

PROGRAMACIÓN PARALELA

José Manuel Navarro Cuartero

Floyd

Se nos proporciona la implementación con una grid unidimensional de bloques unidimensionales.

Nosotros nos encargamos de hacer la implementación para una grid bidimensional de bloques bidimensionales, además de las medidas de ambas, y la comparación con una ejecución secuencial con la CPU.

Medidas

Blocksize 64 / 8x8:

Tamaño	TCPU	TGPU1	SGPU1	TGPU2	SGPU2
400	0.231643	0.007612	30.4313	0.010762	21.5242
1000	2.78737	0.09771	31.0498	0.211933	13.1521
1400	7.00967	0.235699	29.7399	0.382422	18.3296
2000	21.2179	0.586382	23.1844	0.992781	21.3722

Blocksize 256 / 16x16:

Tamaño	TCPU	TGPU1	SGPU1	TGPU2	SGPU2
400	0.201549	0.00599	33.6476	0.018571	10.8529
1000	2.62064	0.060352	43.4227	0.253136	10.3527
1400	7.05886	0.150153	47.0111	0.600222	11.7604
2000	21.3	0.397707	53.557	1.70412	12.4991

Blocksize 1024 / 32x32:

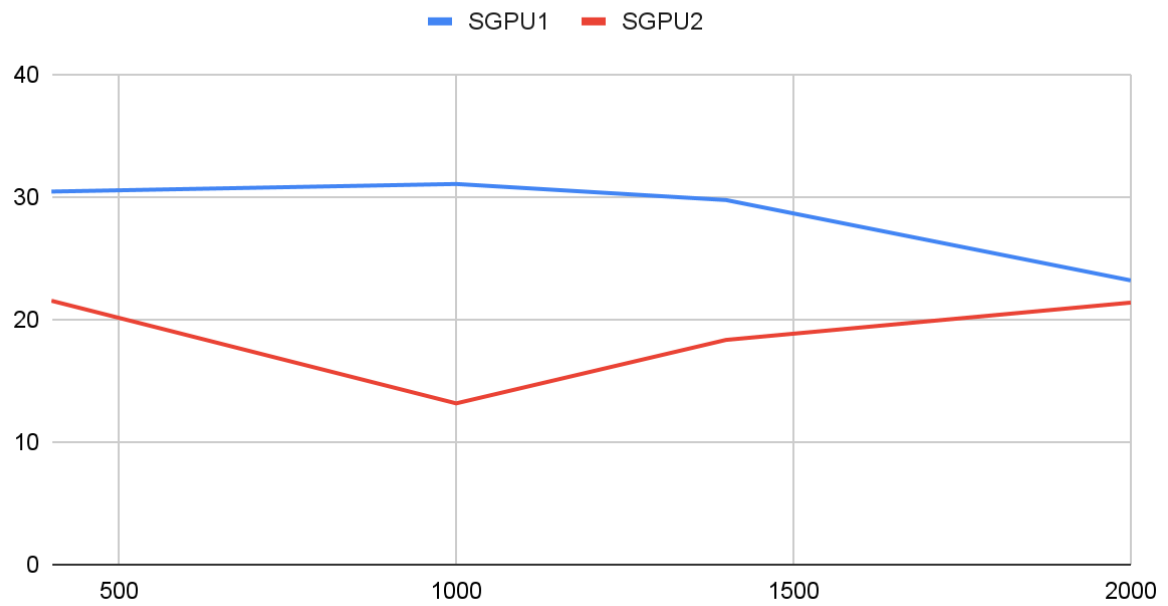
Tamaño	TCPU	TGPU1	SGPU1	TGPU2	SGPU2
400	0.169906	0.006215	27.3381	0.029102	5.83829
1000	2.586	0.066483	38.8972	0.416831	6.20396
1400	7.06238	0.16136	437744	1.11986	6.30651
2000	21.3206	0.4876	43.726	3.24294	6.57452

Observaciones

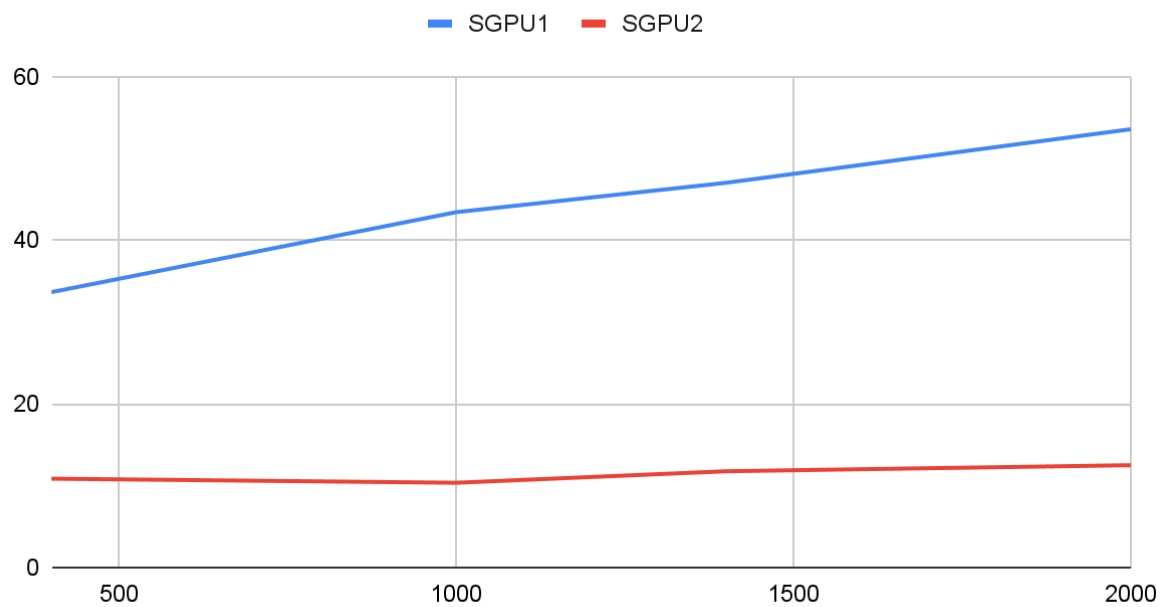
Como podemos ver, ambas implementaciones son mucho más rápidas que la implementación que sólo hace uso de la CPU, sin embargo, la implementación bidimensional pierde coalescencia, y eso puede verse reflejado en los resultados.

Gráficas

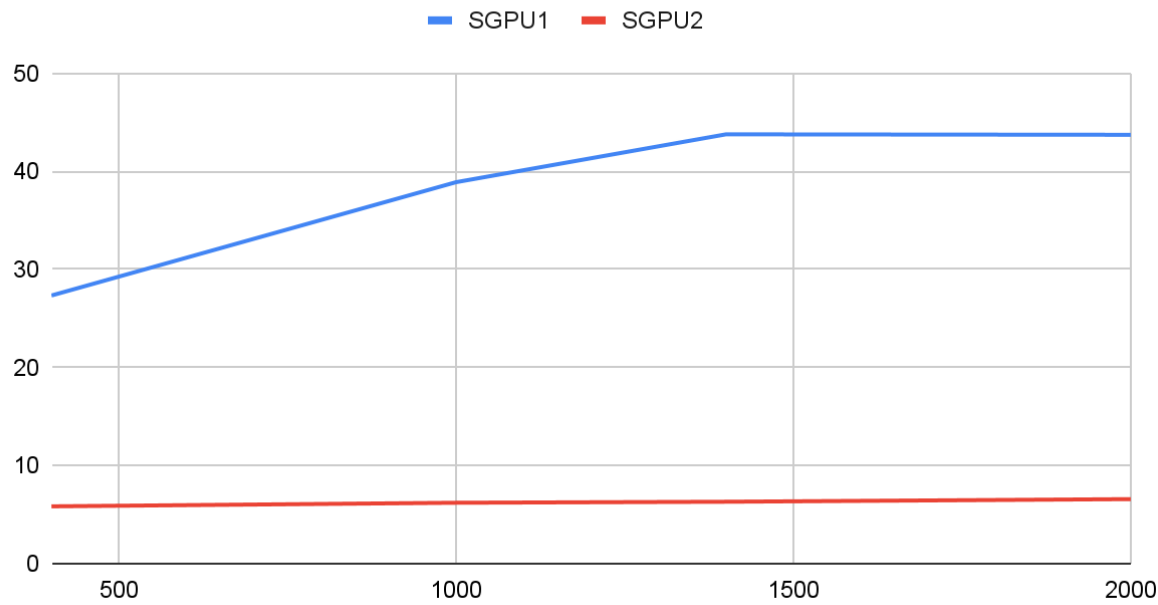
Blocksize 64 / 8x8



Blocksize 256 / 16x16



Blocksize 1024 / 32x32



Operacion Vectorial

En este caso, se nos proporcionan una serie de operaciones sobre dos vectores, que deberemos implementar en dos kernels de CUDA, uno que no haga uso de memoria compartida (T_SIN), y otro que sí lo haga (T_CON).

Medidas

Blocksize 64

NBlocks	N	T_CPU	T_SIN	T_CON
20000	1280000	0.708164	0.003443	0.032174
30000	1920000	0.918289	0.005017	0.035739
50000	3200000	1.09282	0.008311	0.035745
75000	4800000	1.86717	0.012328	0.036166
100000	6400000	2.39484	0.016346	0.035466

Blocksize 128

NBlocks	N	T_CPU	T_SIN	T_CON
20000	2560000	1.66379	0.007556	0.035505
30000	3840000	2.51422	0.011199	0.035517
50000	6400000	4.44753	0.018528	0.036993
75000	9600000	6.54174	0.027763	0.033803
100000	12800000	8.61227	0.03703	0.035877

Blocksize 256

NBlocks	N	T_CPU	T_SIN	T_CON
20000	5120000	6.61531	0.029598	0.036064
30000	7680000	9.97766	0.04437	0.035479
50000	12800000	16.5481	0.073795	0.036017
75000	19200000	25.1214	0.100391	0.034806
100000	25600000	33.1647	0.143646	0.037117

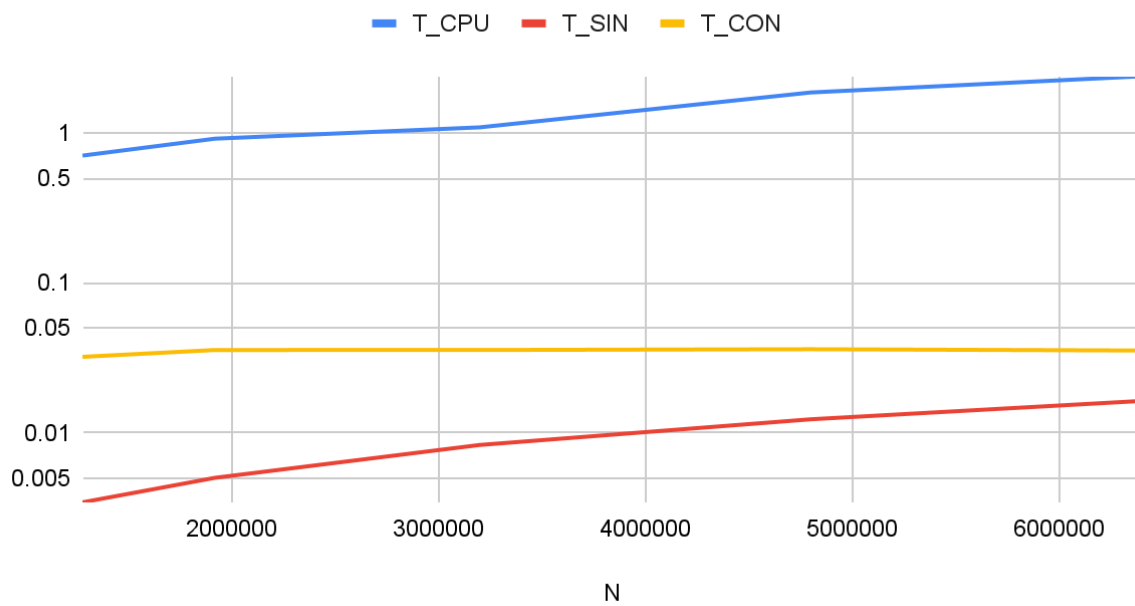
Observaciones

Podemos ver que, aunque ambas implementaciones que hacen uso de la GPU son notablemente más rápidas que la de la CPU, para valores bajos de N, la que no hace uso de memoria compartida es más rápida. Esto es debido a que el gasto de tiempo en inicializar la memoria es mayor que el ahorro en el uso.

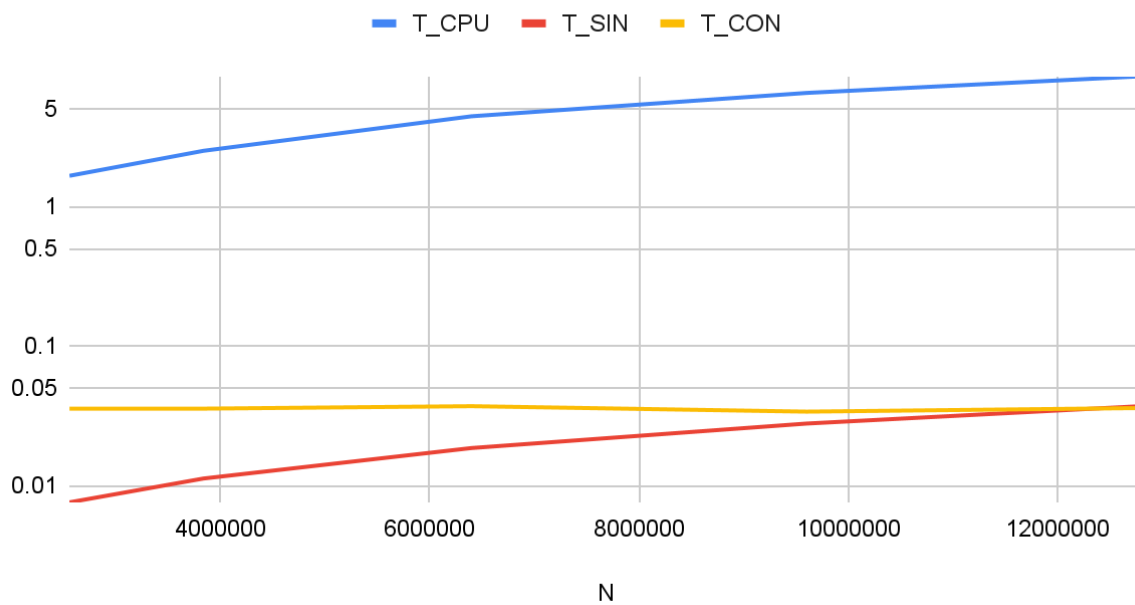
Sin embargo, para valores de N altos, esta tendencia cambia.

Gráficas

Blocksize 64



Blocksize 128



Blocksize 256

