Análisis de la Multiplicación de matrices

Luis Tong

June 22, 2017

1 Matriz Multiplicacion

Cuando se realiza una multiplicación matricial, cada elemento de la matriz de salida P es un producto interno de una fila de M y una columna de N. Continuaremos usando la convención donde P Fila, Columna es el elemento en la posición filaenesima en la dirección vertical y columnaenesima posición en la dirección horizontal.
la distribucion de los hilos se da mediante el elemento P. Los índices de fila y columna para el elemento P que se calculará por cada hilo son los siguientes:
Row = blockIdx.y * blockDim.y + threadIdx.y y
Col = blockIdx.x * blockDim.x + threadIdx.x.
Con esta asignación uno-a-uno, los índices de hilo Row y Col son también la fila y indices de columna para matriz de salida.

2 Matriz Multiplicaion Tile

Podemos intuir problemas en el clasico algoritmo de Matriz Multiplicacion asi como el uso de memorias de dispositivos en CUDA: la memoria global es grande pero lenta, mientras que la memoria compartida es pequeña pero rápida. Una estrategia común es particionar los datos en subconjuntos llamados tiles de modo que cada tile encaje en la memoria compartida. Un criterio importante es que el cálculo del kernel en estos mosaicos se puede realizar independientemente entre sí.

Entonces podemos dividir el contenido de memoria global en tiles, y asi enfocar el cálculo de los hilos en uno o en un pequeño Número de fichas en cada punto en el tiempo.
Otro punto a tomar en cuenta es la sincronización de threads, debido que cuantos estos tienen mismo acceso de sincronización podemos usar tiling de una mejora manera.

La matriz Multiplicación Tile divide la ejecución de cada subproceso en fases, de modo que los accesos de datos por el bloque de hilos en cada fase estén enfocados en un tile de M y un tile de N. El tile es de $BLOCK_SIZE$ elementos en cada dimensión.
Como vimos la sincronizacion de thread mediante un barrier para coordinar los threads dentro del bloque.

3 Comparación Matriz Multiplicación

En primer lugar se compara mediante los tiempos en seg.

Tamaño de matriz	Matriz Mult	Matriz Mult Tiling
4	0.370	0.370
16	0.377	0.343
40	0.372	0.334
80	0.355	0.326

Podemos notar como el algoritmo de multiplicación de matrices Tiling es menor tiempo a comparacion de algoritmo clasiso de mutiplicación de matrices, debido al uso de Tiles que mejoramos el acceso a la localidad de los datos, como explicamos en su seccion, mediante la memoria compartida dentro del bloque, se gana tiempo para acceder a esta memoria mas rapidamente eliminando el problema de acceso a la memoria global.

References

[1] D. Kirk and W. Hwu, Programming massively parallel processors hands-on with CUDA,