

## A ROBUST MULTIGRID SOLVER FOR ISOGEOMETRIC ANALYSIS BASED ON MULTIPLICATIVE SCHWARZ SMOOTHERS\*

ALVARO PE DE LA RIVA<sup>†</sup>, CARMEN RODRIGO<sup>†</sup>, AND FRANCISCO J. GASPAR<sup>‡</sup>

**Abstract.** The design of fast solvers for isogeometric analysis is receiving a lot of attention due to the challenge that offers to find an algorithm with a robust convergence with respect to the spline degree. Here, we analyze the application of geometric multigrid methods to this type of discretization, and we propose a multigrid approach based on overlapping multiplicative Schwarz methods as smoothers. The size of the blocks considered within these relaxation procedures is adapted to the spline degree. A simple multigrid V-cycle with only one step of pre-smoothing results in a very efficient algorithm, whose convergence is independent on the spline degree and the spatial discretization parameter. Local Fourier analysis is shown to be very useful for the understanding of the problems encountered in the design of a robust multigrid method for IGA, and it is performed to support the good convergence properties of the proposed solver. In fact, an analysis for any spline degree and an arbitrary size of the blocks within the Schwarz smoother is presented for the one-dimensional case. The efficiency of the solver is also demonstrated through several numerical experiments, including a two-dimensional problem on a nontrivial computational domain.

**Key words.** isogeometric analysis, multigrid methods, local Fourier analysis, overlapping multiplicative Schwarz methods

**AMS subject classifications.** 65F10, 65M22, 65M55

**DOI.** 10.1137/18M1194407

**1. Introduction.** Isogeometric analysis (IGA) is a computational technique for the numerical solution of partial differential equations (PDEs), which was introduced by Hughes, Cottrell, and Bazilevs in the seminal paper [22]. Since then, this approach has been widely applied within different frameworks, and a detailed presentation of IGA together with a number of engineering applications can be found in the book [8]. IGA is based on the idea of using spline-type functions, which are exploited in computer aided design (CAD) software for the parametrization of the computational domain, in order to approximate the unknown solution of the PDE. B-splines or nonuniform rational B-splines (NURBS) are the most commonly used functions. There are several issues that make this approach advantageous over classical finite element methods (FEMs). First, it allows us to represent exactly some geometries like conic sections, and also more complicated geometries are represented more accurately by this technique than by traditional polynomial based approaches. In addition, this precise description of the geometry is incorporated exactly at the coarsest grid level, making unnecessary further communication with the CAD system in order to do a mesh refinement procedure, which moreover does not modify the geometry. Another important

---

\*Received by the editors June 15, 2018; accepted for publication (in revised form) June 5, 2019; published electronically October 29, 2019.

<https://doi.org/10.1137/18M1194407>

**Funding:** The work of the first author was supported by the grant program of the Diputacion General de Aragon. The work of the second author was partially supported by the Spanish project FEDER/MCYT MTM2016-75139-R and the Diputacion General de Aragon (Grupo de referencia APEDIF, E24.17R). The work of the third author was supported by the European Union's Horizon 2020 research and innovation program under Marie Skłodowska-Curie grant agreement 705402, POROSOS.

<sup>†</sup>IUMA and Applied Mathematics Department, University of Zaragoza, Zaragoza, Spain (apedela.riva@gmail.com, carmenr@unizar.es).

<sup>‡</sup>CWI, Centrum Wiskunde and Informatica, Amsterdam, The Netherlands (gaspar@cwi.nl, <http://www.unizar.es/pde/fjgaspar/>).

advantage is the higher continuity, since IGA provides up to  $C^{p-1}$  interelement continuity, denoting  $p$  the polynomial order (see [7, 10]). This corresponds to the so-called isogeometric  $k$ -method, which is one of the three refinement strategies for IGA proposed in [22], together with the  $h$ -refinement (reducing the mesh size by knot insertion) and  $p$ -refinement (order elevation, i.e., increase of the spline degree). The  $k$ -refinement is unique to IGA and its main advantage is that it maintains the maximum possible smoothness  $C^{p-1}$  for the spline space of degree  $p$ . Due to its high performance, this is the most popular refinement strategy in the IGA community, and this is the one studied in this work.

From the computational point of view, the efficient solution of the linear systems arising from the discretization of a PDE problem is a crucial point for its numerical simulation. When a discretization with high spline degrees is considered, this issue is even more challenging since the condition number of the stiffness matrices grows exponentially with the spline degree. The study of the computational efficiency for direct and iterative solvers was initiated in the papers [5, 6], respectively, and recently the design of iterative solvers has attracted much attention in the isogeometric community. Many efforts have been devoted to develop efficient solvers for this type of discretizations. For example, in [4] a multilevel BPX-preconditioner is developed in the framework of IGA. Beirão da Veiga et al. analyze overlapping Schwarz methods for IGA in [11], whereas in [12] they study BDDC preconditioners by introducing appropriate discrete norms. In [16] algebraic multilevel iteration methods are applied for the isogeometric discretization of scalar second order elliptic problems. The new isogeometric tearing and interconnecting method, which consists of a domain decomposition solver based on the ideas of the finite element tearing/interconnecting method, was proposed in [23]. In [27], the authors propose preconditioners based on fast solvers for the Sylvester equation. In all these works, the difficulty in achieving both robustness and computational efficiency for high-order isogeometric discretizations is reported.

For classical finite element, finite difference, or finite volume discretizations, multigrid methods [2, 17, 29] are well known to be among the fastest solvers showing optimal computational cost and convergence behavior. Thus, it seems natural to try to extend these methods to IGA, and in fact, in the early IGA literature, multigrid solvers for FEMs have been directly transferred to isogeometric discretizations with only minimal adaptations. However, a naive application of these multigrid methods to the isogeometric case results in an important deterioration of the convergence of the algorithms when the spline degree is increased. In particular, multigrid methods based on standard smoothers, like the Gauss–Seidel smoother, are not robust with respect to the spline degree (see, e.g., [15]). It was observed in [14] that the spectral radius of the multigrid iteration matrices based on standard smoothers tends to one exponentially as  $p$  increases. As it was pointed out in [13], this bad behaviour is due to the presence of many small eigenvalues associated with high-frequency eigenvectors. This deterioration of the convergence of standard multigrid algorithms has motivated advances toward robust multigrid methods with respect to the spline degree. In [14] a multigrid method was constructed based on a preconditioned Krylov smoother at the finest level and in [20] a mass matrix was proposed as a smoother within a multigrid framework. For both methods, an increase in the number of smoothing steps was needed in order to obtain robustness with respect to the spline degree. To avoid the lack of robustness of the mass smoother, due to boundary effects, in [19] the authors introduce a boundary correction to that relaxation. In that work, it was not clear, however, how to extend this approach to three dimensions. To overcome this, in [18],

the authors propose a multigrid smoother based on an additive subspace correction technique, applying a different smoother to each of the subspaces. In the regular interior subspace they use a mass smoother, whereas in the other subspaces they consider relaxations which exploit the particular structure of the subspaces. The authors observe a dependence of the convergence of this method on the space dimension and on the geometry transformation. We would like to remark that in all previous works the multigrid method is designed on the parametric domain and it is applied as a preconditioner for solving the problem in general geometries. In this work, however, we aim to propose a robust and efficient geometric multigrid algorithm for directly solving isogeometric discretizations on a general domain. This multigrid method is based on a family of overlapping multiplicative Schwarz-type methods as smoothers. We will show that by choosing an appropriate Schwarz-type smoother we can efficiently remove those high-frequency components of the error associated with the small eigenvalues. This makes it possible to obtain a very simple and efficient solver by using a multigrid  $V(1, 0)$ -cycle.

It is well known that many details are open for discussion and decision in the design of a multigrid method for a target problem, since the performance of multigrid algorithms strongly depends on the choice of their components. There are no rules to facilitate this challenging task, but the local Fourier analysis (LFA) appears as a very useful tool for the design of the algorithm. The LFA or local mode analysis was introduced by Achi Brandt in [2, 3], and since then, it has become the main quantitative analysis for the convergence of multigrid algorithms. This analysis is based on the Fourier transform theory, and a good introduction can be found in [28, 29, 30] and in the LFA monograph [31]. In particular, the application of LFA to analyze the smoothing properties of multiplicative Schwarz-type smoothers is not standard, and a special treatment is needed; see [24, 26]. LFA has not been widely applied for the multigrid solution of isogeometric discretizations. This can be due to the fact that in [21] the authors showed that in some cases the predictions of LFA did not match the real asymptotic convergence factor of multigrid based on Richardson relaxation for IGA as  $p$  increases due to boundary effects. We have seen, however, that this is not the case when standard Jacobi and Gauss–Seidel smoothers are considered, nor for the multiplicative Schwarz relaxations considered in this work. Thus, we want to emphasize that this is the first time that an LFA is successfully used to predict the multigrid convergence for IGA, showing the utility of this analysis for the design of solution methods for IGA. In this work, LFA in one and two dimensions is performed for isogeometric discretizations of the Poisson problem. Moreover, in particular, for the one-dimensional case, we provide an analysis for any spline degree and an arbitrary size of the blocks in the smoother. In this way, we can choose the suitable size of the blocks to perform in the Schwarz smoother depending on the spline degree.

The rest of the paper is structured as follows. In section 2 the considered model problem is described, together with the basics of IGA. Section 3 introduces the proposed multigrid method, giving special emphasis to the description of the class of multiplicative Schwarz methods used as smoothers. Moreover, at the end of this section, the computational cost of these relaxations is derived in terms of the block-size. The LFA technique considered for the study of the convergence of the multigrid algorithm is introduced in section 4. The basics of LFA are included in this section, as well as the nonstandard analysis necessary to study the proposed smoothers. At the end of section 4, some results obtained from the LFA are presented. Section 5 is devoted to choosing the most efficient Schwarz smoother for each spline degree. For this purpose, the computational cost derived in section 3 is combined with the

convergence factors predicted by LFA to obtain the most efficient smoothing strategy. Section 6 deals with the numerical results. One- and two-dimensional tests are presented, including an example with a complex geometry to show the real power of the isogeometric solver. Finally, in section 7 some conclusions are drawn.

**2. Preliminaries.** We consider the Poisson problem in  $d$  spatial dimensions on the domain  $\Omega = (0, 1)^d$  with homogeneous Dirichlet boundary conditions,

$$(1) \quad \begin{aligned} -\Delta u &= f & \text{in } \Omega, \\ u &= 0 & \text{on } \partial\Omega. \end{aligned}$$

The variational formulation of problem (1) is given by the following: find  $u \in H_0^1(\Omega)$  such that

$$a(u, v) = (f, v) \quad \forall v \in H_0^1(\Omega),$$

where

$$a(u, v) = \int_{\Omega} \nabla u \cdot \nabla v \, dx \quad \text{and} \quad (f, v) = \int_{\Omega} f v \, dx.$$

Given a finite dimensional approximation space  $V_h \subset H_0^1(\Omega)$ , the Galerkin approximation of the variational problem reads as follows: find  $u_h \in V_h$  such that

$$(2) \quad a(u_h, v_h) = (f, v_h) \quad \forall v_h \in V_h.$$

If we fix a basis  $\{\varphi_1, \dots, \varphi_{n_h}\}$  for  $V_h$ ,  $\dim V_h = n_h$ , then the solution of problem (2) can be written as  $u_h = \sum_{i=1}^{n_h} u_i \varphi_i$ . The coefficient vector  $\mathbf{u} = (u_1, \dots, u_{n_h})$  can be computed by solving the linear system  $\mathbf{A}\mathbf{u} = \mathbf{b}$ , where  $\mathbf{A}$  is the stiffness matrix obtained from the bilinear form  $a(\cdot, \cdot)$ , i.e.,  $A = (a_{i,j}) = (a(\varphi_j, \varphi_i))_{i,j=1}^{n_h}$ , and  $\mathbf{b} = (f, \varphi_i)_{i=1}^{n_h}$  is the right-hand-side vector. In FEMs, the chosen approximation space  $V_h$  is usually a space of continuous piecewise polynomials, whereas in the IGA  $V_h$  is a space of functions with higher continuity (up to order  $p-1$ ,  $p$  being the polynomial degree of the B-spline). In this work, the solution of the variational formulation of problem (1) is approximated in a spline space of degree  $p$  with maximum smoothness, i.e., with  $C^{p-1}$  regularity.

First, let us consider the simplest case when  $d = 1$  (one-dimensional problem). The computational domain then is the interval  $\Omega = (0, 1)$  and the corresponding two-point boundary value problem is

$$-u''(x) = f(x), \quad x \in \Omega, \quad u(0) = u(1) = 0.$$

Let the interval  $(0, 1)$  be subdivided into  $m \in \mathbb{N}$  subintervals  $I_i = ((i-1)h, ih)$ ,  $i = 1, \dots, m$ , with  $h = 1/m$ , and consider the knot sequence

$$(3) \quad \Xi_{p,h} = \{\xi_1 = \dots = \xi_{p+1} = 0 < \xi_{p+2} < \dots < \xi_{p+m} < 1 = \xi_{p+m+1} = \dots = \xi_{2p+m+1}\},$$

where  $\xi_{p+i+1} = i/m$ ,  $i = 0, \dots, m$ . We define the spline space of degree  $p \geq 1$  with maximum continuity

$$(4) \quad \mathcal{S}_{p,h}(0, 1) = \{u_h \in C^{p-1}(0, 1) : u_h|_{I_i} \in \mathbb{P}^p, i = 1, \dots, m, u_h(0) = u_h(1) = 0\},$$

where  $C^{p-1}(0, 1)$  is the space of all  $p-1$  times continuously differentiable functions on  $(0, 1)$ , and  $\mathbb{P}^p$  is the space of all polynomials of degree less than or equal to  $p$ . The dimension of the space  $\mathcal{S}_{p,h}(0, 1)$  is  $p+m-2$ , and the set of basis functions  $\{N_i^p\}_{i=2}^{p+m-1}$  of this space is defined recursively by the Cox-de-Boor formula (see [9]), starting with  $p = 0$  (piecewise constants).

DEFINITION 1 (basis functions). Given the knot sequence  $\Xi_{p,h}$  as in (3), the  $i$ th piecewise constant B-spline basis function  $N_i^0 : [0, 1] \rightarrow \mathbb{R}$ ,  $i = 1, \dots, m + 2p$ , is defined as

$$(5) \quad N_i^0(\xi) = \begin{cases} 1 & \text{if } \xi_i \leq \xi < \xi_{i+1}, \\ 0 & \text{otherwise.} \end{cases}$$

For every pair  $(k, i)$  such that  $1 \leq k \leq p$ ,  $1 \leq i \leq m + 2p - k$ , the basis functions  $N_i^k : [0, 1] \rightarrow \mathbb{R}$  are given recursively by the Cox-de-Boor formula:

$$(6) \quad N_i^k(\xi) = \frac{\xi - \xi_i}{\xi_{i+k} - \xi_i} N_i^{k-1}(\xi) + \frac{\xi_{i+k+1} - \xi}{\xi_{i+k+1} - \xi_{i+1}} N_{i+1}^{k-1}(\xi).$$

Remark. Note that the previous formula can yield quotients of the form  $0/0$  if there are repeated knots, that is,  $\xi_i = \xi_{i+k}$  or  $\xi_{i+1} = \xi_{i+k+1}$ . In these cases, the denominator is clearly equal to zero and the support of the corresponding B-spline basis functions with polynomial degree  $k - 1$  is empty, making the numerator equal to zero too. Fractions of the form  $0/0$  are considered as zero.

In the case of higher spatial dimensions  $d > 1$ , the previous definitions are easily generalized by means of tensor product. For simplicity, we assume that the spline degree  $p$  and the number of subintervals  $m$  are the same in all directions. However, this is not restrictive for the solver proposed in this work. In this way, over the domain  $\Omega = (0, 1)^d$  we define the spline space

$$\mathcal{S}_{p,h}(\Omega) = \underbrace{\mathcal{S}_{p,h}(0, 1) \otimes \dots \otimes \mathcal{S}_{p,h}(0, 1)}_d.$$

In the two-dimensional case,  $d = 2$ , the knot vector

$$\Xi_{p,h} \times \Xi_{p,h} = \{(\xi, \eta), \xi \in \Xi_{p,h}, \eta \in \Xi_{p,h}\}$$

generates a mesh of rectangular elements in the parametric space. The spline space is  $\mathcal{S}_{p,h}(0, 1)^2 = \mathcal{S}_{p,h}(0, 1) \otimes \mathcal{S}_{p,h}(0, 1)$  and the bivariate B-splines basis is constructed by the tensor product of univariate B-splines basis. In this way a basis function  $N_{i,j}^p : [0, 1]^2 \rightarrow \mathbb{R}$  is defined in terms of the univariate basis functions  $N_i^p, N_j^p : [0, 1] \rightarrow \mathbb{R}$  as follows:

$$N_{i,j}^p(\xi, \eta) = (N_i^p \otimes N_j^p)(\xi, \eta) = N_i^p(\xi)N_j^p(\eta).$$

In particular, for our model problem we can write

$$\mathcal{S}_{p,h}(0, 1)^2 = \text{span} \{N_{i,j}^p(\xi, \eta), i, j = 1, \dots, p + m - 2\}.$$

All the ideas presented here can be extended to solve PDEs in more complicated domains  $\Omega$ . In the IGA,  $\Omega$  is usually given by a NURBS parametrization. In this way, conic sections, such as circles and ellipses, can be represented exactly. NURBS are built through rational functions of B-splines. A NURBS basis function of degree  $p$  is

$$R_i^p(\xi) = \frac{\omega_i N_i^p(\xi)}{\sum_{k=1}^{p+m} \omega_k N_k^p(\xi)},$$

where  $\{\omega_1, \dots, \omega_{p+m}\}$  is a given set of weights. In the two-dimensional case, NURBS basis functions are defined as

$$R_{i,j}^p(\xi, \eta) = \frac{\omega_{i,j} N_{i,j}^p(\xi, \eta)}{\sum_{k,l=1}^{p+m} \omega_{k,l} N_{k,l}^p(\xi, \eta)},$$

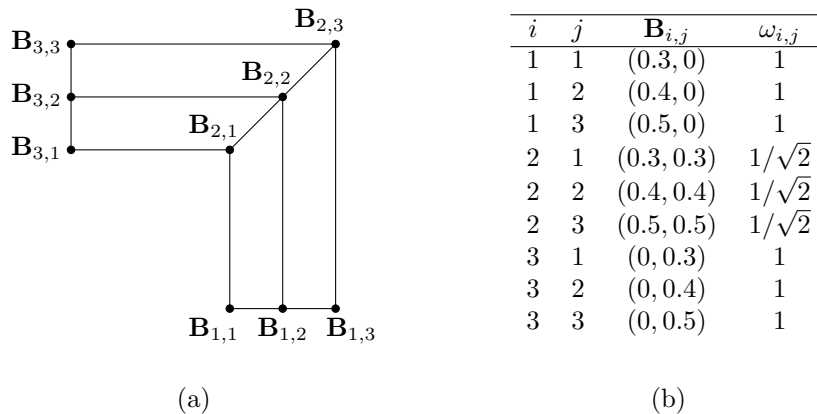


FIG. 1. Example of quadratic NURBS transformation of a quarter of an annulus with small radius  $r = 0.3$  and large radius  $R = 0.5$ : (a) control mesh and (b) control points and their corresponding weights.

and the domain  $\Omega$  is represented by a geometry transformation  $\mathbf{F} : (0, 1)^2 \rightarrow \Omega$  given by

$$\mathbf{F}(\xi, \eta) = \sum_{i=1}^{p+m} \sum_{j=1}^{p+m} \mathbf{B}_{i,j} R_{i,j}^p(\xi, \eta).$$

Thus, a transformation  $\mathbf{F}$  is determined by the so-called control points  $\mathbf{B}_{i,j}$  and their associated weights  $\omega_{i,j}$ . In order to illustrate this type of transformations, we introduce an example of a quadratic NURBS parametrization of a quarter annulus domain, which will be used later for one of our numerical experiments. In Figure 1(a), we show the control mesh needed for mapping this physical domain. Since these transformations are based on noninterpolatory spline basis functions, the control points can be interpreted as a scaffold for the physical mesh. In Figure 1(b) we provide the control points together with the corresponding weights.

Now, the basis functions on  $\Omega$  are defined by composing the basis functions on the parametric domain with the inverse of the geometry transformation, that is,  $R_{i,j} \circ \mathbf{F}^{-1}$ , and therefore the finite dimensional approximation space in our model problem is

$$V_h = \text{span} \{ R_{i,j} \circ \mathbf{F}^{-1}, i, j = 1, \dots, p + m - 2 \}.$$

**3. Multigrid method.** Multigrid methods are based on the smoothing property of a classical iterative algorithm and the acceleration of its convergence by a coarse-grid correction technique. In particular, the first step is to approximate the solution of the system on the fine space  $V_h$  by using the classical iterative method  $S_h$ , which within the multigrid community is called a smoother or relaxation procedure. Given an approximation of the solution  $u_h^m$ , if  $\nu_1$  presmoothing steps are applied, it is obtained that  $\bar{u}_h^m = S_h^{\nu_1} u_h^m$ . The smoother is in charge of annihilating the high-frequency components of the error, yielding a smooth error which can be well represented on a coarser space  $V_{2h}$ , where the computations are much cheaper. Notice that, as usual, we denote the coarse space with the subscript  $2h$  since standard coarsening is assumed. After the relaxation step, the residual  $\bar{r}_h^m = f_h - A_h \bar{u}_h^m$  is transferred to the coarse space,  $\bar{r}_{2h}^m = I_h^{2h} \bar{r}_h^m$ , by means of an appropriate restriction operator  $I_h^{2h}$ . There, the defect equation

$$(7) \quad A_{2h} \widehat{e}_{2h}^m = \bar{r}_{2h}^m$$

must be solved. The error correction obtained on the coarse grid is then transferred back to the fine grid  $\widehat{e}_h^m = I_{2h}^h \widehat{e}_{2h}^m$  by using a suitable prolongation operator  $I_{2h}^h$ . This value is used to correct the previous approximate solution and to obtain the new approximation  $\bar{u}_h^{m+1} = \bar{u}_h^m + \widehat{e}_h^m$  to which we can apply  $\nu_2$  smoothing iterations,  $u_h^{m+1} = S_h^{\nu_2} \bar{u}_h^{m+1}$ . When this algorithm is applied recursively on the coarse level to solve the defect equation (7) we obtain the multigrid method. In our case, we first consider a coarsest B-spline space  $V_{h_0}$  of order  $p$ , which is obtained by a  $k$ -refinement strategy, and then we build a sequence  $V_{h_0} \subset V_{h_1} \subset \dots \subset V_{h_\ell}$  of nested B-spline spaces that have been obtained through uniform and global  $h$ -refinement by knot insertion, where  $V_{h_\ell}$  is our target space. The procedure of knot insertion not only allows for B-spline refinement, but it also provides the way of building the intergrid transfer operators associated with the multigrid method. In this way, we consider the canonical spline embedding operator as the prolongation operators  $I_{h_{k-1}}^{h_k}$  and their adjoint for the restriction operators  $I_{h_k}^{h_{k-1}}$  for  $k = 1, \dots, \ell$ . Then, the coarse-grid operators will be constructed by Galerkin approximation. Regarding the type of cycle, we will demonstrate that  $V$ -cycles perform similarly as  $W$ -cycles, and therefore the former ones will be preferred. Moreover, we will also see that one smoothing step of a multiplicative Schwarz relaxation will be enough to obtain a very efficient and robust multigrid for IGA.

It is well known that multigrid methods based on simple pointwise smoothers such as Gauss–Seidel relaxation do not behave well when applied to isogeometric discretizations with larger values of  $p$ . In order to see this behavior we consider the one-dimensional model problem (1). In Figure 2(a), the number of iterations necessary to reduce the residual until  $10^{-10}$  are displayed for different values of  $p$  from 2 to 8. In this picture, we do not represent all the iteration markers on the convergence lines so that the different markers can be distinguished. Notice that the convergence for  $p = 2$  and  $p = 3$  is very similar and the corresponding convergence lines overlap and almost cannot be distinguished. The deterioration of the multigrid convergence is clear as soon as  $p$  becomes larger. This implies that such a multigrid method is not reliable for large  $p$ , although its convergence is independent on the spatial discretization parameter for a fixed value of  $p$ . This latter can be seen in Figure 2(b), where the history of the multigrid convergence is shown for  $p = 4$  with different target grids.

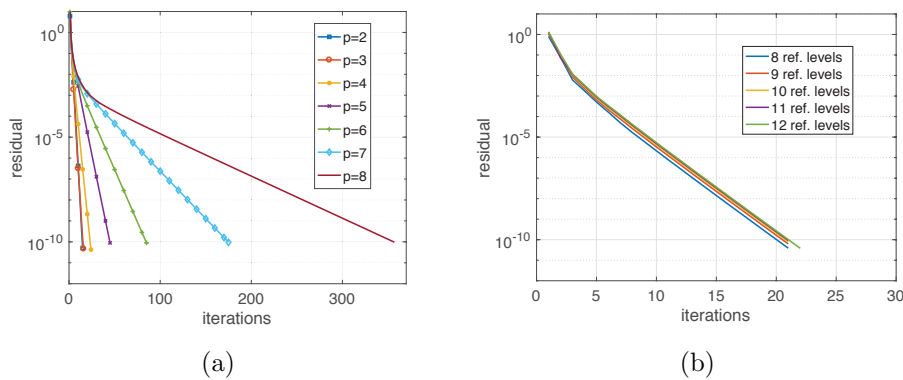


FIG. 2. History of the convergence of a  $V(1,0)$ -multigrid method based on Gauss–Seidel relaxation for (a) different values of order  $p$  and (b) a fixed value  $p = 4$  and different grid-sizes.

Pointwise smoothers can be generalized to blockwise iterations by updating simultaneously a set of unknowns at each time, instead of only one. This is done by splitting the grid into blocks and solving together the equations corresponding to the grid-points in each block. There are many possibilities to construct these blocks. One can allow the blocks to overlap, giving rise to the class of overlapping block iterations, where smaller local problems are solved and combined via an additive or multiplicative Schwarz method.

More specifically, the description of the multiplicative Schwarz iteration applied to the system  $A\mathbf{u} = \mathbf{b}$  of order  $n_h$  (see section 2) is as follows. Let us denote as  $B$  the subset of unknowns involved in an arbitrary block of size  $n^d$ , that is,  $B = \{u_{k_1}, \dots, u_{k_{n^d}}\}$ , where  $k_i$  is the global index of the  $i$ th unknown in the block. In order to construct the matrix to solve associated with such a block, that is,  $A^B$ , we consider the projection operator from the vector of unknowns  $\mathbf{u}$  to the vector of unknowns involved in the block. This results in a matrix  $V_B$  of size  $(n^d \times n_h)$ , whose  $i$ th row is the  $k_i$ th row of the identity matrix of order  $n_h$ . Thus, matrix  $A^B$  is obtained as  $A^B = V_B A V_B^T$ , and the iteration matrix of the multiplicative Schwarz method can be written as

$$\prod_{B=1}^{NB} (I - V_B^T (A^B)^{-1} V_B A),$$

where  $NB$  denotes the number of blocks obtained from the splitting of the grid, which corresponds to the number of small systems that have to be solved in a relaxation step of the multiplicative Schwarz smoother.

Here we consider multiplicative Schwarz methods with maximum overlapping. In the one-dimensional case, our smoother will be based on blocks of three, five, or seven points, depending on the spline degree  $p$ . Our study will be carried out up to  $p = 8$ , but if one is interested in solving isogeometric discretizations with spline degree larger than  $p = 8$ , it is only necessary to find the appropriate number of unknowns involved in the blocks to obtain an efficient multigrid approach. In the one-dimensional case, the block of size  $n$  associated with the  $i$ th unknown is composed of the unknown  $u_i$ , the  $(n-1)/2$  unknowns on the right side of  $u_i$  and the  $(n-1)/2$  unknowns on its left side. More concretely, for the three-point multiplicative Schwarz smoother  $B = \{u_{i-1}, u_i, u_{i+1}\}$ , whereas for the seven-point relaxation, for example, we have  $B = \{u_{i-3}, u_{i-2}, u_{i-1}, u_i, u_{i+1}, u_{i+2}, u_{i+3}\}$ . In Figure 3 the corresponding blocks and the overlapping between them are schematized for the three-, five-, and seven-point multiplicative Schwarz smoothers (Figures 3(a), (b), and (c), respectively). In the two-dimensional case, we use the same idea, that is, we choose square blocks of size  $n \times n$  around the unknown  $u_{i,j}$ , yielding, for example, to the block

$$B = \{u_{i-1,j-1}, u_{i,j-1}, u_{i+1,j-1}, u_{i-1,j}, u_{i,j}, u_{i+1,j}, u_{i-1,j+1}, u_{i,j+1}, u_{i+1,j+1}\}$$

for the nine-point multiplicative Schwarz smoother ( $n = 3$ ). In the same way, we will also consider the 25- and 49-point multiplicative Schwarz smoothers.

In order to choose an appropriate multiplicative Schwarz smoother for each spline degree  $p$ , it is important to take into account the computational cost of these relaxations, as well as the convergence factor that the multigrid method based on these methods provides. In the next section, we will introduce a local Fourier analysis that will help to predict such convergence rates and will give a very good insight about why these relaxations work well. The computational cost of the considered multiplicative Schwarz smoothers is studied next.

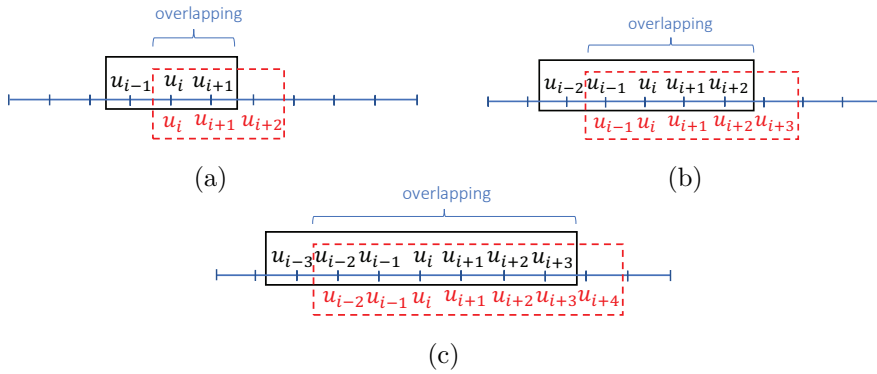


FIG. 3. Size and overlapping of the blocks for the one-dimensional smoothers: (a) three-, (b) five-, and (c) seven-point multiplicative Schwarz iterations.

We would like to study the computational complexity of applying the proposed block smoothers in a general  $d$ -dimensional setting. Since we consider a multiplicative Schwarz smoother with maximum overlapping, small dense systems

$$(8) \quad A^B \delta u_B = d_B,$$

of size  $n^d$ , where  $d_B$  is the local defect and  $\delta u_B$  is the correction, have to be solved for each degree of freedom. The computationally most expensive parts of the smoothing iteration are the matrix-vector multiplication required to calculate the local residual  $d_B$  and the solution of dense system (8). By using a standard approach, the cost of the calculation of each local defect is  $\mathcal{O}(n^{2d})$ . By exploiting the tensor product structure of the B-spline basis functions, sum-factorization [25] can be applied and the cost can be reduced to  $\mathcal{O}(dn^{d+1})$ . The cost of solving system (8) depends on the chosen method. Assuming that the inverse of matrix  $A^B$  is stored or the resulting  $L$  and  $U$  factors are stored in a block  $LU$ -factorization, the cost of calculating the correction is  $\mathcal{O}(n^{2d})$ . Direct solvers such as Gauss elimination methods require a computational cost of order  $\mathcal{O}(n^{3d})$ . As pointed out in [1], if Krylov-subspace methods such as the conjugate gradient iteration are chosen for solving system (8), and if sum-factorization techniques are used, the global complexity is reduced to  $\mathcal{O}(dit(\varepsilon)n^{d+1})$ , where  $it(\varepsilon)$  is the number of iterations required to solve the system to accuracy  $\varepsilon$  (see [1] for more details).

In this work, we have applied sum-factorization to calculate the residual, and the solution of the small dense systems is done by using  $LU$ -factorization. Therefore, the global computational cost of the multiplicative Schwarz smoother per iteration is  $\mathcal{O}(N_{DOF} n^{2d})$ , where  $N_{DOF}$  is the total number of degrees of freedom. In addition, we have to take into account that the  $LU$ -factorization for solving the small dense systems is computed in a setup phase. In the case in which no transformation is necessary to deal with the geometry of the computational domain, the cost of this setup phase is negligible since the factorization is the same for most degrees of freedom. For nontrivial geometries, this factorization, however, is computed for every degree of freedom and then the setup phase increases the computational cost of the algorithm.

**4. Local Fourier analysis.** In this section we introduce the basics of the LFA and describe its application to IGA. First we will apply LFA to understand the difficulties in designing an appropriate multigrid for IGA and how this tool gives us

insight to achieve this challenging task. After that, we will introduce LFA for overlapping multiplicative Schwarz smoothers. For simplicity in the presentation, we will describe the analysis for the one-dimensional problem. This analysis, however, has been performed for the two-dimensional case too. In fact, in the LFA results section, estimates for both the one- and two-dimensional cases are provided.

**4.1. Basics of LFA.** The main idea of LFA is to assume a decomposition of the error function in Fourier modes, and then to study the behavior of each operator involved in the multigrid method on these components. Some assumptions should be considered to perform this analysis. In particular, LFA presumes that all the operations involved in the multigrid algorithm are local processes neglecting the effect of boundary conditions. Then, a regular infinite grid,  $\mathcal{G}_h$ , is assumed, which is obtained by the infinite extension of the considered spatial grid. By imposing some assumptions on the discrete operator  $A_h$ , as linearity and constant coefficients, the rows of the system matrix corresponding to degrees of freedom in the interior of the domain may have a natural Toeplitz or multilevel Toeplitz structure. The key idea of LFA is to ignore the effect of boundary conditions, which would result in a set of rows with quite different values, and to extend the operator from the interior of the domain to the uniform infinite grid  $\mathcal{G}_h$ . On such a grid, the discrete operator is represented as an infinite-grid (multilevel) Toeplitz matrix, which can be diagonalized by the matrix of Fourier modes. These are complex exponential eigenfunctions of the operator defined as  $\varphi_h(\theta, x) = e^{i\theta x/h}$ , with  $x \in \mathcal{G}_h$  and where  $\theta \in \Theta_h := (-\pi, \pi]$ , yielding the so-called Fourier space  $\mathcal{F}(\mathcal{G}_h) := \text{span}\{\varphi_h(\theta, x) \mid \theta \in \Theta_h\}$ . These Fourier components are divided into high- and low-frequency components on  $\mathcal{G}_h$ . We call low-frequency components to those Fourier modes associated with frequencies belonging to  $\Theta_{2h} = (-\pi/2, \pi/2]$  (low frequencies), and the high-frequency components are those corresponding to the high frequencies  $\theta \in \Theta_h \setminus \Theta_{2h}$ . This classification depends on the coarsening strategy that we consider, which in this case is standard coarsening, that is, the coarse-grid step size is obtained by doubling the step size of the fine grid.

Under the previous assumptions on the discrete operator, the Fourier components satisfy that  $A_h \varphi_h(\theta, x) = \tilde{A}_h(\theta) \varphi_h(\theta, x)$ . This means that the Fourier components are “eigenfunctions” of the discrete operator, and the corresponding “eigenvalues” give rise to the so-called Fourier symbol of the operator,  $\tilde{A}_h(\theta)$ . Notice that in our context,  $A_h$  denotes the stencil of the IGA variational form and satisfies the considered assumptions. As an example, the one-dimensional IGA discrete Laplace operator with  $p = 2$  is given in stencil form (see [29] for stencil notation),

$$A_{h,2} = \frac{1}{h} \left[ -\frac{1}{6}, -\frac{1}{3}, 1, -\frac{1}{3}, -\frac{1}{6} \right],$$

and its Fourier symbol is given by

$$\tilde{A}_{h,2}(\theta) = \frac{1}{h} \left( 1 - \frac{2 \cos(\theta)}{3} - \frac{2 \cos(2\theta)}{6} \right) = \frac{2}{3h} (2 - \cos \theta (1 + \cos \theta)).$$

In Figure 4(a) the eigenvalues of  $A_{h,2}$  are displayed, together with those corresponding to the symbols of operators  $A_{h,5}$  and  $A_{h,8}$ , that is, the discrete operators associated with  $p = 5$  and  $p = 8$ , respectively. We can observe an important difference between the eigenvalues of  $A_{h,2}$  and those for  $A_{h,5}$  and  $A_{h,8}$ . Opposite to the case  $p = 2$ , when  $p = 5$  or  $p = 8$ , there are many small eigenvalues associated

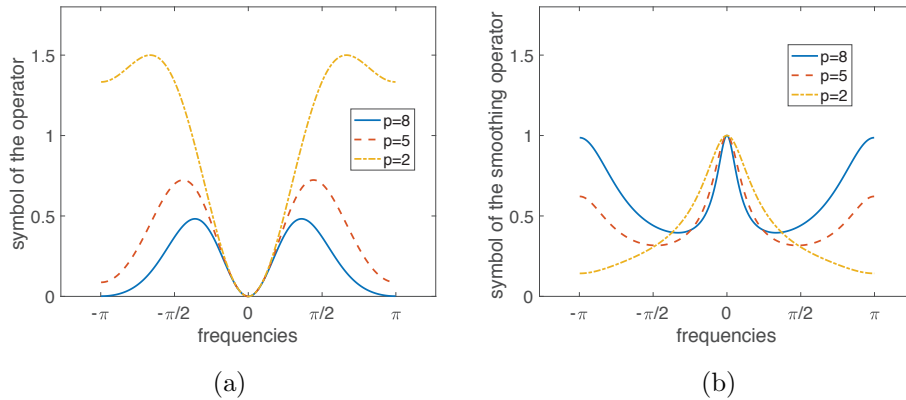


FIG. 4. Symbol of (a) the discrete operator and (b) the Gauss–Seidel smoothing operator, for three different spline degrees  $p = 2, 5, 8$ .

with high-frequency eigenvectors. This was also pointed out in [13], and it slows down the convergence of classical multigrid methods since a standard smoother cannot provide an efficient damping of the components of the error associated with the high frequencies. In order to illustrate this, we analyze the Fourier representation of a Gauss–Seidel relaxation procedure. The Fourier components are also eigenfunctions of the smoothing iteration matrix  $S_h$ , which relates the error grid functions at two consecutive relaxation iterations. This means that we can obtain the error amplification factor or Fourier symbol of  $S_h$  with respect to each frequency  $\theta$ , that is,  $\tilde{S}_h(\theta)$  (see [29, 31] for basic details). This is shown in Figure 4(b) for a lexicographic Gauss–Seidel smoother and the three different values of  $p$  previously considered. It can be observed how this classical relaxation fails to annihilate the high-frequency components for  $p = 5$  and  $p = 8$ , whereas it effectively reduces such Fourier modes for  $p = 2$ . This is the reason why the careless application of a standard multigrid to IGA yields a deterioration of the convergence as the spline degree gets larger, as we previously reported in Figure 2(a). The translation of the smoothing property, that is, the capacity of the smoother to eliminate the high-frequency components of the error, into a quantitative measure is the so-called smoothing factor, which is given as follows:

$$\mu = \sup_{\Theta_h \setminus \Theta_{2h}} \left| \tilde{S}_h(\theta) \right|.$$

This measure, however, is not enough to analyze the interplay between the smoother and the coarse-grid correction technique that accelerates its convergence. In order to get more insight into the behavior of the multigrid method, taking into account the influence of the intergrid transfer operators and the other components involved in the coarse-grid correction, at least a two-grid analysis is needed. The two-grid analysis consists of estimating the spectral radius of the two-grid operator. This error propagation operator is given as

$$M_h^{2h} = S_h^{\nu_2} (I_h - I_{2h}^h A_{2h}^{-1} I_h^{2h} A_h) S_h^{\nu_1},$$

where  $A_h$  and  $A_{2h}$  represent the discrete operators in the fine and coarse grids,  $I_h^{2h}$  and  $I_{2h}^h$  denote the restriction and prolongation operators, respectively, and  $\nu_1$  and  $\nu_2$  are the numbers of pre- and postsmoothing steps. It is well known that the intergrid

transfer operators, and as a consequence the two-grid operator, couple some Fourier components. In particular, in the transition between fine and coarse grids, each low-frequency  $\theta^0 \in \Theta_{2h}$  is coupled with a high-frequency  $\theta^1 = \theta^0 - \text{sign}(\theta^0)\pi$ , giving rise to the so-called spaces of  $2h$ -harmonics:  $\mathcal{F}^2(\theta^0) = \text{span}\{\varphi_h(\theta^0, x), \varphi_h(\theta^1, x)\}$ . This implies that the Fourier representation of the two-grid operator  $M_h^{2h}$  with respect to  $\mathcal{F}^2(\theta^0)$ , denoted by  $\widetilde{M}_h^{2h}(\theta^0)$ , is a  $2 \times 2$ -matrix. We determine the spectral radius of  $M_h^{2h}$  by calculating the spectral radius of these smaller matrices, that is,

$$\rho_{2g} = \rho(M_h^{2h}) = \sup_{\theta^0 \in \Theta_{2h}} \rho(\widetilde{M}_h^{2h}(\theta^0)).$$

Although the two-grid analysis is the basis for the classical asymptotic multigrid convergence estimates, a three-grid (or even a  $k$ -grid) analysis provides a deeper insight into the performance of multigrid. This analysis is crucial, for example, to study the behavior of  $V$ -cycles and the difference between the choice of pre- and postsmoothing steps. The error propagation matrix of a three-grid cycle (see, e.g., [31]) can be written as follows:

$$M_h^{4h} = S_h^{\nu_2} \left( I_h - I_{2h}^h \left( I_{2h} - (M_{2h}^{4h})^\gamma \right) A_{2h}^{-1} I_{2h}^{2h} A_h \right) S_h^{\nu_1},$$

where  $M_{2h}^{4h}$  is the two-grid operator between the two coarse grids, that is,

$$M_{2h}^{4h} = S_{2h}^{\nu_2} \left( I_{2h} - I_{4h}^{2h} A_{4h}^{-1} I_{2h}^{4h} A_{2h} \right) S_{2h}^{\nu_1},$$

and  $\gamma$  denotes the number of times that the latter two-grid operator is applied. This parameter  $\gamma$  is also called the cycle index since its value determines if  $V$ - or  $W$ -cycles are considered ( $\gamma = 1$  or  $\gamma = 2$ , respectively). In order to analyze how this three-grid operator acts on the Fourier modes, we take into account that not only in the transition from the finest to the second grid but also in the transition from the second to the coarsest grid there are some Fourier modes that are coupled. More concretely, four frequencies are coupled, and then we can define the so-called subspaces of  $4h$ -harmonics as  $\mathcal{F}^4(\theta^0) = \text{span}\{\varphi_h(\theta_\beta^\alpha) \mid \alpha, \beta \in \{0, 1\}\}$ , where  $\theta^0 \in \Theta_{4h} = (-\pi/4, \pi/4]$  and  $\theta_\beta^\alpha = \theta^0 - \alpha \text{sign}(\theta^0)\pi/2 + (-1)^{\alpha+\beta} \beta \text{sign}(\theta^0)\pi$ . Thus, based on the decomposition of the Fourier space in terms of the subspaces of  $4h$ -harmonics, we can reduce the computation of the spectral radius of the three-grid operator to the calculation of the supremum of the spectral radii of the  $4 \times 4$  Fourier representations on these subspaces,  $\widetilde{M}_h^{4h}(\theta^0)$ , that is,

$$\rho_{3g} = \rho(M_h^{4h}) = \sup_{\theta^0 \in \Theta_{4h}} \rho(\widetilde{M}_h^{4h}(\theta^0)).$$

This LFA can be applied in two dimensions as well. In particular, we study the Fourier symbol of the discrete Laplace operator  $A_{h,p}$  as we did before for the one-dimensional case. The results for three different spline degrees  $p = 2, 5, 8$  are shown in Figure 5, where again we can observe behavior similar to that obtained in one dimension. For  $p = 2$  the eigenvalues associated with the high frequencies are mainly large, whereas as  $p$  grows up there appear more and more small eigenvalues corresponding to the high-frequency components. This implies that a standard two-dimensional smoother won't be able to completely annihilate the high-frequency components of the error and consequently a simple multigrid method will deteriorate its convergence as  $p$  gets larger.

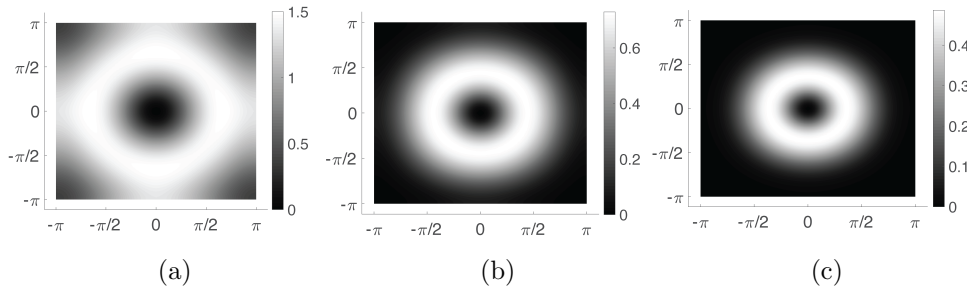


FIG. 5. Symbol of the two-dimensional discrete Laplace operator  $A_{h,p}$  for three different spline degrees: (a)  $p = 2$ , (b)  $p = 5$ , and (c)  $p = 8$ .

**4.2. Local Fourier analysis for overlapping smoothers.** In this section we describe the LFA for the class of overlapping multiplicative Schwarz smoothers considered in this work. With this purpose, we follow the methodology described in [26] and the notation therein. This type of multiplicative Schwarz smoothers satisfy the invariance property, in the sense that the Fourier modes are their eigenvectors (see [24] for a rigorous proof). The trouble, however, comes from their overlapping character. Due to this, the unknowns are updated more than once, which implies that, in addition to the initial and final errors, some intermediate errors appear, which have to be taken into account in the analysis. This makes necessary a special strategy to carry out the LFA for overlapping block smoothers. For simplicity in the presentation, we describe the analysis for the overlapping three-point multiplicative Schwarz smoother in one dimension applied to the isogeometric discretization with  $p = 2$ . At the end of this section we provide an expression to compute the symbol of the smoothing operator for any one-dimensional  $n$ -point multiplicative Schwarz relaxation for an arbitrary spline degree  $p$ .

For the considered case, at any grid-point  $\xi_i$ , three equations corresponding to the unknowns  $u_{i-1}$ ,  $u_i$ , and  $u_{i+1}$  are solved simultaneously. The block to solve, in terms of corrections and residuals, is given as follows:

$$(9) \quad \frac{1}{h} \begin{pmatrix} 1 & -1/3 & -1/6 \\ -1/3 & 1 & -1/3 \\ -1/6 & -1/3 & 1 \end{pmatrix} \begin{pmatrix} \delta u_{i-1} \\ \delta u_i \\ \delta u_{i+1} \end{pmatrix} = \begin{pmatrix} r_{i-1} \\ r_i \\ r_{i+1} \end{pmatrix}.$$

The corrections can be written in terms of the errors:

$$\begin{aligned} \delta u_{i-1} &= e_h^{k+1}(\xi_{i-1}) - e_h^{k+2/3}(\xi_{i-1}), \\ \delta u_i &= e_h^{k+2/3}(\xi_i) - e_h^{k+1/3}(\xi_i), \\ \delta u_{i+1} &= e_h^{k+1/3}(\xi_{i+1}) - e_h^k(\xi_{i+1}), \end{aligned}$$

where  $e_h^k$  denotes the initial error at  $k$ -iteration,  $e_h^{k+1}$  represents the final error, and we define as  $e_h^{k+1/3}$  and  $e_h^{k+2/3}$  the intermediate errors appearing after the unknown has been updated once or twice, respectively. Thus, when solving the  $i$ th block, for example, unknown  $u_{i-1}$  has been already relaxed twice, and it will be updated the final third time. Without loss of generality, we consider that the error is given as

a single Fourier mode multiplied by a coefficient  $\alpha_\theta^{(m)}$ , where  $m = 0, 1, 2, 3$  is the number of times that the unknown has been updated in the current iteration. In this way, we can rewrite system (9) as follows:

$$\begin{aligned} & \frac{1}{h} \begin{pmatrix} 1 & -1/3 & -1/6 \\ -1/3 & 1 & -1/3 \\ -1/6 & -1/3 & 1 \end{pmatrix} \begin{pmatrix} (\alpha_\theta^{(3)} - \alpha_\theta^{(2)}) e^{-i\theta} \\ (\alpha_\theta^{(2)} - \alpha_\theta^{(1)}) \\ (\alpha_\theta^{(1)} - \alpha_\theta^{(0)}) e^{i\theta} \end{pmatrix} \\ &= -\frac{1}{h} \begin{pmatrix} -\frac{1}{6}\alpha_\theta^{(3)} e^{-i3\theta} - \frac{1}{3}\alpha_\theta^{(3)} e^{-i2\theta} + \alpha_\theta^{(2)} e^{-i\theta} - \frac{1}{3}\alpha_\theta^{(1)} - \frac{1}{6}\alpha_\theta^{(0)} e^{i\theta} \\ -\frac{1}{6}\alpha_\theta^{(3)} e^{-i2\theta} - \frac{1}{3}\alpha_\theta^{(2)} e^{-i\theta} + \alpha_\theta^{(1)} - \frac{1}{3}\alpha_\theta^{(0)} e^{i\theta} - \frac{1}{6}\alpha_\theta^{(0)} e^{i2\theta} \\ -\frac{1}{6}\alpha_\theta^{(2)} e^{-i\theta} - \frac{1}{3}\alpha_\theta^{(1)} + \alpha_\theta^{(0)} e^{i\theta} - \frac{1}{3}\alpha_\theta^{(0)} e^{i2\theta} - \frac{1}{6}\alpha_\theta^{(0)} e^{i3\theta} \end{pmatrix}. \end{aligned}$$

Notice that the right-hand side in the previous expression is easily obtained writing the residual in terms of the error. Since our aim is to find the relation between the initial and the fully corrected errors, we rearrange the previous system into a system of equations for the updated coefficients, that is,

$$\underbrace{\begin{pmatrix} -\frac{1}{6}e^{i\theta} & -\frac{1}{3} & e^{-i\theta} - \frac{1}{3}e^{-i2\theta} - \frac{1}{6}e^{-i3\theta} \\ -\frac{1}{3}e^{i\theta} & 1 & -\frac{1}{3}e^{-i\theta} - \frac{1}{6}e^{-i2\theta} \\ e^{i\theta} & -\frac{1}{3} & -\frac{1}{6}e^{-i\theta} \end{pmatrix}}_P \begin{pmatrix} \alpha_\theta^{(1)} \\ \alpha_\theta^{(2)} \\ \alpha_\theta^{(3)} \end{pmatrix} = \underbrace{\begin{pmatrix} 0 \\ \frac{1}{6}e^{i2\theta} \\ \frac{1}{3}e^{i2\theta} + \frac{1}{6}e^{i3\theta} \end{pmatrix}}_Q \alpha_\theta^{(0)}.$$

The amplification factor for the error is given by the last component of  $P^{-1}Q$ , since this one represents how the Fourier coefficient of the fully corrected error  $\alpha_\theta^{(3)}$  is related with that of the initial error  $\alpha_\theta^{(0)}$ . Once that we have the symbol of the smoothing operator  $\tilde{S}_h(\theta) = (P^{-1}Q)_3$ , the smoothing and  $k$ -grid LFA can be carried out as explained before in the standard way.

This analysis can be generalized for any isogeometric discretization with spline degree  $p$ . The corresponding stencil has  $(2p + 1)$  elements and it has the following form:  $A_{h,p} = [a_p, a_{p-1}, \dots, a_1, a_0, a_1, \dots, a_{p-1}, a_p]$ . Then, the matrices  $P$  and  $Q$  for the three-point multiplicative Schwarz smoother are given as follows:

$$P = \begin{pmatrix} a_2 e^{i\theta} & a_1 & \sum_{j=0}^p a_j e^{-i(j+1)\theta} \\ a_1 e^{i\theta} & a_0 & \sum_{j=1}^p a_j e^{-ij\theta} \\ a_0 e^{i\theta} & a_1 & \sum_{j=2}^p a_j e^{-i(j-1)\theta} \end{pmatrix}, \quad Q = \begin{pmatrix} -\sum_{j=3}^p a_j e^{i(j-1)\theta} \\ -\sum_{j=2}^p a_j e^{ij\theta} \\ -\sum_{j=1}^p a_j e^{i(j+1)\theta} \end{pmatrix}.$$

It is even more interesting to analyze multiplicative Schwarz iterations with larger blocks, since our strategy will be to increase the size of the blocks as the spline degree  $p$  increases. LFA for these relaxations can be done in the same way as that for the three-point multiplicative Schwarz smoother, but with heavier computations. Since we consider blocks centered around a grid-point, the number of unknowns within the block,  $n$ , is odd. Matrices  $P$  and  $Q$  for an arbitrary  $n$ -point multiplicative Schwarz iteration are  $(n \times n)$ - and  $(n \times 1)$ -matrices respectively given by

$$P = \bar{P}D_P, \quad \text{and} \quad Q = \begin{pmatrix} -\sum_{j=n}^p a_j e^{i(j-\frac{n-1}{2})\theta} \\ -\sum_{j=n-1}^p a_j e^{i(j-\frac{n-1}{2}+1)\theta} \\ \vdots \\ -\sum_{j=\frac{n-1}{2}+1}^p a_j e^{ij\theta} \\ -\sum_{j=\frac{n-1}{2}}^p a_j e^{i(j+1)\theta} \\ \vdots \\ -\sum_{j=1}^p a_j e^{i(j+\frac{n-1}{2})\theta} \end{pmatrix},$$

where matrix  $\bar{P}$  is

$$\begin{pmatrix} a_{n-1} & a_{n-2} & a_{n-3} & \cdots & a_{\frac{n-1}{2}} & \cdots & a_2 & a_1 & \sum_{j=0}^p a_j e^{-i(j+\frac{n-1}{2})\theta} \\ a_{n-2} & a_{n-3} & \cdots & \vdots & \cdots & \cdots & a_1 & a_0 & \sum_{j=1}^p a_j e^{-i(j+\frac{n-1}{2}-1)\theta} \\ a_{n-3} & \cdots & \vdots & \cdots & \cdots & \cdots & a_0 & a_1 & \sum_{j=2}^p a_j e^{-i(j+\frac{n-1}{2}-2)\theta} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & a_0 & \vdots & \vdots & \vdots & \sum_{j=\frac{n-1}{2}}^p a_j e^{-ij\theta} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ a_2 & a_1 & a_0 & \cdots & \cdots & \cdots & a_{n-4} & \sum_{j=n-3}^p a_j e^{-i(j-\frac{n-1}{2}+2)\theta} \\ a_1 & a_0 & a_1 & \cdots & \cdots & \cdots & a_{n-4} & a_{n-3} & \sum_{j=n-2}^p a_j e^{-i(j-\frac{n-1}{2}+1)\theta} \\ a_0 & a_1 & \cdots & \cdots & a_{\frac{n-1}{2}} & \cdots & a_{n-3} & a_{n-2} & \sum_{j=n-1}^p a_j e^{-i(j-\frac{n-1}{2})\theta} \end{pmatrix},$$

and  $D_P$  is the following diagonal matrix:

$$D_P = \text{diag} \left\{ e^{i\frac{n-1}{2}\theta}, e^{i(\frac{n-1}{2}-1)\theta}, \dots, e^{i\theta}, 1, e^{-i\theta}, \dots, e^{-i(\frac{n-1}{2}-1)\theta}, 1 \right\}.$$

Thus, the presented LFA can be used for the choice of an adequate multiplicative Schwarz smoother for each spline degree. In particular, as we did previously for Gauss–Seidel relaxation, we can study if the class of multiplicative Schwarz smoothers is able to eliminate the high-frequency components of the error for IGA with a large spline degree. In Figure 6 we show the symbol of the smoothing operator of the three-point and five-point Schwarz relaxations for each considered  $p$ . For  $p = 2$ , we can see that the three-point approach is enough to adequately remove the high components. This approach, however, provides worse results for  $p = 8$ . If, alternatively, we choose the five-point multiplicative Schwarz smoother, we can see in the picture that the high frequencies are removed for  $p = 2$  and  $p = 5$  in a very efficient way, but for  $p = 8$  it is not yet very satisfactory, and it would be recommended to increase the size of the blocks. We also analyze the eigenvalues of the two-grid operator based on the overlapping multiplicative Schwarz smoothers. In Figure 7 we can see for a fixed spline degree,  $p = 8$ , that the eigenvalues become smaller when the size of the blocks within the Schwarz iteration gets larger. For an arbitrary spline degree  $p$ , this local Fourier analysis gives us the asymptotic convergence rates of the multigrid method based on the different multiplicative Schwarz smoothers. Thus, this information, together with the computational cost of each relaxation procedure, can be used to justify the proposed strategy of varying the block-size within the smoother for different values of  $p$ .

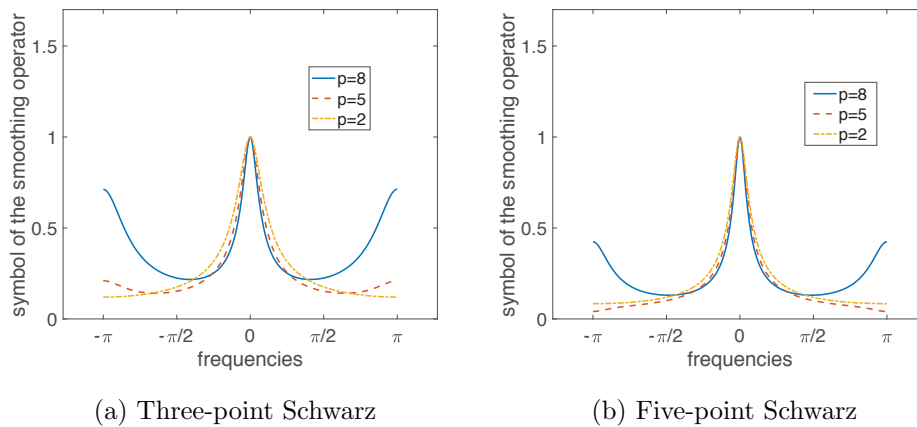


FIG. 6. Symbol of the smoothing operator corresponding to (a) the three-point and (b) the five-point multiplicative Schwarz smoothers, for three different spline degrees  $p = 2, 5, 8$ .

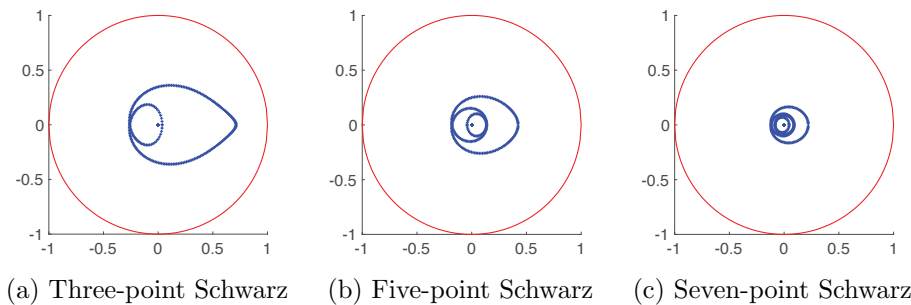


FIG. 7. Distribution of the eigenvalues of the two-grid method based on the three different overlapping multiplicative Schwarz smoothers considered, when applied to  $p = 8$ .

**4.3. Local Fourier analysis results.** In order to demonstrate the utility of the described LFA, next we show the good agreement between the convergence factors predicted by the LFA and the asymptotic convergence factors experimentally obtained by the multigrid method. For this purpose, we consider the multigrid components described in section 3, that is, the canonical spline embedding operator as the prolongation operator  $I_{2h}^h$  and its adjoint for the restriction  $I_h^{2h}$ , a Galerkin approximation on the coarse grids, and for the smoother  $S_h$  we will consider Gauss–Seidel iteration as well as the multiplicative Schwarz smoothers previously explained.

**One-dimensional case.** In Table 1 we provide the smoothing ( $\rho_{1g}$ ), two-grid ( $\rho_{2g}$ ) and three-grid ( $\rho_{3g}^V$ ) convergence factors obtained from LFA, considering one smoothing step of a Gauss–Seidel relaxation, together with the asymptotic convergence factors provided by the  $W(1, 0)$ - and  $V(1, 0)$ -cycle multigrid codes ( $\rho_h^W$  and  $\rho_h^V$ , respectively). We can observe a perfect match between the experimental factors and those predicted by LFA for both W- and V-cycles.

We also perform the analysis presented in section 4.2 for the multiplicative Schwarz smoothers. In Table 2, we show the smoothing and three-grid convergence factors predicted by the analysis and the experimentally obtained asymptotic convergence factors of a V-cycle multigrid with one pre- and no postsmoothing steps. These results are

TABLE 1

One-dimensional case: smoothing ( $\rho_{1g}$ ), two-grid ( $\rho_{2g}$ ), and three-grid ( $\rho_{3g}^V$ ) convergence factors predicted by LFA together with the asymptotic convergence factors provided by the  $W(1,0)$  and  $V(1,0)$  cycle multigrid codes ( $\rho_h^W$  and  $\rho_h^V$ , respectively) for different spline degrees  $p$ .

	Gauss–Seidel				
	$\rho_{1g}$	$\rho_{2g}$	$\rho_h^W$	$\rho_{3g}^V$	$\rho_h^V$
$p = 2$	0.31	0.19	0.19	0.19	0.19
$p = 3$	0.26	0.22	0.22	0.22	0.22
$p = 4$	0.38	0.38	0.38	0.38	0.38
$p = 5$	0.62	0.62	0.62	0.62	0.62
$p = 6$	0.79	0.79	0.80	0.79	0.80
$p = 7$	0.89	0.89	0.90	0.89	0.90
$p = 8$	0.99	0.99	0.96	0.99	0.96

TABLE 2

One-dimensional case: smoothing ( $\rho_{1g}$ ) and three-grid ( $\rho_{3g}^V$ ) convergence factors predicted by LFA together with the asymptotic convergence factors provided by the  $V(1,0)$ -cycle multigrid code ( $\rho_h^V$ ) for different spline degrees  $p$ .

	3p Schwarz			5p Schwarz			7p Schwarz		
	$\rho_{1g}$	$\rho_{3g}^V$	$\rho_h^V$	$\rho_{1g}$	$\rho_{3g}^V$	$\rho_h^V$	$\rho_{1g}$	$\rho_{3g}^V$	$\rho_h^V$
$p = 2$	0.176	0.127	0.127	0.119	0.088	0.087	0.089	0.065	0.065
$p = 3$	0.156	0.114	0.113	0.112	0.086	0.086	0.086	0.066	0.066
$p = 4$	0.146	0.127	0.127	0.104	0.084	0.084	0.082	0.067	0.067
$p = 5$	0.209	0.209	0.211	0.101	0.095	0.095	0.078	0.069	0.069
$p = 6$	0.389	0.389	0.389	0.147	0.147	0.147	0.077	0.077	0.077
$p = 7$	0.564	0.564	0.564	0.279	0.279	0.276	0.119	0.119	0.121
$p = 8$	0.712	0.712	0.712	0.424	0.424	0.426	0.221	0.221	0.224

shown for the three considered multiplicative Schwarz smoothers (three-, five-, and seven-point approaches) and for different spline degrees from  $p = 2$  to  $p = 8$ . We consider  $V$ -cycles because we have seen that their convergence rates are as those provided by  $W$ -cycles. We observe in the table that the smoothing ability of the proposed three-point multiplicative Schwarz relaxation deteriorates when  $p$  becomes larger. This affects the three-grid convergence factor that also gets worse, showing the necessity of considering multiplicative Schwarz smoothers coupling more than three points.

**Two-dimensional case.** We next provide the LFA results for the two-dimensional problem. In Table 3, the three-grid convergence factors predicted by LFA are shown together with the corresponding asymptotic convergence factors calculated by using the proposed  $V(1,0)$  multigrid method. A lexicographic Gauss–Seidel smoother and the multiplicative Schwarz relaxations with blocks of 9, 25, and 49 unknowns (9p Schwarz, 25p Schwarz, and 49p Schwarz, respectively) are considered. The results are presented again for different spline degrees ranging from  $p = 2$  to  $p = 8$ . The good match between the estimates predicted by LFA and the real asymptotic convergence factors is shown in the table. It is also observed that the bad behavior of the multigrid based on the Gauss–Seidel smoother, which was seen in the one-dimensional results, is even more remarkable in this two-dimensional case. Finally, it can be seen that, given a spline degree  $p$ , a multiplicative Schwarz smoother which provides a multigrid method with an efficient performance can be found.

We have demonstrated that the LFA is a very useful tool to obtain information about the performance of multigrid for IGA, since it predicts very accurately the

TABLE 3

Two-dimensional case: three-grid ( $\rho_{3g}^V$ ) convergence factors predicted by LFA together with the asymptotic convergence factors provided by the  $V(1,0)$ -cycle multigrid code ( $\rho_h^V$ ) for different spline degrees  $p$ .

	Gauss–Seidel		9p Schwarz		25p Schwarz		49p Schwarz	
	$\rho_{3g}^V$	$\rho_h^V$	$\rho_{3g}^V$	$\rho_h^V$	$\rho_{3g}^V$	$\rho_h^V$	$\rho_{3g}^V$	$\rho_h^V$
$p = 2$	0.510	0.510	0.099	0.099	0.067	0.066	0.050	0.050
$p = 3$	0.827	0.830	0.212	0.214	0.065	0.067	0.052	0.051
$p = 4$	0.954	0.955	0.452	0.455	0.127	0.145	0.051	0.053
$p = 5$	0.990	0.990	0.701	0.703	0.260	0.262	0.098	0.119
$p = 6$	0.999	0.999	0.870	0.872	0.447	0.440	0.190	0.165
$p = 7$	0.999	0.999	0.947	0.955	0.650	0.657	0.317	0.325
$p = 8$	0.999	0.999	0.985	0.982	0.816	0.807	0.473	0.440

convergence of the method. This is very interesting since for a fixed spline degree  $p$ , we can choose the appropriate number of points in the blocks to construct an efficient multiplicative Schwarz smoother. Of course, the computational cost of the method should be taken into account for this task.

**5. Choice of the smoothing strategy.** This section is devoted to the choice of the block-size of the multiplicative Schwarz smoother for a given spline degree  $p$ . In order to choose the most efficient smoothing strategy for each case, we consider the computational cost of the considered relaxations, which has been described at the end of section 3, where we provided an estimate of the computational complexity per smoothing step based on the block-size of the smoothers, together with the convergence factors predicted by the local Fourier analysis in the previous section. By using such predicted asymptotic convergence factors  $\rho_h$ , we can estimate the number of iterations  $n_{it}$  necessary to reduce the initial residual in a factor of  $10^{-\beta}$ . In particular, we have that

$$(10) \quad (\rho_h)^{n_{it}} \leq 10^{-\beta},$$

which gives rise to the following condition for the number of iterations:

$$n_{it} \geq -\frac{\beta}{\log_{10} \rho_h}.$$

This formula together with the computational cost per iteration of each smoother allows us to find which is the relaxation procedure most efficient for each spline degree  $p$ . After simple calculations, by roughly counting the number of operations needed for each of the considered smoothers to reduce the initial residual in a factor of  $10^{-8}$  (that is  $\beta = 8$  in (10)), we obtain the following conclusion: in the one-dimensional case, the three-point multiplicative Schwarz smoother has to be used when  $p = 2, 3, 4$ , the five-point multiplicative Schwarz relaxation is appropriate when  $p = 5, 6$ , and finally the seven-point multiplicative Schwarz smoother is chosen if  $p = 7, 8$ . Notice that the computations performed to estimate the number of iterations use the asymptotic convergence factor predicted by LFA. Analogously, by following the same procedure, we can state the smoothing strategy for the two-dimensional case: the nine-point multiplicative Schwarz iteration is preferred when  $p = 2, 3, 4$ , the 25-point multiplicative Schwarz relaxation is chosen if  $p = 5, 6$ , and the 49-point multiplicative Schwarz smoother is appropriate when  $p = 7, 8$ . This same procedure can be systematically applied to choose the appropriate block-size of a multiplicative Schwarz smoother for a given spline degree  $p$ .

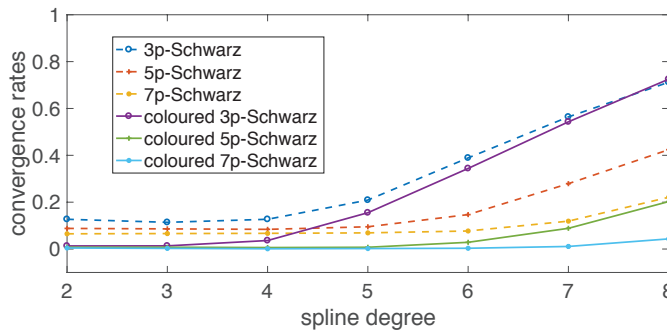


FIG. 8. Convergence rates of the multigrid method based on overlapping multiplicative Schwarz smoothers for the three considered Schwarz relaxations and their colored counterparts and for spline degrees  $p = 2, \dots, 8$ .

Based on the chosen strategy, we can observe that we have roughly that  $n \approx p$ . Therefore, the computational cost of the chosen relaxation is  $\mathcal{O}(N_{DOF} p^{2d})$ . Although the computational complexity is not optimal, it will be seen in the numerical experiments that our method requires a small number of iterations for achieving the desired convergence and that it is robust with respect to the spline degree, the mesh size, the dimension of the problem, and the geometry of the domain. These features, together with its ease of implementation, make this method a good alternative for solving IGA discretizations.

Notice that the different blocks within the multiplicative Schwarz smoothers can also be visited in different orderings; for instance, they can also be treated with some patterning scheme, yielding a multicolored version of these relaxation schemes. This type of approach usually provides better convergence. In our case, we have done a comparison of the convergence rates of the considered multiplicative Schwarz smoothers and their colored counterparts. In particular, we have considered a three-color version of these relaxation procedures.

In Figure 8, we compare the convergence factors of the lexicographic and colored multiplicative Schwarz smoothers for the one-dimensional case. By using dotted lines, we show the convergence rates provided by the proposed multigrid method based on the lexicographic three-, five-, and seven-point multiplicative Schwarz smoothers, for  $p = 2, \dots, 8$ . By using continuous lines, we also display the asymptotic convergence factors provided by a multigrid based on  $V(1,0)$ -cycles and the colored versions of the considered three-, five-, and seven-point multiplicative Schwarz smoothers. We can observe that the qualitative behavior of the method is as in the case for the noncolored smoothers, but the convergence rates are much better. Taking this into account, we can improve the smoothing strategy previously chosen by considering the colored counterparts of the selected relaxations. Notice that, from Figure 8, it is observed that the convergence rates provided with the colored multiplicative Schwarz smoothers are five times smaller than those provided with the lexicographic version of the smoothers for the spline degrees for which they are chosen.

**6. Numerical results.** In this section, we consider three different numerical experiments to illustrate the robustness and efficiency of the proposed multigrid method based on multiplicative Schwarz smoothers. First, we will solve both one- and two-dimensional problems on the parametric space. After that, we will deal with

a two-dimensional problem on a nontrivial computational domain. For the first two numerical experiments we choose B-splines as basis functions, whereas for the third numerical experiment NURBS are used because these functions can exactly describe the geometry for the considered domain.

Since the results obtained by the three-grid Fourier analysis in section 4 demonstrated that V-cycles provide similar convergence as W-cycles, the former are chosen because they have a lower computational cost and therefore a more efficient multigrid method is obtained. Moreover, one smoothing step is considered in all the numerical experiments, in particular V(1,0)-cycles are used. In all the cases the initial guess is taken as a random vector, and the stopping criterion for the multigrid solver is set to reduce the initial residual by a factor of  $10^{-8}$ . All the methods have been implemented in our in-house Fortran code, and the numerical computations were carried out on a MacBook Pro with a Core i5 with 2.7 GHz and 8 GB RAM, running OS X 10.10 (Yosemite). Regarding the presented CPU times, we want to remark that all of them include the setup phase in which the  $LU$ -factorization for solving the small dense systems within the relaxation is computed. Notice that for the experiments on the parametric domain such factorization is the same for most degrees of freedom, whereas for the last experiment, where a nontrivial geometry is considered, this factorization has to be computed for each grid-point.

**6.1. One-dimensional example on the parametric domain.** In the first numerical experiment we consider the following two-point boundary value problem:

$$\begin{cases} -u''(x) = \pi^2 \sin(\pi x), & x \in (0, 1), \\ u(0) = u(1) = 0. \end{cases}$$

We discretize this problem by using an equidistant knot span and maximum continuity splines for different degrees ranging from  $p = 2$  until  $p = 8$ . First, we compare the performance of the multigrid method based on different smoothers: a standard Gauss–Seidel and the considered multiplicative Schwarz relaxations with both lexicographic and colored orderings. With this purpose, we fix the grid-size to  $h = 2^{-18}$ . The comparison is done in Table 4, where the number of V(1,0)-multigrid iterations and the CPU times necessary to reduce the initial residual in a factor of  $10^{-8}$  are shown for different spline degrees for the considered approaches. We have marked in bold the most efficient strategy for each spline degree, and it is observed that it matches the strategy chosen in section 5. Thus, we consider colored multiplicative Schwarz smoothers with blocks composed of three points for the cases  $p = 2, 3, 4$ , five points for  $p = 5, 6$ , and seven points for  $p = 7, 8$ .

The next step is to see the scalability of the chosen strategy with respect to  $h$  and  $p$ . We apply the proposed V(1,0) multigrid cycle based on the colored multiplicative Schwarz smoother for solving the linear system. Table 5 shows the number of iterations needed to reduce the initial residual by a factor of  $10^{-8}$  for several mesh sizes  $h = 2^{-\ell}$  with  $\ell = 16, \dots, 20$  and different spline degrees  $p = 2, \dots, 8$ . We also report the computational time (in seconds) needed for solving the linear system by the proposed multigrid method. We observe that our approach shows an excellent convergence with respect to the mesh refinement for all the polynomial orders. But overall, we would like to note the robustness of the method with respect to the spline degree. In all cases, only four or five V(1,0)-cycles are needed to reach the stopping criterium, independently of the mesh size and the polynomial order. Regarding the computational cost, we observe a very good scaling, not only with respect to  $h$  but also with respect to  $p$ .

TABLE 4

One-dimensional test problem. Number of  $V(1,0)$  multigrid iterations (*it*) and computational time (*cpu*) necessary to reduce the initial residual in a factor of  $10^{-8}$ , for different values of the spline degree  $p$ , using the Gauss–Seidel (GS) smoother or the multiplicative Schwarz relaxations considered here: three-point (3p-Sch), five-point (5p-Sch), seven-point (7p-Sch), colored three-point (c-3p-Sch), colored five-point (c-5p-Sch), and colored seven-point (c-7p-Sch) multiplicative Schwarz smoothers.

$p$	GS		3p-Sch		c-3p-Sch		5p-Sch		c-5p-Sch		7p-Sch		c-7p-Sch	
	it	cpu	it	cpu	it	cpu	it	cpu	it	cpu	it	cpu	it	cpu
2	10	0.28	9	0.36	<b>5</b>	<b>0.22</b>	7	0.43	5	0.32	7	0.62	4	0.36
3	10	0.33	7	0.33	<b>5</b>	<b>0.25</b>	7	0.49	4	0.28	6	0.58	4	0.40
4	16	0.56	8	0.41	<b>5</b>	<b>0.27</b>	7	0.52	4	0.32	6	0.64	4	0.44
5	29	1.24	10	0.55	8	0.48	7	0.57	<b>4</b>	<b>0.35</b>	6	0.71	4	0.47
6	55	2.29	14	0.84	12	0.71	7	0.63	<b>5</b>	<b>0.45</b>	6	0.83	4	0.52
7	106	4.98	20	1.24	19	1.20	10	0.96	6	0.59	6	0.79	<b>4</b>	<b>0.53</b>
8	201	10.44	31	2.05	38	2.52	13	1.26	9	0.88	8	1.15	<b>5</b>	<b>0.68</b>

TABLE 5

One-dimensional test problem. Number of  $V(1,0)$  multigrid iterations (*it*) and computational time (*cpu*) necessary to reduce the initial residual in a factor of  $10^{-8}$ , for different grid-sizes  $h$  and for different values of the spline degree  $p$ , using the most appropriate colored multiplicative Schwarz smoother for each  $p$ .

$\ell$	Color 3p Schwarz			Color 5p Schwarz		Color 7p Schwarz								
	$p = 2$		$p = 3$	$p = 4$		$p = 5$		$p = 6$						
	it	cpu	it	cpu	it	cpu	it	cpu	it	cpu				
16	5	0.07	5	0.07	5	0.08	4	0.11	5	0.13	4	0.15	5	0.19
17	5	0.13	5	0.15	5	0.15	4	0.18	5	0.24	4	0.28	5	0.36
18	5	0.22	5	0.25	5	0.27	4	0.35	5	0.45	4	0.53	5	0.68
19	5	0.43	5	0.47	5	0.51	4	0.66	5	0.88	4	1.07	5	1.35
20	5	0.82	5	0.91	5	0.99	4	1.28	5	1.70	4	2.08	5	2.69

**6.2. Two-dimensional example on the parametric domain.** We now test the performance of the geometric multigrid solver based on overlapping multiplicative Schwarz smoothers on a two-dimensional problem defined on the parametric domain  $\Omega = (0, 1)^2$ . More concretely, we consider the problem,

$$\begin{cases} -\Delta u = 2\pi^2 \sin(\pi x) \sin(\pi y), & (x, y) \in \Omega = (0, 1)^2, \\ u(x, y) = 0, & (x, y) \text{ on } \partial\Omega. \end{cases}$$

Since the problem is solved in the parametric domain, we choose the basis functions as B-splines. Similar to the one-dimensional case, the performance of the geometric multigrid solver based on classical smoothers like Gauss–Seidel iteration is very poor. This can be seen in Table 6, where the behavior of the multigrid method based on different smoothers—Gauss–Seidel and lexicographic/colored multiplicative Schwarz relaxations—is compared. For a mesh of size  $512 \times 512$ , we display the number of  $V(1,0)$ -multigrid iterations together with the CPU times necessary to reduce the initial residual in a factor of  $10^{-8}$ . Again, the bad behavior of the Gauss–Seidel based multigrid method due to the presence of many small eigenvalues associated with oscillatory components of the error is clearly observed. There are some cases, indicated with the symbol –, for which more than 500 iterations would be needed for convergence. We also notice that the optimal strategy, in bold letters, coincides with that proposed in section 5. In particular, we choose the colored multiplicative Schwarz relaxations with blocks of size  $3 \times 3$  for the cases  $p = 2, 3, 4$ , blocks of size  $5 \times 5$  for the cases  $p = 5, 6$ , and blocks of size  $7 \times 7$  for splines degree  $p = 7, 8$ . We

TABLE 6

Two-dimensional test problem. Number of  $V(1,0)$  multigrid iterations (*it*) and computational time (*cpu*) necessary to reduce the initial residual in a factor of  $10^{-8}$ , for different values of the spline degree  $p$ , using the Gauss–Seidel (GS) smoother or the multiplicative Schwarz relaxations considered here: three-point (9p-Sch), five-point (25p-Sch), seven-point (49p-Sch), colored three-point (c-9p-Sch), colored five-point (c-25p-Sch), and colored seven-point (c-49p-Sch) multiplicative Schwarz smoothers.

$p$	GS		9p-Sch		c-9p-Sch		25p-Sch		c-25p-Sch		49p-Sch		c-49p-Sch	
	it	cpu	it	cpu	it	cpu	it	cpu	it	cpu	it	cpu	it	cpu
2	20	1.10	8	1.29	<b>4</b>	<b>0.68</b>	7	4.89	4	2.88	6	14.21	3	7.46
3	57	4.90	8	2.13	<b>4</b>	<b>0.99</b>	6	5.60	4	4.02	6	17.80	3	9.30
4	166	21.04	14	4.66	<b>7</b>	<b>2.40</b>	7	8.38	4	5.08	6	22.40	4	15.56
5	492	96.03	19	17.26	17	10.40	9	17.83	<b>4</b>	<b>6.09</b>	6	39.96	3	14.69
6	–	–	31	21.53	40	28.40	11	20.01	<b>5</b>	<b>9.33</b>	7	39.90	3	16.06
7	–	–	60	48.04	112	98.70	15	34.32	9	21.80	9	60.44	<b>3</b>	<b>17.15</b>
8	–	–	88	105.62	254	310.55	22	71.19	15	48.87	11	88.17	<b>4</b>	<b>26.01</b>

TABLE 7

Two-dimensional test problem. Number of  $V(1,0)$  multigrid iterations (*it*) and computational time (*cpu*) necessary to reduce the initial residual in a factor of  $10^{-8}$ , for different grid-sizes and for different values of the spline degree  $p$ , using the most appropriate colored multiplicative Schwarz smoother for each  $p$ .

Grid	Color 9p Schwarz						Color 25p Schwarz				Color 49p Schwarz			
	$p = 2$		$p = 3$		$p = 4$		$p = 5$		$p = 6$		$p = 7$		$p = 8$	
	it	cpu	it	cpu	it	cpu	it	cpu	it	cpu	it	cpu	it	cpu
$128^2$	4	0.06	4	0.10	7	0.22	4	0.62	5	0.99	3	2.22	4	3.35
$256^2$	4	0.21	4	0.29	7	0.69	4	1.40	5	2.86	3	6.03	4	9.24
$512^2$	4	0.68	4	0.99	7	2.40	4	6.09	5	9.33	3	17.15	4	26.01
$1024^2$	4	2.60	4	3.80	7	9.13	3	16.49	5	33.05	3	53.69	4	80.74

want to notice that, although the computational cost analysis done for the choice of the smoothing strategy stated that the noncolored 49-point smoother was preferred to the noncolored 25-point relaxation for  $p = 7, 8$ , the CPU times in Table 6 say the contrary due to a remarkable reduction of the convergence rate of the method in the first iterations. Due to this, such convergence rate is much smaller than that predicted by the LFA and the real number of iterations is much lower than the one considered for the choice of the strategy in section 5. We want to remark, however, that this occurs just in exceptional cases and for most of the times the provided CPU times support the relaxation choice made in the previous section.

We now apply the proposed strategy for different grids with mesh sizes from  $128 \times 128$  to  $1024 \times 1024$  and various spline degrees ranging from  $p = 2$  to  $p = 8$  in order to show the good scalability of the solver. Again, we use  $V$ -cycles with one pre-smoothing step and no post-smoothing steps. The number of iterations needed to reduce the initial residual by a factor of  $10^{-8}$  together with the computational time (in seconds) are given in Table 7. We observe that the iteration numbers are robust with respect to both the size of the grid  $h$  and the spline degree  $p$ . Moreover, we see that the number of iterations is similar to that reported in the one-dimensional case. We emphasize that a small number of  $V(1,0)$ -cycles are needed to reach the stopping criterium, independently of  $h$  and  $p$ . Just like in the one-dimensional case, we can conclude that the multigrid method based on an appropriate multiplicative Schwarz smoother provides an efficient and robust solver for B-spline isogeometric discretizations.

**6.3. Quarter annulus.** The last experiment demonstrates the efficiency and robustness of the proposed multigrid method to deal with a nontrivial geometry. We take as the domain the quarter of an annulus,

$$\Omega = \{(x, y) \in \mathbb{R}^2 \mid r^2 \leq x^2 + y^2 \leq R^2, x, y \geq 0\},$$

which is sketched in Figure 9. We consider the solution of the Poisson problem in such a domain with homogeneous Dirichlet boundary conditions

$$\begin{cases} -\Delta u = f(x, y), & (x, y) \in \Omega, \\ u(x, y) = 0, & (x, y) \text{ on } \partial\Omega, \end{cases}$$

where  $f(x, y)$  is such that the exact solution is

$$u(x, y) = \sin(\pi x) \sin(\pi y) (x^2 + y^2 - r^2) (x^2 + y^2 - R^2).$$

The geometry of the computational domain is described exactly by quadratic  $C^1$  NURBS, and we choose in our experiments  $r = 0.3$  and  $R = 0.5$ . To discretize this problem, we use NURBS of degree  $p = 2, \dots, 8$  with maximal smoothness. We solve the corresponding linear systems using  $V(1, 0)$ -cycles. If the Gauss–Seidel smoother is considered, the bad behavior of the multigrid method reported in the previous experiment is also observed here. More concretely, for  $p = 2$  around 50 iterations are required to achieve the desired convergence, about 70 iterations are needed when  $p = 3$ , and for  $p \geq 4$  the resulting number of iterations is bigger than 200. We consider then the colored multiplicative Schwarz smoother described in section 5. The size of the blocks of the Schwarz smoother depends on  $p$ , and they are chosen following the same strategy as in the previous experiment. The numbers of iterations and CPU times needed to reach the stopping criterion, for various degrees  $p$  and for different mesh sizes, are reported in Table 8. We observe an excellent performance, obtaining

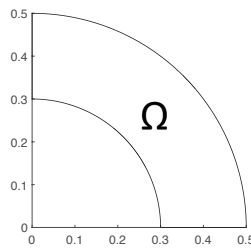


FIG. 9. Computational domain for the Poisson problem on a quarter of an annulus.

TABLE 8

Quarter annulus problem. Number of  $V(1, 0)$  multigrid iterations (*it*) and computational time (*cpu*) necessary to reduce the initial residual in a factor of  $10^{-8}$ , for different grid-sizes and for different values of the spline degree  $p$ , using the most appropriate colored multiplicative Schwarz smoother for each  $p$ .

Grid	Color 9p Schwarz						Color 25p Schwarz				Color 49p Schwarz			
	$p = 2$		$p = 3$		$p = 4$		$p = 5$		$p = 6$		$p = 7$		$p = 8$	
	it	cpu	it	cpu	it	cpu	it	cpu	it	cpu	it	cpu	it	cpu
$32^2$	4	0.02	4	0.02	8	0.05	4	0.13	5	0.20	3	0.65	4	0.72
$64^2$	4	0.03	4	0.04	7	0.08	4	0.30	5	0.46	3	1.28	5	2.15
$128^2$	4	0.07	4	0.10	8	0.25	4	0.96	6	1.42	3	3.70	5	7.01
$256^2$	4	0.20	4	0.30	7	0.74	4	3.44	6	5.01	3	13.13	5	23.75

results similar to those of the example with the parametric domain. We can conclude that the proposed solver is robust with respect to this geometry transformation, having a great potential for solving problems in more complicated multipatch geometries, which is the subject of future research.

**7. Conclusions.** An efficient and robust geometric multigrid method, based on overlapping multiplicative Schwarz smoothers, is proposed for IGA. The robustness of the algorithm with respect to the spline degree is demonstrated through numerical experiments and an LFA. This analysis is carried out for any spline degree  $p$  and an arbitrary size of the blocks to solve in the relaxation procedure. The key point to achieve a robust algorithm is the choice of larger blocks within the Schwarz smoother when the spline degree grows up. Moreover, to improve the convergence rates provided by the multigrid method, we considered a colored version of the multiplicative Schwarz smoothers. A simple multigrid V-cycle based on this type of smoother with one smoothing step results in a robust and efficient algorithm for isogeometric discretizations.

#### REFERENCES

- [1] P. BASTIAN, E. H. MÜLLER, S. MÜTHING, AND M. PIATKOWSKI, *Matrix-Free Multigrid Block-Preconditioners for Higher Order Discontinuous Galerkin Discretisations*, CoRR abs/1805.11930, 2018.
- [2] A. BRANDT, *Multi-level adaptive solutions to boundary-value problems*, Math. Comp., 31 (1977), pp. 333–390.
- [3] A. BRANDT, *Rigorous quantitative analysis of multigrid, I: Constant coefficients two-level cycle with  $L_2$ -norm*, SIAM J. Numer. Anal., 31 (1994), pp. 1695–1730.
- [4] A. BUFFA, H. HARBRECHT, A. KUNOTH, AND G. SANGALLI, *BPX-preconditioning for isogeometric analysis*, Comput. Methods Appl. Mech. Engrg., 265 (2013), pp. 63–70, <https://doi.org/10.1016/j.cma.2013.05.014>.
- [5] N. COLLIER, D. PARDO, L. DALCÍN, M. PASZYNSKI, AND V. CALO, *The cost of continuity: A study of the performance of isogeometric finite elements using direct solvers*, Comput. Methods Appl. Mech. Engrg., 213–216 (2012), pp. 353–361, <https://doi.org/10.1016/j.cma.2011.11.002>.
- [6] N. O. COLLIER, L. DALCÍN, D. PARDO, AND V. M. CALO, *The cost of continuity: Performance of iterative solvers on isogeometric finite elements*, SIAM J. Sci. Comput., 35 (2013), pp. A767–A784.
- [7] J. COTTRELL, T. HUGHES, AND A. REALI, *Studies of refinement and continuity in isogeometric structural analysis*, Comput. Methods Appl. Mech. Engrg., 196 (2007), pp. 4160–4183, <https://doi.org/10.1016/j.cma.2007.04.007>.
- [8] J. A. COTTRELL, T. J. R. HUGHES, AND Y. BAZILEVS, *Isogeometric Analysis: Toward Integration of CAD and FEA*, Wiley, New York, 2009.
- [9] C. D. BOOR, *A Practical Guide to Splines*, Springer, New York, 1978.
- [10] L. BEIRÃO DA VEIGA, A. BUFFA, G. SANGALLI, AND R. VÁZQUEZ, *Mathematical analysis of variational isogeometric methods*, Acta Numer., 23 (2014), 157287, <https://doi.org/10.1017/S096249291400004X>.
- [11] L. BEIRÃO DA VEIGA, D. CHO, L. F. PAVARINO, AND S. SCACCHI, *Overlapping Schwarz methods for isogeometric analysis*, SIAM J. Numer. Anal., 50 (2012), pp. 1394–1416, <https://doi.org/10.1137/110833476>.
- [12] L. BEIRÃO DA VEIGA, D. CHO, L. F. PAVARINO, AND S. SCACCHI, *BDDC preconditioners for isogeometric analysis*, Math. Models Methods Appl. Sci., 23 (2013), pp. 1099–1142, <https://doi.org/10.1142/S0218202513500048>.
- [13] M. DONATELLI, C. GARONI, C. MANNI, S. S. CAPIZZANO, AND H. SPELEERS, *Symbol-based multigrid methods for Galerkin B-spline isogeometric analysis*, SIAM J. Numer. Anal., 55 (2017), pp. 31–62, <https://doi.org/10.1137/140988590>.
- [14] M. DONATELLI, C. GARONI, C. MANNI, S. SERRA-CAPIZZANO, AND H. SPELEERS, *Robust and optimal multi-iterative techniques for IgA Galerkin linear systems*, Comput. Methods Appl. Mech. Engrg., 284 (2015), pp. 230–264, <https://doi.org/10.1016/j.cma.2014.06.001>.

- [15] K. GAHALAUT, J. KRAUS, AND S. TOMAR, *Multigrid methods for isogeometric discretization*, *Comput. Methods Appl. Mech. Engrg.*, 253 (2013), pp. 413–425, <https://doi.org/10.1016/j.cma.2012.08.015>.
- [16] K. GAHALAUT, S. TOMAR, AND J. KRAUS, *Algebraic multilevel preconditioning in isogeometric analysis: Construction and numerical studies*, *Comput. Methods Appl. Mech. Engrg.*, 266 (2013), pp. 40–56, <https://doi.org/10.1016/j.cma.2013.07.002>.
- [17] W. HACKBUSCH, *Multi-grid Methods and Applications*, Springer, New York, 1985.
- [18] C. HOFREITHER AND S. TAKACS, *Robust multigrid for isogeometric analysis based on stable splittings of spline spaces*, *SIAM J. Numer. Anal.*, 55 (2017), pp. 2004–2024, <https://doi.org/10.1137/16M1085425>.
- [19] C. HOFREITHER, S. TAKACS, AND W. ZULEHNER, *A robust multigrid method for isogeometric analysis in two dimensions using boundary correction*, *Comput. Methods Appl. Mech. Engrg.*, 316 (2017), pp. 22–42, <https://doi.org/10.1016/j.cma.2016.04.003>.
- [20] C. HOFREITHER AND W. ZULEHNER, *Mass smoothers in geometric multigrid for isogeometric analysis*, in *Curves and Surfaces*, J.-D. Boissonnat, A. Cohen, O. Gibaru, C. Gout, T. Lyche, M.-L. Mazure, and L. L. Schumaker, eds., Springer, New York, 2015, pp. 272–279.
- [21] C. HOFREITHER AND W. ZULEHNER, *Spectral analysis of geometric multigrid methods for isogeometric analysis*, in *Numerical Methods and Applications*, I. Dimov, S. Fidanova, and I. Lirkov, eds., Springer, New York, 2015, pp. 123–129.
- [22] T. HUGHES, J. COTTRELL, AND Y. BAZILEVS, *Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement*, *Comput. Methods Appl. Mech. Engrg.*, 194 (2005), pp. 4135–4195, <https://doi.org/10.1016/j.cma.2004.10.008>.
- [23] S. K. KLEISS, C. PECHSTEIN, B. JTTLER, AND S. TOMAR, *IETI—Isogeometric tearing and interconnecting*, *Comput. Methods Appl. Mech. Engrg.*, 247–248 (2012), pp. 201–215, <https://doi.org/10.1016/j.cma.2012.08.007>.
- [24] S. P. MACLACHLAN AND C. W. OOSTERLEE, *Local Fourier analysis for multigrid with overlapping smoothers applied to systems of PDEs*, *Numer. Linear Algebra Appl.*, 18 (2011), pp. 751–774, <https://doi.org/10.1002/nla.762>.
- [25] S. A. ORSZAG, *Spectral methods for problems in complex geometries*, *J. Comput. Phys.*, 37 (1980), pp. 70–92, [https://doi.org/10.1016/0021-9991\(80\)90005-4](https://doi.org/10.1016/0021-9991(80)90005-4).
- [26] C. RODRIGO, F. GASPAR, AND F. LISBONA, *On a local Fourier analysis for overlapping block smoothers on triangular grids*, *Appl. Numer. Math.*, 105 (2016), pp. 96–111, <https://doi.org/10.1016/j.apnum.2016.02.006>.
- [27] G. SANGALLI AND M. TANI, *Isogeometric preconditioners based on fast solvers for the Sylvester equation*, *SIAM J. Sci. Comput.*, 38 (2016), pp. A3644–A3671, <https://doi.org/10.1137/16M1062788>.
- [28] K. STÜBEN AND U. TROTTEBERG, *Multigrid methods: Fundamental algorithms, model problem analysis and applications*, in *Multigrid Methods*, W. Hackbusch and U. Trottenberg, eds., *Lecture Notes in Math.* 960, Springer, New York, 1982, pp. 1–176.
- [29] U. TROTTEBERG, C. W. OOSTERLEE, AND A. SCHÜLLER, *Multigrid*, Academic Press, New York, 2001.
- [30] P. WESSELING, *An Introduction to Multigrid Methods*, Wiley, New York, 1992.
- [31] R. WIENANDS AND W. JOPPICH, *Practical Fourier Analysis for Multigrid Methods*, Chapman and Hall/CRC Press, Boca Raton, FL, 2005.