(s(n)^2).$$

Does more space (time) give TMs more power? i.e. they recognize more L?

<u>Thm</u>. For any space-constructible function $s(n)$, $\exists$ language L that can be decided by a TM using space $O(s(n))$ and <u>cannot</u> be decided by any TM using space $o(s(n))$.

<u>Pf</u>. Let L be the language accepted by

Pf. Let L be the language of the following machine D:

on input $\langle M, 1^n \rangle$:

- Mark $s(n)$ space on tape
- Count # steps of D
- Run M on 1^n (if M is not a valid TM description, REJECT)
- if space used exceeds $s(n)$, REJECT
- if time used —— $|\Gamma|^{s(n)}$, REJECT.
- Else, if M accepts REJECT
  if M rejects, ACCEPT.

Claim 1. L can be decided using $O(s(n))$ space.

Pf. D decides L and uses $O(s(n))$ space.
It always terminates.

Claim 2. No TM using space $o(s(n))$ can

Claim 2. No TM using space $o(s(n))$ can decide L.

Pf. Suppose there is such a decider M_L.

Run D on $\langle M_L, 1^n \rangle$. Since M_L uses $o(s(n))$ space, D can simulate it.

Then D accepts iff M_L rejects

rejects — M_L accepts.

Contradiction! they decide the same language L.