

Algoritmo de Agrupamiento K-Means Paralelo

HIGH PERFORMANCE COMPUTING (HPC)

PROYECTO FINAL

ESTUDIANTES:

Diego Alejandro Ramírez Ramírez

Carlos Eduardo Zuleta Uribe

PROFESOR:

John H. Osorio

UNIVERSIDAD TECNOLÓGICA DE PEREIRA

FACULTAD DE INGENIERÍAS

INGENIERÍA DE SISTEMAS Y COMPUTACIÓN

02/12/2015

ESPECIFICACIONES DEL SISTEMA

La máquina en la cual se corrió el algoritmo tiene las siguientes características:

CPU:

processor : 1
vendor_id : GenuineIntel
cpu family : 6
model : 58
model name : Intel® Core™ i7-3770K CPU @ 3.50GHz
stepping : 9
cpu MHz : 1600.000
cache size : 8192 KB
cpu cores : 4

GPU:

Tesla K40c: 3.5
Global memory: 11519mb
Shared memory: 48kb
Constant memory: 64kb
Block registers: 65536
Warp size: 32
Threads per block: 1024
Max block dimensions: [1024, 1024, 64]
Max grid dimensions: [2147483647, 65535, 65535]

PROCESO APLICADO

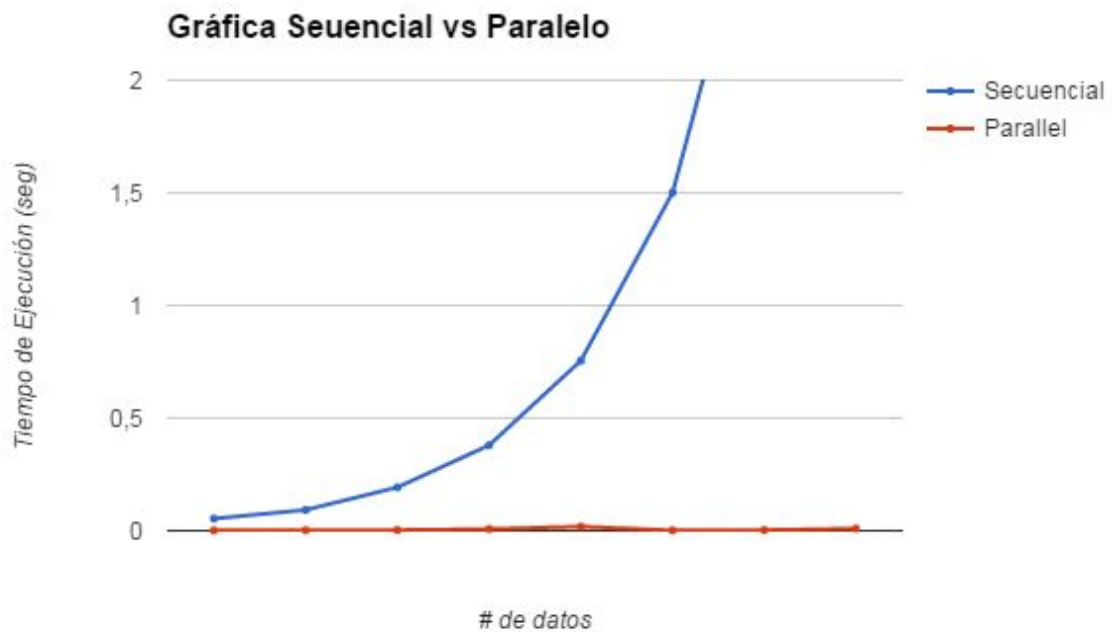
Se tomaron valores para un total de 4 clusters variando las dimensiones del set de datos, con valor mínimo de 256 y valor máximo de 4080. Tomamos para cada una de las implementaciones (Secuencial y Paralela con Memoria Compartida) 5 valores de tiempos de ejecución con las respectivas dimensiones para posteriormente sacar un valor medio y proceder a hallar la aceleración para cada caso.

Guardamos los centros en memoria compartida para permitir que la búsqueda de los valores sea más eficiente, ya que los hilos pueden tener un rápido acceso a dicha memoria para realizar los respectivos cálculos con cada uno de los datos de la matriz.

En la siguiente tabla se muestra con más detalle lo mencionado anteriormente.

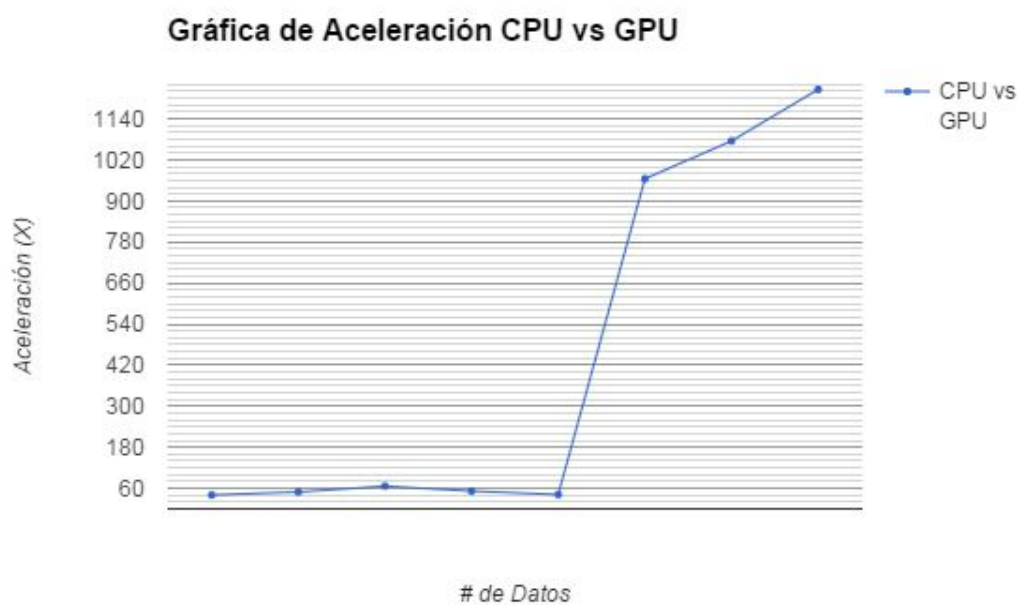
Dim matrices		Time	
N	M	Secuencial	Parallel Shared Memory
256	256	0,0536798	0,0013252
512	256	0,0922846	0,001869
512	512	0,1926096	0,002903
1024	512	0,3794812	0,0073138
1024	1024	0,7548436	0,018077
2048	1024	1,50104	0,0015562
2048	2048	3,010098	0,0027998
4.080	4.080	11,67232	0,0095222

A continuación se mostrará una gráfica comparativa de las dos implementaciones.



En la gráfica podemos ver claramente que la implementación secuencial se tarda mucho más en comparación con la implementación en paralelo, ya que esta última se puede observar que es relativamente constante en ninguno de sus puntos.

Gráfica de aceleración de la gpu con respecto a la cpu



CONCLUSIONES

- Observando la gráfica de la aceleración podemos ver que a una mayor cantidad de datos en la matriz (mayores dimensiones) existe una mayor aceleración de la GPU con respecto a la CPU.
- Pudimos constatar que el uso de memoria compartida para guardar los centros hacen que haya una mayor rapidez al momento de enviar los hilos para realizar los cálculos.
- El algoritmo de K-Means puede encontrar una solución más óptima si se implementa una versión paralela para tomar una gran cantidad de datos y procesarlos en una sola iteración.