El Método Maestro (Master Method)

El método maestro es una manera para estimar funciones a partir de desigualdades recursivas. Es muy útil para estimar el tiempo de ejecución de algoritmos "divide y conquista" como Strassen y Karatsuba.

Teorema [Master method]

Suponga que $T(n)$ satisface

$$T(n) \leq a\,T\left(\frac{n}{b}\right) + O(n^d)$$

para algunos números $a \geq 1$, $b > 1$, $d \geq 0$. Entonces

$$T(n) \sim \begin{cases} O(n^d \log(n)), & \text{si } a = b^d \\ O(n^d), & \text{si } a < b^d \\ O(n^{\log_b(a)}), & \text{si } a > b^d \end{cases}$$

Parámetros:

a = # llamadas recursivas

b = factor por el que reducimos el input

d = orden del trabajo hecho afuera de llamadas recursivas.

Ejemplos: longitud del input $= n$

$$\underbrace{\overbrace{ab}^{n}}_{\text{dígitos}} \times \overbrace{cd}^{n} = (10^{\frac{n}{2}}a + b)(10^{\frac{n}{2}}c + d)$$

$$= 10^n a\cdot c + 10^{\frac{n}{2}}(a\cdot d + b\cdot c) + b\cdot d$$

Multiplicación Recursiva:

$$T(n) \leq \underset{a}{4}\,T\left(\frac{n}{\underset{b}{2}}\right) + O(n^1) \qquad d = 1$$

$$\Rightarrow T(n) = O(n^{\log_2(4)}) = O(n^2)$$

a vs $b^d \iff 4$ vs $2^1 \Rightarrow 4 > 2^1$

Karatsuba:

$$T(n) \leq 3\,T\left(\frac{n}{2}\right) + O(n)$$

mucho mjor!

$$a \text{ vs } b^d \iff 3 \overset{>}{\text{vs}} 2^1 \Rightarrow T(n) = O\left(n^{\log_2(3)}\right) = O\left(n^{1.59}\right)$$