ARTICLE IN PRESS

Computers and Mathematics with Applications ▮ (▮▮▮▮) ▮▮▮–▮▮▮

Contents lists available at ScienceDirect

Computers and Mathematics with Applications

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

Parallelization of continuous and discontinuous Galerkin dual–primal isogeometric tearing and interconnecting methods

Christoph Hofer

Johannes Kepler University (JKU), Altenbergerstr. 69, A-4040 Linz, Austria

ARTICLE INFO

Article history:
Available online xxxx

Keywords:
Diffusion problems
Isogeometric analysis
Discontinuous Galerkin
IETI-DP
Parallelization
MPI

ABSTRACT

In this paper we investigate the parallelization of dual–primal isogeometric tearing and interconnecting (IETI-DP) type methods for solving large-scale continuous and discontinuous Galerkin systems of equations arising from Isogeometric analysis of elliptic boundary value problems. These methods are extensions of the finite element tearing and interconnecting methods to isogeometric analysis. The algorithms are implemented by means of energy minimizing primal subspaces. We discuss how these methods can efficiently be parallelized in a distributed memory setting. Weak and strong scaling studies presented for two and three dimensional problems show an excellent parallel efficiency.

1. Introduction

Isogeometric Analysis (IgA) is a novel methodology for the numerical solution of partial differential equations (PDE). IgA was first introduced by Hughes, Cottrell and Bazilevs in [1], see also the monograph [2] for a comprehensive presentation of the IgA framework and the recent survey article [3]. The main principle is to use the same basis functions for describing the geometry and to represent the discrete solution of the PDE problem under consideration. The most common choices are B-Splines, Non Uniform Rational B-Splines (NURBS), T-Splines, Truncated Hierarchical B-Splines (THB-Splines), etc., see, e.g., [4–6]. One of the strengths of IgA is the capability of creating high-order splines spaces, while keeping the number of degrees of freedom quite small. Moreover, having basis functions with high smoothness is useful when considering higher-order PDEs, e.g., the biharmonic equation.

In many cases the domain cannot be represented with a single mapping, referred to as *geometrical mapping*. Complicated geometries are decomposed into simple domains, called *patches* or *subdomains*, which are the image of the unit cube through a small parametrization. The set of patches forming the computational domain is called multipatch domain. The obtained patch parametrizations and the original geometry may not be identical. The result are small gaps and overlaps occurring at the interfaces of the patches, called *segmentation crimes*, see [7–9] for a comprehensive analysis. Nevertheless, one still wants to solve PDEs on such domains. To do so, numerical schemes based on the discontinuous Galerkin (dG) method for elliptic PDEs were developed in [10–12]. There, the corresponding error analysis is also provided. In addition to domains with segmentation crimes, the dG formulation is very useful when considering different B-Splines spaces on each patch, e.g., non-matching grids at the interface and different spline degrees. An analysis for the dG-IgA formulation with extensions to low regularity solutions can be found in [13]. For a detailed discussion of dG for finite element methods, we refer, e.g., to [14,15].

In the present paper, we are considering fast solution methods for linear systems arising from the discretization of elliptic PDEs by means of IgA. We investigate non-overlapping domain decomposition (DD) methods of the dual–primal tearing and

E-mail address: christoph.hofer@ricam.oeaw.ac.at.

http://dx.doi.org/10.1016/j.camwa.2017.06.051
0898-1221/

interconnecting type. This type of methods are equivalent to the so called Balancing Domain Decomposition by Constraints (BDDC) methods, see [16–18] and references therein. The version based on a conforming Galerkin (cG) discretization, called dual–primal isogeometric tearing and interconnecting (IETI-DP) method was first introduced in [19] and the equivalent IgA BDDC method was analyzed in [20]. Further extensions to the analysis are presented in [21]. The version based on the dG formulation, abbreviated by dG-IETI-DP, was introduced in [22] and analyzed in [23], see [24–26] for the corresponding finite element counterparts.

The aim of this paper is to present the parallel scalability of the cG and dG IETI-DP methods. We investigate weak and strong scaling in two and three dimensional domains for different B-Spline degrees. Our parallelization strategy is based on a splitting of patches via increasing the multiplicity of knots at the desired interfaces, i.e., reducing the continuity there to C^0. We propose a hybrid cG-dG version, where we first use the dG version (C^{-1} continuity) to handle material interfaces (jumping coefficients), non-matching meshes and different polynomial degrees, and as a second step using the cG version with C^0 interfaces for a further splitting of the single patches. As investigated in Section 6.4, enforcing C^0 continuity on edges or faces via knot multiplicity p yields only a small increase in the number of dofs. This approach leads to a great flexibility in parallelization and to high efficiency in terms of the total CPU time. However, one has to keep in mind that for higher order PDEs, the outlined procedure does not work, since higher smoothness than C^0 is required, e.g., C^1 in the case of the biharmonic equation. In such cases, one could work with dG techniques, e.g., [27], or PDE-decomposition techniques [28], where a fourth order problem is decomposed into a sequence of second order problems. In [29], the authors use the same technique, i.e., the introduction of C^0 interfaces at certain meshlines, to increase the performance of multifrontal sparse direct solvers. However, in our work we use the additional C^0 interfaces to provide a further decomposition of the patches to increase the parallel scalability, whereas in [29] the C^0 interfaces are used to reduce the interconnection of the dofs and increase the serial performance of the direct solver. In [20,30], a different approach is presented, where fat interfaces are used to keep the smoothness of the spaces of the interfaces. The advantage is to keep the total number of unknowns small, however, at the cost of larger interfaces, larger number of primal variables and the need to use the *deluxe* scaling to obtain a robust method. The recent publication [31] addresses this issue by using an adaptive selection of primal variables, resulting in small coarse spaces. One obtains an efficient method, which seems to be a promising alternative to the method investigated in the current work.

The implemented algorithms are based on energy minimizing primal subspaces, which simplify the parallelization of the solver part, but having more effort in the setup phase (assembling phase). We rephrase key parts of this algorithm and discuss how to realize the communication by means of Message Passing Interface (MPI). In general, FETI-DP and equivalent BDDC methods are by nature well suited for large-scale parallelization and has been widely studied for solving large-scale finite element equations, e.g., in [32–35] and [36], see also [37] for a hybrid OpenMP/MPI version. Considering a domain decomposition with several ten thousands of patches, the influence of the coarse grid problem becomes more and more significant. Especially, its LU-factorization is the bottleneck of the algorithm. The remedy is to reformulate the FETI-DP system in such a way that the solution of the coarse grid problem is not required in the application of the system matrix, but in the preconditioner. This enables the use of inexact methods like geometric or algebraic multigrid, see, e.g., [32,36,38–40]. Moreover, inexact solvers can also be used in the scaled Dirichlet preconditioner and, if using the saddle point formulation, also for the local solvers, cf., [39], see also [32,33] and references therein for alternative approaches by means of hybrid FETI. FETI-DP methods have also been successfully applied to non-linear problems by means of a non-linear version of FETI-DP. We want to highlight recent advances presented, e.g., in [36,38,41], showing excellent scalability on large-scale supercomputers.

In the present paper, we consider the following second-order elliptic boundary value problem in a bounded Lipschitz domain $\Omega \subset \mathbb{R}^d$, with $d \in \{2, 3\}$: Find $u : \overline{\Omega} \to \mathbb{R}$ such that

$$-\operatorname{div}(\alpha \nabla u) = f \quad \text{in } \Omega,\ u = 0 \text{ on } \Gamma_D, \quad \text{and} \quad \alpha \frac{\partial u}{\partial n} = g_N \text{ on } \Gamma_N, \tag{1}$$

with given, sufficient smooth data f, g_N and α, where the coefficient α is uniformly bounded from below and above by some positive constants α_{min} and α_{max}, respectively. The boundary $\partial\Omega$ of the computational domain Ω consists of a Dirichlet part Γ_D of positive boundary measure and a Neumann part Γ_N. Furthermore, we assume that the Dirichlet boundary Γ_D is always a union of complete patch sides (edges/face in 2d/3d) which are uniquely defined in IgA. Without loss of generality, we assume homogeneous Dirichlet conditions. This can always be obtained by homogenization. By means of integration by parts, we arrive at the weak formulation of (1) which reads as follows: Find $u \in V_D = \{u \in H^1 : \gamma_0 u = 0 \text{ on } \Gamma_D\}$ such that

$$a(u, v) = \langle F, v \rangle \quad \forall v \in V_D, \tag{2}$$

where γ_0 denotes the trace operator. The bilinear form $a(\cdot, \cdot) : V_D \times V_D \to \mathbb{R}$ and the linear form $\langle F, \cdot \rangle : V_D \to \mathbb{R}$ are given by the expressions

$$a(u, v) := \int_\Omega \alpha \nabla u \cdot \nabla v \, dx \quad \text{and} \quad \langle F, v \rangle := \int_\Omega f v \, dx + \int_{\Gamma_N} g_N v \, ds.$$

The remainder of the paper is organized as follows: In Section 2, we give a short introduction to isogeometric analysis, providing the basic definitions and notations. Section 3 describes the different discretizations of the model problem obtained with the continuous and discontinuous Galerkin methods. In Section 4, we formulate the IETI-DP method for both discretizations and provide implementation details. The way how the algorithm is parallelized is explained in Section 5. Numerical examples are presented in Section 6. Finally we draw some conclusions in Section 7.

Please cite this article in press as: C. Hofer, Parallelization of continuous and discontinuous Galerkin dual–primal isogeometric tearing and interconnecting methods, Computers and Mathematics with Applications (2017), http://dx.doi.org/10.1016/j.camwa.2017.06.051.