Heuristic Optimization – Homework 3

Maximilian Brehm & Clemens Frahnow

4 Exercise 4

4.1 Task a

To show:

Let $f: \mathbb{N} \to \mathbb{R}^+$. $\left(\exists c \in (0,1) \exists n_0 > 0 \, \forall n > n_0 : f(n+1) \leq cf(n)\right) \Rightarrow f = 2^{-\Omega(n)}$.

Prove:

We define a helper function *g*: $g(n) = -\log_2(f(n))$ and therefore $f(n) = 2^{-g(n)}$.

We assume $\exists c \in (0,1) \exists n_0 > 0: \quad f(n+1) \leq c \cdot f(n) \quad \Leftrightarrow \quad 2^{-g(n+1)} \leq c \cdot 2^{-g(n)}$

$\Leftrightarrow \quad -g(n+1) \leq \log_2(c) - g(n)$ (take the logarithm)

$\Leftrightarrow \quad g(n+1) \geq g(n) - \log_2(c)$

We define a new variable $d = -\log_2 c$. Since $c \in (0,1)$, $d > 0$.

$\Leftrightarrow \quad g(n+1) \geq g(n) + d$

With this formula we can always find a lower bound for *g(n+1)* using *g(n)*:

$g(n+1) \geq g(n) + d$ and $g(n+2) \geq g(n+1) + d$.

Both combined: $\Rightarrow \quad g(n+2) \geq (g(n) + d) + d$

$\Leftrightarrow \quad g(n+2) \geq g(n) + 2 \cdot d$

$\Rightarrow \quad g(n+3) \geq g(n) + 3 \cdot d$ and so on.

In general: $g(n+x) \geq g(n) + x \cdot d$ for $x \geq 0$

$\Leftrightarrow \quad g(n_0 + x) \geq g(n_0) + x \cdot d$ (using n_0 as *n*)

$\Leftrightarrow \quad g(n) \geq g(n_0) + d \cdot (n - n_0)$ ($n = n_0 + x$)

$\Leftrightarrow \quad g(n) \geq dn + g(n_0) - dn_0$

If we can find a *y*>0, so that $y \cdot g(n) \geq n$, then $g(n) = \Omega(n)$ (by definition of $\Omega(n)$)

Because of $\frac{n}{g(n)} \leq \frac{n}{dn + g(n_0) - dn_0} = \frac{1}{d} \cdot \frac{n}{n + \frac{1}{d} \cdot (g(n_0) - dn_0)} \leq \frac{1}{d}$

we set *y*: $y = \frac{1}{d} \geq \frac{n}{g(n)}$, and therefore $y \cdot g(n) \geq n$

$$\Rightarrow g(n) = \Omega(n)$$

$$\boxed{\Leftrightarrow \quad f = 2^{-\Omega(n)}}$$

4.2 Task b

To show:

$$\exists f = 2^{-\Omega(n)} : !\exists c \in (0,1) : \exists n_0 \forall n > n_0 : \quad f(n+1) \leq c \cdot f(n)$$

Proof:

Let f be the following function:

$$f(n) = \begin{cases} 2^{-n}, \text{if n is even} \\ 2^{-2\text{n}}, \text{if n is odd} \end{cases}$$

n and *2n* are in $\Omega(n)$, so $f(n) = 2^{-\Omega(n)}$

We show that $!\exists c \in (0,1) : \exists n_0 \forall n > n_0 : \quad f(n+1) \leq c \cdot f(n)$ by contrapositive:

We assume $\exists c \in (0,1) : \exists n_0 \forall n > n_0 : \quad f(n+1) \leq c \cdot f(n)$.

Therefore for all odd *n* (*n+1* is even) with $n > n_0$: $f(n+1) \leq c \cdot f(n)$

$$\Rightarrow 2^{-n+1} \leq c \cdot 2^{-2\text{n}}$$

$$\Leftrightarrow n+1 \leq \log_2 c$$

$$\Leftrightarrow n \leq \log_2 c - 1$$

Since *n* cannot stay smaller than a constant with rising *n*, this must be false.

We concluded this false statement from the assumption that

$\exists c \in (0,1) : \exists n_0 \forall n > n_0 : \quad f(n+1) \leq c \cdot f(n)$.

Therefore $!\exists c \in (0,1) : \exists n_0 \forall n > n_0 : \quad f(n+1) \leq c \cdot f(n)$.

With f, we found a function to fulfill

$$\boxed{\exists f = 2^{-\Omega(n)} : !\exists c \in (0,1) : \exists n_0 \forall n > n_0 : \quad f(n+1) \leq c \cdot f(n)}$$

4.3 Task c

To show:

$$\left(\exists n_0 > 0 \,\forall n > n_0:\ f(n) \leq g(n)\right) \ \Rightarrow\ 2^f = O(2^g)$$

Proof:

$$\exists n_0 > 0 \,\forall n > n_0:\ f(n) \leq g(n)$$

$$\Leftrightarrow\ \exists n_0 > 0 \,\forall n > n_0:\ 2^{f(n)} \leq 2^{g(n)}$$

$$\Rightarrow\ \exists c > 0 \,\exists n_0 > 0 \,\forall n > n_0:\ 2^{f(n)} \leq c \cdot 2^{g(n)}$$

$$\boxed{\Leftrightarrow\ 2^f = O(2^g)}$$

4.4 Task d

Given

$$f, g: \mathbb{N} \rightarrow \mathbb{R}^+$$

To show:

$$2^f = O(2^g) \Rightarrow f = O(g) \text{ is wrong}$$

Proof by counter example:

$$f(n) = \frac{1}{n},\ g(n) = \frac{1}{n^2}$$

$$\exists n_0 > 0 \forall n > n_0: \quad n - 1 < n^2$$

$$\Leftrightarrow \exists n_0 > 0 \forall n > n_0: \quad \frac{n-1}{n^2} < 1$$

$$\Leftrightarrow \exists n_0 > 0 \forall n > n_0: \quad 2^{\frac{n-1}{n^2}} \le 2^1$$

$$\Leftrightarrow \exists n_0 > 0 \forall n > n_0: \quad 2^{\frac{1}{n} - \frac{1}{n^2}} \le 2$$

$$\Leftrightarrow \exists n_0 > 0 \forall n > n_0: \quad 2^{\frac{1}{n}} \le 2 \cdot 2^{\frac{1}{n^2}}$$

$$\Rightarrow \exists c > 0 \exists n_0 > 0 \forall n > n_0: \quad 2^{\frac{1}{n}} \le c \cdot 2^{\frac{1}{n^2}}$$

$$\Leftrightarrow 2^f = O(2^g)$$

Assuming $f = O(g)$:

$$\exists c > 0 \exists n_0 > 0 \forall n > n_0: \quad \frac{1}{n} \le c \cdot \frac{1}{n^2}$$

$$\Leftrightarrow \quad \exists c > 0 \exists n_0 > 0 \forall n > n_0: \quad n \le c$$

false.

We concluded this false statement from the assumption that $f = O(g)$.

Therefore $\boxed{f \neq O(g)}$.

To get $f = O(g)$ we add the following constraint: $\lim_{n \to \infty} g(n) \neq 0 \ \wedge \ \forall n > n_0: \ g(n) \neq 0$

Assuming $2^f = O(2^g)$:

$$\Leftrightarrow \quad \exists c > 0 \exists n_0 > 0 \forall n > n_0: \quad 2^{f(n)} \ \le \ c \cdot 2^{g(n)}$$

$$\Leftrightarrow \quad \exists c > 0 \exists n_0 > 0 \forall n > n_0: \quad f(n) \ \le \ \log_2 c + g(n)$$

$$\Leftrightarrow \quad \exists c > 0 \exists n_0 > 0 \forall n > n_0: \quad f(n) \ \le \ g(n) \cdot \left(\frac{\log_2 c}{g(n)} + 1 \right)$$

To show that $f = O(g)$, we just have to prove, that

$$\exists a > 0 \exists n_0 > 0 \forall n > n_0 : \quad f(n) \le g(n) \cdot a$$ (definition of *O(g)*)

So if there is a constant *a>0*, so that $\frac{\log_2 c}{g(n)} + 1 \le a$, then

$$\Leftrightarrow \quad \exists a > 0 \exists c > 0 \exists n_0 > 0 \forall n > n_0 : \quad f(n) \le g(n) \cdot \left(\frac{\log_2 c}{g(n)} + 1 \right) \le g(n) \cdot a$$

$$\Leftrightarrow f = O(g) \ .$$

Since $\lim_{n \to \infty} g(n) \ne 0$ and $\forall n > n_0 : \ g(n) \ne 0$, there is a constant *d>0*, so that

$$\forall n > n_0 : \quad g(n) \ge d$$

$$\Leftrightarrow \quad \forall n > n_0 : \quad \frac{\log_2 c}{g(n)} + 1 \le \frac{\log_2 c}{d} + 1$$

Therefore there is a constant *a*, such that:

$$\Leftrightarrow \quad \exists a > 0 \forall n > n_0 : \quad \frac{\log_2 c}{g(n)} + 1 \le \frac{\log_2 c}{d} + 1 = a$$

$$\Rightarrow \quad \exists a > 0 \forall n > n_0 : \quad \frac{\log_2 c}{g(n)} + 1 \le a$$

$$\Leftrightarrow \quad \exists a > 0 \forall n > n_0 : \quad f(n) \le g(n) \cdot \left(\frac{\log_2 c}{g(n)} + 1 \right) \le g(n) \cdot a$$

$$\Leftrightarrow \quad \exists a > 0 \exists n_0 > 0 \forall n > n_0 : \quad f(n) \le g(n) \cdot a$$

$$\boxed{\Leftrightarrow \quad f = O(g)}$$

Therefore when applying the constraint $\lim_{n \to \infty} g(n) \ne 0 \ \wedge \ \forall n > n_0 : \ g(n) \ne 0$,

then $2^f = O(2^g) \Rightarrow f = O(g)$ is true.

4.5 Task e

To show:

$$\exists f, g: \mathbb{N} \rightarrow \mathbb{R}^+ : f = O(g) \land 2^f \neq O(2^g)$$

Proof:

$$f(n) = 2 \cdot n, \; g(n) = n$$

$$\exists n_0 > 0 \, \forall n > n_0: \quad 2 \cdot n < 3 \cdot n$$

$$\Leftrightarrow \exists n_0 > 0 \, \forall n > n_0: \quad f(n) < 3 \cdot g(n)$$

$$\Rightarrow \exists c > 0 \, \exists n_0 > 0 \, \forall n > n_0: \quad f(n) < c \cdot g(n)$$

$$\boxed{\Leftrightarrow f = O(g)}$$

Assuming $2^f = O(2^g)$:

$$\exists c > 0 \, \exists n_0 > 0 \, \forall n > n_0: \quad 2^{2n} \leq c \cdot 2^n$$

$$\Leftrightarrow \exists c > 0 \, \exists n_0 > 0 \, \forall n > n_0: \quad 2^n \leq c$$

$$\Leftrightarrow \exists c > 0 \, \exists n_0 > 0 \, \forall n > n_0: \quad n \leq \log_2 c$$

false.

We concluded this false statement from the assumption that $2^f = O(2^g)$.

Therefore $\boxed{2^f \neq O(2^g)}$.