

Sistemas paralelos 2015

Entrega 2

Integrantes:

Cuitiño, Juan Alfonso - 10581/3

Keegan, Christian Jonathan - 10694/2

Estrategias:

El problema se basa en tres partes, cada cual tiene una estrategia diferente. En términos generales, el sistema se paraleliza con cuatro threads/procesos, ya que es el hardware sobre el que se va a trabajar.

La primera parte consta de la multiplicación de tres matrices, con el detalle de que la última matriz es diagonal. Lo primero es optimizar la multiplicación por la matriz diagonal; es muy ineficiente tratarla como una matriz normal. Lo ideal es tratarla como un arreglo de N-valores y aplicar las modificaciones necesarias en la multiplicación.

En la parte paralela se trabaja con submatrices, lo que evita cualquier problema con secciones críticas ya que la escritura se da en distintas celdas.

La segunda parte consta de sacar el máximo, mínimo y promedio de las primeras dos matrices. La solución paralela trae problemas con secciones críticas, ya que varios threads/procesos pueden escribir en el máximo, mínimo y total al mismo tiempo. Los métodos se encargan de buscar un máximo y un mínimo local y compararlo con una variable compartida.

La tercer parte es el ordenamiento. El hecho de ordenar por columnas y mover toda la fila, hace que sea imposible utilizar caché en este caso. Esto además hace que cada columna sea dependiente de la anterior, así que esa parte no se puede paralelizar, lo único que queda es paralelizar el ordenamiento interno de cada columna, dividiéndola en cuatro sub-arreglos, ordenarlos y hacer un merge entre los sub-arreglos.

Algoritmo secuencial:

512	1024	2048
1.003617s	11.095742s	114.997488s

Notas:

El trabajo está sin completar, debido a que no llegamos con el tiempo. En la parte de Pthread y OpenMP falta paralelizar la parte del ordenamiento, debido a esto tampoco llegamos a tomar los tiempos de los algoritmos paralelos para calcular el speedup y eficiencia.

En el algoritmo secuencial, el movimiento de fila se hace cada vez que se mueve un valor en la columna, esto hay que optimizarlo. Lo ideal sería que los movimientos se hagan al finalizar el ordenamiento de la columna.