Week 5: Analyzing Algorithm Running Time

Michael Zhou

2023-02-28

1 Asymptotic growth

When we write $f : A \to B$, we are saying the function maps elements of A to elements of B. We will mainly be concerned about mapping natural numbers to the nonnegative real numbers.[1] Or in other words: $f : \mathbb{N} \to \mathbb{R}^{\geq 0}$. We will only care about long term (**asymptotic**) growth.

[1] This is the domain and range that arises for algorithm analysis since an algorithm cannot take negative time to run...

Definition 5.1. Let $f, g : \mathbb{N} \to \mathbb{R}^{\geq 0}$. We say that g is **absolutely dominated by** f if and only if for all $n \in \mathbb{N}, g(n) \leq f(n)$.

Example 5.1. Let $f(n) = n^2$ and $g(n) = n$. Prove that g is absolutely dominated by f.

Translation. $\forall n \in \mathbb{N}, g(n) \leq f(n)$.

Proof. Let $n \in \mathbb{N}$. We want to show that $n \leq n^2$.

Case 1. Assume $n = 0$, then $n^2 = n = 0$, so the statement is True.

Case 2. Assume $n \geq 1$, then we can multiple both sides of the inequality by n, which is $n^2 \geq n$. Thus this is True.

$\blacksquare$

Definition 5.2. Let $f, g : \mathbb{N} \to \mathbb{R}^{\geq 0}$. We say that g **is dominated by** f **up to a constant factor** if and only if there exists a postive real number c such that for all $n \in \mathbb{N}$, $g(n) \leq c \cdot f(n)$.

Example 5.2. Let $f(n) = n^n$ and $g(n) = 2n$. Prove that g is dominated by f up to constant factor.[2]

[2] At $n = 1$, $2n > n^2$.

Translation. $\exists c \in \mathbb{R}^+, \forall n \in \mathbb{N}, g(n) \leq c \cdot f(n)$.

Proof. Let $c = 2$, and let $n \in \mathbb{N}$. We want to prove that $g(n) \leq c\dot{f}(n)$, or in other words, $2n \leq 2n^2$.

Case 1. Assume $n = 0$, then $2n = 2n^2 = 0$. So this case is True.

Case 2. Assume $n \geq 1$. Taking the inequality, we can multiply both sides by $2n$, and we get $2n^2 \geq 2n$. Thus this is True.

$\blacksquare$

These two definitions are still too restrictive for runtimes, where constant factors do not matter. Consider $f(n) = n^2$ and $g(n) = n + 90$, no matter how much we scale $f(n)$, $f(0)$ will always be smaller than $g(0)$. So, we cannot say that $f(n)$ is dominated by $g(n)$ up to a constant factor.

But it is certainly possible to find a constant factor at any value other than $n = 0$, this brings us to the third definition:

Definition 5.3. Let $f, g : \mathbb{N} \to \mathbb{R}^{\geq 0}$. We say that g **is eventually dominated by** f if and only if there exists $n_0 \in \mathbb{R}^+$ such that $\forall n \in \mathbb{N}$, if $n \geq n_0$ then $g(n) \leq f(n)$.

Example 5.3. Let $f(n) = n^2$ and $g(n) = n + 90$. Prove that g is eventually dominated by f.

Translation. $\exists n_0 \in \mathbb{R}^+, \forall n \in \mathbb{N}, n \geq n_0 \Rightarrow g(n) \leq f(n)$.

Proof. Let $n_0 = 90$, let $n \in \mathbb{N}$, and assume $n \geq n_0$. WTS $n + 90 \leq n^2$.

$$\begin{aligned} n + 90 &\leq n + n && \text{since } n \geq 90 \\ &\leq 2n \\ &\leq n \cdot n && \text{since } n \geq 2 \\ &\leq n^2 \end{aligned}$$

■

This definition ignores small values of n, whereas the previous ignores constant factors. Our last definition will combine both of these traits:

Definition 5.4. Let $f, g : \mathbb{N} \to \mathbb{R}^{\geq 0}$. We say that g **is eventually dominated by** f **up to a constant factor** if and only if there exist $c, n_0 \in R^+$, such that for all $n \in \mathbb{N}$, if $n \geq n_0$ then $g(n) \leq c \cdot f(n)$.

In this case, we also say that g **is Big-Oh of** f, and write $g \in \mathcal{O}(f)$.

Definition 5.5. $\mathcal{O}(f)$ is defined as the *set of functions* that are eventually dominated by f up to a constant factor:

$$\mathcal{O}(f) = \{g \mid g : \mathbb{N} \to \mathbb{R}^{\geq 0}, \exists c, n_0 \in \mathbb{R}^+, \forall n \in \mathbb{N}, n \geq n_0 \Rightarrow g(n) \leq c \cdot f(n)\}$$

Example 5.4. Let $f(n) = n^3$ and $g(n) = n^3 + 100n + 5000$. Prove that $g \in \mathcal{O}(f)$.

Translation. $\exists c, n_0 \in \mathbb{R}^+, \forall n \in \mathbb{N}, n \geq n_0 \Rightarrow n^3 + 100n + 5000 \leq cn^3$.

Discussion. We can have two approaches: focus on choosing n_0, or focus on choosing c. Either way, we will divide the inequality to three smaller inequalities:

1. $n^3 \leq n^3$
2. $100n \leq n^3$
3. $5000 \leq n^3$

Proof. Let $c = 3$ and $n_0 = \sqrt[3]{5000}$. Let $n \in \mathbb{N}$, and assume that $n \geq n_0$. WTS $n^3 + 100n + 5000 \leq cn^3$.

We can first prove the three simpler inequalities:

- $n^3 \leq n^3$ (since the two quantities are equal)
- Since $n \geq n_0 \geq 10$, we know that $n^2 \geq 100$, and so $n^3 \geq 100n$.
- Since $n \geq n_0$, we know that $n^3 \geq n_0^3 = 5000$

Adding these up we get

$$n^3 + 100n + 5000 \leq n^3 + n^3 + n^3 = cn^3$$

■

2 One special case of Big-Oh: $\mathcal{O}(1)$

Consider the function $f(n) = 1$, which always outputs the value 1. Unpacking the definition of Big-Oh of f we get

$$\begin{aligned}
&g \in \mathcal{O}(f) \\
&\exists c, n_0 \in \mathbb{R}^+, \forall n \in \mathbb{N}, n \geq n_0 \Rightarrow g(n) \leq c \cdot f(n) \\
&\exists c, n_0 \in \mathbb{R}^+, \forall n \in \mathbb{N}, n \geq n_0 \Rightarrow g(n) \leq c \qquad \text{since } f(n) = 1
\end{aligned}$$

There exists a constant c such that $g(n)$ is eventually always less than or equal to c. We say that such functions g are **asymptotically bounded** with respect to their input, and write $g = \mathcal{O}(1)$.

3 Omega and Theta

Big-Oh is limited in the sense that it is not exact. Consider two functions: $g(n) = n + 1$ and $f(n) = n^{100}$. We can write $n + 10 \in \mathcal{O}(n^{100})$ but it would not be very informative, since $f(n)$ grows *much* faster than $g(n)$.

In this section, we will introduce ways to express tight bounds on the growth of a function.

Definition 5.6. Let $f, g : \mathbb{N} \to \mathbb{R}^{\geq 0}$. We say that g **is Omega of** f if and only if there exist constants $c, n_0 \in \mathbb{R}^+$ such that for all $n \in \mathbb{N}$, if $n \geq n_0$, then $g(n) \geq c \cdot f(n)$. In this case, we write $g \in \Omega(f)$.

Omega is the dual of Big-Oh. When $g \in \Omega(f)$, then f is a *lower bound* on the growth rate of g. We can now express a bound that is tight for a function's growth by combining Big-Oh and Omega: if f is asymptotically both a lower and upper bound for g, then g must grow at the same rate as f.

Definition 5.7. Let $f, g : \mathbb{N} \Rightarrow \mathbb{R}^{\geq 0}$. We say that g **is Theta of** f if and only if g is both Big-Oh of f and Omega of f. In this case, we can write $g \in \Theta(f)$, and say that f is a **tight bound** on g.

Equivalently, g is Theta of f if and only if there exist constants $c_1, c_2, n_0 \in \mathbb{R}^+$ such that for all $n \in \mathbb{N}$ if $n \geq n_0$ then $c_1 f(n) \leq g(n) \leq c_2 f(n)$.

4 Properties of Big-Oh, Omega, and Theta

4.1 Elementary functions

The following theorem tells us how to compare four different types of "elementary" functions: constant functions, logarithms, powers of n, and exponential functions.

Theorem 5.1. For all $a, b \in \mathbb{R}^+$, the following statements are True:

1. $a > 1 \wedge b > 1 \Rightarrow log_a n \in \Theta(log_b n)$
2. $a < b \Rightarrow n^a \in \mathcal{O}(n^b) \wedge n^a \notin \Omega(n^b)$
3. $a < b \Rightarrow a^n \in \mathcal{O}(b^n) \wedge a^n \notin \Omega(b^n)$
4. $a > 1 \Rightarrow 1 \in \mathcal{O}(log_a n) \wedge 1 \notin \Omega(log_a n)$
5. $log_a n \in \mathcal{O}(n^b) \wedge log_a n \notin \Omega(n^b)$

6. $b > 1 \Rightarrow n^a \in \mathcal{O}(b^n) \wedge n^a \notin \Omega(b^n)$

4.2 Basic properties

Theorem 5.2. For all $f : \mathbb{N} \Rightarrow \mathbb{R}^{\geq 0}, f \in \Theta(f)$.

Theorem 5.3. For all $f, g : \mathbb{N} \to \mathbb{R}^{\geq 0}, g \in \mathcal{O}(f)$ if and only if $f \in \Omega(g)$.

Theorem 5.4. For all $f, g, h : \mathbb{N} \to \mathbb{R}^{\geq 0}$:

- If $f \in \mathcal{O}(g)$ and $g \in \mathcal{O}(h)$, then $f \in \mathcal{O}(h)$.
- If $f \in \Omega(g)$ and $g \in \Omega(h)$, then $f \in \Omega(h)$.
- If $f \in \Theta(g)$ and $g \in \Theta(h)$, then $f \in \Theta(h)$.

4.3 Operations on functions

Definition 5.8. Let $f, g : \mathbb{N} \to \mathbb{R}^{\geq 0}$. We can define the **sum of f and g** as the function $f + g : \mathbb{N} \to \mathbb{R}^{\geq 0}$ such that

$$\forall n \in \mathbb{N}, (f + g)(n) = f(n) + g(n)$$

Theorem 5.5. For all $f, g, h : \mathbb{N} \to \mathbb{R}^{\geq 0}$, the following hold:

- If $f \in \mathcal{O}(h)$ and $g \in \mathcal{O}(h)$, then $f + g \in \mathcal{O}(h)$.
- If $f \in \Omega(h)$, then $f + g \in \Omega(h)$.
- If $f \in \Theta(h)$ and $g \in \mathcal{O}(h)$, then $f + g \in \Theta$.

$$\begin{aligned} n + 1 &= \sum_{i=1}^{k} (a_i \cdot i!) + 1 \\ \sum_{i=1}^{k} (a_i \cdot i!) + 1 &= \sum_{i=1}^{k} (a_i \cdot i!) + 1 \end{aligned}$$