Commun. Comput. Phys.
doi: 10.4208/cicp.OA-2022-0153

Vol. **32**, No. 5, pp. 1332-1360
November 2022

The Effect of Global Smoothness on the Accuracy of Treecodes

Henry A. Boateng[1,*] and Svetlana Tlupova[2]

[1] *Department of Mathematics, San Francisco State University, San Francisco, CA 94132, USA.*
[2] *Department of Mathematics, Farmingdale State College, SUNY, Farmingdale, NY 11735, USA.*

Received 4 June 2022; Accepted (in revised version) 12 October 2022

Abstract. Treecode algorithms are widely used in evaluation of N-body pairwise interactions in $\mathcal{O}(N)$ or $\mathcal{O}(N\log N)$ operations. While they can provide high accuracy approximations, a criticism leveled at the methods is that they lack global smoothness. In this work, we study the effect of smoothness on the accuracy of treecodes by comparing three tricubic interpolation based treecodes with differing smoothness properties: a global $\mathcal{C}^1$ continuous tricubic, and two new tricubic interpolants, one that is globally $\mathcal{C}^0$ continuous and one that is discontinuous. We present numerical results which show that higher smoothness leads to higher accuracy for properties dependent on the derivatives of the kernel, nevertheless the global $\mathcal{C}^0$ continuous and discontinuous treecodes are competitive with the $\mathcal{C}^1$ continuous treecode. One advantage of the discontinuous treecode over the $\mathcal{C}^1$ continuous is that, in addition to function evaluations, the discontinuous treecode only requires evaluations of the first derivatives of the kernel while the $\mathcal{C}^1$ continuous treecode requires evaluations up to third order derivatives. When the first derivatives are computed using finite differences, the discontinuous version can be viewed as kernel independent and of utility for a wider array of kernels with minimal effort.

AMS subject classifications: 70-08, 70-10, 76-04, 85-04, 65D99

Key words: Fast summation, treecode, tricubic interpolation.

1 Introduction

The evaluation of sums of the form

$$\phi(\mathbf{x}_m)=\sum_{n=1}^{N}\mathcal{K}(\mathbf{x}_m,\mathbf{y}_n)f_n, \qquad m=1,\cdots,M, \tag{1.1}$$

*Corresponding author. *Email addresses:* `boateng@sfsu.edu` (H. A. Boateng), `tlupovs@farmingdale.edu` (S. Tlupova)

is of interest in many applications in physics, chemistry, fluid dynamics, etc. Here $\{\mathbf{y}_n\}$, $n=1,\cdots,N$ is a set of particles with weights $\{f_n\}$, and $\phi(\mathbf{x})$ is a potential (or force or velocity). The kernel $\mathcal{K}(\mathbf{x},\mathbf{y})$ represents the pairwise interaction between a target particle $\mathbf{x}$ and a source particle $\mathbf{y}$. The kernel may be a scalar or a tensor, and it is understood that if the kernel is singular for $\mathbf{x}=\mathbf{y}$, then the sum omits the $n=m$ term.

The particle-particle interactions in Eq. (1.1) can be summed directly but that requires $\mathcal{O}(MN)$ operations. This is a computationally intensive calculation when M,N are large, and many fast summation techniques have been developed to reduce the cost. Most of these methods can be grouped into tree-based methods and particle-mesh methods. In particle-mesh methods, the particles are projected onto a uniform grid where the FFT or multigrid can be used (e.g. P3M [20], particle-mesh Ewald [14], spectral Ewald [1], multilevel summation [19]).

In the tree-based approach, the particles are hierarchically partitioned into a tree structure, and the particle-particle interactions in Eq. (1.1) are replaced with particle-cluster or cluster-cluster approximations, and reduce the cost to $\mathcal{O}(N)$ [17, 18] or $\mathcal{O}(N\log N)$ [2]. The original treecode [2] used a monopole far-field approximation, while the fast multipole method (FMM) [17, 18] improved on this by employing higher-order far-field and near-field approximations expressed in terms of classical analytic multipole expansions. Later versions of the FMM used plane wave expansions for the 3D Laplace kernel [10] and spherical Bessel function expansions for the Yukawa potential [16]. Methods based on Cartesian Taylor expansions were also developed for some common kernels [3, 4, 12, 13, 24, 25, 28, 29]. These methods use analytic expansions specific to each kernel. More recently, kernel-independent methods have been developed that only require kernel evaluations and as such are applicable to more general kernel functions. Among these are the kernel-independent FMM which uses equivalent particle distribution determined by solving linear systems [32, 33], a black-box FMM which uses polynomial interpolation at Chebyshev points combined with SVD compression [15], and a kernel-independent treecode which uses barycentric Langrange interpolation at Chebyshev points [30]. Another recent development is a treecode based on barycentric Hermite polynomial interpolation [21] which requires third order derivatives of the kernel at Chebyshev evaluation points. The black-box FMM [15], the barycentric Lagrange interpolation [30] and the barycentric Hermite interpolation treecodes [21] all use a tensor product of three one-dimensional polynomials to approximate the three-dimensional kernel function.

While treecodes can provide high accuracy approximations, the approximations lack global continuity [27]. For example, in molecular dynamics (MD) simulations, the discontinuity can lead to an energy drift when the treecode multipole expansion is low order [27]. Higher order expansions resolve the issue of energy drift but at higher computational cost. In this paper, we seek to understand any effect that the smoothness of the approximations may have on the accuracy of the overall treecode algorithm. We do this by comparing treecodes based on tricubic interpolations of different smoothness. Tricubic interpolation [23] is a method of local approximation of a function defined on a regular