Today
- Universal Turing Machines
- DTIME $(T(n))$
- Non-determinism, NP
- Reductions
- NP-completeness

CS5.203.1
Computational Complexity
- Lecture #2 (17 Feb,'21)

Instructor: Prahladh Harsha

Last time
- Model of computation - Turing Machine (k, Γ, Q, δ)
- Robustness
- Alphabet $B = \{0, 1, \triangleright, \square\}$

Claim 1 For every $f : \{0,1\}^* \to \{0,1\}^*$ & time-constructible $T : \mathbb{N} \to \mathbb{N}$

f is computable by a TM M using alphabet Γ in time T

$\Downarrow$

f is computable by a TM M' using alphabet B in time $4 \log |\Gamma| T$.

- # tapes

Single tape: TM only one Read/Write tape

Claim 2: (k tapes $\to$ single tape)

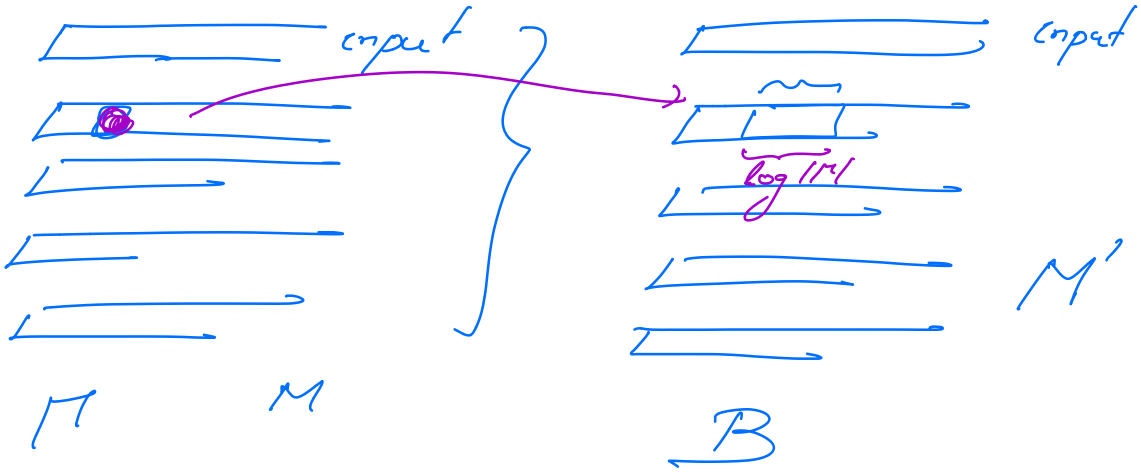
For every $f : \{0,1\}^* \to \{0,1\}^*$ & time-constructible $T : \mathbb{N} \to \mathbb{N}$

f is computable in time T by a TM using k tapes

$\Downarrow$

f is computable in time $5kT^2(n)$ by a single tape TM M'

Alphabet Reduction to B:

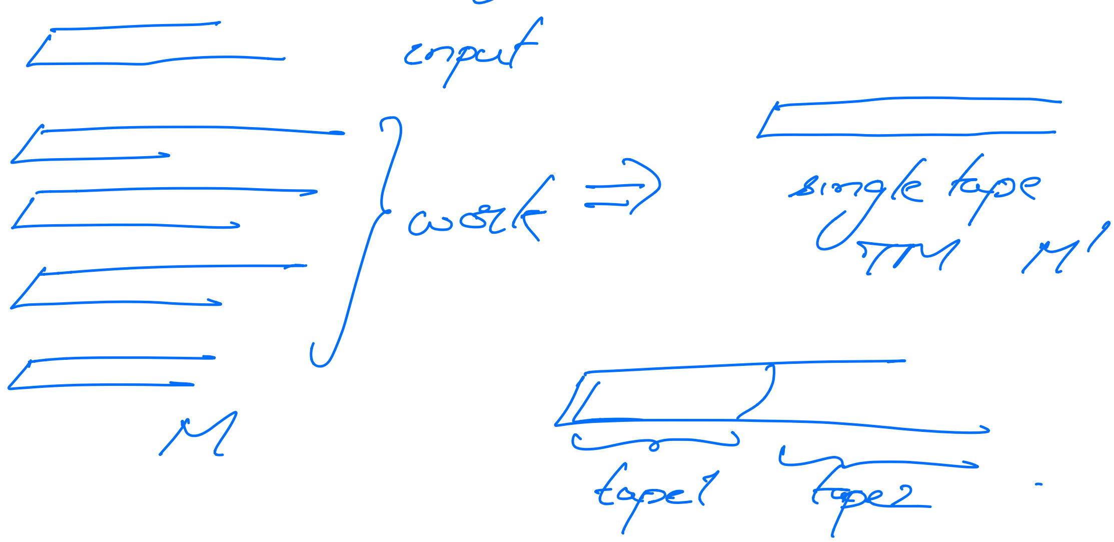


M': Encode Γ in binary using $\log|\Gamma|$ bits

Q' - state space - expanded

Proof of Claim 2:

(k tapes to single tape)



Interleave tapes & write all on single tape.

1^{st} tape — $1, k+1, 2k+1, 3k+1 \ldots$

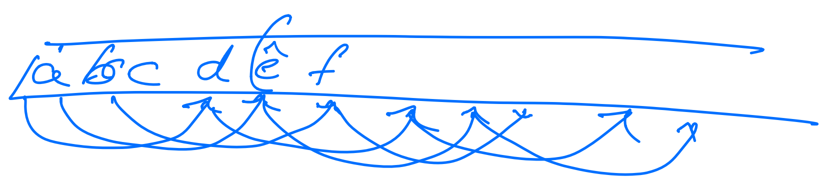
2^{nd} tape — $2, k+2, 2k+2, \ldots$

k^{th} tape — $k, 2k, 3k$

Symbol of original TM

← 2 corresponding symbols tap

$a \longrightarrow a, \hat{a}$



Single step of TM M

— simulated by a forward & backward sweep of $T(n)$ steps each.

Total # steps — ~~G~~ $T(n) \cdot T(n)$

$= T^2(n)$ steps.

Position of head of TM at time i

— function of $\begin{cases} M \\ i \\ x \text{ input} \end{cases}$

Simulation above
- head position depends only on $|x|$
but not x itself.
$(n = |x|)$.

Oblivious TM:

TMs - where head position at time i is a function of M, i, & length of input x
(not the actual i/p x)

Any k-tape machine M running in time T can be simulated by a single-tape oblivious TM in time $O(kT^2)$

Different TM:
Universal TM:

TM - has a string description
M — $\underbrace{(k, \Gamma, Q, \delta)}_{\text{string} \cdot \to \text{description}}$

$\mathcal{U}(\alpha, x)$
α - description of a TM M_α
x - real i/p

$\mathcal{U}(\alpha, x) = M_\alpha(x)$.

TM — strings

$M \longleftrightarrow (k, \Gamma, Q, \delta)$

$M \longmapsto \langle M \rangle, \lfloor M \rfloor$

$M_\alpha \longleftarrow\!\!\! \dashv \alpha$

Conventions:

1. Every string corresponds to some TM.
2. Every TM is represented by infinitely many strings.

Universality:

Universal TM: $\mathcal{U}$

two i/ps

α — description of M_α.

$x \rightarrow$ (real) input.

$\mathcal{U}$: On input (α, x)

Runs the TM M_α on x.

Thm: $\exists$ a UTM $\mathcal{U}$ s.t. $\forall\ \alpha, x \in \{0,1\}^*$

$$\mathcal{U}(\alpha, x) = M_\alpha(x).$$

But furthermore, $\forall \alpha$, $\exists$ $C = f(\alpha)$ s.t.

M_α halts in time T on i/p x

$\Downarrow$

$\mathcal{U}$ halts in time $CT \cdot \log T$ on i/p (α, x).

In class: weaker version w/ $CT\log T$ replaced by CT^2

Pf: UTM $\mathcal{U}$

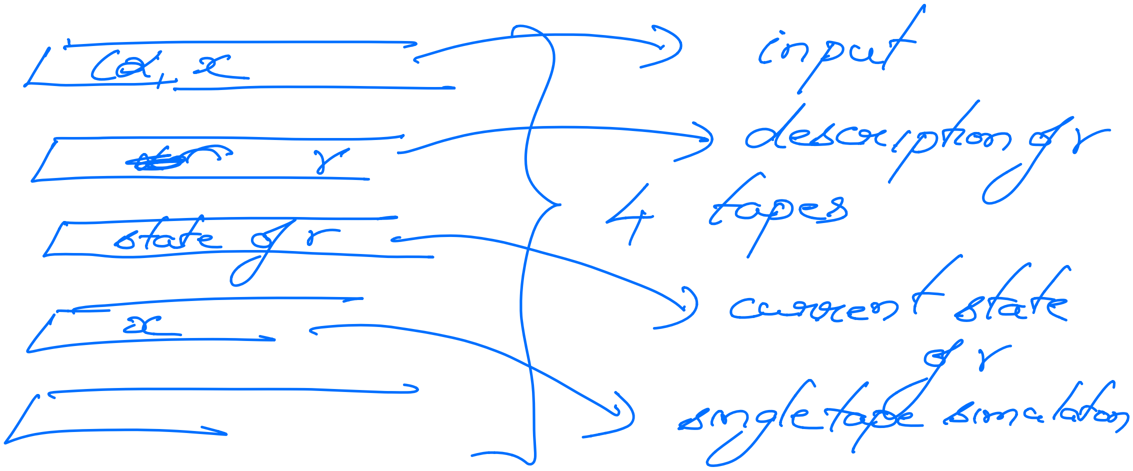
On input (α, x)

α -- description of some k-tape TM on Γ

i/p

$\mathcal{U}$ needs to simulate M_α even if M_α #tapes > #tapes of $\mathcal{U}$ & $\Gamma(M_\alpha) \neq \Gamma(\mathcal{U})$.

α (k-tape TM w/ alphabet Γ) $\xRightarrow{\text{Claim 1}}$ β (k-tape TM w/ alphabet B) $\xRightarrow{\text{Claim 2}}$ γ (single -tape w/ B')



UTM $\mathcal{U}$ is a 4-tape TM w/ B' that simulates any other TM w/ at most a quadratic overhead.

UTM is oblivious w.r.t input x (not necessarily w/ input α)

Efficient Computation:

Parity
Matching.
Connectivity

— "Easy" $f:\{0,1\}^* \to \{0,1\}^*$

Decision Problem: 0/1 answers
$f:\{0,1\}^* \to \{0,1\}$ (Languages

$L_f = \{x \mid f(x) = 1\}$.

Given a problem
↳ decision problem "equivalent" to the problem

Restrict attention to
Decision Problems / Languages

T- time constructible fn

DTIME(T(n)):

Language $L \in$ DTIME(T(n)) if
$\exists$ a TM M & a constant c
s.t. M "decides" L on time c·T
compute
for every i/p.

Parity $\in$ DTIME(n).

Connectivity
CONN = $\{(G, s, t) \mid G$ is a graph & there exists a path from s to t in G$\}$

CONN $\in$ DTIME(n^2)

$SAT = \{ \varphi \mid \varphi$ is a satisfiable CNF formula $\}$

$SAT \in DTIME(2^n)$

Matching $\in DTIME(n^c)$ for some constant c.

$$P = \bigcup_{c : c \geq 1} DTIME(n^c)$$

P — surrogate — "efficient" computation

Discussion: ① Randomized / Quantum

Caveats

② Worst-case input analysis (run in poly time on every input rigid constraint)

③ n^2 vs n^{1000} (coarse notion of efficiency).

$$EXP = \bigcup_{c : c \geq 1} DTIME(2^{n^c})$$

$SAT \in EXP$

$P \subseteq EXP$

Examples of Problems

① SAT
INDSET (decision of finding the largest independent set in a graph)

② INDSET $= \{ (G, k) \mid \exists$ an ind set of size $\geq k \}$

③ COMPOSITE $= \{ n \mid n$ is written in binary, n is not prime$\}$.

④ TSP (Travelling Salesperson Problem).

— When the answer is YES, there is a "short" proof (polynomial-sized) that proves why the answer is YES

— NP:
$L \in NP$
iff
$\exists$ two poly p, q & a TM M st
$x \in L \iff \exists\, w \in \{0,1\}^{p(|x|)}$, $M(x, w) = 1$

Verifier V

& furthermore M runs in time at most $q(|x|)$.

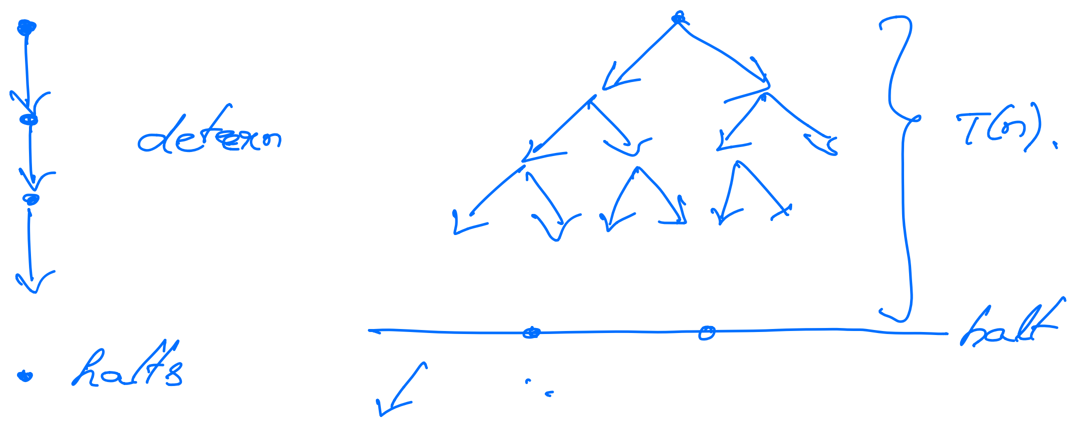
NP — terms of efficient Verifier.

Non-deterministic Turing Machines

$$\delta : Q \times \Gamma^k \to Q \times \Gamma^{k-1} \times \{L, R, S\}^k$$

Non-deterministic TM

· δ_0 · & δ_1



$L \in NTIME(T(n))$ if there exists a NTM M st

for every

$x \in L \iff \exists$ a path that causes the m/c to accept within $\le T$ time steps.

Proposition: $NP = \bigcup_{c \geq 1} NTIME(n^c)$

Verifiers

Non-determinism

Next time:- - Reduction

- Complete Problems
- Cook-Levin Theorem
- Properties of NP.