A Single-level Implementation for a Fast Direct Method of Moments Solver on Electrically Large Scattering Problems using a GPU based Reduced Singular Value Decomposition block LU Factorization

M. Horn*, T. Killian, D. Faircloth
IERUS Technologies, Huntsville, AL, USA

In recent years there has been considerable advancement in solving large-scale electromagnetic scattering problems using fast direct solve techniques with the traditional Rao-Wilton-Glisson (RWG) Method of Moments (MoM) computational framework, and extensions to Higher Order Basis Functions (HOBF) over curvilinear elements. The direct solve techniques are typically formulated with compression algorithms such as Adaptive-Cross-Approximation (ACA/ACA+). Early attempts for PEC bodies on the CPU (J. Shaeffer, IEEE Trans. Ant. and Prop., vol. 56, no. 8, pp. 2306-2313, Aug. 2008) and dielectric composite bodies on both the CPU and GPU (M. A. Horn, T. N. Killian, and D. L. Faircloth, 2014 IEEE Ant. and Prop. Society International Symposium (APSURSI), Memphis, TN, 2014, pp. 1630-1631) implemented a Single-level (SL) block clustering scheme in which ACA/ACA+ is used for compression in both the fill and block LU factorization. More recent attempts implement Multi-level (ML) block clustering schemes utilizing hierarchical matrix (H -Matrix) theory (W. Chai and D. Jiao, 2010 IEEE Ant. and Prop. Society International Symposium (APSURSI), Toronto, ON, 2010, pp. 1-4). ML schemes rely on the compression preserving properties of the Reduced Singular Value Decomposition (rSVD) and thus do not employ ACA/ACA+ during block LU factorization.

Some progress has been made in parallelization of ML block LU factorization on the CPU (R. Kriemann, Max Planck Inst. for Math. in the Sciences, online, Nov. 2014), including the commercially available HLIBpro. The author's do not know of a successful GPU based ML block LU factorization. Initial attempts at GPU parallelization for HLIBpro did not show performance boosts on currently available GPU HW (private e-mail communication with HLIBpro developers).

It turns out that most all the rSVD techniques underpinning the ML schemes can actually be used in a SL algorithm. The primary difference is SL does not require promotion and/or demotion of hierarchical blocks since the SL block grid is flat. The authors have found a SL CPU block LU factorization based on rSVD is far superior to a simple row / column slicing technique using ACA/ACA+ (a common first approach). Moreover, all the low level BLAS/LAPACK calls to implement rSVD are available for GPUs. For example, NVIDIA based CULA has implementations for CGESVD, CUNMQR, and CGEMM. Thus, any SL rSVD based LU factorization can be extended to a GPU architecture. As noted before, although ML uses rSVD techniques, the H-matrix operations do not efficiently extend to a GPU implementation. We shall demonstrate implementation of a GPU accelerated SL rSVD based LU factorization for electrically large composite bodies and compare to CPU based implementations and / or implementations that use ACA/ACA+ for block LU factorization.