CS 579: Computational Complexity: Lecture 2

admin: put out, due 02-05 3:30pm

sign up for pizza

today: Turing Machine Review

what is computation?

Q: can we schedule all courses, no conflicts?

A: decision

\[ \text{find a solution} \]

search

\[ \text{try to find a solution} \]

validity

\[ \text{verify a solution} \]

optimization

\[ \text{best solution} \]

counting

\[ \text{how many solutions?} \]

def: a language is \( L \subseteq S^* \)

def: a Turing Machine (TM) x

finite automaton

\[ \text{Finite tape} \]

Actually...

- \( Q \) set of states
- \( \Gamma \) tape alphabet
- \( \delta \) transition function

\[ \delta : (\text{current symbol, state}) \rightarrow (\text{head movement, new symbol}) \]

a TM computes \( x \)

- tape initialized to \( x \)

- head placed at start of tape

- input = until reach \( \text{halt} \)

language \( L \) of TM \( M \):

\[ L(M) = \{ \text{all } x \in \Sigma^* \text{ such that } M \text{ accepts } x \} \]

more general:

output \( E \)

\[ \text{Min } x \text{ such that } y \subseteq \Sigma \text{ output } = \text{ accept or not accept} \]

def: a TM \( M \) runs in \( \text{time } t(n) \) if for all \( x \in \Sigma^* \)

\[ \text{Min } x \text{ halts in } t(n) \text{ steps} \]

\[ \text{want cases not in I} \]

space \( s(n) \)

\[ \text{use } s(n) \text{ time and } \Sigma(n) \text{ tape} \]

def: \( \text{TIME}(s(n)) = \{ \text{all } L \text{ such that } L \text{ decided by TM in time } \leq O(s(n)) \text{ if big Oh} \}

space \( s(n) \)

\[ \text{space } \leq O(s(n)) \]

\[ \text{L } \end{equation} \]
Church-Turing Thesis: anything "computable" is computable by a TM

\[ \text{def: } P = \bigcup_n \text{TIME}(n^k) \text{ if efficient in theory?} \]

\[ \text{Q: } \text{what is } \#P? \text{ if no one really knows? } \]

\[ \text{def: } \text{EXP} = \bigcup_n \text{TIME}(2^{n^k}) \]

\[ \text{Prop: } P \subseteq \text{EXP} \]

\[ \text{then: } 2^2 = 4 \text{ and } 2 \]

= Question:

Simulation: Hardware \[ \not\equiv \] Software

TM M \[ \approx \] description \langle M \rangle of TM M

Theorem: There is a universal TM U that on input any string \( s \) can simulate \( M \) on input \( s \).

(1) \( M \) acc/rej \( s \) in time \( O(C_m \cdot s \cdot s) \leq O(2^{c_m \cdot s^2}) \)

Intuition: CPU can run any programming language in space \( O(C_m \cdot s) \)

Key points: U is fixed, fixed tape alphabet \( \Gamma \) depends on \( M \)

tape alphabet \( \Gamma \) of \( M \) may be large

Proof Sketch:

a single step

Time: \( O(c_m \cdot s) + \)

Space: \( O(c_m \cdot s) \)

Rank: \( n \) tape steps can see \( O(c_m \cdot t \cdot s) \) tape. I see qualitatively correct
Theorem: For any "nice" function \( f(n) \) with \( f(n)^2 = O(g(n)) \),
\[
\text{TIME}(f(n)) \subseteq \text{TIME}(g(n))
\]

Proof: Suppose \( M \) is a TM running in time \( f(n) \).
\[
\text{If } M \text{ accepts } x \Rightarrow \text{acceptable in time } O(f(n))
\]

- \( f(n) \) is time constructible if \( n \rightarrow (\log n)^f \)

- \text{monotonically increasing}

- \( f(n) \) can directly decide \( \emptyset \)

Proposition: If \( f(n) \) is time constructible, then \( \text{TIME}(f(n)) \subseteq \text{TIME}(g(n)) \)

Proof: Consider \( M \) running in time \( f(n) \).

- \( M \) halts in time \( O(f(n)) \)

- \( M \) accepts \( x \) if \( g(n) \)

- \( M \) rejects \( x \) if \( \neg \text{accepts} \)

- \( M \) is in \( \text{TIME}(g(n)) \)

- \( \text{TIME}(f(n)) \subseteq \text{TIME}(g(n)) \)

Claim: \( L(0) \leq \text{TIME}(g(n)) \)

Proof: Use \( U \) to simulate \( M \) on \( \langle M, t^k \rangle \)

- \( U \) is in \( \text{TIME}(g(n)) \)

- \( U \) accepts \( \langle M, t^k \rangle \)

- \( U \) simulates \( M \) in \( \langle M, t^k \rangle \)

- \( \text{TIME}(f(n)) \subseteq \text{TIME}(g(n)) \)

Rmk: Need \( f(n) \) time constructible

Claim: \( L(0) \leq \text{TIME}(f(n)) \)

Proof: Suppose \( \langle M, t^k \rangle \) in \( \text{TIME}(f(n)) \)

- \( M \) runs in \( \text{TIME}(f(n)) \)

- \( f(n) \) depends only on \( U \)

Rmk: Need \( M \) time constructible

Claim: \( L(0) \leq \text{TIME}(f(n)) \)

Proof: Suppose \( \langle M, t^k \rangle \) in \( \text{TIME}(f(n)) \)

- \( M \) runs in \( \text{TIME}(f(n)) \)

- \( f(n) \) depends only on \( U \)

Rmk: "constant" \( C_M \) is not constant
What is \( \mathcal{L}(D) \)? I'm unsure.

**Def.** BOUNDED HAlting = \( \{ <M, x> : M \text{ accepts } <M, x> \text{ in } f(|x|) \text{ steps} \} \)

\( f(n) \text{ is constructible} \in \text{TIME}(f^2) \text{ and some } \sqrt{f} \text{ and } \text{clock p} \)

This SPACE hierarchy:

\[ \text{SPACE}(f(n)) \subseteq \text{SPACE}(g(n)) \]

Which simulates has no penalty

ps: 1 on

sign up for pizza

return: nondeterminism

NP

reducibility

NP-complete

NP-intermediate