Big O notation sum rule

I understand that when adding functions, the behavior is dominated by the highest power. But what I am having trouble is understanding the proof. Could anyone help me step by step in explaining the proof behind T1(n) + T2(n) = O(max (f(n), g(n)))? Thank you very much

(asymptotics) (proof-writing)

asked Mar 8 '13 at 0:40
user65674
21 1 2

Do you mean $f(n) + g(n) = O(\max(f(n), g(n)))$? – MJD Mar 8 '13 at 1:58

I guess but doesn't it not matter? If T(n) is in O(f(n)) it doesnt necessarily have to mean that O(f(n)) is the same function does it? – user65674 Mar 8 '13 at 3:04

Well, you didn't state any relation between T_1 and T_2 on the one hand, and f and g on the other hand, and without that there's no way to say anything about them. Did you mean to say that $T_1(n) = O(f(n))$ and that $T_2(n) = O(g(n))$? – MJD Mar 8 '13 at 3:15

Yes sorry I guess I shouldve specified. I was assuming formal definition of Big O for both T(n) and T2(n) – user65674 Mar 8 '13 at 18:50

1 Answer

Given $T_1(n) = O(f(n))$ and $T_2(n) = O(g(n))$, we are to prove

$$T_1(n) + T_2(n) = O(\max(f(n), g(n)) \tag{0}$$

- Write down exactly what the first assumption says: there exists a constant C_1 and an index N_1 such that

$$|T_1(n)| \leq C_1 f(n) \quad \text{when } n \geq N_1 \tag{1}$$

- Write down exactly what the second assumption says: there exists a constant C_2 and an index N_2 such that

$$|T_2(n)| \leq C_2 g(n) \quad \text{when } n \geq N_2 \tag{2}$$

- Prepare to combine (1) and (2) by introducing $N = \max(N_1, N_2)$ and $C = \max(C_1, C_2)$.
- Add (1) and (2):

$$|T_1(n)| + |T_2(n)| \leq C_1 f(n) + C_2 g(n) \leq C(f(n) + g(n)) \quad \text{when } n \geq N \tag{3}$$

- Check that for any two real numbers a, b we have

$$a + b \leq 2\max(a, b) \tag{4}$$

- Use (4) in (3) to obtain

$$|T_1(n)| + |T_2(n)| \leq 2C\max(f(n), g(n)) \quad \text{when } n \geq N \tag{5}$$

- Conclude that (0) holds.

answered Jul 24 '13 at 0:24
40 votes
7,191 20 83

Canonical job. +1. – Rick Decker Jul 24 '13 at 0:26