Time Complexity (Chapters 7.1, 7.2)

Definition:

Let $f, g: S \to \mathbb{N}$ be two functions, then we say "f is dominated by g", written $f \preccurlyeq g$ if $\exists c > 0 \forall s \in S: f(s) \leq c \cdot g(s)$.

Also: "g dominates f".

f, g are co-dominant if $f \preccurlyeq g, g \preccurlyeq f$.

g strictly dominates f if $f \preccurlyeq g \wedge \neg(g \sim f)$ – the "~" denotes co-dominance.

Properties:

- Reflexivity: $f \preccurlyeq f$ – simply choose $c = 1$.
- Transitivity: $f \preccurlyeq g, g \preccurlyeq h \Rightarrow f \preccurlyeq h$

  Proof: $\forall s[f(s) \leq c \cdot g(s)] \wedge \forall s[g(s) \leq d \cdot h(s)] \Rightarrow \forall s[f(s) \leq c \cdot d \cdot h(s)]$ where $c \cdot d > 0$.
- **Not a symmetric, not a total order**: for instance, sin and cos (on $S = \mathbb{R}$).

Theorem 2.5.2

Let $f, f_1, f_2, g, g_1, g_2: S \to \mathbb{R}^{\geq 0}$, and let $c > 0$, then:

i. $f \sim c \cdot f$:

We need to show $f \preccurlyeq c \cdot f$ and $c \cdot f \preccurlyeq f$, so we need to find a $d > 0$ such that $\forall s \in S: f(s) \leq d \cdot c \cdot f(s)$ – we can set $d = \frac{1}{c}$ and then there's equality. Since $c > 0$ then $d = \frac{1}{c}$ is defined.

Next, let $e > 0$ such that $c \cdot f(s) \leq w \cdot f(s)$ – we can choose $e \coloneqq c > 0$.

ii. $f_1 \preccurlyeq f_2 \wedge g_1 \preccurlyeq g_2 \Rightarrow f_1 + g_1 \preccurlyeq f_2 + g_2, f_1 \cdot g_1 \preccurlyeq f_2 \cdot g_2$

Proof:

$f_1 \preccurlyeq f_2 \Rightarrow \exists c > 0: \forall s\ f_1(s) \leq c \cdot f_2(s)$

$g_1 \preccurlyeq g_2 \Rightarrow \exists d > 0: \forall s\ g_1(s) \leq d \cdot g_2(s)$

Adding them together:

$f_1(s) + g_1(s) \leq c \cdot f_2(s) + d \cdot g_2(s) \leq e\big(f_2(s) + g_2(s)\big)$ – how do we choose e? We can choose $e \coloneqq \max\{c, d\}$, and it will satisfy the last $\leq$. For product we can choose $e = c \cdot d$ to derive the same result.

iv. $\max(f, g) \sim f + g$:

- First direction: $\forall s\ \max(f, g)(s) \leq 1 \cdot (f + g)(s)$
- Second direction: $\forall s\ f(s) + g(s) \leq 2 \cdot \max(f, g)(s)$ since $f \leq \max(f, g)$ and $f \leq \max(f, g)$

v. $1 \preccurlyeq f \wedge 1 \preccurlyeq g \Rightarrow f + g \preccurlyeq f \cdot g$ where 1 is the constant function 1.

$\forall s\ 1 \leq c \cdot f(s) \Rightarrow \frac{1}{c} \leq f(s)$ and since $c > 0$ then $\frac{1}{c}$ is defined and > 0 as well. it means that f is always greater than some positive constant.

The proof takes $f \preccurlyeq f, 1 \preccurlyeq g$ and derives $f \preccurlyeq f \cdot g$, from previous sections. The same is shown for $f \preccurlyeq f \cdot g$ and then by previous sections $f + g \preccurlyeq f \cdot g$.

viii. If $S_1, S_2 \subset S$, $f \preccurlyeq g$ on S_1 and $f \preccurlyeq g$ on S_2 then $f \preccurlyeq g$ on $S_1 \cup S_2$.

If $|S| = 1$ and $f, g: S \to \mathbb{R}^{>0}$ positive functions then $f \sim g$ on S:

$S = \{s\}, f(s), g(s) > 0 \Rightarrow f(s) \leq \frac{f(s)}{g(s)} \cdot g(s)$ where $\frac{f(s)}{g(s)}$ is a constant since there is only one $s \in S$.

On any finite set, any two positive functions are co-dominant.

The finiteness is important for the theorem. For instance, take $S = \mathbb{N}, f \equiv 1, g(n) = n$, then for any constant $c > 0$ g strictly dominates f so the two are not equivalent.

Theorem

Let $k > 0$ be a positive integer, then $1^k + 2^k + 3^k + \cdots + n^k \sim n^{k+1}$.

Proof:

Consider $f(x) = x^k$. Looking at the graph of the function, we can enclose rectangles between $[0,1], [1,2], [2,3]$ … with the maximum value of f in that rectangle, i.e. the area of the 1st is 1^k, of the second is 2^k and so on $\Rightarrow$

$$1^k + \cdots + n^k \geq \int_0^n x^k \partial x = \frac{1}{k+1} x^{k+1}|_0^n = \frac{n^{k+1}}{k+1} \Rightarrow$$

$$n^{k+1} \leq (k+1)(1^k + \cdots + n^k) \Rightarrow n^{k+1} \preccurlyeq 1^k + \cdots n^k$$

The other direction is taking the area of the rectangle below the function.

Exercise 2.5.5

$\forall a > 0 \ \ln x \preccurlyeq x^a$ on $\mathbb{R}^{\geq 1}$:

Proof:

$x^a = e^{a \ln x} \underset{e^y = \sum_{k=0}^{\infty} \frac{y^k}{k!}}{\geq} a \ln x \Rightarrow \ln x \leq \frac{1}{a} x^a$ and $\frac{1}{a}$ is our positive constant.

Computing time functions

The constant c makes step-count functions machine-independent, because every basic operation on machine M_1 can be simulated by at most c basic operations on machine M_2, so $t_1(w) \leq c \cdot t_2(w)$ and vice-versa, so $t_1 \sim t_2$.

Set of inputs $S = \bigcup_{i=1}^{\infty} S_i$ such that S_1 is a finite set for all i.

- Maximum computing time function: $t_A^+(i) \coloneqq \max_{s \in S_i} t_A(s)$
- Minimum computing time function: $t_A^-(i) \coloneqq \min_{s \in S_i} t_A(s)$
- Average: $t_A^*(i) \coloneqq \left(\sum_{s \in S_i} t_A(s)\right) / |S_i|$

Let $L = \{0^k 1^k \mid k > 0\}$

$n :=$ the length of the input.

1) A TM that decides it:

- Scan: n steps
- $\frac{n}{2}$ passes over $\leq n$? $\sim n^2$

The total: $\sim n^2$

2)

- Scan: n steps.
- Scan: total 0's, 1's are of an even number
- Mark every other 0, every other 1, and loop until all marked.

The total: $n \lg n$

<u>Note</u>: $\log_\alpha \sim \log_\beta$

3) 2 tapes: time $\sim n$

Theorem:

Simulating a k-tape TM with a single tape TM, the computing time is at most squared. Formally:

Any computation on a k-tape TM that takes time $t(n)$ where n is the length of the input, can be simulated on a single tape TM in time $t(n)^2$.

The idea of the proof is that for each of the $t(n)$ steps of the k-tape machine, we need to do at most $k \cdot t(n)$ steps, concluding to a $t(n)^2$ time.

Simulating non-deterministic $t(n)$ will become a deterministic $b^{t(n)}$.

The book denotes "$2^{O(t(n))}$", but O is a set. The correct way is: $2^{c \cdot t(n)} = (2^c)^{t(n)}$ where $b = 2^c$.

Note that the base matters: $2^n \prec 4^n$ (strictly dominated), so the c matters in the power.

We simulated non-determinism by using a 3-tape TM with:

- Non-writable input tape.
- Simulation tape.
- Bookkeeping tape for enumerating over all "choices" in the non-determinism tree.

But, this computation is very expensive – exponential.

<u>Definition</u>:

$P = \bigcup_{k=1}^{\infty} TIME(n^k)$, where $TIME(n^k)$ is the class of problems that can be solved by a deterministic Turing machine.

Note that the number of tapes doesn't matter because it is time-class invariant (at most squares the time).

Example:

Let G be a graph, and s, t nodes in the graph. Is there a path from s to t?

Brute-force is exponential. Of course it can be easily done in polynomial time.