

Paralelización de un código

Laboratorio 5 - Hilos → Kevin Martínez Gallego, Melissa Barba.

Se presentan las gráficas de los valores promedios de tiempo obtenidos, versus el número de hilos utilizados para realizar el cálculo del producto punto entre dos vectores.

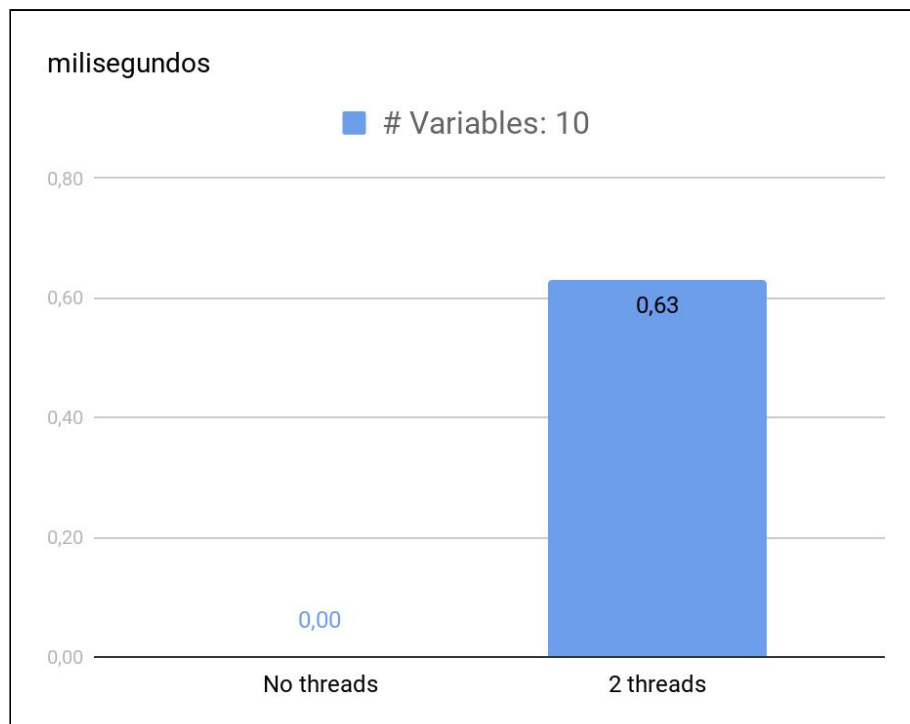
(No threads corresponde a que la ejecución fue secuencial, es decir, solo se empleo el hilo principal del programa. No se usaron hilos adicionales a este).

La arquitectura del computador utilizado para tomar los tiempos es la siguiente:

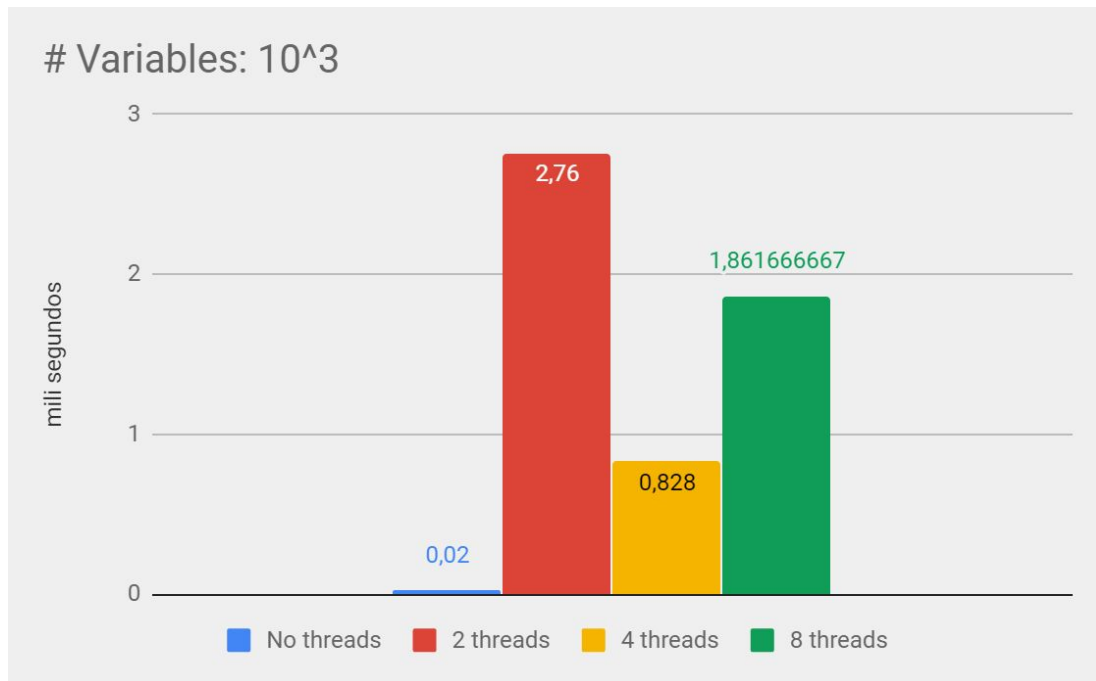
```
Architecture:      x86_64
CPU op-mode(s):    32-bit, 64-bit
Byte Order:        Little Endian
CPU(s):            4
On-line CPU(s) list: 0-3
Thread(s) per core: 2
Core(s) per socket: 2
Socket(s):         1
Vendor ID:         GenuineIntel
CPU family:        6
Model:            78
Model name:        Intel(R) Core(TM) i5-6200U CPU @ 2.30GHz
Stepping:          3
CPU MHz:           2401.000
CPU max MHz:       2401.0000
```

Tiempos promedios

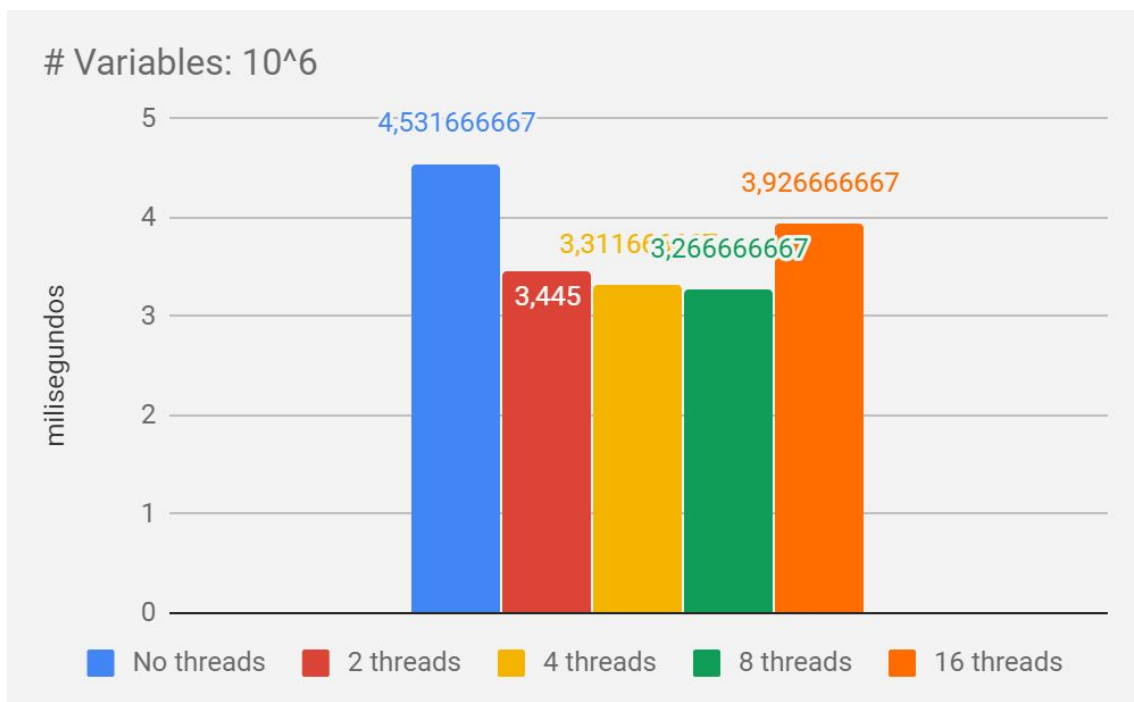
Vectores de tamaño 10^1



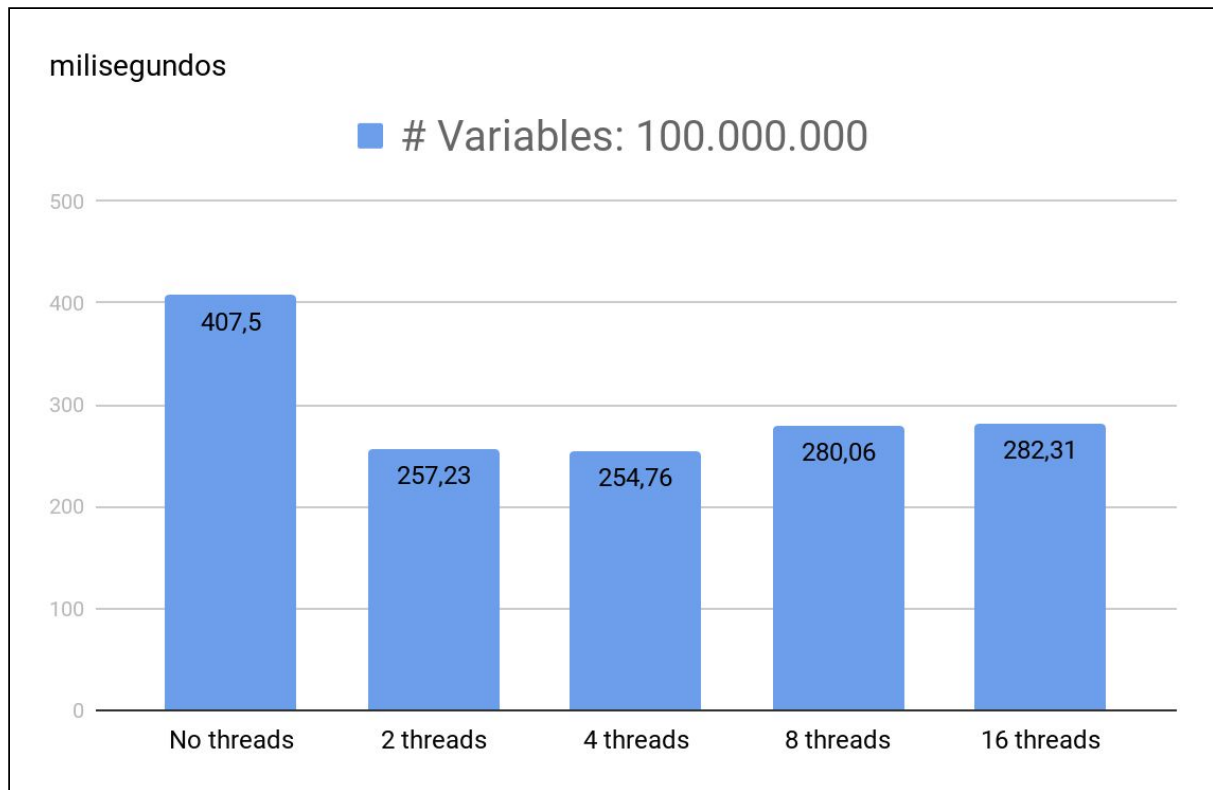
Vectores de tamaño 10^3



Vectores de tamaño 10^6



Vectores de tamaño 10^8



De los resultados obtenidos se pueden sacar varias conclusiones. Por ejemplo, se observa que en el cálculo del producto punto para vectores con relativamente pocas variables (esto es 10 o 1000 variables), se obtienen mejores tiempos de cálculo utilizando pocos hilos o ninguno, es decir, realizando una ejecución secuencial; esto es debido a que, como en el código paralelo la toma de tiempo incluye la creación de los hilos, el join y la unificación del resultado, estas acciones toman cierto tiempo de procesamiento lo que resulta eventualmente en un tiempo ponderado y comparado más elevado.

Considerando ahora el cálculo del producto punto para vectores con mayor número de variables (esto es un millón o cien millones de variables), se observa que, en general, se obtienen mejores tiempos de cálculo con una ejecución en paralelo que con una secuencial. Sin embargo, se debe notar que de acuerdo al número de hilos empleados, se presentan variaciones en los tiempos de respuesta del código paralelo. Por ejemplo, se advierte que las ejecuciones con 16 hilos toman más tiempo, en promedio, que las demás ejecuciones en paralelo, esto quizá debido a lo que se mencionó en el párrafo anterior (impacto de la creación y join de hilos así como de la unificación del resultado); además, se podría decir que entre los resultados más satisfactorios se encuentran aquellos obtenidos usando un valor entre 2, 4 u 8 hilos; esto debido a que con este número de hilos la carga de trabajo se distribuye de tal manera que se aprovechen las bondades del paralelismo sin degradar mucho el resultado creando demasiados hilos y empleando tiempo en actividades diferentes al cálculo de interés.

Téngase en cuenta que todos los tiempos presentados están dados en milisegundos, por lo tanto, las comparaciones realizadas deben entenderse e interpretarse en esta escala.