Nondeterministic TMs

- An <u>NDTM</u> is similar to TM.
  Except that now there are two transition functions $(\Gamma, Q, \delta_0, \delta_1)$.

- At any configuration C its transition is no more unique.
  It has two allowed moves, one following δ_0 & the other following δ_1.

- An NDTM M is said to <u>accept</u> an input x, if $\exists$ a sequence of choices leading to the accept (i.e. output 1).
  If $\nexists$ such a choice then M is said to <u>reject</u> x (i.e. output 0).

- The <u>time taken by M on x</u> is the max. #(steps by M to halt on x); the max. being over all possible sequence of choices.

- NDTMs are much more abstract than the TMs; we cannot identify them with a "physical" device.

- NDTMs motivate a complexity class, analogous to Dtime:

$$\underline{\mathrm{Ntime}(T(n))} := \{L \subseteq \{0,1\}^* \mid L \text{ is decided by a NDTM } M, \ \mathrm{time}_M(n) = O(T(n))\}.$$

<u>Theorem</u>: $NP = \bigcup_{c \in \mathbb{N}} \mathrm{Ntime}(n^c)$.

<u>Proof</u>: • Let $L \in NP$ with the verifier M, & the setting: $x \in L$ iff $\exists u \in \{0,1\}^{|x|^c}, M(x,u) = 1$.

• Define an NDTM N as: On input x,

In the first $|x|^c$ many transitions δ_0 writes a 0 (δ_1 writes a 1) on the work-tape & moves right.

After N has written a $|x|^c$-bit string w, it simulates $M(x, w)$.

- Clearly, if $x \in L$ then at a certificate w, N will <u>accept</u>.
  Otherwise, N <u>rejects</u> (for all w).

$\Rightarrow L \in \text{Ntime}(n^c + n^d)$, where n^d accounts for simulating $M(x, w)$.

- Conversely, let $L \in \text{Ntime}(n^c)$ with the NDTM N of time complexity $< n^c$.

- Define a <u>verifier</u> TM M, that on input x & $u \in \{0,1\}^{|x|^c}$:
  Simulate N on x, using the transition given by u, for each step.

- Clearly, $L \in NP$. $\square$

Satisfiability

- Now we present a problem, that is the "hardest" in all of NP!

<u>Defn</u>: SAT := $\{\phi \mid \phi$ is a boolean formula in CNF, ϕ is satisfiable$\}$.

- I.e. the <u>formula</u> $\phi(x_1, .., x_n)$ has an expression $\bigwedge_i \underbrace{(\bigvee_j v_{ij})}_{\text{clause}}$, where $v_{ij} \in \{x_1, .., x_n, \bar{x}_1, .., \bar{x}_n\}$ is a <u>literal</u>.

- ϕ is called <u>satisfiable</u> if $\exists x \in \{0,1\}^n$ s.t. $\phi(x) = 1$.
- Eg. $(x_1 \vee \bar{x}_2) \wedge \bar{x}_1 \in$ SAT,

  $x_1 \wedge \bar{x}_1 \notin$ SAT.

- <u>Lemma 1</u>: SAT $\in$ NP.

<u>Pf</u>: • Accept a boolean formula $\phi(x_1, .., x_n)$ iff $\exists x \in \{0,1\}^n$, $\phi(x) = 1$. Is easy to verify. □

- Lemma 2: Let $L \in NP$. Then, L can be "reduced" to SAT in det. poly-time.

I.e. $\exists$ poly-time TM N that on input x outputs $N(x)$ s.t.

$x \in L$ iff $N(x) \in SAT$.

Proof:

- As $L \in NP$, there is a poly-time TM M (verifier) s.t.

  $x \in L$ iff $\exists u \in \{0,1\}^{|x|^c}$, $M(x,u)=1$.

- Say, M takes $< T$ steps to halt on (x,u).

- Idea: Capture the steps of $M(x,u)$ by a boolean formula.

- With each configuration C we associate a bunch of variables:

$$[\underset{\text{state}}{s(C)}, \underset{\substack{\text{head-position}\\\text{(input-tape)}}}{p'(C)}, \underset{\substack{\text{head}\\\text{(work-tape)}}}{p(C)}, \underbrace{a'_0, \ldots, a'_{T-1}}_{\substack{\text{input-}\\\text{tape}}}, \underbrace{a_0, \ldots, a_{T-1}}_{\substack{\text{work-tape}\\\text{string}}}].$$

• Final formula $\phi_{x,u}$ looks like :
$\text{start}(C_1, x, u) \wedge \text{compute}(C_1, C_2) \wedge$
$\text{stop}(C_2)$.

start(C_1, x, u) : asserts the start configuration,
$$s(C_1) = q_s \wedge p'(C_1) = p(C_1) = 0 \wedge$$
$$a'_0 \cdots a'_{T-1} = xu \wedge a_0 \cdots a_{T-1} = \triangleright \square \cdots \square .$$

stop(C_2) : asserts that M stops & outputs 1,
$$s(C_2) = q_f \wedge p(C_2) = 0 \wedge$$
$$a_0 \cdots a_{T-1} = \triangleright 1 \square \cdots \square .$$

compute(C_1, C_2) : asserts that there is a configuration sequence $\langle g_0, \ldots, g_{T-1} \rangle$ of M starting from C_1 & ending at C_2,
$$g_0 = C_1 \wedge g_{T-1} = C_2 \wedge$$
$$(\forall i < T) \Big\{ \bigvee_{I \in \delta_M} \text{step}_I (g_i, g_{i+1}) \Big\} .$$

there are only $O(1)$ many I's

$\underline{step_I(C_3, C_4)}$: asserts that there is a step from the config C_3 to C_4 following the transition $I: (s, b_1, b_2) \to (s', b_2', \varepsilon_1, \varepsilon_2)$.

- For $(\varepsilon_1, \varepsilon_2) = (S, S)$ we have it as:

$$s(C_3) = s \wedge s(C_4) = s' \wedge \exists k', k \, (p'(C_3) = p'(C_4) = k' \wedge p(C_3) = p(C_4) = k \wedge$$
$$a_k(C_3) = b_2 \wedge a_k(C_4) = b_2' \wedge$$
$$a'_{k'}(C_3) = b_1 \wedge$$
$$\langle \text{rest of } \bar{a}, \bar{a}' \text{ unchanged} \rangle \,).$$

- Similarly, for other ε_1, ε_2.

Remarks about the above formula:

(1) Note that n is fixed but u is <u>free</u>.

(2) '=' can be expressed as CNF:

$$a_1 = a_2 \text{ iff } (\bar{a}_1 \vee a_2) \wedge (a_1 \vee \bar{a}_2).$$

(3) $(\forall i < T)$ can be expressed as $\wedge$'s.

(4) Writing $\exists k(--)$ as CNF is tricky! We

(k, k' in general depend on x, u & step i)
assume that M is an <u>oblivious TM</u>, i.e. the head-position only depends on $|x|$. So, k is known as a fn. of step i, & we do not need the quantifier $\exists k$.

(5) $\phi_x(u) \in SAT$ iff $x \in L$.

(6) $|\phi_x(u)| = O(T^2)$.

- This finishes the proof of $L \leq_p SAT$. $\square$

- In fact, a CNF formula ϕ can be reduced to another formula $\psi := \bigwedge_i (v_{i1} \vee v_{i2} \vee v_{i3})$ s.t. $\psi \in SAT$ iff $\phi \in SAT$.

<u>Proof sketch</u>: Convert a clause with more than 3 literals, eg. $(x_1 \vee \bar{x}_2 \vee x_3 \vee \bar{x}_4)$, to $(x_1 \vee \bar{x}_2 \vee z) \wedge (\bar{z} \vee x_3 \vee \bar{x}_4)$. $\square$