Running time of a TM M.

- Let M be a total TM, running time of M is a function $f: \mathbb{N} \to \mathbb{R}^+$ where $f(n)$ is the maximum number of steps that M takes on any input of length n.

  M runs in time $f(n)$ where n is the length of the input.

Asymptotic upper bound

Let $f, g: \mathbb{N} \to \mathbb{R}^+$. We say that $f(n) = O(g(n))$ if $\exists c$ and n_0 s.t $\forall n \geq n_0$ $f(n) \leq c\, g(n)$

$f(n) = O(g(n))$: $g(n)$ is the asymptotic upper bound.

Running time of M is $t: \mathbb{N} \to \mathbb{R}^+$ if $\forall x \in \Sigma^*$, M halts in atmost $O(t(|x|))$ steps.

Example $A = \{0^k 1^k \mid k \geq 0\}$. Let $M : L(M) = A$.
↳ (the obvious M)

Running time of M

$$O(n) + O(n^2) + O(n) \approx O(n^2)$$

Let $t : \mathbb{N} \to \mathbb{R}^+$

$TIME(t(n)) = \{L \mid \exists$ a deterministic TM having running time $O(t(n))$ s.t. $L = L(M)\}$

Example: $A \in TIME(n^2)$

Is there a TM that decides A asymptotically faster?

Is $A \in TIME(t(n))$ for $t(n) = o(n^2)$?

Another Solution: $A \in TIME(n \log n)$.

1. Scan - reject if 0 appears after 1.

2. Repeat a long as some 0 or 1 remain on tape.

$O(\log n)$

3. Scan. Check if total number of 0's and 1's on tape is even or odd. if odd, reject. $O(n)$

4. Scan. replace every other 0 with ⊔
Replace every other 1 with ⊔ $O(n)$

5. if no 0's and 1's remain then accept.

Complexity class P.

$$P = \bigcup_k TIME(n^k)$$

P is the class of languages that are decidable in polynomial time on a deterministic single tape TM.

Example.

PATH: Given a directed graph G along with two nodes u and v, determine if a path exists from u to v

PATH $= \{G\#u\#v \mid G$ is a directed graph that has a path from u to $v\}$

PATH $\in$ P.

CFL-membership $= \{G\#x \mid G$ is a CFG and $x \in L(G)\}$.

CFL-memership $\in$ P.

$SAT = \{ \alpha \mid \exists$ a valuation v s.t $v \models \alpha \}$.

$HPATH = \{ G\#s\#t \mid G$ is a directed graph with Hamiltonian path from s to $t \}$.

Non-deterministic Turing machine.

$$\delta : Q \times \Gamma \to P(Q \times \Gamma \times \{L, R\})$$

$NTIME(t(n)) = \{ L \mid \exists$ a non-deterministic TM M running in time $t(n)$ s.t $L = L(M) \}$

$$NP = \bigcup_k NTIME(n^k).$$

NP - class of languages that have a polynomial time verifier.

A verifier for a language L is an algorithm V where

$$L = \{ w \mid V \text{ accepts } w\#c \text{ for some } c \}$$

A polynomial time verifier runs in time polynomial in the length of w.

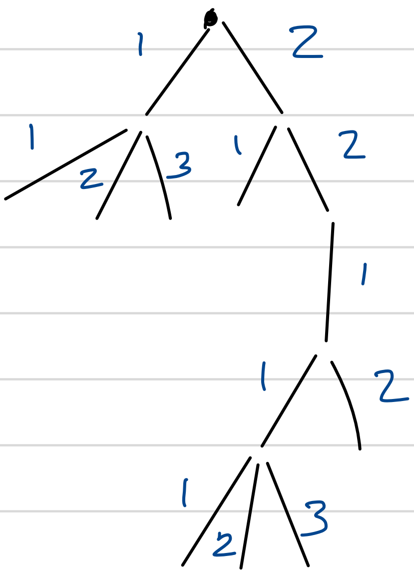
P – class of languages for which membership can be decided efficiently.

NP – class of languages for which membership can be verified efficiently.

$\delta : Q \times \Gamma \to P(Q \times \Gamma \times \{L, R\})$

Theorem. For every non-deterministic TM there is an equivalent deterministic TM.

Run of a NTM N



DTM M