

Algorithmique pour la programmation parallèle

SoA (structure of array) et AoS (array of structure)

Double et simple précision

Impact sur l'occupation mémoire, sur les performances, sur la précision des résultats (stabilité algorithmique). Nombre cores disponibles selon la précision (utiliser deviceQuery) et l'architecture. Minimum capacité CUDA 1.3 ou supérieure.

Compute capability	1.0	1.3	2.0	3.0	3.5
Representative device	Tesla C870	Tesla C1060	Tesla C2050	Tesla K10	Tesla K20
Number of multiprocessors	16	30	14	2 × 8	13
Single-precision cores per multiprocessor	8	8	32	192	192
Total single-precision cores	128	240	448	2 × 1536	2496
Double-precision cores per multiprocessor	-	1	16*	8	64
Total double-precision cores	-	30	224*	2 × 64	832
Max threads per multiprocessor	786	1024	1536	2048	2048

**GeForce GPUs have fewer double-precision units.*

Single and double precision en fonction de CC

Comment changer la précision sur CUDA ?

En utilisant les options de compilation de PTX (nvcc). Minimum pour la double précision : Compute Capabilities 1.3. Compiler le code CUDA avec « sm_13 ».

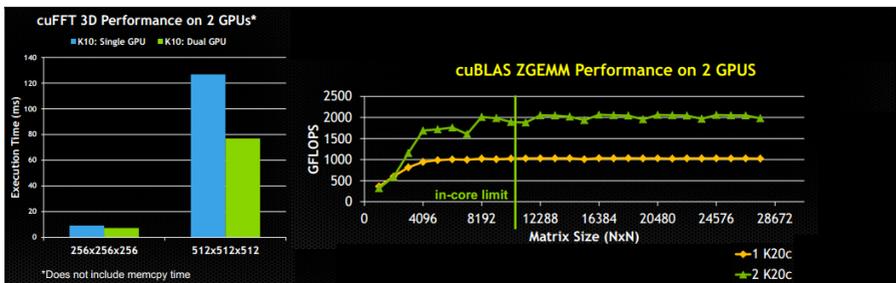
« map_f64_to_f32 » permet de forcer l'utilisation en 32 bits lorsque le GPU ne supporte pas la double précision.

Améliorer la ré-utilisabilité en utilisant les options de compilation

Code de correction des erreurs

Error-correcting code (ECC) : correction du code pour éviter les erreurs. Pas sur tous les processeurs. Impact sur l'occupation mémoire, sur les performances, sur la précision des résultats (stabilité algorithmique). Teste par `cudaDeviceProp.ECCEnabled`

Libs cuda



utilisation automatique de gpu ou cpu. Possibilité de changer directement le link d'une appli existante utilisant BLAS3 pour utiliser NVBLAS