

# Ejercicio de paralelización con OpenMP

Pablo Renero Balgación, Antonio Román López

# Índice

<b>1. Introducción</b>	<b>3</b>
1.1. Objeto del documento . . . . .	3
1.2. Metodología . . . . .	3
1.3. Realización . . . . .	3
1.4. Pasos Finales . . . . .	4
<b>2. Resultados</b>	<b>4</b>

# 1. Introducción

## 1.1. Objeto del documento

El objeto del presente documento es mostrar los resultados obtenidos en el proceso de análisis, optimización y paralelización del código secuencial proporcionado en la asignatura con el fin de obtener el mínimo tiempo de cómputo posible.

## 1.2. Metodología

La metodología seguida para la realización de la práctica ha sido la siguiente:

- Análisis del código secuencial.
- Optimizaciones sobre el código secuencial.
- Paralelización con *OpenMP*.

## 1.3. Realización

Tras el análisis del código secuencial hicimos unas pruebas de paralelización sobre el bucle que hace la llamada a actualización, con el fin de ver las diferencias sobre la ejecución secuencial, cuyo resultado fue una bajada de 3 minutos a 32 segundos. Tras esta breve prueba de paralelización decidimos hacer *function inlining*, borrando la función actualizar e introduciendo su código en donde se la llamaba.

Tras una análisis en profundidad sobre el funcionamiento de la función de energía 1, nos dimos cuenta que no es necesario recorrer todas las capas viendo si afectaba o no la energía debido a que podemos calcular ese inicio y final con estas fórmulas 2 y 3.

$$\frac{\text{energia}}{\sqrt{\text{distancia} + 1}} \geq \text{UMBRAL} \quad (1)$$

$$\text{inicio} = \text{posicion} + 1 - 1000000 * \text{energia}^2 \quad (2)$$

$$\text{fin} = \text{posicion} + 1 + 1000000 * \text{energia}^2 \quad (3)$$

Tras realizar estas modificaciones tuvimos varios resultados erróneos debido a fallos de redondeo en el cálculo de inicio y fin, para solucionarlos decidimos recorrer los  $n$  primeros y los  $n$  últimos, siendo  $n$  un número aleatorio pero que sabemos que es pequeño, para encontrar el inicio y final real.

Handwritten mathematical derivations on a chalkboard:

$$a = \frac{e}{9001} = \sqrt{d+1}$$

$$10^6 e^2 = I - K + 1$$

$$10^6 e^2 = K - I - 1$$

$$10^6 e^2 = K - I - 1$$

$$10^6 e^2 = K - I - 1$$

Figura 1: Foto del proceso de deducción de la fórmula.

#### 1.4. Pasos Finales

Tras las optimizaciones secuenciales, realizamos diversos intentos con *OpenMP* para paralelizar el bucle superior que contenía el bucle principal donde se actualizaban los datos de los bombardeos pero daba lugar una región crítica que no realizaba ninguna mejora sobre el código secuencial. Por lo tanto decidimos solo paralelizar aquellos bucles que no tuvieran ninguna posible región crítica, estableciendo todos los punteros a vectores donde se realizaban lecturas como *firstprivate* lo cual mejora el tiempo de acceso a los arrays al tener una copia privada para el hilo.

Tras estas mejoras solo nos quedaba por paralelizar el bucle que encontraba los máximos. Tras diversos intentos nos dimos cuenta que las entradas del leaderboard tenía muy pocos mínimos, por lo que si rentaba paralelizar dicho bucle dejando una región crítica a la que apenas se entraba.

## 2. Resultados

Tras estas mejoras conseguimos poder superar la segunda referencia, permitiéndonos ser los primeros durante tres días con un tiempo de 15seg.