

PROGRAMACIÓN CONCURRENTE Y DE TIEMPO REAL

Práctica 5 - Análisis producto matricial

Nicolás Ruiz Requejo

Índice

1	Análisis de Speed-Up del problema de multiplicación de matrices cuadradas	2
1.1	Tabla de las mediciones obtenidas	2
1.2	Gráfica de Speed-Up en función del número de tareas	2
1.3	Conclusiones	2

1. Análisis de Speed-Up del problema de multiplicación de matrices cuadradas

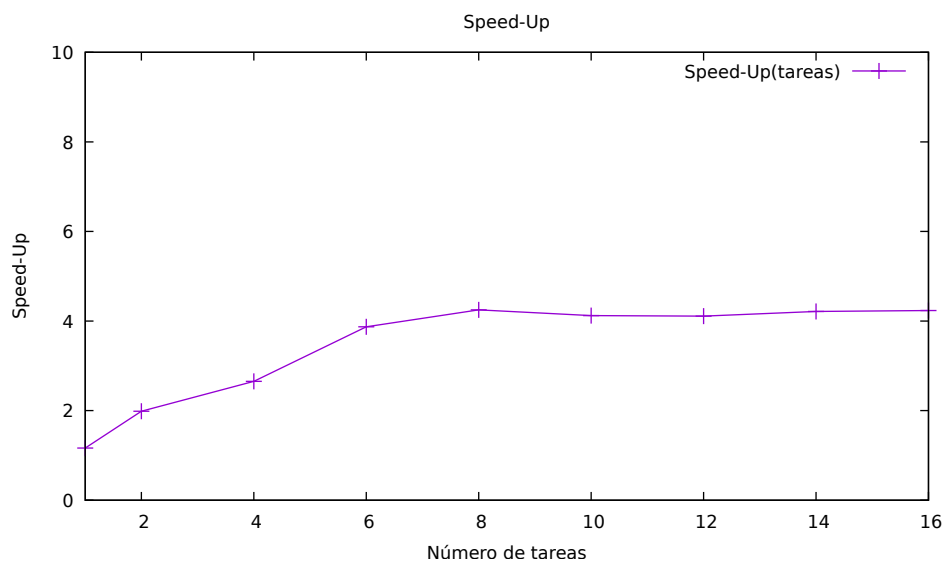
Para minimizar el tiempo de ejecución en las pruebas hemos utilizado matrices de 1000×1000 números de coma flotante de doble precisión. La máquina donde se han ejecutado las pruebas tiene un procesador de 8 núcleos (AMD FX-8350).

1.1. Tabla de las mediciones obtenidas

Los tiempos se miden en segundos.

Tiempo solución secuencial	9,28	
Número de hilos	Tiempo paralelo en segundos	Speed-Up
1	7,97	1,16
2	4,67	1,98
4	3,50	2,65
6	2,40	3,87
8	2,18	4,25
10	2,25	4,12
12	2,26	4,11
14	2,20	4,21
16	2,19	4,23

1.2. Gráfica de Speed-Up en función del número de tareas



1.3. Conclusiones

Según los Speed-up obtenidos, la versión paralela reduce el tiempo de ejecución con respecto a la versión secuencial.

Si nos fijamos en el máximo de la gráfica, el Speed-Up más alto se obtiene con 8 hilos de ejecución. Que coincide con el número de núcleos de la máquina donde se realizan las ejecuciones.

Este resultado confirma la ecuación de Subramaniam. donde para las tareas de computación intensiva que no requieren bloqueos por I/O el coeficiente de bloqueo es 0, siendo el número óptimo de hilos el número de núcleos de la máquina.

Por debajo de los 8 hilos los Speed-Up obtenidos son inferiores.

Y por encima de los 8 hilos el Speed-Up se mantienen casi en forma lineal. No esperamos un empeoramiento con un número de hilos, aun siendo superior que el máximo, tan cercano a él. Tendríamos que seguir aumentando notablemente el número de hilos hasta que el tiempo debido a los cambios de contexto supere la mejora proporcionada por el paralelismo.