Journal of Computational Physics 229 (2010) 1311–1326

Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

The uselessness of the Fast Gauss Transform for summing Gaussian radial basis function series

John P. Boyd *

Department of Atmospheric, Oceanic and Space Science, University of Michigan, 2455, Hayward Avenue, Ann Arbor, MI 48109-2143, United States

ARTICLE INFO

Article history:
Received 8 February 2009
Received in revised form 4 June 2009
Accepted 19 October 2009
Available online 3 November 2009

Keywords:
Fast Gauss Transform
Radial basis functions
Fast multipole
Treecode

ABSTRACT

The Fast Gauss Transform is an algorithm for summing a series of Gaussians which is sometimes much faster than direct summation. Gaussian series in d dimensions are of the form $\sum_{j=1}^{N} \lambda_j \exp(-[\alpha/h]^2\ \|\mathbf{x}-\mathbf{x}_j\|^2)$ where the $\mathbf{x}_j$ are the centers, h is the average separation between centers and α is the relative inverse width parameter. We show that the speed-up of the Fast Gauss Transform is bounded by a factor $\Omega(\alpha)$. When $\alpha \ll 1$, Ω can be large. However, when applied to Gaussian radial basis function interpolation, it is difficult to apply the Gaussian basis in this parameter range because the interpolation matrix is exponentially ill-conditioned: the condition number $\kappa \sim (1/2)\exp\left(\frac{\pi^2}{4\alpha^2}\right)$ for a uniform, one-dimensional grid, and larger still in two dimensions or when the grid is irregular. Furthermore, the Gaussian RBF interpolant is ill-conditioned for most series in the sense that the interpolant is the small difference of terms with exponentially large coefficients. Fornberg and Piret developed a "QR-basis" that ameliorates this difficulty for approximations on the surface of a sphere, but because the recombined basis functions are perturbed spherical harmonics, not Gaussians, the Fast Gauss Transform is no longer applicable. The solution of the interpolation matrix system by a preconditioned iteration is less sensitive to condition numbers than direct methods because iterations are self-correcting and also because the preconditioning reduces the spread of the eigenvalues. However, each iteration requires a matrix–vector multiply which is fast only if this operation can be performed by some species of Fast Summation. When $\alpha \sim O(1)$, alas, we show that Ω is not large and the Fast Gauss Transform is not accelerative. Gaussian RBFs are unusual among RBF species through the absence of long-range interactions due to the exponential decay of the Gaussians with distance from their centers; many other RBF species do have long-range interactions, and it is well-established that these can be accelerated by fast multipole and treecode algorithms. We offer a less rigorous scale analysis argument to explain why the underlying difficulty in accelerating short-range interactions is not peculiar to the Gaussian RBF basis or to the Fast Gauss Transform, but rather is likely to be a generic difficulty in accelerating the short-range interactions of almost any RBF basis with almost any Fast Summation.

1. Introduction

Radial basis functions (RBFs) have been successfully applied to solve many species of partial differential equations. However, most of these applications have been restricted to rather small N where N is the total number of degrees of freedom. For $N \leqslant 4000$, it does not matter that the RBF interpolation, differentiation and summation matrices are dense $N \times N$ matrices

* Tel.: +1 734 764 3338; fax: +1 734 764 5137.
E-mail address: jpboyd@umich.edu

0021-9991/$ - see front matter
doi:10.1016/j.jcp.2009.10.032

Table 1
List of symbols.

a_n	nth Chebyshev coefficient
d	Spatial dimension
FGT	Fast Gauss Transform
h	Average grid spacing
IFGT	Improved (Taylor series-based) Fast Gauss Transform
m	Radius from an RBF center in units of h
N_B	Number of RBFs with centers in a given block
p	Degree of Hermite series or Chebyshev series
$\tilde{p}$	$p-1$, one less than the degree of a Taylor series
P	One less than the degree of Taylor series
r	$(\equiv \epsilon s/\sqrt{2})$ length of a side of a block relative to the RBF e-folding scale
$\mathcal{R}$	Parameter in Taylor series-based IFGT
q	Exponent of Generalized Multiquadric
s	Length of one side of the block enclosing a cluster of RBFs
x	Coordinate (one dimension)
$\mathbf{x}$	Coordinate (two or more dimensions)
TGT	Truncated Gauss Transform
α	$(=\epsilon h)$ inverse width relative to the grid spacing
ϵ	Absolute inverse width of the RBFs
λ_j	Coefficient of jth term in RBF series
$\phi(r)$	RBF function
ρ	$(\equiv r\sqrt{\exp(1)/p})$ parameter that must be small for small Hermite series error

because these can still be inverted or multiplied in a few minutes on a workstation. Because of its ease of programming, RBFs are a great "small-N" method. Applications to "high-N" problems will be prohibitively costly, however, unless some analogue to the Fast Fourier Transform can be found.

Truncation, short-range interactions and long-range interactions offer what military strategists call "lines of attack". If the RBFs are spatially localized, as true of Gaussians, then great savings can be realized in summation at a point $\mathbf{x}$ by discarding the very tiny contributions from RBFs whose peaks are sufficiently distant from $\mathbf{x}$. This gives the "Truncated Gauss Transform" for Gaussians, and similarly for other RBFs that decay exponentially away from their peaks such as sech's.

Fast Summation methods, which we shall use as a shorthand for a broad family of algorithms that include the Fast Multipole Method, Fast Gauss Transform, and treecodes, are designed to evaluate a summation of N terms at each of L points where $L \geqslant O(N)$. Instead of evaluating the sum at a given point independently of all the other points at a cost of $O(N)$ operations per point, Fast Summations share work between the sums at different points so as to greatly reduce the cost. Many species of Fast Summations including the Fast Gauss Transform replace interactions among particles, vortices or radial basis functions by a *proxy* series, a sort of mathematical stunt double. Some Fast Summations attack only long-range interactions, some only short-range interactions, and some both kinds of interactions.

For example, the earliest Fast Summation method, the "Fast Multipole Method", was invented to accelerate computation of the long-range gravitational interactions among N stars. The stars — RBFs in our application — are grouped into clusters containing N_B elements; the combined influence of all the stars in the group is approximated by a series of negative powers of the distance ("multipole expansion") from a group of stars to the evaluation point. If the proxy series can be truncated to only M terms where $M \ll N_B$, then the summation is accelerated by a factor of $\Omega \equiv M/N_B$. (A full list of symbols is given in Table 1.) Treecodes are similar but use a Taylor series as a proxy [33].

Because both Fast Summations and RBFs have bloomed into a bewildering variety of genera and species, we shall focus primarily on two summation methods for one type of RBFs: the Fast Gauss Transform (FGT) and Truncated Gauss Transform (TGT) for Gaussian RBFs. The FGT and TGT are illuminating to analyze because both are particularly tailored to Gaussians. Gaussian RBFs are useful examples because (i) they are popular [45,51,52,54,12,13,30,34,37,49,59,3],[1] widely used and converge exponentially fast for smooth functions $f(x)$ and (ii) have *only* short-range interactions, allowing us to focus on that part of summing RBFs where Fast Summations have difficulties. (As we shall explain later, long-range interactions, for those RBF species that have them, are well accelerated by some Fast Summations.)

In d dimensions where d is an arbitrary positive integer, a Gaussian RBF approximation to a function $f(x)$ has the form:

$$f(\mathbf{x}) = \sum_{j=1}^{N} \lambda_j \exp(-(\alpha/h)^2 \; \|\mathbf{x}-\mathbf{x}_j\|_2^2) \tag{1}$$

where $\mathbf{x}$ is a point in R^d, the λ_j are the RBF coefficients and the $\mathbf{x}_j$ are the "centers", α is a user-chosen constant ("inverse width parameter") and h is the *average* grid spacing; one of the charms of RBFs is that the grid can be *irregular*. Because the basis functions are not orthogonal, the RBF coefficients λ_j are almost always found by interpolation at a set of interpolation points

[1] The *Web of Science* returned 197 articles for the keyword "Gaussian radial basis function". We do not cite them all, but it should be noted that Gaussian RBFs are *very* popular in neural networks.