defmm: A FFT-based Kernel-independent Directional Fast Multipole Library for Arbitrary Particle Distributions at All Frequency Regimes

Igor Chollet[1], **Xavier Claeys**[2], **Pierre Fortin**[3], **Laura Grigori**[4]

[1]Laboratoire de Chimie Théorique, UMR 7616, Sorbonne Université

[2]Sorbonne Université, Université Paris-Diderot SPC, CNRS, Inria, Laboratoire Jacques-Louis Lions, équipe Alpines, F-75005 Paris

[3]Sorbonne Université, CNRS, LIP6, F-75005 Paris, France; Univ. Lille, CNRS, Centrale Lille,UMR 9189 CRIStAL, F-59000 Lille, France

[4]Sorbonne Université, Université Paris-Diderot SPC, CNRS, Inria, Laboratoire Jacques-Louis Lions, équipe Alpines, F-75005 Paris

Abstract

Hierarchical methods for N-body problems have important applications in the fast solving of integral equations. When dealing with wave scattering problems in the high frequency regime, the design of efficient hierarchical methods is still a hard problem. In this talk, we aim at presenting our work on *defmm*, the c++ library which implements our new and efficient approach exploiting Fast Fourier Transforms (FFTs) for this class of problem. Both the mathematical and high performance computing aspects are studied and comparison results with a *state-of-the-art* library are provided.

Keywords: Fast Fourier Transforms, directional Fast Multipole Methods, integral equations, oscillatory kernels, symmetries, High performance computing

Introduction

We are concerned by the fast evaluation of N-body problems of the form

$$p(\boldsymbol{x}) = \sum_{\boldsymbol{y}\in Y} G(\boldsymbol{x},\boldsymbol{y})q(\boldsymbol{y}), \quad \boldsymbol{x}\in X$$

where $X, Y \subset \mathbb{R}^3$ are two large point clouds and G is an asymptotically smooth function, possibly oscillatory. Typically, G can refer to the Helmholtz Green's kernel $e^{i\kappa|\boldsymbol{x}-\boldsymbol{y}|}/(4\pi|\boldsymbol{x}-\boldsymbol{y}|)$ involved in integral equations in wave scattering problems. Fast Multipole Methods (FMMs) [1] allow this problem to be solved in a linear or linearithmic time (depending on the frequency regime), at the cost of a controlable approximation error. Kernel-independent approaches (i.e. requiring only the user to provide routines for evaluating G but no explicit expansion of it) exploit *directional* approximations of G in the high frequency regime [2,3], leading to complex algorithmic and strongly increasing the precomputation cost compared to the low-frequency or non-oscillatory cases.

A FFT-based directional FMM

Interpolation-based FMMs using equispaced interpolation [4] are kernel-independent methods lowering the precomputation cost while keeping fast evaluation timings for non-oscillatory kernels, compared to low-rank compression-based techniques. This existing approach uses the Fourier expression of the far-field matrix blocks $\boldsymbol{M} = [G(\boldsymbol{x}_k,\boldsymbol{y}_l)]_{k,l}$ obtained after a coarse interpolation on equispaced grids $\{\boldsymbol{x}_k\}_k, \{\boldsymbol{y}_l\}_l$:

$$\boldsymbol{M} = \mathbb{F}^*\boldsymbol{D}\mathbb{F}$$

where $\mathbb{F}$ refers to the Fourier matrix (that can be evaluated in linearithmic time through Fast Fourier Tranforms) and $\boldsymbol{D}$ is a diagonal matrix (evaluated in linear time). However, this interpolation process is known to suffer from the Runge's phenomenon. We thus introduced the first FFT-accelerated wideband directional FMM based on exploiting equispaced interpolation, for which we demonstrated the convergence. Because we target possibly highly non-uniform particle distributions (such as appearing in Boundary Element Method, especially with mesh refinement), we considered a slightly modified Dual Tree Traversal method [5] to traverse the trees obtained thanks to the hierarchical space decomposition in the FMM algorithm, allowing to flexibly switch between high- and low-frequency regimes depending on the local particle distribution (and exploiting directional or non-directional approximations accordingly). This led us to the design, implementation and optimization of the *defmm* (for **d**irectional **e**quispaced

interpolation-based **f**ast **m**ultipole **m**ethod) library, which is the synthesis of our work.

Symmetries

defmm mathematically-based optimizations are widely built on the invariances of both the kernel G and the underlying octree structure in the method: both are invariant under the action of $\mathfrak{D}_3$, the set of rotations preserving the cube

$$G(g \cdot \boldsymbol{x}, g \cdot \boldsymbol{y}) = G(\boldsymbol{x}, \boldsymbol{y}), \ \forall g \in \mathfrak{D}_3.$$

This allows to strongly reduce the number of far-field matrices $\boldsymbol{M}$ to be precomputed at the cost of application of permutation matrices P_g associated to each $g \in \mathfrak{D}_3$ (i.e. computations of $P_g^{-1}\boldsymbol{M}P_g$). Because the far-field matrices are evaluated in the Fourier domain, we extended these symmetries to this domain, leading to a new formulation of the diagonalized far-field matrices:

$$P_g^{-1}\boldsymbol{M}P_g = \mathbb{F}^*\tilde{\boldsymbol{D}}_g\mathbb{F}$$

where $\tilde{\boldsymbol{D}}_g$ is a diagonal matrix whose diagonal terms are permutations of the $\boldsymbol{D}$ ones. We then derived fast evaluations of the far-field matrices under rotations, thanks to vectorized Hadamard products for the application of $\tilde{\boldsymbol{D}}_g$ to a vector.

High Performance Computing

Our FMM is optimized for one CPU core and we provided efficient approaches for the fast evaluation of the different FMM operators. Among them, the deinterleaving of complex data plays an important role, for which we described different applications leading to important gains. By comparing our work to a *state-of-the-art* library (namely *dfmm* [3]), we validated the efficiency of our new approach, as illustrated on Fig. 1. We shall discuss in details how the algorithmic design of *defmm* combined with our handling of symmetries, as well as low-level optimizations allow us to obtain these performance results, especially on realistic test cases with space decomposition tree nodes in the high-frequency regime near the root as well as leaves at very low-frequency regimes.

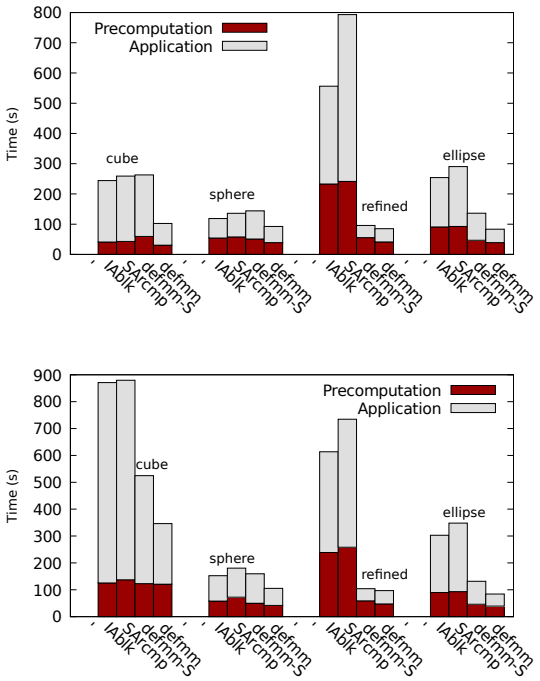

Figure 1: Timing comparison between *defmm* with vectorization (*defmm*) or without vectorization (*defmm-S*) and two variants of *dfmm* [3] (*IAblk* and *SArcmp*) for various particle distribitions. The particles are distributed uniformly in the volume of a *cube*, quasi-uniformly on the surface of a *sphere*, non-uniformly on the surface of an *ellispe* and non-uniformly on the boundaries of a cube with recursive refinement around the corners (*refined*). Two frequency regimes are considered: low-frequency (top) and high-frequency (bottom).

References

[1] L. Greengard, V. Rokhlin, A fast algorithm for particle simulations, *J. Comput. Phys.* **73** (1987), 325-348

[2] Fast Directional Multilevel Algorithms for Oscillatory Kernels B. Engquist, L. Ying, *SIAM J. Sci. Comput.*, **29** (2007), 1710-1737

[3] M. Messner, M. Schanz, E. Darve, Fast directional multilevel summation for oscillatory kernels based on Chebyshev interpolation, *J. Comput. Phys.*, **231(4)** (2012), 1175-1196

[4] P. Blanchard, O. Coulaud, E. Darve, Fast hierarchical algorithms for generating Gaussian random fields, https://hal.inria.fr/hal-01228519 (2015)

[5] W. Dehnen, A hierarchical $\mathcal{O}(N)$ force calculation algorithm, *J. Comput. Phys.*, **179(1)** (2002), 27-42