

Cálculo de la eficiencia del algoritmo usado para calcular las inversiones:  
 Suponemos que  $n$  es potencia de 2:

$$T(n) \begin{cases} c_1 & \text{si } n = 1 \\ 2T(n/2) + c_2n & \text{si } n > 1, n = 2^k \end{cases}$$

Para saber la eficiencia, podemos usar expansión:

$$T(n) = 2T(n/2) + c_2n$$

$$T(n/2) = 2T(n/4) + c_2n/2$$

Es decir,

$$T(n) = 4T(n/4) + 2c_2n \text{ o}$$

$$T(n) = 8T(n/8) + 3c_2n$$

En general:

$$T(n) = 2^i T(n/2^i) + ic_2n$$

Tomando  $n = 2^k$ , la expansión termina cuando llegamos a  $T(1)$  en el lado de la derecha, lo que ocurre cuando  $i = k$

$$T(n) = 2^k T(1) + kc_2n$$

Ya que  $2^k = n$ , operando,  $k = \log(n)$ ;

Además,  $T(1) = c_1$ , por lo que tenemos:

$$T(n) = c_1n + c_2n\log(n)$$

Por tanto el tiempo del algoritmo es  $O(n\log(n))$