Electronic Transactions on Numerical Analysis.
Volume 51, pp. 432–450, 2019.
Copyright © 2019, Kent State University.
ISSN 1068–9613.
DOI: 10.1553/etna_vol51s432

ETNA
Kent State University and
Johann Radon Institute (RICAM)

PRECONDITIONING THE COARSE PROBLEM OF BDDC METHODS—THREE-LEVEL, ALGEBRAIC MULTIGRID, AND VERTEX-BASED PRECONDITIONERS[*]

AXEL KLAWONN[†‡], MARTIN LANSER[†‡], OLIVER RHEINBACH[§], AND JANINE WEBER[†]

Abstract. A comparison of three Balancing Domain Decomposition by Constraints (BDDC) methods with an approximate coarse space solver using the same software building blocks is attempted for the first time. The comparison is made for a BDDC method with an algebraic multigrid preconditioner for the coarse problem, a three-level BDDC method, and a BDDC method with a vertex-based coarse preconditioner. It is new that all methods are presented and discussed in a common framework. Condition number bounds are provided for all approaches. All methods are implemented in a common highly parallel scalable BDDC software package based on PETSc to allow for a simple and meaningful comparison. Numerical results showing the parallel scalability are presented for the equations of linear elasticity. For the first time, this includes parallel scalability tests for a vertex-based approximate BDDC method.

Key words. approximate BDDC, three-level BDDC, multilevel BDDC, vertex-based BDDC

AMS subject classifications. 68W10, 65N22, 65N55, 65F08, 65F10, 65Y05

1. Introduction.

During the last decade, approximate variants of the BDDC (Balancing Domain Decomposition by Constraints) and FETI-DP (Finite Element Tearing and Interconnecting-Dual-Primal) methods have become popular for the solution of various linear and nonlinear partial differential equations [1, 8, 9, 12, 14, 15, 17, 19, 21, 24, 25]. These methods differ from their exact relatives by using an approximate solution of components of the preconditioner, most notably the coarse problem. An approximate solution of the coarse problem can reduce the numerical robustness slightly but can increase the scalability of the method significantly. While multilevel BDDC (see [20, 22, 24, 25] and, recently, [1]) is constructed by applying exact BDDC recursively to the coarse problem, in other approximate BDDC variants, cycles of AMG (algebraic multigrid) are applied to the coarse problem; see, e.g., [8, 13, 19]. Recently, vertex-based coarse spaces of reduced size have been suggested to approximate the original coarse problem [9].

In [13], we have already considered, in a common framework, several linear and nonlinear BDDC variants using AMG-based approximations, following the BDDC formulation of [19] for linear problems. We have also compared their performance using our ultra scalable PETSc-based [4, 5, 6] BDDC implementation applying BoomerAMG [11] for all AMG solves. In the current paper, we continue these efforts and include the aforementioned vertex-based BDDC as well as three-level and multilevel BDDC in our framework as well as in our software package. In addition to a description of all methods and their condition number bounds, we also include a numerical and parallel comparison. To the best of our knowledge, a comparison between three-level BDDC and BDDC with AGM-based coarse approximations using implementations

[*] Received June 17, 2019. Accepted November 13, 2019. Published online on December 5, 2019. Recommended by O. Widlund. This work was supported in part, as project EXASTEEL, by Deutsche Forschungsgemeinschaft (DFG) through the Priority Programme 1648 "Software for Exascale Computing" (SPPEXA) under grant number 230723766.

[†] Department of Mathematics and Computer Science, University of Cologne, Weyertal 86-90, 50931 Köln, Germany, http://www.numerik.uni-koeln.de
({axel.klawonn, martin.lanser, janine.weber}@uni-koeln.de).

[‡] Center for Data and Simulation Science, University of Cologne, Germany,
http://www.cds.uni-koeln.de

[§] Fakultät für Mathematik und Informatik, Institut für Numerische Mathematik und Optimierung, Technische Universität Bergakademie Freiberg and Universitätsrechenzentrum (URZ), Technische Universität Bergakademie Freiberg, 09596 Freiberg, Germany (oliver.rheinbach@math.tu-freiberg.de).

based on the same building blocks has not been considered before. Also, for the first time, parallel scalability tests for a vertex-based BDDC method [9] are presented.

As a common baseline in all our comparisons, we include the approximate AMG-based preconditioner which performed best in [13]. This specific variant is also related but not identical to three preconditioners suggested in [8]. This was already discussed in detail in [13].

The remainder of this paper is organized as follows: In Section 2, we introduce the model problem, outline the domain decomposition approach, and present an exact BDDC preconditioner for the globally assembled system. In Sections 3 and 4, we describe three different approximate BDDC preconditioners in a common framework. Namely, we consider an approximate BDDC preconditioner using AMG, a three-level BDDC method, and a vertex-based BDDC preconditioner using a Gauss-Seidel method. Section 5 gives the theory and the condition number bounds for all three approximate BDDC preconditioners. In Section 6, we provide some details of our parallel implementation. In particular, we have implemented all three approximate preconditioners with the same building blocks, which allows us to directly compare the methods with each other regarding their computing time and parallel scalability. Finally, in Section 7, we present and compare results in three spatial dimensions. For all our numerical tests, we consider linear elasticity problems as a model problem since, within our SPPEXA project EXASTEEL, we are particularly interested in problems from solid mechanics. The comparison presented here therefore applies to linear elasticity problems. Note that there are different versions of the vertex-based coarse space, which are also tested for scalar elliptic problems in [9, Section 7].

2. Exact BDDC preconditioner and model problem.

2.1. Linear elasticity and finite elements.

We consider an elastic domain $\Omega \subset \mathbb{R}^3$. We denote by $u : \Omega \to \mathbb{R}^3$ the displacement of the domain, by f a given volume force, and by g a given surface force, respectively. In particular, we assume that one part of the boundary of the domain, $\partial\Omega_D$, is clamped, i.e., has homogeneous Dirichlet boundary conditions, and that the rest, $\partial\Omega_N := \partial\Omega \setminus \partial\Omega_D$, is subject to the surface force g, i.e., a natural boundary condition.

With $\mathbf{H}^1(\Omega) := (H^1(\Omega))^3$, the appropriate space for a variational formulation is the Sobolev space $\mathbf{H}^1_0(\Omega, \partial\Omega_D) := \{\mathbf{v} \in \mathbf{H}^1(\Omega) : \mathbf{v} = \mathbf{0} \text{ on } \partial\Omega_D\}$. The problem of linear elasticity then consists in finding the displacement $\mathbf{u} \in \mathbf{H}^1_0(\Omega, \partial\Omega_D)$ such that

$$\int_\Omega G(\mathbf{x})\, \varepsilon(\mathbf{u}) : \varepsilon(\mathbf{v})\, d\mathbf{x} + \int_\Omega G(\mathbf{x})\beta(\mathbf{x})\, \operatorname{div}\mathbf{u}\, \operatorname{div}\mathbf{v}\, d\mathbf{x} = \langle \mathbf{F}, \mathbf{v} \rangle,$$

for all $\mathbf{v} \in \mathbf{H}^1_0(\Omega, \partial\Omega_D)$ for given material parameters G and β and the right-hand side

$$\langle \mathbf{F}, \mathbf{v} \rangle = \int_\Omega \mathbf{f}^T \mathbf{v}\, d\mathbf{x} + \int_{\partial\Omega_N} \mathbf{g}^T \mathbf{v}\, d\sigma.$$

The material parameters G and β depend on the Young modulus $E > 0$ and the Poisson ratio $\nu \in (0, 1/2)$ given by $G = E/(1+\nu)$ and $\beta = \nu/(1-2\nu)$. Furthermore, the linearized strain tensor $\varepsilon = (\varepsilon_{ij})_{ij}$ is defined by $\varepsilon_{ij}(\mathbf{u}) := \frac{1}{2}(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i})$, and we use the notation

$$\varepsilon(\mathbf{u}) : \varepsilon(\mathbf{v}) := \sum_{i,j=1}^{3} \varepsilon_{ij}(\mathbf{u})\varepsilon_{ij}(\mathbf{v}) \qquad \text{and} \qquad (\varepsilon(\mathbf{u}), \varepsilon(\mathbf{v}))_{L_2(\Omega)} := \int_\Omega \varepsilon(\mathbf{u}) : \varepsilon(\mathbf{v})\, d\mathbf{x}.$$

The corresponding bilinear form associated with linear elasticity can then be written as

$$a(\mathbf{u}, \mathbf{v}) = (G\, \varepsilon(\mathbf{u}), \varepsilon(\mathbf{v}))_{L_2(\Omega)} + (G\beta\, \operatorname{div}\mathbf{u}, \operatorname{div}\mathbf{v})_{L_2(\Omega)}.$$

We discretize our elliptic problem of linear elasticity by low-order, conforming finite elements and thus obtain the linear system of equations

$$K_g u = f_g. \tag{2.1}$$

2.2. Exact BDDC preconditioner for the assembled system. The exact BDDC preconditioner formulation from [19] is applied directly to the system (2.1).

Given is a nonoverlapping domain decomposition $\Omega_i,\ i = 1, \ldots, N$, of Ω such that $\overline{\Omega} = \bigcup_{i=1}^{N} \overline{\Omega}_i$. The interface between the subdomains is defined as $\Gamma := \bigcup_{i=1}^{N} \partial\Omega_i \setminus \partial\Omega$. Each subdomain Ω_i is a union of finite elements, the spaces $W_i,\ i = 1, \ldots, N$, are the local finite element spaces, and the product space is defined by $W = W_1 \times \cdots \times W_N$. The global finite element space V^h corresponds to the triangulation of Ω, and we assume to have an assembly operator R^T, where $R^T : W \to V^h$. By a discretization of the given partial differential equation restricted to Ω_i, we obtain a set of local problems

$$K_i u_i = f_i, \qquad i = 1, \cdots, N.$$

Defining the block operators and right-hand sides

$$K = \begin{bmatrix} K_1 & & \\ & \ddots & \\ & & K_N \end{bmatrix}, \qquad f = \begin{bmatrix} f_1 \\ \vdots \\ f_N \end{bmatrix},$$

we can write $K_g := R^T K R$ and $f_g := R^T f$.

We use the index Γ for degrees of freedom on Γ and the index I for the remaining degrees of freedom except for those on the Dirichlet boundary $\partial\Omega_D$. For the construction of a BDDC preconditioner directly applicable to the assembled linear system $K_g u = f_g$, the interface set of variables on Γ is split into primal (Π) and the remaining dual (Δ) degrees of freedom. Usually, vertex variables are chosen as primal variables, and the primal space is augmented by averages over edges and/or faces.

Let us introduce the space $\widetilde{W} \subset W$ of functions which are continuous in all primal variables and the assembly operators $\breve{R}^T$ and $\widetilde{R}^T$ with $\breve{R}^T : W \to \widetilde{W}$ and $\widetilde{R}^T : \widetilde{W} \to V^h$. Using $\breve{R}$, we can form the partially assembled system

$$\widetilde{K} := \breve{R}^T K \breve{R},$$

and we can also obtain the globally assembled finite element matrix K_g from $\widetilde{K}$ by

$$K_g = \widetilde{R}^T \widetilde{K} \widetilde{R}. \tag{2.2}$$

We denote the interior and interface variables with the indices I and Γ, respectively. Ordering the interior variables first and the interface variables last, we obtain

$$\widetilde{K} = \begin{bmatrix} K_{II} & \widetilde{K}_{\Gamma I}^T \\ \widetilde{K}_{\Gamma I} & \widetilde{K}_{\Gamma\Gamma} \end{bmatrix}.$$

The matrix K_{II} is block-diagonal and applications of K_{II}^{-1} only require local solves on the interior parts of the subdomains and are thus easily parallelizable. We further introduce the union of subdomain interior (I) and dual (Δ) interface degrees of freedom as an extra set of degrees of freedom denoted by the index B, and we obtain an alternative representation of the partially assembled system $\widetilde{K}$ as

$$\widetilde{K} = \begin{bmatrix} K_{BB} & \widetilde{K}_{\Pi B}^T \\ \widetilde{K}_{\Pi B} & \widetilde{K}_{\Pi\Pi} \end{bmatrix}.$$

Like K_{II}, the matrix K_{BB} is a block-diagonal matrix, and applications of K_{BB}^{-1} only require local solves.

Introducing a scaling, e.g., ρ-scaling [16] or deluxe-scaling [7], to the prolongation operators and thereby defining $\widetilde{R}_D : V^h \to \widetilde{W}$, we obtain the BDDC preconditioner for K_g by

$$M_{\text{BDDC}}^{-1} := \left(\widetilde{R}_D^T - \mathcal{H}\, P_D\right) \widetilde{K}^{-1} \left(\widetilde{R}_D - P_D^T \mathcal{H}^T\right); \tag{2.3}$$

see [19]. Here, the operator $\mathcal{H} : \widetilde{W} \to V^h$ is the discrete harmonic extension to the interior of the subdomains given by

$$\mathcal{H} := \begin{bmatrix} 0 & -\left(K_{II}\right)^{-1} \widetilde{K}_{\Gamma I}^T \\ 0 & 0 \end{bmatrix}.$$

Finally, $P_D : \widetilde{W} \to \widetilde{W}$ is a scaled jump operator defined by

$$P_D = I - E_D := I - \widetilde{R}\widetilde{R}_D^T.$$

The original definition often used in the literature is $P_D := B_D^T B$; see [23, Chapter 6] and [19] for more details. There, B is the jump matrix used in FETI-type methods. Please note that in the standard definition, the BDDC preconditioner is formulated for the reduced interface problem, i.e., as

$$M_{\text{BDDC-}\Gamma}^{-1} S_{\Gamma\Gamma} := \widetilde{R}_{D,\Gamma}^T \widetilde{S}_{\Gamma\Gamma}^{-1} \widetilde{R}_{D,\Gamma} S_{\Gamma\Gamma}. \tag{2.4}$$

Here, the prolongation operator $\widetilde{R}_{D,\Gamma}$ is formed in the same way as $\widetilde{R}_D$ but is restricted to the interface variables on Γ, and $S_{\Gamma\Gamma}$ and $\widetilde{S}_{\Gamma\Gamma}$ are the subdomain interface Schur complements of the matrices K_g and $\widetilde{K}$, respectively. Let us remark that the preconditioned system $M_{\text{BDDC}}^{-1} K_g$ has, except for some eigenvalues equal to 1, the same spectrum as the standard BDDC preconditioner formulated using the Schur complement; see [19, Theorem 1]. Here, we provide a related but slightly more direct proof: we first explicitly write the BDDC preconditioner M_{BDDC}^{-1} as

$$\begin{aligned} M_{\text{BDDC}}^{-1} :&= \left(\widetilde{R}_D^T - \mathcal{H}\, P_D\right) \widetilde{K}^{-1} \left(\widetilde{R}_D - P_D^T \mathcal{H}^T\right) \\ &= \begin{bmatrix} I & K_{II}^{-1} \widetilde{K}_{\Gamma I}^T (I - \widetilde{R}_\Gamma \widetilde{R}_{D,\Gamma}^T) \\ 0 & \widetilde{R}_{D,\Gamma}^T \end{bmatrix} \widetilde{K}^{-1} \begin{bmatrix} I & 0 \\ (I - \widetilde{R}_{D,\Gamma} \widetilde{R}_\Gamma^T) \widetilde{K}_{\Gamma I} K_{II}^{-1} & \widetilde{R}_{D,\Gamma} \end{bmatrix} \\ &= \begin{bmatrix} I & K_{II}^{-1} \widetilde{K}_{\Gamma I}^T (I - E_{D,\Gamma}) \\ 0 & \widetilde{R}_{D,\Gamma}^T \end{bmatrix} \widetilde{K}^{-1} \begin{bmatrix} I & 0 \\ (I - E_{D,\Gamma}^T) \widetilde{K}_{\Gamma I} K_{II}^{-1} & \widetilde{R}_{D,\Gamma} \end{bmatrix}. \end{aligned}$$

Using the block factorization

$$\widetilde{K}^{-1} = \begin{bmatrix} I & -K_{II}^{-1} \widetilde{K}_{\Gamma I}^T \\ 0 & I \end{bmatrix} \begin{bmatrix} K_{II}^{-1} & 0 \\ 0 & \widetilde{S}_{\Gamma\Gamma}^{-1} \end{bmatrix} \begin{bmatrix} I & 0 \\ -\widetilde{K}_{\Gamma I} K_{II}^{-1} & I \end{bmatrix},$$

we obtain by direct computation the alternative representation

$$M_{\text{BDDC}}^{-1} = \begin{bmatrix} K_{II}^{-1} + K_{II}^{-1}\widetilde{K}_{\Gamma I}^T E_{D,\Gamma}\widetilde{S}_{\Gamma\Gamma}^{-1}E_{D,\Gamma}^T\widetilde{K}_{\Gamma I}K_{II}^{-1} & -K_{II}^{-1}\widetilde{K}_{\Gamma I}^T E_{D,\Gamma}\widetilde{S}_{\Gamma\Gamma}^{-1}\widetilde{R}_{D,\Gamma} \\ -\widetilde{R}_{D,\Gamma}^T\widetilde{S}_{\Gamma\Gamma}^{-1}E_{D,\Gamma}^T\widetilde{K}_{\Gamma I}K_{II}^{-1} & \widetilde{R}_{D,\Gamma}^T\widetilde{S}_{\Gamma\Gamma}^{-1}\widetilde{R}_{D,\Gamma} \end{bmatrix}.$$

Forming $M_{\text{BDDC}}^{-1}K_g$ finally yields

$$M_{\text{BDDC}}^{-1}K_g = \begin{bmatrix} I & U \\ 0 & M_{\text{BDDC}-\Gamma}^{-1}S_{\Gamma\Gamma} \end{bmatrix}$$

with $U = K_{II}^{-1}K_{I\Gamma} - K_{II}^{-1}K_{\Gamma I}^T\widetilde{R}_{D,\Gamma}^T\widetilde{S}_{\Gamma\Gamma}^{-1}\widetilde{R}_{D,\Gamma}S_{\Gamma\Gamma}$, using that $E_{D,\Gamma} = \widetilde{R}_\Gamma\widetilde{R}_{D,\Gamma}^T$ and that $K_{\Gamma I} = \widetilde{R}_\Gamma^T\widetilde{K}_{\Gamma I}$. Here, $M_{\text{BDDC}-\Gamma}^{-1}$ is the classical BDDC preconditioner for the Schur complement; see equation (2.4). The result then follows from the fact that the set of eigenvalues of a block-triangular matrix equals the union of the sets of eigenvalues of the diagonal blocks.

3. Approximate BDDC preconditioners.

All approximate BDDC methods considered in this paper are based on an approximate solution of the coarse problem of BDDC. To ensure a simple comparison, all approximate preconditioners are implemented using the same software framework; see also [12, 13].

By block factorization we obtain

$$\widetilde{K}^{-1} = \begin{bmatrix} K_{BB}^{-1} & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} -K_{BB}^{-1}\widetilde{K}_{\Pi B}^T \\ I \end{bmatrix}\widetilde{S}_{\Pi\Pi}^{-1}\begin{bmatrix} -\widetilde{K}_{\Pi B}K_{BB}^{-1} & I \end{bmatrix}, \tag{3.1}$$

where $\widetilde{S}_{\Pi\Pi}$ is the Schur complement

$$\widetilde{S}_{\Pi\Pi} = \widetilde{K}_{\Pi\Pi} - \widetilde{K}_{\Pi B}\,K_{BB}^{-1}\,\widetilde{K}_{\Pi B}^T.$$

Note that $\widetilde{S}_{\Pi\Pi}$ represents the coarse BDDC operator.

Replacing $\widetilde{S}_{\Pi\Pi}^{-1}$ by an approximation $\widehat{S}_{\Pi\Pi}^{-1}$ in (3.1), we obtain an approximation for $\widetilde{K}^{-1}$ by

$$\widehat{K}^{-1} = \begin{bmatrix} K_{BB}^{-1} & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} -K_{BB}^{-1}\widetilde{K}_{\Pi B}^T \\ I \end{bmatrix}\widehat{S}_{\Pi\Pi}^{-1}\begin{bmatrix} -\widetilde{K}_{\Pi B}K_{BB}^{-1} & I \end{bmatrix}. \tag{3.2}$$

Replacing $\widetilde{K}^{-1}$ in (2.3) by $\widehat{K}^{-1}$, we define an approximation to the BDDC preconditioner, i.e.,

$$\widehat{M}^{-1} := \left(\widetilde{R}_D^T - \mathcal{H}P_D\right)\widehat{K}^{-1}\left(\widetilde{R}_D - P_D^T\mathcal{H}^T\right). \tag{3.3}$$

For the remainder of the article, all approximate BDDC preconditioners are marked with a hat. In the following sections, we compare three different approaches to form $\widehat{S}_{\Pi\Pi}^{-1}$ for the approximation of the coarse solve:

a) using AMG (algebraic multigrid) denoted by $\widehat{M}_{\text{BDDC, AMG}}^{-1}$;
b) using exact BDDC recursively denoted by $\widehat{M}_{\text{BDDC, 3L}}^{-1}$;
c) using an exact solution of a smaller vertex-based coarse space denoted by $\widehat{M}_{\text{BDDC, VB}}^{-1}$.

Let us remark that $\widehat{M}_{\text{BDDC, AMG}}^{-1}$ was denoted as $\widehat{M}_3^{-1}$ in [13].

4. Examples of approximate BDDC preconditioners.

In this section, we give three examples of approximate BDDC preconditioners presented by adopting the notation introduced in Section 3. First, we consider an approximate BDDC preconditioner using AMG to precondition $\widetilde{S}_{\Pi\Pi}$, second, a three-level BDDC method using BDDC itself to precondition $\widetilde{S}_{\Pi\Pi}$, and third, a vertex-based BDDC preconditioner using a Jacobi/Gauss-Seidel method in combination with a vertex-based coarse space to precondition $\widetilde{S}_{\Pi\Pi}$.

4.1. BDDC preconditioner with AMG coarse preconditioner. Let us denote the application of a fixed number of V-cycles of an AMG method to $\widetilde{S}_{\Pi\Pi}$ by M_{AMG}^{-1}. By choosing M_{AMG}^{-1} in (3.2) as an approximation of $\widetilde{S}_{\Pi\Pi}$, i.e., by choosing $\widehat{S}_{\Pi\Pi}^{-1} := M_{\text{AMG}}^{-1}$, we obtain

$$\widehat{K}_{\text{AMG}}^{-1} = \begin{bmatrix} K_{BB}^{-1} & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} -K_{BB}^{-1}\widetilde{K}_{\Pi B}^{T} \\ I \end{bmatrix} M_{\text{AMG}}^{-1} \begin{bmatrix} -\widetilde{K}_{\Pi B}K_{BB}^{-1} & I \end{bmatrix}.$$

Again, by using $\widehat{K}_{\text{AMG}}^{-1}$ as an approximation for $\widetilde{K}^{-1}$ in (3.3), we obtain the inexact reduced preconditioner $\widehat{M}_{\text{BDDC, AMG}}^{-1}$.

4.2. A three-level BDDC. Alternatively, if we construct an exact BDDC preconditioner $\widehat{S}_{\Pi\Pi}^{-1}$ for the Schur complement matrix $\widetilde{S}_{\Pi\Pi}$, then (3.3) will become a three-level BDDC preconditioner $\widehat{M}_{\text{BDDC, 3L}}^{-1}$. This approach is equivalent to the three-level preconditioner introduced in [24] but formulated for the original matrix K_g. In [24], the BDDC formulation for the Schur complement system on the interface is used and applied recursively. Since we use the BDDC formulation for the complete system matrix K_g, we consequently apply this approach to form the third level. We thus follow Section 2.2 and mark all operators and spaces defined for the third level with bars, e.g., $\overline{I}$ represents the interior variables of the third level, while I represent those on the second level. In Section 5, we derive the same condition number bound as in [24, 25].

Let us now describe the application of BDDC to $\widetilde{S}_{\Pi\Pi}$ in some more details. The basic idea of the three-level BDDC preconditioner is to recursively introduce a further level of the decomposition of the domain Ω into $\overline{N}$ subregions $\Omega^1, ..., \Omega^{\overline{N}}$. Each subregion is built from a given number of subdomains. All primal variables Π on the subdomain level are then again partitioned into interior, primal, and dual variables, denoted by $\overline{I}, \overline{\Pi}$, and $\overline{\Delta}$, with respect to the subregions; see also Figure 4.1 for a possible selection in 2D. Now, in principle, the subdomains take over the role of finite elements on the third level and the subregions the role of the subdomains. The basis functions of the third level are the coarse basis functions of the second level, localized to the subregions.

We therefore first define the space $\overline{V}^h$, which is spanned by all coarse basis functions of the second level and denote by $\overline{W}_i$, $i = 1, ..., \overline{N}$, the spaces which are spanned by the restrictions of the coarse basis functions to the subregions Ω^i, $i = 1, ..., \overline{N}$. The product space $\overline{W}$ is now defined as $\overline{W} = \overline{W}_1 \times ... \times \overline{W}_{\overline{N}}$.

Using local Schur complements $S_{\Pi\Pi}^{(i)} = K_{\Pi\Pi}^{(i)} - K_{\Pi B}^{(i)} K_{BB}^{(i)-1} K_{\Pi B}^{(i)T}$ on the subdomains and the block matrix $S_{\Pi\Pi} = \operatorname{diag}(S_{\Pi\Pi}^{(1)}, ..., S_{\Pi\Pi}^{(N)})$, we can redefine

$$\widetilde{S}_{\Pi\Pi} = \sum_{i=1}^{N} R_{\Pi}^{(i)T} S_{\Pi\Pi}^{(i)} R_{\Pi}^{(i)},$$

where $R^T = \left(R^{(1)T}, ..., R^{(N)T}\right)$ and $R^{(i)} = \operatorname{diag}\left(R_B^{(i)}, R_{\Pi}^{(i)}\right)$, $i = 1, ..., N$. We can now perform this assembly process only on the subregions, i.e.,

$$\overline{S}_j = \sum_{i=1}^{N_j} R_{\Pi}^{(i)T} S_{\Pi\Pi}^{(i)} R_{\Pi}^{(i)}, \qquad \forall j = 1, ..., \overline{N},$$

where N_j is the number of subdomains belonging to the subregion Ω^j. Obviously, $\widetilde{S}_{\Pi\Pi}$ takes over the role of K_g on the third level, while $\overline{S}_j$ takes over the role of K_i. Consequently, defining a prolongation $\overline{R} : \overline{V}^h \to \overline{W}$, we can also write

$$\widetilde{S}_{\Pi\Pi} = \overline{R}^T \overline{S}\, \overline{R},$$

with $\overline{S} = \operatorname{diag}(\overline{S}_1, ..., \overline{S}_{\overline{N}})$.

Let us introduce the space $\widetilde{\overline{W}} \subset \overline{W}$ of functions which are continuous in all primal variables $\overline{\Pi}$ on the third level and the assembly operators $\check{\overline{R}}^T : \overline{W} \to \widetilde{\overline{W}}$ and $\widetilde{\overline{R}}^T : \widetilde{\overline{W}} \to \overline{V}^h$. Using $\check{\overline{R}}$, we can form the partially assembled system

$$\widetilde{\overline{S}} := \check{\overline{R}}^T \overline{S} \check{\overline{R}}.$$

Adding scalings to the prolongations as before and thus defining $\widetilde{\overline{R}}_D : \overline{V}^h \to \widetilde{\overline{W}}$, we obtain the BDDC preconditioner for the third level by

$$\overline{M}_{\text{BDDC}}^{-1} := \left(\widetilde{\overline{R}}_D^T - \overline{\mathcal{H}} \, \overline{P}_D \right) \widetilde{\overline{S}}^{-1} \left(\widetilde{\overline{R}}_D - \overline{P}_D^T \overline{\mathcal{H}}^T \right).$$

The operator $\overline{\mathcal{H}} : \widetilde{\overline{W}} \to \overline{V}^h$ is the discrete harmonic extension to the interior of the subregions and writes

$$\overline{\mathcal{H}} := \begin{bmatrix} 0 & -\left(\overline{S}_{\overline{II}}\right)^{-1} \widetilde{\overline{S}}_{\overline{\Gamma I}}^T \\ 0 & 0 \end{bmatrix},$$

with the blocks $\overline{S}_{\overline{II}}$ and $\widetilde{\overline{S}}_{\overline{\Gamma I}}$ of the partially assembled matrix

$$\widetilde{\overline{S}} = \begin{bmatrix} \overline{S}_{\overline{II}} & \widetilde{\overline{S}}_{\overline{\Gamma I}}^T \\ \widetilde{\overline{S}}_{\overline{\Gamma I}} & \widetilde{\overline{S}}_{\overline{\Gamma\Gamma}} \end{bmatrix}$$

and the jump operator defined as $\overline{P}_D := I - \widetilde{\overline{R}}\widetilde{\overline{R}}_D^T$.

Now, by choosing $\widehat{S}_{\Pi\Pi}^{-1} := \overline{M}_{\text{BDDC}}^{-1}$ as an approximation for $\widetilde{S}_{\Pi\Pi}^{-1}$ in (3.2), we obtain

$$\widehat{K}_{\text{3L}}^{-1} = \begin{bmatrix} K_{BB}^{-1} & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} -K_{BB}^{-1} \widetilde{K}_{\Pi B}^T \\ I \end{bmatrix} \overline{M}_{\text{BDDC}}^{-1} \begin{bmatrix} -\widetilde{K}_{\Pi B} K_{BB}^{-1} & I \end{bmatrix}$$

and can define

$$\widehat{M}_{\text{BDDC, 3L}}^{-1} := \left(\widetilde{R}_D^T - \mathcal{H} P_D \right) \widehat{K}_{\text{3L}}^{-1} \left(\widetilde{R}_D - P_D^T \mathcal{H}^T \right).$$

Instead of inverting $\widetilde{\overline{S}}$ directly, we again can use a block factorization

$$\widetilde{\overline{S}}^{-1} = \begin{bmatrix} \overline{S}_{\overline{BB}}^{-1} & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} -\overline{S}_{\overline{BB}}^{-1} \widetilde{\overline{S}}_{\overline{\Pi B}}^T \\ I \end{bmatrix} \widetilde{\overline{T}}_{\overline{\Pi\Pi}}^{-1} \begin{bmatrix} -\widetilde{\overline{S}}_{\overline{\Pi B}} \overline{S}_{\overline{BB}}^{-1} & I \end{bmatrix},$$

where the primal Schur complement on the subregion level is

$$\widetilde{\overline{T}}_{\overline{\Pi\Pi}} = \widetilde{\overline{S}}_{\overline{\Pi\Pi}} - \widetilde{\overline{S}}_{\overline{\Pi B}} \overline{S}_{\overline{BB}}^{-1} \widetilde{\overline{S}}_{\overline{\Pi B}}^T.$$

Note that, following [19, Theorem 1], the preconditioned system $\overline{M}_{\text{BDDC}}^{-1} \widetilde{S}_{\Pi\Pi}$ on the subregion level has the same eigenvalues as $\widetilde{\overline{R}}_{D,\overline{\Gamma}}^T \widetilde{\overline{T}}_{\overline{\Gamma\Gamma}}^{-1} \widetilde{\overline{R}}_{D,\overline{\Gamma}} \widetilde{T}_{\overline{\Gamma\Gamma}}$ except for some eigenvalues equal to 1. Here, we have the Schur complement $\widetilde{T}_{\overline{\Gamma\Gamma}}$ of $\widetilde{S}_{\Pi\Pi}$ on the interface of the subregions, the primally assembled Schur complement $\widetilde{\overline{T}}_{\overline{\Gamma\Gamma}}$ of $\widetilde{\overline{S}}$ on the interface of the subregions, and the splitting $\widetilde{\overline{R}}_D = \operatorname{diag}(I_{\overline{I}} \quad \widetilde{\overline{R}}_{D,\overline{\Gamma}})$. Therefore, we can use the condition number estimates provided in [24, 25] as in Section 5.

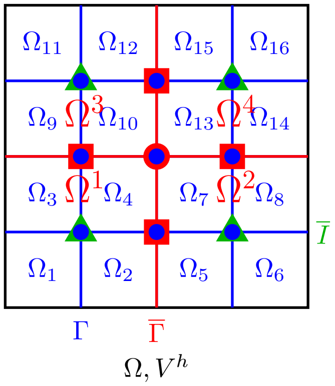

FIG. 4.1. *Example of a domain decomposition in 2D into 16 subdomains and 4 subregions recursively. We mark in blue the interface Γ between subdomains and in red the interface $\overline{\Gamma}$ between subregions. Primal nodes $\overline{\Pi}$ w.r.t. the subregions are depicted as red circles, while primal nodes Π w.r.t. the subdomains are depicted as blue circles. Inner or dual nodes w.r.t the subregions, i.e., $\overline{I}$ or, respectively, $\overline{\Delta}$ are depicted as green triangles or, respectively, red squares.*

4.3. Vertex-based BDDC preconditioner. We finally describe the vertex-based preconditioner for the coarse problem as introduced by Dohrmann, Pierson, and Widlund [9] in our framework—denoting it as the vertex-based preconditioner $\widehat{M}^{-1}_{\text{BDDC, VB}}$. Here the preconditioner for the coarse problem can be interpreted as a standard two-level additive or multiplicative Schwarz algorithm. In particular, the direct solution of the coarse problem $\widetilde{S}^{-1}_{\Pi\Pi}$ is replaced by a preconditioner M^{-1}_{VB} based on a smaller vertex-based coarse space.

It was shown early in the history of FETI-DP and BDDC that vertex nodes alone as coarse nodes do not give us competitive algorithms [10, 16]. Instead, coarse degrees of freedom for BDDC or FETI-DP are often associated with average values over certain equivalence classes, i.e., edges and/or faces. The basic idea of the coarse component of the preconditioner M^{-1}_{VB} is to approximate the averages over edges or faces using adjacent vertex values. This technique allows to delay the point when a new level has to be introduced, and, in a multilevel context, may help to reduce the number of levels.

We denote the vertex-based coarse space by $\widetilde{W}_{\Psi}$ and the original coarse space by $\widetilde{W}_{\Pi}$. Then as in [9] we define $\Psi : \widetilde{W}_{\Psi} \to \widetilde{W}_{\Pi}$ as the coarse interpolant between the coarse space based on vertices and the original coarse space based on certain equivalence classes. It is important that the coarse basis functions of $\widetilde{W}_{\Psi}$, i.e., the columns of Ψ, provide a partition of unity in the original coarse space $\widetilde{W}_{\Pi}$. This is fulfilled, e.g., for the following definition of Ψ suggested in [9]. Let us first assume that $\widetilde{W}_{\Pi}$ consists of edge averages only. Then, each row of Ψ corresponds to a single edge constraint and has, in the case of an inner edge, two entries of $1/2$ in the two columns corresponding to the two vertices located at the endpoints of the edge. All other entries of the row are zero. In case of an edge touching the Dirichlet boundary with one endpoint, the corresponding row has a single entry of 1 in the column corresponding to the vertex located at the other end of the edge. Analogously, a partition of unity can be formed for the coarse spaces $\widetilde{W}_{\Pi}$ consisting of face constraints.

Again as in [9], we define $\widetilde{S}_{\Pi\Pi,r} := \Psi^T \widetilde{S}_{\Pi\Pi} \Psi$ as the reduced coarse matrix. Note that the number of rows and columns of $\widetilde{S}_{\Pi\Pi,r}$ equals the number of vertices for scalar problems. The preconditioner M^{-1}_{VB} for the coarse matrix $\widetilde{S}_{\Pi\Pi}$ is then given as

$$M^{-1}_{\text{VB}} = \Psi \widetilde{S}^{-1}_{\Pi\Pi,r} \Psi^T + \text{GS}(\widetilde{S}_{\Pi\Pi}), \tag{4.1}$$

where GS denotes the application of a Gauss-Seidel preconditioner. In fact, M_{VB}^{-1} is simply a Gauss-Seidel preconditioner with an additive coarse correction.

In [9], only edge averages or solely face averages are used which are each reduced to vertex-based coarse spaces as described above. In general, also the combination of vertices, edge, and face averages as coarse components can be considered and can be reduced to a solely vertex-based coarse space.

Now we can define the vertex-based approximate BDDC preconditioner by choosing $\widehat{S}_{\Pi\Pi}^{-1} := M_{\text{VB}}^{-1}$ as an approximation for $\widetilde{S}_{\Pi\Pi}^{-1}$ in (3.2). We then obtain the approximation $\widehat{K}_{\text{VB}}^{-1}$ of $\widetilde{K}^{-1}$ as

$$\widehat{K}_{\text{VB}}^{-1} = \begin{bmatrix} K_{BB}^{-1} & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} -K_{BB}^{-1}\widetilde{K}_{\Pi B}^{T} \\ I \end{bmatrix} M_{\text{VB}}^{-1} \begin{bmatrix} -\widetilde{K}_{\Pi B}K_{BB}^{-1} & I \end{bmatrix},$$

and finally

$$\widehat{M}_{\text{BDDC, VB}}^{-1} = \left(\widetilde{R}_D^T - \mathcal{H}\,P_D\right)\widehat{K}_{\text{VB}}^{-1}\left(\widetilde{R}_D - P_D^T\mathcal{H}^T\right)$$

using the notation from (3.3); see also [9].

5. Condition number bounds. First, we need to make two assumptions, which are equivalent to Assumptions 1 and 2 in [19].

ASSUMPTION 1. *For the averaging operator $E_{D,2} := \widetilde{R}(\widetilde{R}_D^T - \mathcal{H}P_D)$ we have*

$$|E_{D,2}|_{\widetilde{K}}^2 \leq \Phi(H,h)|w|_{\widetilde{K}}^2, \qquad \forall w \in \widetilde{W},$$

with $\Phi(H,h)$ being a function of the mesh size h and the subdomain diameter H.

Under Assumption 1, the condition number of the exactly preconditioned system is bounded by

$$\kappa(M_{\text{BDDC}}^{-1}K_g) \leq \Phi(H,h); \tag{5.1}$$

see, e.g., Theorem 3 in [19].

If appropriate primal constraints, e.g., edge averages and vertex constraints, are chosen, then we obtain the condition number bound with $\Phi(H,h) = C(1+\log(H/h))^2$ for our homogeneous linear elasticity test case; see Section 6.

ASSUMPTION 2. *There are positive constants $\tilde{c}$ and $\widetilde{C}$, which might depend on h and H, such that*

$$\tilde{c}u^T\widetilde{K}u \leq u^T\widehat{K}u \leq \widetilde{C}u^T\widetilde{K}u, \qquad \forall u \in \widetilde{W}.$$

We can now prove a result, Theorem 5.1, for the preconditioned operator $\widehat{M}^{-1}K_g$. In the proof, we basically follow the arguments in the proof of Theorem 4 in [19], but here we use exact discrete harmonic extension operators, i.e., an exact $E_{D,2}$. This is in contrast to Theorem 4 in [19], where inexact discrete harmonic extensions are used, which is not necessary in our case. Although large parts of the proof are identical, we include the complete line of arguments here for the convenience of the reader.

THEOREM 5.1. *Let Assumptions 1 and 2 hold. Then the preconditioned operator $\widehat{M}^{-1}K_g$ is symmetric, positive definite with respect to the bilinear form $\langle\cdot,\cdot\rangle_{K_g}$, and we have*

$$\frac{1}{\widetilde{C}}\langle u,u\rangle_{K_g} \leq \langle\widehat{M}^{-1}K_g u,u\rangle_{K_g} \leq \frac{\Phi(H,h)}{\tilde{c}}\langle u,u\rangle_{K_g}, \qquad \forall u \in V^h.$$

Therefore, we obtain the condition number bound $\kappa(\widehat{M}^{-1}K_g) \leq \frac{\widetilde{C}}{\tilde{c}}\Phi(H,h)$.

Proof. Let $u \in V^h$ be given. We define

$$w = \widehat{K}^{-1}(\widetilde{R}_D - P_D^T \mathcal{H}^T) K_g u \ \in \widetilde{W} \tag{5.2}$$

and thus also have

$$\widehat{K} w = (\widetilde{R}_D - P_D^T \mathcal{H}^T) K_g u.$$

Using $\widetilde{R}^T \widetilde{R}_D = I$ yields $\widetilde{R}^T P_D^T = \widetilde{R}^T (I - \widetilde{R}_D \widetilde{R}^T) = 0$ and thus $\operatorname{range}(P_D^T) \subset \operatorname{null}(\widetilde{R}^T)$. Hence, we obtain

$$\langle u, u \rangle_{K_g} = u^T \widetilde{R}^T (\widetilde{R}_D - P_D^T \mathcal{H}^T) K_g u = u^T \widetilde{R}^T \widehat{K} w = \langle w, \widetilde{R} u \rangle_{\widehat{K}}. \tag{5.3}$$

Using the Cauchy-Schwarz inequality and Assumption 2, we can further estimate

$$\begin{aligned} \langle w, \widetilde{R} u \rangle_{\widehat{K}} \leq \langle w, w \rangle_{\widehat{K}}^{1/2} \langle \widetilde{R} u, \widetilde{R} u \rangle_{\widehat{K}}^{1/2} &\overset{\text{Asm. 2}}{\leq} \sqrt{\widetilde{C}} \langle w, w \rangle_{\widehat{K}}^{1/2} \langle \widetilde{R} u, \widetilde{R} u \rangle_{\widetilde{K}}^{1/2} \\ &\overset{(2.2)}{=} \sqrt{\widetilde{C}} \langle w, w \rangle_{\widehat{K}}^{1/2} \langle u, u \rangle_{K_g}^{1/2}. \end{aligned} \tag{5.4}$$

Combining equations (5.3) and (5.4), we have $\langle u, u \rangle_{K_g} \leq \widetilde{C} \langle w, w \rangle_{\widehat{K}}$. Using (5.2) and (3.3), we can prove the lower bound.

$$\begin{aligned} \frac{1}{\widetilde{C}} \langle u, u \rangle_{K_g} &\leq \langle w, w \rangle_{\widehat{K}} \\ &\overset{(5.2)}{=} u^T K_g (\widetilde{R}_D^T - \mathcal{H} P_D) \widehat{K}^{-1} \widehat{K} \widehat{K}^{-1} (\widetilde{R}_D - P_D^T \mathcal{H}^T) K_g) u \\ &= \langle u, (\widetilde{R}_D^T - \mathcal{H} P_D) \widehat{K}^{-1} (\widetilde{R}_D - P_D^T \mathcal{H}^T) K_g u \rangle_{K_g} \\ &\overset{(3.3)}{=} \langle u, \widehat{M}^{-1} K_g u \rangle_{K_g}. \end{aligned} \tag{5.5}$$

Let us now prove the upper bound using Assumption 1, (5.2), and (3.3).

$$\begin{aligned} \langle \widehat{M}^{-1} K_g u, \widehat{M}^{-1} K_g u \rangle_{K_g} &= \langle (\widetilde{R}_D^T - \mathcal{H} P_D) w, (\widetilde{R}_D^T - \mathcal{H} P_D) w \rangle_{K_g} \\ &= \langle \widetilde{R} (\widetilde{R}_D^T - \mathcal{H} P_D) w, \widetilde{R} (\widetilde{R}_D^T - \mathcal{H} P_D) w \rangle_{\widetilde{K}} \\ &= \langle E_{D,2} w, E_{D,2} w \rangle_{\widetilde{K}} = |E_{D,2} w|_{\widetilde{K}}^2 \\ &\overset{\text{Asm. 1}}{\leq} \Phi(H, h) |w|_{\widetilde{K}}^2. \end{aligned} \tag{5.6}$$

Together with Assumption 2, we obtain

$$\begin{aligned} \langle \widehat{M}^{-1} K_g u, \widehat{M}^{-1} K_g u \rangle_{K_g} \overset{(5.6)}{\leq} \Phi(H, h) |w|_{\widetilde{K}}^2 &\overset{Asm. 2}{\leq} \frac{1}{\tilde{c}} \Phi(H, h) |w|_{\widehat{K}}^2 \\ &\overset{(5.5)}{=} \frac{1}{\tilde{c}} \Phi(H, h) \langle u, \widehat{M}^{-1} K_g u \rangle_{K_g}. \end{aligned} \tag{5.7}$$

Using the Cauchy-Schwarz inequality in combination with (5.7), we finally obtain

$$\langle u, \widehat{M}^{-1} K_g u \rangle_{K_g} \leq \frac{\Phi(H, h)}{\tilde{c}} \langle u, u \rangle_{K_g}. \qquad \square$$

For the preconditioners considered here, we replace the inverse operator of the Schur complement in the primal variables $\widetilde{S}_{\Pi\Pi}^{-1}$ by an approximation $\widehat{S}_{\Pi\Pi}^{-1}$. Therefore, we have to show that Assumption 2 used in the proof of Theorem 5.1 is still relevant and holds under certain assumptions.

ASSUMPTION 3. *There are positive constants $\hat{c}$ and $\widehat{C}$, which might depend on h and H, such that*

$$\hat{c}\tilde{u}_\Pi^T \widetilde{S}_{\Pi\Pi}\tilde{u}_\Pi \le \tilde{u}_\Pi^T \widehat{S}_{\Pi\Pi}\tilde{u}_\Pi \le \widehat{C}\tilde{u}_\Pi^T \widetilde{S}_{\Pi\Pi}\tilde{u}_\Pi, \qquad \forall \tilde{u}_\Pi \in \widetilde{W}_\Pi.$$

We can now prove the following lemma.

LEMMA 5.2. *Let Assumption 3 hold and $\widehat{K}^{-1}$ be defined as in equation (3.2). Then Assumption 2 holds with $\tilde{c} := \min(\hat{c}, 1)$ and $\widetilde{C} := \max(\widehat{C}, 1)$.*

Proof. We first split $\widehat{K}^{-1} = A_1 + A_2$ into its two additive parts

$$A_1 := \begin{bmatrix} K_{BB}^{-1} & 0 \\ 0 & 0 \end{bmatrix} \qquad \text{and} \qquad A_2 := \begin{bmatrix} -K_{BB}^{-1}\widetilde{K}_{\Pi B}^T \\ I \end{bmatrix} \widehat{S}_{\Pi\Pi}^{-1} \begin{bmatrix} -\widetilde{K}_{\Pi B}K_{BB}^{-1} & I \end{bmatrix}.$$

The product $A_1\widetilde{K}$ yields

$$A_1\widetilde{K} = \begin{bmatrix} K_{BB}^{-1} & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} K_{BB} & \widetilde{K}_{\Pi B}^T \\ \widetilde{K}_{\Pi B} & \widetilde{K}_{\Pi\Pi} \end{bmatrix} = \begin{bmatrix} I & K_{BB}^{-1}\widetilde{K}_{\Pi B}^T \\ 0 & 0 \end{bmatrix}. \tag{5.8}$$

By a direct computation we obtain

$$\begin{aligned} A_2\widetilde{K} &= \begin{bmatrix} -K_{BB}^{-1}\widetilde{K}_{\Pi B}^T \\ I \end{bmatrix} \widehat{S}_{\Pi\Pi}^{-1} \begin{bmatrix} -\widetilde{K}_{\Pi B}K_{BB}^{-1} & I \end{bmatrix} \begin{bmatrix} K_{BB} & \widetilde{K}_{\Pi B}^T \\ \widetilde{K}_{\Pi B} & \widetilde{K}_{\Pi\Pi} \end{bmatrix} \\ &= \begin{bmatrix} -K_{BB}^{-1}\widetilde{K}_{\Pi B}^T \\ I \end{bmatrix} \widehat{S}_{\Pi\Pi}^{-1} \begin{bmatrix} 0 & \widetilde{S}_{\Pi\Pi} \end{bmatrix} = \begin{bmatrix} -K_{BB}^{-1}\widetilde{K}_{\Pi B}^T \\ I \end{bmatrix} \begin{bmatrix} 0 & \widehat{S}_{\Pi\Pi}^{-1}\widetilde{S}_{\Pi\Pi} \end{bmatrix} \\ &= \begin{bmatrix} 0 & -K_{BB}^{-1}\widetilde{K}_{\Pi B}^T\widehat{S}_{\Pi\Pi}^{-1}\widetilde{S}_{\Pi\Pi} \\ 0 & \widehat{S}_{\Pi\Pi}^{-1}\widetilde{S}_{\Pi\Pi} \end{bmatrix}. \end{aligned} \tag{5.9}$$

Adding (5.8) and (5.9) yields the final result

$$\widehat{K}^{-1}\widetilde{K} = \begin{bmatrix} I & G \\ 0 & \widehat{S}_{\Pi\Pi}^{-1}\widetilde{S}_{\Pi\Pi} \end{bmatrix}$$

with $G = K_{BB}^{-1}\widetilde{K}_{\Pi B}^T(I - \widehat{S}_{\Pi\Pi}^{-1}\widetilde{S}_{\Pi\Pi})$. Therefore, except for additional eigenvalues equal to 1, $\widehat{K}^{-1}\widetilde{K}$ and $\widehat{S}_{\Pi\Pi}^{-1}\widetilde{S}_{\Pi\Pi}$ have the same spectrum, and we have

$$\lambda_{\min}(\widehat{K}^{-1}\widetilde{K}) = \min\Big(\lambda_{\min}(\widehat{S}_{\Pi\Pi}^{-1}\widetilde{S}_{\Pi\Pi}), 1\Big), \quad \lambda_{\max}(\widehat{K}^{-1}\widetilde{K}) = \max\Big(\lambda_{\max}(\widehat{S}_{\Pi\Pi}^{-1}\widetilde{S}_{\Pi\Pi}), 1\Big).$$

Consequently, Assumption 2 holds with $\tilde{c} := \min(\hat{c}, 1)$ and $\widetilde{C} := \max(\widehat{C}, 1)$. □

For the preconditioner $\widehat{M}_{\text{BDDC, AMG}}^{-1}$, we now find that $\widehat{C}$ and $\hat{c}$ depend on the properties of the AMG V-cycle used and therefore

$$\kappa(\widehat{M}_{\text{BDDC, AMG}}^{-1}K_g) \le \frac{\widetilde{C}}{\tilde{c}}\Phi(H, h) = \frac{\max(\widehat{C}, 1)}{\min(\hat{c}, 1)}\Phi(H, h). \tag{5.10}$$

For the three-level BDDC preconditioner $\widehat{M}_{\text{BDDC, 3L}}^{-1}$ we obtain with [25, Lemma 4.6] in two spatial dimensions and [24, Lemma 4.7] in three spatial dimensions that $\hat{c} = \frac{1}{C_{3L}\left(1+\log(\frac{\hat{H}}{H})\right)^2}$

and $\widehat{C} = 1$. Here, $\hat{H}$ is the maximum diameter of a subregion, and of course, depending on the problem and dimension, sufficient primal constraints on the second level have to be chosen; see [24, 25]. Let us note that the results in [24, 25] are only proven for scalar diffusion problems. To the best of our knowledge an extension to linear elasticity has not been published so far and might still be an open problem. Using Lemma 5.2 and Theorem 5.1, we obtain for scalar elliptic problems the condition number bound

$$\kappa(\widehat{M}_{\text{BDDC, 3L}}^{-1} K_g) \leq \frac{\widetilde{C}}{\tilde{c}} \Phi(H,h) = C_{\text{3L}} \left(1 + \log\left(\frac{\hat{H}}{H}\right)\right)^2 \Phi(H,h); \tag{5.11}$$

see also [24, 25].

For the vertex-based BDDC preconditioner $\widehat{M}_{\text{BDDC, VB}}^{-1}$ we obtain, with [9, Theorem 3] for edge-based or face-based coarse spaces and quasi-monotone face-connected paths that $\hat{c} \geq \frac{1}{C_1}$, $\max(\widehat{C}, 1) \leq C_C$ and $\Phi(H,h) = C\left(1 + \log(\frac{H}{h})\right)^2$; see [9, Theorem 3]. Here, C_C is obtained by a coloring argument and therefore usually $C_C \geq 1$. The constant C_1 depends on geometric constants, e.g., the maximum number of subdomains connected by an edge (see [9, Lemma 2]), the maximum number of neighbors of a subdomain (see [9, (4.3)]), or typical subdomain sizes (see [9, Assumption 3]). Additionally, C_1 depends on a tolerance for the lowest coefficient along an acceptable path; see [9, Assumption 1 and 2]; cf. also [18]. The results in [9] are proven for scalar diffusion and linear elasticity problems. All together, with another constant C_{VB}, we obtain

$$\frac{\max(\widehat{C}, 1)}{\min(\hat{c}, 1)} \leq C_{\text{VB}};$$

see also [9, Theorem 1 and 3] where $\hat{c} = \beta_1$ and $\widehat{C} = \beta_2$ for the constants β_1 and β_2 used in [9]. Typically, we have $C_1 \geq 1$, and we can then define $C_{\text{VB}} = C_1 \cdot C_C$. Using Theorem 5.1, we thus obtain the condition number bound

$$\kappa(\widehat{M}_{\text{BDDC, VB}}^{-1} K_g) \leq \frac{\widetilde{C}}{\tilde{c}} \Phi(H,h) \leq C_{\text{VB}} \Phi(H,h); \tag{5.12}$$

see also [9, Theorem 3].

5.1. The GM (Global Matrix) interpolation. Good constants $\tilde{c}, \widetilde{C}$ in Assumption 2 or, respectively, $\hat{c}, \widehat{C}$ in Assumption 3 are important for a small condition number and therefore a fast convergence of the approximate BDDC method. It is well known that for scalability of multigrid methods, the preconditioner should preserve nullspace or near-nullspace vectors of the operator. Therefore, the AMG method should preserve the nullspace of the operator on all levels, and these nullspace vectors have to be in the range of the AMG interpolation. While classical AMG guarantees this property only for constant vectors, the global matrix approach (GM), introduced in [3], allows the user to specify certain near-nullspace vectors, which are interpolated exactly from the coarsest to the finest level; details on the method and its scalability for linear elasticity problems can be found in [2, 3]. Since we are interested in linear elasticity problems, we choose the rotational rigid body modes for the exact interpolation. All translations of the body are already interpolated exactly in classical AMG approaches for systems of PDEs since they use classical interpolation applied component-by-component. In $\widehat{M}_{\text{BDDC, AMG}}^{-1}$, AMG is applied to $\widetilde{S}_{\Pi\Pi}$, and thus we need the three rotations in the space $\widetilde{W}_{\Pi}$, which is the restriction of $\widetilde{W}$ to the primal constraints. Therefore, we first assemble the rotations of the subdomains Ω_i locally, extract the primal components, and finally insert them into three global vectors in $\widetilde{W}_\Pi$. In our implementation, we always use BoomerAMG from the HYPRE package [11], where a highly scalable implementation of the GM2 approach is implemented; see [2]. Let us remark that BoomerAMG provides two variants of interpolations, and GM2 is recommended for use instead of GM1. In [2] it also showed a better scalability than GM1. We will compare the use of the GM2 approach with a hybrid AMG approach for systems of PDEs. By hybrid AMG approaches, we refer to methods where the coarsening is based on the physical nodes (nodal coarsening) but the interpolation is based on the degrees of freedom. In general, a nodal coarsening approach is beneficial for the solution of systems of PDEs, and all degrees of freedom belonging to the same physical node are either all coarse or fine on each level. The latter is also mandatory for the GM2 approach. Therefore, GM2 is based on the same nodal coarsening and can also be considered as a hybrid approach.

6. Implementation and model problems.

Our parallel implementation uses C/C++ and PETSc version 3.9.2 [6]. All matrices are completely local to the computational cores. All assemblies and prolongations are performed using PETSc *VecScatter* and *VecGather* operations. A more detailed description of the parallel data structures of our implementation of the linear BDDC preconditioner can be found in [13], where different nonlinear BDDC methods are applied to hyperelasticity and elasto-plasticity problems.

Since the preconditioners for the coarse problem are the focus of this paper, we include some details on the implementation of the different variants. In general, the coarse problem $\widetilde{S}_{\Pi\Pi}$ is assembled on a subset of the available cores. The number of cores can be chosen arbitrarily and should depend on the size of the coarse problem to obtain a good performance. While BoomerAMG and BDDC themselves can be applied to $\widetilde{S}_{\Pi\Pi}$ in parallel, for exact BDDC (M_{BDDC}^{-1}), a sequential copy of $\widetilde{S}_{\Pi\Pi}$ is sent to each computational core and a sparse direct solver is applied. The coarse problem is thus solved redundantly on all cores. Alternatively, one could just create a single sequential copy on a single core.

When constructing $\widehat{M}_{\text{BDDC, VB}}^{-1}$, we always build Ψ and $\widetilde{S}_{\Pi\Pi}$ as parallel matrices on the same subset of cores and perform a parallel Galerkin product to build $\widetilde{S}_{\Pi\Pi,r} := \Psi^T \widetilde{S}_{\Pi\Pi}\Psi$. Afterwards, sequential copies of $\widetilde{S}_{\Pi\Pi}$ and $\widetilde{S}_{\Pi\Pi,r} := \Psi^T \widetilde{S}_{\Pi\Pi}\Psi$ are created in order to perform a redundant sparse factorization of $\widetilde{S}_{\Pi\Pi,r}$ and a redundant Gauss-Seidel procedure for $\widetilde{S}_{\Pi\Pi}$. When using a sequential Gauss-Seidel implementation with $\widehat{M}_{\text{BDDC, VB}}^{-1}$, for simplicity, we also create a sequential copy of Ψ, which could be avoided.

When using PETSc's parallel SOR/Gauss-Seidel with $\widehat{M}_{\text{BDDC, VB}}^{-1}$, no sequential copies of Ψ are created in our implementation. Since the parallel SOR/Gauss-Seidel in PETSc is in fact a block Jacobi preconditioner in between the local blocks associated with the different MPI ranks and an SOR/Gauss-Seidel preconditioner on the local blocks themselves, its use can obviously decrease the convergence rate of the method. As an advantage, we only have to build a sequential copy of $\widetilde{S}_{\Pi\Pi,r}$, which is much smaller compared to $\widetilde{S}_{\Pi\Pi}$. The matrices $\widetilde{S}_{\Pi\Pi}$ and Ψ are kept in a distributed fashion as described above. Let us finally remark that we can apply the Gauss-Seidel preconditioner additively as described in equation (4.1) as well as multiplicatively, which is of course more robust.

7. Numerical results.

In this paper, we restrict ourselves to homogeneous linear elasticity problems, i.e., with constant coefficients. For heterogeneous examples or different model problems, we refer to [13] for $\widehat{M}_{\text{BDDC, AMG}}^{-1}$ and to [9] for $\widehat{M}_{\text{BDDC, VB}}^{-1}$. All computations are performed on the magnitUDE supercomputer (University of Duisburg-Essen) or on JUWELS (FZ Juelich).

7.1. Three-level BDDC and BDDC with an AMG coarse preconditioner.

We first concentrate on a comparison between $\widehat{M}_{\text{BDDC, 3L}}^{-1}$ and $\widehat{M}_{\text{BDDC, AMG}}^{-1}$, which clearly have the largest parallel potential, especially due to the large coarsening ratio from the second to the coarsest level. Also $\widehat{M}^{-1}_{\mathrm{BDDC,\,3L}}$ can be easily extended to a multilevel preconditioner, while $\widehat{M}^{-1}_{\mathrm{BDDC,\,AMG}}$ already consists of several levels. The alternative $\widehat{M}^{-1}_{\mathrm{BDDC,\,VB}}$ is limited in scalability by construction since the vertex-based coarse space is always solved by a sparse direct solver in our implementation. We therefore analyze and compare $\widehat{M}^{-1}_{\mathrm{BDDC,\,VB}}$ separately in Section 7.2.

To have a theoretical baseline, we always include the exact BDDC preconditioner M^{-1}_{BDDC} in all figures. To verify the quadratic dependence of the condition number on the logarithm of H/h, which can be seen as a measure of the subdomain size, we provide Figure 7.1. There, we consider a linear elastic cube decomposed into 512 subdomains with Young modulus $E = 210\,$GPa and different Poisson ratios. As a coarse space we enforce continuity of the values at all subdomain vertices and in all edge averages. With a Poisson ratio of 0.3 (Figure 7.1 (top)), all methods show the expected logarithmic dependency of the condition number on the subdomain size. All in all, $\widehat{M}^{-1}_{\mathrm{BDDC,\,AMG}}$ has slightly higher condition numbers than the competing methods. For $\widehat{M}^{-1}_{\mathrm{BDDC,\,AMG}}$ it is useful to include the GM approach. For $\widehat{M}^{-1}_{\mathrm{BDDC,\,3L}}$, both tested setups, i.e., 8 or 64 subdomains per subregion, have nearly the same condition number. Choosing a larger Poisson ratio of 0.49 (Figure 7.1 (bottom)), $\widehat{M}^{-1}_{\mathrm{BDDC,\,AMG}}$ has higher condition numbers especially for small subdomain sizes. For larger subdomain sizes and using GM, $\widehat{M}^{-1}_{\mathrm{BDDC,\,AMG}}$ is again competitive with the three-level BDDC. Let us remark that $\widehat{M}^{-1}_{\mathrm{BDDC,\,AMG}}$ for this example shows the logarithmic dependency of the condition number only for H/h larger than 16, and the condition numbers are larger. Let us remark that we always use a highly scalable AMG setup, i.e., aggressive HMIS coarsening, $ext + i$ long range interpolation, nodal coarsening, a threshold of 0.3, and a maximum of three entries per row in the AMG interpolation matrices. Less aggressive strategies might show lower condition numbers, but we explicitly choose the parameters to obtain good parallel scalability; see [2].

For the same setup with a Poisson ratio of 0.3 but fixed $H/h = 24$, we perform a weak scaling study in Figure 7.2 up to 4096 cores. Considering the number of CG iterations until convergence (Figure 7.2 (top)), the GM approach is necessary in $\widehat{M}^{-1}_{\mathrm{BDDC,\,AMG}}$ to obtain results of similar quality as $\widehat{M}^{-1}_{\mathrm{BDDC,\,3L}}$. The same can be observed considering the time to solution; see Figure 7.2 (bottom). The time to solution is always the complete runtime measured from the program start to finish. This especially includes the assembly of the linear system, the setup of the preconditioner, and the iteration/solution. For detailed timings we refer to Figure 7.3, where the assembly of the stiffness matrix, the BDDC setup, and the CG iteration are shown separately for the largest experiment from Figure 7.2. Of course, the exact BDDC preconditioner does not scale due to the sequential coarse solve.

7.2. Vertex-based BDDC.

We have implemented an "economic" variant of the edge-based coarse space [9, Section 6], i.e., three translational degrees of freedom on edges are reduced to three translational degrees of freedom of an adjacent vertex. We provide a weak scaling test up to 5 832 cores for $\widehat{M}^{-1}_{\mathrm{BDDC,\,VB}}$ for a similar model problem, i.e., linear elasticity with a Poisson ratio of 0.3 and a Young modulus of $210\,$GPa. In Figure 7.4, we provide CG iterations and time to solution for exact BDDC and $\widehat{M}^{-1}_{\mathrm{BDDC,\,AMG}}$ using GM. We also provide more detailed timings in Figure 7.5, where the assembly of the stiffness matrix, the BDDC setup, and the CG iteration are shown for the largest experiment from Figure 7.4 separately. For $\widehat{M}^{-1}_{\mathrm{BDDC,\,VB}}$, a multiplicative combination of Gauss-Seidel applied to $\widetilde{S}_{\Pi\Pi}$ and the direct solve of the vertex-based coarse problem is always a better choice than an additive variant. The parallel Gauss-Seidel method, which—as implemented in PETSc—is in fact a block Jacobi with SOR/Gauss-Seidel blocks, always results in more CG iterations but faster runtimes. With respect to parallel scalability, the best variant of $\widehat{M}^{-1}_{\text{BDDC, VB}}$ is competitive with $\widehat{M}^{-1}_{\text{BDDC, AMG}}$ at least up to the moderate core count of 5 832. For an increasing number of cores, we expect $\widehat{M}^{-1}_{\text{BDDC, AMG}}$ to outperform $\widehat{M}^{-1}_{\text{BDDC, VB}}$ due to its inherent multilevel structure. Here, a three-level extension of $\widehat{M}^{-1}_{\text{BDDC, VB}}$ would be needed. Instead, $\widehat{M}^{-1}_{\text{BDDC, VB}}$ could be used to precondition the coarsest level of a three-level BDDC method.

8. Conclusion.

We have presented different approaches to approximate the coarse solve in BDDC and compared them with respect to theory and parallel scalability for the first time. If an appropriate AMG approach is available, e.g., the GM approach in the case of linear elasticity problems, then $\widehat{M}^{-1}_{\text{BDDC, AMG}}$ and $\widehat{M}^{-1}_{\text{BDDC, 3L}}$ show a very similar behavior and both variants can be recommended. Up to a moderate number of computing cores, also $\widehat{M}^{-1}_{\text{BDDC, VB}}$ can be an adequate alternative. An advantage of $\widehat{M}^{-1}_{\text{BDDC, VB}}$ is the fact that neither a further decomposition into subregions is necessary nor an appropriate AMG method has to be chosen. On the other hand, the parallel potential of $\widehat{M}^{-1}_{\text{BDDC, VB}}$ as a two-level method is limited. For a large number of subdomains, a three-level extension of $\widehat{M}^{-1}_{\text{BDDC, VB}}$ would be necessary but is not available yet. However, instead, $\widehat{M}^{-1}_{\text{BDDC, VB}}$ could be used on the coarsest level of a three-level BDDC method.

Acknowledgments. The authors gratefully acknowledge the computing time granted by the Center for Computational Sciences and Simulation (CCSS) of the Universität of Duisburg-Essen and provided on the supercomputer magnitUDE (DFG grants INST 20876/209-1 FUGG, INST 20876/243-1 FUGG) at the Zentrum für Informations- und Mediendienste (ZIM).

The authors also gratefully acknowledge the Gauss Centre for Supercomputing e.V. (www.gauss-centre.eu) for funding this project by providing computing time on the GCS Supercomputer JUWELS at Jülich Supercomputing Centre (JSC).

REFERENCES

[1] S. Badia, A. F. Martín, and J. Principe, *Multilevel balancing domain decomposition at extreme scales*, SIAM J. Sci. Comput., 38 (2016), pp. C22–C52.

[2] A. H. Baker, A. Klawonn, T. Kolev, M. Lanser, O. Rheinbach, and U. M. Yang, *Scalability of classical algebraic multigrid for elasticity to half a million parallel tasks*, in Software for Exascale Computing—SPPEXA 2013–2015, H.-J. Bungartz, P. Neumann, and W. E. Nagel, eds., vol. 113 of Lect. Notes Comput. Sci. Eng., Springer, Cham, 2016, pp. 113–140.

[3] A. H. Baker, T. V. Kolev, and U. M. Yang, *Improving algebraic multigrid interpolation operators for linear elasticity problems*, Numer. Linear Algebra Appl., 17 (2010), pp. 495–517.

[4] S. Balay, S. Abhyankar, M. F. Adams, J. Brown, P. Brune, K. Buschelman, L. Dalcin, A. Dener, V. Eijkhout, W. D. Gropp, D. Kaushik, M. G. Knepley, D. A. May, L. C. McInnes, R. T. Mills, T. Munson, K. Rupp, P. Sanan, B. F. Smith, S. Zampini, H. Zhang, and H. Zhang, *PETSc users manual*, Tech. Rep. ANL-95/11-Revision 3.10, Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, 2018.

[5] ———, *PETSc Web page*, 2018. http://www.mcs.anl.gov/petsc

[6] S. Balay, W. D. Gropp, L. C. McInnes, and B. F. Smith, *Efficient Management of Parallelism in Object-Oriented Numerical Software Libraries*, in Modern Software Tools in Scientific Computing, E. Arge, A. M. Bruaset, and H. P. Langtangen, eds., Birkhäuser, Boston, 1997, pp. 163–202.

[7] L. Beirão da Veiga, L. F. Pavarino, S. Scacchi, O. B. Widlund, and S. Zampini, *Parallel sum primal spaces for isogeometric deluxe BDDC preconditioners*, in Domain Decomposition Methods in Science and Engineering XXIII, C. -O. Lee, X.-C. Cai, D. E. Keyes, H. H. Kim, A. Klawonn, E.-J. Park, and O. B. Widlund, eds., vol. 116 of Lect. Notes Comput. Sci. Eng., Springer, Cham, 2017, pp. 17–29.

[8] C. R. Dohrmann, *An approximate BDDC preconditioner*, Numer. Linear Algebra Appl., 14 (2007), pp. 149–168.

[9] C. R. Dohrmann, K. H. Pierson, and O. B. Widlund, *Vertex-based preconditioners for the coarse problems of BDDC*, SIAM J. Sci. Comput., 41 (2019), pp. A3021–A3044.

[10] C. FARHAT, M. LESOINNE, AND K. PIERSON, *A scalable dual-primal domain decomposition method*, Numer. Linear Algebra Appl., 7 (2000), pp. 687–714.

[11] V. E. HENSON AND U. M. YANG, *BoomerAMG: a parallel algebraic multigrid solver and preconditioner*, Appl. Numer. Math., 41 (2002), pp. 155–177.

[12] A. KLAWONN, M. LANSER, AND O. RHEINBACH, *Toward extremely scalable nonlinear domain decomposition methods for elliptic partial differential equations*, SIAM J. Sci. Comput., 37 (2015), pp. C667–C696.

[13] ———, *Nonlinear BDDC methods with approximate solvers*, Electron. Trans. Numer. Anal., 49 (2018), pp. 244–273.
http://etna.ricam.oeaw.ac.at/vol.49.2018/pp244-273.dir/pp244-273.pdf

[14] A. KLAWONN, M. LANSER, O. RHEINBACH, AND M. URAN, *Nonlinear FETI-DP and BDDC methods: a unified framework and parallel results*, SIAM J. Sci. Comput., 39 (2017), pp. C417–C451.

[15] A. KLAWONN AND O. RHEINBACH, *Inexact FETI-DP methods*, Internat. J. Numer. Methods Engrg., 69 (2007), pp. 284–307.

[16] ———, *Robust FETI-DP methods for heterogeneous three dimensional elasticity problems*, Comput. Methods Appl. Mech. Engrg., 196 (2007), pp. 1400–1414.

[17] ———, *Highly scalable parallel domain decomposition methods with an application to biomechanics*, ZAMM Z. Angew. Math. Mech., 90 (2010), pp. 5–32.

[18] A. KLAWONN AND O. B. WIDLUND, *Dual-primal FETI methods for linear elasticity*, Comm. Pure Appl. Math., 59 (2006), pp. 1523–1572.

[19] J. LI AND O. B. WIDLUND, *On the use of inexact subdomain solvers for BDDC algorithms*, Comput. Methods Appl. Mech. Engrg., 196 (2007), pp. 1415–1428.

[20] J. MANDEL, B. SOUSEDÍK, AND C. R. DOHRMANN, *Multispace and multilevel BDDC*, Computing, 83 (2008), pp. 55–85.

[21] O. RHEINBACH, *Parallel iterative substructuring in structural mechanics*, Arch. Comput. Methods Eng., 16 (2009), pp. 425–463.

[22] B. SOUSEDÍK, J. ŠÍSTEK, AND J. MANDEL, *Adaptive-multilevel BDDC and its parallel implementation*, Computing, 95 (2013), pp. 1087–1119.

[23] A. TOSELLI AND O. WIDLUND, *Domain Decomposition Methods—Algorithms and Theory*, Springer, Berlin, 2005.

[24] X. TU, *Three-level BDDC in three dimensions*, SIAM J. Sci. Comput., 29 (2007), pp. 1759–1780.

[25] ———, *Three-level BDDC in two dimensions*, Internat. J. Numer. Methods Engrg., 69 (2007), pp. 33–59.

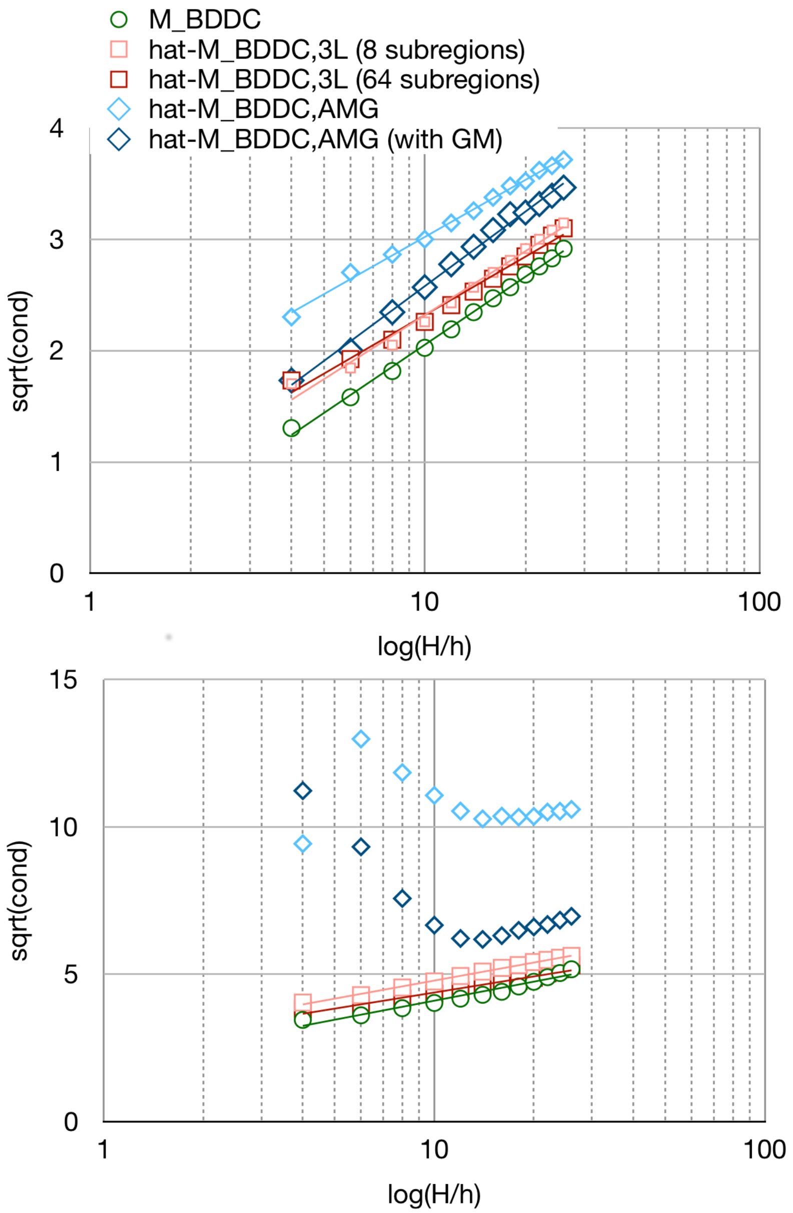

FIG. 7.1. *Homogeneous linear elastic cube decomposed into 512 subdomains with $H/h = 4, 6, ..., 26$. Top: $E = 210.0$ and $\nu = 0.3$; Bottom: $E = 210.0$ and $\nu = 0.49$. We vary the number of subdomains per subregion in $\widehat{M}^{-1}_{\text{BDDC, 3L}}$, and we compare nodal AMG and AMG-GM in $\widehat{M}^{-1}_{\text{BDDC, AMG}}$. Computed on the magnitUDE supercomputer.*

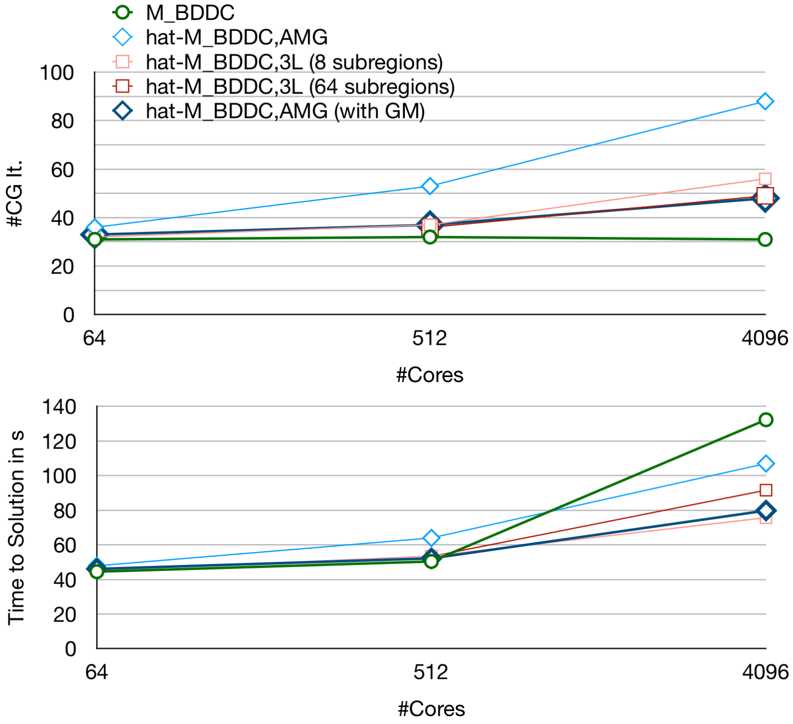

FIG. 7.2. *Comparison of M_{BDDC}^{-1}, $\widehat{M}_{\mathrm{BDDC,3L}}^{-1}$ with 8 or 64 subregions and $\widehat{M}_{\mathrm{BDDC,AMG}}^{-1}$ with and without GM. Using vertex and edge constraints. Homogeneous linear elastic cube decomposed into 64,512, and* 4 096 *subdomains with $H/h = 24$. Top: Number of CG iterations; Bottom: Total time to solution including assembly of stiffness matrices, setup of the preconditioner, and solution phase. See also Figure 7.3 for the detailed setup and solve times for the largest experiment. Computed on JUWELS.*

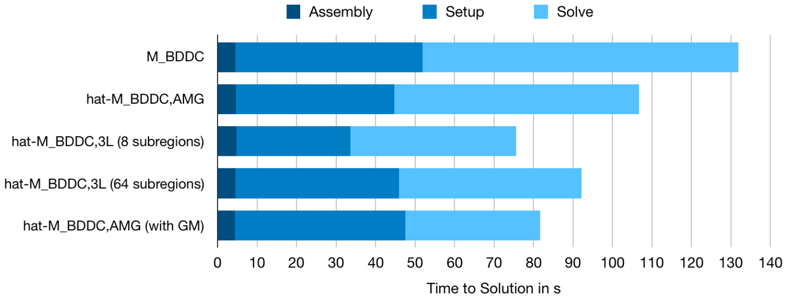

FIG. 7.3. *Comparison of BDDC setup and solve times of M_{BDDC}^{-1}, $\widehat{M}_{\mathrm{BDDC,3L}}^{-1}$ with 8 or 64 subregions and $\widehat{M}_{\mathrm{BDDC,AMG}}^{-1}$ with and without GM. Using vertex and edge constraints. Homogeneous linear elastic cube decomposed into* 4 096 *subdomains with $H/h = 24$. Corresponding weak scaling experiments can be found in Figure 7.2. Computed on JUWELS.*

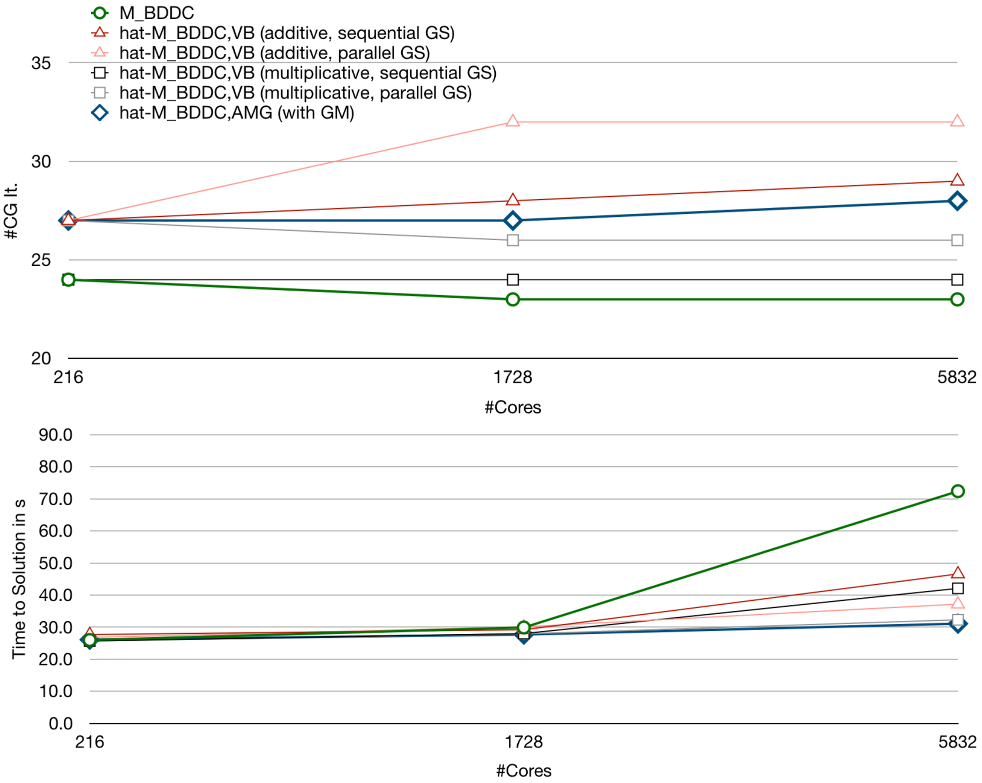

FIG. 7.4. *Comparison of M_{BDDC}^{-1}, $\widehat{M}_{\mathrm{BDDC,VB}}^{-1}$ using additive/multiplicative sequential/parallel Gauss-Seidel and $\widehat{M}_{\mathrm{BDDC,AMG}}^{-1}$ with GM. Using only edge constraints. Homogeneous linear elastic cube with $H/h = 22$. Top: Number of CG iterations; Bottom: Total time to solution including assembly of stiffness matrices, setup of the preconditioner, and solution phase. See also Figure 7.5 for the detailed setup and solve times for the largest experiment. Computed on the magnitUDE supercomputer.*

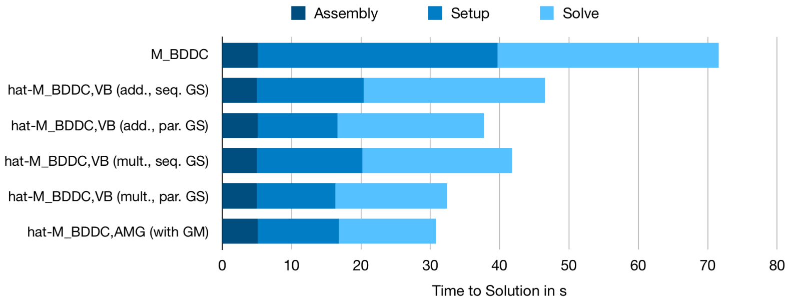

FIG. 7.5. *Comparison of BDDC setup and solve times of M_{BDDC}^{-1}, $\widehat{M}_{\mathrm{BDDC,VB}}^{-1}$ with different Gauß-Seidel setups and $\widehat{M}_{\mathrm{BDDC,AMG}}^{-1}$ with GM. Using edge translations as constraints. Homogeneous linear elastic cube decomposed into 5 832 subdomains with $H/h = 22$. Corresponding weak scaling experiments can be found in Figure 7.4. Computed on magnitUDE supercomputer.*