

SCHWARZ DOMAIN DECOMPOSITION PRECONDITIONERS FOR DISCONTINUOUS GALERKIN APPROXIMATIONS OF ELLIPTIC PROBLEMS: NON-OVERLAPPING CASE *

PAOLA F. ANTONIETTI¹ AND BLANCA AYUSO²

Abstract. We propose and study some new additive, two-level non-overlapping Schwarz preconditioners for the solution of the algebraic linear systems arising from a wide class of discontinuous Galerkin approximations of elliptic problems that have been proposed up to now. In particular, two-level methods for both symmetric and non-symmetric schemes are introduced and some interesting features, which have no analog in the conforming case, are discussed. Both the construction and analysis of the proposed domain decomposition methods are presented in a unified framework. For symmetric schemes, it is shown that the condition number of the preconditioned system is of order $O(H/h)$, where H and h are the mesh sizes of the coarse and fine grids respectively, which are assumed to be nested. For non-symmetric schemes, we show by numerical computations that the Eisenstat *et al.* [*SIAM J. Numer. Anal.* **20** (1983) 345–357] GMRES convergence theory, generally used in the analysis of Schwarz methods for non-symmetric problems, cannot be applied even if the numerical results show that the GMRES applied to the preconditioned systems converges in a finite number of steps and the proposed preconditioners seem to be scalable. Extensive numerical experiments to validate our theory and to illustrate the performance and robustness of the proposed two-level methods are presented.

Mathematics Subject Classification. 65N30, 65N55.

Received: October 6, 2005. Revised: September 29, 2006.

1. INTRODUCTION

Over the last twenty years there has been an active development of domain decomposition (DD) techniques for the solution of large algebraic systems arising from the numerical approximation of partial differential equations. However, although the analysis and development of DD techniques for classical discretisation methods (finite difference, finite volume and finite element – conforming, nonconforming and mixed – methods) is by now well understood and rather mature (see the pioneering work [31] or [40, 43] and [46–48]), for discontinuous Galerkin (DG) discretisations it is still under development. In fact, very few contributions to the field can be found in the recent literature and all of them are in the framework of overlapping and non-overlapping Schwarz methods. More precisely, we are only aware of three works where two-level methods for DG discretisations are proposed: elliptic problems are treated by Feng and Karakashian in [23], advection-diffusion problems are considered by

Keywords and phrases. Domain decomposition methods, discontinuous Galerkin, elliptic problems.

* Blanca Ayuso was supported by the IHP Network Breaking Complexity at the IMATI, CNR (Pavia).

¹ Dipartimento di Matematica, Università di Pavia, via Ferrata 1, 27100 Pavia, Italy. paola.antonietti@unipv.it

² Istituto di Matematica Applicata e Tecnologie Informatiche, CNR, via Ferrata 1, 27100 Pavia, Italy. blanca@imati.cnr.it

© EDP Sciences, SMAI 2007

Lasser and Toselli in [30], and Brenner and Wang have studied the case of fourth order problems in [10]. In this paper we propose and analyse in a unified framework some new non-overlapping Schwarz preconditioners for the algebraic linear systems obtained from a wide class of DG discretisations of elliptic problems proposed up to now.

The first DG method was introduced by Reed and Hill in 1973 [35] for hyperbolic equations, and since that time there has been an active development of DG methods for hyperbolic [12, 29], nearly hyperbolic [17–19] problems and for purely elliptic [3, 6, 8, 21, 42, 44] problems. Furthermore, DG approximations of elliptic equations have become increasingly popular in recent years [11, 15, 20, 28, 36, 37] (see in particular [4] for a unified analysis and [2] for the spectral analysis). DG methods use piecewise polynomial function spaces which can be totally discontinuous at the interelement boundaries and have several advantages over other types of finite element methods: for example, they allow for a great flexibility in the mesh design and for handling inhomogeneous boundary conditions in a natural way. However, in spite of the great success of DG methods, it is usually recognized that their main drawback is the larger number of degrees of freedom compared to other classical discretisation methods. Therefore, some attention needs to be devoted to devise efficient techniques and remedies to offset this possible limitation of DG methods.

The aim of the present paper is twofold. The first goal is to propose additive two-level methods for the DG approximation of elliptic problems, which really share the features of the classical Schwarz technique. The second goal is to provide the construction and the analysis for the proposed two-level methods in the unified framework of [4]. Therefore, our construction and analysis covers at the same time a wide class DG methods proposed in the literature up to now. It is worth noticing that, in contrast to classical discretisation methods, the lack of continuity constraints across the element interfaces inherent to DG approximations leads to some freedom in the definition of the local solvers. In the two-level methods proposed in [23] and [30], the local solvers are defined as the restriction to each subdomain of the global DG approximation. Our approach is rather different since we define the local solvers as the corresponding DG approximation to the continuous problem, but set in the subdomain. In the framework of Schwarz methods, it results that our local solvers are *approximate* rather than exact as the ones proposed in [23, 30]. Nevertheless, our setting present several advantages over the previously mentioned works in terms of parallelization, since each block in the preconditioner could be computed on a different processor and so the communication between different processors is minimized. Furthermore, the coarse solver can be either discontinuous, as advocated in [23], or conforming, as suggested in [30], and we have addressed the issue of which gives better performance by means of numerical experiments.

The theoretical analysis is carried out under the assumption that the subdomain partition \mathcal{T}_S , the coarse partition \mathcal{T}_H and the fine partition \mathcal{T}_h are nested, *i.e.*, $\mathcal{T}_S \subseteq \mathcal{T}_H \subseteq \mathcal{T}_h$. As a first step we show, in a unified way and for all the DG methods (regardless they are symmetric or not), the existence of a stable splitting for the discontinuous local spaces and the corresponding bilinear forms. This is, in fact, a crucial issue in the convergence analysis of the Schwarz methods. In the convergence analysis of the resulting preconditioned systems, we have to distinguish between symmetric and non-symmetric DG approximations.

For symmetric DG approximations, following the lines of the classical theory for Schwarz methods [43], we prove that the condition number of the preconditioned systems is of the order $O(H/h)$ when \mathcal{T}_H and \mathcal{T}_h are quasi-uniform (*cf.* Th. 5.1). These estimates are in agreement with the results proved earlier in [23] for the SIPG(δ) method proposed in [28, 42]. Furthermore, we also prove that the condition number of the preconditioned systems depends linearly on the stabilisation parameter.

For the non-symmetric NIPG method introduced in [36] and the recently proposed IIPG method [20], we show numerically that the proposed Schwarz preconditioners are scalable and perform in an optimal way. However, due to the lack of symmetry of these methods to study the convergence of the preconditioned systems, one has to appeal to one of the existing *Generalized Minimal Residual* (GMRES) theories which are all far from being optimal. We have addressed the issue of GMRES convergence by following the Eisenstat *et al.* convergence theory [22], which is generally advocated by the analysis of Schwarz methods for non-symmetric problems. Such a GMRES theory guarantees convergence of the iterative solver provided that: (i) the minimum eigenvalue of the

symmetric part of the Schwarz operator is strictly positive; (ii) the norm of the Schwarz operator is bounded. Unfortunately, the results we present are mostly negative. More precisely, we show by means of numerical computations that the symmetric part of the preconditioned operator is generally not positive definite. As a consequence and although we prove condition (ii), we conclude that the Eisenstat *et al.* theory cannot be applied for proving the observed convergence of the proposed preconditioners.

To the best of our knowledge, this is the first time that a non-symmetric additive Schwarz preconditioner has been proposed for a diffusion problem. Furthermore, it is worth noticing that the proposed two-level methods for non-symmetric DG approximations could be used for more general elliptic problems where the skew-symmetric part of the operator is still “small” with respect to the symmetric part, typically a low-order relatively compact perturbation of an elliptic operator.

Other theories and techniques, such as the Saad-Schultz theory [38] and the method based on the field-of-values analysis [41] have been also used to study the convergence of the preconditioned GMRES method. The look to these theories for analysing the convergence of the proposed methods might merit without doubt further investigation, and will be the subject of future research.

We remark that, in this work, we are only concerned with DG methods that are stable and consistent in the sense of [4]. However, we have also included some numerical tests with our two-level Schwarz method when discretising with the super penalty method proposed in [5] which is not consistent hence to which our analysis does not apply. Nevertheless, a dramatic reduction on the number of iterates needed for convergence and on the condition number for the preconditioned system is observed.

The paper is organized as follows. In Section 2, we recall the unified framework of DG approximations for elliptic problems given in [4] and revise some well known results, required in our subsequent analysis. In Section 3, we provide a unified framework for the construction of the additive Schwarz preconditioners. Section 4 is devoted to show that a *stable decomposition* and a *local stability* properties hold true. The convergence analysis of the preconditioned system is performed in Section 5 for symmetric DG methods. In Section 6 we study the case of the non-symmetric NIPG and IIPG methods. In Section 7 we provide a large set of numerical experiments for many DG methods with different polynomial degrees, to assess the efficiency of the proposed preconditioners and to validate our theoretical results. In Section 8 we draw some conclusions. Finally, some technical results are proved in the Appendix.

2. UNIFIED FRAMEWORK OF DISCONTINUOUS GALERKIN METHODS FOR ELLIPTIC PROBLEMS

Throughout this paper, we shall use the standard notation for Sobolev spaces (see [1]). For a bounded domain B in \mathbb{R}^d , $d = 2, 3$, we denote by $H^m(B)$ the standard Sobolev space of order $m \geq 0$, and by $\|\cdot\|_{m,B}$ and $|\cdot|_{m,B}$ the usual Sobolev norm and seminorm, respectively. For $m = 0$, we write $L^2(B)$ instead of $H^0(B)$.

In this section, following [4], we briefly review the unified framework for DG methods. For the sake of simplicity and to ease the presentation we restrict ourselves to the following model problem (see however Rem. 2.1):

$$Lu = -\Delta u = f \quad \text{in } \Omega, \quad u = 0 \quad \text{on } \partial\Omega, \quad (1)$$

where $\Omega \subset \mathbb{R}^d$, $d = 2, 3$, is (a smooth domain or) a convex polygon or polyhedron and f is a given function in $L^2(\Omega)$.

Remark 2.1. We wish to note that the results presented here apply to the more general case in which L is a second order elliptic operator in divergence form. Therefore, other kinds of equations and boundary conditions can be considered, but discussing the necessary changes for every particular situation is outside the scope of the present paper.

We start by setting some notation and the functional setting, then we briefly review the derivation of the primal formulation of the DG methods and the key ingredients needed for our analysis of the Schwarz method.

2.1. Notation

Let $\{\mathcal{T}_h\}_{h>0}$ be a family of partitions of the domain Ω made of triangles or parallelograms (if $d = 2$), or tetrahedra or parallelepipeds (if $d = 3$) which we assume to be *regular* and *quasi-uniform* (see [16] for details). We denote by $H^m(\mathcal{T}_h)$ the space of functions on \mathcal{T}_h whose restriction to each element $T \in \mathcal{T}_h$ belongs to $H^m(T)$. Let \mathcal{E}^I and \mathcal{E}^B be the sets of all interior and boundary faces, respectively, we set $\mathcal{E} = \mathcal{E}^I \cup \mathcal{E}^B$. We will use the convention that

$$\int_{\mathcal{E}} v \, ds = \sum_{e \in \mathcal{E}} \int_e v \, ds.$$

We also define $T(\mathcal{E}) = \prod_{T \in \mathcal{T}_h} L^2(\partial T)$. We next introduce two finite element spaces: for all $\ell_h \geq 1$, we set

$$\Sigma_h = \{\boldsymbol{\sigma}_h \in [L^2(\Omega)]^d : \boldsymbol{\sigma}_h|_T \in [\mathcal{M}^{\ell_h}(T)]^d \quad \forall T \in \mathcal{T}_h\}, \quad V_h = \{v_h \in L^2(\Omega) : v_h|_T \in \mathcal{M}^{\ell_h}(T) \quad \forall T \in \mathcal{T}_h\},$$

where $\mathcal{M}^{\ell_h}(T)$ is either the space $\mathcal{P}^{\ell_h}(T)$ of polynomials of degree at most ℓ_h on T , for T a triangle or a tetrahedron, or $\mathcal{Q}^{\ell_h}(T)$ for T a parallelogram or a parallelepiped, where $\mathcal{Q}^{\ell_h}(T)$ is the mapping to T of $\widehat{\mathcal{Q}}^{\ell_h}(\widehat{T})$ (*i.e.*, polynomials of degree at most ℓ_h in each variable on \widehat{T}), and \widehat{T} is the open unit hypercube in \mathbb{R}^d .

For piecewise smooth vector-valued and scalar functions $\boldsymbol{\tau}$ and u , respectively, we introduce the following trace operators. Let T^+ and T^- be two adjacent elements of \mathcal{T}_h , let $e = \partial T^+ \cap \partial T^- \in \mathcal{E}^I$ be the interior of the $(d-1)$ -dimensional face (if $d = 2$, “face” means “edge”) shared by these elements and let \mathbf{n}^\pm be the corresponding outward normal unit vectors pointing exterior to T^\pm , respectively. Denoting by $\boldsymbol{\tau}^\pm$ and u^\pm the traces of $\boldsymbol{\tau}$ and u on ∂T^\pm taken from the interior of T^\pm , respectively, we define the average $\{\!\{ \cdot \}\!\}$ and jump $[\![\cdot]\!]$ operators by

$$\{\!\{ v \}\!\} = (v^+ + v^-)/2, \quad \{\!\{ \boldsymbol{\tau} \}\!\} = (\boldsymbol{\tau}^+ + \boldsymbol{\tau}^-)/2, \quad [v] = v^+ \mathbf{n}^+ + v^- \mathbf{n}^-, \quad [\![\boldsymbol{\tau}]\!] = \boldsymbol{\tau}^+ \cdot \mathbf{n}^+ + \boldsymbol{\tau}^- \cdot \mathbf{n}^-.$$

The *weighted* average $\{\!\{ \cdot \}\!\}_\delta$ with a parameter $\delta \in [0, 1]$, is defined as follows:

$$\{\!\{ v \}\!\}_\delta = \delta v^+ + (1 - \delta)v^-, \quad \{\!\{ \boldsymbol{\tau} \}\!\}_\delta = \delta \boldsymbol{\tau}^+ + (1 - \delta)\boldsymbol{\tau}^-.$$

On a boundary face $e \in \mathcal{E}^B$ we set $\{\!\{ v \}\!\} = \{\!\{ v \}\!\}_\delta = v$, $\{\!\{ \boldsymbol{\tau} \}\!\} = \{\!\{ \boldsymbol{\tau} \}\!\}_\delta = \boldsymbol{\tau}$, $[v] = v\mathbf{n}$ and $[\![\boldsymbol{\tau}]\!] = \boldsymbol{\tau} \cdot \mathbf{n}$.

We introduce next the global lifting operators (see [4, 11]):

$$\mathcal{R} : [L^2(\mathcal{E})]^d \longrightarrow \Sigma_h, \quad \int_{\Omega} \mathcal{R}(\boldsymbol{\phi}) \cdot \boldsymbol{\tau} \, dx = - \int_{\mathcal{E}} \boldsymbol{\phi} \cdot \{\!\{ \boldsymbol{\tau} \}\!\} \, ds \quad \forall \boldsymbol{\tau} \in \Sigma_h, \quad (2a)$$

$$\mathcal{L} : L^2(\mathcal{E}^I) \longrightarrow \Sigma_h, \quad \int_{\Omega} \mathcal{L}(q) \cdot \boldsymbol{\tau} \, dx = - \int_{\mathcal{E}^I} q [\![\boldsymbol{\tau}]\!] \, ds \quad \forall \boldsymbol{\tau} \in \Sigma_h. \quad (2b)$$

The local lifting operators $r_e : [L^2(e)]^d \longrightarrow \Sigma_h$ and $l_e : L^2(e) \longrightarrow \Sigma_h$ are defined, respectively, as

$$\int_{\Omega} r_e(\boldsymbol{\phi}) \cdot \boldsymbol{\tau} \, dx = - \int_e \boldsymbol{\phi} \cdot \{\!\{ \boldsymbol{\tau} \}\!\} \, ds, \quad \int_{\Omega} l_e(q) \cdot \boldsymbol{\tau} \, dx = - \int_e q [\![\boldsymbol{\tau}]\!] \, ds \quad \forall \boldsymbol{\tau} \in \Sigma_h. \quad (3)$$

From the definitions of l_e , r_e , the jump and average operators, it follows that for $e \subset \partial T$,

$$l_e(q) = 2r_e(q\mathbf{n}_T) \quad \text{on } T,$$

where \mathbf{n}_T denotes the outward normal unit vector to ∂T . Moreover, note that the supports of $r_e(\boldsymbol{\phi})$ and $l_e(q)$ are confined within the elements T having $e \subset \partial T$, and that

$$\mathcal{R}(\boldsymbol{\phi}) = \sum_{e \in \mathcal{E}} r_e(\boldsymbol{\phi}), \quad \mathcal{L}(q) = \sum_{e \in \mathcal{E}^I} l_e(q). \quad (4)$$

2.2. Discontinuous Galerkin methods for the model problem

Following [4], we start by rewriting the model problem (1) as a first order system of equations:

$$\boldsymbol{\sigma} = \nabla u \quad \text{in } \Omega, \quad -\nabla \cdot \boldsymbol{\sigma} = f \quad \text{in } \Omega, \quad u = 0 \quad \text{on } \partial\Omega. \quad (5)$$

To obtain the weak formulation, we multiply the first two equations of (5) by arbitrary smooth test functions $\boldsymbol{\tau}$ and v , respectively, and integrate by parts formally over an element $T \in \mathcal{T}_h$. Then, we replace the exact solution $(\boldsymbol{\sigma}, u)$ by its approximation $(\boldsymbol{\sigma}_h, u_h)$ in the finite element space $\boldsymbol{\Sigma}_h \times V_h$ associated to \mathcal{T}_h and, denoting by \boldsymbol{n}_T the outward normal unit vector to ∂T , the weak formulation reads: Find $(\boldsymbol{\sigma}_h, u_h) \in \boldsymbol{\Sigma}_h \times V_h$ such that for all $T \in \mathcal{T}_h$:

$$\begin{aligned} \int_T \boldsymbol{\sigma}_h \cdot \boldsymbol{\tau} \, dx &= - \int_T u_h \nabla \cdot \boldsymbol{\tau} \, dx + \int_{\partial T} \hat{u} \boldsymbol{\tau} \cdot \boldsymbol{n}_T \, ds \quad \forall \boldsymbol{\tau} \in [\mathcal{M}^{\ell_h}(T)]^d, \\ \int_T \boldsymbol{\sigma}_h \cdot \nabla v \, dx &= \int_T f v \, dx + \int_{\partial T} \hat{\boldsymbol{\sigma}} \cdot \boldsymbol{n}_T v \, ds \quad \forall v \in \mathcal{M}^{\ell_h}(T), \end{aligned} \quad (6)$$

where $\hat{\boldsymbol{\sigma}} = \hat{\boldsymbol{\sigma}}(\boldsymbol{\sigma}_h, u_h)$ and $\hat{u} = \hat{u}(u_h)$ are the vector and scalar *numerical fluxes*, respectively, to be defined:

$$\hat{\boldsymbol{\sigma}} : [\mathbf{H}^1(\mathcal{T}_h)]^d \times \mathbf{H}^2(\mathcal{T}_h) \longrightarrow [T(\mathcal{E})]^d, \quad \hat{u} : \mathbf{H}^1(\mathcal{T}_h) \longrightarrow T(\mathcal{E}),$$

which are nothing but approximations to the traces of $\boldsymbol{\sigma} = \nabla u$ and u , respectively, on the boundary of T , and whose definition determine completely the DG method. Summation in (6) over all elements $T \in \mathcal{T}_h$ and integration by parts combined with the application of the identity

$$\sum_{T \in \mathcal{T}_h} \int_{\partial T} v \boldsymbol{\tau} \cdot \boldsymbol{n}_T \, ds = \int_{\mathcal{E}} \llbracket v \rrbracket \cdot \{\{\boldsymbol{\tau}\}\} \, ds + \int_{\mathcal{E}^I} \{\{v\}\} \llbracket \boldsymbol{\tau} \rrbracket \, ds, \quad (7)$$

permit us to rewrite the first equation in (6) in the following form:

$$\int_{\Omega} \boldsymbol{\sigma}_h \cdot \boldsymbol{\tau} \, dx = \int_{\Omega} \nabla_h u_h \cdot \boldsymbol{\tau} \, dx + \int_{\mathcal{E}} \llbracket \hat{u} - u_h \rrbracket \cdot \{\{\boldsymbol{\tau}\}\} \, ds + \int_{\mathcal{E}^I} \{\{\hat{u} - u_h\}\} \llbracket \boldsymbol{\tau} \rrbracket \, ds, \quad (8)$$

where ∇_h denotes the elementwise application of the operator ∇ . Note that provided \hat{u} is chosen independently of $\boldsymbol{\sigma}_h$, the variable $\boldsymbol{\sigma}_h$ can actually be eliminated in an element-by-element manner, by using equation (8). Therefore, by rewriting (8) with the lifting operators (2) and noticing that $\nabla_h V_h \subset \boldsymbol{\Sigma}_h$, we have

$$\boldsymbol{\sigma}_h = \boldsymbol{\sigma}_h(u_h) = \nabla_h u_h - \mathcal{R}(\llbracket \hat{u} - u_h \rrbracket) - \mathcal{L}(\{\{\hat{u} - u_h\}\}).$$

Substitution of the above expression in the second equation of (6) and a standard application of (7) yield the *primal formulation* of DG methods:

$$\text{Find } u_h \in V_h : \quad \mathcal{A}_h^{DG}(u_h, v) = \int_{\Omega} f v \, dx \quad \forall v \in V_h,$$

where

$$\begin{aligned} \mathcal{A}_h^{DG}(u_h, v) &= \int_{\Omega} \nabla_h u_h \cdot \nabla_h v \, dx + \int_{\mathcal{E}} \llbracket \hat{u} - u_h \rrbracket \cdot \{\{\nabla_h v\}\} \, ds + \int_{\mathcal{E}^I} \{\{\hat{u} - u_h\}\} \llbracket \nabla_h v \rrbracket \, ds \\ &\quad - \int_{\mathcal{E}} \{\{\hat{\boldsymbol{\sigma}}\}\} \cdot \llbracket v \rrbracket \, ds - \int_{\mathcal{E}^I} \llbracket \hat{\boldsymbol{\sigma}} \rrbracket \cdot \{\{v\}\} \, ds. \end{aligned} \quad (9)$$

TABLE 1. Numerical fluxes on interior faces.

Method	$\widehat{u}(u_h)$	$\widehat{\boldsymbol{\sigma}}(\boldsymbol{\sigma}_h, u_h)$	Symmetry
SIPG [3]	$\{\{u_h\}\}$	$\{\{\nabla_h u_h\}\} - \alpha_e h_e^{-1} \llbracket u_h \rrbracket$	Y
BRMPS [7]	$\{\{u_h\}\}$	$\{\{\nabla_h u_h\}\} + \alpha_e \{\{r_e(\llbracket u_h \rrbracket)\}\}$	Y
SIPG(δ) [42]	$\{\{u_h\}\}_{(1-\delta)}$	$\{\{\nabla_h u_h\}\}_\delta - \alpha_e h_e^{-1} \llbracket u_h \rrbracket$	Y
NIPG [36]	$\{\{u_h\}\} + \llbracket u_h \rrbracket \cdot \mathbf{n}_T$	$\{\{\nabla_h u_h\}\} - \alpha_e h_e^{-1} \llbracket u_h \rrbracket$	N
IIPG [20]	$\{\{u_h\}\} + 1/2 \llbracket u_h \rrbracket \cdot \mathbf{n}_T$	$\{\{\nabla_h u_h\}\} - \alpha_e h_e^{-1} \llbracket u_h \rrbracket$	N
BMMPR [11]	$\{\{u_h\}\}$	$\{\{\boldsymbol{\sigma}_h\}\} + \alpha_e \{\{r_e(\llbracket u_h \rrbracket)\}\}$	Y
LDG [18]	$\{\{u_h\}\} - \boldsymbol{\beta} \cdot \llbracket u_h \rrbracket$	$\{\{\boldsymbol{\sigma}_h\}\} + \boldsymbol{\beta} \llbracket \boldsymbol{\sigma}_h \rrbracket - \alpha_e h_e^{-1} \llbracket u_h \rrbracket$	Y
BZ [5]	$(u_h _T) _{\partial T}$	$-\alpha_e h_e^{-(2\ell_h+1)} \llbracket u_h \rrbracket$	Y
BMMPR 2 [11]	$(u_h _T) _{\partial T}$	$\alpha_e h_e^{-2\ell_h} \{\{r_e(\llbracket u_h \rrbracket)\}\}$	Y

In Table 1 (first and second part) we collect the definitions of the interior numerical fluxes for the DG methods considered in this paper. On boundary faces, the corresponding definitions modify according to [4], Section 3.4. The properties of consistency and conservativity of the numerical fluxes play a key role in the analysis of corresponding DG methods, since they are reflected, respectively, in the *consistency* and *adjoint consistency* of the primal formulation. We refer to [4], Section 3.3, for further details.

The stability of the DG methods is ensured by means of the stabilisation term $\mathcal{S}(\cdot, \cdot)$ that penalizes the jumps over the skeleton of the partition. For the DG methods we consider, it has been defined in two different ways:

$$\mathcal{S}^h(u_h, v) = \sum_{e \in \mathcal{E}} \alpha_e h_e^{-1} \int_e \llbracket u_h \rrbracket \cdot \llbracket v \rrbracket ds, \quad \mathcal{S}^r(u_h, v) = \sum_{e \in \mathcal{E}} \alpha_e \int_\Omega r_e(\llbracket u_h \rrbracket) \cdot r_e(\llbracket v \rrbracket) dx, \quad (10)$$

where $\alpha_e > 0$ is a parameter. We define $\alpha_* = \min_{e \in \mathcal{E}} \alpha_e$ and $\alpha^* = \max_{e \in \mathcal{E}} \alpha_e$, and suppose, for simplicity, that $\alpha^* \geq \alpha_* \geq 1$. To ensure stability, a few methods require a value for α_e sufficiently large. This is the case, for instance, of the SIPG and the SIPG(δ) method, the BRMPS method and the non-symmetric IIPG method. For the others, stability is guaranteed provided that $\alpha_e > 0$. In this paper we mainly focus on consistent and stable DG methods. In Section 7 some discussions and numerical results are included for the two non-consistent methods BZ [5] and BMMPR2 [11] (see Tab. 1, third part) which are characterized by a super penalty stabilisation term. The analysis for these methods will be the subject of future research.

The fact that the numerical fluxes are conservative will have deep implications in the analysis of the two-level methods. Note that, if $\widehat{\boldsymbol{\sigma}}$ is conservative (and so it is defined as a single valued function), as is the case of all the methods we consider, $\llbracket \widehat{\boldsymbol{\sigma}} \rrbracket \equiv 0$ on each $e \in \mathcal{E}$. Similarly if the scalar flux \widehat{u} is conservative, then $\llbracket \widehat{u} \rrbracket \equiv 0$ on each $e \in \mathcal{E}$. A method for which both fluxes are conservative is called completely conservative.

To simplify the notation, in the rest of the paper we drop the subindex h in the scalar and vector-valued functions belonging to V_h and $\boldsymbol{\Sigma}_h$. Notice that no confusion might arise since we shall only deal with discrete functions.

2.3. Theoretical tools

We briefly review the basic tools that we shall require in our subsequent analysis of the two-level Schwarz method. We start with the local inverse and trace inequalities (see [3, 16]): there exist two positive constants C_{inv} , C_t (depending on the shape regularity of the partition \mathcal{T}_h) such that, for all $T \in \mathcal{T}_h$,

$$|\varphi|_{m,T} \leq C_{inv} h_T^{s-m} |\varphi|_{s,T} \quad \forall \varphi \in \mathcal{M}^{\ell_h}(T) \quad \forall s \leq m, \quad (11)$$

and for any $e \in \mathcal{E}$, face of the element $T \in \mathcal{T}_h$,

$$\|\varphi\|_{0,e}^2 \leq C_t^2 (h_e^{-1} \|\varphi\|_{0,T}^2 + h_e |\varphi|_{1,T}^2) \quad \forall \varphi \in \mathbf{H}^1(T), \quad (12)$$

$$\left\| \frac{\partial \varphi}{\partial n} \right\|_{0,e}^2 \leq C_t^2 (h_e^{-1} |\varphi|_{1,T}^2 + h_e |\varphi|_{2,T}^2) \quad \forall \varphi \in \mathbf{H}^2(T), \quad (13)$$

where C_{inv} also depends on the polynomial degree ℓ_h . For $v \in V_h$, we define the following seminorms and norm, respectively:

$$|v|_{1,h}^2 = \sum_{T \in \mathcal{T}_h} |v|_{1,T}^2, \quad |v|_*^2 = \sum_{e \in \mathcal{E}} h_e^{-1} \|\llbracket v \rrbracket\|_{0,e}^2, \\ \|v\|_{\mathbf{H}^1(\Omega, \mathcal{T}_h)}^2 = |v|_{1,h}^2 + |v|_*^2.$$

For the primal bilinear form, continuity and stability can be easily shown over V_h for all the methods that we shall consider, that is: $\exists C_c, C_s > 0$ such that

$$\begin{aligned} \text{Continuity} \quad & |\mathcal{A}_h^{DG}(u, v)| \leq C_c \|u\|_{\mathbf{H}^1(\Omega, \mathcal{T}_h)} \|v\|_{\mathbf{H}^1(\Omega, \mathcal{T}_h)} \quad \forall u, v \in V_h, \\ \text{Coercivity} \quad & \mathcal{A}_h^{DG}(v, v) \geq C_s \|v\|_{\mathbf{H}^1(\Omega, \mathcal{T}_h)}^2 \quad \forall v \in V_h. \end{aligned}$$

The constant C_c only depends on the shape regularity of \mathcal{T}_h , the polynomial degree ℓ_h , an upper bound on the face dependent penalty parameters and, for the LDG method, an upper bound on the function β which enters into the definition of the numerical fluxes.

We next recall some properties of the lifting operators. From the definition (3) of r_e , and after using the trace inequality (12) as in [11], we have

$$C_1 h_e^{-1} \|\phi\|_{0,e}^2 \leq \|r_e(\phi)\|_{0,\Omega}^2 \leq C_2 h_e^{-1} \|\phi\|_{0,e}^2 \quad \forall \phi \in [\mathcal{M}^{\ell_h}(e)]^d, \quad (14)$$

with constants C_1 and C_2 only depending on the minimum angle of the partition and the polynomial degree ℓ_h . Hence by taking $\phi = \llbracket v \rrbracket$ and summing over all faces, we have

$$C_1 |v|_*^2 \leq \sum_{e \in \mathcal{E}} \|r_e(\llbracket v \rrbracket)\|_{0,\Omega}^2 \leq C_2 |v|_*^2. \quad (15)$$

Recalling the identities (4), it follows

$$\|\mathcal{R}(\phi)\|_{0,\Omega}^2 \leq C_{\mathcal{R}} C_2 |\phi|_*^2, \quad (16a)$$

$$\|\mathcal{L}(q)\|_{0,\Omega}^2 \leq 4C_{\mathcal{R}} \sum_{e \in \mathcal{E}^I} \|r_e(q\mathbf{n}_e)\|_{0,\Omega}^2, \quad (16b)$$

where $C_{\mathcal{R}} = 3$ (or 4) if \mathcal{T}_h is made of triangles (or parallelograms) in $d = 2$, $C_{\mathcal{R}} = 4$ (or 6) if \mathcal{T}_h is made of tetrahedra (or parallelepipeds) in $d = 3$.

An important tool in the analysis of Schwarz methods is represented by a Friedrichs-Poincaré type inequality valid for Sobolev spaces. For a given convex polygonal or polyhedral domain $D \subset \Omega \subset \mathbb{R}^d$, $d = 2, 3$, we define $\mathbf{H}^1(D, \mathcal{T}_h) = \{u \in \mathbf{L}^2(\Omega) : u|_T \in \mathbf{H}^1(T), T \in \mathcal{T}_h, T \subset D\}$. We state the following lemma (see [3, 9, 23]).

Lemma 2.2 (Friedrichs-Poincaré inequality). *Let $D \subset \Omega \subset \mathbb{R}^d$, $d = 2, 3$, be a convex polygonal or polyhedral domain that can be covered by the union of some elements in \mathcal{T}_h . Then, there exists a positive constant C , depending only on the geometry of D and the shape-regularity constant of \mathcal{T}_h , such that, for all $u \in \mathbf{H}^1(D, \mathcal{T}_h)$*

with zero average over D , it holds:

$$\|u\|_{0,D}^2 \leq C (\text{diam}(D))^2 \left(\sum_{\substack{T \in \mathcal{T}_h \\ T \subset D}} |u|_{1,T}^2 + \sum_{\substack{e \in \mathcal{E} \\ e \subset D}} h_e^{-1} \|[[u]]\|_{0,e}^2 \right). \quad (17)$$

To conclude, we revise the following trace inequality shown in [23], valid for any bounded, simply-connected star-like domain $D \subset \mathbb{R}^d$ which is the union of some elements in \mathcal{T}_h :

$$\|u\|_{0,\partial D}^2 \leq c \left[\text{diam}(D)^{-1} \|u\|_{0,D}^2 + \text{diam}(D) \left(\sum_{\substack{T \in \mathcal{T}_h \\ T \subset D}} |u|_{1,T}^2 + \sum_{\substack{e \in \mathcal{E}_h \\ e \subset D}} h_e^{-1} \|[[u]]\|_{0,F}^2 \right) \right] \quad \forall u \in \mathbf{H}^1(D, \mathcal{T}_h), \quad (18)$$

where the constant $c > 0$ only depends on D .

3. NON-OVERLAPPING SCHWARZ METHODS

In this section, we present our two-level algorithms for symmetric and non-symmetric DG methods. We start by setting the notation and introducing the assumptions on the partitions. Then we describe the two-level algorithms for the DG methods in an abstract general form and from the algebraic point of view.

We denote by \mathcal{T}_S a family of partitions of Ω into N_s non-overlapping subdomains $\Omega = \bigcup_{i=1}^{N_s} \Omega_i$, and by $\{\mathcal{T}_H\}_{H>0}$ and $\{\mathcal{T}_h\}_{h>0}$ two families of coarse and fine partitions, respectively, with mesh sizes H and h . All the partitions are assumed to be regular and quasi-uniform and we shall always proceed under the assumption that $\mathcal{T}_h, \mathcal{T}_H$ and \mathcal{T}_S are nested:

$$\mathcal{T}_S \subseteq \mathcal{T}_H \subseteq \mathcal{T}_h, \quad (19)$$

i.e., each Ω_i , $i = 1, \dots, N_s$ can be written as the union of some elements D of \mathcal{T}_H , each of which is a union of elements of the finer partition \mathcal{T}_h . For each subdomain Ω_i of \mathcal{T}_S , we denote by \mathcal{E}_i the set of all faces of \mathcal{E} (recall that \mathcal{E} is the set of all faces of the fine triangulation \mathcal{T}_h) in the interior or on the boundary of Ω_i . We also set:

$$\mathcal{E}_i^I = \{e \in \mathcal{E}_i : e \cap \partial\Omega_i = \emptyset\}, \quad \mathcal{E}_i^B = \{e \in \mathcal{E}_i : e \cap \partial\Omega_i \cap \partial\Omega \neq \emptyset\},$$

and

$$\Gamma_{ik} = \{e \in \mathcal{E}_i : e \subset \partial\Omega_i \cap \partial\Omega_k\}, \quad \Gamma_i = \{e \in \mathcal{E}_i : e \subset \partial\Omega_i \setminus \partial\Omega\}, \quad \Gamma = \bigcup_i \Gamma_i.$$

Notice that $\mathcal{E}_i = \mathcal{E}_i^I \cup \mathcal{E}_i^B \cup \Gamma_i$. The elements of the coarse partition \mathcal{T}_H will be denoted by D and we shall designate by E their faces.

For $i = 1, \dots, N_s$, we define the following *local spaces*

$$V_h^i = \{v_h \in \mathbf{L}^2(\Omega_i) : v_h|_T \in \mathcal{M}^{\ell_h}(T) \quad \forall T \in \mathcal{T}_h, T \subseteq \Omega_i\}, \quad \Sigma_h^i = [V_h^i]^d,$$

where \mathcal{M}^{ℓ_h} is defined as before. We define the prolongation operators $R_i^T : V_h^i \rightarrow V_h$ as the injection operators from V_h^i into V_h . The corresponding restriction operators R_i are defined as the transpose of R_i^T with respect to the Euclidean scalar product. For vector-valued functions R_i^T is defined componentwise. With the above definitions, it is straightforward to see that the following space splittings hold: $V_h = R_1^T V_h^1 \oplus \dots \oplus R_{N_s}^T V_h^{N_s}$ and $\Sigma_h = R_1^T \Sigma_h^1 \oplus \dots \oplus R_{N_s}^T \Sigma_h^{N_s}$.

We next introduce our *local-solvers*. Their definition, which differs significantly from the ones introduced in [23, 30], follows by considering the DG approximation of the problem:

$$-\Delta u_i = f|_{\Omega_i} \quad \text{in } \Omega_i, \quad u_i = 0 \quad \text{on } \partial\Omega_i.$$

Therefore, taking into account the primal formulation given in (9), we define the local bilinear forms $\mathcal{A}_i^{DG} : V_h^i \times V_h^i \rightarrow \mathbb{R}$ as

$$\begin{aligned} \mathcal{A}_i^{DG}(u_i, v_i) = & \int_{\Omega_i} \nabla_h u_i \cdot \nabla_h v_i \, dx + \int_{\mathcal{E}_i} [\widehat{u}_i - u_i] \cdot \{\{\nabla_h v_i\}\} \, ds + \int_{\mathcal{E}_i^I} \{\{\widehat{u}_i - u_i\}\} [\nabla_h v_i] \, ds \\ & - \int_{\mathcal{E}_i} \{\{\widehat{\sigma}_i\}\} \cdot [v_i] \, ds - \int_{\mathcal{E}_i^I} [[\widehat{\sigma}_i]] \{\{v_i\}\} \, ds, \end{aligned} \quad (20)$$

where \widehat{u}_i and $\widehat{\sigma}_i$ are the “local” numerical fluxes which are defined as for the global DG method (see Rem. 3.1 below).

Remark 3.1. Notice that since $\mathcal{E}_i = \mathcal{E}_i^I \cup \mathcal{E}_i^B \cup \Gamma_i$, for the local bilinear forms, interior faces are only those belonging to \mathcal{E}_i^I , while boundary faces are those $e \in \mathcal{E}_i^B \cup \Gamma_i$. The local numerical fluxes are defined on $e \in \mathcal{E}_i^I$ as the numerical fluxes of the global DG method on interior faces (see Tab. 1), and on $\mathcal{E}_i^B \cup \Gamma_i$, as the numerical fluxes of the global DG method on boundary faces. Note that any $e \in \Gamma_i$ is an internal face of the global partition, but a boundary face of the local partition.

To complete the description of the local numerical fluxes, we introduce the *local*-local lifting operators $r_e^i : [L^2(e)]^d \rightarrow \Sigma_h^i$ and $l_e^i : L^2(e) \rightarrow \Sigma_h^i$, which are defined according to (2) but on Ω_i , that is

$$\begin{aligned} \int_{\Omega_i} r_e^i(\phi_i) \cdot \tau_i \, dx &= - \int_e \phi_i \cdot \{\{\tau_i\}\} \, ds & \forall \tau_i \in \Sigma_h^i & \quad \forall e \in \mathcal{E}_i^I, \\ \int_{\Omega_i} r_e^i(\phi_i) \cdot \tau_i \, dx &= - \int_e \phi_i \cdot \tau_i \, ds & \forall \tau_i \in \Sigma_h^i & \quad \forall e \in \mathcal{E}_i^B \cup \Gamma_i, \\ \int_{\Omega_i} l_e^i(q_i) \cdot \tau_i \, dx &= - \int_e q_i [[\tau_i]] \, ds & \forall \tau_i \in \Sigma_h^i & \quad \forall e \in \mathcal{E}_i^I. \end{aligned}$$

Analogously to what is done in Section 2.1 we define $\mathcal{R}^i(\phi_i) = \sum_{e \in \mathcal{E}_i} r_e^i(\phi_i)$ and $\mathcal{L}^i(q_i) = \sum_{e \in \mathcal{E}_i^I} l_e^i(q_i)$.

Taking into account the definition of the prolongation operators, the following relation can be established between the local and global scalar numerical fluxes:

$$\widehat{u}_i(u_i) = \widehat{u}(R_i^T u_i) \quad \text{on } e \in \mathcal{E}_i^I \cup \mathcal{E}_i^B \quad \text{and} \quad [[\widehat{u}_i(u_i)]] = [[\widehat{u}(R_i^T u_i)]] \quad \text{on } e \in \mathcal{E}_i.$$

For the vector numerical fluxes, it can be easily seen that except for mixed methods (those in which $\widehat{\sigma}$ depends on σ), $\widehat{\sigma}_i(\sigma_i(u_i), u_i) = \widehat{\sigma}(\sigma(R_i^T u_i), R_i^T u_i)$ on each face $e \in \mathcal{E}_i^I \cup \mathcal{E}_i^B$. For mixed methods, the discrepancy between the definitions of the local lifting operators r_e^i, l_e^i and r_e, l_e , respectively, yields

$$\begin{aligned} \widehat{\sigma}(R_i^T \sigma_i, R_i^T u_i) - \widehat{\sigma}_i(\sigma_i, u_i) &= \{\{ \sum_{e \in \mathcal{E}_i} (r_e([\![R_i^T u_i]\!]) - r_e^i([\![u_i]\!])) \}\} + \{\{ \sum_{e \in \mathcal{E}_i^I \cup \Gamma_i} l_e(\beta \cdot [\![R_i^T u_i]\!]) - \sum_{e \in \mathcal{E}_i^I} l_e^i(\beta \cdot [\![u_i]\!]) \}\} \\ &+ \beta \cdot [\![\sum_{e \in \mathcal{E}_i} (r_e([\![R_i^T u_i]\!]) - r_e^i([\![u_i]\!])) \]\!] \\ &+ \beta \cdot [\![\sum_{e \in \mathcal{E}_i^I \cup \Gamma_i} l_e(\beta \cdot [\![R_i^T u_i]\!]) - \sum_{e \in \mathcal{E}_i^I} l_e^i(\beta \cdot [\![u_i]\!]) \]\!], \quad e \in \mathcal{E}_i^I \cup \mathcal{E}_i^B, \end{aligned}$$

with $\boldsymbol{\beta} \neq \mathbf{0}$ on $e \in \mathcal{E}_i^I$ only for the LDG method. Furthermore, it is not hard to see that for any $e \in \mathcal{E}_i^I \cup \mathcal{E}_i^B$, far from Γ_i , equality between the global and local vector numerical fluxes holds, *i.e.*, $\widehat{\boldsymbol{\sigma}}_i(\boldsymbol{\sigma}_i(u_i), u_i) = \widehat{\boldsymbol{\sigma}}(\boldsymbol{\sigma}(R_i^T u_i), R_i^T u_i)$. Note that, by definition, the properties of consistency and conservativity of the global numerical fluxes are inherited by the local ones.

By taking into account the fact that for all the DG methods considered the vector flux $\widehat{\boldsymbol{\sigma}}$ is conservative and using the definition of the jump and average operators on the boundary faces, the expression in (20) can be rewritten as

$$\begin{aligned} \mathcal{A}_i^{DG}(u_i, v_i) &= \int_{\Omega_i} \nabla_h u_i \cdot \nabla_h v_i \, dx - [1 - \gamma_{[\widehat{u}]}] \int_{\mathcal{E}_i^I} [[u_i]] \cdot \{\{\nabla_h v_i\}\} \, ds + \int_{\mathcal{E}_i^I} \{\{\widehat{u}_i - u_i\}\} [[\nabla_h v_i]] \, ds - \int_{\mathcal{E}_i^I} \{\{\widehat{\boldsymbol{\sigma}}_i\}\} \cdot [[v_i]] \, ds \\ &\quad - [1 - \gamma_{[\widehat{u}]}] \int_{\mathcal{E}_i^B} u_i \mathbf{n}_i \cdot \nabla_h v_i \, ds - \int_{\mathcal{E}_i^B} \widehat{\boldsymbol{\sigma}}_i \cdot v_i \mathbf{n}_i \, ds - [1 - \gamma_{[\widehat{u}]}] \int_{\Gamma_i} u_i \mathbf{n}_i \cdot \nabla_h v_i \, ds - \int_{\Gamma_i} \widehat{\boldsymbol{\sigma}}_i \cdot v_i \mathbf{n}_i \, ds, \end{aligned}$$

where $\gamma_{[\widehat{u}]}$ is zero for completely conservative methods (*i.e.*, those having not only $[\widehat{\boldsymbol{\sigma}}] = 0$ but also $[\widehat{u}] = 0$), while $\gamma_{[\widehat{u}]} = 2$ for the NIPG method, and $\gamma_{[\widehat{u}]} = 1$ for the IIPG method. The parameter $\gamma_{[\widehat{u}]}$ appears naturally by noting that, for all DG methods considered here (see Tab. 1 first and second parts), the function $[\widehat{u}(u)]$ might be written as $[\widehat{u}(u)] = \gamma [[u]]$ for some constant γ , which is nothing but what we have denoted by $\gamma_{[\widehat{u}]}$. In particular, for $\gamma_{[\widehat{u}]} \neq 0$, the resulting bilinear form is no longer symmetric.

Unlike to [23] and [30], the resulting local solvers (20) are approximate¹ rather than exact solvers. Nevertheless, as it is shown in next section they are *locally stable*; that is, $\mathcal{A}_h^{DG}(R_i^T u, R_i^T u) \leq \omega \mathcal{A}_i^{DG}(u_i, u_i)$ with $0 < \omega \neq 1$.

A key issue in the analysis of Schwarz methods concerns the relation between the sum of the local solvers and the global bilinear form. Along the lines of [24], we can prove the following result.

Proposition 3.2. *For any $u, v \in V_h$, let $u_i, v_i \in V_h^i$, $i = 1, \dots, N_s$, be the (unique) functions such that $u = \sum_{i=1}^{N_s} R_i^T u_i$, $v = \sum_{i=1}^{N_s} R_i^T v_i$. Then, the following identity holds:*

$$\mathcal{A}_h^{DG}(u, v) = \sum_{i=1}^{N_s} \mathcal{A}_i^{DG}(u_i, v_i) + I_h^{DG}(u, v),$$

where

$$\begin{aligned} I_h^{DG}(u, v) &= (1 - \gamma_{[\widehat{u}]}) \left[- \int_{\Gamma} u_i \mathbf{n}_i \cdot \nabla_h v_j \, ds - \int_{\Gamma} u_j \mathbf{n}_j \cdot \nabla_h v_i \, ds + \int_{\Gamma} [[u]] \cdot \{\{\nabla_h v\}\} \, ds \right] \\ &\quad \left[- \int_{\Gamma} \nabla_h u_i \cdot v_j \mathbf{n}_j \, ds - \int_{\Gamma} \nabla_h u_j \cdot v_i \mathbf{n}_i \, ds + \int_{\Gamma} \{\{\nabla_h u\}\} \cdot [[v]] \, ds \right] + \mathcal{G}_{\Gamma}(u, v) + \mathcal{H}(u, v) + \mathcal{F}_{\Gamma}(u, v), \quad (21) \end{aligned}$$

with $\mathcal{G}_{\Gamma}(u, v) = \sum_{e \in \Gamma} \int_e \alpha_e h_e^{-1} (u_i \mathbf{n}_i \cdot v_j \mathbf{n}_j + u_j \mathbf{n}_j \cdot v_i \mathbf{n}_i) \, ds$, if $\mathcal{S}(\cdot, \cdot) = \mathcal{S}^h(\cdot, \cdot)$, and

$$\mathcal{G}_{\Gamma}(u, v) = \sum_{e \in \Gamma} \left[\int_{\Omega} \alpha_e r_e^e([u]) \cdot r_e([v]) \, dx - \int_{\Omega_i} \alpha_e r_e^i([u_i]) \cdot r_e^i([v_i]) \, dx - \int_{\Omega_j} \alpha_e r_e^j([u_j]) \cdot r_e^j([v_j]) \, dx \right],$$

¹Approximate in the sense that $\mathcal{A}_h^{DG}(R_i^T u, R_i^T u) \neq \mathcal{A}_i^{DG}(u_i, u_i)$.

if $\mathcal{S}(\cdot, \cdot) = \mathcal{S}^r(\cdot, \cdot)$. Moreover, for all methods in mixed form $\mathcal{H}(\cdot, \cdot)$ is defined as

$$\begin{aligned} \mathcal{H}(u, v) &= \int_{\Omega} (\mathcal{R}(\llbracket u \rrbracket) + \mathcal{L}(\boldsymbol{\beta} \cdot \llbracket u \rrbracket)) \cdot (\mathcal{R}(\llbracket v \rrbracket) + \mathcal{L}(\boldsymbol{\beta} \cdot \llbracket v \rrbracket)) \, dx \\ &\quad - \sum_{i=1}^{N_s} \int_{\Omega_i} (\mathcal{R}^i(\llbracket u_i \rrbracket) + \mathcal{L}^i(\boldsymbol{\beta} \cdot \llbracket u_i \rrbracket)) \cdot (\mathcal{R}^i(\llbracket v_i \rrbracket) + \mathcal{L}^i(\boldsymbol{\beta} \cdot \llbracket v_i \rrbracket)) \, dx, \end{aligned}$$

with $\boldsymbol{\beta} \neq \mathbf{0}$ only for the LDG method while $\mathcal{H}(u, v) = 0$ for all the methods in primal form. Finally,

$$\mathcal{F}_{\Gamma}(u, v) = - \int_{\Gamma} (\tilde{\boldsymbol{\beta}} \cdot \llbracket u \rrbracket \llbracket \nabla_h v \rrbracket + \tilde{\boldsymbol{\beta}} \llbracket \nabla_h u \rrbracket \cdot \llbracket v \rrbracket) \, ds,$$

where $\tilde{\boldsymbol{\beta}} = \boldsymbol{\beta}$ for the LDG method, $\tilde{\boldsymbol{\beta}} = -\llbracket 1/2 - \delta \rrbracket / 2$ for the SIPG(δ) method and $\tilde{\boldsymbol{\beta}} = \mathbf{0}$ otherwise.

Proof. We briefly sketch the idea of the proof, which is derived along the same lines of [23], Lemma 4.2. For the sake of simplicity we restrict ourselves to the case of two subdomains $\bar{\Omega} = \bar{\Omega}_1 \cup \bar{\Omega}_2$. Using the expressions of the primal bilinear forms \mathcal{A}_h^{DG} (9) and of \mathcal{A}_i^{DG} , $i = 1, 2$, (20), it is straightforward to see that

$$\begin{aligned} \mathcal{A}_h^{DG}(u, v) - \mathcal{A}_1^{DG}(u_1, v_1) - \mathcal{A}_2^{DG}(u_2, v_2) &= \int_{\Gamma} (\llbracket \hat{u} - u \rrbracket \cdot \llbracket \nabla_h v \rrbracket - \llbracket \hat{u}_1 - u_1 \rrbracket \cdot \llbracket \nabla_h v_1 \rrbracket - \llbracket \hat{u}_2 - u_2 \rrbracket \cdot \llbracket \nabla_h v_2 \rrbracket) \, ds \\ &\quad + \int_{\Gamma} \llbracket \hat{u} - u \rrbracket \llbracket \nabla_h v \rrbracket \, ds - \int_{\mathcal{E}} \llbracket \hat{\boldsymbol{\sigma}} \rrbracket \cdot \llbracket v \rrbracket \, ds + \int_{\mathcal{E}_1} \llbracket \hat{\boldsymbol{\sigma}}_1 \rrbracket \cdot \llbracket v_1 \rrbracket \, ds + \int_{\mathcal{E}_2} \llbracket \hat{\boldsymbol{\sigma}}_2 \rrbracket \cdot \llbracket v_2 \rrbracket \, ds, \quad (22) \end{aligned}$$

where we have taken into account that $\llbracket \hat{\boldsymbol{\sigma}} \rrbracket$, $\llbracket \hat{\boldsymbol{\sigma}}_1 \rrbracket$ and $\llbracket \hat{\boldsymbol{\sigma}}_2 \rrbracket$ vanish for all the considered DG methods. For the first integral in (22), since $\llbracket \nabla_h v \rrbracket = (\nabla_h v_1 + \nabla_h v_2)/2$, $\llbracket \nabla_h v_i \rrbracket = \nabla_h v_i$, ($i = 1, 2$) and

$$\begin{aligned} \llbracket \hat{u} - u \rrbracket &= - (1 - \gamma_{\llbracket \hat{u} \rrbracket}) \llbracket u \rrbracket = - (1 - \gamma_{\llbracket \hat{u} \rrbracket}) (u_1 \mathbf{n}_1 + u_2 \mathbf{n}_2), \\ \llbracket \hat{u}_i - u_i \rrbracket &= - (1 - \gamma_{\llbracket \hat{u} \rrbracket}) \llbracket u_i \rrbracket = - (1 - \gamma_{\llbracket \hat{u} \rrbracket}) u_i \mathbf{n}_i, \quad i = 1, 2, \end{aligned}$$

we immediately have

$$\begin{aligned} \int_{\Gamma} (\llbracket \hat{u} - u \rrbracket \cdot \llbracket \nabla_h v \rrbracket - \llbracket \hat{u}_1 - u_1 \rrbracket \cdot \llbracket \nabla_h v_1 \rrbracket - \llbracket \hat{u}_2 - u_2 \rrbracket \cdot \llbracket \nabla_h v_2 \rrbracket) \, ds &= \\ (1 - \gamma_{\llbracket \hat{u} \rrbracket}) \left[- \int_{\Gamma} u_1 \mathbf{n}_1 \cdot \nabla_h v_2 \, ds - \int_{\Gamma} u_2 \mathbf{n}_2 \cdot \nabla_h v_1 \, ds + \int_{\Gamma} \llbracket u \rrbracket \cdot \llbracket \nabla_h v \rrbracket \, ds \right]. \end{aligned}$$

For the second integral, one can proceed in a similar way and, by using that, on each internal face $e \in \mathcal{E}_h^I$, $\llbracket u \rrbracket_{\delta} = \llbracket u \rrbracket - \llbracket 1/2 - \delta \rrbracket \cdot \llbracket u \rrbracket / 2$ and $\llbracket u \rrbracket_{1-\delta} = \llbracket u \rrbracket + \llbracket 1/2 - \delta \rrbracket \cdot \llbracket u \rrbracket / 2$, we get

$$\int_{\Gamma} \llbracket \hat{u} - u \rrbracket \llbracket \nabla_h v \rrbracket \, ds = - \int_{\Gamma} \tilde{\boldsymbol{\beta}} \cdot \llbracket u \rrbracket \llbracket \nabla_h v \rrbracket \, ds,$$

where $\tilde{\boldsymbol{\beta}} = \boldsymbol{\beta}$ for the LDG method, $\tilde{\boldsymbol{\beta}} = -\llbracket 1/2 - \delta \rrbracket / 2$ for the SIPG(δ) method and $\tilde{\boldsymbol{\beta}} = \mathbf{0}$ for all the others DG methods. Finally, taking into account all the possible definitions of the numerical fluxes, all the remaining terms can be written as

$$- \int_{\Gamma} \nabla_h u_1 \cdot v_2 \mathbf{n}_2 \, ds - \int_{\Gamma} \nabla_h u_2 \cdot v_1 \mathbf{n}_1 \, ds + \int_{\Gamma} \llbracket \nabla_h u \rrbracket \cdot \llbracket v \rrbracket \, ds + \mathcal{G}_{\Gamma}(u, v) + \mathcal{H}(u, v) - \int_{\Gamma} \tilde{\boldsymbol{\beta}} \llbracket \nabla_h u \rrbracket \cdot \llbracket v \rrbracket \, ds. \quad \square$$

The last step in the definition of the Schwarz method is the construction of the *coarse solver*. For a given approximation order $0 \leq \ell_H \leq \ell_h$, the coarse spaces are defined as

$$V_h^0 = V_H = \{v_H \in L^2(\Omega) : v_H|_D \in \mathcal{M}^{\ell_H}(D), \quad \forall D \in \mathcal{T}_H\}, \quad \Sigma_h^0 = \Sigma_H = [V_h^0]^d.$$

In view of (19), it is straightforward to note that, with the above definitions, V_h^0 and Σ_h^0 are subspaces of V_h and Σ_h , respectively. The prolongation operator $R_0^T : V_h^0 \rightarrow V_h$, is defined as the natural injection operator and R_0 is its transpose with respect to the Euclidean scalar product. The coarse solver $\mathcal{A}_0^{DG} : V_h^0 \times V_h^0 \rightarrow \mathbb{R}$ is defined as the restriction of \mathcal{A}_h^{DG} to $V_h^0 \times V_h^0$:

$$\mathcal{A}_0^{DG}(u_0, v_0) = \mathcal{A}_h^{DG}(R_0^T u_0, R_0^T v_0) \quad \forall u_0, v_0 \in V_h^0 = V_H. \quad (23)$$

We wish to stress that, $\mathcal{A}_0^{DG}(u_0, v_0) = \mathcal{A}_h^{DG}(R_0^T u_0, R_0^T v_0) \neq \mathcal{A}_H^{DG}(u_0, v_0)$, where $\mathcal{A}_H^{DG}(u_0, v_0)$ is the bilinear form (9) built on the coarse partition instead of the fine partition.

3.1. Algebraic formulation and projection operators

To define the additive Schwarz operator, we introduce the following projection-like operators:

$$P_i = R_i^T \tilde{P}_i : V_h \rightarrow R_i^T V_h^i \subset V_h, \quad i = 0, \dots, N_s, \quad (24)$$

where the operators $\tilde{P}_i : V_h \rightarrow V_h^i$ are defined by

$$\mathcal{A}_i^{DG}(\tilde{P}_i u, v_i) = \mathcal{A}_h^{DG}(u, R_i^T v_i) \quad \forall v_i \in V_h^i. \quad (25)$$

The coercivity of the local and coarse bilinear forms \mathcal{A}_i^{DG} guarantees that the operators \tilde{P}_i , and therefore P_i , are well defined. We define the additive Schwarz operator by

$$P_{ad} = \sum_{i=0}^{N_s} P_i. \quad (26)$$

This can be seen as a preconditioned linear system. More precisely, let \mathbb{A} , \mathbb{A}_i , \mathbb{A}_0 and \mathbb{R}_i^T be the matrix representation of \mathcal{A}_h^{DG} , \mathcal{A}_i^{DG} , \mathcal{A}_0^{DG} and R_i^T , respectively; and let \mathbf{u} and \mathbf{f} be the vector representation of the unknown function u and of the right hand side f . Then, the matrix representation \mathbb{P}_{ad} of P_{ad} is given by

$$\mathbb{P}_{ad} = \sum_{i=0}^{N_s} \mathbb{P}_i = \sum_{i=0}^{N_s} [R_i^T \mathbb{A}_i^{-1} R_i] \mathbb{A} = \mathbb{B} \mathbb{A}. \quad (27)$$

The matrix \mathbb{B} is called Schwarz preconditioner. The preconditioned system to be solved is then

$$\mathbb{B} \mathbb{A} \mathbf{u} = \mathbb{B} \mathbf{f}.$$

We wish to emphasize that in contrast to the general situation in the classical theory of Schwarz methods for the Laplacian operator, only for completely conservative DG methods, the local, coarse and global solvers are symmetric. Therefore, to solve the preconditioned linear system one needs to resort to different iterative techniques for symmetric and non-symmetric DG methods. For the former class, the PCG method will be the optimal choice, while for the latter class we shall consider the GMRES and the BiCGSTAB methods.

4. STABLE DECOMPOSITION AND LOCAL STABILITY

Although in our analysis of the two-level Schwarz methods, we have to distinguish between symmetric and non-symmetric methods, a common step is to ensure that a *stable splitting* can be found for the family of local spaces and the corresponding bilinear forms (see Prop. 4.1 below), and that, since the local solvers are approximate, a *local stability* property holds true (see Lem. 4.4 below). In this section we show these results in a unified way for both symmetric and non-symmetric DG methods.

Proposition 4.1 (stable decomposition). *Let \mathcal{A}_h^{DG} be the bilinear form of a stable and consistent DG method. For any $u \in V_h$, let $u = \sum_{i=0}^{N_s} R_i^T u_i$, $u_i \in V_h^i$, $i = 0, \dots, N_s$, where $u_0 \in V_h^0 = V_H$ is defined by*

$$u_0|_D = \frac{1}{\text{meas}(D)} \int_D u \, dx \quad \forall D \in \mathcal{T}_H,$$

and u_1, \dots, u_{N_s} are determined (uniquely) by $u - R_0^T u_0 = R_1^T u_1 + \dots + R_{N_s}^T u_{N_s}$. Then,

$$\sum_{i=0}^{N_s} \mathcal{A}_i^{DG}(u_i, u_i) \leq \alpha^* C_0^2 \mathcal{A}_h^{DG}(u, u), \quad \text{with} \quad C_0^2 = \mathcal{O}\left(\frac{H}{h}\right) \quad \text{and} \quad \alpha^* = \max_{e \in \mathcal{E}} \alpha_e.$$

For the proof of Proposition 4.1 we require some technical results. In view of Proposition 3.2, the first one is an estimate for the interface bilinear form $I_h^{DG}(\cdot, \cdot)$.

Lemma 4.2. *Let \mathcal{A}_h^{DG} be the bilinear form of a stable and consistent DG method. For any $u \in V_h$, let $u_i \in V_h^i$, $i = 1, \dots, N_s$, be the (unique) functions such that $u = \sum_{i=1}^{N_s} R_i^T u_i$. Then, there exist two constants $C_1, C_2 > 0$ only depending on the shape regularity of \mathcal{T}_h , the polynomial degree and, for the LDG method, an upper bound on the function β which enters into the definition of the numerical fluxes, such that for all $u \in V_h$,*

$$|I_h^{DG}(u, u)| \leq C_1 \alpha^* \left[\sum_{D \in \mathcal{T}_H} H_D^{-1} h^{-1} \|u\|_{L^2(D)}^2 \right] + C_2 \alpha^* H h^{-1} \mathcal{A}_h^{DG}(u, u),$$

where $H_D = \text{diam}(D)$ and $\alpha^* = \max_{e \in \mathcal{E}} \alpha_e$.

The proof of the Lemma 4.2 is technical and so we postpone it to the Appendix. The second result concerns the coarse solver.

Lemma 4.3. *Let \mathcal{A}_h^{DG} be the bilinear form of a stable and consistent DG method. For any $u \in V_h$, let $u_0 \in V_h^0 = V_H$ be defined by the elementwise average of u on \mathcal{T}_H . Then, there exists a constant $C > 0$ only depending on the shape regularity of the partition \mathcal{T}_h , the polynomial degree and, for the LDG method, an upper bound on the function β which enters into the definition of the numerical fluxes, such that*

$$\mathcal{A}_h^{DG}(R_0^T u_0, R_0^T u_0) \leq C \alpha^* (1 + H h^{-1}) \mathcal{A}_h^{DG}(u, u),$$

where $\alpha^* = \max_{e \in \mathcal{E}} \alpha_e$.

Proof. To simplify the notation, we set $\tilde{u}_0 = R_0^T u_0$ along the proof. We treat separately methods in primal and mixed form. For the DG methods in primal form, note that since u_0 is piecewise constant on \mathcal{T}_H , so is \tilde{u}_0 on \mathcal{T}_h , and therefore all terms in $\mathcal{A}_h^{DG}(\tilde{u}_0, \tilde{u}_0)$ vanish except for the stability term $\mathcal{S}(\tilde{u}_0, \tilde{u}_0)$. Furthermore, in view of the norm equivalence (15), it is enough to bound the term $\mathcal{S}^h(\tilde{u}_0, \tilde{u}_0)$, for which by adding and subtracting $u \in V_h$ we find

$$\mathcal{S}^h(\tilde{u}_0, \tilde{u}_0) \leq 2\mathcal{S}^h(\tilde{u}_0 - u, \tilde{u}_0 - u) + 2\alpha^* |u|_*^2.$$

To estimate the first term in the sum above, notice that $[[\tilde{u}_0]] \equiv 0$ on those faces internal to some coarse element $D \in \mathcal{T}_H$ (i.e., on $e \in \mathcal{E}$, $e \subset D$), while if $e \subset \partial D$, $[[\tilde{u}_0]]$ will not vanish, in general. Thus, we have

$$\mathcal{S}^h(\tilde{u}_0 - u, \tilde{u}_0 - u) = \sum_{e \in \mathcal{E}} \alpha_e h_e^{-1} \|[[\tilde{u}_0 - u]]\|_{0,e}^2 \leq \mathcal{S}_1 + \mathcal{S}_2 + \mathcal{S}_3,$$

where

$$\begin{aligned} \mathcal{S}_1 &= \sum_{D \in \mathcal{T}_H} \sum_{e \subset D} \alpha_e h_e^{-1} \|[[u]]\|_{0,e}^2, & \mathcal{S}_2 &= \sum_{D \in \mathcal{T}_H} \sum_{\substack{e \subset \partial D \\ \partial D \cap \partial \Omega = \emptyset}} \alpha_e h_e^{-1} \|[[\tilde{u}_0 - u]]\|_{0,e}^2, \\ \mathcal{S}_3 &= \sum_{D \in \mathcal{T}_H} \sum_{\substack{e \subset \partial D \\ \partial D \cap \partial \Omega \neq \emptyset}} \alpha_e h_e^{-1} \|\tilde{u}_0 - u\|_{0,e}^2, \end{aligned}$$

and where we have already used the definition of the jump operator on boundary faces. The term \mathcal{S}_1 is directly bounded by $\mathcal{S}_1 \leq C\alpha^* |u|_*^2$. For the second term, noting that $[[v]]^2 = (v^+ \mathbf{n}^+ + v^- \mathbf{n}^-)^2 \leq 2(|v^+|^2 + |v^-|^2)$, together with the estimate (44) in the Appendix, we have

$$\mathcal{S}_2 \leq 2 \sum_{D \in \mathcal{T}_H} \sum_{\substack{e \subset \partial D \\ \partial D \cap \partial \Omega = \emptyset}} \alpha_e h_e^{-1} \left(\|\tilde{u}_0^+ - u^+\|_{0,e}^2 + \|\tilde{u}_0^- - u^-\|_{0,e}^2 \right) \leq C\alpha^* \sum_{\substack{D \in \mathcal{T}_H \\ \partial D \cap \partial \Omega = \emptyset}} h^{-1} \|\tilde{u}_0 - u\|_{0,\partial D}^2.$$

Then, arguing for \mathcal{S}_3 similarly (i.e., using estimate (44) in the Appendix), and taking into account the above estimate together with the trace inequality (18) and the Poincaré inequality (17), gives

$$\begin{aligned} \mathcal{S}_2 + \mathcal{S}_3 &\leq C\alpha^* \sum_{D \in \mathcal{T}_H} h^{-1} \|\tilde{u}_0 - u\|_{0,\partial D}^2 \\ &\leq C\alpha^* \sum_{D \in \mathcal{T}_H} \left[H_D^{-1} h^{-1} \|\tilde{u}_0 - u\|_{0,D}^2 + H_D h^{-1} \left(\sum_{\substack{T \in \mathcal{T}_h \\ T \subset D}} |u|_{1,T}^2 + \sum_{\substack{e \in \mathcal{E} \\ e \subset D}} h_e^{-1} \|[[u]]\|_{0,e}^2 \right) \right] \\ &\leq C\alpha^* H h^{-1} \mathcal{A}_h^{DG}(u, u). \end{aligned}$$

Therefore, we finally have

$$\mathcal{A}_h^{DG}(\tilde{u}_0, \tilde{u}_0) \leq 2\mathcal{S}^h(\tilde{u}_0 - u, \tilde{u}_0 - u) + 2\alpha^* |u|_*^2 \leq C\alpha^* (|u|_*^2 + H h^{-1} \mathcal{A}_h^{DG}(u, u)).$$

For the DG methods in mixed form, we have

$$\mathcal{A}_h(\tilde{u}_0, \tilde{u}_0) = \int_{\Omega} |\mathcal{R}([[\tilde{u}_0]]) + \eta \mathcal{L}(\boldsymbol{\beta} \cdot [[\tilde{u}_0]])|^2 dx + \mathcal{S}(\tilde{u}_0, \tilde{u}_0),$$

where the stability term is either \mathcal{S}^h or \mathcal{S}^r already estimated, and $\eta = 0$ for the BMMPR method, while $\eta = 1$ for the LDG method. Arguing as before, and using the relations (16a) and (16b) together with norm equivalence (15), one easily gets

$$\int_{\Omega} |\mathcal{R}([[\tilde{u}_0]]) + \eta \mathcal{L}(\boldsymbol{\beta} \cdot [[\tilde{u}_0]])|^2 dx \leq C |\tilde{u}_0|_*^2 \leq C \mathcal{S}^h(\tilde{u}_0, \tilde{u}_0),$$

where the constant C also depends on the parameter η and, for the LDG method, an upper bound on the function $\boldsymbol{\beta}$ which enters into the definition of the numerical fluxes. Using the previous estimates of $\mathcal{S}^h(\tilde{u}_0, \tilde{u}_0)$ concludes the proof. \square

We are now ready to prove Proposition 4.1.

Proof of Proposition 4.1. Given $u \in V_h$, we decompose $u - R_0^T u_0$ as $\sum_{i=1}^{N_s} R_i^T u_i$. As in the proof of Lemma 4.3, we shall denote by $\tilde{u}_0 = R_0^T u_0$. By virtue of Proposition 3.2 and the definition of the coarse solver, we can write

$$\sum_{i=0}^{N_s} \mathcal{A}_i^{DG}(u_i, u_i) = \mathcal{A}_h^{DG}(u - \tilde{u}_0, u - \tilde{u}_0) + \mathcal{A}_0^{DG}(u_0, u_0) - I_h^{DG}(u - \tilde{u}_0, u - \tilde{u}_0).$$

Hence, owing to the definition of \mathcal{A}_0^{DG} , the second term at the right-hand side is readily estimated by using Lemma 4.3. For the first term, by using the Cauchy Schwarz inequality² and Lemma 4.3, we find

$$\mathcal{A}_h^{DG}(u - \tilde{u}_0, u - \tilde{u}_0) \leq 2[\mathcal{A}_h^{DG}(u, u) + \mathcal{A}_h^{DG}(\tilde{u}_0, \tilde{u}_0)] \leq 2C\alpha^*(2 + Hh^{-1})\mathcal{A}_h^{DG}(u, u). \quad (28)$$

As regards the term involving the interface bilinear form, by using estimate (45) in the Appendix, together with the trace inequality (18) and taking into account (28), we find

$$\begin{aligned} |I_h^{DG}(u - \tilde{u}_0, u - \tilde{u}_0)| &\leq C\alpha^* \left[\sum_{D \in \mathcal{T}_H} H_D^{-1} h^{-1} \|u - \tilde{u}_0\|_{0,D}^2 \right] + C\alpha^* Hh^{-1} \mathcal{A}_h^{DG}(u, u) \\ &\leq C\alpha^* Hh^{-1} \mathcal{A}_h^{DG}(u, u), \end{aligned}$$

where the last step follows from the Poincaré inequality (17). \square

We now show a local stability property that gives a one-sided measure of the approximation properties of the local bilinear forms.

Lemma 4.4 (Local stability). *Let \mathcal{A}_h^{DG} be the bilinear form of a stable and consistent DG method. There exists $C_\omega > 0$ such that, for all $u_i \in V_h^i$*

$$\mathcal{A}_h^{DG}(R_i^T u_i, R_i^T u_i) \leq \omega \mathcal{A}_i^{DG}(u_i, u_i), \quad \omega = 1 + C_\omega, \quad i = 1, \dots, N_s, \quad (29)$$

where C_ω only depends on the shape regularity of \mathcal{T}_h , the polynomial degree and, for the LDG method, an upper bound on the function β which enters into the definition of the numerical fluxes.

The proof of the Lemma 4.4 is technical and so it is given in the Appendix.

5. RATES OF CONVERGENCE FOR SYMMETRIC DG METHODS

In this section, following the general abstract convergence theory of Schwarz methods [32–34], we give an estimate for the condition number of P_{ad} , corresponding to any of the considered symmetric DG methods.

Theorem 5.1. *Let \mathcal{A}_h^{DG} be the primal bilinear form of a symmetric DG method. Let P_{ad} be the additive Schwarz operator defined in (26). Then, the condition number $\kappa(P_{ad})$ satisfies*

$$\kappa(P_{ad}) \leq \alpha^* C_0^2 (1 + \omega[1 + N_c]) \leq C\alpha^* (1 + \omega[1 + N_c]) \frac{H}{h}, \quad (30)$$

where $\alpha^* = \max_{e \in \mathcal{E}} \alpha_e$, $C_0^2 = O(Hh^{-1})$ is the stable decomposition constant of Proposition 4.1, ω is the local stability constant given in Lemma 4.4 and N_c denotes the maximum number of adjacent subdomains a given subdomain can have.

²Notice that, for non-symmetric DG approximations, $\mathcal{A}_h^{DG}(v, v)$ is indeed symmetric for all $v \in V_h$.

Proof. Following [43], Chapter 2, we need to show the following three properties:

- (i) *Stable decomposition:* there exists a constant C_0 such that every $u \in V_h$ admits a decomposition $u = \sum_{i=0}^{N_s} R_i^T u_i$, with $u_i \in V_i \forall i = 0, \dots, N_s$, satisfying

$$\sum_{i=0}^{N_s} \mathcal{A}_i^{DG}(u_i, u_i) \leq C_0^2 \mathcal{A}_h^{DG}(u, u);$$

- (ii) *Strengthened Cauchy-Schwarz inequalities:* there exist constants $0 \leq \varepsilon_{ij} \leq 1$, $1 \leq i, j \leq N_s$ such that

$$|\mathcal{A}_h^{DG}(R_i^T u_i, R_j^T u_j)| \leq \varepsilon_{ij} \mathcal{A}_h^{DG}(R_i^T u_i, R_i^T u_i)^{1/2} \mathcal{A}_h^{DG}(R_j^T u_j, R_j^T u_j)^{1/2}, \quad (31)$$

for $u_i \in V_h^i$ and $u_j \in V_h^j$. We denote by $\rho(\mathcal{E})$ the spectral radius of $\mathcal{E} = \{\varepsilon_{ij}\}_{1 \leq i, j \leq N_s}$;

- (iii) *Local stability:* there exists $\omega > 0$ such that

$$\mathcal{A}_h^{DG}(R_i^T u_i, R_i^T u_i) \leq \omega \mathcal{A}_i^{DG}(u_i, u_i) \quad \forall u_i : u_i \in \text{range}(\tilde{P}_i) \subset V_h^i.$$

Then, (i) implies that $\lambda_{\min}(P_{ad})$ is bounded from below by C_0^{-2} , (ii) and (iii) imply that $\lambda_{\max}(P_{ad})$ is bounded from above by $\omega\rho(\mathcal{E}) + 1$. The proof of (i) is a consequence of Proposition 4.1. Lemma 4.4 ensure that (iii) is satisfied. As far as (ii) is concerned, note that, from the Cauchy-Schwarz inequality, estimate (31) trivially holds with $\varepsilon_{ij} = 1$. Furthermore, by our definition of the local solvers and local subspaces, it is straightforward to see that the diagonal elements ε_{ii} , are always equal to 1, for $i = 1, \dots, N_s$, and for $i \neq j$, noting that $\mathcal{A}_h^{DG}(R_i^T u_i, R_j^T u_j)$ vanish everywhere except on $\Gamma_{ij} \subset \Gamma$, we find $\mathcal{A}_h^{DG}(R_i^T u_i, R_j^T u_j) \neq 0$ if $\text{meas}(\Gamma_{ij}) \neq 0$, and so we may set $\varepsilon_{ij} = 1$ for those cases and zero otherwise. Hence, from Geršgorin's theorem (see [45]), we find

$$\rho(\mathcal{E}) = \max_i |\lambda_i(\mathcal{E})| \leq \max_i \sum_{j=1}^{N_s} |\varepsilon_{ij}| \leq 1 + N_c,$$

where N_c is the maximum number of adjacent subdomains that a given subdomain Ω_i might have. Hence, the proof of the theorem is complete. \square

We wish to stress that, even for unusual partitions \mathcal{T}_S , N_c is bounded by a moderate constant, say 4 or 6.

Remark 5.2. It should be noticed that the condition number of the preconditioned system depends linearly on the maximum value of the penalty parameters; this will be confirmed by numerical experiments in Section 7.1.

6. THE CASE OF NON-SYMMETRIC DG METHODS

In this section we discuss in detail some additive Schwarz preconditioners and we deal with the analysis for non-symmetric DG approximations of model problem (1). Due to the lack of symmetry, we have followed the abstract framework of two-level Schwarz methods for non-symmetric problems given in [13] (see also [40] Chap. 5.4). The key idea is to verify that the skew-symmetric part of the operator is “small” with respect to the symmetric part, typically a low order relatively compact perturbation. In our case, due to the lack of adjoint consistency, for the bound of the skew symmetric part of the operator one can not apply the classical framework and have to proceed in a different way (see (33) below). The results presented in this section are indeed not satisfactory, since we do not succeed in providing a complete convergence theory. The main reason is related to the convergence theory of GMRES generally advocated by the analysis of Schwarz methods; namely, the Eisenstat *et al.* theory given in [22] which provides bounds that are far from being sharp and give a condition for the GMRES convergence which, in our case, might not be satisfied (see Sect. 6.3 for further discussions). However, we will show in some numerical results (see Sect. 7.2) that the GMRES applied to the preconditioned system may still converge in a finite number of steps and the proposed preconditioners seem to be scalable.

6.1. Preliminaries

We recall that, for any $u, v \in V_h$, the primal bilinear form of the non-symmetric DG methods is given by

$$\mathcal{A}_h^{DG}(u, v) = \int_{\Omega} \nabla_h u \cdot \nabla_h v \, dx - [1 - \gamma_{[\hat{u}]}] \int_{\mathcal{E}} \llbracket u \rrbracket \cdot \{\!\{ \nabla_h v \}\!\} \, ds - \int_{\mathcal{E}} \{\!\{ \nabla_h u \}\!\} \cdot \llbracket v \rrbracket \, ds + \mathcal{S}^h(u, v), \quad (32)$$

where $\mathcal{S}^h(u, v)$ is defined according to (10) and where $\gamma_{[\hat{u}]} = 2$ for the NIPG method and $\gamma_{[\hat{u}]} = 1$ for the IIPG method. We consider the symmetric and skew-symmetric part of \mathcal{A}_h^{DG} , denoted by a_h and s_h , respectively, and given by

$$\begin{aligned} a_h(u, v) &= \int_{\Omega} \nabla_h u \cdot \nabla_h v \, dx + \frac{[\gamma_{[\hat{u}]} - 2]}{2} \left\{ \int_{\mathcal{E}} \llbracket u \rrbracket \cdot \{\!\{ \nabla_h v \}\!\} \, ds + \int_{\mathcal{E}} \{\!\{ \nabla_h u \}\!\} \cdot \llbracket v \rrbracket \, ds \right\} + \mathcal{S}^h(u, v), \quad u, v \in V_h, \\ s_h(u, v) &= \frac{\gamma_{[\hat{u}]}}{2} \int_{\mathcal{E}} \llbracket u \rrbracket \cdot \{\!\{ \nabla_h v \}\!\} \, ds - \frac{\gamma_{[\hat{u}]}}{2} \int_{\mathcal{E}} \{\!\{ \nabla_h u \}\!\} \cdot \llbracket v \rrbracket \, ds, \quad u, v \in V_h, \end{aligned}$$

and observe that $\mathcal{A}_h^{DG}(u, v) = a_h(u, v) + s_h(u, v)$ for all $u, v \in V_h$. Our analysis of the additive Schwarz method for non-symmetric DG approximations will be carried out by using the inner product defined by $a_h(\cdot, \cdot)$, and its induced norm $\|\cdot\|_a^2 = a_h(\cdot, \cdot)$. Notice that, for the NIPG method, $\|\cdot\|_a$ is indeed a norm for any $\alpha_e > 0$, while, for the IIPG method, this is true provided α_e is taken large enough (so as to guarantee the coercivity of $a_h(\cdot, \cdot)$). Moreover, if $\alpha^* = \max_{e \in \mathcal{E}} \alpha_e \geq \alpha_* = \min_{e \in \mathcal{E}} \alpha_e \geq 1$, with $\frac{\alpha^*}{\alpha_*} \approx C$, $C > 0$ a positive constant then $\alpha_* |u|_*^2 \leq \mathcal{S}^h(u, u) \leq \alpha^* |u|_*^2$. By using the last estimate, the trace (13) and inverse (11) inequalities it is straightforward to see that, for all $u, v \in V_h$,

$$\left| \int_{\mathcal{E}} \{\!\{ \nabla_h u \}\!\} \cdot \llbracket v \rrbracket \, ds \right| \leq C_t \sqrt{(1 + C_{inv}^2)} |u|_{1,h} |v|_* \leq \frac{C_{ss}}{\sqrt{\alpha_*}} \left[|u|_{1,h} \mathcal{S}^h(v, v)^{1/2} \right] \leq \frac{C_{ss}}{\sqrt{\alpha_*}} \|u\|_a \|v\|_a,$$

where C_{ss} only depends on the shape regularity of \mathcal{T}_h and the polynomial degree ℓ_h .

From the last inequality and by taking into account the expression of s_h given above, we get

$$|s_h(u, v)| \leq \frac{\gamma_{[\hat{u}]}}{2} \left| \int_{\mathcal{E}} \llbracket u \rrbracket \cdot \{\!\{ \nabla_h v \}\!\} \, ds \right| + \frac{\gamma_{[\hat{u}]}}{2} \left| \int_{\mathcal{E}} \{\!\{ \nabla_h u \}\!\} \cdot \llbracket v \rrbracket \, ds \right| \leq C_{ss} \frac{\gamma_{[\hat{u}]}}{\sqrt{\alpha_*}} \|u\|_a \|v\|_a. \quad (33)$$

Then, for all $u, v \in V_h$, the following properties are clearly satisfied:

$$\text{Continuity} \quad |\mathcal{A}_h^{DG}(u, v)| \leq |a_h(u, v)| + |s_h(u, v)| \leq (1 + C_{ss} \gamma_{[\hat{u}]} \alpha_*^{-1/2}) \|u\|_a \|v\|_a = C_M \|u\|_a \|v\|_a, \quad (34)$$

$$\text{Coercivity} \quad \mathcal{A}_h^{DG}(u, u) = a_h(u, u) = \|u\|_a^2. \quad (35)$$

6.2. Projection-like operators

The first non-overlapping Schwarz method we consider is defined according to Section 3. In particular, following (20), for any $u_i, v_i \in V_h^i$, the bilinear form associated to the local solvers, denoted by \mathcal{A}_i^{DG} , reads

$$\begin{aligned} \mathcal{A}_i^{DG}(u_i, v_i) &= \int_{\Omega_i} \nabla_h u_i \cdot \nabla_h v_i \, dx - [1 - \gamma_{[\hat{u}]}] \int_{\mathcal{E}} \llbracket u_i \rrbracket \cdot \{\!\{ \nabla_h v_i \}\!\} \, ds \\ &\quad - \int_{\mathcal{E}} \{\!\{ \nabla_h u_i \}\!\} \cdot \llbracket v_i \rrbracket \, ds + \sum_{e \in \mathcal{E}_i} \alpha_e h_e^{-1} \int_e \llbracket u_i \rrbracket \cdot \llbracket v_i \rrbracket \, ds. \end{aligned}$$

The corresponding symmetric and skew-symmetric parts, will be designated by a_i and s_i , respectively. We remark that all the tools and results given in Sections 3 and 4, remain valid for the non-symmetric DG methods.

In Section 3.1 we have introduced the general form of a first additive Schwarz operator P_{ad} by means of the projection-like operators \tilde{P}_i (see (25)), which map the finite element space V_h into the local spaces V_h^i . In the same spirit we shall define next a second additive Schwarz method for the non-symmetric DG approximations, which we shall designate by T_{ad} and involves the corresponding projection-like operators associated with the symmetric part of \mathcal{A}_h^{DG} . For $i = 1, \dots, N_s$, we define

$$\tilde{Q}_i : V_h \longrightarrow V_h^i, \quad a_i(\tilde{Q}_i u, v_i) = \mathcal{A}_h^{DG}(u, R_i^T v_i) \quad \forall v_i \in V_h^i, \quad (36)$$

and set

$$Q_i = R_i^T \tilde{Q}_i : V_h \longrightarrow R_i^T V_h^i \subset V_h. \quad (37)$$

Clearly, \tilde{Q}_i and therefore Q_i are well defined, since the local bilinear forms are positive definite. Our second additive Schwarz method is given by $T_{ad} = P_0 + \sum_{i=1}^{N_s} Q_i$ and can be written in matrix notation as

$$\mathbb{T}_{ad} = R_0^T \mathbb{A}_0^{-1} R_0 \mathbb{A} + \sum_{i=1}^{N_s} [R_i^T \mathbb{A}_i^{-1} R_i] \mathbb{A} = \mathbb{B}_t \mathbb{A}, \quad (38)$$

where we have denoted by \mathbb{A}_i the stiffness matrices associated with the local bilinear forms a_i and, as usual, by \mathbb{A}_i and \mathbb{A} those associated with \mathcal{A}_i^{DG} and \mathcal{A}_h^{DG} , respectively. Notice that, if the method defined by \mathcal{A}_h^{DG} were symmetric the above additive Schwarz method would coincide with the additive Schwarz method P_{ad} already defined (see (26)). To solve the preconditioned systems, either $\mathbb{P}_{ad} \mathbf{u} = \mathbb{B} \mathbf{f}$ or $\mathbb{T}_{ad} \mathbf{u} = \mathbb{B}_t \mathbf{f}$, we consider the GMRES method.

6.3. Preconditioned GMRES method: the issue of convergence

In this section we deal with the convergence analysis for the Schwarz preconditioners accelerated with the GMRES method. Following [22], the convergence of the GMRES method can be characterized in terms of the minimum eigenvalue of the symmetric part of P_{ad} and the norm of the operator. Hence, by setting

$$c_p(P_{ad}) = \inf_{\substack{u \in V_h \\ u \neq 0}} \frac{a_h(u, P_{ad} u)}{a_h(u, u)}, \quad C_p(P_{ad}) = \sup_{\substack{u \in V_h \\ u \neq 0}} \frac{\|P_{ad} u\|_a}{\|u\|_a}, \quad (39)$$

one can conclude that provided that $c_p(P_{ad}) > 0$, the GMRES method applied to $\mathbb{P}_{ad} \mathbf{u} = \mathbb{B} \mathbf{f}$ converges in a finite number of iterations and the norm of the residual of the preconditioned system after m iterations, $\mathbf{r}_m = \mathbb{B} \mathbf{f} - \mathbb{P}_{ad} \mathbf{u}_m$, can be bounded by

$$\|\mathbf{r}_m\|_a \leq \left(1 - \frac{c_p^2}{C_p^2}\right)^{m/2} \|\mathbf{r}_0\|_a.$$

We also point out that the above estimate provides the worst case scenario. The same happens when considering the preconditioned system $\mathbb{T}_{ad} \mathbf{u} = \mathbb{B}_t \mathbf{f}$.

Proposition 6.1 below provides an upper bound for the quantity C_p defined in (39). As regards the lower bound for c_p , we shall show by means of numerical computations that, in general, a strictly positive lower bound for c_p cannot be guaranteed. More precisely, we demonstrate that, the symmetric parts of P_{ad} and T_{ad} might have at least a negative eigenvalue. Therefore, the convergence theory of Eisentat *et al.* [22] cannot be applied for proving the convergence of our Schwarz methods accelerated with the GMRES linear solver. Nevertheless, the numerical tests in Section 7.2 show that the GMRES applied to the preconditioned systems converges in a finite number of steps and that the proposed preconditioners seem to be scalable.

Proposition 6.1. *Let P_{ad} and T_{ad} be the additive Schwarz methods defined in (26) and (38), respectively, for the non-symmetric DG methods given in (32). Then,*

$$a_h(P_{ad}u, P_{ad}u) \leq 2C_M^2(1 + \omega^2(N_c + 1)^2)a_h(u, u), \quad a_h(T_{ad}u, T_{ad}u) \leq 2C_M^2(1 + \omega^2(N_c + 1)^2)a_h(u, u),$$

where N_c is the maximum number of adjacent subdomains that a given subdomain Ω_i can have, ω is the constant given in Lemma 4.4 and C_M is the continuity constant (34).

To prove Proposition 6.1, we need a few lemmas.

Lemma 6.2. *Let $v = \sum_{i=1}^{N_s} R_i^T v_i$, with $v_i \in V_h^i$. Then,*

$$\|v\|_a^2 = \left\| \sum_{i=1}^{N_s} R_i^T v_i \right\|_a^2 \leq (N_c + 1) \sum_{i=1}^{N_s} \|R_i^T v_i\|_a^2,$$

where N_c denotes the maximum number of adjacent subdomains that a given subdomain Ω_i can have.

The proof proceeds along the lines of (31), but replacing \mathcal{A}_h^{DG} by its symmetric part a_h . We omit the details for the sake of brevity.

Lemma 6.3. *Let N_c be the maximum number of adjacent subdomains that a given subdomain Ω_i can have and let ω be the local stability constant given in Lemma 4.4 and C_M the continuity constant (34). Then, for all $u \in V_h$, it holds true*

$$\sum_{i=1}^{N_s} a_h(P_i u, P_i u) \leq C_M^2 \omega^2 (N_c + 1) a_h(u, u), \quad (40)$$

$$\sum_{i=1}^{N_s} a_h(Q_i u, Q_i u) \leq C_M^2 \omega^2 (N_c + 1) a_h(u, u). \quad (41)$$

Proof. To obtain the upper bound (40), by means of the definition (25) of \tilde{P}_i , Lemma 4.4, the continuity property (34) and Lemma 6.2, we find

$$\begin{aligned} \sum_{i=1}^{N_s} a_h(P_i u, P_i u) &= \sum_{i=1}^{N_s} \mathcal{A}_h^{DG}(P_i u, P_i u) = \sum_{i=1}^{N_s} \mathcal{A}_h^{DG}(R_i^T \tilde{P}_i u, R_i^T \tilde{P}_i u) \leq \omega \sum_{i=1}^{N_s} \mathcal{A}_i^{DG}(\tilde{P}_i u, \tilde{P}_i u) \\ &= \omega \sum_{i=1}^{N_s} \mathcal{A}_h^{DG}(u, P_i u) = \omega \mathcal{A}_h^{DG}(u, \sum_{i=1}^{N_s} P_i u) \leq C_M \omega \sqrt{N_c + 1} \|u\|_a \left(\sum_{i=1}^{N_s} \|P_i u\|_a^2 \right)^{1/2}. \end{aligned}$$

By cancelling the common factor and squaring we get (40). The bound (41) can be easily obtained by arguing similarly but taking into account the definition (36) of \tilde{Q}_i . \square

Now, we can prove Proposition 6.1.

Proof of Proposition 6.1. We start by showing the upper bound $\|P_{ad}u\|_a^2 \leq 2C_M^2(1 + \omega^2(N_c + 1)^2)\|u\|_a^2$. We first notice that, from the definitions (25) and (23) and the continuity property (34) we obtain

$$\|P_0 u\|_a^2 = a_h(P_0 u, P_0 u) = \mathcal{A}_h^{DG}(R_0^T \tilde{P}_0 u, R_0^T \tilde{P}_0 u) = \mathcal{A}_0^{DG}(\tilde{P}_0 u, \tilde{P}_0 u) = \mathcal{A}_h^{DG}(u, P_0 u) \leq C_M \|u\|_a \|P_0 u\|_a.$$

TABLE 2. NIPG method: $\ell_h = \ell_H = 1$, $\alpha = 10$.(a) ESTIMATE $c_p(P_{ad})$: unstructured triangular grids.

$1/H \backslash 1/h$	8	16	32	64
4	0.2103	0.0484	-0.0104	-0.0360
8	0.9636	0.2175	0.0662	0.0060
16	-	0.9590	0.2119	0.0625
32	-	-	0.9557	0.1880

(b) ESTIMATE $c_p(T_{ad})$: Cartesian grids.

$1/H \backslash 1/h$	8	16	32	64
4	0.1749	0.0639	0.0188	-0.0011
8	0.9839	0.1931	0.0776	0.0265
16	-	0.9794	0.1930	0.0778
32	-	-	0.9774	0.1919

TABLE 3. NIPG method: $\ell_h = \ell_H = 1$, unstructured triangular grids.(a) ESTIMATE $c_p(P_{ad})$: $\alpha = 1$.

$1/H \backslash 1/h$	8	16	32	64
4	0.0634	-0.1588	-0.3130	-0.4017
8	0.6374	0.0115	-0.2603	-0.4012
16	-	0.6344	-0.0198	-0.2700
32	-	-	0.6227	-0.0498

(b) ESTIMATE $c_p(P_{ad})$: $\alpha = 100$.

$1/H \backslash 1/h$	8	16	32	64
4	0.0309	0.0078	0.0001	-0.0032
8	0.9965	0.0301	0.0089	0.0014
16	-	0.9959	0.0311	0.0089
32	-	-	0.9955	0.0301

Hence, we have $a_h(P_0u, P_0u) \leq C_M^2 a_h(u, u)$. Then, the upper bound for $\|P_{ad}u\|_a^2$ follows by using the previous estimate, Lemmas 6.2 and 6.3

$$\begin{aligned} \|P_{ad}u\|_a^2 &= \left\| \sum_{i=0}^{N_s} P_i u \right\|_a^2 \leq 2 \|P_0u\|_a^2 + 2 \left\| \sum_{i=1}^{N_s} P_i u \right\|_a^2 \\ &\leq 2C_M^2 \|u\|_a^2 + 2(N_c + 1) \sum_{i=1}^{N_s} \|P_i u\|_a^2 \leq 2C_M^2 (1 + \omega^2 (N_c + 1)^2) \|u\|_a^2. \end{aligned}$$

The upper bound $\|T_{ad}u\|_a^2 \leq 2C_M^2 (1 + \omega^2 (N_c + 1)^2) \|u\|_a^2$ follows analogously. \square

As regards the lower bound for $c_p(P_{ad})$ (and $c_p(T_{ad})$), notice that $c_p(P_{ad})$ corresponds indeed to the minimum eigenvalue of the following generalised eigenvalue problem:

$$\text{Find } (\lambda, v \neq 0) \in \mathbb{C} \times V_n \text{ such that } a_h(P_{ad}v, v) = \lambda a_h(v, v).$$

We provide a numerical evaluation of $c_p(P_{ad})$ (and $c_p(T_{ad})$) in order to demonstrate that, if the mesh size h is small enough, the symmetric part of P_{ad} (and T_{ad}) might have at least a negative eigenvalue.

In Table 2 we have reported the estimate of $c_p(P_{ad})$ (right) and $c_p(T_{ad})$ (left) obtained with the NIPG method on unstructured triangular grids and on Cartesian grids, respectively. More precisely, the results shown have been obtained with $\ell_h = \ell_H = 1$ and $\alpha_e = \alpha = 10$ for any $e \in \mathcal{E}$. We notice that in both cases, and regardless the structure of the mesh, $c_p < 0$ for h relatively small compared to H . We have also addressed the dependence of c_p on the choice of the penalty parameter. In Table 3 we have reported the estimate of $c_p(P_{ad})$ for $\alpha = 1, 100$. More precisely, the results shown in Table 3 has been carried out with NIPG method, on unstructured triangular grids and with $\ell_h = \ell_H = 1$. From the results reported in Table 3 (see also Tab. 2a) it seems that, even by increasing the value of the penalty parameter, c_p is still negative for h small enough but the first negative eigenvalue appears for a smaller value of h . Therefore, the symmetric part of the proposed Schwarz operators is not in general positive definite, and the Eisenstat *et al.* theory [22] cannot be applied. Nevertheless, the numerical tests in Section 7.2 show that the GMRES applied to the preconditioned systems

converges in a finite number of steps and that the proposed preconditioners seem to be scalable. Finally, we wish to note that it might be worthy to look to the other existing theories for the GMRES convergence (see, *i.e.*, [39,41]) to analyse the Schwarz methods introduced here. This will be done in a future work.

7. NUMERICAL RESULTS

In this section we present some numerical experiments to illustrate the performance of our non-overlapping Schwarz methods for different DG schemes (see Tab. 1). All the experiments have been carried out on the two-dimensional domain $\Omega = (0,1)^2$. The right-hand side f and the non-homogeneous boundary conditions are chosen such that the exact solution of (1) is given by $u(x,y) = \exp(xy)$.

We have considered three levels of nested refinements: a subdomain partition \mathcal{T}_S , a coarse triangulation \mathcal{T}_H (with mesh size H) and a fine triangulation \mathcal{T}_h (with mesh size h). The subdomain partitions consist of squares (recalling that N_s denotes the number of subdomains, we will consider the cases $N_s = 4$ and $N_s = 16$ squares). For the coarse and the fine grids, we consider both Cartesian grids consisting of N^2 squares, and unstructured triangular meshes made of $2N^2$ triangles with $N = 4, 8, 16, 32$ and $N = 8, 16, 32, 64$ for the coarse and the fine meshes, respectively. We observe that, for all the types of triangulation considered, we have $1/H \sim N$ and $1/h \sim N$, with N chosen as before. Through the whole section, we use the notation $\{(\mathcal{M}^{\ell_h}, \mathcal{T}_h), (\mathcal{M}^{\ell_H}, \mathcal{T}_H)\}$ to indicate that we use polynomials of order ℓ_h for the fine mesh space V_h and of order ℓ_H for the coarse space V_H . For the sake of simplicity we have chosen $\alpha_e = \alpha$ for all $e \in \mathcal{E}$ (see (10)). All the linear systems have been solved with the Preconditioned Conjugate Gradient (PCG) method for the symmetric DG schemes, and the Generalized Minimal Residual (GMRES) method for the non-symmetric schemes. Taking into account definition of the preconditioner given in (27), solving these systems requires $N_s + 1$ subtasks of the form $\mathbf{x} \rightarrow \mathbb{R}_i \mathbf{x} \rightarrow \mathbb{A}_i^{-1} \mathbb{R}_i \mathbf{x} \rightarrow \mathbb{R}_i^T \mathbb{A}_i^{-1} \mathbb{R}_i \mathbf{x}$ that correspond to the resolution of the same subproblem on the small domain Ω_i , $i = 0, \dots, N_s$. We have solved each of these subtasks using a direct solver but it should be pointed out that, when H and therefore h are small the direct resolution of the coarse system turns out to be a big task, the same happens for the local systems, *i.e.*, $i = 1, \dots, N_s$, if the size of the subdomains does not follow h . In those cases, multigrid solvers as those considered in [26,27] will be more appropriate. By recalling that \mathbf{r}_k is the residual at k -th iteration (*i.e.*, $\mathbf{r}_k = \mathbb{B}\mathbf{f} - \mathbb{B}\mathbb{A}\mathbf{u}_k$) and denoting with $\|\cdot\|_2$ the standard Euclidean norm, we use as stopping criteria $\|\mathbf{r}_k\|_2 \leq 10^{-9} \|\mathbf{r}_0\|_2$, allowing at most for 300 iterations. In the solution of the non-preconditioned system we admit at most 2000 iterations.

The rest of this section is organized as follows: in Section 7.1 we discuss the performance of the two-level method for some of the symmetric schemes summarized in Table 1, while Section 7.2 is devoted to show the performance of the proposed preconditioners for the non-symmetric schemes NIPG [36] and IIPG [20]. All the presented computations have been performed in MATLAB.

7.1. Symmetric methods

In this section we present the results of some numerical experiments designed to test the performance of our non-overlapping Schwarz method (27), for some of the symmetric DG schemes summarized in Table 1, namely, the SIPG method and the LDG method (see [3], and [18], respectively, for details). At the end of this section, we present some experiments carried out with the super penalty method proposed by Babuška-Zlámal in [5].

A major point of interest is the experimental estimation of the condition number $\kappa(\mathbb{B}\mathbb{A})$ that has been obtained by exploiting the analogies between the Lanczos technique and the PCG method. In fact, a tridiagonal matrix can be built in the PCG code with the property that, during the iterative procedure, the approximation of extremal eigenvalues of $\mathbb{B}\mathbb{A}$ becomes better and better (see [25], Sects. 9.3, 10.2. for more details). We stress that no additional work is involved, since all the entries of the tridiagonal matrix are readily available during the PCG iteration. The experimental estimation of $\kappa(\mathbb{A})$ has been carried out in the same way.

We first address the scalability of our Schwarz method (27), that is the independence of the performance of the preconditioner of the number of subdomains. In Tables 4a and 4c, the condition number estimates for

TABLE 4. LDG method ($\alpha = 10$, $\beta = [0.5, 0.5]'$): $\{(\mathcal{Q}^1, \mathcal{T}_h), (\mathcal{Q}^1, \mathcal{T}_H)\}$, $N_s = 4, 16$, Cartesian grids.

(a) CONDITION NUMBER: $N_s = 4$.					(b) ITERATION COUNTS: $N_s = 4$.				
$1/H \backslash 1/h$	8	16	32	64	$1/H \backslash 1/h$	8	16	32	64
4	18.7	40.9	85.4	174.2	4	28	44	65	85
8	3.2	19.1	42.1	88.0	8	15	33	51	70
16	-	3.2	19.5	42.5	16	-	15	35	53
32	-	-	3.2	19.5	32	-	-	15	35
$\kappa(\mathbb{A})$	376.5	1468.3	5838.9	23324.3	#iter(\mathbb{A})	85	161	342	674

(c) CONDITION NUMBER: $N_s = 16$.					(d) ITERATION COUNTS: $N_s = 16$.				
$1/H \backslash 1/h$	8	16	32	64	$1/H \backslash 1/h$	8	16	32	64
4	19.8	44.9	95.1	189.8	4	33	52	74	106
8	3.2	18.7	40.7	84.8	8	15	33	51	72
16	-	3.2	19.3	42.3	16	-	15	35	52
32	-	-	3.2	19.5	32	-	-	15	36
$\kappa(\mathbb{A})$	376.5	1468.3	5838.9	23324.3	#iter(\mathbb{A})	85	161	342	674

the LDG method ($\alpha = 10$, $\beta = [0.5, 0.5]'$) on Cartesian grids with $\{(\mathcal{Q}^1, \mathcal{T}_h), (\mathcal{Q}^1, \mathcal{T}_H)\}$ on the two different subdomain partitions ($N_s = 4, 16$) are reported. The corresponding iteration counts are reported in Tables 4b and 4d, respectively. The dashes in the tables indicate that we are not in the case $\mathcal{T}_H \subseteq \mathcal{T}_h$ and therefore it does not make any sense to build the preconditioner. From the results reported in Table 4, it seems that our preconditioner is substantially insensitive to the number of the subdomains, and the convergence rates predicted by Theorem 5.1 are actually achieved. Notice that, if we refine both the fine and the coarse mesh keeping their ratio constant, we observe, as predicted, that both the condition number and the iteration counts remain, substantially, unchanged. We point out, that the lower diagonals reported in Table 4 (and also in all the tables below) correspond to the limit case $h = H$. In this limit case the coarse level is a solver of the problem and the subgrid solutions spoil the result. We have addressed also this “purely academic” case $h = H$ to show that the preconditioner works also in the limit situation and to give an idea of the spoiling effects of the subgrid solutions.

Now we consider the Schwarz method (27) for higher order approximation degrees. For the sake of brevity, we focus our attention on a 16 subdomain partition. In Table 5a we report the condition number estimates for the SIPG method ($\alpha = 10$) with $\{(\mathcal{Q}^2, \mathcal{T}_h), (\mathcal{Q}^2, \mathcal{T}_H)\}$ on Cartesian grids. The corresponding iteration counts are given in Table 5b. In Table 6 we have reported the results obtained for the SIPG method ($\alpha = 10$) with $\{(\mathcal{P}^2, \mathcal{T}_h), (\mathcal{P}^1, \mathcal{T}_H)\}$ on unstructured triangular grids. It can be observed that, even in the case of higher order polynomial approximations, the convergence rates predicted by Theorem 5.1 are achieved. This is attested by both the condition number estimates and the iteration counts.

Next, we investigate the effect of a coarse mesh space V_H made of piecewise constants on the performances of our preconditioner. The first set of experiments has been carried out on Cartesian grids with the LDG method ($\alpha = 10$, $\beta = [0.5, 0.5]'$). In Table 7a the condition number estimates with $\{(\mathcal{Q}^1, \mathcal{T}_h), (\mathcal{Q}^0, \mathcal{T}_H)\}$ are shown; the corresponding iteration counts are reported in Table 7b. In Table 8 we have reported the results obtained for the SIPG method ($\alpha = 10$) with $\{(\mathcal{P}^2, \mathcal{T}_h), (\mathcal{P}^0, \mathcal{T}_H)\}$ on unstructured triangular grids. If we compare Table 6 with Table 8, it is clear that, by choosing \mathcal{P}^2 approximation for the fine mesh spaces, our preconditioner performs better with a piecewise linear coarse space than with a piecewise constant coarse space. The same happens if we compare Table 4 with Table 7 (where we have used bilinear polynomials for the approximation

TABLE 5. SIPG method ($\alpha = 10$): $\{(\mathcal{Q}^2, \mathcal{T}_h), (\mathcal{Q}^2, \mathcal{T}_H)\}$, $N_s = 16$, Cartesian grids.

(a) CONDITION NUMBER.					(b) ITERATION COUNTS.				
$1/H \backslash 1/h$	8	16	32	64	$1/H \backslash 1/h$	8	16	32	64
4	29.4	61.9	126.2	254.0	4	38	57	82	112
8	3.1	29.8	62.2	127.0	8	13	38	59	83
16	-	3.1	30.5	63.2	16	-	13	39	60
32	-	-	3.1	30.6	32	-	-	12	38
#iter(\mathbb{A})	1330.2	5258.2	20972.2	83828.5	#iter(\mathbb{A})	167	330	647	1266

 TABLE 6. SIPG method ($\alpha = 10$): $\{(\mathcal{P}^2, \mathcal{T}_h), (\mathcal{P}^1, \mathcal{T}_H)\}$, $N_s = 16$, unstructured triangular grids.

(a) CONDITION NUMBER.					(b) ITERATION COUNTS.				
$1/H \backslash 1/h$	8	16	32	64	$1/H \backslash 1/h$	8	16	32	64
4	50.1	106.6	219.1	459.8	4	59	84	114	165
8	18.5	53.3	116.2	251.1	8	35	58	87	124
16	-	20.6	54.3	123.9	16	-	35	58	85
32	-	-	23.1	60.3	32	-	-	36	58
$\kappa(\mathbb{A})$	1963.2	7750.5	31436.4	130745.3	#iter(\mathbb{A})	193	360	707	1423

 TABLE 7. LDG method ($\alpha = 10$, $\beta = [0.5, 0.5]'$): $\{(\mathcal{Q}^1, \mathcal{T}_h), (\mathcal{Q}^0, \mathcal{T}_H)\}$, $N_s = 16$, Cartesian grids.

(a) CONDITION NUMBER.					(b) ITERATION COUNTS.				
$1/H \backslash 1/h$	8	16	32	64	$1/H \backslash 1/h$	8	16	32	64
4	99.3	203.5	412.6	831.5	4	62	92	125	176
8	57.5	124.0	251.7	507.7	8	53	81	109	155
16	-	68.7	143.3	289.4	16	-	64	92	128
32	-	-	76.1	156.0	32	-	-	71	100
$\kappa(\mathbb{A})$	376.5	1468.3	5838.9	23324.3	#iter(\mathbb{A})	85	161	342	674

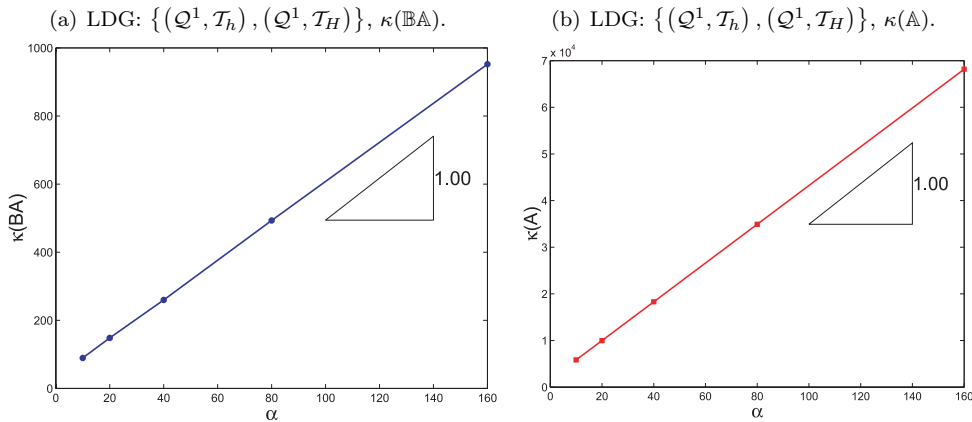
of the fine mesh space). From the results in Table 7 and Table 8 we note that by choosing $(\mathcal{M}^0, \mathcal{T}_H)$ as³ a coarse solver, for fixed H the convergence rates linearly depend on h , while for fixed h the rate $O(H)$ seems to be achieved asymptotically. A possible reason for this behaviour, already observed in [23], could be that the restriction of the global bilinear form to the poor piecewise constant coarse space does not provide a sufficiently accurate transportation of the information.

Now we investigate numerically the dependence of the condition number on the value of the penalty parameter. Results reported in Figures 1a, 1b have been carried out with the LDG method on Cartesian grids by choosing a $\{(\mathcal{Q}^1, \mathcal{T}_h), (\mathcal{Q}^1, \mathcal{T}_H)\}$ approximation and with different choices of the penalty parameter. More precisely, in Figure 1 we have reported the condition number estimates with $1/h = 32$, $1/H = 4$ for $\alpha = 10, 20, 40, 80, 160$. We have repeated the same set of experiments by using piecewise biquadratic polynomials both for the fine and the coarse mesh spaces and we have observed the same behaviour; we omit the results

³Where \mathcal{M} is either \mathcal{P} or \mathcal{Q} .

TABLE 8. SIPG method ($\alpha = 10$): $\{(\mathcal{P}^2, \mathcal{T}_h), (\mathcal{P}^0, \mathcal{T}_H)\}$, $N_s = 16$, unstructured triangular grids.

(a) CONDITION NUMBER.					(b) ITERATION COUNTS.				
$1/H \backslash 1/h$	8	16	32	64	$1/H \backslash 1/h$	8	16	32	64
4	235.3	478.0	972.5	2033.0	4	101	146	204	297
8	121.7	248.4	506.6	1060.4	8	78	108	155	214
16	-	141.0	291.4	616.9	16	-	84	119	170
32	-	-	170.8	368.7	32	-	-	96	139
$\kappa(\mathbb{A})$	1963.2	7750.5	31 436.4	130 745.3	#iter(\mathbb{A})	193	360	707	1423

FIGURE 1. LDG method: $N_s = 16$, $1/h = 32$, $1/H = 4$, $\alpha = 10, 20, 40, 80, 160$.

for the sake of brevity. From the results shown in Figure 1 it is clear that, the condition number of the preconditioned matrix (and of the non preconditioned one) depends linearly on the value of the penalty parameter.

Now, we present a numerical comparison between our Schwarz preconditioner and the one proposed in [23] in which the local solvers are defined as the restriction of the global bilinear form to each subdomain. In Table 9a we have reported the condition number estimate carried out with the SIPG method ($\alpha = 10$) on Cartesian grids by choosing $\{(\mathcal{Q}^1, \mathcal{T}_h), (\mathcal{Q}^1, \mathcal{T}_H)\}$ approximation. The analogous results obtained by preconditioning with the method proposed in [23] are reported in Table 9b. It is clear that the performance of our Schwarz method is comparable with the one of the preconditioner proposed in [23].

To conclude this section, we present a numerical experiment carried out with the super penalty method proposed by Babuška and Zlámal in [5]. This method is not consistent and to ensure stability (in some appropriate norm, see [4] for details) the jumps must be penalised with a power of h depending on the polynomial approximation degree ℓ_h . As a result, the condition number of the resulting matrices behaves like $O(h^{-2(\ell_h+1)})$ (see [14] for details), whereas for the other DG methods the condition number is $O(h^{-2})$. Although our analysis does not apply to this kind of methods, we have tested our preconditioner in a set of experiments carried out on Cartesian grids with low order approximations: $\{(\mathcal{Q}^1, \mathcal{T}_h), (\mathcal{Q}^1, \mathcal{T}_H)\}$ (see Tab. 10). Notice that, in this case, the condition number of the non preconditioned system behaves like $O(h^{-4})$. The results in Table 10 show that, although the convergence rate (30) is not achieved, our preconditioner seems to improve the condition number of the preconditioned matrix and to significantly reduce the iteration counts needed for solving the linear system.

TABLE 9. SIPG method ($\alpha = 10$): condition number, $\{(\mathcal{Q}^1, \mathcal{T}_h), (\mathcal{Q}^1, \mathcal{T}_H)\}$, $N_s = 16$, Cartesian grids.

(a) $\mathbb{B}\mathbf{A}\mathbf{u} = \mathbb{B}\mathbf{f}$.					(b) PRECONDITIONER PROPOSED IN [23].				
$1/H \backslash 1/h$	8	16	32	64	$1/H \backslash 1/h$	8	16	32	64
4	13.1	28.9	60.7	124.1	4	13.7	28.9	58.6	117.7
8	3.0	12.0	25.8	54.1	8	2.9	12.6	26.5	53.7
16	-	3.1	12.5	26.8	16	-	2.9	12.9	27.2
32	-	-	3.1	12.6	32	-	-	2.9	13.0
$\kappa(\mathbb{A})$	265.3	1043.1	4155.5	16 605.7	$\kappa(\mathbb{A})$	265.3	1043.1	4155.5	16 605.7

 TABLE 10. Babuška-Zlámal method ($\alpha = 10$): $\{(\mathcal{Q}^1, \mathcal{T}_h), (\mathcal{Q}^1, \mathcal{T}_H)\}$, $N_s = 16$, Cartesian grids.

(a) CONDITION NUMBER.					(b) ITERATION COUNTS.				
$1/H \backslash 1/h$	8	16	32	64	$1/H \backslash 1/h$	8	16	32	64
4	278.2	2489.0	20 558.2	124 769.5	4	47	96	152	224
8	3.0	1006.4	9045.4	72 004.0	8	12	58	104	157
16	-	3.0	4142.3	35 186.5	16	-	11	65	119
32	-	-	3.0	15 636.6	32	-	-	10	72
$\kappa(\mathbb{A})$	6046.0	94 584.0	1 505 151.5	24 050 499.6	#iter(\mathbb{A})	133	356	816	1753

7.2. Non-symmetric methods

We consider the performance of the two proposed preconditioners for non-symmetric methods

$$\mathbb{B} = \sum_{i=0}^{N_s} R_i^T \mathbb{A}_i^{-1} R_i \quad \text{and} \quad \mathbb{B}_t = R_0^T \mathbb{A}_0^{-1} R_0 + \sum_{i=1}^{N_s} R_i^T \mathbb{A}_i^{-1} R_i,$$

defined in (27) and in (38), respectively. We remark that our implementation of the GMRES solver uses the standard Euclidean norm even if, in general, the convergence bounds for non-symmetric Schwarz preconditioned systems are given in a suitable energy norm. Nevertheless, in view of the results in [39] and taking into account that the relative tolerance is set to 10^{-9} , one might expect that the number of iterates needed for convergence is almost the same. In all our numerical experiments we have used the non-restarted version of GMRES, since we are only interested in a fair comparison of the performance of the proposed preconditioners. In practical applications, in order to avoid excessive storage requirements and computational costs for the orthogonalization, GMRES is usually restarted after a suitable number of iterations.

We first address the scalability of the proposed preconditioners. In Tables 11a and 11b we have reported the iteration counts needed for solving the linear system $\mathbb{B}\mathbf{A}\mathbf{x} = \mathbb{B}\mathbf{f}$ by using the NIPG method ($\alpha = 1$) with 4 and 16 subdomains, respectively. The analogous results computed with the preconditioner \mathbb{B}_t are shown in Tables 11c and 11d, respectively. All the results reported in Table 11 have been carried out with $\{(\mathcal{P}^1, \mathcal{T}_h), (\mathcal{P}^1, \mathcal{T}_H)\}$ on unstructured triangular grids. From these results, it can be inferred that both preconditioners are virtually insensitive to the number of subdomains, and the GMRES applied to the preconditioned systems converges into a finite number of steps and we observe a converge rate of order $\sqrt{H/h}$. In Table 11, the iteration counts for the non-preconditioned system of the finest mesh are not reported because, due to the large memory storage

TABLE 11. NIPG method ($\alpha = 1$): $\{(\mathcal{P}^1, \mathcal{T}_h), (\mathcal{P}^1, \mathcal{T}_H)\}$, $N_s = 4, 16$, unstructured triangular grids.

(a) $\mathbb{B}\mathbf{A}\mathbf{u} = \mathbb{B}\mathbf{f}$: GMRES, $N_s = 4$.	(b) $\mathbb{B}\mathbf{A}\mathbf{u} = \mathbb{B}\mathbf{f}$: GMRES, $N_s = 16$.																																																																								
<table style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="border: none;"></th> <th style="border: none;">$1/h$</th> <th style="border: none;">8</th> <th style="border: none;">16</th> <th style="border: none;">32</th> <th style="border: none;">64</th> </tr> </thead> <tbody> <tr> <td style="border: none;">$1/H$</td> <td style="border: none;">4</td> <td style="border: none;">28</td> <td style="border: none;">37</td> <td style="border: none;">47</td> <td style="border: none;">57</td> </tr> <tr> <td style="border: none;"></td> <td style="border: none;">8</td> <td style="border: none;">14</td> <td style="border: none;">31</td> <td style="border: none;">39</td> <td style="border: none;">47</td> </tr> <tr> <td style="border: none;"></td> <td style="border: none;">16</td> <td style="border: none;">-</td> <td style="border: none;">16</td> <td style="border: none;">31</td> <td style="border: none;">40</td> </tr> <tr> <td style="border: none;"></td> <td style="border: none;">32</td> <td style="border: none;">-</td> <td style="border: none;">-</td> <td style="border: none;">16</td> <td style="border: none;">32</td> </tr> <tr> <td style="border: none;">#iter(\mathbb{A})</td> <td style="border: none;"></td> <td style="border: none;">54</td> <td style="border: none;">94</td> <td style="border: none;">192</td> <td style="border: none;">x</td> </tr> </tbody> </table>		$1/h$	8	16	32	64	$1/H$	4	28	37	47	57		8	14	31	39	47		16	-	16	31	40		32	-	-	16	32	#iter(\mathbb{A})		54	94	192	x	<table style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="border: none;"></th> <th style="border: none;">$1/h$</th> <th style="border: none;">8</th> <th style="border: none;">16</th> <th style="border: none;">32</th> <th style="border: none;">64</th> </tr> </thead> <tbody> <tr> <td style="border: none;">$1/H$</td> <td style="border: none;">4</td> <td style="border: none;">29</td> <td style="border: none;">41</td> <td style="border: none;">50</td> <td style="border: none;">59</td> </tr> <tr> <td style="border: none;"></td> <td style="border: none;">8</td> <td style="border: none;">15</td> <td style="border: none;">31</td> <td style="border: none;">40</td> <td style="border: none;">49</td> </tr> <tr> <td style="border: none;"></td> <td style="border: none;">16</td> <td style="border: none;">-</td> <td style="border: none;">16</td> <td style="border: none;">33</td> <td style="border: none;">41</td> </tr> <tr> <td style="border: none;"></td> <td style="border: none;">32</td> <td style="border: none;">-</td> <td style="border: none;">-</td> <td style="border: none;">17</td> <td style="border: none;">31</td> </tr> <tr> <td style="border: none;">#iter(\mathbb{A})</td> <td style="border: none;"></td> <td style="border: none;">54</td> <td style="border: none;">94</td> <td style="border: none;">192</td> <td style="border: none;">x</td> </tr> </tbody> </table>		$1/h$	8	16	32	64	$1/H$	4	29	41	50	59		8	15	31	40	49		16	-	16	33	41		32	-	-	17	31	#iter(\mathbb{A})		54	94	192	x
	$1/h$	8	16	32	64																																																																				
$1/H$	4	28	37	47	57																																																																				
	8	14	31	39	47																																																																				
	16	-	16	31	40																																																																				
	32	-	-	16	32																																																																				
#iter(\mathbb{A})		54	94	192	x																																																																				
	$1/h$	8	16	32	64																																																																				
$1/H$	4	29	41	50	59																																																																				
	8	15	31	40	49																																																																				
	16	-	16	33	41																																																																				
	32	-	-	17	31																																																																				
#iter(\mathbb{A})		54	94	192	x																																																																				
(c) $\mathbb{B}_t\mathbf{A}\mathbf{u} = \mathbb{B}_t\mathbf{f}$: GMRES, $N_s = 4$.	(d) $\mathbb{B}_t\mathbf{A}\mathbf{u} = \mathbb{B}_t\mathbf{f}$: GMRES, $N_s = 16$.																																																																								
<table style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="border: none;"></th> <th style="border: none;">$1/h$</th> <th style="border: none;">8</th> <th style="border: none;">16</th> <th style="border: none;">32</th> <th style="border: none;">64</th> </tr> </thead> <tbody> <tr> <td style="border: none;">$1/H$</td> <td style="border: none;">4</td> <td style="border: none;">38</td> <td style="border: none;">47</td> <td style="border: none;">57</td> <td style="border: none;">74</td> </tr> <tr> <td style="border: none;"></td> <td style="border: none;">8</td> <td style="border: none;">23</td> <td style="border: none;">41</td> <td style="border: none;">51</td> <td style="border: none;">63</td> </tr> <tr> <td style="border: none;"></td> <td style="border: none;">16</td> <td style="border: none;">-</td> <td style="border: none;">26</td> <td style="border: none;">44</td> <td style="border: none;">56</td> </tr> <tr> <td style="border: none;"></td> <td style="border: none;">32</td> <td style="border: none;">-</td> <td style="border: none;">-</td> <td style="border: none;">30</td> <td style="border: none;">50</td> </tr> <tr> <td style="border: none;">#iter(\mathbb{A})</td> <td style="border: none;"></td> <td style="border: none;">54</td> <td style="border: none;">94</td> <td style="border: none;">192</td> <td style="border: none;">x</td> </tr> </tbody> </table>		$1/h$	8	16	32	64	$1/H$	4	38	47	57	74		8	23	41	51	63		16	-	26	44	56		32	-	-	30	50	#iter(\mathbb{A})		54	94	192	x	<table style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="border: none;"></th> <th style="border: none;">$1/h$</th> <th style="border: none;">8</th> <th style="border: none;">16</th> <th style="border: none;">32</th> <th style="border: none;">64</th> </tr> </thead> <tbody> <tr> <td style="border: none;">$1/H$</td> <td style="border: none;">4</td> <td style="border: none;">38</td> <td style="border: none;">46</td> <td style="border: none;">57</td> <td style="border: none;">74</td> </tr> <tr> <td style="border: none;"></td> <td style="border: none;">8</td> <td style="border: none;">22</td> <td style="border: none;">40</td> <td style="border: none;">49</td> <td style="border: none;">61</td> </tr> <tr> <td style="border: none;"></td> <td style="border: none;">16</td> <td style="border: none;">-</td> <td style="border: none;">25</td> <td style="border: none;">41</td> <td style="border: none;">52</td> </tr> <tr> <td style="border: none;"></td> <td style="border: none;">32</td> <td style="border: none;">-</td> <td style="border: none;">-</td> <td style="border: none;">28</td> <td style="border: none;">45</td> </tr> <tr> <td style="border: none;">#iter(\mathbb{A})</td> <td style="border: none;"></td> <td style="border: none;">54</td> <td style="border: none;">94</td> <td style="border: none;">192</td> <td style="border: none;">x</td> </tr> </tbody> </table>		$1/h$	8	16	32	64	$1/H$	4	38	46	57	74		8	22	40	49	61		16	-	25	41	52		32	-	-	28	45	#iter(\mathbb{A})		54	94	192	x
	$1/h$	8	16	32	64																																																																				
$1/H$	4	38	47	57	74																																																																				
	8	23	41	51	63																																																																				
	16	-	26	44	56																																																																				
	32	-	-	30	50																																																																				
#iter(\mathbb{A})		54	94	192	x																																																																				
	$1/h$	8	16	32	64																																																																				
$1/H$	4	38	46	57	74																																																																				
	8	22	40	49	61																																																																				
	16	-	25	41	52																																																																				
	32	-	-	28	45																																																																				
#iter(\mathbb{A})		54	94	192	x																																																																				

TABLE 12. IIPG method ($\alpha = 10$): $\{(\mathcal{Q}^2, \mathcal{T}_h), (\mathcal{Q}^2, \mathcal{T}_H)\}$, $N_s = 16$, Cartesian grids.

(a) $\mathbb{B}\mathbf{A}\mathbf{u} = \mathbb{B}\mathbf{f}$: GMRES.	(b) $\mathbb{B}_t\mathbf{A}\mathbf{u} = \mathbb{B}_t\mathbf{f}$: GMRES.																																																																								
<table style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="border: none;"></th> <th style="border: none;">$1/h$</th> <th style="border: none;">8</th> <th style="border: none;">16</th> <th style="border: none;">32</th> <th style="border: none;">64</th> </tr> </thead> <tbody> <tr> <td style="border: none;">$1/H$</td> <td style="border: none;">4</td> <td style="border: none;">35</td> <td style="border: none;">54</td> <td style="border: none;">77</td> <td style="border: none;">104</td> </tr> <tr> <td style="border: none;"></td> <td style="border: none;">8</td> <td style="border: none;">13</td> <td style="border: none;">37</td> <td style="border: none;">56</td> <td style="border: none;">79</td> </tr> <tr> <td style="border: none;"></td> <td style="border: none;">16</td> <td style="border: none;">-</td> <td style="border: none;">12</td> <td style="border: none;">36</td> <td style="border: none;">57</td> </tr> <tr> <td style="border: none;"></td> <td style="border: none;">32</td> <td style="border: none;">-</td> <td style="border: none;">-</td> <td style="border: none;">11</td> <td style="border: none;">36</td> </tr> <tr> <td style="border: none;">#iter(\mathbb{A})</td> <td style="border: none;"></td> <td style="border: none;">160</td> <td style="border: none;">311</td> <td style="border: none;">600</td> <td style="border: none;">1153</td> </tr> </tbody> </table>		$1/h$	8	16	32	64	$1/H$	4	35	54	77	104		8	13	37	56	79		16	-	12	36	57		32	-	-	11	36	#iter(\mathbb{A})		160	311	600	1153	<table style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="border: none;"></th> <th style="border: none;">$1/h$</th> <th style="border: none;">8</th> <th style="border: none;">16</th> <th style="border: none;">32</th> <th style="border: none;">64</th> </tr> </thead> <tbody> <tr> <td style="border: none;">$1/H$</td> <td style="border: none;">4</td> <td style="border: none;">36</td> <td style="border: none;">56</td> <td style="border: none;">79</td> <td style="border: none;">107</td> </tr> <tr> <td style="border: none;"></td> <td style="border: none;">8</td> <td style="border: none;">14</td> <td style="border: none;">39</td> <td style="border: none;">58</td> <td style="border: none;">80</td> </tr> <tr> <td style="border: none;"></td> <td style="border: none;">16</td> <td style="border: none;">-</td> <td style="border: none;">14</td> <td style="border: none;">39</td> <td style="border: none;">60</td> </tr> <tr> <td style="border: none;"></td> <td style="border: none;">32</td> <td style="border: none;">-</td> <td style="border: none;">-</td> <td style="border: none;">14</td> <td style="border: none;">39</td> </tr> <tr> <td style="border: none;">#iter(\mathbb{A})</td> <td style="border: none;"></td> <td style="border: none;">160</td> <td style="border: none;">311</td> <td style="border: none;">600</td> <td style="border: none;">1153</td> </tr> </tbody> </table>		$1/h$	8	16	32	64	$1/H$	4	36	56	79	107		8	14	39	58	80		16	-	14	39	60		32	-	-	14	39	#iter(\mathbb{A})		160	311	600	1153
	$1/h$	8	16	32	64																																																																				
$1/H$	4	35	54	77	104																																																																				
	8	13	37	56	79																																																																				
	16	-	12	36	57																																																																				
	32	-	-	11	36																																																																				
#iter(\mathbb{A})		160	311	600	1153																																																																				
	$1/h$	8	16	32	64																																																																				
$1/H$	4	36	56	79	107																																																																				
	8	14	39	58	80																																																																				
	16	-	14	39	60																																																																				
	32	-	-	14	39																																																																				
#iter(\mathbb{A})		160	311	600	1153																																																																				

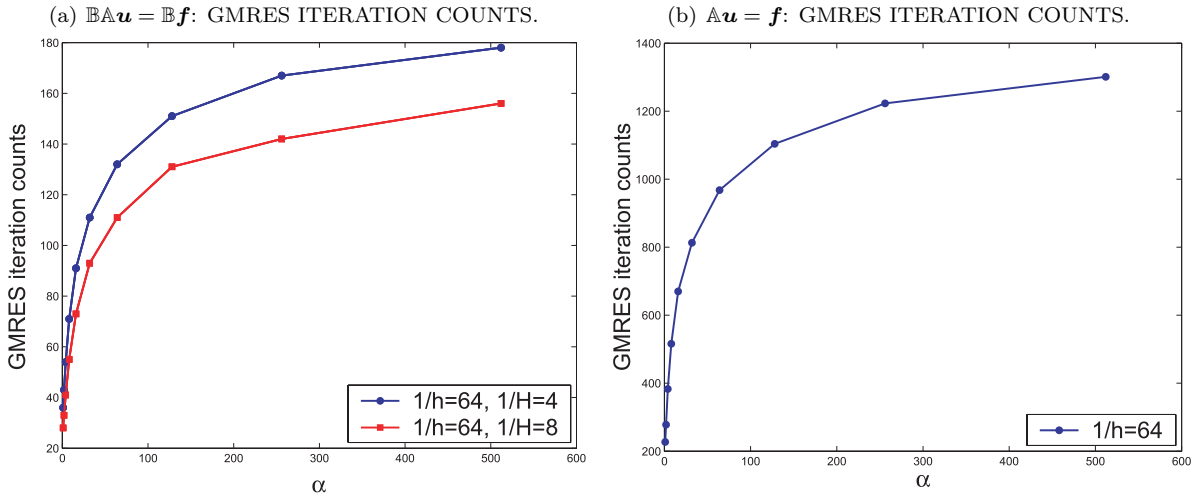
required by GMRES per iteration step, we were not able to solve the non-preconditioned system (the same in Tab. 13).

Now we consider the two Schwarz methods for higher order approximation degrees. As before, we focus our attention on a 16 subdomain partition. In Tables 12a and 12b we have reported the iteration counts required by the two proposed Schwarz methods for the IIPG method ($\alpha = 10$) with $\{(\mathcal{Q}^2, \mathcal{T}_h), (\mathcal{Q}^2, \mathcal{T}_H)\}$ on Cartesian grids. It can be observed that, even in the case of higher order polynomial approximations, the GMRES applied to the preconditioned systems converges.

Next, we investigate the effect of a coarse mesh space V_H made of piecewise constants on the performances of our Schwarz methods. The results reported in Table 13 have been carried out with the NIPG method ($\alpha = 1$). More precisely, in Table 13a the iteration counts needed for solving $\mathbb{B}\mathbf{A}\mathbf{u} = \mathbb{B}\mathbf{f}$ with $\{(\mathcal{P}^1, \mathcal{T}_h), (\mathcal{P}^0, \mathcal{T}_H)\}$ on unstructured triangular grids are shown; the analogous results obtained with the Schwarz preconditioner \mathbb{B}_t are reported in Table 13b. As already observed in the case of symmetric methods, we note that, when using a piecewise constant coarse solver, the performance of our preconditioners slightly deteriorates.

TABLE 13. NIPG method ($\alpha = 1$): $\{(\mathcal{P}^1, \mathcal{T}_h), (\mathcal{P}^0, \mathcal{T}_H)\}$, $N_s = 16$, unstructured triangular grids.

(a) $\mathbb{B}\mathbf{A}\mathbf{u} = \mathbb{B}\mathbf{f}$: GMRES.					(b) $\mathbb{B}_t\mathbf{A}\mathbf{u} = \mathbb{B}_t\mathbf{f}$: GMRES.				
$1/H \backslash 1/h$	8	16	32	64	$1/H \backslash 1/h$	8	16	32	64
4	38	49	62	81	4	44	58	81	117
8	37	43	54	69	8	37	48	66	90
16	-	38	45	57	16	-	41	54	75
32	-	-	42	48	32	-	-	45	62
#iter(\mathbb{A})	54	94	192	x	#iter(\mathbb{A})	54	94	192	x


 FIGURE 2. NIPG method: $\{(\mathcal{Q}^1, \mathcal{T}_h), (\mathcal{Q}^1, \mathcal{T}_H)\}$, $N_s = 16$, Cartesian grids, $\alpha = 1, 2, 4, 8, 16, 32, 64, 128, 256, 512$.

Now we investigate the dependence of the iteration counts needed for convergence on the value of the penalty parameter. Results reported in Figure 2 have been carried out with the NIPG method on Cartesian grids ($1/h = 64$) by choosing $\{(\mathcal{Q}^1, \mathcal{T}_h), (\mathcal{Q}^1, \mathcal{T}_H)\}$ approximation and varying the penalty parameter. More precisely, in Figure 2a we have reported the GMRES iteration counts of the preconditioned system $\mathbb{B}\mathbf{A}\mathbf{u} = \mathbb{B}\mathbf{f}$ with $1/H = 4, 8$, for $\alpha = 1, 2, 4, 8, 16, 32, 64, 128, 256, 512$. The analogous results obtained with the non preconditioned system are reported in Figure 2b. From the results shown in Figure 2 it is clear that, the iteration counts depend on the value of the penalty parameter; but from the numerical results we are not able to deduce explicitly the dependence.

Finally, we present some results carried out by using piecewise continuous polynomials \mathcal{Q}_C^1 for the coarse solver. In Tables 14a and 14b we have compared the iteration counts for the NIPG method ($\alpha = 1$) with $\{(\mathcal{Q}^1, \mathcal{T}_h), (\mathcal{Q}_C^1, \mathcal{T}_H)\}$ and $\{(\mathcal{Q}^1, \mathcal{T}_h), (\mathcal{Q}^1, \mathcal{T}_H)\}$, respectively on Cartesian grids by using as preconditioners the matrix \mathbb{B} . For the sake of brevity we have omitted the analogous results carried out with the preconditioner \mathbb{B}_t . Notice that no significant differences can be observed on the performance of the two-level methods when using a conforming or a discontinuous coarse solver.

TABLE 14. NIPG method ($\alpha = 1$): $N_s = 16$, Cartesian grids.

(a) $\mathbb{B}\mathbf{A}\mathbf{u} = \mathbb{B}\mathbf{f}: \{(\mathcal{Q}^1, \mathcal{T}_h), (\mathcal{Q}_C^1, \mathcal{T}_H)\}$.					(b) $\mathbb{B}\mathbf{A}\mathbf{u} = \mathbb{B}\mathbf{f}: \{(\mathcal{Q}^1, \mathcal{T}_h), (\mathcal{Q}^1, \mathcal{T}_H)\}$.				
$1/H \backslash 1/h$	8	16	32	64	$1/H \backslash 1/h$	8	16	32	64
4	21	36	35	45	8	25	26	29	36
8	21	21	32	42	16	14	21	24	28
16	-	20	21	32	32	-	14	20	23
32	-	-	20	21	64	-	-	14	19
#iter(\mathbb{A})	33	61	117	227	#iter(\mathbb{A})	33	61	117	227

All the experiments presented in this section have been performed also with the BiCGSTAB linear solver instead of GMRES, and we have observed the same behaviour as for the GMRES linear solver. For the sake of brevity, these results have been omitted. We remark that, the GMRES method has a considerably higher cost in terms of memory storage than BiCGSTAB and that the computational cost required by BiCGSTAB is just two matrix vector products per iteration step. Moreover the number of iterations needed for convergence by BiCGSTAB are approximately half as many GMRES steps.

8. CONCLUSIONS

In this paper we have proposed and analysed some new non-overlapping Schwarz methods for the algebraic linear systems obtained from DG discretisations of elliptic problems. Moreover, the results presented in this work apply to more general second order elliptic operators with possibly other kinds of boundary conditions. Both the construction and analysis of the Schwarz methods are provided in a unified framework for a wide class of DG approximations. For symmetric DG methods we have proved optimal estimates for the condition number of the preconditioned systems. For non-symmetric DG approximations, we have demonstrated by numerical computations that the Eisenstat *et al.* GMRES convergence theory [22] cannot be applied for explaining the optimal performance exhibited by the proposed preconditioners. We have presented extensive numerical experiments that confirm the developed theory and assess the optimal convergence of the proposed Schwarz methods, making them a suitable choice for more general and practical applications.

APPENDIX

The appendix is devoted to show Lemma 4.2 that gives an upper bound of the interface bilinear form $I_h^{DG}(\cdot, \cdot)$ defined in (21) and to prove Lemma 4.4 that ensures that a local stability property holds true. We start to show Lemma 4.2. For any $u, v \in V_h$, $I_h^{DG}(u, v)$ could be rewritten as $I_h^{DG}(u, v) = (1 - \gamma_{[\bar{u}]}) E_1^{DG}(u, v) + E_2^{DG}(u, v) + E_3^{DG}(u, v)$, where

$$E_1^{DG}(u, v) = \left[- \int_{\Gamma} u_i \mathbf{n}_i \cdot \nabla_h v_j \, ds - \int_{\Gamma} u_j \mathbf{n}_j \cdot \nabla_h v_i \, ds + \int_{\Gamma} \llbracket u \rrbracket \cdot \{ \nabla_h v \} \, ds \right], \quad (42a)$$

$$E_2^{DG}(u, v) = \left[- \int_{\Gamma} \nabla_h u_i \cdot v_j \mathbf{n}_j \, ds - \int_{\Gamma} \nabla_h u_j \cdot v_i \mathbf{n}_i \, ds + \int_{\Gamma} \{ \nabla_h u \} \cdot \llbracket v \rrbracket \, ds \right], \quad (42b)$$

$$E_3^{DG}(u, v) = \mathcal{G}_{\Gamma}(u, v) + \mathcal{H}(u, v) + \mathcal{F}_{\Gamma}(u, v), \quad (42c)$$

Then, the proof of Lemma 4.2 is a consequence of the following two lemmas, which we show next.

Lemma 8.1. *For any $u \in V_h$, let $u_i \in V_h^i$, $i = 1, \dots, N_s$, be the (unique) functions such that $u = \sum_{i=1}^{N_s} R_i^T u_i$. Then, there exists a constant $C > 0$ depending only on the shape regularity of \mathcal{T}_h and the polynomial degree l_h ,*

such that, for any $u \in V_h$,

$$|E_1^{DG}(u, u)| = |E_2^{DG}(u, u)| \leq C|u|_{1,h} \left\{ \left(\sum_{D \in \mathcal{T}_H} \sum_{E \subset \partial D} h^{-1} \|u\|_{0,E}^2 \right)^{1/2} + |u|_* \right\}. \quad (43)$$

Proof. For any $u \in V_h$, noticing that $\mathbf{n}_i = -\mathbf{n}_j$, we can rewrite $E_1^{DG}(u, u)$ as

$$|E_1^{DG}(u, u)| = \left| \int_{\Gamma} u_i \mathbf{n}_j \cdot \nabla_h u_j \, ds + \int_{\Gamma} u_j \mathbf{n}_i \cdot \nabla_h u_i \, ds + \int_{\Gamma} \llbracket u \rrbracket \cdot \{\!\{ \nabla_h u \}\!\} \, ds \right| = |T_1 + T_2 + T_3|$$

The estimate for T_1 follows straightforwardly by using the Cauchy-Schwarz inequality, the trace estimate (13), together with the inverse inequality (11):

$$|T_1| \leq \left(\sum_{e \in \Gamma} h_e^{-1} \|u_i\|_{0,e}^2 \right)^{1/2} \left(\sum_{e \in \Gamma} h_e \left\| \frac{\partial u_j}{\partial \mathbf{n}_j} \right\|_{0,e}^2 \right)^{1/2} \leq C|u|_{1,h} \left(\sum_{e \in \Gamma} h_e^{-1} \|u_i\|_{0,e}^2 \right)^{1/2}.$$

In view of hypothesis (19), each subdomain Ω_i is the union of some elements $D \in \mathcal{T}_H$ and so, we have

$$\sum_{\Gamma_{ij} \in \Gamma} \sum_{e \in \Gamma_{ij}} h_e^{-1} \|u_i\|_{0,e}^2 \leq C \sum_{D \in \mathcal{T}_H} \sum_{E \subset \partial D} h^{-1} \|u\|_{0,E}^2, \quad (44)$$

where we have also used the regularity and quasi-uniformity properties of the triangulation. The estimate for T_2 follows arguing as for T_1 . To estimate T_3 , proceeding similarly one finds $|T_3| \leq C|u|_*|u|_{1,h}$. Thus, collecting all the estimates we have

$$|T_1| + |T_2| + |T_3| \leq C \left[|u|_{1,h} \left(\sum_{D \in \mathcal{T}_H} \sum_{E \subset \partial D} h^{-1} \|u\|_{0,E}^2 \right)^{1/2} + |u|_* |u|_{1,h} \right],$$

which is (43). The proof is concluded by noticing that $E_1^{DG}(u, u) = E_2^{DG}(u, u)$, for any $u \in V_h$. \square

We finally deal with the estimate for $E_3^{DG}(\cdot, \cdot)$ (see (42c)).

Lemma 8.2. *For any $u \in V_h$, let $u_i \in V_h^i$, $i = 1, \dots, N_s$, be the (unique) functions such that $u = \sum_{i=1}^{N_s} R_i^T u_i$. Then, there exist constants $C, C' > 0$ depending only on the shape regularity of \mathcal{T}_h , the polynomial degree ℓ_h and, for the LDG method, an upper bound on the function β which enters into the definition of the numerical fluxes, such that*

$$|E_3^{DG}(u, u)| \leq C' \alpha^* \sum_{D \in \mathcal{T}_H} \sum_{E \subset \partial D} h^{-1} \|u\|_{0,E}^2 + C \mathcal{A}_h^{DG}(u, u),$$

where $\alpha^* = \max_{e \in \mathcal{E}} \alpha_e$.

Proof. For any $u \in V_h$, by recalling that $|E_3^{DG}(u, u)| = |\mathcal{G}_{\Gamma}(u, u) + \mathcal{H}(u, u) + \mathcal{F}_{\Gamma}(u, u)| \leq |\mathcal{G}_{\Gamma}(u, u)| + |\mathcal{H}(u, u)| + |\mathcal{F}_{\Gamma}(u, u)|$, we proceed to estimate each of the terms. The term $|\mathcal{F}_{\Gamma}(u, u)|$, which is different from zero only for the SIPG(δ) and the LDG methods, can be bounded by using the trace (13) and inverse (11) estimates as in [4]

$$|\mathcal{F}_{\Gamma}(u, u)| = 2 \left| \int_{\Gamma} \tilde{\beta} \cdot \llbracket u \rrbracket \llbracket \nabla_h u \rrbracket \, ds \right| \leq C|u|_{1,h}|u|_* \leq C \|u\|_{\mathbf{H}^1(\Omega, \mathcal{T}_h)}^2 \leq C \mathcal{A}_h^{DG}(u, u),$$

where, for the LDG method, C also depends on an upper bound on the function β which enters into the definition of the numerical fluxes. As far as the term $|\mathcal{H}(u, u)|$ is concerned, by using (16) and the equivalence (15)

we obtain

$$\begin{aligned}
|\mathcal{H}(u, u)| &= \left| \|\mathcal{R}(\llbracket u \rrbracket) + \mathcal{L}(\boldsymbol{\beta} \cdot \llbracket u \rrbracket)\|_{0,\Omega}^2 - \sum_{i=1}^{N_s} \|\mathcal{R}^i(\llbracket u_i \rrbracket) + \mathcal{L}^i(\boldsymbol{\beta} \cdot \llbracket u_i \rrbracket)\|_{0,\Omega_i}^2 \right| \\
&\leq \left| \|\mathcal{R}(\llbracket u \rrbracket) + \mathcal{L}(\boldsymbol{\beta} \cdot \llbracket u \rrbracket)\|_{0,\Omega}^2 \right| + \left| \sum_{i=1}^{N_s} \|\mathcal{R}^i(\llbracket u_i \rrbracket) + \mathcal{L}^i(\boldsymbol{\beta} \cdot \llbracket u_i \rrbracket)\|_{0,\Omega_i}^2 \right| \\
&\leq C|u|_*^2 + \left| \sum_{i=1}^{N_s} \|\mathcal{R}^i(\llbracket u_i \rrbracket) + \mathcal{L}^i(\boldsymbol{\beta} \cdot \llbracket u_i \rrbracket)\|_{0,\Omega_i}^2 \right|.
\end{aligned}$$

The second term on the right hand side can be bounded by using (16), noticing that on each $e \in \mathcal{E}_i^I \cup \mathcal{E}_i^B$ $\llbracket u_i \rrbracket = \llbracket R_i^T u_i \rrbracket = \llbracket u \rrbracket$ and that, thanks to the definition of the jumps on boundary faces, on each $e \in \Gamma_i$ $\|\llbracket u_i \rrbracket\|_{0,e} = \|u_i\|_{0,e}$

$$\begin{aligned}
\left| \sum_{i=1}^{N_s} \|\mathcal{R}^i(\llbracket u_i \rrbracket) + \mathcal{L}^i(\boldsymbol{\beta} \cdot \llbracket u_i \rrbracket)\|_{0,\Omega_i}^2 \right| &\leq C \sum_{i=1}^{N_s} \sum_{e \in \mathcal{E}_i} h_e^{-1} \|\llbracket u_i \rrbracket\|_{0,e}^2 \leq C \left(|u|_*^2 + \sum_{i=1}^{N_s} \sum_{e \in \Gamma_i} h_e^{-1} \|u_i\|_{0,e}^2 \right) \\
&\leq C \left(|u|_*^2 + \sum_{D \in \mathcal{T}_H} \sum_{E \subset \partial D} h^{-1} \|u\|_{0,E}^2 \right),
\end{aligned}$$

where the last step follows from (44).

We are left to show the bound for the term $|\mathcal{G}_\Gamma(u, u)|$. We start with the case $\mathcal{S} = \mathcal{S}^h$ for which $|\mathcal{G}_\Gamma(u, u)|$ is defined as

$$|\mathcal{G}_\Gamma(u, u)| = 2 \left| \sum_{e \in \Gamma} \int_e \alpha_e h_e^{-1} u_i \mathbf{n}_i \cdot u_j \mathbf{n}_j \, ds \right|.$$

The Cauchy-Schwarz inequality, the arithmetic-geometric mean inequality ($2ab \leq a^2 + b^2$) and (44) yield

$$\begin{aligned}
|\mathcal{G}_\Gamma(u, u)| &\leq 2 \left(\sum_{e \in \Gamma} \alpha_e h_e^{-1} \|u_i\|_{0,e}^2 \right)^{1/2} \left(\sum_{e \in \Gamma} \alpha_e h_e^{-1} \|u_j\|_{0,e}^2 \right)^{1/2} \\
&\leq \sum_{e \in \Gamma} \alpha_e h_e^{-1} \|u_i\|_{0,e}^2 + \sum_{e \in \Gamma} \alpha_e h_e^{-1} \|u_j\|_{0,e}^2 \leq C \alpha^* \sum_{D \in \mathcal{T}_H} \sum_{E \subset \partial D} h^{-1} \|u\|_{0,E}^2,
\end{aligned}$$

where, as usual, $\alpha^* = \max_{e \in \mathcal{E}} \alpha_e$. If $\mathcal{S} = \mathcal{S}^r$, then by using the equivalence (14), recalling that on each $e \in \Gamma$, $\|\llbracket u_i \rrbracket\|_{0,e} = \|u_i\|_{0,e}$ and $\|\llbracket u \rrbracket\|_{0,e}^2 = \|\llbracket R_i^T u_i + R_j^T u_j \rrbracket\|_{0,e}^2$, we obtain

$$\begin{aligned}
|\mathcal{G}_\Gamma(u, u)| &\leq C \alpha^* \left(\sum_{e \in \Gamma} h_e^{-1} \|\llbracket R_i^T u_i + R_j^T u_j \rrbracket\|_{0,e}^2 + \sum_{e \in \Gamma} h_e^{-1} \|u_i\|_{0,e}^2 + \sum_{e \in \Gamma} h_e^{-1} \|u_j\|_{0,e}^2 \right) \\
&\leq C \alpha^* \left(\sum_{e \in \Gamma} h_e^{-1} \|u_i\|_{0,e}^2 + \sum_{e \in \Gamma} h_e^{-1} \|u_j\|_{0,e}^2 \right) \leq C \alpha^* \sum_{D \in \mathcal{T}_H} \sum_{E \subset \partial D} h^{-1} \|u\|_{0,E}^2,
\end{aligned}$$

where we have also used that, on each $e \in \Gamma$, $\|\llbracket R_i^T u_i \rrbracket\|_{0,e}^2 = \|u_i \mathbf{n}_i\|_{0,e}^2 = \|u_i\|_{0,e}^2$ and the inequality (44). Taking into account all the previous contributions the proof is complete. \square

We notice that Lemmas 8.1 and 8.2 imply

$$|I_h^{DG}(u, u)| \leq C \left(\mathcal{A}_h(u, u) + \alpha^* \sum_{D \in \mathcal{T}_H} h^{-1} \|u\|_{0, \partial D}^2 \right) \quad \forall u \in V_h, \quad (45)$$

from which Lemma 4.2 follows thanks to the trace estimate (18).

Proof of Lemma 4.4. By taking into account the definition of the global and local bilinear forms, a straightforward calculation show that, for all $u_i, v_i \in V_h^i$,

$$\mathcal{A}_h^{DG}(R_i^T u_i, R_i^T u_i) = \mathcal{A}_i^{DG}(u_i, u_i) + E(u_i, u_i), \quad (46)$$

where

$$\begin{aligned} E(u_i, u_i) = & \int_{\Gamma_i} \left[\frac{1}{2} \llbracket \widehat{u}(R_i^T u_i) - R_i^T u_i \rrbracket - (\widehat{u}_i(u_i) - u_i) \mathbf{n}_i \right] \cdot \nabla_h u_i \, ds + \int_{\Gamma_i} \{ \widehat{u}(R_i^T u_i) - R_i^T u_i \} \cdot \llbracket \nabla_h R_i^T u_i \rrbracket \, ds \\ & - \int_{\mathcal{E}} \widehat{\boldsymbol{\sigma}}(\boldsymbol{\sigma}(R_i^T u_i), R_i^T u_i) \cdot \llbracket R_i^T u_i \rrbracket \, ds + \int_{\mathcal{E}_i} \widehat{\boldsymbol{\sigma}}_i(\boldsymbol{\sigma}_i(u_i), u_i) \cdot \llbracket u_i \rrbracket \, ds, \end{aligned} \quad (47)$$

where we have taken into consideration that for all the methods we are considering the global and local numerical fluxes are conservative, *i.e.*, $\llbracket \widehat{\boldsymbol{\sigma}} \rrbracket = \llbracket \widehat{\boldsymbol{\sigma}}_i \rrbracket = 0$, $\{ \widehat{\boldsymbol{\sigma}} \} = \widehat{\boldsymbol{\sigma}}$ and $\{ \widehat{\boldsymbol{\sigma}}_i \} = \widehat{\boldsymbol{\sigma}}_i$. Thanks to (46), in order to prove (29) it is enough to show that $|E(u_i, u_i)| = |E_a + E_b + E_c| \leq C_\omega \mathcal{A}_i^{DG}(u_i, u_i)$, where we have denoted by E_a and E_b the first and second integral, respectively, that appear in (47) and by E_c all the remaining terms. We first recall the following useful inequality that trivially follows from a standard application of the inverse (11) and trace (13) inequalities

$$\int_{\Gamma_i} \nabla_h u_i \cdot \mathbf{n}_i u_i \, ds \leq \frac{C}{\sqrt{\alpha_*}} \left(\sum_{\substack{T \in \mathcal{T}_h \\ T \subset \Omega_i}} |u_i|_{1, T}^2 \right)^{1/2} \left(\sum_{e \in \mathcal{E}_i} \alpha_e h_e^{-1} \llbracket u_i \rrbracket_{0, e}^2 \right)^{1/2} \leq \frac{C}{\sqrt{\alpha_*}} \mathcal{A}_i^{DG}(u_i, u_i),$$

where C only depends on the shape regularity of \mathcal{T}_h and the polynomial degree and where $\alpha_* = \min_{e \in \mathcal{E}} \alpha_e$. For all the methods we are considering, by recalling the definitions of the global numerical fluxes and using the definition on R_i^T on the interface we obtain $\llbracket \widehat{u}(R_i^T u_i) - R_i^T u_i \rrbracket = -(1 - \gamma_{[\widehat{u}]}) u_i \mathbf{n}_i$ on each $e \in \Gamma_i$, where $\gamma_{[\widehat{u}]} = 0$ for all the symmetric DG methods and $\gamma_{[\widehat{u}]} = 2, 1$ for the NIPG and IIPG methods, respectively. Moreover we have $\{ \widehat{u}(R_i^T u_i) - R_i^T u_i \} = -\tilde{\boldsymbol{\beta}} \cdot u_i \mathbf{n}_i$ with $\tilde{\boldsymbol{\beta}} = \boldsymbol{\beta}$ for the LDG method, $\tilde{\boldsymbol{\beta}} = -\llbracket 1/2 - \delta \rrbracket / 2$ for the SIPG(δ) method and $\tilde{\boldsymbol{\beta}} = \mathbf{0}$ otherwise. Then, by using that $(\widehat{u}_i(u_i) - u_i) \mathbf{n}_i = -(1 - \gamma_{[\widehat{u}]}) u_i \mathbf{n}_i$ and $|1 - \gamma_{[\widehat{u}]}| \leq 1$, we get

$$|E_a + E_b| = \left| \frac{1}{2} (1 - \gamma_{[\widehat{u}]}) \int_{\Gamma_i} \nabla_h u_i \cdot \mathbf{n}_i u_i \, ds - \int_{\Gamma_i} \tilde{\boldsymbol{\beta}} \cdot \mathbf{n}_i \nabla_h u_i \cdot \mathbf{n}_i u_i \, ds \right| \leq \frac{C}{\sqrt{\alpha_*}} \mathcal{A}_i^{DG}(u_i, u_i),$$

where, for the LDG method, C also depends on an upper bound on the function $\boldsymbol{\beta}$ which enters into the definition of the numerical fluxes. Concerning the estimate of the term E_c , for all the methods in primal form stabilised with the jump term, *i.e.*, all except the BRMPS method, by taking into account the definition of the global and local numerical fluxes on each face $e_i \in \Gamma_i$, we have $\widehat{\boldsymbol{\sigma}}(\boldsymbol{\sigma}(R_i^T u_i), R_i^T u_i) = (1/2 + \tilde{\boldsymbol{\beta}} \cdot \mathbf{n}_i) \nabla_h u_i - \alpha_e h_e^{-1} u_i \mathbf{n}_i$ and $\widehat{\boldsymbol{\sigma}}_i(\boldsymbol{\sigma}_i(u_i), u_i) = \nabla_h u_i - \alpha_e h_e^{-1} u_i \mathbf{n}_i$, respectively, where $\tilde{\boldsymbol{\beta}} = -\llbracket 1/2 - \delta \rrbracket / 2 \neq \mathbf{0}$ only for the SIPG(δ) method. Then,

$$|E_c| = \left| \int_{\Gamma_i} \left(\frac{1}{2} - \tilde{\boldsymbol{\beta}} \cdot \mathbf{n}_i \right) \nabla_h u_i \cdot \mathbf{n}_i u_i \, ds \right| \leq \frac{C}{\sqrt{\alpha_*}} \mathcal{A}_i^{DG}(u_i, u_i),$$

where in the last step we have used that $|1/2 - \tilde{\boldsymbol{\beta}} \cdot \mathbf{n}_i| \leq 1$ thanks to the fact that $\delta \in [0, 1]$. For the BRMPS method, proceeding as before, on each face $e_i \in \Gamma_i$, we get $\hat{\boldsymbol{\sigma}}(\boldsymbol{\sigma}(R_i^T u_i), R_i^T u_i) = (\nabla_h u_i)/2 + \alpha_e \{r_e(\llbracket R_i^T u_i \rrbracket)\}$ and $\hat{\boldsymbol{\sigma}}_i(\boldsymbol{\sigma}_i(u_i), u_i) = \nabla_h u_i + \alpha_e r_e^i(u_i \mathbf{n}_i)$. Then,

$$|E_c| \leq \left| \frac{1}{2} \int_{\Gamma_i} \nabla_h u_i \cdot \mathbf{n}_i u_i \, ds \right| + \left| \int_{\Gamma_i} \alpha_e \{r_e(\llbracket R_i^T u_i \rrbracket)\} \cdot \mathbf{n}_i u_i \, ds \right| + \left| \int_{\Gamma_i} \alpha_e r_e^i(u_i \mathbf{n}_i) \cdot \mathbf{n}_i u_i \, ds \right| \leq |I| + |II| + |III|.$$

The term $|I|$ can be directly estimate by $|I| \leq C \alpha_*^{-1/2} \mathcal{A}_i^{DG}(u_i, u_i)$. For the term $|II|$, by recalling the definition of the lifting operators and the equivalence (14)

$$|II| = \left| \int_{\Gamma_i} \alpha_e \{r_e(\llbracket R_i^T u_i \rrbracket)\} \cdot \llbracket R_i^T u_i \rrbracket \, ds \right| = \sum_{e \in \Gamma_i} \alpha_e \|r_e(\llbracket R_i^T u_i \rrbracket)\|_{0,\Omega}^2 \leq C \sum_{e \in \Gamma_i} \alpha_e h_e^{-1} \|\llbracket R_i^T u_i \rrbracket\|_{0,e}^2 \leq C \mathcal{A}_i^{DG}(u_i, u_i).$$

For the term $|III|$, reasoning in the same way and taking into account that each $e \in \Gamma_i$ is a boundary face for the local bilinear form, we obtain

$$|III| = \left| \int_{\Gamma_i} \alpha_e r_e^i(u_i \mathbf{n}_i) \cdot u_i \mathbf{n}_i \, ds \right| = \sum_{e \in \Gamma_i} \alpha_e \|r_e^i(u_i \mathbf{n}_i)\|_{0,\Omega_i}^2 \leq C \sum_{e \in \Gamma_i} \alpha_e h_e^{-1} \|u_i \mathbf{n}_i\|_{0,e}^2 \leq C \mathcal{A}_i^{DG}(u_i, u_i).$$

By collecting the estimates on the terms, we obtain $|E_c| \leq C(1 + \alpha_*^{-1/2}) \mathcal{A}_i^{DG}(u_i, u_i)$.

We now deal with methods in mixed form. For the LDG method, we first notice that E_c can be rewritten as

$$\begin{aligned} E_c &= - \int_{\mathcal{E}} \hat{\boldsymbol{\sigma}}(R_i^T u_i) \cdot \llbracket R_i^T u_i \rrbracket \, ds + \int_{\mathcal{E}_i} \hat{\boldsymbol{\sigma}}_i(u_i) \cdot \llbracket u_i \rrbracket \, ds \\ &= \sum_{e \in \Gamma_i} \int_e \left(\frac{1}{2} - \boldsymbol{\beta} \cdot \mathbf{n}_i \right) \nabla_h u_i \cdot \mathbf{n}_i u_i \, ds + \underbrace{\| \mathcal{R}(\llbracket R_i^T u_i \rrbracket) + \mathcal{L}(\boldsymbol{\beta} \cdot \llbracket R_i^T u_i \rrbracket) \|_{0,\Omega}^2 - \| \mathcal{R}^i(\llbracket u_i \rrbracket) + \mathcal{L}^i(\boldsymbol{\beta} \cdot \llbracket u_i \rrbracket) \|_{0,\Omega_i}^2}_{IV}. \end{aligned}$$

The term IV can be bounded directly by using the Cauchy-Schwarz inequality and estimates (16) and (14)

$$|IV| \leq C \left(\sum_{e \in \mathcal{E}} h_e^{-1} \|\llbracket R_i^T u_i \rrbracket\|_{0,e}^2 + \sum_{e \in \mathcal{E}_i} h_e^{-1} \|\llbracket u_i \rrbracket\|_{0,e}^2 \right) \leq 2C \sum_{e \in \mathcal{E}_i} h_e^{-1} \|\llbracket u_i \rrbracket\|_{0,e}^2 \leq 2 \frac{C}{\alpha_*} \mathcal{A}_i^{DG}(u_i, u_i).$$

Then, taking into account the previous bound, we reach

$$|E_c| \leq \left| \int_{\Gamma_i} \left(\frac{1}{2} - \boldsymbol{\beta} \cdot \mathbf{n}_i \right) \nabla_h u_i \cdot \mathbf{n}_i u_i \, ds \right| + 2 \frac{C}{\alpha_*} \mathcal{A}_i^{DG}(u_i, u_i) \leq \frac{C}{\sqrt{\alpha_*}} \mathcal{A}_i^{DG}(u_i, u_i).$$

For the BMMPR method, by reasoning in the same way as for the LDG method and taking into account the different type of stabilisation term, we get $|E_c| \leq C(1 + \alpha_*^{-1/2}) \mathcal{A}_i^{DG}(u_i, u_i)$. Collecting all the previous estimates we finally obtain

$$|E(u_i, u_i)| \leq C \left(C_{r_e} + \frac{1}{\sqrt{\alpha_*}} \right) \mathcal{A}_i^{DG}(u_i, u_i) = C_\omega \mathcal{A}_i^{DG}(u_i, u_i),$$

where C only depends on the shape regularity constant, the polynomial degree and, for the LDG method, an upper bound on the function $\boldsymbol{\beta}$ which enters into the definition of the numerical fluxes. Moreover $C_{r_e} \neq 0$ only for the methods in which the stability term involves the local lifting operator r_e . \square

Acknowledgements. The authors are grateful to Annalisa Buffa e Ilaria Perugia for the careful reading of the manuscript and for the multiple discussions which took place while developing this work. Special thanks go to Valeria Simoncini for her precious guide on linear algebra topics. We are in-debt to Franco Brezzi for being, as usual, a source of inspiration. The authors are also extremely thankful to the referees for the thorough and constructive comments which have greatly contributed to the improvement of the paper.

REFERENCES

- [1] R.A. Adams, *Sobolev spaces*. Academic Press [A subsidiary of Harcourt Brace Jovanovich, Publishers], New York-London, Pure and Applied Mathematics, Vol. **65** (1975).
- [2] P.F. Antonietti, A. Buffa and I. Perugia, Discontinuous Galerkin approximation of the Laplace eigenproblem. *Comput. Methods Appl. Mech. Engrg.* **195** (2006) 3483–3503.
- [3] D.N. Arnold, An interior penalty finite element method with discontinuous elements. *SIAM J. Numer. Anal.* **19** (1982) 742–760.
- [4] D.N. Arnold, F. Brezzi, B. Cockburn and L.D. Marini, Unified analysis of discontinuous Galerkin methods for elliptic problems. *SIAM J. Numer. Anal.* **39** (2001/02) 1749–1779 (electronic).
- [5] I. Babuška and M. Zlámal, Nonconforming elements in the finite element method with penalty. *SIAM J. Numer. Anal.* **10** (1973) 863–875.
- [6] F. Bassi and S. Rebay, A high-order accurate discontinuous finite element method for the numerical solution of the compressible Navier-Stokes equations. *J. Comput. Phys.* **131** (1997) 267–279.
- [7] F. Bassi, S. Rebay, G. Mariotti, S. Pedinotti and M. Savini, A high-order accurate discontinuous finite element method for inviscid and viscous turbomachinery flows, in *Proceedings of the 2nd European Conference on Turbomachinery Fluid Dynamics and Thermodynamics*, R. Decuyper and G. Dibelius Eds., Antwerpen, Belgium (1997) 99–108, Technologisch Instituut.
- [8] C.E. Baumann and J.T. Oden, A discontinuous *hp* finite element method for convection-diffusion problems. *Comput. Methods Appl. Mech. Engrg.* **175** (1999) 311–341.
- [9] S.C. Brenner, Poincaré-Friedrichs inequalities for piecewise H^1 functions. *SIAM J. Numer. Anal.* **41** (2003) 306–324 (electronic).
- [10] S.C. Brenner and K. Wang, Two-level additive Schwarz preconditioners for C^0 interior penalty methods. *Numer. Math.* **102** (2005) 231–255.
- [11] F. Brezzi, G. Manzini, D. Marini, P. Pietra and A. Russo, Discontinuous Galerkin approximations for elliptic problems. *Numer. Methods Partial Differ. Equ.* **16** (2000) 365–378.
- [12] F. Brezzi, L.D. Marini and E. Süli, Discontinuous Galerkin methods for first-order hyperbolic problems. *Math. Models Methods Appl. Sci.* **14** (2004) 1893–1903.
- [13] X.-C. Cai and O.B. Widlund, Domain decomposition algorithms for indefinite elliptic problems. *SIAM J. Sci. Statist. Comput.* **13** (1992) 243–258.
- [14] P.E. Castillo, *Local Discontinuous Galerkin methods for convection-diffusion and elliptic problems*. Ph.D. thesis, University of Minnesota, Minneapolis (2001).
- [15] P. Castillo, B. Cockburn, I. Perugia and D. Schötzau, An a priori error analysis of the local discontinuous Galerkin method for elliptic problems. *SIAM J. Numer. Anal.* **38** (2000) 1676–1706 (electronic).
- [16] P.G. Ciarlet, *The finite element method for elliptic problems*. North-Holland Publishing Co., Amsterdam, Studies in Mathematics and its Applications, Vol. **4** (1978).
- [17] B. Cockburn, Discontinuous Galerkin methods for convection-dominated problems, in *High-order methods for computational physics*, Springer, Berlin, *Lect. Notes Comput. Sci. Eng.* **9** (1999) 69–224.
- [18] B. Cockburn and C.-W. Shu, The local discontinuous Galerkin method for time-dependent convection-diffusion systems. *SIAM J. Numer. Anal.* **35** (1998) 2440–2463 (electronic).
- [19] B. Cockburn, G.E. Karniadakis, and C.-W. Shu. The development of discontinuous Galerkin methods, in *Discontinuous Galerkin methods* (Newport, RI, 1999), Springer, Berlin, *Lect. Notes Comput. Sci. Eng.* **11** (2000) 3–50.
- [20] C. Dawson, S. Sun and M.F. Wheeler, Compatible algorithms for coupled flow and transport. *Comput. Methods Appl. Mech. Engrg.* **193** (2004) 2565–2580.
- [21] J. Douglas, Jr., and T. Dupont. Interior penalty procedures for elliptic and parabolic Galerkin methods, in *Computing methods in applied sciences* (Second Internat. Sympos., Versailles, 1975), Springer, Berlin, *Lect. Notes Phys.* **58** (1976) 207–216.
- [22] S.C. Eisenstat, H.C. Elman and M.H. Schultz, Variational iterative methods for nonsymmetric systems of linear equations. *SIAM J. Numer. Anal.* **20** (1983) 345–357.
- [23] X. Feng and O.A. Karakashian, Two-level additive Schwarz methods for a discontinuous Galerkin approximation of second order elliptic problems. *SIAM J. Numer. Anal.* **39** (2001) 1343–1365 (electronic).
- [24] X. Feng and O.A. Karakashian, Analysis of two-level overlapping additive Schwarz preconditioners for a discontinuous Galerkin method. In *Domain decomposition methods in science and engineering* (Lyon, 2000), Theory Eng. Appl. Comput. Methods, Internat. Center Numer. Methods Eng. (CIMNE), Barcelona (2002) 237–245.
- [25] G.H. Golub and C.F. Van Loan, *Matrix computations*. Johns Hopkins Studies in the Mathematical Sciences, Johns Hopkins University Press, Baltimore, MD, third edition (1996).

- [26] J. Gopalakrishnan and G. Kanschat. Application of unified DG analysis to preconditioning DG methods, in *Computational Fluid and Solid Mechanics 2003*, K.J. Bathe Ed., Elsevier (2003) 1943–1945.
- [27] J. Gopalakrishnan and G. Kanschat, A multilevel discontinuous Galerkin method. *Numer. Math.* **95** (2003) 527–550.
- [28] B. Heinrich and K. Pietsch, Nitsche type mortaring for some elliptic problem with corner singularities. *Computing* **68** (2002) 217–238.
- [29] P. Houston and E. Süli, *hp*-adaptive discontinuous Galerkin finite element methods for first-order hyperbolic problems. *SIAM J. Sci. Comput.* **23** (2001) 1226–1252 (electronic).
- [30] C. Lasser and A. Toselli, An overlapping domain decomposition preconditioner for a class of discontinuous Galerkin approximations of advection-diffusion problems. *Math. Comp.* **72** (2003) 1215–1238 (electronic).
- [31] P. Le Tallec, Domain decomposition methods in computational mechanics. *Comput. Mech. Adv.* **1** (1994) 121–220.
- [32] P.-L. Lions, On the Schwarz alternating method. I, in *First International Symposium on Domain Decomposition Methods for Partial Differential Equations* (Paris, 1987), SIAM, Philadelphia, PA (1988) 1–42.
- [33] P.-L. Lions, On the Schwarz alternating method. II. Stochastic interpretation and order properties, in *Domain decomposition methods* (Los Angeles, CA, 1988), SIAM, Philadelphia, PA (1989) 47–70.
- [34] P.-L. Lions, On the Schwarz alternating method. III. A variant for nonoverlapping subdomains, in *Third International Symposium on Domain Decomposition Methods for Partial Differential Equations* (Houston, TX, 1989), SIAM, Philadelphia, PA (1990) 202–223.
- [35] W.H. Reed and T. Hill, Triangular mesh methods for the neutron transport equation. Technical Report LA-UR-73-479, Los Alamos Scientific Laboratory (1973).
- [36] B. Rivière, M.F. Wheeler and V. Girault, Improved energy estimates for interior penalty, constrained and discontinuous Galerkin methods for elliptic problems. I. *Comput. Geosci.* **3** (1999) 337–360.
- [37] B. Rivière, M.F. Wheeler and V. Girault, A priori error estimates for finite element methods based on discontinuous approximation spaces for elliptic problems. *SIAM J. Numer. Anal.* **39** (2001) 902–931 (electronic).
- [38] Y. Saad and M.H. Schultz, GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM J. Sci. Statist. Comput.* **7** (1986) 856–869.
- [39] M. Sarkis and D.B. Szyld, Optimal left and right additive Schwarz preconditioning for Minimal Residual methods with euclidean and energy norms. *Comput. Methods Appl. Mech. Engrg.* **196** (2007) 1612–1621.
- [40] B.F. Smith, P.E. Bjørstad and W.D. Gropp, *Domain decomposition*. Cambridge University Press, Cambridge, Parallel multi-level methods for elliptic partial differential equations (1996).
- [41] G. Starke, Field-of-values analysis of preconditioned iterative methods for nonsymmetric elliptic problems. *Numer. Math.* **78** (1997) 103–117.
- [42] R. Stenberg, Mortaring by a method of J. A. Nitsche, in *Computational mechanics* (Buenos Aires, 1998), pages CD-ROM file. Centro Internac. Métodos Numér. Ing., Barcelona (1998).
- [43] A. Toselli and O. Widlund, *Domain decomposition methods—algorithms and theory*, Springer Series in Computational Mathematics **34**, Springer-Verlag, Berlin (2005).
- [44] M.F. Wheeler, An elliptic collocation-finite element method with interior penalties. *SIAM J. Numer. Anal.* **15** (1978) 152–161.
- [45] J.H. Wilkinson, *The algebraic eigenvalue problem*. Monographs on Numerical Analysis, The Clarendon Press Oxford University Press, New York (1988), Oxford Science Publications.
- [46] J. Xu, Iterative methods by space decomposition and subspace correction. *SIAM Rev.* **34** (1992) 581–613.
- [47] J. Xu and L. Zikatanov, The method of alternating projections and the method of subspace corrections in Hilbert space. *J. Amer. Math. Soc.* **15** (2002) 573–597 (electronic).
- [48] J. Xu and J. Zou. Some nonoverlapping domain decomposition methods. *SIAM Rev.* **40** (1998) 857–914 (electronic).