GPU algorithms for hierarchical matrix operations

George Turkiyyah, American University of Beirut

We describe high performance hierarchical matrix algorithms on GPUs for fast matrix-vector operations and for the generation and update of an approximate inverse, to be used in solving energy minimization problems. Our algorithms use the H^2 variant of hierarchical matrices, which may be viewed as a purely algebraic version of FMM. The H^2 variant exploits, in addition to the hierarchical block partitioning, hierarchical bases for the block representations and results in a scheme that requires only O(n) storage and O(n) complexity for the mat-vec, low-rank updates, and many other basic kernels. The difficulties in developing efficient GPU algorithms come primarily from the irregular tree data structures that underlie the hierarchical representations, and the key to performance is to recast the computations on flattened trees in ways that allow batched linear algebra operations to be performed. This requires marshaling the irregularly laid out data in a way that allows them to be used by the batched routines. Marshaling operations only involve pointer arithmetic with no data movement and as a result have minimal overhead. Our GPU-resident algorithms permit real time performance on substantial problems. As an example, a matvec operation involving over a million points in 3D can be performed in under 30ms to a relative accuracy of 10^{-4} on a single GPU, and achieves 78% of the theoretical peak bandwidth. A multi-GPU version of these algorithms is under development and early results show competitive weak and strong scalability.