Fast Direct Solver for Integral Equations on Massively Parallel Architectures

C. Augonnet(1), A. Pujols* (1), and M.Sesques(1)
(1) CEA/CESTA, France

The Arlene code developed at the CEA (French Atomic Energy Agency) solves 3D electromagnetic problems in the frequency domain using a classical RWG Galerkin formulation of surface integral equations. Complex structures made up of metallic and dielectric materials with all configurations of junctions and discrete symmetries can be handled. This massively parallel code efficiently exploits the thousands of cores of the TERA100 cluster of CEA and also runs on a hybrid partition with GPU accelerators. The direct solve techniques compared to the iterative ones have always been preferred due to a good accuracy requirement, a poor matrix conditioning and high efficiency for multiple incidence problems. However direct methods become prohibitive for electrically large objects.

Recently, fast direct solvers using the compression algorithm named Adaptive Cross Approximation (*ACA*) (M. Bebendorf, Numer. Math., 86, 565-589, 2000) successfully solved problems with more than one million unknowns. The hierarchical factorization based on the H-matrix framework (W. Hackbush, Computing, 62, 89-108, 1999) has very attractive features but its parallelization on massively parallel distributed computers remains a challenge. As a preliminary trade-off, we thus only consider a single level of compression. This results in a solver that is easier to parallelize in spite of its higher computational complexity.

The construction in the compressed format of the integral equation matrix is achieved with a spatial grouping of unknowns which leads to a block partitioning of this matrix. The blocks that represent interactions between spatially separated groups are compressed using the *ACA* algorithm followed by a recompression technique (based on *QR* factorization and truncated *SVD*) to obtain an optimal approximation rank. Due to composite objects, some blocks with zero-filled sub-blocks could be problematic even for the improved variant of ACA (*ACA+*) (L. Grasedyck, Computing, 74, 205-223, 2005) and must have a special treatment.

Our symmetric complex LL^T decomposition is similar to the ZPOTRF kernel of the ScaLAPACK library. It was previously rewritten into a tiled algorithm based on a task paradigm that is well suited for modern architectures, so that considering compressed blocks only required adapting a few kernels. However, block compression as well as matrix fill-in during factorization result in a more complex implementation on the TERA100 cluster especially due to load imbalance issues.

Numerical examples involving composite objects with dielectric and metallic parts are presented to show the accuracy and efficiency of the implementation.