

Time & Space

Monday, October 28, 2019

6:09 AM

Ques
- 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 = $\langle \text{TAPE CONTENT}, \text{STATE}, \begin{matrix} \text{HEAD} \\ \text{POS.} \end{matrix} \rangle$

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

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

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

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

BFS.

What about space $s(n)$?

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

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:

$s: \mathbb{N} \rightarrow \mathbb{N}$, $s(n) \geq \log_2 n$, $\exists \text{TM } M$ 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. $\text{NSPACE}(s(n)) \subseteq \text{DSPACE}(O(s(n)^2))$.

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

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

V = configurations

E = transitions of TM.

∃ directed path from starting configuration
to accepting configuration?

DFS?

needs depth # configurations
in memory

BFS?

needs width # configurations
in memory

... with nondeterminism: guess the ,

Easy with nondeterminism: guess the next vertex!

space = $S(n)$.

Idea: try all possible "middle" vertices.

$\text{PATH}(u, v, k)$: \exists path of length $\leq k$,
between u and v ?

If $u=v$ OR $((u, v) \in E \text{ AND } k \geq 1)$

return YES

Else for $w \in V \setminus \{u, v\}$:

if $\text{PATH}(u, w, \lfloor \frac{k}{2} \rfloor) \text{ AND } \text{PATH}(w, v, \lceil \frac{k}{2} \rceil)$:

RETURN YES

RETURN NO.

Total space used?

$S(n) \times \text{Depth of recursion}$

$\dots \dots \dots \cdot S(k)$

$$d(n) \leq 1 + d(\lceil \frac{n}{2} \rceil)$$

$$\leq \lceil \log_2 n \rceil$$

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 ?

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

Pf. Let L be the language accepted by

PT: Let L be the language
the following machine D :

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

- Mark $s(n)$ space on tape
- Count # steps of D
- Run M on $\langle M, I^n \rangle$ (if M is not
a valid TM description, REJECT)
- if space used exceeds $s(n)$, REJECT
- if time used — $C^{s(n)}$, 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 $O(\Delta(n))$ 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 $O(\Delta(n))$ space, D can simulate it.

Then D accepts iff M_L rejects
rejects — M_L accepts.

Contradiction! they decide the same language L .
