Available online at www.sciencedirect.com
ScienceDirect
Comput. Methods Appl. Mech. Engrg. 333 (2018) 311–330
Computer methods in applied mechanics and engineering
www.elsevier.com/locate/cma

A black-box low-rank approximation algorithm for fast matrix assembly in Isogeometric Analysis

Clemens Hofreither

Institute of Computational Mathematics, Johannes Kepler University Linz, Altenberger Str. 69, 4040 Linz, Austria

Received 27 April 2017; received in revised form 11 January 2018; accepted 12 January 2018

Abstract

We present a fast algorithm for the computation of stiffness matrices in Isogeometric Analysis with tensor product spline spaces, for both scalar and vector-valued problems. The procedure exploits the facts that (a) such matrices have block-banded structure, and (b) they often have low Kronecker rank. Combined, these two properties allow us to reorder the nonzero entries of the stiffness matrix into a relatively small, dense matrix or tensor of low rank. A suitable black-box low-rank approximation algorithm is then applied to this matrix or tensor. This allows us to approximate the nonzero entries of the stiffness matrix while explicitly computing only relatively few of them, leading to a fast assembly procedure.

The algorithm does not require any further knowledge of the used spline spaces, the geometry transform, or the partial differential equation, and thus is black-box in nature. Existing assembly routines can be reused with minor modifications. A reference implementation is provided which can be integrated into existing code.

Numerical examples demonstrate significant speedups over a standard Gauss quadrature assembler for several geometries in two and three dimensions. The runtime scales sublinearly with the number of degrees of freedom in a large pre-asymptotic regime.

Keywords: Isogeometric Analysis; Matrix assembly; Low-rank approximation; Tensor approximation; Adaptive Cross Approximation

1. Introduction

An often-cited obstacle in the practical use of Isogeometric Analysis (IgA; [1]) is the high computational effort required to assemble the involved stiffness matrices, especially for higher spline degrees. For a d-dimensional tensor product spline space with n^d degrees of freedom and spline degree p, standard assemblers based on tensor product Gauss quadrature require $\mathcal{O}(n^d p^{3d})$ operations to compute the stiffness matrix, which has $\mathcal{O}(n^d p^d)$ nonzero entries.

For this reason, various approaches have been proposed to speed up matrix assembly in IgA. Rather than attempt to reiterate all prior work, we refer to the two recent articles [2,3], whose introductory sections contain an overview of the state of the art. In short, many approaches aim at developing quadrature rules which are more efficient than the naive tensor product Gauss quadrature approach with $\mathcal{O}(p)$ nodes per knot span. On the other hand, the method

E-mail address: chofreither@numa.uni-linz.ac.at.

https://doi.org/10.1016/j.cma.2018.01.014
0045-7825/

presented in [3] is based on techniques of low-rank tensor approximation. Although its applicability depends on the existence of good low-rank approximations to the geometry map and coefficient functions, these techniques appear to be the only known ones to achieve running times that scale sublinearly with the number of degrees of freedom in a significant pre-asymptotic regime. For surveys of the large field of low-rank tensor methods, see [4–6].

Much like the approach in [3], the algorithm presented in the present work is based on low-rank approximation and shares many of the same properties. In particular, it achieves comparable speedups to those claimed in [3], can achieve sublinear scaling, and its efficiency depends significantly on the low-rank approximability of the involved quantities. The major advantage of the proposed method is that, in contrast to [3], it is purely algebraic and black-box in nature:

- it does not require explicit knowledge of the PDE, the geometry map or the used basis functions and works without modifications for a wide class of problems;
- it can reuse existing assembler routines with minor modifications;
- it is much easier to implement;
- a black-box implementation can be provided which can be integrated into existing codebases relatively easily.

Rather than an entirely new assembler, the proposed algorithm may thus be considered an algebraic "accelerator" to be wrapped around existing assemblers. Two open-source implementations of this algorithm are made available to the community (see Section 5.1).

Our approach is based on a matrix reordering first described by Van Loan and Pitsianis [7] which converts Kronecker products into outer products of vectors, and matrices with low Kronecker rank into matrices with low rank. The particular block structure of IgA matrices allows us to find a relatively small, dense submatrix of the reordered stiffness matrix which contains exactly the nonzeros of the original matrix. Since IgA matrices often have low Kronecker rank or can be well approximated by matrices with low Kronecker rank, the reordered matrix typically has low rank. By applying Adaptive Cross Approximation (ACA) [8,9] to this low-rank matrix, we can compute it very accurately while evaluating only relatively few of its entries. Once this is done, we recover the original stiffness matrix by simply reverting the reordering operation.

The idea of the matrix reordering from [7] was already exploited for approximations with low Kronecker rank in [10]; there, however, the Kronecker factors were not banded, but instead approximated using $\mathcal{H}$-matrices (cf. [9,11]). A similar approach where the blocks have Toeplitz structure is given in [12]. Tyrtyshnikov [13] gives some error estimates for a related approximation problem. The idea of exploiting the block-banded structure to represent the reordered matrix compactly appears to be novel; likewise the application of ACA to the reordered matrix in order to recover Kronecker approximations. A previous use of ACA in the IgA context was for the efficient approximation of bivariate functions by sums of separable splines [14].

The remainder of the paper is structured as follows. In Section 2, we describe two crucial but natural properties of IgA stiffness matrices which we exploit in the construction of our algorithm, namely the particular block-banded structure and the low numerical Kronecker rank. Based on these two properties, we develop our fast assembly algorithm in the 2D case in Section 3. The extension to the 3D case is then rather straightforward and is described in Section 4. We discuss an implementation and present numerical examples in Section 5.

2. Properties of IgA stiffness matrices

2.1. Hierarchical block-banded structure

We first choose a set of univariate basis functions

$$\Phi = (\varphi_i)_{i=1}^n$$

over the interval $[0, 1]$. We make the assumption that the basis functions have local support in the sense that, for some $p \in \mathbb{N}$, we have

$$|i - j| > p \quad \Longrightarrow \quad \left|\operatorname{supp} \varphi_i \cap \operatorname{supp} \varphi_j\right| = 0 \qquad \forall i, j \in \{1, \dots, n\}.$$

Here $|\cdot|$ on the right-hand side denotes measure. Basis functions typically used in IgA, such as B-splines and NURBS, naturally satisfy this assumption with p being the spline degree. In higher dimensions, we may take tensor product bases, e.g., with univariate bases Φ_1 and Φ_2,

$$\varphi_{i_1,i_2} \in \Phi_1 \times \Phi_2 : \quad \varphi_{i_1,i_2}(x, y) = \varphi_{i_1}(x)\varphi_{i_2}(y) \tag{1}$$