

Elapsed Time: 5.924s

Total Thread Count: 4

Paused Time: 0s

CPU Time: 4.198s

Effective Time: 2.622s

Spin Time: 1.498s

Overhead Time: 0.078s

OpenMP Analysis. Collection Time: 5.924

Serial Time (outside any parallel region): 4.010s (67.7%)

Serial Time of your application is high. It directly impacts application part of the application.

Parallel Region Time: 1.914s (32.3%)

Estimated Ideal Time: 0.472s (8.0%)

Potential Gain: 1.442s (24.3%)

The time wasted on load imbalance or parallel work arrangement is the highest metric values. Make sure the workload of the region is high.

Top Hotspots

This section lists the most active functions.

Function	CPU Time
quickSort	1.030s
_kmpc_barrier	0.936s
main	0.858s
_kmp_fork_barrier	0.562s
rand	0.538s
[Others]	0.275s

Basic Hotspots Hotspots by CPU Usage viewpoint (change)

Analysis Target: Analysis Type: Collection Log: Summary: Bottom-up: Ca

Grouping: Function / Call Stack

Function / Call Stack	CPU Time	Module
quickSort	1.030s	parallel_sort.exe
quickSort	0.910s	parallel_sort.exe
quickSort	0.780s	parallel_sort.exe
main ← [OpenMP dispatcher] ← _kmp_fork_call ← [OpenMP dispatcher]	0.720s	parallel_sort.exe
quickSort	0.060s	parallel_sort.exe
main ← [OpenMP dispatcher] ← _kmp_fork_call ← [OpenMP dispatcher]	0.130s	parallel_sort.exe
main ← [OpenMP dispatcher] ← _kmp_fork_call ← [OpenMP dispatcher]	0.120s	parallel_sort.exe
main ← [OpenMP dispatcher] ← _kmp_fork_call ← [OpenMP dispatcher]	0.060s	parallel_sort.exe

Selected 4 row(s)

2. Basándonos en la información que obtenemos en la pregunta anterior, ¿cuál es la mejor forma de paralelizar?
 - En los problemas de ordenamiento suele haber muchas secciones críticas, así que lo mejor es independizar lo más pronto posible las zonas de la estructura de datos.
 - Ya independizadas, estos bloques de datos se pueden asignar a tareas separadas.
 - En cierto punto cada tarea debe ser limitada por alguna función que indique bajo que “tamaño de grano” trabajar serialmente.
3. Basándonos en la ley de Amdahl y el reporte que nos muestra Parallel Amplifier, ¿Cuál es la aceleración esperada a lograr con 2 y 4 procesadores?

Considerando que se empleó un 32.3% del tiempo en trabajo paralelo (67.7% no), si duplicamos o cuadruplicamos el número de procesadores, la aceleración sería:

- Aceleración 4 procesadores (+0) = **1.31**
- Aceleración 6 procesadores (+2) = **1.36**
- Aceleración 8 procesadores (+4) = **1.39**

proc	ac
1	1
4	1.31
6	1.36
8	1.39

