An accepted abuse of the Big-Oh notation when writing recurrences[1]

1 Abuse of Notation: what does it mean?

As if the abuse $f(n) = O(g(n))$ instead of $f(n) \in O(g(n))$ was not enough, in this course We will see statements of the form

$$\text{For every } n > 0, , \quad f(n) \leq g(n) + O(h(n)) \tag{1}$$

Again recall, $O(h(n))$ is a *set* of functions. What does it mean to "add a set"!? In fact, this is most prevalent in recurrence inequalities. For instance, we see recurrence inequalities of the form

$$T(1) = O(1), \quad \text{and for all } n \geq 2, \; T(n) \leq T(n-1) + O(n) \tag{2}$$

What do these mean?

So, whenever you see a recurrence as above, what it really means is that that *there exists* a function $e(n)$, such that (a) for all $n \geq 2$, $T(n) \leq g(n) + e(n)$, and (b) $e(n) \in O(n)$.

Let us unpack what $e(n) \in O(n)$ means. It means that there exists positive constants a, b such that for all $n \geq b$, $e(n) \leq a \cdot n$. We are now going to simplify our lives a bit and see that we can actually state there exists a constant A such that

$$\forall n \geq 2, \quad e(n) \leq A \cdot n$$

How can we say this? Well, if $n \geq b$, we get this with $A = a$. Let $B = \max_{n<b} e(n)$; this is some constant. Note, that for all $2 \leq n \leq b$, $e(n) \leq B \leq B \cdot n$. Therefore, $A = \max(a, B)$ satisfies our needs.

Putting all together, the recurrence (2) actually means the existence of a constant A, B such that

$$T(1) \leq B, \quad \text{and} \quad T(n) \leq T(n-1) + A \cdot n, \qquad \text{for all } n \geq 2$$

The reason for this abuse is brevity – writing out these constants and keeping track of them is tedious. One of the nice things about the Big Oh notation is the fact that we can express things succinctly. Of course brevity comes at a cost – awareness! Be wary, especially when you are solving recurrence inequalities with Big-Oh's and Theta's floating about. Let me point out a mistake which you should never make.

Theorem 1. Warning: This is a **wrong** theorem followed by an **erroneous** proof.
Consider the following recurrence inequality:

$$T(1) = O(1); \quad \forall n > 1, \quad T(n) \leq T(n-1) + O(1)$$

The solution to the above is $T(n) = O(1)$.

Wrong Proof of Wrong Theorem 1. We proceed by induction. The predicate $P(n)$ is true if $T(n) = O(1)$

Base Case: When $n = 1$, $T(n) = O(1)$. This is given to us.

[1] *Lecture notes by Deeparnab Chakrabarty. Last modified : 19th Mar, 2022*
These have not gone through scrutiny and may contain errors. If you find any, or have any other comments, please email me at deeparnab@dartmouth.edu. Highly appreciated!

We fix a $k \geq 1$ and assume $P(1), \ldots, P(k)$ are all true. That is, $T(m) = O(1)$ for all $m \leq k$. We need to show $P(k+1)$ is true. That is, we need to show $T(k+1) = O(1)$.

Well, $T(k+1) \leq T(k) + O(1)$. Since $P(k)$ is true, we get $T(k) = O(1)$. Therefore, $T(k+1) = O(1) + O(1)$. But adding a constant with another constant, is again a constant. So, $T(k+1) = O(1)$. Hence proved. $\square$

The above theorem is wrong, because we know what the recurrence really means there exists some constants A, B such that

$$T(n) \leq \begin{cases} B & \text{if } n = 1 \\ T(n-1) + A & \text{if } n \geq 2 \end{cases}$$

what the solution to $T(n)$ is by the "opening up brackets" or the "kitty collection" method. For any $n \geq 2$,

$$\begin{aligned} T(n) &\leq T(n-1) + A \\ &\leq T(n-2) + A + A \\ &\vdots \\ &= T(1) + A(n-1) \leq An + (B - A) \quad = O(n) \end{aligned}$$

Where was the bug in the above induction proof? Whenever you see a wrong proof, you must find out where were it is wrong. And the mistake crept in due to the abuse of notation. Consider what the Theorem 1 is asserting : it says there is an (unspecified) constant $C \geq 0$ such that $T(n) \leq C$ for all n. When we try to apply induction, we must ***first agree upon this constant***. We don't know what it is, but it exists, and we stick with it. Base Case: $T(1) \leq C$ – fine. Induction Hypothesis: for all $1 \leq n < m$, we have $T(n) \leq C$ – sure. Now we need to prove $T(m) \leq C$ as well. What do we know – we know that $T(m) \leq T(m-1) + \Theta(1)$. Again what does this mean? It means there is some other unspecified constant $C' \geq 0$ such that $T(m) \leq C + C'$. But now we are in trouble – we can't say $T(m) \leq C$ unless $C' = 0$, and that cannot be assumed.