Appl. Comput. Harmon. Anal. • • • (• • • •) • • •–• • •

Contents lists available at ScienceDirect

Applied and Computational Harmonic Analysis

www.elsevier.com/locate/acha

Letter to the Editor

Multidimensional butterfly factorization

Yingzhou Li [b,*], Haizhao Yang [c], Lexing Ying [a,b]

[a] Department of Mathematics, Stanford University, United States
[b] ICME, Stanford University, United States
[c] Department of Mathematics, Duke University, United States

A R T I C L E   I N F O

Article history:
Received 26 September 2015
Received in revised form 30 March 2017
Accepted 11 April 2017
Available online xxxx
Communicated by Joel A. Tropp

MSC:
44A55
65R10
65T50

Keywords:
Data-sparse matrix factorization
Operator compression
Butterfly algorithm
Randomized algorithm
Fourier integral operators

A B S T R A C T

This paper introduces the multidimensional butterfly factorization as a data-sparse representation of multidimensional kernel matrices that satisfy the complementary low-rank property. This factorization approximates such a kernel matrix of size $N \times N$ with a product of $O(\log N)$ sparse matrices, each of which contains $O(N)$ nonzero entries. We also propose efficient algorithms for constructing this factorization when either (i) a fast algorithm for applying the kernel matrix and its adjoint is available or (ii) every entry of the kernel matrix can be evaluated in $O(1)$ operations. For the kernel matrices of multidimensional Fourier integral operators, for which the complementary low-rank property is not satisfied due to a singularity at the origin, we extend this factorization by combining it with either a polar coordinate transformation or a multiscale decomposition of the integration domain to overcome the singularity. Numerical results are provided to demonstrate the efficiency of the proposed algorithms.

1. Introduction

1.1. Problem statement

This paper is concerned with the efficient evaluation of

$$u(x) = \sum_{\xi \in \Omega} K(x, \xi) g(\xi), \quad x \in X, \tag{1}$$

where X and Ω are typically point sets in $\mathbb{R}^d$ for $d \geq 2$, $K(x, \xi)$ is a kernel function that satisfies a complementary low-rank property, $g(\xi)$ is an input function for $\xi \in \Omega$, and $u(x)$ is an output function

* Corresponding author.
E-mail address: ryanli@stanford.edu (Y. Li).

http://dx.doi.org/10.1016/j.acha.2017.04.002
1063-5203/

for $x \in X$. To define this complementary low-rank property for multidimensional kernel matrices, we first assume that without loss of generality there are N points in each point set. In addition, the domains X and Ω are associated with two hierarchical trees T_X and T_Ω, respectively, where each node of these trees represents a subdomain of X or Ω. Both T_X and T_Ω are assumed to have $L = O(\log N)$ levels with X and Ω being the roots at level 0. The computation of (1) is essentially a matrix–vector multiplication

$$u = Kg,$$

where $K := (K(x,\xi))_{x\in X,\xi\in\Omega}$, $g := (g(\xi))_{\xi\in\Omega}$, and $u := (u(x))_{x\in X}$ by a slight abuse of notations. The matrix K is said to satisfy the *complementary low-rank property* if for any level ℓ between 0 and L and for any node A on the ℓ-th level of T_X and any node B on the $(L-\ell)$-th level of T_Ω, the submatrix $K_{A,B} := (K(x_i,\xi_j))_{x_i\in A,\xi_j\in B}$ is numerically low-rank with the rank bounded by a uniform constant independent of N. In most applications, this numerical rank is bounded polynomially in $\log(1/\epsilon)$ for a given precision ϵ. A well-known example of such a matrix is the multidimensional Fourier transform matrix.

For a complementary low-rank kernel matrix K, the *butterfly algorithm* developed in [1,2,13,14,16] enables one to evaluate the matrix–vector multiplication in $O(N\log N)$ operations. More recently in [7], we introduced the *butterfly factorization* as a data-sparse multiplicative factorization of the kernel matrix K in the one-dimensional case ($d = 1$):

$$K \approx U^L G^{L-1} \cdots G^{L/2} M^{L/2} \left(H^{L/2}\right)^* \cdots \left(H^{L-1}\right)^* \left(V^L\right)^*, \tag{2}$$

where the depth $L = O(\log N)$ is assumed to be an even number and every factor in (2) is a sparse matrix with $O(N)$ nonzero entries. Here the superscript of a matrix denotes the level of the factor rather than the power of a matrix. This factorization requires $O(N\log N)$ memory and applying (2) to any vector takes $O(N\log N)$ operations once the factorization is computed. In fact, one can view the factorization in (2) as a compact algebraic representation of the butterfly algorithm. In [7], we also introduced algorithms for constructing the butterfly factorization for the following two cases:

(i) A black-box routine for rapidly computing Kg and K^*g in $O(N\log N)$ operations is available;
(ii) A routine for evaluating any entry of K in $O(1)$ operations is given.

In this paper, we turn to the butterfly factorization for the multidimensional problems and describe how to construct them for these two cases.

When the kernel strictly satisfies the complementary low-rank property (e.g., the non-uniform FFT), the algorithms proposed in [7] can be generalized in a rather straightforward way. This is presented in detail in Section 2.

However, many important multidimensional kernel matrices fail to satisfy the complementary low-rank property in the entire domain $X \times \Omega$. Among them, the most significant example is probably the Fourier integral operator, which typically has a singularity at the origin $\xi = 0$ in the Ω domain. For such an example, existing butterfly algorithms provide two solutions:

- The first one, proposed in [2], removes the singularity by applying a polar transformation that maps the domain Ω into a new domain P. After this transformation, the new kernel matrix defined on $X \times P$ satisfies the complementary low-rank property and one can then apply the butterfly factorization in the X and P domain instead. This is discussed in detail in Section 3 and we refer to this algorithm as the *polar butterfly factorization* (PBF).
- The second solution proposed in [8] is based on the observation that, though not on the entire Ω domain, the complementary low-rank property holds in subdomains of Ω that are well separated from the origin