BDDC DELUXE ALGORITHMS FOR TWO-DIMENSIONAL H($curl$) ISOGEOMETRIC ANALYSIS

OLOF B. WIDLUND *, SIMONE SCACCHI †, AND LUCA F. PAVARINO ‡

January 2, 2022

Abstract. Isogeometric analysis (IGA) has been introduced as an alternative to finite elements to simplify the integration of computer-aided design and the discretization of variational problems of continuum mechanics. In contrast to finite elements, the basis functions of isogeometric analysis, B–splines and nonuniform rational basis splines (NURBS), are typically not nodal which makes the interface between subdomains fat with several layers of knots and which creates new issues in the design and analysis of iterative solvers based on domain decomposition methods. The resulting systems of algebraic equations also tend to be much more ill-conditioned than those derived from finite elements. The deluxe variant of the BDDC (balancing domain decomposition by constraints) algorithms have proven successful and in this study, they are applied to two-dimensional problems formulated in $\mathbf{H}(curl)$. Numerical results show very good performance even for high degrees of the underlying B–splines and fast convergence has been found for $p \leq 7$ using no more than $2(p^2+p-1)$ global (primal) variables per subdomain where p is the degree of the B–splines. All our results are focused on the most difficult case with the maximum degree of continuity of the B–splines.

Key words. $\mathbf{H}(curl)$, two dimensions, isogeometric analysis, BDDC deluxe algorithms

AMS subject classifications. 65F08, 65F10, 65N30, 65N55

1. Introduction.

Isogeometric analysis has been developed as an alternative to finite elements for a number of applications; see, e.g., [11, 2]. In this paper, we will focus on problems formulated in $\mathbf{H}(curl)$ in two space dimension, (2D); see [7, 8], as well as [2]. In the current study, we will use results for lower order finite element problems given in [19] and we could also use a result from [24], which provides an edge lemma which is crucial in all work of this kind. We also note an early paper, [23], on FETI algorithms for two-dimensional problems formulated in $\mathbf{H}(curl)$, and that the theory for FETI and BDDC algorithms is very closely related. BDDC algorithms for finite elements discretizations of $\mathbf{H}(curl)$ were constructed and analyzed in [14, 15, 10, 9, 29]; see also [13, 17] for earlier work on $\mathbf{H}(curl)$ for iterative substructuring algorithms and [1, 16, 18] for multilevel and multigrid algorithms.

Here we extend the BDDC algorithms to isogeometric discretizations of 2D $\mathbf{H}(curl)$. The earliest paper on BDDC algorithms for isogeometric analysis appears to be [3]. Our current study is a continuation of work on domain decomposition algorithms of the BDDC deluxe family of iterative substructuring algorithms for isogeometric analysis approximations of scalar elliptic problems in two dimensions, [4, 5], compressible linear elasticity, [20], and, most recently, almost incompressible elasticity, [27], approximated by a mixed isogeometric analysis algorithm, as developed in [6], using continuous pressure fields. The success with these earlier studies has inspired us to undertake our current work.

The linear systems of equations that arise in isogeometric analysis often are extremely ill-conditioned, in particular for splines and NURBS of high order, and we have found that use of deluxe variants outperforms the earlier BDDC variants; see e.g. [4]. Algorithms of this family also

*Courant Institute of Mathematical Sciences, 251 Mercer Street, New York, NY 10012, USA. widlund@cims.nyu.edu.

†Dipartimento di Matematica, Università degli Studi di Milano, Via Saldini 50, 20133 Milano, Italy. simone.scacchi@unimi.it. This work was supported by grants of M.I.U.R. (PRIN 201289A4LX_002) and of Istituto Nazionale di Alta Matematica (INDAM-GNCS).

‡Dipartimento di Matematica, Università degli Studi di Pavia, Via Ferrata 5, 27100 Pavia, Italy. luca.pavarino@unipv.it. This work was partially supported by the European Research Council through the FP7 Ideas Consolidator Grant *HIGEOM* n. 616563, by the Italian Ministry of Education, University and Research (MIUR) through the "Dipartimenti di Eccellenza Program 2018-22 - Dept. of Mathematics, University of Pavia", and by the Istituto Nazionale di Alta Matematica (INdAM - GNCS), Italy.

lend themselves straightforwardly to the use of adaptive algorithms for designing the global, primal, part of these preconditioners; see e.g. [5]. Thus, not only have we developed algorithms that work very well even for isogeometric elements based on high order B-splines and NURBS but we have also used adaptive algorithms for finding the minimal number of primal constraints which guarantees rapid convergence of our preconditioned conjugate gradient (PCG) algorithms. For a recent survey paper on BDDC algorithms, see [26].

The isogeometric analysis methods considered in this paper are based on B-splines and NURBS, see, e.g., [5, 4]. In the simplest cases of lowest order and with minimal interelement continuity, these methods reduce to Nédélec elements of the first kind on a mesh which is the image of a very regular square mesh; a similar case has been analyzed in great detail for $H(div)$ problems in three dimensions, approximated by the lowest order Raviart–Thomas elements, in [19]. We note that for two dimensions a problem in $\mathbf{H}(curl)$ using Nédélec elements can be transformed into a problem in $H(div)$ using Raviart–Thomas elements by a simple rotation of the coordinate system. In contrast, $\mathbf{H}(\mathbf{curl})-$problems in three dimensions pose very significant new challenges compared to those for $H(div)$.

Our results immediately extend to 2D $H(div)$ problems and it might be possible to extend them to 3D $H(div)$ guided by the results in [19]. However, we note that while that paper clearly spells out that the theory for BDDC deluxe algorithms, in the finite element case, depends entirely on establishing a face lemma, additional tools associated with the subdomain edges and vertices would be required in a study of isogeometric analysis.

The principal complications in the design and analysis of iterative methods for isogeometric analysis stem from the fact that the B–spline basis functions generally lack a nodal basis and as a consequence the interface between the subdomains, into which the domain of the original problem is being subdivided, is *fat*, i.e., consists of several layers of knots; see, e.g., Fig. 3.1 and Fig. 3.2 or [20, Fig. 1]. This *fat interface* will be divided into *fat edges* and *fat vertices* in Subsection 3.2.

The remainder of this paper is organized as follows: Our continuos model problem is introduced in Section 2. The isogeometric analysis algorithms for our 2D $\mathbf{H}(curl)-$problem is introduced in Section 3 where we also introduce a lowest order Nédélec problem, which is important in our development. All the problems encountered will have fat interfaces. Section 4 introduces our adaptive BDDC deluxe algorithms. In Section 5, we formulate a conjecture which highlights that the number of global, primal variables associated with a fat edge, required to assure scalability and fast convergence of our preconditioned conjugate gradient algorithms, will not exceed the number of thin edges - layers of knots - of the fat edge. Finally, in Section 6, we report on a selection of numerical experiments, which supports our conjecture.

2. The continuous model problem. The bounded Lipschitz domain of our problem, $\Omega \subset \mathbb{R}^2$, is partitioned into non-overlapping subdomains Ω_i. Each of these subdomains is the image of a square reference subdomain $\widehat{\Omega}_i$ under a smooth mapping $\boldsymbol{F}$, with a smooth inverse, constructed using the same space of NURBS used to approximate the solution. The bilinear form which provides the variational formulation of our problem is given by

$$a_\Omega(\boldsymbol{u},\boldsymbol{v}) := \sum_i a_{\Omega_i}(\boldsymbol{u},\boldsymbol{v}) := \sum_i \int_{\Omega_i} a_i \; curl(\boldsymbol{u}) \; curl(\boldsymbol{v}) + \sum_i \int_{\Omega_i} b_i \; \boldsymbol{u}\cdot\boldsymbol{v}, \tag{2.1}$$

where the terms in the sums represent the contributions from the subdomains Ω_i; we will refer to $a_{\Omega_i}(\boldsymbol{u},\boldsymbol{u})$ as the energy contributed by Ω_i. We will assume, for simplicity, that the coefficients a_i and b_i are strictly positive constants in each subdomain; they could also be replaced by symmetric, positive definite matrices.

The variational form of our continuous model problem reads: given $\boldsymbol{f} \in \boldsymbol{L}^2(\Omega) := (L^2(\Omega))^2$,

find $\boldsymbol{u} \in \mathbf{H}_0(curl;\Omega)$ such that

$$a_\Omega(\boldsymbol{u},\boldsymbol{v}) = \int_\Omega \boldsymbol{f}\cdot\boldsymbol{v}, \quad \forall \boldsymbol{v} \in \mathbf{H}_0(curl;\Omega), \tag{2.2}$$

where $\mathbf{H}_0(curl;\Omega)$ consists of vector fields in $\boldsymbol{L}^2(\Omega)$ with $curl(\boldsymbol{v}) in L^2(\Omega)$ and with a vanishing tangential trace on $\partial\Omega$. More general boundary conditions with vanishing tangential trace, on only part of $\partial\Omega$, could be considered as well.

3. The discrete problem. Following [7], we now introduce our discrete problem as well as a related problem based on the lowest order Nédélec elements of the first kind which will be used in developing our understanding; we will need to consider Nedédélec element problems with fat interfaces.

3.1. Isogeometric discretization. To simplify the writing, we will assume that Ω consists of only one patch and that the boundary condition is given by a zero tangential component along the entire boundary of the patch; our results are equally valid for multi-patch cases. The introduction of the discrete space of [7] begins by introducing the B–splines, $B_i^p(x)$ and $B_j^q(y)$, which are familiar from work on problems formulated in H^1, and the tensor-product B–spline basis functions $B_{ij}^{pq}(x,y) := B_i^p(x)B_j^q(y)$ on the reference domain $\widehat{\Omega}$. Two spaces spanned by these basis functions are denoted, following [7], by

$$\mathcal{S}^p_{\alpha_1} := span\{B_i^p\} \qquad \text{and} \qquad \mathcal{S}^{pq}_{\alpha_1\alpha_2} := \mathcal{S}^p_{\alpha_1}\otimes\mathcal{S}^q_{\alpha_2} = span\{B_i^pB_j^q\}.$$

The components of the vectors α_1 and α_2 encode the number of continuous derivatives at the knots of the partitioning of the intervals of the B–splines. As in previous studies, since the mapping $\boldsymbol{F}$, which defines Ω as an image of the reference domain $\widehat{\Omega}$ and its inverse are very regular, our preconditioners designed using B–splines can easily be extended to cases where the B–splines are replaced by NURBS. Therefore, we can focus our discussion on problems associated with the reference domain $\widehat{\Omega}$, which we assume to be a unit square.

As shown in [7], using a simple dimensional argument, the gradient operator maps $\mathcal{S}^{pq}_{\alpha_1\alpha_2}$ onto the curl-free subspace of

$$(\mathcal{S}^{p-1}_{\alpha_1-1}\otimes\mathcal{S}^q_{\alpha_2}, \mathcal{S}^p_{\alpha_1}\otimes\mathcal{S}^{q-1}_{\alpha_2-1}).$$

We will use all elements of this space as our approximations of $\mathbf{H}(curl;\Omega)$. Here $\alpha_i - 1, i = 1,2$ is obtained from α_i by subtracting 1 from each of its components except if a component equals -1; such a component will not be changed. Applying the curl operator to the elements in that space gives us elements in the space $\mathcal{S}^{p-1}_{\alpha_1-1}\otimes\mathcal{S}^{p-1}_{\alpha_2-1}$. Thus, if in a reference subdomain $\widehat{\Omega}_i$,

$$\boldsymbol{u} := (\sum_{i=m_1+1}^{m_4}\sum_{j=n_1}^{n_4}\gamma_{ij}^{(1)}B_i^{p-1}(x)B_j^q(y), \sum_{i=m_1}^{m_4}\sum_{j=n_1+1}^{n_4}\gamma_{ij}^{(2)}B_i^p(x)B_j^{q-1}(y)), \tag{3.1}$$

we find, assuming that we have a uniform mesh, that for any subdomain in the interior of the patch,

$$curl(\boldsymbol{u}) = \frac{p}{h}\sum_{i=m_1+1}^{m_4}\sum_{j=n_1+1}^{n_4}(-\gamma_{ij}^{(1)} + \gamma_{i,j-1}^{(1)} + \gamma_{ij}^{(2)} - \gamma_{i-1,j}^{(2)})B_i^{p-1}(x)B_j^{q-1}(y), \tag{3.2}$$

by using [3, (2.7)], a formula for the derivative of $B_i^p(x)$:

$$\frac{d}{dx}B_i^p(x) = \frac{p}{h}(B_i^{p-1}(x) - B_{i+1}^{p-1}(x)).$$

When considering boundary value problems which define the minimal energy - the discrete harmonic - extensions of values given on the interface, that is central to developing our method, the tangential component of $\boldsymbol{u}$, as in (3.1), will enter. We note that for any domain Ω, defined by a non-trivial map $\boldsymbol{F} : \widehat{\Omega} \rightarrow \Omega$, both the horizontal and vertical components will play a role given that the relevant vector-valued space, which is central in developing our method, is defined by the space V_h given by [7, (19)] which involves the Jacobian $\boldsymbol{J}_F$ of $\boldsymbol{F}$:

$$V_h := \{\boldsymbol{u} : \boldsymbol{u} \circ \boldsymbol{F} := (\boldsymbol{J}_F)^{-T}\hat{\boldsymbol{u}}, \quad \hat{\boldsymbol{u}} \in (\mathcal{S}^{p-1}_{\alpha_1-1} \otimes \mathcal{S}^{q}_{\alpha_2}, \mathcal{S}^{p}_{\alpha_1} \otimes \mathcal{S}^{q-1}_{\alpha_2-1})\}.$$

This Jacobian will typically not be diagonal. It is important to note that the mapping defining this space is curl-preserving and, as a consequence, the tangential component of any element will be preserved.

By imposing the boundary conditions, we then have the discrete spaces

$$\widehat{V}_{h,0} := (\mathcal{S}^{p-1,q}_{\alpha_1-1,\alpha_2} \times \mathcal{S}^{p,q-1}_{\alpha_1,\alpha_2-1}) \cap \mathbf{H}_0(curl;\widehat{\Omega}),$$

$$V_{h,0} := V_h \cap \mathbf{H}_0(curl;\Omega) := \{\boldsymbol{u} : \boldsymbol{u} \circ \boldsymbol{F} = \boldsymbol{J}_F^{-T}\widehat{\boldsymbol{u}}, \quad \widehat{\boldsymbol{u}} \in \widehat{V}_{h,0}\},$$

and the discrete problem reads: given $\boldsymbol{f} \in \mathbf{L}^2(\Omega)$, find $\boldsymbol{u}_h \in V_{h,0}$ such that

$$a(\boldsymbol{u}_h, \boldsymbol{v}_h) = \int_\Omega \boldsymbol{f} \cdot \boldsymbol{v}_h, \quad \forall \boldsymbol{v}_h \in V_{h,0}.$$

3.2. The fat interface. Due to their high regularity, B-spline basis functions have a large support. As a consequence, more than one layer of index pairs (i, j) will be associated with basis functions which do not vanish on all of the interface of any subdomain. As in our previous papers [3, 4], we will refer to this index set as the *fat interface*. The thickness of the fat interface will depend on p and q, the degrees of the spline spaces, and on the degree of continuity of the splines. We will now assume that $q = p$, i.e., we have the same degree for the two coordinates x and y and that the number of continuous derivatives, k, is the same and maximal, i.e., $k = p - 1$, at all the interior knots of $\widehat{\Omega}$. With no interior multiple knots, as we have just assumed, the support of B^p_i is the union of $p + 1$ closed intervals between consecutive knots numbered $i, \ldots, i + p + 1$. The lower part of the fat interface parallel to the $x-$axis, is determined by the values of (i, j) of the tensor product splines with supports which are intersected, in their interiors, by the lower part of the boundary of the subdomain $\widehat{\Omega_i}$. This defines equivalence classes of index pairs (i, j) associated with the fat interface and the interior of the subdomain. The fat interface is further divided into equivalence classes associated with the edges and the vertices of the subdomain, respectively.

A fat vertex set consists of the index pairs with basis functions that do not vanish at a particular subdomain vertex shared by four subdomains. A fat edge set consists of the index pairs with basis functions that vanish at all subdomain vertices but have supports which intersect, in their interiors, one of the subdomain edges. The remaining pairs of index pairs for a subdomain belong to the interior set of the subdomain and their associated variables can therefore be eliminated locally. As for the index pairs associated with the boundary of the patch, their basis functions have supports which intersect either one subdomain or two. For the first case these index pairs are assigned to the interior set while in the second case they are associated with a fat edge.

We can examine formula (3.1) for $p = 3$ and $k = 2$ to determine the sets of knots belonging to a fat vertex and a horizontal fat edge. Fig. 3.1 depicts the fat vertex of the lower left vertex of $\widehat{\Omega}_i$ and lists its pairs of indices for the case where $(i^\star, j^\star)$ marks the index pair of that subdomain vertex. The figure also introduces the parameters, borrowed from (3.1), for the Nédélec space which is introduced later in this section. We note that the squares of this figure are subsets of four different subdomains, potentially, with different material coefficients.

Turning to the lower, horizontal, fat edge, we find that it consists of three relevant layers of basis functions for the first component of $\boldsymbol{u}$ and two for the second. We call each such layer a *thin edge*. See Figure 3.2, which also provides the set of index pairs of this fat edge. The elements of the top half of this fat edge belong to $\widehat{\Omega}_i$ and the rest to a subdomain, $\widehat{\Omega}_j$, just below.

Once these sets of pairs of indices have been determined, we can easily determine subspaces associated with any fat vertex, fat edge, and the interior of any subdomain by simply making all coefficients zero except for those of the relevant pairs of indices.

We remark thatthe discussion based on Figures 3.1 and 3.2 directly extend to any odd degree p, for uniform meshes, and away from the boundary of the patch. The same concepts can easily be extended to even degree and more general meshes.

For any choice of p, we will have interfaces with $2p-1$ layers of thin edges for $k=p-1$. We can also develop algorithms for smaller values of k for which the number of layers will be smaller and, in our experience, will provide faster convergence. In fact, given our experience with other problem classes, we expect that convergence rates, using smaller primal spaces than for $k=p-1$, will be just as satisfactory for cases when $k<p-1$.

Using formula (3.2) and temporarily ignoring the material coefficients, we find that $\|curl(\boldsymbol{u})\|^2_{L^2}$ will involve elements

$$(\frac{p^2}{h^2})\int B_i^{p-1}(x)B_j^{p-1}(y)B_k^{p-1}(x)B_\ell^{p-1}(y)dxdy,$$

of a matrix with up to 25 non–zero elements per row for the case of $p=3$. Given the fact that the B-splines are linearly independent, the resulting quadratic form can be bounded from above and below by positive constants times

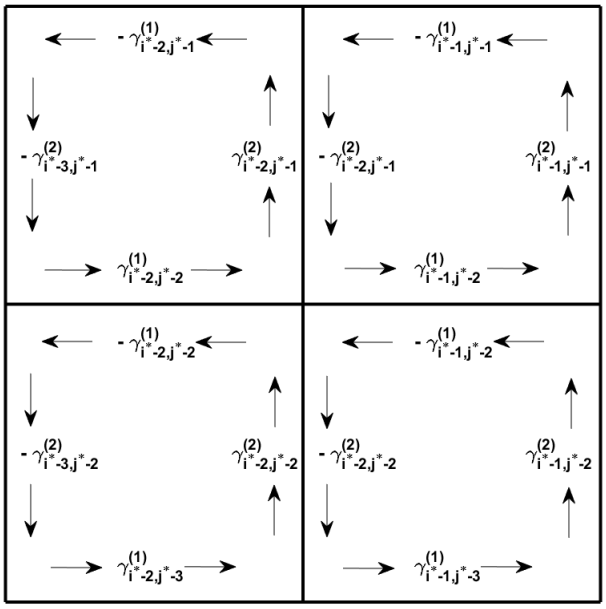
$$(\frac{p^2}{h^2})\sum_i\sum_j(-\gamma_{ij}^{(1)}+\gamma_{i,j-1}^{(1)}+\gamma_{ij}^{(2)}-\gamma_{i-1,j}^{(2)})^2. \tag{3.3}$$

These constants are independent of h, since they result from a local argument which depends on a fixed number of elements, but will depend on the parameter p.

The quadratic form (3.3), with $p=1$, also appears if we compute the contribution to the square of the L^2-norm of the curl of an element of ND_1^h, the lowest order Nédélec finite element space of the first kind, defined using the $\gamma_{ij}^{(1)}$ and $\gamma_{ij}^{(2)}$ as parameters. When doing so, we should orient the tangent vectors of the four sides of any element with the element to the left. Starting with the lowest edge of an element, we assign the parameter $\gamma_{i,j-1}^{(1)}$ to that edge and moving counter-clock-wise the values $\gamma_{i,j}^{(2)}$, $-\gamma_{i,j}^{(1)}$, and $-\gamma_{i-1,j}^{(2)}$ to the other edges; see Figures 3.1 and 3.2. The computation is then completed after introducing the simple basis functions of the Nédélec space.

We can also use the same type of arguments, which are essentially about mass matrices and linear independence of basis functions, to bound the quadratic form originating from the second term of the bilinear form of the given problem from below and above in terms of the quadratic form generated for the lowest order Nédélec elements. A similar result is established in [3, Sect. 5] using different arguments.

4. Adaptive BDDC deluxe: the choice of primal constraints using generalized eigenvalue problems. The BDDC algorithms are based on partitioning the space of interface functions into a *primal* subspace of functions, which have continuous tangential components all across the interface and belongs to the fully assembled space, and a much larger *dual* subspace which allows for multiple values of the parameters across the interface. The primal degrees of freedom (dofs) are assigned separately for each equivalence class of the interface. In our case, the equivalence classes are associated with the fat edges and fat vertices as defined in Section 3 and illustrated in Fig. 3.1 and Fig. 3.2. While the dual variables and those of the interior of the subdomains are associated



Lower left fat vertex of subdomain $\widehat{\Omega}_i$ associated with the index pair $(i^\star, j^\star)$	
first component of $\boldsymbol{u}$	second component of $\boldsymbol{u}$
$(i^\star-2, j^\star-1)$, $(i^\star-2, j^\star-2)$, $(i^\star-2, j^\star-3)$,	$(i^\star-3, j^\star-1)$, $(i^\star-3, j^\star-2)$,
$(i^\star-1, j^\star-1)$, $(i^\star-1, j^\star-2)$, $(i^\star-1, j^\star-3)$,	$(i^\star-2, j^\star-1)$, $(i^\star-2, j^\star-2)$,
	$(i^\star-1, j^\star-1)$, $(i^\star-1, j^\star-2)$

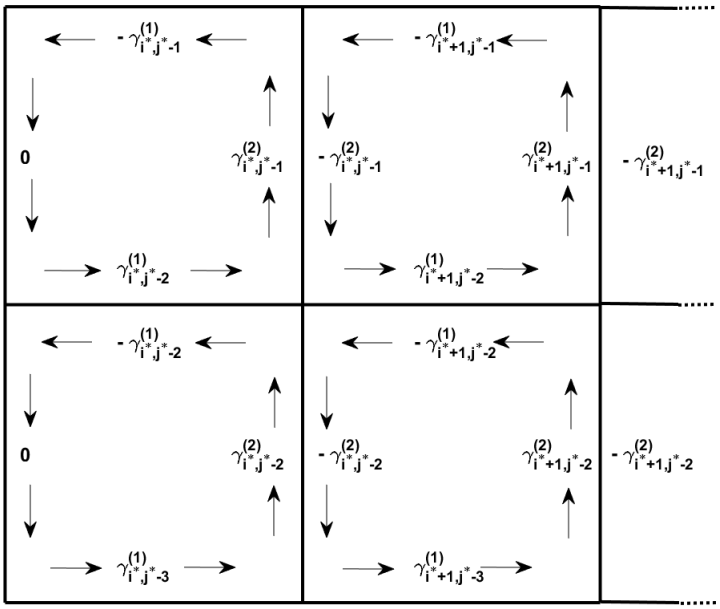
Lower right fat vertex of subdomain $\widehat{\Omega}_i$ associated with the index pair $(i^\star+N, j^\star)$	
first component of $\boldsymbol{u}$	second component of $\boldsymbol{u}$
$(i^\star+N-2, j^\star-1), (i^\star+N-2, j^\star-2), (i^\star+N-2, j^\star-3),$	$(i^\star+N-3, j^\star-1), (i^\star+N-3, j^\star-2),$
$(i^\star+N-1, j^\star-1), (i^\star+N-1, j^\star-2), (i^\star+N-1, j^\star-3)$	$(i^\star+N-2, j^\star-1), (i^\star+N-2, j^\star-2),$
	$(i^\star+N-1, j^\star-1), (i^\star+N-1, j^\star-2)$

FIG. 3.1. *Nédélec elements of lowest order corresponding to the case $p = 3, k = 2$, with arrows indicating the direction of the tangent. Lower left fat vertex, with the 12 degrees of freedom associated with the 12 pairs of indices given in the table. The center of this figure is at a vertex of $\widehat{\Omega}_i$ represented by $(i^\star, j^\star)$. The same information is also provided for the lower right fat vertex of the subdomain centered at $(i^\star + N, j^\star)$.*

with local subdomain problems, which can be handled by individual processors of a parallel computing system, the primal variables are part of a global problem. Therefore it is very important to use primal spaces that are as small as possible while obtaining robust and fast convergence of our iterations. The best way of assuring this is to select the primal variables using adaptive algorithms; see [5] for an early study of such BDDC algorithms and [19] and [21] for further details.

We now derive suitable primal constraints by using generalized eigenvalue problems for the different equivalence classes associated with the fat interface. In our two dimensional case, a subdomain edge is shared by two subdomains, whereas a subdomain vertex is shared by four.

4.1. Fat edge primal subspaces. Let us first focus on the case of an edge of the subdomain Ω_i and denote by E the set of dofs of the associated fat edge. Let Γ_i be the set of all dofs of the



Lower horizontal fat edge of subdomain $\widehat{\Omega}_i$	
first component of $\boldsymbol{u}$	second component of $\boldsymbol{u}$
$(i^\star, j^\star - 1), (i^\star, j^\star - 2), (i^\star, j^\star - 3),$	$(i^\star, j^\star - 1), (i^\star, j^\star - 2),$
$(i^\star + 1, j^\star - 1), (i^\star + 1, j^\star - 2), (i^\star + 1, j^\star - 3),$	$(i^\star + 1, j^\star - 1), (i^\star + 1, j^\star - 2),$
$(i^\star + 2, j^\star - 1), (i^\star + 2, j^\star - 2), (i^\star + 2, j^\star - 3),$	$(i^\star + 2, j^\star - 1), (i^\star + 2, j^\star - 2),$
$\dots$	$\dots$

FIG. 3.2. *Nédélec elements of lowest order corresponding to the case $p = 3, k = 2$, with arrows indicating the direction of the tangent. Lower fat edge, with the degrees of freedom associated with the pairs of indices given in the table. The left-most, lower vertex of $\widehat{\Omega}_i$ is represented by the index pair $(i^\star, j^\star)$.*

fat interface of Ω_i and let $E' := \Gamma_i \setminus E$. Then, the subdomain Schur complement $S^{(i)}$, obtained by eliminating the interior variables of the subdomain, can be written as

$$S^{(i)} = \begin{pmatrix} S^{(i)}_{E'E'} & S^{(i)}_{E'E} \\ S^{(i)T}_{E'E} & S^{(i)}_{EE} \end{pmatrix},$$

and where a Schur complement of $S^{(i)}$ is given by

$$\widetilde{S}^{(i)}_{EE} := S^{(i)}_{EE} - S^{(i)T}_{E'E} S^{(i)-1}_{E'E'} S^{(i)}_{E'E},$$

which represents the minimal energy extension of the values on the fat edge E onto the rest of Γ_i. Assuming that the fat edge E is shared by the subdomains Ω_i and Ω_j, consider now the generalized eigenvalue problem

$$\widetilde{S}^{(i)}_{EE} : \widetilde{S}^{(j)}_{EE}\phi = \lambda S^{(i)}_{EE} : S^{(j)}_{EE}\phi, \tag{4.1}$$

recalling that, for any two positive definite matrices A and B, their parallel sum is defined by

$$A : B := (A^{-1} + B^{-1})^{-1}.$$

After selecting the eigenvectors $\{v_1, v_2, \ldots, v_{N_{\Pi E}}\}$ of the eigenproblem (4.1) that are associated with the $N_{\Pi E}$ smallest eigenvalues, we perform a BDDC change of basis introducing the selected eigenvectors as elements of the primal space. This change of basis uses all the eigenvectors of the generalized eigenvalue problems; see, e.g., [5, eq. (3.16)] for the matrix form of the BDDC preconditioner.

4.2. Fat vertex primal subspaces. Consider now a vertex of the subdomain Ω_i and denote by V the set of dofs of the associated fat vertex and let $V' := \Gamma_i \setminus V$. Then the subdomain Schur complement $S^{(i)}$ can be written as

$$S^{(i)} = \begin{pmatrix} S^{(i)}_{V'V'} & S^{(i)}_{V'V} \\ S^{(i)T}_{V'V} & S^{(i)}_{VV} \end{pmatrix},$$

and

$$\widetilde{S}^{(i)}_{VV} := S^{(i)}_{VV} - S^{(i)T}_{V'V} S^{(i)-1}_{V'V'} S^{(i)}_{V'V}$$

represents the minimal energy extension of the values of the fat vertex V onto the rest of Γ_i. Assuming that the fat vertex V is shared by the four subdomains Ω_i, Ω_j, Ω_k, and Ω_ℓ, we consider the generalized eigenvalue problem

$$\widetilde{S}^{(i)}_{VV} : \widetilde{S}^{(j)}_{VV} : \widetilde{S}^{(k)}_{VV} : \widetilde{S}^{(\ell)}_{VV} \phi = \lambda S^{(i)}_{VV} : S^{(j)}_{VV} : S^{(k)}_{VV} : S^{(\ell)}_{VV} \phi, \tag{4.2}$$

where

$$A : B : C : D := (A^{-1} + B^{-1} + C^{-1} + D^{-1})^{-1}.$$

After selecting the eigenvectors $\{v_1, v_2, \ldots, v_{N_{\Pi V}}\}$ of the eigenproblem (4.2) that are associated with the $N_{\Pi V}$ smallest eigenvalues, we perform a BDDC change of basis to introduce the selected eigenvectors as elements of the primal space.

Alternative generalized eigenvalue problems have been considered for equivalence classes with more than two subdomains; see [5] for a comparative study of several alternatives for standard elliptic problems. In this study, we have selected the variant called V_{par} in [5], based on using parallel sums on both sides of the vertex eigenvalue problems (4.2), which has proven most successful in that study. We have also obtained results with the variant V_{mix} in [5], based on replacing ths parallel sum with a standard sum in the left-hand side of the generalized eigenvalue problems (4.2) and found no real difference.

The condition number of our algorithm can be estimated using the smallest eigenvalues not chosen when selecting elements of the primal space for the fat edges and the fat vertices; see, e.g., [5].

5. The fat interface and its primal variables. In attempting to develop a further understanding, we cannot rely on eigenvectors of generalized eigenvalue problems but must construct a primal space. For any primal subspace the complimentary subspace is known as the *dual subspace.* The restriction of the dual component of $\boldsymbol{w}$ to a fat edge E will be denoted by $\boldsymbol{w}^{(i)}_{E,\Delta}$. If all the fat vertex dofs are chosen to be primal, all that is then required to prove an estimate of the condition number of our BDDC deluxe algorithm would be a bound of the form

$$\boldsymbol{w}^{(i)T}_{E,\Delta} S^{(i)}_{EE} \boldsymbol{w}^{(i)}_{E,\Delta} \leq C(p)(1 + \log(H/h))^2 \boldsymbol{w}^{(i)T}_{E} \widetilde{S}^{(i)}_{EE} \boldsymbol{w}^{(i)}_{E}. \tag{5.1}$$

We note that this is an estimate for only one subdomain at a time, a feature of BDDC deluxe, which can greatly simplifies the analysis; see [4, Sect. 4] or [26, Sect. 4].

In what follows, we also denote by a thin edge, the mesh line associated with the basis functions of Nedélec space introduced in Subsection 3.2.

Given the close connection between 2D problems in $\mathbf{H}(curl)$ and $H(div)$, we will try to use the two-dimensional counter-part of [19, Lemma 4.8]. That result, for a standard lowest order finite element problem, would be the origin of a factor $(1+\log(H/h))^2$ in our final result with H/h standing for $\max_i H_i/h_i$ where H_i is the diameter of the subdomain Ω_i and h_i its minimal mesh size of the elements of Ω_i. Translating this result for $H(div)$ and Raviart-Thomas elements to $\mathbf{H}(curl)$, we find that for the lowest order Nédélec elements and with an edge e of a thin interface, we have a *thin edge lemma*:

$$\boldsymbol{w}_{e,\Delta}^{(i)T} S_{ee}^{(i)} \boldsymbol{w}_{e,\Delta}^{(i)} \leq C(1+\log(H/h))^2 \boldsymbol{w}_e^{(i)T} \widetilde{S}_{ee}^{(i)} \boldsymbol{w}_e^{(i)}. \tag{5.2}$$

Here $\boldsymbol{w}_{e,\Delta}^{(i)}$ is the dual part of the restriction of $\boldsymbol{w}^{(i)}$ to the thin edge e obtained by subtracting the average of $\boldsymbol{w}^{(i)} \cdot \boldsymbol{t}$ over e from the tangential component of $\boldsymbol{w}_e^{(i)}$; here $\boldsymbol{t}$ is the unit tangent vector along e. We will also refer to this condition as a *primal constraint*. This edge lemma suggests obvious primal elements for our algorithm making the average over each thin edge of the tangential component of all elements of the dual space vanish. The proofs in [19] were developed for tetrahedral meshes, but they can easily be revised to hold for the hexahedral case.

We have tried to find a similar, simple, and effective choice of primal constraints for the fat vertices of the fat interface but without success. We then noticed, in our experiments, that for all p, our adaptive algorithm assigns a relative large number of primal variables to each fat vertex when we insist on scalability and small condition numbers for our BDDC deluxe algorithm; see Section 6. Thus, while the overall number of dofs for a fat vertex equals $2(p-1)p$, in all our successful runs, the number of fat vertex primal variables have exceeded $(p-1)p$. We note that the number of dofs of a fat edge typically is considerably much larger than that of a fat vertex and that our numerical experiments, indicate that $2p-1$ primal variables per fat edge serve our purposes very well.

For a subdomain in the interior of a patch, each fat edge is shared by two subdomains and each vertex by four. The primal variables are shared between all relevant subdomains. Taking these facts into account, we find that the number of primal variables per subdomain is $(\frac{1}{2}4)(2p-1) + (\frac{1}{4}4), 2p(p-1) = 2(p^2+p-1)$, if we use one primal variable for each thin edge and make all fat vertex variables primal.

We remark that in our work on adaptive BDDC deluxe algorithms for scalar elliptic problems reported in [5], we have found that smaller primal spaces have been sufficient for rapid convergence of our iterations.

The following lemma is central in the analysis of the finite element case.

LEMMA 1. *(trace theorem) For all* $\boldsymbol{w} \in \mathbf{H}(curl; \widetilde{\Omega})$,

$$\|\boldsymbol{w} \cdot \boldsymbol{t}\|^2_{H^{-1/2}(\partial\widetilde{\Omega})} \leq C(H^2 \|curl(\boldsymbol{w})\|^2_{L^2(\widetilde{\Omega})} + \|\boldsymbol{w}\|^2_{L^2(\widetilde{\Omega})}).$$

Here C is a constant independent of H, the diameter of $\widetilde{\Omega}$.

Proof. A standard result for $H(div)$, which is a counterpart of this lemma, obtained by using Green's formula, is given in [28, Lemma 2.1]. Our proof is then obtained for $\mathbf{H}(curl)$ by a rotation of the variables. □

We find by using Lemma 1 that

$$b\|\boldsymbol{w} \cdot \boldsymbol{t}\|^2_{H^{-1/2}(\partial\widetilde{\Omega})} \leq \max(1, bH^2/a)(a\|curl(\boldsymbol{w})\|^2_{L^2(\widetilde{\Omega})} + b\|\boldsymbol{w}\|^2_{L^2(\widetilde{\Omega})})$$

To advance our understanding, we could assume the $(b_i H_i^2/a_i)$ have modest values for all i; we will refer to such problems as being curl-dominated. However, our experiments suggest good convergence also for mass-dominated problems; see Tests 5 and 6 in the next section..

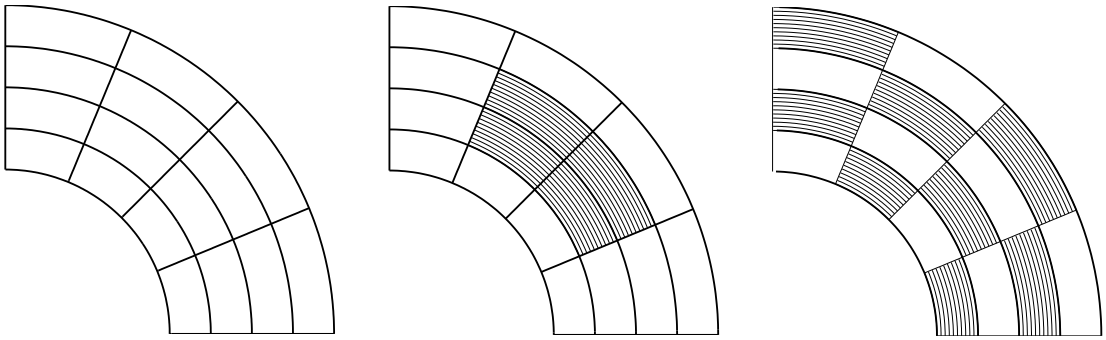


FIG. 6.1. *a) Example of quarter-ring domain with* 4×4 *subdomains. b) Domain for central jumping coefficient test with central region denoted in gray. c) Domain for checkerboard jumping coefficient test.*

In the finite element case, the energy defined by the bilinear form $a_{\Omega_i}(\cdot,\cdot)$ of (2.1), restricted to the discrete harmonic functions, can be bounded by b_i times the square of the $H^{-1/2}(\partial\Omega_i)-$norm of the tangential component provided that the average over $\partial\Omega_i$ of that component vanishes; see [19, Corollary 4.3] for the corresponding result for $H(div)$.

We are now ready to formulate a conjecture which is strongly supported by our numerical experiments:

CONJECTURE 1. *Consider a BDDC deluxe algorithm with all fat vertex variables primal and choose the primal subspace associated with the fat edges such that* $\int_e \boldsymbol{w}_\Delta \cdot \boldsymbol{t} = 0$ *for all thin edges and all elements of the dual space. Then, the condition number of the BDDC deluxe algorithm is bounded by*

$$C(p)(1+\log(H/h))^2.$$

Here $C(p)$ *is independent of the number of subdomains and the dimension of the subdomain problems, i.e., of* H/h.

In our numerical experiments, we essentially see no growth of $C(p)$ with p. Therefore, a thin edge lemma similar to (5.2), with a constant growing only logarithmically with p, might be within reach for $p > 1$. The arguments using the lowest order Nedélec elements could then potentially be eliminated. We note that such a result is available for 2D elliptic problems formulated in H^1 by combining [22, Lemmas 4.14 and 4.15]. That interesting paper focuses on cases without fat interfaces, and also develops other tools for higher order isogeometric analysis which are required for the analysis of FETI-DP and BDDC algorithms which are not of the deluxe family.

6. Numerical results. The model problem (2.2) is discretized on a 2D unit square and a quarter-ring domain, see Fig. 6.1, using isogeometric NURBS spaces with a mesh size h, spline degree p and, in all experiments reported, with a maximum regularity of $k = p-1$. The domain is decomposed into K non-overlapping subdomains of characteristic size H. The Schur complement problems are solved by the PCG method with our isogeometric BDDC deluxe preconditioners, with a zero initial guess and a stopping criterion of a 10^{-6} reduction of the Euclidean norm of the PCG residual.

In the following tests, we study how the convergence rate of the BDDC preconditioner depends on the discretization parameters h, K, p and on the number of primal constraints for each fat vertex ($N_{\Pi V}$) and fat edge ($N_{\Pi E}$). We recall that the number of dofs for each fat vertex is $2p(p-1)$, while the number of thin edges for each fat edge is $2p-1$. Except for the experiments of the first subsection,

$p=2,\ k=1,\ N_{\Pi E}=2,\ N_{\Pi V}=4$										
	$h=1/8$		$h=1/16$		$h=1/32$		$h=1/64$		$h=1/128$	
K	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$
2×2	4.80	7	5.21	7	5.69	7	6.16	7	6.63	7
4×4			20.77	14	22.49	16	24.26	17	25.82	19
8×8					74.24	26	80.77	29	87.48	29
16×16							287.52	47	311.48	52
32×32									1144.54	66

TABLE 6.1

Test 1, quasi-optimality and scalability test with non-adaptive coarse space. Unit square domain, $p=2$, $k=1$. BDDC deluxe condition number cond and iteration counts $\mathrm{n_{it}}$ as functions of the number of subdomains K and mesh size h. $N_{\Pi E}$:=number of primal dofs for each fat edge; $N_{\Pi V}$:=number of primal dofs for each fat vertex.

the primal space is build using the adaptive strategy described in Section 4. The condition number, *cond*, of the BDDC deluxe preconditioned operator is estimated using the parameters of the conjugate gradient algorithm and Lanczos' algorithm.

The numerical tests were performed with a MATLAB code based on the GeoPDEs library, see [12, 25], run on a Linux workstation.

6.1. Test 1: quasi-optimality and scalability with non-adaptive coarse space. The 2D $\mathbf{H}(curl)$ discrete model problem is solved on the unit square for fixed spline parameters $p=2, k=1$. All fat vertex dofs are primal, i.e. $N_{\Pi V}=4$, while for each fat edge, we select only one primal constraint on each of the two outmost thin edges, thus $N_{\Pi E}=2$. Table 6.1 shows the results varying the number of subdomains K and the fine mesh size h. Moving along the diagonal of the table, the subdomain to element mesh size ratio H/h is kept constant and we see that this primal space choice is not scalable for an increasing number of subdomains K, since both iteration counts and condition numbers grow with K. On the other hand, we have quasi-optimality with respect to the ratio H/h, since moving along each row the ratio H/h increases and the growth now appears to be logarithmic. This experiment shows that even for this quite simple case, two thin edge constraints per fat edge will not always result in a scalable algorithm.

6.2. Test 2: quasi-optimality and scalability with an adaptive coarse space. In all tests from now on, the 2D $\mathbf{H}(curl)$ discrete model problem is solved adopting the adaptive coarse space strategy described in Section 4.

We first consider the unit square domain and make all fat vertex variables primal, thus using adaptivity only for the fat edges (Tables 6.2 and 6.3). Table 6.2 shows the results varying the number of subdomains K and the fine mesh size h for fixed $p=2, k=1$, with two choices of primal constraints: ($N_{\Pi E}=2,\ N_{\Pi V}=4$) and ($N_{\Pi E}=3,\ N_{\Pi V}=4$). In the first case, ($N_{\Pi E}=2,\ N_{\Pi V}=4$), moving along the diagonal of the table, the subdomain to element mesh size ratio H/h is kept constant and we see that this primal space choice is not scalable for an increasing number of subdomains K, since both iteration counts and condition numbers grow rapidly with K. However, we do observe quasi-optimality with respect to the ratio H/h, since moving along each row the ratio H/h increases and the growth appears to be logarithmic. Conversely, in the second case, ($N_{\Pi E}=3,\ N_{\Pi V}=4$), the solver seems both to be scalable in K, since iterations and condition numbers remain bounded moving along the diagonals, and quasi-optimal in H/h. A different behavior occurs in the case $p=3, k=2$, reported in Table 6.3. Here the choices of primal constraints are the following: ($N_{\Pi E}=2,\ N_{\Pi V}=12$) and ($N_{\Pi E}=3,\ N_{\Pi V}=12$). In both cases, moving along the diagonals of the table, the solver appears scalable in K, even though for large values of the ratio H/h we are still far from the asymptotic behavior. On the other hand, moving along each row, the condition number growth seems more than logarithmic. We note that the second

$p=2,\ k=1,\ N_{\Pi E}=2,\ N_{\Pi V}=4$										
	$h=1/8$		$h=1/16$		$h=1/32$		$h=1/64$		$h=1/128$	
K	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$
2×2	1.16	5	1.30	5	1.48	6	1.69	6	1.92	6
4×4			61.28	16	71.48	18	83.86	20	97.35	22
8×8					398.30	60	470.20	67	556.20	73
16×16							1705.37	94	2014.53	137
32×32									6917.75	154
$p=2,\ k=1,\ N_{\Pi E}=3,\ N_{\Pi V}=4$										
	$h=1/8$		$h=1/16$		$h=1/32$		$h=1/64$		$h=1/128$	
K	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$
2×2	1.16	5	1.05	4	1.07	4	1.10	4	1.15	5
4×4			1.61	8	2.21	9	2.89	10	3.69	11
8×8					1.70	8	2.38	10	3.17	12
16×16							1.73	8	2.44	10
32×32									1.73	8

TABLE 6.2

Test 2, quasi-optimality and scalability test with adaptive coarse space. Unit square domain, $p=2,\ k=1$. BDDC deluxe condition number cond and iteration counts $\mathrm{n_{it}}$ as functions of the number of subdomains K and mesh size h. $N_{\Pi E}$:=number of primal dofs for each fat edge; $N_{\Pi V}$:=number of primal dofs for each fat vertex.

$p=3,\ k=2,\ N_{\Pi E}=2,\ N_{\Pi V}=12$												
	$h=1/8$		$h=1/16$		$h=1/32$		$h=1/64$		$h=1/128$		$h=1/256$	
K	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$
2×2	3.13	7	6.80	7	10.37	7	13.34	7	16.05	8	18.74	8
4×4			5.27	14	20.07	18	50.09	21	93.92	25	129.70	27
8×8					6.46	16	30.89	32	100.00	45	255.09	58
16×16							6.99	16	39.44	36	151.54	64
32×32									7.32	16	44.25	36
$p=3,\ k=2,\ N_{\Pi E}=3,\ N_{\Pi V}=12$												
	$h=1/8$		$h=1/16$		$h=1/32$		$h=1/64$		$h=1/128$		$h=1/256$	
K	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$
2×2	1.14	5	1.23	5	1.39	6	1.57	6	1.78	6	2.02	6
4×4			3.45	11	14.02	14	37.85	18	66.49	21	91.83	23
8×8					4.49	15	23.12	31	89.45	46	251.19	63
16×16							4.91	16	28.44	34	131.82	66
32×32									5.06	16	31.32	34

TABLE 6.3

Test 2, quasi-optimality and scalability test with adaptive coarse space. Unit square domain, $p=3,\ k=2$. BDDC deluxe condition number cond and iteration counts $\mathrm{n_{it}}$ as functions of the number of subdomains K and mesh size h. $N_{\Pi E}$:=number of primal dofs for each fat edge; $N_{\Pi V}$:=number of primal dofs for each fat vertex.

choice of $N_{\Pi E}$ results in a much better performance.

Then, we consider the quarter-ring domain and use adaptivity for both the fat vertices and edges. Table 6.4 shows the results varying the number of subdomains K and the fine mesh size h for fixed $p=2, k=1$, with two choices of primal constraints: ($N_{\Pi E}=3,\ N_{\Pi V}=2$) and ($N_{\Pi E}=3,\ N_{\Pi V}=3$). In the first case, ($N_{\Pi E}=3,\ N_{\Pi V}=2$), the primal space choice is not scalable for an increasing number of subdomains K, since both iteration counts and condition numbers grow

$p=2,\ k=1,\ N_{\Pi E}=3,\ N_{\Pi V}=2$										
	$h=1/8$		$h=1/16$		$h=1/32$		$h=1/64$		$h=1/128$	
K	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$
2×2	1.06	5	1.02	4	1.02	4	142.26	5	520.22	5
4×4			54.02	13	221.69	19	849.28	23	3427.22	30
8×8					281.07	40	1161.11	64	4559.94	79
16×16							1360.93	102	5540.98	141
32×32									6168.46	190
$p=2,\ k=1,\ N_{\Pi E}=3,\ N_{\Pi V}=3$										
	$h=1/8$		$h=1/16$		$h=1/32$		$h=1/64$		$h=1/128$	
K	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$
2×2	1.06	4	1.02	4	1.02	4	1.05	4	1.08	4
4×4			1.76	8	2.40	9	3.11	10	3.96	10
8×8					2.10	10	3.01	12	3.99	14
16×16							2.29	10	3.42	13
32×32									2.40	10

TABLE 6.4

Test 2, quasi-optimality and scalability test with adaptive coarse space. Quarter ring domain, $p=2,\ k=1$. BDDC deluxe condition number cond and iteration counts $\mathrm{n_{it}}$ as functions of the number of subdomains K and mesh size h. $N_{\Pi E}$:=number of primal dofs for each fat edge; $N_{\Pi V}$:=number of primal dofs for each fat vertex.

$p=3,\ k=2,\ N_{\Pi E}=5,\ N_{\Pi V}=6$										
	$h=1/8$		$h=1/16$		$h=1/32$		$h=1/64$		$h=1/128$	
K	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$
2×2	1.19	5	1.08	5	1.05	5	1.04	4	1.04	4
4×4			17.68	11	24.75	13	28.19	16	29.98	18
8×8					53.02	29	64.22	34	71.22	38
16×16							130.04	54	136.79	57
32×32									248.96	83
$p=3,\ k=2,\ N_{\Pi E}=5,\ N_{\Pi V}=7$										
	$h=1/8$		$h=1/16$		$h=1/32$		$h=1/64$		$h=1/128$	
K	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$
2×2	1.09	5	1.05	5	1.05	4	1.04	4	1.04	4
4×4			2.80	10	2.90	10	2.97	11	3.34	12
8×8					2.93	12	2.95	12	3.25	13
16×16							2.97	12	3.00	12
32×32									3.01	12

TABLE 6.5

Test 2, quasi-optimality and scalability test with adaptive coarse space. Quarter ring domain, $p=3,\ k=2$. BDDC deluxe condition number cond and iteration counts $\mathrm{n_{it}}$ as functions of the number of subdomains K and mesh size h. $N_{\Pi E}$:=number of primal dofs for each fat edge; $N_{\Pi V}$:=number of primal dofs for each fat vertex.

with K. Nor do we have quasi-optimality with respect to the ratio H/h, since moving along each row the ratio H/h increases and the growth appears to be quadratic. Conversely, in the second case, ($N_{\Pi E}=3,\ N_{\Pi V}=3$), the solver seems both scalable in K, since iterations and condition numbers remain bounded moving along the diagonals, and quasi-optimal in H/h, since the growth of the condition number now appears to be logarithmic moving along the rows. Analogous findings hold in case of spline parameters $p=3, k=2$, see Table 6.5, where the scalable and quasi-optimal

$p=4,\ k=3,\ N_{\Pi E}=7,\ N_{\Pi V}=17$								
	$h=1/16$		$h=1/32$		$h=1/64$		$h=1/128$	
K	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$
2×2	1.45	5	7.99	7	16.28	6	23.86	5
4×4			28.91	16	36.93	18	40.02	20
8×8					45.36	32	48.53	35
16×16							54.59	39
$p=4,\ k=3,\ N_{\Pi E}=7,\ N_{\Pi V}=18$								
	$h=1/16$		$h=1/32$		$h=1/64$		$h=1/128$	
K	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$
2×2	1.08	5	1.05	5	1.05	5	1.04	4
4×4			2.89	10	2.52	11	3.07	11
8×8					2.95	12	2.88	13
16×16							3.30	13

TABLE 6.6

Test 2, quasi-optimality and scalability test with adaptive coarse space. Quarter-ring domain, $p=4$, $k=3$. BDDC deluxe condition number cond and iteration counts $\mathrm{n_{it}}$ as functions of the number of subdomains K and mesh size h. $N_{\Pi E}$:=number of primal dofs for each fat edge; $N_{\Pi V}$:=number of primal dofs for each fat vertex.

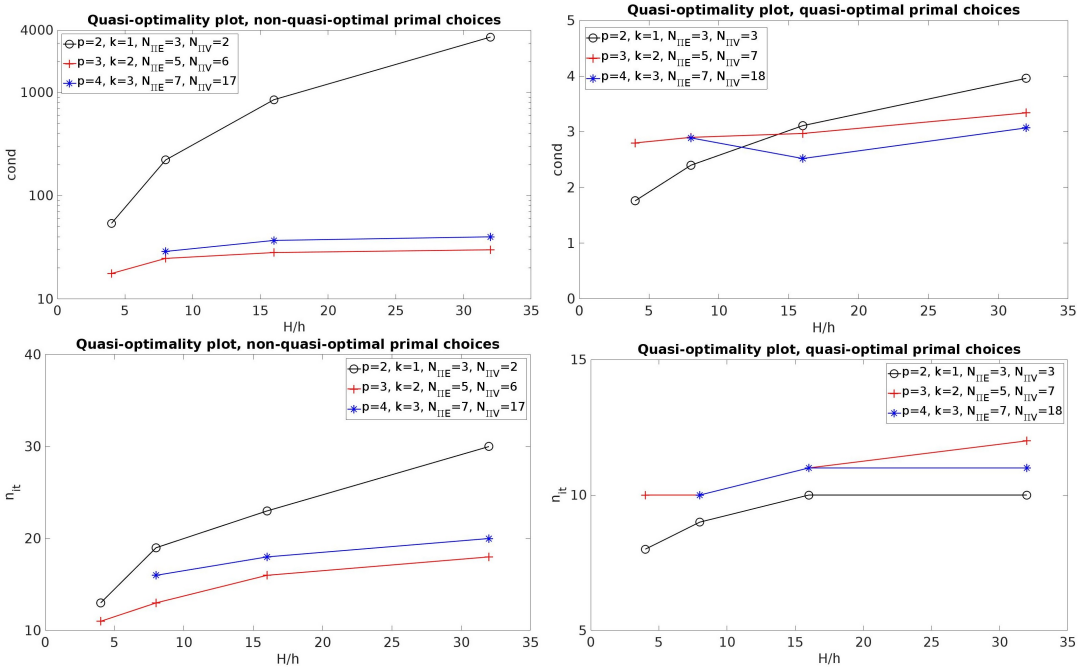


FIG. 6.2. *Test 2, quasi-optimality test with adaptive coarse space for spline parameter pairs ($p=2,\ k=1$), ($p=3,\ k=2$), ($p=4,\ k=3$) on the quarter-ring domain. Plots of BDDC deluxe condition number cond (top panels) and iteration counts $\mathrm{n_{it}}$ (bottom panels) as functions of the ratio H/h. $N_{\Pi E}$:=number of primal dofs for each fat edge; $N_{\Pi V}$:=number of primal dofs for each fat vertex.*

primal choice is ($N_{\Pi E}=5,\ N_{\Pi V}=7$), and for $p=4, k=3$, see Table 6.6, where the scalable and quasi-optimal primal choice is ($N_{\Pi E}=7,\ N_{\Pi V}=18$). We have also plotted the condition numbers and iteration counts in Figures 6.2 and 6.3 for greater clarity. While we do not have a precise rule

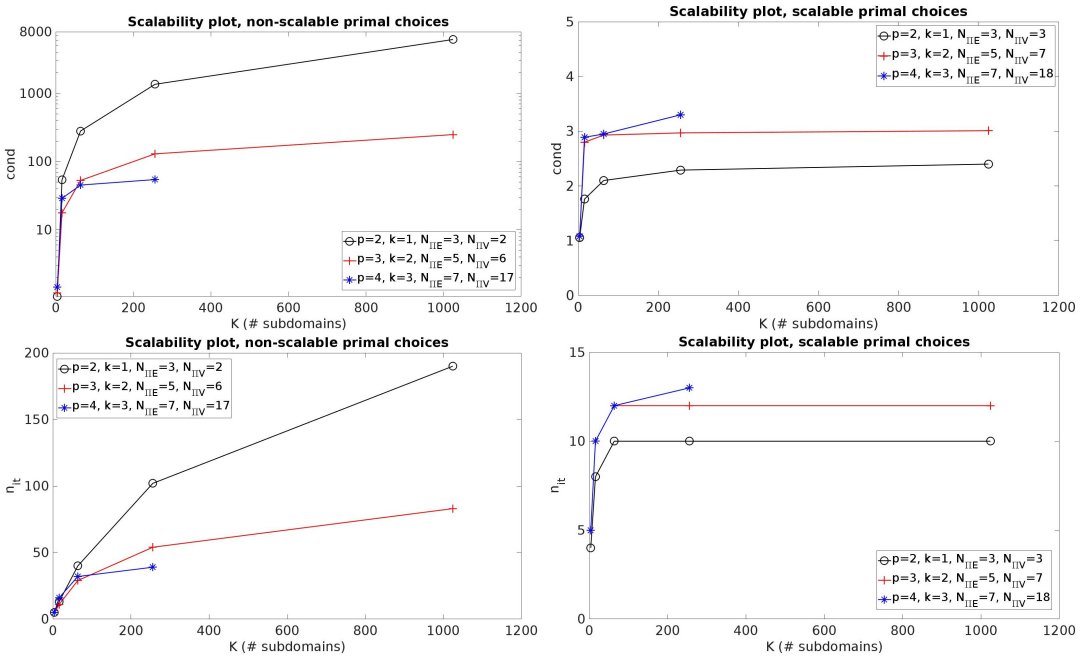


FIG. 6.3. *Test 2, scalability test with adaptive coarse space for spline parameter pairs ($p = 2,\ k = 1$), ($p = 3,\ k = 2$), ($p = 4,\ k = 3$) on the quarter-ring domain. Plots of BDDC deluxe condition number cond (top panels) and iteration counts* $\mathrm{n_{it}}$ *(bottom panels) as functions of the number of subdomains K. $N_{\Pi E}$:=number of primal dofs for each fat edge; $N_{\Pi V}$:=number of primal dofs for each fat vertex.*

$p = 2,\ k = 1,\ h = 1/32,\ K = 8 \times 8$						
unit square domain						
	$N_{\Pi V} = 1$		$N_{\Pi V} = 2$		$N_{\Pi V} = 3$	
	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$
$N_{\Pi E} = 1$	975.50	99	764.97	122	405.96	59
$N_{\Pi E} = 2$	972.20	96	646.67	109	398.18	60
$N_{\Pi E} = 3$	878.96	49	560.94	52	1.70	8
quarter-ring domain						
	$N_{\Pi V} = 1$		$N_{\Pi V} = 2$		$N_{\Pi V} = 3$	
	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$
$N_{\Pi E} = 1$	565.67	101	370.12	82	203.98	56
$N_{\Pi E} = 2$	516.69	93	322.76	80	188.87	54
$N_{\Pi E} = 3$	457.42	46	281.07	41	2.10	10

TABLE 6.7

Test 3, choice of adaptive primal constraints for each fat edge ($N_{\Pi E}$) and fat vertex ($N_{\Pi V}$). BDDC deluxe condition number cond and iteration counts $\mathrm{n_{it}}$ *as functions of $N_{\Pi E}$ and $N_{\Pi V}$. Fixed mesh size $h = 1/32$, number of subdomains $K = 8 \times 8$, $p = 2,\ k = 1$.*

for the minimal number of vertex and edge constraints required in order to guarantee scalability and/or quasi-optimality of our BDDC preconditioner, we present in the next section a numerical study of the optimal choices of vertex and edge primal constraints for specific test cases.

$p=3,\ k=2,\ h=1/32,\ K=4\times 4$						
unit square domain						
	$N_{\Pi V}=5$		$N_{\Pi V}=6$		$N_{\Pi V}=7$	
	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$
$N_{\Pi E}=3$	103.58	29	90.00	24	24.34	21
$N_{\Pi E}=4$	82.79	23	26.41	20	19.20	20
$N_{\Pi E}=5$	4.81	11	4.81	11	2.90	12
quarter-ring domain						
	$N_{\Pi V}=5$		$N_{\Pi V}=6$		$N_{\Pi V}=7$	
	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$
$N_{\Pi E}=3$	240.97	33	124.78	24	39.48	18
$N_{\Pi E}=4$	237.58	28	68.17	20	20.53	15
$N_{\Pi E}=5$	174.45	19	24.74	13	2.90	10

TABLE 6.8

Test 3, choice of adaptive primal constraints for each fat edge ($N_{\Pi E}$) and fat vertex ($N_{\Pi V}$). BDDC deluxe condition number cond and iteration counts $\mathrm{n_{it}}$ as functions of $N_{\Pi E}$ and $N_{\Pi V}$. Fixed mesh size $h = 1/32$, number of subdomains $K = 4\times 4$, $p=3,\ k=2$.

$k=p-1,\ h=1/64,\ K=4\times 4$								
	unit square domain				quarter-ring domain			
p	$N_{\Pi E}$	$N_{\Pi V}$	cond	$\mathrm{n_{it}}$	$N_{\Pi E}$	$N_{\Pi V}$	cond	$\mathrm{n_{it}}$
2	3	3	2.89	10	3	3	3.11	10
3	5	7	2.85	11	5	7	2.97	11
4	7	18	2.13	10	7	18	2.52	11
5	9	29	2.24	10	9	32	2.15	10
6	11	59	1.04	5	11	56	1.60	8
7	13	82	2.03	9	13	82	6.81	14

TABLE 6.9

Test 4, robustness with respect to the spline polynomial degree p. $N_{\Pi E}$:=number of primal dofs for each fat edge, $N_{\Pi V}$:=number of primal dofs for each fat vertex, BDDC deluxe condition number cond and iteration counts $\mathrm{n_{it}}$ as functions of the spline polynomial degree p. Fixed mesh size $h = 1/64$, number of subdomains $K = 4\times 4$, $k = p-1$.

6.3. Test 3: choice of adaptive primal constraints for each fat edge ($N_{\Pi E}$) and fat vertex ($N_{\Pi V}$). In this test, we study, in more depth, the number of primal constraints for fat edges ($N_{\Pi E}$) and fat vertices ($N_{\Pi V}$), needed to obtain a robust BDDC solver. In Tables 6.7 and 6.8, we report on results for the case of spline parameters ($p=2,\ k=1$) and ($p=3,\ k=2$), respectively. For ($p=2,\ k=1$), both on the unit square and quarter-ring domain, the robust choice of primal constraints is ($N_{\Pi E}=3,\ N_{\Pi V}=3$), since the condition number drops from several hundreds to 1.70 and 2.10, respectively, only in this case. For ($p=3,\ k=2$), on the unit square, the robust choice of primal constraints yielding single digits condition numbers is ($N_{\Pi E}=5,\ N_{\Pi V}=5$), whereas on the quarter-ring, we need ($N_{\Pi E}=5,\ N_{\Pi V}=7$) primal constraints.

6.4. Test 4: robustness with respect to the spline polynomial degree p. In Table 6.9, we report the BDDC condition numbers and iteration counts for an increasing polynomial degrees up to $p=7$ and for maximal regularity $k=p-1$, with fixed parameters $K=4\times 4, h=1/64$, on both the unit square and the quarter-ring domains. By fixing $N_{\Pi E}=2p-1$, i.e. the number of thin edges, and with a suitable choice $N_{\Pi V}$, reported in the table, the solver is robust with respect to growth of p, since single digits condition numbers are obtained with a linear increase of the number

$p=2,\ k=1,\ 1/h=64\ , K=4\times 4$				
	$N_{\Pi E}=3,\ N_{\Pi V}=3$		$N_{\Pi E}=3,\ N_{\Pi V}=4$	
a	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$
$1e-6$	1.13	4	1.00	3
$1e-4$	1.18	6	1.18	6
$1e-2$	2.08	9	2.07	9
$1e+0$	3.11	10	3.11	10
$1e+2$	3.16	10	3.16	9
$1e+4$	4.88	11	3.16	10
$1e+6$	7.33	14	3.16	10
$p=3,\ k=2,\ 1/h=64\ , K=4\times 4$				
	$N_{\Pi E}=5,\ N_{\Pi V}=9$		$N_{\Pi E}=5,\ N_{\Pi V}=10$	
a	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$
$1e-6$	1.10	5	1.07	4
$1e-4$	1.19	6	1.14	6
$1e-2$	1.55	8	1.55	8
$1e+0$	2.59	9	2.59	9
$1e+2$	2.64	9	2.64	9
$1e+4$	1.74e+6	32	2.64	11
$1e+6$	2.83e+8	90	2.64	11

TABLE 6.10

Test 5, robustness with respect to the coefficient a in the $\mathbf{H}(curl)$ bilinear form. BDDC deluxe condition number cond and iteration counts $\mathrm{n_{it}}$ as functions of coefficient a. $N_{\Pi E}$:=number of primal dofs for each fat edge; $N_{\Pi V}$:=number of primal dofs for each fat vertex.

of edge constraints and a slightly faster than linear increase of the number of vertex constraints.

6.5. Test 5: robustness with respect to the coefficient a in the $\mathbf{H}(curl)$ bilinear form. In Table 6.10, we study the BDDC deluxe performance when the coefficient a in the $\mathbf{H}(curl)$ bilinear form (2.1) varies by 12 orders of magnitude, from 10^{-6} to 10^6. The quarter-ring domain is discretized with mesh size $1/h=64$, $K=4\times 4$ subdomains, selecting $p=2,k=1$ (top) or $p=3,k=2$ (bottom) and different choices of the number of primal constraints for each fat edge ($N_{\Pi E}$) and for each fat vertex ($N_{\Pi V}$). In the mass-dominated case (small values of a), the BDDC deluxe preconditioner performs very well in all tests considered. In the curl-dominated case (large values of a), the BDDC performance degenerates when $p=3,k=2$ and $N_{\Pi E}=5$, $N_{\Pi V}=9$ primal constraints are selected, while a robust performance is restored by increasing the number of primal dofs for each fat vertex to $N_{\Pi V}=10$. The condition number reduction is striking, since *cond* drops 6 orders of magnitude by selecting just one more primal constraints per fat vertex.

6.6. Test 6: robustness with respect to jumping coefficients. We study here the BDDC deluxe performance in case of central and checkerboard jumping coefficient test. The quarter-ring domain is discretized with mesh size $1/h=64$, $K=4\times 4$ subdomains, selecting $p=2,k=1$ (top) or $p=3,k=2$ (bottom) and different choices of number of primal dofs for each fat edge ($N_{\Pi E}$) and for each fat vertex ($N_{\Pi V}$).

In the central jumping coefficient test (Table 6.11 and Fig. 6.1 b)), the coefficient a of the $\mathbf{H}(curl)$ bilinear form varies by 12 orders of magnitude, from 10^{-6} to 10^6, in the central region consisting of 2×2 gray subdomains, whereas $a=1$ outside the central region. In the mass-dominated case, the BDDC deluxe preconditioner performs very well in all tests considered. In the curl-dominated case, the BDDC performance degenerates when $p=3,k=2$ and $N_{\Pi E}=5$, $N_{\Pi V}=9$ primal constraints are selected, while a robust performance is restored by increasing the number of primal dofs for each

$p=2,\ k=1,\ 1/h=64\ , K=4\times 4$				
	$N_{\Pi E}=3,\ N_{\Pi V}=3$		$N_{\Pi E}=3,\ N_{\Pi V}=4$	
a in the jumping region	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$
$1e-6$	2.14	7	1.15	6
$1e-4$	1.19	6	1.14	6
$1e-2$	2.30	9	2.30	9
$1e+0$	3.11	10	3.11	10
$1e+2$	3.14	10	3.14	10
$1e+4$	14.08	12	3.14	10
$1e+6$	11.39	16	3.14	11
$p=3,\ k=2,\ 1/h=64\ , K=4\times 4$				
	$N_{\Pi E}=5,\ N_{\Pi V}=9$		$N_{\Pi E}=5,\ N_{\Pi V}=10$	
a in the jumping region	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$
$1e-6$	5.42	13	4.62	13
$1e-4$	4.08	11	2.12	10
$1e-2$	2.12	10	1.65	8
$1e+0$	2.59	9	2.59	9
$1e+2$	2.62	10	2.62	10
$1e+4$	6.65e+5	21	2.90	11
$1e+6$	5.40e+7	52	12.67	18

TABLE 6.11

Test 6, central jumping coefficient test, quarter ring domain. BDDC deluxe condition number cond and iteration counts $\mathrm{n_{it}}$ *as functions of coefficient* a *in the central jump region of* 2×2 *gray subdomains, while* $a=1$ *outside the central region (see Fig. 6.1 b).* $N_{\Pi E}$*:=number of primal dofs for each fat edge;* $N_{\Pi V}$*:=number of primal dofs for each fat vertex.*

fat vertex to $N_{\Pi V}=10$.

In the checkerboard jumping coefficient test (Table 6.12 and Fig. 6.1 c)), the coefficient a of the $\mathbf{H}(curl)$ bilinear form varies of 12 orders of magnitude, from 10^{-6} to 10^{6}, in the gray subdomains, whereas $a=1$ in the white subdomains. The behavior of the preconditioner is analogous to the central jumping coefficient case, with a good BDDC performance in the mass-dominated case, while in the curl-dominated case, for $p=3, k=2$, we need to increase the number of vertex primal constraints $N_{\Pi V}$ for each fat vertex from 9 to 10 in order to obtain a robust performance, even though the increase of the condition number with $N_{\Pi V}=9$ is less abrupt than in the central jumping coefficient case.

We remark that in this test we have maintained the maximal regularity at the interface of the subdomains, in order to challenge the performance of the solver. We believe that reducing the regularity of the approximation space to C^0 at the jumping interface would improve the behavior of the BDDC preconditioner, as observed for different PDE problems in our previous works [3, 4].

REFERENCES

[1] Douglas N. Arnold, Richard S. Falk, and Ragnard Winter. Multigrid in H(div) and H(curl). *Numer. Math.*, 85:197–217, 2000.

[2] Lourenço Beirão da Veiga, Annalisa Buffa, G. Sangalli, and Rafael Vázquez. Mathematical analysis of variational isogeometric methods. *Acta Numer.*, 23:157–287, 2014.

[3] Lourenço Beirão da Veiga, Durkbin Cho, Luca F. Pavarino, and Simone Scacchi. BDDC preconditioners for isogeometric analysis. *Math. Models Methods Appl. Sci.*, 23(6):1099–1142, 2013.

[4] Lourenço Beirão da Veiga, Luca F. Pavarino, Simone Scacchi, Olof B. Widlund, and Stefano Zampini. Isogeometric BDDC preconditioners with deluxe scaling. *SIAM J. Sci. Comput.*, 36(3):A1118–A1139, 2014.

$p=2,\ k=1,\ 1/h=64\ , K=4\times 4$				
	$N_{\Pi E}=3,\ N_{\Pi V}=3$		$N_{\Pi E}=3,\ N_{\Pi V}=4$	
a in gray subds.	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$
$1e-6$	2.60	8	2.37	7
$1e-4$	2.33	7	2.30	7
$1e-2$	2.45	10	2.45	10
$1e+0$	3.11	10	3.11	10
$1e+2$	3.14	10	3.14	10
$1e+4$	3.14	11	3.14	10
$1e+6$	6.95	17	3.14	11
$p=3,\ k=2,\ 1/h=64\ , K=4\times 4$				
	$N_{\Pi E}=5,\ N_{\Pi V}=9$		$N_{\Pi E}=5,\ N_{\Pi V}=10$	
a in gray subds.	cond	$\mathrm{n_{it}}$	cond	$\mathrm{n_{it}}$
$1e-6$	7.03	15	5.22	13
$1e-4$	5.32	15	4.51	13
$1e-2$	1.88	9	1.85	8
$1e+0$	2.59	9	2.59	9
$1e+2$	2.62	9	2.62	9
$1e+4$	4.91	13	4.24	12
$1e+6$	59.89	26	11.09	15

TABLE 6.12

Test 6, checkerboard jumping coefficient test, quarter ring domain. BDDC deluxe condition number cond and iteration counts $\mathrm{n_{it}}$ *as functions of coefficient a in the gray subdomains, while* $a=1$ *in the white subdomains (see Fig. 6.1 c).* $N_{\Pi E}$*:=number of primal dofs for each fat edge;* $N_{\Pi V}$*:= number of primal dofs for each fat vertex.*

[5] Lourenço Beirão da Veiga, Luca F. Pavarino, Simone Scacchi, Olof B. Widlund, and Stefano Zampini. Adaptive selection of primal constraints for isogeometric BDDC deluxe preconditioners. *SIAM J. Sci. Comput.*, 39(1):A281–A302, 2017.

[6] Andrea Bressan and Giancarlo Sangalli. Isogeometric discretizations of the Stokes problem: stability analysis by the macroelement technique. *IMA J. Numer. Anal.*, 33(2):629–651, 2013.

[7] Annalisa Buffa, Giancarlo Sangalli, and Rafael Vázquez. Isogeometric analysis in electromagnetics: B-splines approximation. *Comput. Methods Appl. Mech. Engrg.*, 199(17-20):1143–1152, 2010.

[8] Annalisa Buffa, Giancarlo Sangalli, and Rafael Vázquez. Isogeometric methods for computational electromagnetics: B-spline and T-spline discretizations. *J. Comput. Phys.*, 257(part B):1291–1320, 2014.

[9] Juan G. Calvo. A BDDC algorithm with deluxe scaling for $H(\mathrm{curl})$ in two dimensions with irregular subdomains. *Math. Comp.*, 85(299):1085–1111, 2016.

[10] Juan G. Calvo and Olof B. Widlund. An adaptive choice of primal constraints for BDDC domain decomposition algorithms. *Electron. Trans. Numer. Anal.*, 45:524–544, 2016.

[11] J. Austin Cottrell, Thomas J.R. Hughes, and Yuri Bazilevs. *Isogeometric Analysis. Towards integration of CAD and FEA*. Wiley, 2009.

[12] Carlo de Falco, Alessandro Reali, and Rafael Vazquez. GeoPDEs: A research tool for isogeometric analysis of PDEs. *Adv. Eng. Softw.*, 42(12):1020 – 1034, 2011.

[13] Clark R. Dohrmann and Olof B. Widlund. An iterative substructuring algorithm for two-dimensional problems in H(curl) problems. *SIAM J. Numer. Anal.*, 50(3):1004–1028, 2012.

[14] Clark R. Dohrmann and Olof B. Widlund. Some recent tools and a BDDC algorithm for 3D problems in H(curl). In Randolph Bank, Michael Holst, Olof Widlund, and Jinchao Xu, editors, *Domain Decomposition Methods in Science and Engineering XX. Proceedings of the Twentieth International Conference on Domain Decomposition Methods*, number 91 in Lecture Notes in Computational Science and Engineering, pages 15–26, Heidelberg–Berlin, 2013. Springer. Held at the University of California at San Diego, CA, February 9–13, 2011.

[15] Clark R. Dohrmann and Olof B. Widlund. A BDDC algorithm with deluxe scaling for three-dimensional H(curl) problems. *Comm. Pure Appl. Math.*, 69(4):745–770, 2016.

[16] Ralf Hiptmair and Jinchao Xu. Nodal auxiliary space preconditioning in $\mathbf{H}(\mathbf{curl})$ and $\mathbf{H}(\mathrm{div})$ spaces. *SIAM J. Numer. Anal.*, 45(6):2483–2509, 2007.

[17] Qiya Hu, Shi Shu, and Jun Zou. A substructuring preconditioner for three-dimensional Maxwells equations. In Randolph Bank, Michael Holst, Olof Widlund, and Jinchao Xu, editors, *Domain Decomposition Methods in Science and Engineering XX. Proceedings of the Twentieth International Conference on Domain Decomposition Methods*, number 91 in Lecture Notes in Computational Science and Engineering, pages 73–84, Heidelberg–Berlin, 2013. Springer. Held at the University of California at San Diego, CA, February 9–13, 2011.

[18] Tzanio Kolev and Panayot S. Vassilevski. Parallel auxiliary space AMG for H(curl) problems. *J. Comput. Math.*, 27(5):604–623, 2012.

[19] Duk-Soon Oh, Olof B. Widlund, Stefano Zampini, and Clark R. Dohrmann. BDDC algorithms with deluxe scaling and adaptive selection of primal constraints for Raviart-Thomas vector fields. *Math. Comp.*, 87(310):659–692, 2018.

[20] Luca F. Pavarino, Simone Scacchi, Olof B. Widlund, and Stefano Zampini. Isogeometric BDDC deluxe preconditioners for linear elasticity. *Math. Models Methods Appl. Sci.*, 28(7):1337–1370, 2018.

[21] Clemens Pechstein and Clark. R. Dohrmann. A unified framework for adaptive BDDC. *Electr. Trans. Numer. Anal*, 46:273–336, 2017.

[22] Rainer Schneckenleitner and Stefan Takacs. Convergence theory for IETI-DP solvers for discontinuous Galerkin Isogeometric Analysis that is explicit in h and p. *Math. Models Methods Appl. Sci.*, 30(11):2067–2103, 2020.

[23] Andrea Toselli and Axel Klawonn. A FETI domain decomposition method for edge element approximations in two dimensions with discontinuous coefficients. *SIAM J. Numer. Anal.*, 39(3):932–956, 2001.

[24] Andrea Toselli, Olof B. Widlund, and Barbara I. Wohlmuth. An iterative substructuring method for Maxwell's equations in two dimensions. *Math. Comp.*, 70(235):935–949, 2001.

[25] Rafael Vazquez. A new design for the implementation of isogeometric analysis in Octave and MATLAB: GeoPDEs 3.0. *Comput. Math. with Appl.*, 72(3):523 – 554, 2016.

[26] Olof B. Widlund. BDDC domain decomposition algorithms. In *75 Years of Mathematics of Computation*, number 754 in Contemporary Mathematics, pages 261–281, Providence, R.I., 2020. American Mathematical Society.

[27] Olof B. Widlund, Stefano Zampini, Simone Scacchi, and Luca F. Pavarino. Block FETI–DP/BDDC preconditioners for mixed isogeometric discretizations of three-dimensional almost incompressible elasticity. *Math. Comp.*, 90(330):1773–1797, 2021.

[28] Barbara I. Wohlmuth, Andrea Toselli, and Olof B. Widlund. An iterative substructuring method for Raviart-Thomas vector fields in three dimensions. *SIAM J. Numer. Anal.*, 37(5):1657–1676, 2000.

[29] Stefano Zampini. Adaptive BDDC deluxe methods for H(curl). In *Domain decomposition methods in science and engineering XXIII*, volume 116 of *Lect. Notes Comput. Sci. Eng.*, pages 285–292. Springer, Cham, 2017.