ARTICLE IN PRESS

Journal of Computational and Applied Mathematics ▮ (▮▮▮▮) ▮▮▮–▮▮▮

Contents lists available at ScienceDirect

Journal of Computational and Applied Mathematics

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

Robust and scalable domain decomposition solvers for unfitted finite element methods☆

Santiago Badia *, Francesc Verdugo

Universitat Politècnica de Catalunya, Jordi Girona 1-3, Edifici C1, E-08034 Barcelona, Spain
Centre Internacional de Mètodes Numèrics en Enginyeria, Parc Mediterrani de la Tecnologia, Esteve Terrades 5, E-08860 Castelldefels, Spain

ARTICLE INFO

Article history:
Received 16 March 2017
Received in revised form 24 July 2017

Keywords:
Unfitted finite elements
Embedded boundary methods
Linear solvers
Parallel computing
Domain decomposition

ABSTRACT

Unfitted finite element methods, e.g., extended finite element techniques or the so-called finite cell method, have a great potential for large scale simulations, since they avoid the generation of body-fitted meshes and the use of graph partitioning techniques, two main bottlenecks for problems with non-trivial geometries. However, the linear systems that arise from these discretizations can be much more ill-conditioned, due to the so-called small cut cell problem. The state-of-the-art approach is to rely on sparse direct methods, which have quadratic complexity and are thus not well suited for large scale simulations. In order to solve this situation, in this work we investigate the use of domain decomposition preconditioners (balancing domain decomposition by constraints) for unfitted methods. We observe that a straightforward application of these preconditioners to the unfitted case has a very poor behavior. As a result, we propose a customization of the classical BDDC methods based on the stiffness weighting operator and an improved definition of the coarse degrees of freedom in the definition of the preconditioner. These changes lead to a robust and algorithmically scalable solver able to deal with unfitted grids. A complete set of complex 3D numerical experiments shows the good performance of the proposed preconditioners.

1. Introduction

The use of unfitted finite element methods (FEMs) is an appealing approach for different reasons. They are interesting in coupled problems that involve interfaces (e.g., fluid–structure interaction [1] or free surface flows), or situations in which one wants to avoid the generation of body-fitted meshes. This type of techniques have been named in different ways. Unfitted FEMs for capturing interfaces are usually denoted as eXtended FEM (XFEM) [2], whereas these techniques are usually denoted as embedded (or immersed) boundary methods, when the motivation is to simulate a problem using a (usually simple Cartesian) background mesh. Recently, different realizations of the method have been coined in different ways, depending on the way the numerical integration is performed, how Dirichlet boundary conditions are enforced on no-matching surfaces, or the type of stabilization, if any, being used. To mention some examples, the finite cell method combines

☆ This work has partially been funded by the European Research Council under the Proof of Concept Grant No. 737439 NuWaSim—On a Nuclear Waste Deep Repository Simulator and the EC-International Cooperation in Aeronautics with China (Horizon 2020) under the project: Efficient Manufacturing for Aerospace Components USing Additive Manufacturing, Net Shape HIP and Investment Casting (EMUSIC). SB gratefully acknowledges the support received from the Catalan Government through the ICREA Acadèmia Research Program.

* Corresponding author at: Centre Internacional de Mètodes Numèrics en Enginyeria, Parc Mediterrani de la Tecnologia, Esteve Terrades 5, E-08860 Castelldefels, Spain.

E-mail addresses: sbadia@cimne.upc.edu (S. Badia), fverdugo@cimne.upc.edu (F. Verdugo).

https://doi.org/10.1016/j.cam.2017.09.034

an XFEM-type functional space, a numerical integration based on adaptive cell refinement and full sub-cell integration, to perform integration on cut cells, and a Nitsche weak imposition of boundary conditions [3]. CutFEM makes also use of Nitsche's method, but includes additional "ghost penalty" stabilization terms to improve the stability of the algorithm [4]. The huge success of isogeometrical analysis methods (spline-based discretization) and the severe limitations of this approach in complex 3D geometries will probably increase the interest of unfitted methods in the near future [5].

The main showstopper up to now for the successful application of unfitted methods for realistic applications is the linear solver step. The condition number of the resulting linear system does not only depend on the characteristic size of the background mesh elements, but also on the characteristic size of the cut elements; cut elements can be arbitrarily small and can have arbitrarily high aspect ratios for small cut cell situations. Enforcing some reasonable thresholds, one can use robust sparse direct solvers for these problems. However, sparse direct methods are very expensive, due to their quadratic complexity, which is especially dramatic at large scales. Scalability is also hard to get, especially at very large scales. For large scale applications with body-fitted meshes in CSE, iterative solvers, usually Krylov solvers combined with preconditioners, is the natural way to go. Scalable preconditioners at large scales are usually based on algebraic multigrid (AMG) and domain decomposition solvers [6–9]. Unfortunately, the lack of robust and scalable iterative solvers for unfitted FEM has limited the applicability of unfitted methods in real applications. It is well-known that some rudimentary methods, e.g., Jacobi preconditioners, can solve the issue of the ill-conditioning that comes from cut elements. However, these preconditioners are neither scalable nor optimal. Different serial linear solvers for unfitted FEMs have been recently proposed (see, e.g., [10–13]). The methods in [10,11] consider a segregation of nodes into healthy and ill nodes, and a domain decomposition solver for the ill nodes. The method is only applied to 2D problems in [10] and serial computations, and the domain decomposition solver is not scalable, due to the fact that no coarse correction is proposed. The method in [11] proposes to use an AMG method for the healthy nodes. A specific-purpose serial AMG solver is designed in [12] and a serial incomplete factorization solver is considered in [13].

Motivated by the lack of robust (with respect to the element cuts) and scalable parallel solvers for unfitted methods, we develop in this work a domain decomposition solver with these desired properties based on the balancing domain decomposition by constraints (BDDC) framework. First, we provide a short description of the mathematical analysis that proves the scalability of BDDC methods on body-fitted meshes. Next, we show an example that proves that the use of standard BDDC methods cannot be robust with respect to the element cuts. In fact, the method performs poorly when straightforwardly applied to unfitted meshes. Next, we propose (based on heuristic arguments and numerical experimentation, but motivated from the mathematical body of domain decomposition methods) a modified BDDC method. First, we consider stiffness weighting operator (already proposed in the original BDDC article [14]), relying on the diagonal of the (sub-assembled stiffness matrix). The use of this weighting operator proves to be essential for the robustness of the solver. Next, we consider enhanced coarse spaces, with a sub-partition of the edge constraints. Since no mathematical analysis is available for domain decomposition methods with stiffness weighting, we have performed a comprehensive set of numerical experiments on 3D geometries with different levels of complexity. The methods are not only robust and algorithmically weakly scalable, but what is more surprising, the number of iterations seems to be independent of the domain shape too. Furthermore, the cost of the modified weighted operator is identical to the standard one, and the algorithm has the same building blocks as the standard BDDC preconditioner. This is good, since one can use the extremely scalable implementation of these methods, e.g., in the FEMPAR library [15,16] or in the PETSc library [17]. In any case, the cost of the new method can differ from the standard one in the number of coarse degrees of freedom (DOFs). We have observed that in practice, the CPU cost of the modified formulation is very close to the one of the standard preconditioner. In fact, as we increase the size of the coarse problem of the modified solver (which is robust for unfitted methods) tends to the one of the standard preconditioner (which performs very bad for unfitted meshes). We note that the preconditioner proposed herein could also be readily applied to XFEM-enriched interface problems and other discretization techniques (e.g., discontinuous Galerkin methods).

Let us describe the outline of this work. In Section 2 we show the unfitted FE method considered in this paper, in particular our choice to integrate in cut cells and the surface intersection algorithm. (In any case, the preconditioners proposed later on do not depend on these choices, and can be used in other situations). We also state the model problem, its discretization, and the mesh and subdomain partitions. The standard BDDC preconditioner and its building blocks for body-fitted meshes are presented in Section 3. In Section 4, we show a breakdown example of the preconditioner on unfitted meshes, an expensive solution that is provably robust, and a cheap solution based on the stiffness weighting and (optionally) two slightly larger coarse spaces. A complete set of complex 3D numerical experiments is included in Section 5 to show experimentally the good performance of the proposed preconditioner. Finally, some conclusions are drawn and future work is described in Section 6.

2. Unfitted FE method

2.1. Cell partition and subdomain partition

Let $\Omega \subset \mathbb{R}^d$ be an open bounded polygonal domain, with $d \in \{2, 3\}$ the number of spatial dimensions. For the sake of simplicity and without loss of generality, we consider that the domain boundary is defined as the zero level-set of a given scalar function ϕ^{ls}, namely $\partial\Omega \doteq \{x \in \mathbb{R}^d : \phi^{\mathrm{ls}}(x) = 0\}$. We note that the problem geometry could be described using 3D CAD data instead of level-set functions, by providing techniques to compute the intersection between cell edges and surfaces (see, e.g., [18]). In any case, the way the geometry is handled does not affect the domain decomposition solver presented