Michael A. Forbes
miforbes@illinois.edu
2024-09-03.1
cs579

cs579 Computational Complexity: Lecture 3 (2024-09-03)

logistics: pset 1 out, due 09-12

last lecture: time complexity
- def [#steps, worst case wrt input]
- examples [0^n1^n — n^2 1-tape; — $n\lg n$ 1-tape (tight); — n 2-tape]
- robustness [2-tape vs 1-tape] [model indep]

vs lecture 2.
n vs m
vs 2023
vs 2023

today: nondeterminism
[move to front]
[laptops]

def: PATH = $\{\langle G,s,t\rangle : G=(V,E)$ directed, $s,t\in V$, G has $s \rightsquigarrow t$ path$\}$

^ any reasonable encoding into Σ^* [alphabet] [adj matrix, list]

{ efficiently transformable into other encoding [encoding indep] [poly time algo, worst case]

prop: PATH $\in$ P [374]

sketch: depth first search

def. HAMPATH = $\{\langle G,s,t\rangle : G$ has $s\rightsquigarrow t$ path visiting each vertex exactly once$\}$

↳ Hamiltonian path

Q: HAMPATH $\in$ P? [open]

vs 2023

A: [no answer] "algo" [not real] "on input $\langle G=(V,E),s,t\rangle$:

n^n guesses | poly(n)
- for $V=\{v_{i_1},\ldots,v_{i_n}\}$ guess $v_{i_1},\ldots,v_{i_n}\in V$, $n=|V|$ [check] [a case corr]
- reject if repeated vertex
- reject if $v_{i_1}\neq s$, $v_{i_n}\neq t$ or $(v_{i_j},v_{i_{j+1}})\notin E$ some j
- accept"

correctness - clear

complexity - if guess is ham-path, then [accepting] efficient

n^n guesses [accepting]

accepting.

goal: model "guessing" as a computational resource [non determinism] [non realistic, is useful to characterize problems]

def a nondeterministic Turing machine (NTM) is a TM

with transition function $\delta: Q \times \Gamma \to 2^{Q\times\Gamma\times\{L,R,S\}}$ [states] [tape read] [power set] [state] [write] [move] [>1 choice] [set of possible transitions] [guess within set]

!! vs 2023. δ.

a branch is a sequence of valid transitions

a branch accepts input x if sequence reaches q_{acc} [then halts]

—— " —— rejects —— " —— q_{rej} [then halts]

[a] branch. vs 2023 ↳ same def

or - sequence dies $\leftarrow \delta(-)=\emptyset$ [no valid transition]
- loops [no halting]

vs 2023.

NTM N accepts input x if some branch of N on x accepts [guess cert valid]

—— " —— rejects —— " —— all —— " —— rejects [all guess fail]

vs 2023.

time of N on x is length of longest branch [some branch loops ⇒ N on x loops]

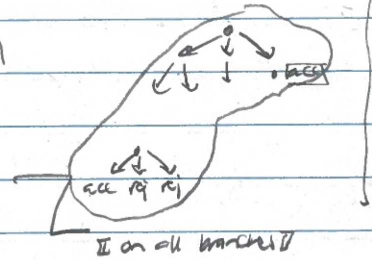
vs 2023.
vs 2023.
miforbes

Michael A. Forbes
miforbes@illinois.edu
2024-09-03.2
CS579

• vs 2023
• – vs 0

eg: $|\delta(-)| = 1$ ⇒ deterministic [non-deterministic]
[branches]

TM ↓ ↓ ⋮ ↓ acc/rej | t | NTM

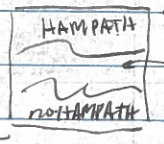


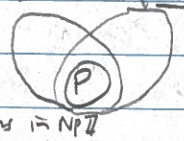
def: NTM N runs in time $t(n): \mathbb{N} \to \mathbb{N}$ if
for all $x \in \Sigma^*$ N acc/rej in $\leq t(n)$ steps [accepted inputs]
$NTIME(t(n)) = \{L : L = L(N), N \text{ runs in time } O(t(n))\}$
$NP = \cup_k NTIME(n^k)$ [nondeterministic polytime]
lem: $P \subseteq NP$ [use single transition]
rmk: NP is model invariant [1-tape vs 2-tape]
Q: what is in NP?
eg: HAMPATH ∈ NP
pf: idea: use many branches to try all possible paths
algo: N = " on input $\langle G=(V,E), s, t\rangle$:
– for $V = \{v_1, \ldots, v_n\}$, non-deterministically write down $v_{i_1}, \ldots, v_{i_n} \in V$ [$n = |V|$]
eg: $\delta(q, \gamma) \mapsto \{(q', 0, R), (q', 1, R)\}$ [guesses 1 bit [onto tape]]
⇒ guessing $\lg n$ bits produces $v \in V$
⇒ guessing $n \lg n$ produces
– accept if $v_{i_1} \to v_{i_2} \to \cdots \to v_{i_n}$ is valid s→t hampath"
($v_{i_1} = s$, $v_{i_n} = t$)

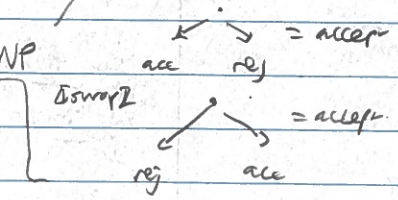
[delete?] correctness: clear [same algo as above]
complexity: each branch is poly(n) steps [but many branches] ∎
[another view]
def: verifier for language L is TM V w/
$L = \{x : V \text{ accepts } \langle x, w\rangle, \text{ some } w \in \Sigma^*\}$ [encoding] [witness]
V runs in time $t(n): \mathbb{N} \to \mathbb{N}$ if for all $x \in \Sigma^n$
$x \in L$ iff V accepts $\langle x, w\rangle$ for some $w \in \Sigma^*$ in $\leq t(n)$ steps.
⇒ $|w| \leq t(n)$ [can only read that much]
• 2023
• vs Sipser
prop: $L \in NP$ iff L has polytime verifier
pf: ⇒ NTM N acc L in $\leq t(n)$ steps
$A = \{\langle x, w\rangle : w \text{ is description of accepting branch}\}$ [sequence of transitions]
⇒ $x \in L$ iff $|w| \leq O(t(n))$ exists
each transition takes $O(1)$ bits to describe
and checking if w is "clearly" efficient [simulate N]

Michael A. Forbes
miforbes@illinois.edu
2024-09-03.3
CS579

$\in$: V $t(n)$-time verifier for L
else N = "on input x:
- guess $w \in \Sigma^*$ of length $\leq t(n)$
- accept if V accepts $\langle x, w \rangle$"
correctness, complexity: clear $\blacksquare$
rmk. both viewpoints useful
Q: P vs NP?
def: $t(n): \mathbb{N} \to \mathbb{N}$. $TIME(t(n)) \subseteq^{(1)} NTIME(t(n)) \subseteq^{(2)} TIME(2^{O(t(n))})$
[single transition options]
pf: (1): clear.
(2): verifier version [NTM is exercise]
V $t(n)$ time verifier for L
M = "on input x:
- for all $w \in \Sigma^{\leq t(n)}$ [$|\Sigma^{\leq t}| = 2^{O(t)}$, $O(t(n))$]
- check if V accepts $\langle x, w \rangle$
- accept if V ever accepts"
correctness: clear [accounting]
complexity: $poly(t, 2^{O(t)}) = 2^{O(t)}$ $\blacksquare$
Q: HAMPATH $= \Sigma^* \setminus$ HAMPATH $\in$ NP? [verify non-existence] [open]
def: noHAMPATH $= \{\langle G, s, t \rangle : G$ has no $s \to t$ hampath$\}$
lem: HAMPATH $\leq$ noHAMPATH [can formalize via next lecture] [by handling improper encodings] [not $\langle G, s, t \rangle$]
def: $coNP = \{\overline{L} : L \in NP\}$ [complement]
lem: $coP = \{\overline{L} : L \in P\} = P$ [swap acc/rej]
rmk: swapping acc/rej state does not yield NP = coNP
Q: P vs NP vs coNP?
[control] conj: $P \neq NP$ [lots of problems in NP] [exponential easier than obvious (guess, proof)] [hard! many clever also, need to rule all out]







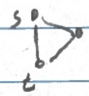
eg: PRIMES $= \{\langle n \rangle : n$ is prime integer$\}$
[binary] encoding, length $O(\lg n)$
n prime iff [no] non-trivial factorization $n = a \cdot b$, $a, b < n$
$\Rightarrow$ PRIMES $\in TIME(poly(n))$ [brute force] [not efficient]
n [not] prime iff [factors]
$\Rightarrow$ PRIMES $\in NTIME(poly(\lg n))$
$\Rightarrow$ PRIMES $\in coNP$
"N = on input $\langle n \rangle$:
- guess $a, b < n$
- accept if $n = a \cdot b$"

Michael A. Forbes
mforbes@illinois.edu
2024-09-03 . 4
CS 579

Q: dec faster?

thm [Pratt 75]: PRIMES ∈ NP [in coNP] [elementary number theory] [took a while] [more involved]
thm [Agrawal Kayal Saxena 02] PRIMES ∈ P [many intermediate works] [established evidence in P] [more elementary number theory]

def [another example] $G=(V,E)$ [un]directed, $s \neq t \in V$ [can repeat vertices]
$s \rightsquigarrow t$ path is Eulerian if all edges are used exactly [once]

eg:  no eulerian path,  eulerian

def: EULERPATH = {<G,s,t>: G undirected, [connected] w/ $s \rightsquigarrow t$ eulerian path} [no euler path if unconnected] [as in HAMPATH]
prop: ——"—— ∈ NP

prop: G connected $s \neq t$
G has $s \rightsquigarrow t$ eulerian path iff
- deg s, deg t odd [# incident edges]
- $v \neq s,t$, deg v even
[efficiently checkable!] [non obvious]

cor: EULERPATH ∈ P

sketch: ⟹: $p: v_0 \to v_1 \to \dots \to v_{m-1} \to v_m$, $v_0 = s$, $v_m = t$
adds 1 to deg s; adds 1[2] to deg v_1; adds 1 to deg t

$$\deg v = \deg_p v = \begin{cases} 2(\#\text{ occur of } v \text{ in } p) & v \neq s,t \text{ [even]} \\ \text{——"——} - 1 & v = s,t \text{ [odd]} \end{cases}$$
p eulerian

⟸: exercise [interesting direction] [elem. combin]

today: nondeterministic TMs [not realistic] [models guessing] [verifiers via]
NTIME, NP
HAMPATH [in NP, open if in P] [cleverer algs - primes - euler path]

next lecture: NP-completeness
logistics: pset 1 due 09-12
office hours now