



HAL
open science

PHYSICS-BASED BALANCING DOMAIN DECOMPOSITION BY CONSTRAINTS FOR HETEROGENEOUS PROBLEMS

Santiago Badia, Alberto F Martín, Hieu Nguyen

► **To cite this version:**

Santiago Badia, Alberto F Martín, Hieu Nguyen. PHYSICS-BASED BALANCING DOMAIN DECOMPOSITION BY CONSTRAINTS FOR HETEROGENEOUS PROBLEMS. 2016. hal-01337968v3

HAL Id: hal-01337968

<https://hal.science/hal-01337968v3>

Preprint submitted on 23 Nov 2016 (v3), last revised 27 Nov 2018 (v5)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

PHYSICS-BASED BALANCING DOMAIN DECOMPOSITION BY CONSTRAINTS FOR HETEROGENEOUS PROBLEMS *

SANTIAGO BADIA^{†‡}, ALBERTO F. MARTÍN^{‡†}, AND HIEU NGUYEN^{§‡}

Abstract. In this work, we present a novel balancing domain decomposition by constraints preconditioner that is robust for multi-material and heterogeneous problems. We start with a well-balanced subdomain partition and, based on an aggregation of elements according to their physical coefficients, we end up with a finer *physics-based* (PB) subdomain partition. Next, we define geometrical objects (corners, edges, and faces) for this PB partition, and select some of them to enforce subdomain continuity (primal objects). When the physical coefficient in each PB subdomain is constant and the primal objects satisfy a mild condition on the existence of acceptable paths, we can show both theoretically and numerically that the condition number does not depend on the contrast of the coefficient. The constant coefficient condition is computationally feasible for multi-material problems. However, for highly heterogeneous problems, such restriction might result into a large coarse problem. In this case, we propose a relaxed version of the method where we only require that the maximal contrast of the physical coefficient in each PB subdomain is smaller than a predefined threshold. The threshold can be chosen so that the condition number is reasonably small while the size of the coarse problem is not too large. An extensive set of numerical experiments is provided to support our findings. In particular, we show a robustness and a weak scalability analysis up to 8000 cores of the new preconditioner when applied to a 3D heterogeneous problem with more than 260 million degrees of freedom. For the scalability analysis, we have exploited a highly scalable advanced inter-level overlapped implementation of the preconditioner that deals very efficiently with the coarse problem computation.

Key words. BDDC, heterogeneous problem, adaptive coarse space, parallel solver, parallel preconditioner

AMS subject classifications. 65N55, 65N22, 65F08

1. Introduction. Many realistic simulations in science and engineering, such as subsurface flow simulations in a nuclear waste repository or in an oil reservoir, or heat conduction in composites, involve heterogeneous materials. The linear systems resulting from the discretization of these problems are hard to solve. The use of direct solvers at a sufficiently fine scale can be prohibitively expensive, even with modern supercomputers, due to their high complexity and scalability issues. In addition, the high contrast of the physical properties significantly increases the condition number of the resulting linear systems, posing great challenges for iterative solvers. In this work, we will focus on developing a domain decomposition (DD) preconditioner that is robust with the variation of the coefficients of the PDEs. For a different but related

[†]Universitat Politècnica de Catalunya, Jordi Girona 1-3, Edifici C1, 08034 Barcelona, Spain (sbadia@cimne.upc.edu).

[‡]CIMNE – Centre Internacional de Mètodes Numèrics en Enginyeria, Parc Mediterrani de la Tecnologia, UPC, Esteve Terradas 5, 08860 Castelldefels, Spain (amartin@cimne.upc.edu).

[§]Institute of Research and Development, Duy Tan University, 3 Quang Trung, Danang, Vietnam (nguyentrunghieu14@dtu.edu.vn).

*This work has partially been funded by the European Research Council under the FP7 Program Ideas through the Starting Grant No. 258443 - COMFUS: Computational Methods for Fusion Technology and the FP7 NUMEXAS project under grant agreement 611636. Financial support from 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) and the H2020-FoF-2015 under the project: Computer Aided Technologies for Additive Manufacturing (CAxMan) are also acknowledged. S. Badia gratefully acknowledges the support received from the Catalan Government through the ICREA Acadèmia Research Program. Finally, the authors thankfully acknowledge the computer resources at Marenostrum III and the technical support provided by BSC under the RES (Spanish Supercomputing Network).

approach to find a reasonably accurate heterogeneous solution on a coarse mesh, we refer interested readers to [19, 1] and references therein.

DD is one of the most popular approaches to solve large-scale problems on parallel supercomputers. It splits a problem into weakly coupled subproblems on smaller subdomains and use parallel local solutions on these subdomains to form a parallel preconditioner for the original problem [53, 44]. In DD, the coarse space plays an important role in achieving scalability as well as robustness with respect to variations in the coefficient. Many early DD methods, such as those in [11, 18, 17, 33, 56], are robust for heterogeneous problems when the subdomain partition is a geometric coarse grid that resolves the discontinuities in the properties of the media. This is a strong requirement, since the properties of the media might have complicated variations on many scales and be difficult to capture by a geometric coarse grid. Further, it is impractical, since it would not lead to load-balanced partitions with a reduced interface.

Recently, there have been works on coarse grids that do not resolve the heterogeneity in the media [26, 48, 26, 46, 47], and especially automatic coarse spaces that adapt to the variation in the properties of the media [23, 24, 45, 50, 15, 52, 51, 29, 28, 30, 38, 25]. In the latter, the coarse spaces are constructed from eigenfunctions associated with small eigenvalues (low-frequency modes) of appropriate generalised eigenvalue problems. This approach is backed up by rigorous mathematical theory and has been numerically shown to be robust for general heterogeneous problems. However, solving eigenvalue problems is expensive and extra implementation effort is required as coarse spaces in DD methods are not naturally formulated as eigenfunctions. Another approach is to use the deluxe scaling technique where local auxiliary Dirichlet problems are solved to compute efficient averaging operators [35, 14, 55]. The approach yields robust DD methods, but it incurs extra implementation and computation cost due to the auxiliary problems. In this paper, we formulate a *new balancing DD by constraints (BDDC) preconditioner that requires no eigenvalue or auxiliary problem and is very robust with the contrast of the coefficient*. The main motivation behind this work is to achieve such goal while *maintaining the simplicity* of the BDDC preconditioner.

The BDDC method was introduced by Dohrmann in 2003 [12]. It is an improved version of the balancing DD (BDD) method by Mandel [40] and has a close connection with the FETI-DP method [21, 20]. In fact, it can be shown that the eigenvalues of the preconditioned operators associated with BDDC and FETI-DP are almost identical [42, 36, 10]. The BDDC method is particularly well suited for extreme scale simulations, since it allows for a very aggressive coarsening, the computations at different levels can be computed in parallel, the subdomain problems can be solved inexactly [13, 37] by, e.g., one AMG cycle, and it can straightforwardly be extended to multiple levels [54, 43]. All of these properties have been carefully exploited in the series of articles [3, 4, 5, 6] where an extremely scalable implementation of these algorithms has been proposed, leading to excellent weak scalability on nearly half a million cores in its multilevel version.

Our new BDDC method is motivated from the fact that non-overlapping DD methods, such as BDDC and FETI-DP, are robust with the variation and contrast of the coefficient if it is constant (or varies mildly) in each subdomain [33, 32, 53]. This implies that in order to have robustness for BDDC methods one could use a physics-based (PB) partition obtained by aggregating elements of the same coefficient value. However, using this type of partition is impractical as the number of the subdomains might be too large and can lead to a poor load balancing among subdomains and large

interfaces. In order to solve this dilemma, we propose to use a well-balanced partition, e.g., one obtained from an automatic graph partitioner like METIS [27], to distribute the workload among processors. Then, we consider a sub-partition of subdomains based on the physical coefficients, leading to a PB partition. Continuity constraints among subdomains will be defined through the definition of objects based on the PB partition. Consequently, the interface objects are adaptively defined according to the variation of the coefficient. The resulting BDDC preconditioner with constraints imposed on subfaces, subedges, and vertices, will be called PB-BDDC. These ideas can readily be applied to FETI-DP preconditioners.

For multi-material problems, e.g., problems with isolated channels or inclusions, it is computationally feasible to require the physical coefficient in each PB subdomain to be constant. In this situation, we are able to prove that *the new BDDC method is scalable and its convergence is independent of the contrast of the coefficient*. An important feature of our result is that not all PB objects are necessary and any selected set of PB objects satisfying a mild condition on the so-called acceptable paths is enough to guarantee robustness.

For heterogeneous problems with a wide spectrum of values in a small spatial scale the requirement on the coefficient is too strong and might result in too many coarse objects (large coarse problem). As a result, we also propose a relaxed definition of the PB partition where we only require that the maximal contrast of the physical coefficient in each PB subdomain is smaller than a predefined threshold. The threshold can be chosen so that the condition number is reasonably small while the size of the coarse problem is not too large. We empirically show that this relaxed version of PB-BDDC, called rPB-BDDC, is robust and efficient for different difficult distributions of the coefficient.

By definition, (r)PB-BDDC methods (as well as any other adaptive coarse space technique) start with a coarse space that is enough to have a non-singular preconditioner and enlarges it to make it robust for heterogeneities. In order to alleviate the effect of potentially larger coarse spaces compared to homogeneous problems, we propose two remedies:

1. In [subsection 3.5](#), taking advantage of the fact that not all PB objects are required for robustness, we study how to keep the size of the coarse problem close to minimal. For the sake of illustration, we present some examples with different distribution of the coefficient and the corresponding minimal set of PB objects required for robustness. The number of required PB objects is the same as the number of extra eigenfunctions needed in other approaches for adaptive coarse spaces based on eigenvalue problems [51, 29, 25]. For general cases, a simple procedure to construct a close to minimal coarse space is also provided. The procedure is based on a mathematically supported definition of *acceptable paths*.
2. For highly heterogeneous problems, optimal robust coarse spaces can still be large. In order to cope larger coarse spaces and keep good scalability properties, we have considered an implementation of the PB-BDDC method in the extremely scalable BDDC code in FEMPAR [3, 4, 5, 6]. It exploits inter-level overlapping of computations and communications so that coarse tasks run in parallel and can be masked in run time by fine tasks. The implementation also provides a recursive multilevel extension which deals efficiently with large coarse spaces. This is particularly important for the (r)PB-BDDC methods presented herein. In [subsection 4.6](#), we show excellent scalability results up to 8000 processors and more than 260 million unknowns

for a 3D problem.

Summarizing, the (r)PB-BDDC method does not require to solve eigenvalue or auxiliary problems on patches of subdomains, which could incur high computational cost and the formulation and implementation of the (r)PB-BDDC methods are very much the same as for the standard BDDC method. The only difference is in identifying and defining BDDC objects to impose constraints. In other words, *the simplicity of the standard BDDC method is maintained*. On the other hand, the (r)PB-BDDC methods involve a richer interface with the application software, e.g., access to the physical properties of the problem.

The rest of the paper is organised as follows. In [section 2](#), we introduce the model problem, the domain partitions and the BDDC object classification. In [section 3](#), we present the formulation of the (r)PB-BDDC methods as well as its key ingredients, namely coarse degrees of freedom (coarse DOFs), weighting and harmonic extension operators. The convergence analysis is also provided in this section. In [section 4](#), we provide an extensive set of numerical experiments to demonstrate the robustness and efficiency of the (r)PB-BDDC methods. We finally draw some conclusions in [section 5](#).

2. Problem setting. Let $\Omega \subset \mathbb{R}^d$, with $d = 2, 3$ being the space dimension, be a bounded polyhedral domain. For a model problem, we study the Poisson's equation with non-constant diffusion and homogeneous Dirichlet conditions (the non-homogeneous case only involves an obvious modification of the right-hand side). Thus, the problem at hand is: find $u \in H_0^1(\Omega)$ such that $-\alpha\Delta u = f$ in $H^{-1}(\Omega)$ sense, with $f \in H^{-1}(\Omega)$ and $\alpha \in L^\infty(\Omega)$ strictly positive. The weak form of the problem reads as: find $u^* \in H_0^1(\Omega)$ such that

$$(1) \quad \int_{\Omega} \alpha \nabla u^* \cdot \nabla v \, dx = \int_{\Omega} f v \, dx, \quad \text{for any } v \in H_0^1(\Omega).$$

Let \mathcal{T} be a shape-regular quasi-uniform mesh of Ω with characteristic size h . It can consist of tetrahedra or hexahedra for $d = 3$, or triangles or quadrilaterals for $d = 2$. For simplicity of exposition, we assume that α is constant on each element $\tau \in \mathcal{T}$.

2.1. Domain partitions. We first consider a partition Θ of the domain Ω into non-overlapping open subdomains. This partition must be driven by computational efficiency in distributed memory platforms, i.e., it should have a reduced interface size and lead to a well-balanced distribution of workload among processors. In a parallel implementation, each subdomain in Θ is generally assigned to a processor. We further assume that every $\mathcal{D} \in \Theta$ can be obtained by *aggregation of elements* in \mathcal{T} and is connected. We denote by $\Gamma(\Theta)$ the interface of the partition Θ , i.e., $\Gamma(\Theta) \doteq (\cup_{\mathcal{D} \in \Theta} \partial \mathcal{D}) \setminus \partial \Omega$, and by $\Gamma_h(\Theta)$ the discrete version of the interface.

We also consider a PB subdomain partition. This partition is used later in the new definition of coarse objects and in the analysis. It is, however, not used for work distribution. Given a subdomain $\mathcal{D} \in \Theta$, we can further consider its partition $\Theta_{\text{pb}}(\mathcal{D})$ into a set of “sub-subdomains” with constant α . Clearly, the resulting global PB partitions $\Theta_{\text{pb}} \doteq \{\Theta_{\text{pb}}(\mathcal{D})\}_{\mathcal{D} \in \Theta}$ is also a partition of Ω (into PB subdomains). The interface of this partition is $\Gamma(\Theta_{\text{pb}}) \doteq (\cup_{\hat{\mathcal{D}} \in \Theta_{\text{pb}}} \partial \hat{\mathcal{D}}) \setminus \partial \Omega$. For a subdomain $\mathcal{D} \in \Theta$ (analogously for $\hat{\mathcal{D}} \in \Theta_{\text{pb}}$), we denote by $\mathcal{T}_{\mathcal{D}}$ the submesh of \mathcal{T} associated with \mathcal{D} , $\mathcal{T}_{\hat{\mathcal{D}}} \doteq \{\tau \in \mathcal{T} : \tau \subset \mathcal{D}\} \subset \mathcal{T}$. For any $\hat{\mathcal{D}} \in \Theta_{\text{pb}}$, let $\omega(\hat{\mathcal{D}})$ be the only subdomain in Θ that contains $\hat{\mathcal{D}}$. In [Figure 1](#), we show an example of the original partition Θ and the PB partition Θ_{pb} for a simple problem. The meaning of $\Theta_{\text{pb}}(\mathcal{D})$ and $\omega(\mathcal{D})$ is also

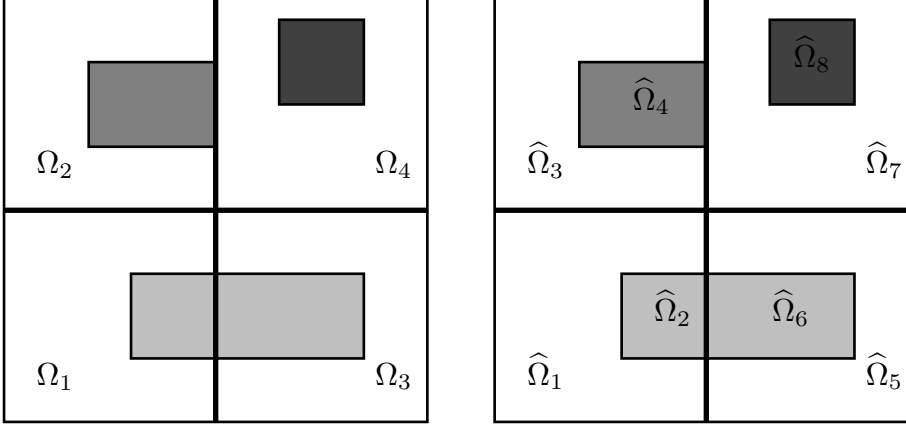


Fig. 1: An example of an original partition Θ (left) and a physics-based partition Θ_{pb} (right) of a square domain where different colors represent different values of α . On the left, we have a $\Theta = \{\Omega_1, \Omega_2, \Omega_3, \Omega_4\}$. On the right, we show the corresponding PB-partition for every subdomain in Θ : $\Theta_{\text{pb}}(\Omega_1) = \{\hat{\Omega}_1, \hat{\Omega}_2\}$, $\Theta_{\text{pb}}(\Omega_2) = \{\hat{\Omega}_3, \hat{\Omega}_4\}$, $\Theta_{\text{pb}}(\Omega_3) = \{\hat{\Omega}_5, \hat{\Omega}_6\}$, and $\Theta_{\text{pb}}(\Omega_4) = \{\hat{\Omega}_7, \hat{\Omega}_8\}$. The complete PB-partition is $\Theta_{\text{pb}} = \{\hat{\Omega}_1, \dots, \hat{\Omega}_8\}$. Further, we have $\omega(\hat{\Omega}_1) = \omega(\hat{\Omega}_2) = \Omega_1$, $\omega(\hat{\Omega}_3) = \omega(\hat{\Omega}_4) = \Omega_2$, $\omega(\hat{\Omega}_5) = \omega(\hat{\Omega}_6) = \Omega_3$, $\omega(\hat{\Omega}_7) = \omega(\hat{\Omega}_8) = \Omega_4$.

illustrated.

2.2. Finite element spaces. Let us perform a discretization of (1) by a continuous finite element (FE) space $\bar{\mathbb{V}}$ associated with the mesh \mathcal{T} . The discontinuous Galerkin (DG) case will not be considered in this work, but we refer the reader to [16] for more information.

For every subdomain $\mathcal{D} \in \Theta$, we consider a FE space $\mathbb{V}_{\mathcal{D}}$ associated with the local mesh $\mathcal{T}_{\mathcal{D}}$. Let $H(\mathcal{D})$ be the characteristic length of the subdomain \mathcal{D} and $h(\mathcal{D})$ be the characteristic length of the FE mesh $\mathcal{T}_{\mathcal{D}}$. We define the Cartesian product of local FE spaces as $\mathbb{V} = \prod_{\mathcal{D} \in \Theta} \mathbb{V}_{\mathcal{D}}$. We note that functions in this space are allowed to be discontinuous across the interface $\Gamma(\Theta)$. Clearly, $\bar{\mathbb{V}} \subset \mathbb{V}$.

For a subdomain $\mathcal{D} \in \Theta$, we also define the subdomain FE operator $\mathcal{A}_{\mathcal{D}} : \mathbb{V}_{\mathcal{D}} \rightarrow \mathbb{V}'_{\mathcal{D}}$ as $\langle \mathcal{A}_{\mathcal{D}}u, v \rangle \doteq \int_{\mathcal{D}} \alpha \nabla u \cdot \nabla v \, dx$, for all $u, v \in \mathbb{V}_{\mathcal{D}}$, and the sub-assembled operator $\mathcal{A}^{\Theta} : \mathbb{V} \rightarrow \mathbb{V}'$ as $\langle \mathcal{A}^{\Theta}u, v \rangle \doteq \sum_{\mathcal{D} \in \Theta} \langle \mathcal{A}_{\mathcal{D}}u, v \rangle$, for all $u, v \in \mathbb{V}$.

A function $u \in \mathbb{V}_{\mathcal{D}}$ is said to be discrete α -harmonic in \mathcal{D} if

$$\langle \mathcal{A}_{\mathcal{D}}u, v \rangle = 0, \quad \text{for any } v \in \mathbb{V}_{0,\mathcal{D}},$$

where $\mathbb{V}_{0,\mathcal{D}} \doteq \{v \in \mathbb{V}_{\mathcal{D}} : v = 0 \text{ on } \partial\mathcal{D}\}$. It should be noted that if u is discrete α -harmonic in \mathcal{D} then it satisfies the energy minimising property, namely

$$\langle \mathcal{A}_{\mathcal{D}}u, u \rangle \leq \langle \mathcal{A}_{\mathcal{D}}v, v \rangle, \quad \forall v \in \mathbb{V}_{\mathcal{D}}, v|_{\partial\mathcal{D}} = u|_{\partial\mathcal{D}}.$$

In addition, we consider the assembled operator $\mathcal{A} : \bar{\mathbb{V}} \rightarrow \bar{\mathbb{V}}'$, defined by $\langle \mathcal{A}u, v \rangle = \int_{\Omega} \alpha \nabla u \cdot \nabla v \, dx$, for all $u, v \in \bar{\mathbb{V}}$. This operator is the Galerkin projection of \mathcal{A}^{Θ} onto $\bar{\mathbb{V}}$. We want to compute a FE approximation $u \in \bar{\mathbb{V}}$ of u^* in (1) such that

$$(2) \quad \langle \mathcal{A}u, v \rangle = \langle f, v \rangle, \quad \text{for any } v \in \bar{\mathbb{V}}.$$

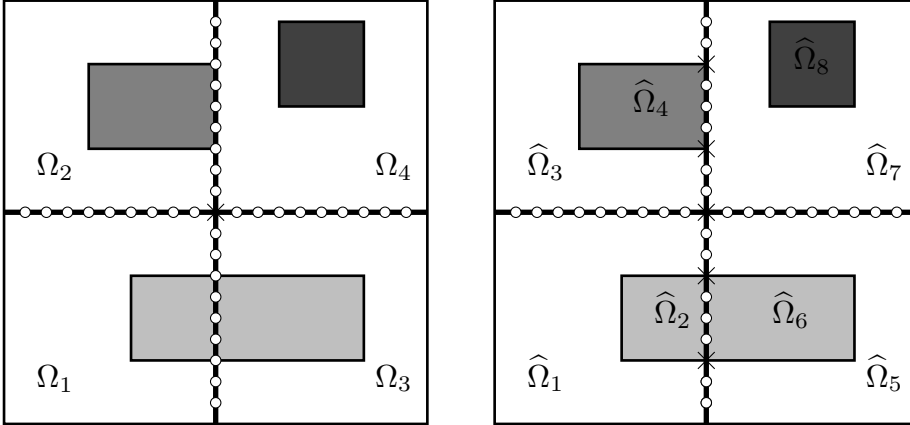


Fig. 2: An example of how FE nodes (on the interface of the original partition Θ in Figure 1) are classified in the standard way (left) using neigh_{Θ} , and in the physics-based way (right) using $\text{neigh}_{\Theta_{\text{pb}}}$. Corner nodes are marked with crosses while nodes in edges are marked with small circles. Using the standard classification, on the left, we obtain $\Lambda(\Theta)$ with one corner and four edges. With the new classification, on the right, we have $\Lambda_{\text{pb}}(\Theta)$ with five corners and six edges (eight edges if we only consider connected objects).

2.3. Object classification. This subsection concerns with objects on subdomain interfaces and their classification. It provides foundations for the definition of coarse DOFs in BDDC methods later on.

Given a subdomain partition Θ , and a point $\xi \in \Gamma(\Theta)$, let us denote by $\text{neigh}_{\Theta}(\xi)$ the set of subdomains in Θ that contain ξ . We can introduce the concept of objects as a classification of points in $\Gamma(\Theta)$. A *geometrical object* is a maximal set λ of points in $\Gamma(\Theta)$ with identical subdomain set. We denote by $\text{neigh}_{\Theta}(\lambda)$ the set of subdomains in Θ containing λ . It should be noted that the set of all geometrical objects, denoted by $\Lambda(\Theta)$, is a partition of $\Gamma(\Theta)$.

Remark 1. Since the set of points in the interface is infinite, the previous classification of $\Gamma(\Theta)$ into geometrical objects is performed in practice by the classification of vertices, edges, and faces of elements in the mesh \mathcal{T} based on their subdomain set.

Denote by $\text{ndof}(\lambda)$ the number of DOFs belonging to λ . We further consider the following standard classification of geometrical objects. In the three-dimensional case, $\lambda \in \Lambda(\Theta)$ is a *face* if $|\text{neigh}_{\Theta}(\lambda)| = 2$ and $\text{ndof}(\lambda) > 1$, is an *edge* if $|\text{neigh}_{\Theta}(\lambda)| > 2$ and $\text{ndof}(\lambda) > 1$, and is a *corner* if $\text{ndof}(\lambda) = 1$. In the two-dimensional case, $\lambda \in \Lambda(\Theta)$ is an *edge* if $|\text{neigh}_{\Theta}(\lambda)| = 2$ and $\text{ndof}(\lambda) > 1$, and is a *corner* if $\text{ndof}(\lambda) = 1$. In the literature, e.g., [33, 53], corners are also referred to as vertices. Analogous definitions are also used frequently for FETI-DP methods (see [53]). In Figure 2 (left), an illustration of this classification is shown for a simple example.

In the next step, we define PB objects, which is the main ingredient of the PB-BDDC methods proposed herein. We consider the set of objects $\Lambda_{\text{pb}}(\Theta)$ obtained by applying the previous classification of $\Gamma(\Theta)$ into corners/edges/faces but with $\text{neigh}_{\Theta}(\cdot)$ replaced by $\text{neigh}_{\Theta_{\text{pb}}}(\cdot)$. In other words, we use sets of subdomains in Θ_{pb} to classify geometrical objects on $\Gamma(\Theta)$. Figure 2 (right) shows objects in $\Lambda_{\text{pb}}(\Theta)$ for a simple example.

LEMMA 2. $\Lambda_{\text{pb}}(\Theta)$ is a refinement of $\Lambda(\Theta)$.

Proof. The statement holds if for every object $\lambda_{\text{pb}} \in \Lambda_{\text{pb}}(\Theta)$ there exists one and only one object $\lambda \in \Lambda(\Theta)$ containing it. Since all points in λ_{pb} belong to the same set of PB subdomains, $\text{neigh}_{\Theta_{\text{pb}}}(\lambda)$, they are in the same set of subdomains in Θ , namely $\{\omega(\hat{\mathcal{D}})\}_{\hat{\mathcal{D}} \in \text{neigh}_{\Theta_{\text{pb}}}(\lambda)}$. As a result, all these points belong to the same object in $\Lambda(\Theta)$. \square

Remark 3. In some cases, the DOF-based classification into corners, edges, and faces might need some modification in order to ensure well-posedness of the BDDC method with corner constraints only. This usually involves the use of a kernel detection mechanism (see, e.g., [49]). A new approach based on perturbations has recently been proposed in [8, 7], where the method is well-posed in all cases.

3. PB-BDDC preconditioning. In this section, we present the PB-BDDC preconditioner. The basic idea behind BDDC methods is first to define a sub-assembled operator (no assembling among subdomains), and the global space of functions that are fully independent (“discontinuous”) among subdomains. Secondly, we have to define the under-assembled space (the BDDC space) of functions for which continuity among subdomains is enforced only on a set of coarse DOFs. In order to be robust for heterogeneous problems, the PB-BDDC method utilises new definitions of the BDDC space (i.e., new coarse DOF continuity among subdomains) and a new weighting operator.

3.1. Coarse degrees of freedom. Similarly to other BDDC methods, in the PB-BDDC method, some (or all) of the objects in $\Lambda_{\text{pb}}(\Theta)$ are associated with a *coarse* DOF. We denote this set of objects by Λ_O and call it the set of primal or coarse objects. Obviously, $\Lambda_O \subseteq \Lambda_{\text{pb}}(\Theta)$. Typical choices of Λ_O are $\Lambda_O \doteq \Lambda_C$, when only corners are considered, $\Lambda_O \doteq \Lambda_C \cup \Lambda_E$, when corners and edges are considered, or $\Lambda_O \doteq \Lambda_{\text{pb}}(\Theta)$, when corners, edges, and faces are considered. These choices lead to three variants of the PB-BDDC method, referred to as PB-BDDC(c), PB-BDDC(ce) and PB-BDDC(cef), respectively. Figure 2 (right) actually shows the coarse objects of PB-BDDC(ce) for a simple 2D problem.

Given an object $\lambda \in \Lambda_O$, we define its coarse DOF as the mean value on λ . The rigorous definition is as follows. Assume $\lambda \in \Lambda_O$ is associated with a subdomain $\mathcal{D} \in \Theta$. We define the coarse DOF $c_\lambda^{\mathcal{D}}$ corresponding to λ as

$$(3) \quad c_\lambda^{\mathcal{D}}(u_{\mathcal{D}}) \doteq \frac{\int_\lambda u_{\mathcal{D}} ds}{\int_\lambda 1 ds}, \quad \text{for } u_{\mathcal{D}} \in \mathbb{V}_{\mathcal{D}}.$$

Clearly, $c_\lambda^{\mathcal{D}}$ is a functional in $\mathbb{V}'_{\mathcal{D}}$. When λ is a corner, $c_\lambda^{\mathcal{D}}$ is simply the value at that corner. Once we have defined the coarse DOFs, we can define the BDDC space as follows

$$(4) \quad \tilde{\mathbb{V}} \doteq \{v \in \mathbb{V} : c_\lambda^{\mathcal{D}}(v) = c_\lambda^{\mathcal{D}'}(v), \forall \lambda \in \Lambda_O, \forall \mathcal{D}, \mathcal{D}' \in \text{neigh}_\Theta(\lambda)\},$$

i.e., the subspace of functions in \mathbb{V} that are continuous “at” coarse DOFs. Clearly, $\bar{\mathbb{V}} \subset \tilde{\mathbb{V}} \subset \mathbb{V}$.

For BDDC methods, solving the coarse problem is usually the bottleneck (cf. [2, 3, 4, 8]). Therefore, it is of great interest to find a minimal set of coarse objects (the number of the coarse objects is the number of the coarse DOFs and also is the size of the coarse problem), so that BDDC methods can achieve their potential of fast convergence and perfect weak scalability. According to [33, 53], in the case where

the physical coefficient in each subdomain is constant, the set of coarse objects only needs to guarantee the existence of the so-called *acceptable paths*. We need a similar concept here for the PB-BDDC method.

The definition below is modelled after [53, Definition 6.26], [33] and [34]. For the rest of the subsection, we consider the 3D case. The 2D case follows a straightforward restriction where faces become edges and edges become corners.

DEFINITION 4 (Acceptable path). *Let $\Theta_{\text{pb}}^\partial$ be the set of PB subdomains $\hat{\mathcal{D}} \in \Theta_{\text{pb}}$ touching the interface $\Gamma(\Theta)$, i.e., $\partial\hat{\mathcal{D}} \cap \Gamma(\Theta) \neq \emptyset$. For two subdomains $\hat{\mathcal{D}}_a, \hat{\mathcal{D}}_b \in \Theta_{\text{pb}}^\partial$ that share at least one object $\lambda \in \Lambda_{\text{pb}}(\Theta)$, an acceptable path connecting them is a sequence $\{\hat{\mathcal{D}}_a = \hat{\mathcal{D}}_1, \hat{\mathcal{D}}_2, \dots, \hat{\mathcal{D}}_n = \hat{\mathcal{D}}_b\}$ of PB subdomains in Θ_{pb} , which satisfy the following properties:*

- i) All $\hat{\mathcal{D}}_k$, $k = 1, \dots, \hat{n} - 1$, share λ .*
- ii) Subdomains $\hat{\mathcal{D}}_k$ and $\hat{\mathcal{D}}_{k+1}$, $k = 1, \dots, \hat{n} - 1$, must share a primal edge in Λ_O .*
- iii) Their (constant) coefficients satisfy*

$$\text{TOL } \alpha_k \geq R(k) \min(\alpha_a, \alpha_b), \quad 1 \leq k \leq n$$

where TOL is some predefined tolerance and $R(k) = h(\hat{\mathcal{D}}_k)/H(\hat{\mathcal{D}}_k)$ if λ is a corner and $R(k) = 1$ otherwise.

Assumption 5. Given a predefined tolerance TOL for the definition of acceptable paths, we assume that the set of PB-BDDC objects Λ_O satisfies the following properties:

1. For any pair of subdomains $\hat{\mathcal{D}}_a, \hat{\mathcal{D}}_b \in \Theta_{\text{pb}}^\partial$ sharing a face in $\Lambda_{\text{pb}}(\Theta)$, either the face is primal or there is an edge belonging to that face for which there exists an acceptable path between the two subdomains.
2. For all pairs of subdomains $\hat{\mathcal{D}}_a, \hat{\mathcal{D}}_b \in \Theta_{\text{pb}}^\partial$ not sharing a face, but having at least one common object in $\Lambda_{\text{pb}}(\Theta)$, there exists an acceptable path for one of the objects in common.

Remark 6. We note that **Assumption 5** is weaker than the standard one in the literature (see, e.g., [53, Assumption 6.27]). The difference is that, for every face, we do not assume the existence of a primal edge in Λ_O that belongs to its boundary. Instead, we only require the existence of an acceptable path for one edge of the face. This sharper definition is particularly important in PB-BDDC, because it allows one to reduce the size of the coarse space needed. One example is the case in **Figure 5c**, where no primal constraints are needed between PB subdomains with low values of α .

Remark 7. In **Definition 4**, if $\{\hat{\mathcal{D}}_a, \hat{\mathcal{D}}_b\}$ share an object belonging to Λ_O , then there exists a trivial acceptable path $\{\hat{\mathcal{D}}_a, \hat{\mathcal{D}}_b\}$ with $\text{TOL} = 1$ and $n = 2$. Thus, BDDC(ce) and BDDC(cef) always satisfy **Assumption 5** for $\text{TOL} = 1$.

3.2. Restriction operator. Let us define the projection $\mathcal{Q} : \mathbb{V} \rightarrow \bar{\mathbb{V}}$ as some weighted average of interface values together with an α -harmonic extension to subdomain interiors (see, e.g., [41]). We define these ingredients as follows.

For $u \in \mathbb{V}$, $\xi \in \Gamma(\Theta)$ and an associated PB partition Θ_{pb} of Θ , the weighting operator \mathcal{W} is defined as

$$(5) \quad \mathcal{W}u(\xi) \doteq \sum_{\mathcal{D} \in \text{neigh}_\Theta(\xi)} \delta_{\mathcal{D}}^\dagger(\xi) u_{\mathcal{D}}(\xi), \quad \text{with } \delta_{\mathcal{D}}^\dagger(\xi) \doteq \frac{\sum_{\hat{\mathcal{D}} \in \text{neigh}_{\Theta_{\text{pb}}}(\xi) \cap \Theta_{\text{pb}}(\mathcal{D})} \alpha_{\hat{\mathcal{D}}}}{\sum_{\hat{\mathcal{D}} \in \text{neigh}_{\Theta_{\text{pb}}}(\xi)} \alpha_{\hat{\mathcal{D}}}},$$

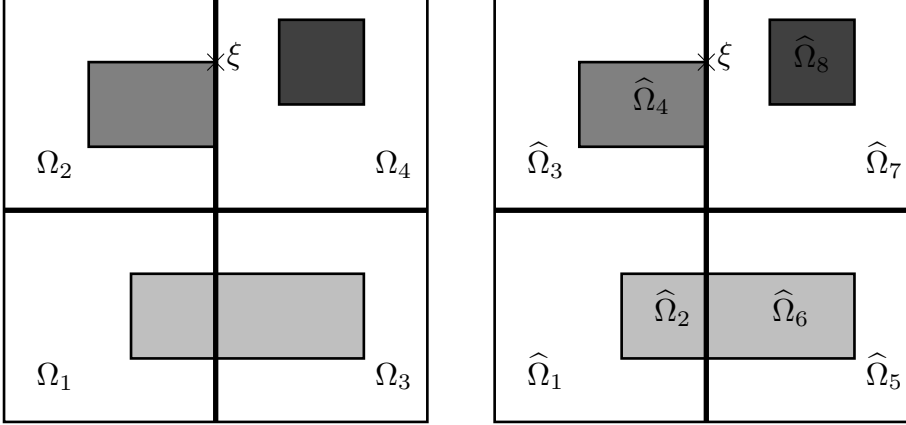


Fig. 3: Illustration for the weighting \mathcal{W} at ξ on the interface $\Gamma_h(\Theta)$ of the original partition (left) with an associated PB partition (right).

where $\alpha_{\hat{\mathcal{D}}}$ denotes the constant coefficient in the PB subdomain $\hat{\mathcal{D}}$.

We note that (5) defines the weighting for points on the original interface $\Gamma(\Theta)$. However, information of the associated PB partition Θ_{pb} is also incorporated. For illustration, the action of \mathcal{W} on $u \in \mathbb{V}$ at ξ shown in Figure 3 is

$$(6) \quad \mathcal{W}u(\xi) = \frac{\hat{\alpha}_3 + \hat{\alpha}_4}{\hat{\alpha}_3 + \hat{\alpha}_4 + \hat{\alpha}_7} u_2(\xi) + \frac{\hat{\alpha}_7}{\hat{\alpha}_3 + \hat{\alpha}_4 + \hat{\alpha}_7} u_4(\xi),$$

where u_i denotes the local ‘‘component’’ of u associated with subdomain Ω_i .

The α -harmonic extension operator \mathcal{E} taking data on the interface $\Gamma_h(\Theta)$ and α -harmonically extending it to each subdomain $\mathcal{D} \in \Theta$ is formally defined as

$$\mathcal{E}u \doteq (1 - \mathcal{A}_0^{-1}\mathcal{A})u,$$

where \mathcal{A}_0 is the Galerkin projection of \mathcal{A} onto the bubble space $\mathbb{V}_0 \doteq \{v \in \mathbb{V} : v = 0 \text{ on } \Gamma(\Theta)\}$.

We finally define $\mathcal{Q} = \mathcal{E}\mathcal{W}$.

3.3. Preconditioner statement. In this subsection, we present the PB-BDDC preconditioner, and describe its set-up and formulation. The PB-BDDC preconditioner is a BDDC preconditioner in which the set of coarse DOFs enforce continuity on a set of PB coarse objects, thus modifying the BDDC space being used. Once one has defined the set of PB coarse objects $\Lambda_{\mathcal{O}}$, the rest of ingredients of the PB-BDDC preconditioner are identical to the ones of a standard BDDC preconditioner. In any case, the definition of the weighting operator introduced in (5) is new.

The BDDC preconditioner is a Schwarz-type preconditioner that combines interior corrections with corrections in the BDDC space (see, e.g., [9, 53]). In case of the PB-BDDC preconditioner, the BDDC correction is expressed as $\mathcal{Q}(\tilde{\mathcal{A}}^\Theta)^{-1}\mathcal{Q}^T$, where $\tilde{\mathcal{A}}^\Theta$ is the Galerkin projection of \mathcal{A}^Θ onto $\tilde{\mathbb{V}}$. More specifically, the PB-BDDC preconditioner reads as follows:

$$\mathcal{B} = \mathcal{A}_0^{-1} + \mathcal{Q}(\tilde{\mathcal{A}}^\Theta)^{-1}\mathcal{Q}^T.$$

Apart from the task of identifying and defining coarse objects, the implementation of the PB-BDDC method is identical to that of the standard BDDC method. We

refer the interested reader to [12, 13, 43, 9] for more details on the formulation of BDDC methods and to [2, 4, 6] for an efficient implementation of BDDC methods on distributed memory machines, which requires much further elaboration.

3.4. Condition number estimates. In order to prove condition number estimates for the PB-BDDC preconditioner, we first need to introduce $\widehat{\mathcal{B}}$, an auxiliary BDDC preconditioner. The definition of this preconditioner follows verbatim that of the PB-BDDC preconditioner. There are only changes in the partition, the weighting, and the coarse objects being used.

The partition that $\widehat{\mathcal{B}}$ is defined on is the PB partition. Given the FE mesh \mathcal{T} , the FE space type, and the PB subdomain partition Θ_{pb} , one can similarly build the FE spaces and operators as in subsection 2.2, leading to the sub-assembled space \mathbb{V}^{pb} and operator $\mathcal{A}^{\Theta_{\text{pb}}}$.

As for the weighting, $\widehat{\mathcal{W}}$, its rigorous formulation is given as follows. For $\hat{u} \in \mathbb{V}^{\text{pb}}$, and $\xi \in \Gamma(\Theta_{\text{pb}})$, the weighting operator $\widehat{\mathcal{W}}$ is defined as

$$(7) \quad \widehat{\mathcal{W}}\hat{u}(\xi) \doteq \sum_{\widehat{\mathcal{D}} \in \text{neigh}_{\Theta_{\text{pb}}}(\xi)} \delta_{\widehat{\mathcal{D}}}^{\dagger}(\xi) \hat{u}_{\widehat{\mathcal{D}}}(\xi), \quad \text{where } \delta_{\widehat{\mathcal{D}}}^{\dagger}(\xi) \doteq \frac{\alpha_{\widehat{\mathcal{D}}}}{\sum_{\widehat{\mathcal{D}} \in \text{neigh}_{\Theta_{\text{pb}}}(\xi)} \alpha_{\widehat{\mathcal{D}}}}.$$

For ξ illustrated in Figure 3, this definition yields

$$(8) \quad \widehat{\mathcal{W}}\hat{u}(\xi) = \frac{\hat{\alpha}_3}{\hat{\alpha}_3 + \hat{\alpha}_4 + \hat{\alpha}_7} \hat{u}_3(\xi) + \frac{\hat{\alpha}_4}{\hat{\alpha}_3 + \hat{\alpha}_4 + \hat{\alpha}_7} \hat{u}_4(\xi) + \frac{\hat{\alpha}_7}{\hat{\alpha}_3 + \hat{\alpha}_4 + \hat{\alpha}_7} \hat{u}_7(\xi),$$

where \hat{u}_i denotes the local ‘‘component’’ of \hat{u} associated with subdomain $\widehat{\Omega}_i$. In other words, $\widehat{\mathcal{W}}$ is nothing but the pseudoinverse of a classical weighted counting function introduced in [33] (see also [53, 6.2.1]).

It is well known that $\delta_{\widehat{\mathcal{D}}}^{\dagger}$ is constant in each (PB) coarse object associated with $\widehat{\mathcal{D}}$ and the following important inequality, cf. [53, (6.19)], holds

$$(9) \quad \alpha_{\widehat{\mathcal{D}}_a} \left(\delta_{\widehat{\mathcal{D}}_b}^{\dagger}(\xi) \right)^2 \leq \min \left(\alpha_{\widehat{\mathcal{D}}_a}, \alpha_{\widehat{\mathcal{D}}_b} \right), \quad \forall \widehat{\mathcal{D}}_a, \widehat{\mathcal{D}}_b \in \text{neigh}_{\Theta_{\text{pb}}}(\xi).$$

LEMMA 8. *The weighting $\widehat{\mathcal{W}}$ and \mathcal{W} satisfy*

$$\widehat{\mathcal{W}}u(\xi) = \mathcal{W}u(\xi) \quad \text{for all } u \in \mathbb{V}, \xi \in \Gamma(\Theta).$$

Proof. The proof comes directly from (5), (7), the facts that $\mathbb{V} \subset \mathbb{V}^{\text{pb}}$ and $u_{\widehat{\mathcal{D}}_a} = u_{\widehat{\mathcal{D}}_b} = u_{\mathcal{D}}$ if $u \in \mathbb{V}$ and $\widehat{\mathcal{D}}_a, \widehat{\mathcal{D}}_b \in \Theta_{\text{pb}}(D)$. \square

With the definition of $\widehat{\mathcal{W}}$ in place, we can define $\widehat{\mathcal{Q}} = \widehat{\mathcal{E}}\widehat{\mathcal{W}}$, where $\widehat{\mathcal{E}}$ is the extension operator taking data on $\Gamma_h(\Theta_{\text{pb}})$ and α -harmonically extending to the interiors of each PB subdomain (cf. subsection 3.2).

The definition of the set of coarse objects of $\widehat{\mathcal{B}}$ requires further elaboration. The set of objects $\Lambda(\Theta_{\text{pb}})$ obtained by applying the classification in subsection 2.3 for the PB subdomain partition Θ_{pb} provides a classification of $\Gamma(\Theta_{\text{pb}}) \supset \Gamma(\Theta)$. We have the following relation between the PB objects $\Lambda_{\text{pb}}(\Theta)$ and the (standard) objects of the PB partition $\Lambda(\Theta_{\text{pb}})$.

LEMMA 9. *All the objects in $\Lambda_{\text{pb}}(\Theta)$ are also in $\Lambda(\Theta_{\text{pb}})$, i.e., $\Lambda_{\text{pb}}(\Theta) \subset \Lambda(\Theta_{\text{pb}})$.*

Proof. Let us consider an object $\lambda_{\text{pb}} \in \Lambda_{\text{pb}}(\Theta)$. In both object partitions $\Lambda_{\text{pb}}(\Theta)$ and $\Lambda(\Theta_{\text{pb}})$, we are using the same criteria, i.e., $\text{neigh}_{\Theta_{\text{pb}}}(\cdot)$, to classify points. The difference is that $\Lambda_{\text{pb}}(\Theta)$ is the result of a classification of points in $\Gamma(\Theta)$ whereas $\Lambda(\Theta_{\text{pb}})$ is obtained from a classification of points in $\Gamma(\Theta_{\text{pb}})$. Since $\Gamma(\Theta) \subset \Gamma(\Theta_{\text{pb}})$, all points in λ_{pb} belong to the same object $\lambda' \in \Lambda(\Theta_{\text{pb}})$. Since λ_{pb} is on the interface $\Gamma(\Theta)$, there exist at least two subdomains $\hat{\mathcal{D}}, \hat{\mathcal{D}}' \in \text{neigh}_{\Theta_{\text{pb}}}(\lambda_{\text{pb}})$ such that $\omega(\hat{\mathcal{D}}) \neq \omega(\hat{\mathcal{D}}')$. Let us assume there is a point $\xi \in \lambda'$ such that $\xi \notin \lambda_{\text{pb}}$. Then, $\xi \in \Gamma(\Theta_{\text{pb}}) \setminus \Gamma(\Theta)$, i.e., it only belongs to one subdomain in Θ . As a result, $\omega(\hat{\mathcal{D}})$ is the same for all $\hat{\mathcal{D}} \in \text{neigh}_{\Theta_{\text{pb}}}(\xi)$. Thus, we have a contradiction, since $\text{neigh}_{\Theta_{\text{pb}}}(\xi)$ cannot be the same as $\text{neigh}_{\Theta_{\text{pb}}}(\lambda_{\text{pb}})$. \square

With the theoretical support from [Lemma 9](#), we can define the set of coarse objects $\hat{\Lambda}_O$ of $\hat{\mathcal{B}}$ as a classification of $\Gamma(\Theta_{\text{pb}})$ as follows. On $\Gamma(\Theta)$, we consider the same set of objects Λ_O used in the PB-BDDC preconditioner, i.e., Λ_C , or $\Lambda_C \cup \Lambda_E$, or $\Lambda_{\text{pb}}(\Theta)$. For the rest of the interface $\Gamma(\Theta_{\text{pb}}) \setminus \Gamma(\Theta)$, we enforce *full continuity* among PB subdomains. It can be understood as treating all FE nodes on $\Gamma(\Theta_{\text{pb}}) \setminus \Gamma(\Theta)$ as corners. Denote this set of objects by $\hat{\Lambda}^*$, we have $\hat{\Lambda}_O = \Lambda_O \cup \hat{\Lambda}^*$. [Figure 4](#) illustrates the partitions and coarse objects of \mathcal{B} and $\hat{\mathcal{B}}$ when $\Lambda_O = \Lambda_C \cup \Lambda_E$.

Remark 10. By construction, the BDDC space $\tilde{\mathbb{V}}^{\text{pb}}$ of the auxiliary BDDC preconditioner $\hat{\mathcal{B}}$ contains the BDDC space $\tilde{\mathbb{V}}$, defined in [\(4\)](#), of the PB-BDDC preconditioner.

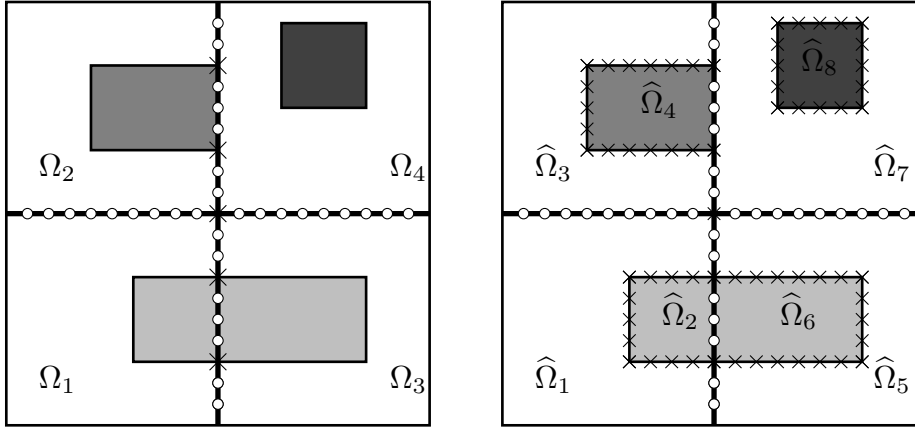


Fig. 4: Partitions and coarse objects of the PB-BDDC preconditioner \mathcal{B} (left) and the auxiliary BDDC preconditioner $\hat{\mathcal{B}}$ (right) when $\Lambda_O = \Lambda_C \cup \Lambda_E$: corner objects are labeled with crosses while nodes of other objects are labeled with circles.

LEMMA 11. *The condition number $\kappa(\mathcal{BA})$ of the PB-BDDC preconditioned operator is bounded by*

$$(10) \quad \kappa(\mathcal{BA}) \leq \max_{v \in \tilde{\mathbb{V}}^{\text{pb}}} \frac{\langle \mathcal{A}^{\Theta_{\text{pb}}} \hat{Q}v, \hat{Q}v \rangle}{\langle \mathcal{A}^{\Theta_{\text{pb}}} v, v \rangle}.$$

Proof. According to [43, Theorem 15] (see also [41]), $\kappa(\mathcal{BA})$ is bounded by

$$(11) \quad \kappa(\mathcal{BA}) \leq \max_{v \in \tilde{\mathbb{V}}} \frac{\langle \mathcal{A}^\Theta \mathcal{Q}v, \mathcal{Q}v \rangle}{\langle \mathcal{A}^\Theta v, v \rangle}.$$

Now we only need to bound the right-hand-side in (11) by the one in (10).

On the one hand, using the fact that $\tilde{\mathbb{V}} \subset \tilde{\mathbb{V}}^{\text{pb}}$ (cf. Remark 10), we have $\langle \mathcal{A}^\Theta v, v \rangle = \langle \mathcal{A}^{\Theta_{\text{pb}}} v, v \rangle$ for all $v \in \tilde{\mathbb{V}}$ because any $v \in \tilde{\mathbb{V}}$ is continuous in each subdomain of Θ . On the other hand, let us prove that the weighting operator $\widehat{\mathcal{W}}$ defined by (5) for Θ_{pb} restricted to \mathbb{V} is identical to the weighting operator \mathcal{W} defined by (5) for Θ . Let us consider a subdomain $\mathcal{D} \in \Theta$ and its PB partition $\Theta_{\text{pb}}(\mathcal{D})$. We have

$$\delta_{\mathcal{D}}^\dagger(\xi) = \sum_{\hat{\mathcal{D}} \in \Theta_{\text{pb}}(\mathcal{D}), \hat{\mathcal{D}} \ni \xi} \delta_{\hat{\mathcal{D}}}^\dagger(\xi),$$

by the definition in (5). For an arbitrary function $v \in \mathbb{V} \subset \mathbb{V}^{\text{pb}}$, we find that

$$\begin{aligned} \mathcal{W}v(\xi) &= \sum_{\mathcal{D} \in \text{neigh}_\Theta(\xi)} \delta_{\mathcal{D}}^\dagger(\xi) v_{\mathcal{D}}(\xi) = \sum_{\mathcal{D} \in \text{neigh}_\Theta(\xi)} \sum_{\hat{\mathcal{D}} \in \Theta_{\text{pb}}, \hat{\mathcal{D}} \ni \xi} \delta_{\hat{\mathcal{D}}}^\dagger(\xi) v_{\mathcal{D}}(\xi) \\ &= \sum_{\hat{\mathcal{D}} \in \text{neigh}_{\Theta_{\text{pb}}}(\xi)} \delta_{\hat{\mathcal{D}}}^\dagger(\xi) v_{\hat{\mathcal{D}}}(\xi) = \widehat{\mathcal{W}}v(\xi). \end{aligned}$$

Therefore, $\widehat{\mathcal{Q}}v$ and $\mathcal{Q}v$ are identical on $\Gamma(\Theta)$ and $\widehat{\mathcal{Q}}v$ is continuous across $\Gamma(\Theta_{\text{pb}})$. In addition, $\mathcal{Q}v$ is discrete α -harmonic in each $\mathcal{D} \in \Theta$ and has minimal energy norm with respect to \mathcal{A}^Θ . As a consequence,

$$\begin{aligned} \langle \mathcal{A}^\Theta \mathcal{Q}v, \mathcal{Q}v \rangle &= \sum_{\mathcal{D} \in \Theta} \langle \mathcal{A}_{\mathcal{D}}^\Theta \mathcal{Q}v, \mathcal{Q}v \rangle \\ &\leq \sum_{\mathcal{D} \in \Theta} \langle \mathcal{A}_{\mathcal{D}}^\Theta \widehat{\mathcal{Q}}v, \widehat{\mathcal{Q}}v \rangle = \sum_{\hat{\mathcal{D}} \in \Theta_{\text{pb}}} \langle \mathcal{A}_{\hat{\mathcal{D}}}^{\Theta_{\text{pb}}} \widehat{\mathcal{Q}}v, \widehat{\mathcal{Q}}v \rangle = \langle \mathcal{A}^{\Theta_{\text{pb}}} \widehat{\mathcal{Q}}v, \widehat{\mathcal{Q}}v \rangle. \end{aligned}$$

This finishes the proof. \square

We could stop here and derive the estimate for $\kappa(\mathcal{BA})$ knowing that the condition number of the auxiliary BDDC preconditioned operator $\widehat{\mathcal{B}}\mathcal{A}$ is estimated by an upper bound of the last quantity on the right of (11). However, we will go a bit further to obtain a stronger result.

LEMMA 12. *Assume that Λ_O is such that Assumption 5 holds. Then we have the following inequality:*

$$(12) \quad \max_{v \in \tilde{\mathbb{V}}^{\text{pb}}} \frac{\langle \mathcal{A}^{\Theta_{\text{pb}}} \widehat{\mathcal{Q}}v, \widehat{\mathcal{Q}}v \rangle}{\langle \mathcal{A}^{\Theta_{\text{pb}}} v, v \rangle} \leq C \max\{1, \text{TOL}\} \max_{\mathcal{D} \in \Theta_{\text{pb}}^o} \left(1 + \log \left(\frac{H(\mathcal{D})}{h(\mathcal{D})} \right) \right)^2,$$

where the constant C is independent of the number of subdomains, $H(\hat{\mathcal{D}})$, $h(\hat{\mathcal{D}})$ and the physical coefficient α .

Proof. By triangle inequality, we have

$$(13) \quad \max_{v \in \tilde{\mathbb{V}}^{\text{pb}}} \frac{\langle \mathcal{A}^{\Theta_{\text{pb}}} \widehat{\mathcal{Q}}v, \widehat{\mathcal{Q}}v \rangle}{\langle \mathcal{A}^{\Theta_{\text{pb}}} v, v \rangle} \leq 1 + \max_{v \in \tilde{\mathbb{V}}^{\text{pb}}} \frac{\langle \mathcal{A}^{\Theta_{\text{pb}}} (\widehat{\mathcal{Q}}v - v), (\widehat{\mathcal{Q}}v - v) \rangle}{\langle \mathcal{A}^{\Theta_{\text{pb}}} v, v \rangle}.$$

Let $w = \widehat{\mathcal{Q}}v - v$. Then its component on $\widehat{\mathcal{D}}_a$, $w_{\widehat{\mathcal{D}}_a}$, can be explicitly written as

$$w_{\widehat{\mathcal{D}}_a}(\xi) = \sum_{\widehat{\mathcal{D}}_b \in \text{neigh}_{\text{pb}}(\xi)} \delta_{\widehat{\mathcal{D}}_b}^\dagger(\xi) (w_{\widehat{\mathcal{D}}_a}(\xi) - w_{\widehat{\mathcal{D}}_b}(\xi)).$$

Given a FE function $u \in \mathbb{V}_{\widehat{\mathcal{D}}}$, we denote by $\theta_\lambda^{\widehat{\mathcal{D}}}(u) \in \mathbb{V}_{\widehat{\mathcal{D}}}$ the FE function that is discrete α -harmonic in $\widehat{\mathcal{D}}$ and agrees with u at the FE nodes in the object λ and vanishes at all the other nodes on $\partial\widehat{\mathcal{D}}$. Since $\Lambda(\Theta_{\text{pb}})$ is a partition of $\Gamma(\Theta_{\text{pb}})$, we can split w into object and subdomain contributions as follows:

$$(14) \quad w = \sum_{\lambda \in \Lambda(\Theta_{\text{pb}})} \sum_{\widehat{\mathcal{D}} \in \text{neigh}_{\Theta_{\text{pb}}}(\lambda)} \theta_\lambda^{\widehat{\mathcal{D}}}(w).$$

By construction of the set of objects $\widehat{\Lambda}_O = \Lambda_O \cup \widehat{\Lambda}^*$ and the definition of $\widetilde{\mathbb{V}}^{\text{pb}}$, w vanishes at all coarse objects in $\widehat{\Lambda}^*$, i.e., at all FE nodes in $\Gamma(\Theta_{\text{pb}}) \setminus \Gamma(\Theta)$. Consequently, (14) can be simplified as follows:

$$(15) \quad w = \sum_{\lambda \in \Lambda_{\text{pb}}(\Theta)} \sum_{\widehat{\mathcal{D}} \in \Theta_{\text{pb}}^\partial} \theta_\lambda^{\widehat{\mathcal{D}}}(w).$$

When Λ_O satisfies [Assumption 5](#), the set of objects in $\widehat{\Lambda}_O$ also fulfills [[53](#), Assumption 6.27]. Consequently, using the fact that $\delta_{\widehat{\mathcal{D}}}^\dagger$ is constant in each (PB) coarse object associated with $\widehat{\mathcal{D}}$ and (9), we can perform an analysis similar to that in the proof [[53](#), Lemma 6.36] (see also [[33](#), Lemma 10]) to obtain

$$\begin{aligned} & \langle \mathcal{A}_{\widehat{\mathcal{D}}}^{\Theta_{\text{pb}}} \theta_\lambda^{\widehat{\mathcal{D}}}(w), \theta_\lambda^{\widehat{\mathcal{D}}}(w) \rangle \\ & \leq C \max\{1, \text{TOL}\} \left(1 + \log \left(\frac{H(\widehat{\mathcal{D}})}{h(\widehat{\mathcal{D}})} \right) \right)^2 \sum_{\widehat{\mathcal{D}} \in \text{neigh}_{\Theta_{\text{pb}}}(\lambda)} \langle \mathcal{A}_{\widehat{\mathcal{D}}}^{\Theta_{\text{pb}}} v, v \rangle, \end{aligned}$$

for any $\widehat{\mathcal{D}} \in \Theta_{\text{pb}}^\partial$ and $\lambda \in \Lambda(\Theta_{\text{pb}})$. Here the constant C is independent of $H(\widehat{\mathcal{D}})$, $h(\widehat{\mathcal{D}})$ and the physical coefficient α . We note that the standard analysis in, e.g., [[53](#), Lemma 6.36], assumes that for every face there is at least one primal edge (see [Remark 6](#)). [Assumption 5](#) is weaker, since for every face \mathcal{F} we simply assume that there is at least one edge \mathcal{E} on the boundary of this face for which there is an acceptable path. All the technical analysis for the standard case can easily be generalized for the one suggested herein. First, we use the bounds for the face term in [[53](#), p. 182], the only difference being that the mean value of the function on \mathcal{E} is not continuous among subdomains. Next, we have to bound the jump on this edge. Since the edge has an acceptable path, it can be readily bounded using the analysis in [[53](#), p. 183] for edge terms.

Adding up the estimate for all subdomains $\widehat{\mathcal{D}} \in \Theta_{\text{pb}}$, we find that

$$(16) \quad \langle \mathcal{A}^{\Theta_{\text{pb}}} w, w \rangle \leq C \max\{1, \text{TOL}\} \max_{\widehat{\mathcal{D}} \in \Theta_{\text{pb}}^\partial} \left(1 + \log \left(\frac{H(\widehat{\mathcal{D}})}{h(\widehat{\mathcal{D}})} \right) \right)^2 \langle \mathcal{A}^{\Theta_{\text{pb}}} v, v \rangle.$$

This finishes the proof. \square

Combining results in [Lemma 11](#) and [Lemma 12](#), we have the final bound for the PB-BDDC preconditioner, which is both weakly scalable and independent of the coefficient α .

THEOREM 13. *If the set of PB-BDDC coarse objects $\Lambda_{\mathcal{O}}$ satisfies [Assumption 5](#), then the condition number of the PB-BDDC preconditioned operator $\kappa(\mathcal{BA})$ is bounded by*

$$\kappa(\mathcal{BA}) \leq C \max\{1, \text{TOL}\} \max_{\hat{\mathcal{D}} \in \Theta_{\text{pb}}^{\theta}} \left(1 + \log \left(\frac{H(\hat{\mathcal{D}})}{h(\hat{\mathcal{D}})} \right) \right)^2,$$

where the constant C is independent of the number of subdomains, $H(\hat{\mathcal{D}})$, $h(\hat{\mathcal{D}})$ and the physical coefficient α .

Remark 14. As seen in [Lemma 12](#) and [Theorem 13](#), the condition number associated with the PB-BDDC method depends only on the characteristic size and mesh size of PB subdomains touching the original interface $\Gamma(\Theta)$. Further, the convergence of the PB-BDDC is independent of variations of the coefficient. The main target of this work is achieved.

Remark 15. For a fixed number of subdomains, partitions Θ with larger aspect ratios lead to larger values of $\max_{\hat{\mathcal{D}} \in \Theta} \frac{H(\hat{\mathcal{D}})}{h(\hat{\mathcal{D}})}$ (see also [\[22\]](#)). The value of $\max_{\hat{\mathcal{D}} \in \Theta_{\text{pb}}^{\theta}} \frac{H(\hat{\mathcal{D}})}{h(\hat{\mathcal{D}})}$ for the PB partition is equal or less than that for the original partition. However, in the PB partition, irregular subdomains that do not fulfill [\[53, Assumption 4.3 \(1.\)\]](#) may arise, affecting the constant in the estimate. Extension of these results to more complex geometries can be found in [\[31\]](#). In this sense, we know that (1) the subdomains interface (in which we allow for discontinuities) remains the same in PB-BDDC (e.g., flat interfaces in the original partition remain flat in the PB partition), and (2) the PB-BDDC space is a subspace of the BDDC space, and can never perform worse whatever geometry we consider for the PB partition.

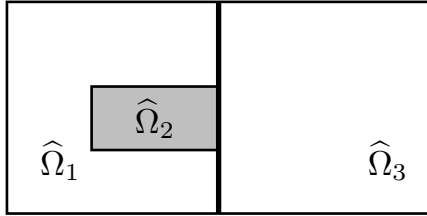
3.5. Reducing the size of the coarse space in PB-BDDC. If all the constraints associated with the PB partition are used, we might end up using too many constraints and the size of the coarse space can be quite large. This could affect the overall performance of the PB-BDDC method since the coarse problem is usually the bottleneck in large scale calculation (cf. [\[2, 3, 4, 8\]](#)). However, according to [Theorem 13](#), *not all the constraints associated with the PB partition are required to guarantee robustness*. We only need to choose the set of PB-BDDC coarse objects $\Lambda_{\mathcal{O}}$ so that [Assumption 5](#) holds.

Roughly speaking, for any two PB subdomains that share at least one point on the original subdomain interface, we need at least one acceptable path between them. From [Definition 4](#), it implies that for any pair of PB subdomains, if one or both of them have a small coefficient, then the other neighboring PB subdomains with larger or equal coefficients can be included in the acceptable path without increasing the tolerance TOL. In other words, one only needs to worry about pairs of connecting PB subdomains with large coefficients. There can be as many inclusions and channels as wanted inside a subdomain and not a single extra PB object is required. We note that these inclusions and channels can touch but not cross the original interface.

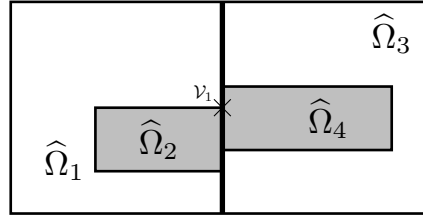
In [Figure 5](#), we illustrate some common scenarios of the coefficient and indicate the number of extra constraints, apart from the ones needed for well-posedness, required to guarantee the existence of acceptable paths. We only use channels for the clarity of the illustration even though the procedure can be applied to general scenarios.

For docking channels that do not cross the subdomain interface as in Figure 5a, no extra constraint is required provided that the original subdomains are *connected*, i.e., there is at least an acceptable path that connects them. For a long channel crossing the subdomain interface (see Figure 5b), one extra constraint is required. This constraint can be a vertex, an edge, or a face constraint. For channels that are connected inside subdomains, as illustrated in Figure 5c, they all belong to the same PB subdomain and only one extra vertex, edge or face constraint is sufficient (due to the weaker restriction on face-connected subdomains in Assumption 5). When channels are disjoint even in one of the subdomains only, we need at least one extra constraint for each non-connected channel, because every channel of this type leads to a different PB subdomain. This is illustrated in Figure 5d.

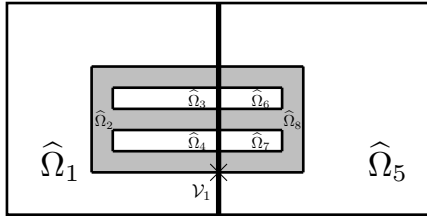
In summary, we need one extra PB constraint for every non-connected channel of high coefficient crossing the original interface. This is similar to spectral-based approaches of adaptive coarse spaces, e.g., [51, 29, 25], where the number of eigenfunctions needed is the same as the number of high coefficient channels crossing the interface. The conclusion will be numerically verified in subsection 4.1 and subsection 4.3.



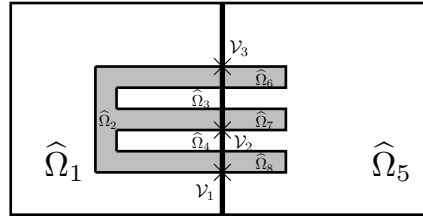
(a) Docking channel: no extra constraint is required.



(b) Long channel crossing boundary interface: one extra constraint is sufficient, e.g., $\hat{\Omega}_2 \xrightarrow{\nu_1} \hat{\Omega}_4$.



(c) Channels that are connected: one extra constraint is sufficient e.g., $\hat{\Omega}_2 \xrightarrow{\nu_1} \hat{\Omega}_8$.



(d) Disjoint channels: one extra constraint is required for each channel, e.g., $\hat{\Omega}_2 \xrightarrow{\nu_1} \hat{\Omega}_8$, $\hat{\Omega}_2 \xrightarrow{\nu_2} \hat{\Omega}_7$, $\hat{\Omega}_2 \xrightarrow{\nu_3} \hat{\Omega}_6$.

Fig. 5: Some scenarios of coefficient distribution and the minimal number of extra constraints required.

In practice, it can be complicated to select *the minimal* set of constraints. However, it is not difficult to choose a reasonably small set of constraints that is sufficient for PB-BDDC to be α -robust. For example, we can consider only PB subdomains touching the original interface and having coefficient relatively larger than the smallest coefficient of the neighboring PB subdomains in the same original subdomain.

For two PB subdomains that are of this type and belong to two different original subdomains, we will select a vertex, an edge, or a face constraint to “connect” them if they share at least a point. For simplicity, vertex constraints only can be used in 2D. This approach is utilized in our numerical experiments in [section 4](#).

3.6. Relaxed PB-BDDC preconditioner. The definition of the coarse objects for the PB-BDDC preconditioner is based on the requirement that the coefficient has to be constant in each PB subdomain. This could result in a large coarse space even after the procedure discussed in [subsection 3.5](#) is used. That is the case for heterogeneous problems where the physical coefficient varies across a wide spectrum of values in a small spatial scale. Variation of coefficient along the interface does affect the condition number. However, relatively small variation should not have significant effect on the condition number.

In order to deal with a more general class of problems, we propose the relaxed PB-BDDC preconditioner (rPB-BDDC) where we only require that the maximal contrast in each PB subdomain is less than some predefined tolerance r . We consider a relaxed PB partition Θ_{pb} such that

$$(17) \quad \max_{\tau, \tau' \subset \hat{\mathcal{D}}} \frac{\alpha_\tau}{\alpha_{\tau'}} \leq r, \quad \text{for any } \hat{\mathcal{D}} \in \Theta_{\text{pb}}.$$

Here the threshold r is equal or greater than 1. This way, we can control the size of the coarse problem and the condition number bounds with the choice of r .

As the coefficient is no longer constant in each PB subdomain, we need to use a weighted-constraint in the definition of coarse DOFs. More specifically, instead of using [\(3\)](#), we use

$$(18) \quad c_\lambda^{\mathcal{D}}(u) \doteq \frac{\int_\lambda \bar{\alpha} u \, ds}{\int_\lambda \bar{\alpha} 1 \, ds}, \quad \text{where } \bar{\alpha}(\xi) \doteq \max_{\tau \in \mathcal{T}, \tau \ni \xi} \alpha_\tau.$$

For rPB-BDDC, the weighting defined by [\(5\)](#) can still be used but it requires some sort of “average” coefficient for each PB subdomain. Therefore, in stead of using [\(5\)](#) as is, we propose to use the following alternative

$$(19) \quad Wu(\xi) \doteq \sum_{\mathcal{D} \in \text{neigh}_\Theta(\xi)} \delta_{\mathcal{D}}^{\dagger, c}(\xi) u_{\mathcal{D}}(\xi), \quad \text{with } \delta_{\mathcal{D}}^{\dagger, c}(\xi) \doteq \frac{\sum_{\tau \in \mathcal{T}_{\mathcal{D}}, \tau \ni \xi} \alpha_\tau |\tau|}{\sum_{\tau \in \mathcal{T}, \tau \ni \xi} \alpha_\tau |\tau|},$$

where $|\tau|$ denotes the area/volume of element τ . It should be noted that when $|\tau|$ is dropped, the weighting in [\(19\)](#) becomes the weighting defined in [\(5\)](#).

Remark 16. The larger r becomes the smaller the size of the coarse problem of the rPB-BDDC preconditioner is and the larger its condition number becomes. When $r = 1$ the rPB-BDDC preconditioner becomes the PB-BDDC preconditioner. By tuning the threshold r , one can obtain a right balance between the time spent on setting up the preconditioner (especially in forming the coarse space) and the time spent on applying the preconditioner in a Krylov solver. The optimal threshold is of course problem dependent. However, finding a good threshold is not tricky. This is illustrated in [section 4](#).

Remark 17. The rPB-BDDC preconditioner makes use of a threshold. This is similar to the adaptive coarse space approach where only eigenfunctions associated with eigenvalues below a predefined threshold are included in the coarse space. However, the rPB-BDDC preconditioner does not involve any eigenvalue or auxiliary problems and is far simpler and cheaper.

Remark 18. In terms of implementation, the PB partition for the rPB-BDDC method can be defined using element aggregation. In order to define such aggregation, a breadth-first search of the dual graph associated to the mesh can be performed. For a given seed element, neighbor elements are only aggregated if it does not involve the violation of condition (17). Next, we proceed analogously for the neighbors of the new aggregated elements. If we have not aggregated all elements yet, we start again with a new seed.

4. Numerical experiments. In this section, we test the robustness and efficiency of the PB-BDDC and rPB-BDDC preconditioners for the system matrix associated with (2) for different types of variation in the coefficient α , which are similar but generally harder than the ones in [45, 29, 38].

We consider the problem (1) with homogeneous Dirichlet boundary condition and the forcing term $f = 1$. In most cases, the physical domain is $\Omega = (0, 1)^2$. Unless otherwise stated, we use uniform triangular meshes of size $h = 1/72$ and a regular 3×3 subdomain partition. We report the dimension of the coarse space, denoted by \dim , and the number of iterations required for the preconditioned conjugate gradients method to reduce the residual norm by a factor of 10^6 . We also provide the computed (not estimated) condition number κ of the preconditioned operator in most examples.

We will compare the performance of BDDC(ce), the standard BDDC method with all possible subdomain corner and edge coarse objects, with that of PB-BDDC(ce), PB-BDDC(e) and PB-BDDC([c]), three variants of the PB-BDDC method where all possible physics-based corner and edge or edge coarse objects, or selected physics-based corner coarse objects are used. No corner detection mechanism (see, e.g., [49]) has been needed. Alternatively, one might want to consider the perturbed formulation introduced in [7, 8].

In addition, we will also compare the performance of BDDC(ce) with that of rPB-BDDC(ce) and rPB-BDDC(e), two variants of the relaxed version of PB-BDDC.

4.1. Two channels. In this test case, we consider two channels of high α cutting through vertical subdomain edges (see Figure 6). The coefficient in the channels α_{\max} takes the values $\{10^2, 10^4, 10^6, 10^8\}$, while the coefficient in the rest of the domain is equal to 1.

From Table 1, we can see that the condition number and the number of iterations for the standard BDDC preconditioner, BDDC(ce), definitely increase with α_{\max} , whereas they remain practically constant for the three variants of the PB-BDDC preconditioner: PB-BDDC(ce), PB-BDDC(e) and PB-BDDC([c]). In other words, the convergence of the PB-BDDC method is independent of the contrast and the PB-BDDC method is perfectly robust for this test case.

In addition, we want to emphasize that PB-BDDC([c]) can deliver perfect robustness with a coarse space of the same dimension as the coarse space of the standard method BDDC(ce). The coarse objects of PB-BDDC(ce) and PB-BDDC([c]) are illustrated in Figure 6 and Figure 7 respectively.

4.2. Channels and inclusions. In this test case, we consider both channels and inclusions of high coefficient. First, the three channels include all the elements whose centroids are less than $2 \cdot 10^{-2}$ from one of the following three lines:

$$\begin{aligned} L1 : x_1 - x_2 - 0.2 &= 0, \\ L2 : x_1 + x_2 - 0.7 &= 0, \\ L3 : x_1 - 0.7x_2 - 0.7 &= 0. \end{aligned}$$

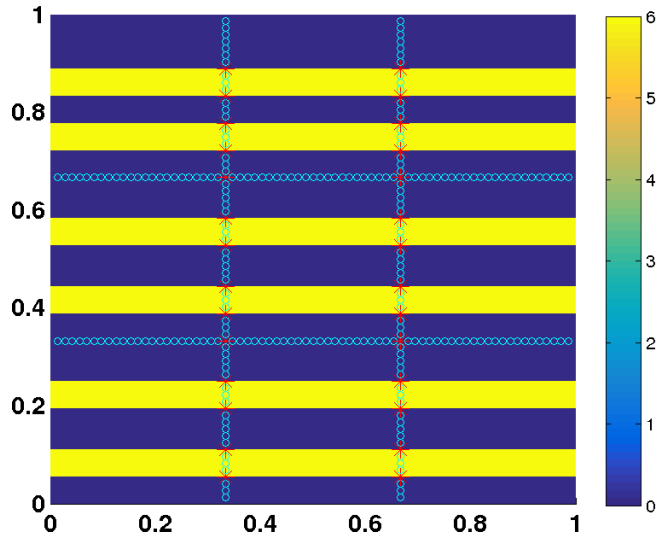


Fig. 6: Two channels test case: the coefficient distribution when $\alpha_{\max} = 10^6$. The coarse objects used by PB-BDDC(ce) are shown on the interface with corners labeled by stars and DOFs in edges labeled by circles.

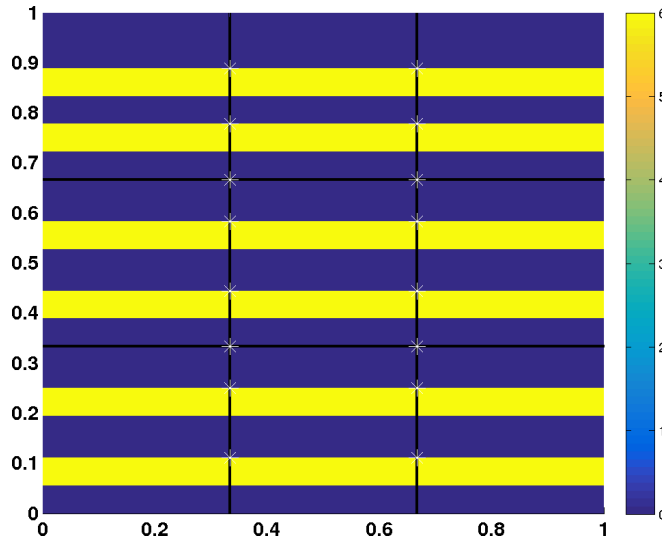


Fig. 7: Two channels test case: only a small number of critical corners, and subdomain corners are required by PB-BDDC($[c]$) to have perfect robustness. These corners are shown on the interface as stars.

The coefficient α_{\max} in these channels takes the values $\{10^2, 10^4, 10^6, 10^8\}$. Secondly, the inclusions are defined as the regions of elements whose all vertices x satisfy

$$\text{mod}(\text{floor}(10x_i), 2) = 1, \text{ for } i = 1, 2.$$

Table 1: Comparison of the iteration count and condition number in the two channels test case.

	BDDC(ce)	PB-BDDC(ce)	PB-BDDC(e)	PB-BDDC([c])
dim	16	64	36	16
α_{\max}	# it. (κ)	# it. (κ)	# it. (κ)	# it. (κ)
10^2	9(8.68e2)	4 (1.38e0)	5 (3.01e0)	6 (3.77e0)
10^4	13(1.83e5)	4 (1.36e0)	5 (2.92e0)	5 (2.25e0)
10^6	15(1.85e7)	4 (1.36e0)	5 (2.92e0)	5 (2.24e0)
10^8	20(1.89e9)	4 (1.36e0)	5 (2.92e0)	5 (2.24e0)

For an element τ that belongs to one of the inclusions and is not in the channels, its coefficient is defined as

$$(20) \quad \alpha|_{\tau} = (\alpha_{\max}/10)^{1/5 * \text{floor}(0.5 * \text{floor}(10 x_1(c_{\tau})) + 1)}, \quad \text{where } c_{\tau} \text{ is the centroid of } \tau.$$

The coefficient in (20) is: a) constant in each inclusion; b) increasing from left to right; c) increasing as α_{\max} increases; and d) always belongs to $(1, \alpha_{\max})$. For the rest of the domain, we set $\alpha = 1$. The maximal contrast ratio in this experiment is 10^8 . The coefficient distribution when $\alpha = 10^6$ is shown in Figure 8.

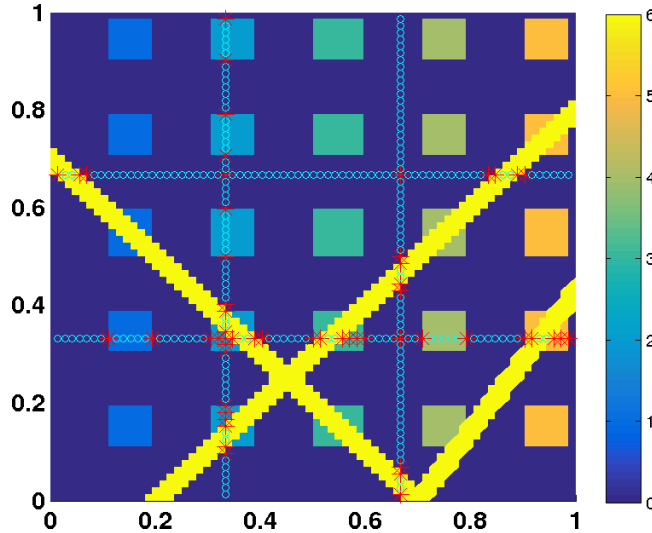


Fig. 8: Channels and inclusions test case: the coefficient distribution when $\alpha_{\max} = 10^6$. The coarse objects used by PB-BDDC(ce) are shown on the interface with corners labeled by stars and DOFs in edges labeled by circles.

We can see from Table 2 that as α_{\max} becomes larger the condition number and the number of iterations associated with the standard BDDC(ce) method increases significantly. In contrast, both variants of the PB-BDDC method, PB-BDDC(ce) and PB-BDDC(e), are perfectly robust with respect to the changes of the coefficient in the channels and in the inclusions. Especially, PB-BDDC(e) maintains its robustness

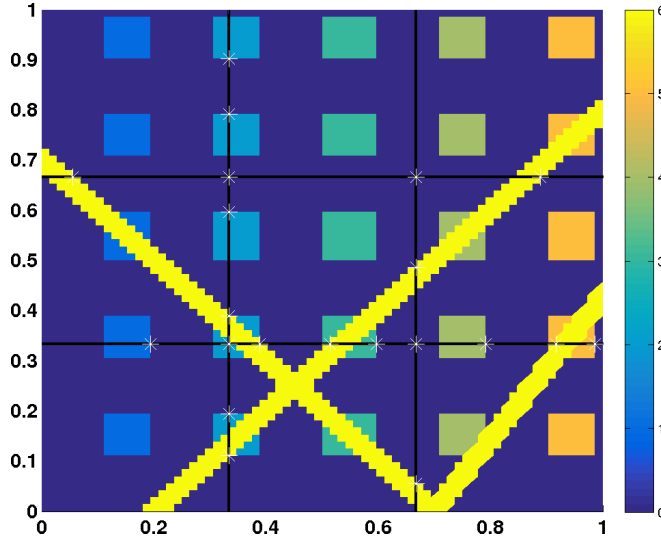


Fig. 9: Channels and inclusions test case: only a small number of critical corners, and subdomain corners are required by PB-BDDC([c]) to have α -robustness. These corners are shown on the interface as stars.

with a reasonably small coarse space (see also Figure 9).

Table 2: Comparison of the iteration count and condition number in the channels and inclusions test case.

	BDDC(ce)	PB-BDDC(ce)	PB-BDDC(e)	PB-BDDC([c])
dim	16	89	39	19
α_{\max}	# it. (κ)	# it. (κ)	# it. (κ)	# it. (κ)
10^2	10(2.57e5)	6 (1.91e0)	10 (4.84e1)	12 (2.25e2)
10^4	24(1.79e7)	6 (1.99e0)	10 (7.00e1)	13 (3.44e2)
10^6	36(1.54e9)	6 (2.04e0)	11 (7.03e1)	14 (3.49e2)
10^8	61(1.47e11)	6 (2.04e0)	11 (7.03e1)	14 (3.50e2)

4.3. Complex channels. In this test case, we demonstrate the importance of having acceptable paths. We consider a distribution with multiple channels of high coefficient α_{\max} taking values in $\{10^2, 10^4, 10^6, 10^8\}$ (see Figure 10 for the case when $\alpha_{\max} = 10^6$).

From Table 3, we can see that PB-BDDC(ce) is perfectly robust. On the other hand, the condition number and number of iterations of the PB-BDDC(e) preconditioner increase significantly as α_{\max} increases. The reason is that there are some pairs of channels sharing a corner but not an edge. In PB-BDDC(e), none of these corners are selected as a coarse object. Consequently, there is no acceptable path with TOL independent of the contrast between the associated paired of channels (PB subdomains) and Assumption 5 does not hold. By using a small number of these critical corners in order to satisfy Assumption 5 and the subdomain corners to guar-

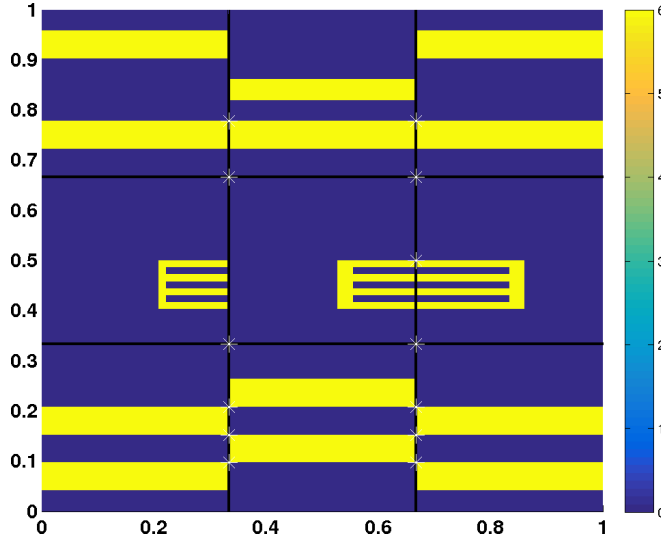


Fig. 10: Complex channels test case: the coefficient distribution when $\alpha_{\max} = 10^6$. Only few corners are required by PB-BDDC([c]) to guarantee perfect robustness (to have acceptable paths). These corners are shown on the interface as stars.

antee invertibility, the resulting preconditioner, PB-BDDC([c]), is perfectly robust with respect to changes in the contrast of the coefficient (see Table 3).

Table 3: Comparison of the iteration count and condition number in the complex channels test case.

	BDDC(ce)	PB-BDDC(ce)	PB-BDDC(e)	PB-BDDC([c])
dim	16	78	46	13
α_{\max}	# it. (κ)	# it. (κ)	# it. (κ)	# it. (κ)
10^2	16 (2.40e3)	6 (2.75e0)	14 (1.42e3)	10 (8.79e1)
10^4	26 (4.22e5)	6 (2.82e0)	19 (1.96e5)	10 (9.83e1)
10^6	38 (4.25e7)	6 (2.82e0)	25 (1.97e7)	10 (9.84e1)
10^8	55 (4.93e9)	6 (2.82e0)	29 (1.97e9)	10 (9.84e1)

4.4. Sinusoidal variation. In this experiment, we consider a coefficient that varies like a sinusoid. We use a finer uniform triangular mesh of size $h = 1/144$. For an element $\tau \in \mathcal{T}$, the coefficient α_τ is defined by

$$\log_{10}(\alpha_\tau) = \kappa \sin(w\pi(x_1(c_\tau) + x_2(c_\tau))) + \alpha_{\text{shift}},$$

where $\kappa = 3$, $w = 14$, and c_τ is the centroid of τ . We note that when κ and/or w become larger the problem is more difficult. The distribution when $\alpha_{\text{shift}} = 0$ is shown in Figure 11. It is as if there are many channels going through subdomain edges at the same time.

In this test case, the coefficient varies very rapidly. We test the standard BDDC method and the rPB-BDDC method introduced in subsection 3.6, by allowing the

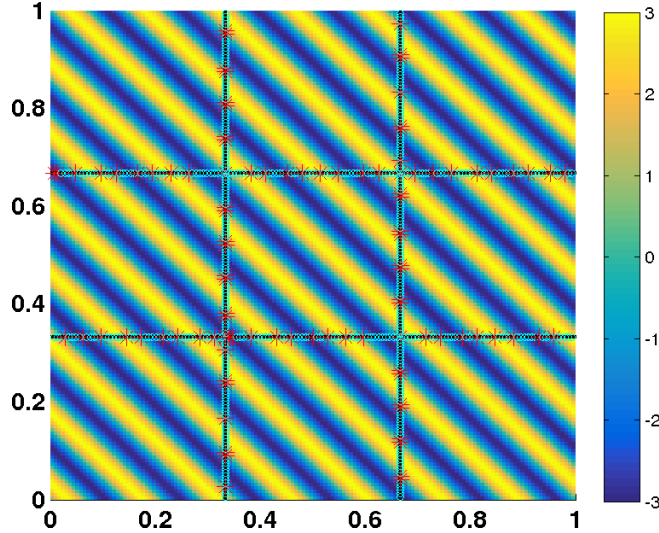


Fig. 11: Distribution of the coefficient mimicking sin function. The coarse objects of rPB-BDDC(ce) with $r = 10^3$ are shown on the interface with corners labeled by stars and DOFs in edges labeled by circles.

upper bound r for the maximal contrast in each PB subdomain to vary among $\{10^1, 10^2, 10^3\}$. Only iteration counts are reported as computing (not estimating) condition numbers becomes too expensive for the mesh being used.

This is a difficult problem and the standard BDDC(ce) method requires 43 iterations to converge (see Table 4). The relaxed physics-based methods, rPB-BDDC(ce) and rPB-BDDC(e), are able to significantly reduce the number of iterations. That comes with the cost of solving larger coarse problems. However, by using a suitable threshold r , we can obtain a decent preconditioner, e.g., rPB-BDDC(e) with $r = 10^3$, which requires only 8 iterations using a reasonably small coarse space of size 64.

In addition, the rPB-BDDC method is also perfectly robust with shifting in the value of the coefficient. The iteration count does not change when α_{shift} takes values in $\{0, 6\}$.

Table 4: Comparison of the iteration count in the continuous sin test case.

	r	BDDC(ce)			rPB-BDDC(ce)			rPB-BDDC(e)			
		dim	10	10^2	10^3	10	10^2	10^3	10	10^2	10^3
	dim	16	474	292	188	212	116	64			
$\alpha_{\text{shift}} = 0$	# it.	43	4	5	5	7	8	8			
$\alpha_{\text{shift}} = 6$	# it.	43	4	5	5	7	8	8			

4.5. Log-Normal. In this test case, we test the performance of the rPB-BDDC method for a log-normal distribution of the coefficient. This type of distribution is particularly important for geoscience and petroleum engineering applications. We consider $\alpha_{\text{cont}}(x, w) = 10^{Z(x, w)}$, where $Z(x, w)$ is a Gaussian random field with zero

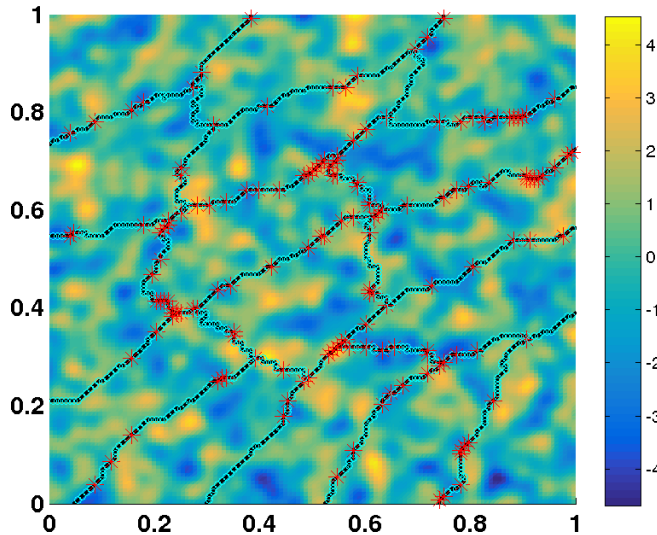


Fig. 12: Distribution of the coefficient in the log-normal test case. The coarse objects of rPB-BDDC(ce) with $r = 10^2$ are shown on the interface with corners labeled by stars and DOFs in edges labeled by circles.

mean and Gaussian covariance

$$C(x, y) = \sigma^2 \exp\left(-\frac{\|x - y\|^2}{\ell^2}\right), \quad \text{with } \sigma = 1.5, \ell^2 = 1e-3.$$

For this experiment, a uniform triangular mesh of size $h = 1/128$ is utilized. Using the spectral decomposition method described in [39], we are able to obtain a realization of $\alpha_{\text{cont}}(x, w)$ at mesh vertices. The piecewise coefficient α_τ on an element τ is then defined as the average of $\alpha_{\text{cont}}(x, w)$ at the three vertices. The distribution of α with a partition obtained from METIS [27] is shown in Figure 12. The contrast ratio in this test case is nearly 10^{10} . The upper bound r for the maximal contrast in each PB subdomain varies among $\{10^1, 5 \times 10^1, 10^2\}$. The coarse objects of rPB-BDDC(ce) when $r = 10^2$ are also illustrated.

In Table 5, we can see that, compared to the standard BDDC(ce) method, rPB-BDDC(ce) and rPB-BDDC(e) preconditioners require much fewer iterations to converge. They, however, have a larger coarse space. By adjusting the threshold for the maximal contrast in each PB subdomain, we can reduce the size of the coarse space while maintaining a reasonably fast convergence. This is clearly illustrated in Table 5.

Table 5: Comparison of the iteration count in the log-normal test case.

	BDDC(ce)	rPB-BDDC(ce)			rPB-BDDC(e)		
		10^1	5×10^1	10^2	10^1	5×10^1	10^2
r							
dim	49	438	265	228	187	112	98
# it.	25	9	11	13	13	15	17

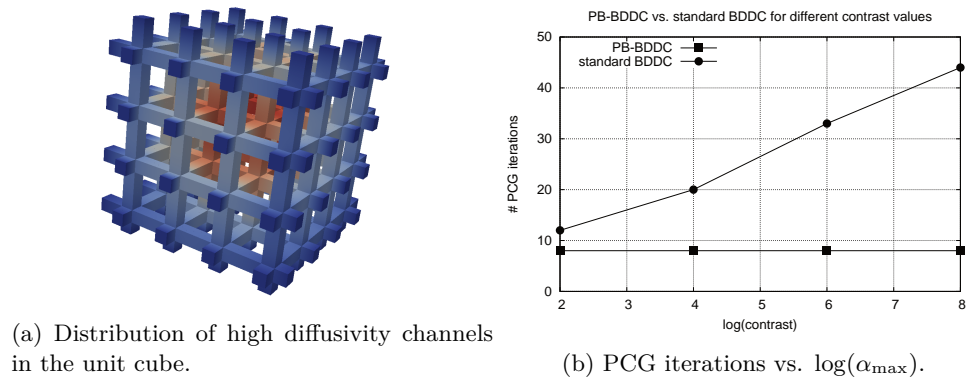


Fig. 13: 3D channels problem.

4.6. 3D channels. Next, we analyze the robustness and performance of the PB-BDDC preconditioner when applied to a 3D problem. We consider the unit cube $[0, 1]^3$. Its volume is filled with two different materials, a matrix with diffusion $\alpha = 1.0$ and a set of fibers (see Figure 13a) with $\alpha = 10^\ell$, for $\ell = 2, 4, 6, 8$. In Figure 13b, we compare the number of PCG iterations using the standard BDDC preconditioner and the PB-BDDC (both equipped with corner, edge, and face constraints) with a fixed mesh. Clearly, the PB-BDDC preconditioner is robust with respect to the jump of the diffusion between materials while the standard BDDC preconditioner is not.

We want to analyze the scalability of the PB-BDDC preconditioner for the problem at hand on a supercomputer. The experiments reported in the following have been obtained on a large-scale multicore-based distributed-memory machine, Marenostrum III, located at the Barcelona Supercomputing Center (BSC-CNS). The Marenostrum III is a FDR10 Infiniband interconnected cluster with 36 IBM System x iDataPlex racks devoted to computations. Each rack is composed of 84 IBM dx360 M4 compute nodes, each equipped with two Intel Xeon E5-2670 EightCore processors running at 2.6 GHz (16 computational cores in total) and 32 GBytes of DDR3 memory (2 GBytes per core), and runs a full-featured Linux OS. The codes were compiled using the GNU Fortran compiler (6.1.0) with recommended optimization flags and we used OpenMPI (2.0.1) tools and libraries for message-passing. The codes were linked against the BLAS/LAPACK and PARDISO available on the Intel MKL library (included in Intel Parallel Studio XE 2017). All floating-point calculations were performed in IEEE double precision.

The PB-BDDC preconditioner has been implemented in FEMPAR, developed by the members of the LSSC team at CIMNE, which is a parallel hybrid OpenMP/MPI, object-oriented software package for the massively parallel Finite Element (FE) simulation of multiphysics problems governed by PDEs. Among other features, it provides the basic tools for the efficient parallel distributed-memory implementation of substructuring DD solvers [3]. The parallel codes in FEMPAR heavily use standard computational kernels provided by (highly-efficient vendor implementations of) the BLAS and LAPACK. Besides, through proper interfaces to several third party libraries, the local Dirichlet and constrained Neumann problems at each intermediate level of the hierarchy, and the global coarsest-grid problem at the last level, can be solved via

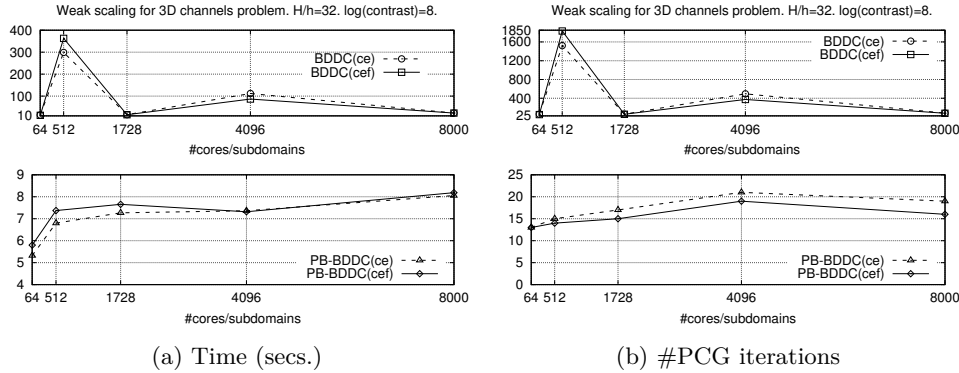


Fig. 14: Weak scalability in (a) wall clock time in seconds (of both preconditioner set-up and PCG stages) and (b) number of PCG iterations of the PB-BDDC(ce/cef) algorithm compared to those of the standard BDDC(ce/cef) solver when applied to the 3D channels problem ($\ell = 8$) on Marenstrum III.

sparse direct solvers. FEMPAR is released under the GNU GPL v3 license, and is more than 200K lines of Fortran03/08 code long. The FEMPAR library includes a highly scalable implementation of the BDDC-PCG parallel linear solver based on overlapping fine-grid/coarse-grid duties in time. The global set of MPI tasks is split into those that have fine-grid duties and those that have coarse-grid duties. Next, the different computations and communications arising in the BDDC-PCG algorithm are scheduled and mapped in such a way that the maximum degree of overlapping is achieved. We refer to [4] for the presentation of such implementation. The use of inexact solvers within this library has been presented in [5] and a recursive implementation of multilevel BDDC preconditioners is described in [6].

We consider a weak scalability analysis of PB-BDDC and standard BDDC methods when applied to the 3D channels problem with $\ell = 8$, a set uniform meshes with $32m \times 32m \times 32m$ hexahedra, and regular partitions of this mesh into $m \times m \times m$ subdomains, for $m = 2, 4, 6, 8, 10$ (the local problem size per processor is independent of m). The largest test case has more than 260 million unknowns. The wall clock time (in seconds) associated to both the setup of the preconditioner and the solution of the linear system using PB-BDDC versus standard BDDC-preconditioned conjugate gradients are reported in Figure 14a. The number of iterations required to converge is reported in Figure 14b. We can observe excellent weak scalability and robustness of PB-BDDC (bottom row of Figure 14), especially, when compared to those of the standard BDDC algorithm (top row of Figure 14). The use of the overlapped implementation within FEMPAR is essential to provide such results, because it masks the CPU time at the coarse solver level with embarrassingly parallel fine level duties. The number of iterations for the standard case has a very rough behavior, related to the fact that the method is not α -robust and the different situations encountered at different core counts. On the contrary, the PB-BDDC method is robust and has a very smooth behavior.

5. Conclusions. In this work, we have proposed a novel type of BDDC preconditioners that are robust for heterogeneous problems with high contrast. The

underlying idea is to modify the continuity constraints enforced among subdomains making use of the knowledge about the physical coefficients. In order to do that, we rely on a physically motivated partition of standard coarse objects (corners, edges, and faces) into coarse sub-objects. The motivation for that is the well-known robustness of DD methods when there are only jumps of physical coefficients across the interface between subdomains. All these ideas can also be used in the frame of FETI methods.

In cases where the physical coefficient is constant in each PB subdomain, we are able to prove that the associated condition number can be bounded independent of the number of the subdomains and the contrast of the physical coefficient. In other words, the new preconditioner is scalable and robust for heterogeneous problems. In addition, not all but any set of PB objects satisfying a mild condition on acceptable paths is sufficient to guarantee robustness.

Apart from the new set of coarse objects and a new weighting operator, the (r)PB-BDDC preconditioners are very much the same as the standard BDDC preconditioner. As a result, the implementation of the new preconditioners involve a very simple modification of the standard BDDC implementation. In all of our experiments, the new preconditioners deliver fast, robust and contrast-independent convergence while maintaining the simplicity of BDDC methods at a reasonable computational cost. Compared to the other robust DD solvers for heterogeneous problems currently available, such as the ones in [26, 48, 26, 46, 47, 23, 24, 45, 50, 15, 52, 51, 29, 28, 30, 38, 25], our new methods do not involve any type of eigenvalue or auxiliary problems.

The use of (r)PB-BDDC methods for heterogeneous problems lead to generally larger coarse problems than in the homogeneous case. In order to cope this problem, we have considered two remedies. First, we provide a simple procedure to find small and close to minimal coarse spaces which guarantee robustness. Second, we have implemented the algorithms in the extremely scalable BDDC code in FEMPAR [3, 4, 5, 6], which exploits an inter-level task-overlapping implementation to efficiently deal with the coarse space computation in weak scalability analyses. We have shown excellent scalability results up to 8000 cores and more than 260 million unknowns on a supercomputer. Future work will include the multilevel extension of the present approach for the overlapped recursive multilevel implementation in FEMPAR, which will be particularly interesting in the rPB-BDDC case, and the application of the preconditioner to real 3D applications, e.g., in geosciences.

REFERENCES

- [1] A. ABDULLE, W. E. B. ENGQUIST, AND E. VANDEN-EIJNDEN, *The heterogeneous multiscale method*, Acta Numer., 21 (2012), pp. 1–87.
- [2] S. BADIA, A. F. MARTÍN, AND J. PRINCIPE, *Enhanced balancing Neumann-Neumann preconditioning in computational fluid and solid mechanics*, International Journal for Numerical Methods in Engineering, 96 (2013), pp. 203–230.
- [3] ———, *Implementation and scalability analysis of balancing domain decomposition methods*, Archives of Computational Methods in Engineering, 20 (2013), pp. 239–262.
- [4] ———, *A highly scalable parallel implementation of balancing domain decomposition by constraints*, SIAM Journal on Scientific Computing, 36 (2014), pp. C190–C218.
- [5] ———, *On the scalability of inexact balancing domain decomposition by constraints with overlapped coarse/fine corrections*, Parallel Computing, 50 (2015), pp. 1 – 24.
- [6] ———, *Multilevel balancing domain decomposition at extreme scales*, SIAM J. Sci. Comput., 38 (2016), pp. C22–C52.
- [7] S. BADIA AND H. NGUYEN, *Relaxing the roles of corners in BDDC by perturbed formulation*, 2016.
- [8] ———, *Balancing domain decomposition by constraints and perturbation*, SIAM Journal on Numerical Analysis, (accepted).

- [9] S. C. BRENNER AND L. R. SCOTT, *The mathematical theory of finite element methods*, vol. 15 of Texts in Applied Mathematics, Springer, New York, third ed., 2008.
- [10] S. C. BRENNER AND L.-Y. SUNG, *BDDC and FETI-DP without matrices or vectors*, *Computer Methods in Applied Mechanics and Engineering*, 196 (2007), pp. 1429–1435.
- [11] T. F. CHAN AND T. P. MATHEW, *Domain decomposition algorithms*, *Acta numerica*, 3 (1994), pp. 61–143.
- [12] C. R. DOHRMANN, *A Preconditioner for Substructuring Based on Constrained Energy Minimization*, *SIAM Journal on Scientific Computing*, 25 (2003), pp. 246–258.
- [13] ———, *An approximate BDDC preconditioner*, *Numerical Linear Algebra with Applications*, 14 (2007), pp. 149–168.
- [14] C. R. DOHRMANN AND O. B. WIDLUND, *A BDDC algorithm with deluxe scaling for three-dimensional $H(\text{curl})$ problems*, *Communications on Pure and Applied Mathematics*, 69 (2016), pp. 745–770.
- [15] V. DOLEAN, F. NATAF, R. SCHEICHL, AND N. SPILLANE, *Analysis of a two-level Schwarz method with coarse spaces based on local Dirichlet-to-Neumann maps*, *Comput. Methods Appl. Math.*, 12 (2012), pp. 391–414.
- [16] M. DRYJA, J. GALVIS, AND M. SARKIS, *BDDC methods for discontinuous Galerkin discretization of elliptic problems*, *J. Complexity*, 23 (2007), pp. 715–739.
- [17] M. DRYJA, M. V. SARKIS, AND O. B. WIDLUND, *Multilevel Schwarz methods for elliptic problems with discontinuous coefficients in three dimensions*, *Numer. Math.*, 72 (1996), pp. 313–348.
- [18] M. DRYJA, B. F. SMITH, AND O. B. WIDLUND, *Schwarz analysis of iterative substructuring algorithms for elliptic problems in three dimensions*, *SIAM J. Numer. Anal.*, 31 (1994), pp. 1662–1694.
- [19] W. E, B. ENGQUIST, X. LI, W. REN, AND E. VANDEN-ELJNDEN, *Heterogeneous multiscale methods: a review*, *Commun. Comput. Phys.*, 2 (2007), pp. 367–450.
- [20] C. FARHAT, M. LESOINNE, P. LE TALLEC, K. PIERSON, AND D. RIXEN, *FETI-DP: a dual-primal unified FETI method-part I: A faster alternative to the two-level FETI method*, *International Journal for Numerical Methods in Engineering*, 50 (2001), pp. 1523–1544.
- [21] C. FARHAT, M. LESOINNE, AND K. PIERSON, *A scalable dual-primal domain decomposition method*, *Numerical Linear Algebra with Applications*, 7 (2000), pp. 687–714.
- [22] C. FARHAT, N. MAMAN, AND G. W. BROWN, *Mesh partitioning for implicit computations via iterative domain decomposition: Impact and optimization of the subdomain aspect ratio*, *International Journal for Numerical Methods in Engineering*, 38 (1995), pp. 989–1000.
- [23] J. GALVIS AND Y. EFENDIEV, *Domain decomposition preconditioners for multiscale flows in high-contrast media*, *Multiscale Model. Simul.*, 8 (2010), pp. 1461–1483.
- [24] ———, *Domain decomposition preconditioners for multiscale flows in high contrast media: reduced dimension coarse spaces*, *Multiscale Model. Simul.*, 8 (2010), pp. 1621–1644.
- [25] M. J. GANDER, A. LONELAND, AND T. RAHMAN, *Analysis of a new harmonically enriched multiscale coarse space for domain decomposition methods*, Dec. 2015.
- [26] I. G. GRAHAM, P. O. LECHNER, AND R. SCHEICHL, *Domain decomposition for multiscale PDEs*, *Numer. Math.*, 106 (2007), pp. 589–626.
- [27] G. KARYPIS AND V. KUMAR, *A fast and high quality multilevel scheme for partitioning irregular graphs*, *SIAM J. Sci. Comput.*, 20 (1998), pp. 359–392 (electronic).
- [28] H. H. KIM AND E. T. CHUNG, *A BDDC algorithm with enriched coarse spaces for two-dimensional elliptic problems with oscillatory and high contrast coefficients*, *Multiscale Model. Simul.*, 13 (2015), pp. 571–593.
- [29] A. KLAWONN, P. RADTKE, AND O. RHEINBACH, *FETI-DP methods with an adaptive coarse space*, *SIAM Journal on Numerical Analysis*, 53 (2015), pp. 297–320.
- [30] A. KLAWONN, P. RADTKE, AND O. RHEINBACH, *Adaptive coarse spaces for BDDC with a transformation of basis*, in *Domain Decomposition Methods in Science and Engineering XXII*, T. Dickopf, J. M. Gander, L. Halpern, R. Krause, and F. L. Pavarino, eds., Springer International Publishing, 2016, pp. 301–309.
- [31] A. KLAWONN, O. RHEINBACH, AND O. B. WIDLUND, *An analysis of a FETI-DP algorithm on irregular subdomains in the plane*, *SIAM Journal on Numerical Analysis*, 46 (2008), pp. 2484–2504.
- [32] A. KLAWONN AND O. B. WIDLUND, *Dual-primal FETI methods for linear elasticity*, *Communications on Pure and Applied Mathematics*, 59 (2006), pp. 1523–1572.
- [33] A. KLAWONN, O. B. WIDLUND, AND M. DRYJA, *Dual-primal FETI methods for three-dimensional elliptic problems with heterogeneous coefficients*, *SIAM J. Numer. Anal.*, 40 (2002), pp. 159–179 (electronic).
- [34] ———, *Dual-primal FETI methods with face constraints*, in *Recent developments in domain*

- decomposition methods (Zürich, 2001), vol. 23 of Lect. Notes Comput. Sci. Eng., Springer, Berlin, 2002, pp. 27–40.
- [35] J. H. LEE, *A balancing domain decomposition by constraints deluxe method for Reissner-Mindlin plates with Falk-Tu elements*, SIAM J. Numer. Anal., 53 (2015), pp. 63–81.
- [36] J. LI AND O. B. WIDLUND, *FETI-DP, BDDC, and block Cholesky methods*, International Journal for Numerical Methods in Engineering, 66 (2006), pp. 250–271.
- [37] ———, *On the use of inexact subdomain solvers for BDDC algorithms*, Computer Methods in Applied Mechanics and Engineering, 196 (2007), pp. 1415–1428.
- [38] S. LOISEL, H. NGUYEN, AND R. SCHEICHL, *Optimized schwarz and 2-lagrange multiplier methods for multiscale elliptic pdes*, SIAM Journal on Scientific Computing, 37 (2015), pp. A2896–A2923.
- [39] G. J. LORD, C. E. POWELL, AND T. SHARDLOW, *An Introduction to Computational Stochastic PDEs*, Cambridge Texts in Applied Mathematics, 2014.
- [40] J. MANDEL, *Balancing domain decomposition*, Communications in Numerical Methods in Engineering, 9 (1993), pp. 233–241.
- [41] J. MANDEL AND C. R. DOHRMANN, *Convergence of a balancing domain decomposition by constraints and energy minimization*, Numerical Linear Algebra with Applications, 10 (2003), pp. 639–659.
- [42] J. MANDEL, C. R. DOHRMANN, AND R. TEZAU, *An algebraic theory for primal and dual substructuring methods by constraints*, Applied Numerical Mathematics, 54 (2005), pp. 167–193.
- [43] J. MANDEL, B. SOUSEDÍK, AND C. DOHRMANN, *Multispace and multilevel BDDC*, Computing, 83 (2008), pp. 55–85.
- [44] T. P. A. MATHÉW, *Domain decomposition methods for the numerical solution of partial differential equations*, vol. 61, Springer, 2008.
- [45] F. NATAF, H. XIANG, V. DOLEAN, AND N. SPILLANE, *A coarse space construction based on local Dirichlet-to-Neumann maps*, SIAM J. Sci. Comput., 33 (2011), pp. 1623–1642.
- [46] C. PECHSTEIN AND R. SCHEICHL, *Analysis of FETI methods for multiscale PDEs*, Numer. Math., 111 (2008), pp. 293–333.
- [47] ———, *Analysis of FETI methods for multiscale PDEs. Part II: interface variation*, Numer. Math., 118 (2011), pp. 485–529.
- [48] R. SCHEICHL AND E. VAINIKKO, *Additive Schwarz with aggregation-based coarsening for elliptic problems with highly variable coefficients*, Computing, 80 (2007), pp. 319–343.
- [49] J. ŠÍSTEK, M. ČERTÍKOVÁ, P. BURDA, AND J. NOVOTNÝ, *Face-based selection of corners in 3D substructuring*, Math. Comput. Simulation, 82 (2012), pp. 1799–1811.
- [50] N. SPILLANE, V. DOLEAN, P. HAURET, F. NATAF, C. PECHSTEIN, AND R. SCHEICHL, *A robust two-level domain decomposition preconditioner for systems of PDEs*, C. R. Math. Acad. Sci. Paris, 349 (2011), pp. 1255–1259.
- [51] N. SPILLANE, V. DOLEAN, P. HAURET, F. NATAF, C. PECHSTEIN, AND R. SCHEICHL, *Abstract robust coarse spaces for systems of PDEs via generalized eigenproblems in the overlaps*, Numer. Math., 126 (2014), pp. 741–770.
- [52] N. SPILLANE, V. DOLEAN, P. HAURET, F. NATAF, AND D. J. RIXEN, *Solving generalized eigenvalue problems on the interfaces to build a robust two-level FETI method*, C. R. Math. Acad. Sci. Paris, 351 (2013), pp. 197–201.
- [53] A. TOSELLI AND O. WIDLUND, *Domain decomposition methods—algorithms and theory*, vol. 34 of Springer Series in Computational Mathematics, Springer-Verlag, Berlin, 2005.
- [54] X. TU, *Three-Level BDDC in Three Dimensions*, SIAM Journal on Scientific Computing, 29 (2007), pp. 1759–1780.
- [55] O. B. WIDLUND AND C. R. DOHRMANN, *BDDC deluxe domain decomposition*, in Domain Decomposition Methods in Science and Engineering XXII, T. Dickopf, J. M. Gander, L. Halpern, R. Krause, and F. L. Pavarino, eds., Cham, 2016, Springer International Publishing, pp. 93–103.
- [56] Y. ZHU, *Domain decomposition preconditioners for elliptic equations with jump coefficients*, Numer. Linear Algebra Appl., 15 (2008), pp. 271–289.