Complexity Classes

Definition 1. [Complexity Class] *Set of all languages decided by some multi-string TM M operating in either deterministic or nondeterministic mode, and such that, for any input x, M consumes at most $f(|x|)$ units of the specified resource (time or space.), where f is a* proper complexity function *from the nonnegative integers to nonnegative integers.*

Proper Complexity Function

Definition 2. [Proper Complexity Function] *A function f from nonnegative integers to nonnegative integers is a proper complexity function if:*

1. f is nondecreasing.

2. There is a k-string TM $M_f = (K, \Sigma, \delta, s)$ with input/output such that, for any integer n, and any input x of length n,

$$(s, \triangleright, x, \triangleright, \epsilon, \ldots, \triangleright, \epsilon) \rightarrow^{M_f^t}$$

$$(h, x, \triangleright, \sqcup^{j_2}, \sqcup^{j_3}, \ldots, \triangleright, \sqcup^{j_{k-1}}, \triangleright, \#^{f(|x|)})$$

such that $t = O(n + f(n))$ and $j_i = O(f(|x|)$ for $i = 2, \ldots, k-1$, with t and j_i's depending only on n.

Examples: n, n^2, $\lceil \log n \rceil$, 2^n, $\sqrt{n}$.

Precise Turing Machines

Definition 3. [Precise Turing Machines] *A TM (or NTM) M (with I/O or not) is precise if there are functions f and g such that for every $n \geq 0$, for every input x of length n, and for every computation of M, M halts after precisely $f(n)$ steps, and all its strings, except for the first and last are at halting of length precisely $g(n)$.*

Lemma 1. *Suppose that a TM M decides a language L within time (or space) $f(n)$, where f is a proper function. Then there is a precise TM M' which decides the same language in time (or space) $O(f(n))$ (and hence $f(n)$).*

(Deterministic) Time Complexity

The time complexity of a TM M on a input is the number of steps (transitions) to halting.

We say M operates **within time bound** $f(n)$ if, for any string x, the time required by M on x is at most $f(|x|)$.

If $L \subset (\Sigma - \{\sqcup\})^*$ is decided by a **multistring** TM operating in time $f(n)$, then we say $L \in TIME(f(n))$, i.e., belongs to complexity class $TIME(f(n))$.

Example: PALINDROME $\in TIME(O(n))$.

$P = TIME(n^k) = \cup_{j>0} TIME(n^j)$

$EXP = TIME(2^{n^k})$

(Deterministic) Space Complexity

Definition 4. *Suppose that, for a k-string TM and an input x:*

$$(s, \triangleright, x, \ldots, \triangleright, \epsilon) \quad \rightarrow^{M^*} \quad (H, w_1, u_1, \ldots, w_k, u_k)$$

where $H = \{h, yes, no\}$. Then the space required by M on input x is $\sum_{i=1}^{k} |w_i u_i|$.

If M is a TM with I/O, then space required is $\sum_{i=2}^{k-1} |w_i u_i|$.

A TM M operates within space bound $f(n)$, if for any input x, M requires space at most $f(|x|)$.

Let L be a language. If there is a TM with I/O that can decide L and operates within space bound $f(n)$ we say $L \in SPACE(f(n))$.

Example: PALINDROMES $\in SPACE(\log n) = L$.

$PSPACE = \cup_{j>0} SPACE(n^j)$

NTIME

The set of languages decided by NTMs within time f is called $NTIME(f(n))$.

NP is the set of languages that can be a decided by a NTM in polynomial time.

$NP = \cup_{j>0} NTIME(n^j)$

$P \subseteq NP$.

Example: HAMILTONIAN CYCLE $\in NP$.

Can be decided by a (2-string) NTM in $O(n^2)$ time.

Space Complexity of NTMs

Definition 5. *Given an k-string NTM with input and output $N = (K, \Sigma, \Delta, s)$ we say that N decides language L within space $f(n)$ if N decides L and moreover, for any $x \in (\Sigma - \{\sqcup\})^*$, if $(s, \triangleright, x, \ldots, \triangleright, \epsilon) \to^{N^*} (q, w_1, u_1, \ldots, w_k, u_k)$, then*

$\sum_{j=2}^{k-1} |w_i u_i| \leq f(|x|)$.

Let L be a language. If there is a NTM with I/O that can decide L and operates within space bound $f(n)$ we say $L \in NSPACE(f(n))$.

Example: REACHABILITY $\in NSPACE(O(\log n))$.

$NPSACE = \cup_{j>0} NSPACE(n^j)$

(Deterministic) Hierarchy Theorems

Theorem 1. [Time Hierarchy Theorem] *If $f(n) \geq n$ is a proper complexity function, then the class $TIME(f(n))$ is strictly contained within $TIME((f(2n+1)^3)$.*

Corollary 1. *P is a proper subset of EXP.*

Proof: $P \subseteq TIME(2^n) \subseteq EXP$.

$TIME(2^n) \subset TIME((2^{2n+1})^3) \subseteq TIME(2^{n^2}) \subset EXP$.

Theorem 2. [Space Hierarchy Theorem] *If $f(n)$ is a proper complexity function, then the class $SPACE(f(n))$ is strictly contained within $SPACE((f(n) \log f(n))$.*

Proof of Time Hierarchy Theorem

Definition 6. *Let $f(n) \geq n$ be a proper complexity function, and define H_f to be the following time-bounded version of the HALTING language H:*

$H_f = \{M; x : M$ *accepts input* x *after at most*

$f(|x|)$ *steps* $\}$

where M ranges over all descriptions of multi-string TMs.

Lemma 2. $H_f \notin TIME(f(\lfloor \frac{n}{2} \rfloor))$.

Proof: By contradiction. Assume that there is a TM M_{H_f} that decides H_f in time $f(\lfloor \frac{n}{2} \rfloor)$.

Construct the following machine D_f:

$D_f(M):$ if $M_{H_f}(M, M) = yes$ then no else yes

D_f on input M runs in time $f(\lfloor \frac{2n+1}{2} \rfloor) = f(n)$.

What about $D_f(D_f)$?

If $D_f(D_f) = yes$ then $M_{H_f}(D_f, D_f) = no$, i.e., $(D_f; D_f) \notin H_f$.

Thus D_f fails to accept its own description in $f(n)$ steps — implies $D_f(D_f) = no$.

Similarly, $D_f(D_f) = no$ implies $D_f(D_f) = yes$.

Contradiction in both cases.