# Proof of Asymptotic Analysis of Algorithm Time Complexity

Scse,Uestc,Chengdu,China,Jianghong Huang

September 13, 2023

## 1 Introduction

In the field of computer science, we encounter various algorithms that assist in solving specific practical problems. During this stage, it is common to analyze the asymptotic expression of the algorithm in terms of time and space complexity to measure its efficiency.

## 2 Question

Please prove the expression ?$\mathbf{O}\left(\mathbf{f}\left(\mathbf{n}\right)+\mathbf{g}\left(\mathbf{n}\right)\right)=\mathbf{O}\left(\max\left\{\mathbf{f}\left(\mathbf{n}\right),\mathbf{g}\left(\mathbf{n}\right)\right\}\right)$, where $f\left(n\right)$ and $g\left(n\right)$ are two functions with respect to n. For any n in natural numbers, $f(n) \geq 0$ and $g(n) \geq 0$.

## 3 Assumptions

For any $f_1(n) \in O(f(n))$, there exists positive constant $c_1$ and a natural number $n_1$ such that for any $n \geq n_1$, $0 \leq f_1(n) \leq c_1 \times f(n)$.

Similarly,it can be derived that for any $g_1(n) \in O(g(n))$,there exists positive constant $c_2$ and a natural number $n_2$ such that for any $n \geq n_2$ ,$0 \leq g_1(n) \leq c_2 \times g(n)$.

Now let's assume that $c_3 = \max\left\{c_1, c_2\right\}, c_4 \geq c_3, n_3 = \max\left\{n_1, n_2\right\}$,and $h(n) = \max\left\{f(n), g(n)\right\}$.

# 4 Proof

Based on the above assumptions, for any $n \geq n_3$, we have

$$\begin{aligned} g_1(n) + f_1(n) &\leq c_1 \times f(n) + c_2 \times g(n) \quad (1) \\ &\leq c_3 \times f(n) + c_3 \times g(n) \\ &= c_3 \times (f(n) + g(n)) \quad (2) \end{aligned}$$

According to the definition of asymptotic upper bound, when $c_4 \geq c_3$, the following expression holds.

$$0 \leq c_3 \times (f(n) + g(n)) \leq c_4 \times (f(n) + g(n)) \quad (3)$$

From expression (3), we obtain $O(f(n)+g(n))$.Therefore, for all natural numbers n, when $n \geq n_4$, we could deduce that

$$\begin{aligned} c_4 \times (f(n) + g(n)) &\leq 2 \times c_4 \times \max\{f(n), g(n)\} \quad (4) \\ &= 2 \times c_4 \times h(n) \quad (5) \\ &= O(\max\{f(n), g(n)\}) \quad (6) \end{aligned}$$

In the above steps, we employed the arithmetic operation expression $O(cf(n)) = O(f(n))$ in asymptotic analysis to facilitate the derivation of expression (6) from expression (5).

Since we have shown that any function in $O(f(n) + g(n))$ is also in $O(\max\{f(n), g(n)\})$ and vice versa, we can conclude that $O(f(n) + g(n)) = O(\max\{f(n), g(n)\})$.

Therefore, the given statement has been proven.

# 5 Conclusion

Based on the preceding proof steps,we state the final conclusion that the sum of two polynomial complexities belongs to the maximum polynomial complexity among them.