Journal of Computational and Applied Mathematics 275 (2015) 61–78

Contents lists available at ScienceDirect

Journal of Computational and Applied Mathematics

journal homepage: www.elsevier.com/locate/cam

Scalable two-level preconditioning and deflation based on a piecewise constant subspace for (SIP)DG systems for diffusion problems

P. van Slingerland, C. Vuik*

Delft Institute of Applied Mathematics, Delft University of Technology, Mekelweg 4, 2628 CD Delft, The Netherlands

HIGHLIGHTS

- We consider a solver for the discontinuous Galerkin method.
- We use the Symmetric Interior Penalty variant.
- The deflated solver appears to be scalable.
- Our solver is independent of the jump in the coefficients.

ARTICLE INFO

Article history:
Received 10 October 2012
Received in revised form 28 May 2014

MSC:
65F10
65N30
65N55

Keywords:
Two-level preconditioning
Deflation
Symmetric interior penalty Galerkin discretization
Conjugate gradient method
Diffusion problems

ABSTRACT

This paper is focused on the preconditioned Conjugate Gradient (CG) method for linear systems resulting from Symmetric Interior Penalty (discontinuous) Galerkin (SIPG) discretizations for stationary diffusion problems. In particular, it concerns two-level preconditioning strategies where the coarse space is based on piecewise constant DG basis functions. In this paper, we show that both the two-level preconditioner and the corresponding BNN (or ADEF2) deflation variant yield scalable convergence of the CG method (independent of the mesh element diameter). These theoretical results are illustrated by numerical experiments.

1. Introduction

The discontinuous Galerkin method can be interpreted as a finite volume method that uses piecewise polynomials of degree p rather than piecewise constant functions. As such, it combines the best of both classical finite element methods and finite volume methods, making it particularly suitable for handling non-matching grids and designing hp-refinement strategies. However, the resulting linear systems are usually larger than those for the aforementioned classical discretization schemes. This is due to the larger number of unknowns per mesh element. At the same time, the condition number typically increases with the number of mesh elements, the polynomial degree, and the stabilization factor [1,2]. Problems with strong variations in the coefficients pose an extra challenge. Altogether, standard linear solvers often result in long computational times and/or low accuracy.

* Corresponding author.
E-mail address: c.vuik@tudelft.nl (C. Vuik).

http://dx.doi.org/10.1016/j.cam.2014.06.028

In search of efficient and scalable algorithms (for which the number of iterations does not increase with e.g. the number of mesh elements), much attention has been paid to subspace correction methods [3]. Examples include classical geometric (h-)multigrid [4–6], spectral (p-)multigrid [7–9], algebraic multigrid [10,11], Schwarz domain decomposition [12,13], and mixtures of these [14]. Usually, these methods can either be used as a standalone solver, or as a preconditioner within an iterative Krylov method. The latter tends to be more robust for problems with a few isolated 'bad' eigenvalues, as is typical for problems with large contrasts in the coefficients.

This research is focused on the preconditioned Conjugate Gradient (CG) method for linear systems resulting from Symmetric Interior Penalty (discontinuous) Galerkin (SIPG) discretizations for stationary diffusion problems. In particular, it concerns two-level preconditioning strategies where the coarse space is based on the piecewise constant DG basis functions.

The latter strategy has been introduced by Dobrev et al. [15]. In [16], their method has been carried over to deflation with the help of the analysis in [17]: it has been demonstrated numerically that both two-level variants yield fast and scalable CG convergence using (damped) block Jacobi smoothing.

To provide theoretical support for this, we derive bounds for the condition number of the preconditioned/deflated system which are independent of the mesh element diameter (the influence of the polynomial degree and the diffusion coefficient on these theoretical bounds is not considered in this paper). Such bounds are already available for the preconditioning variant with $p = 1$, as derived in [15] using the work of Falgout et al. [18]. Here, we extend these results by allowing $p \geq 1$. Furthermore, we include BNN/ADEF2 deflation in the analysis. Additionally, we demonstrate that the required restrictions on the smoother are satisfied for (damped) block Jacobi smoothing. Finally, we extend the numerical support in [16] for these results by studying test problems with strong variations in the coefficients.

The outline of this paper is as follows. Section 2 specifies both two-level methods for the linear SIPG systems under consideration. Section 3 derives an auxiliary regularity result, which is used to derive the main scalability result in Section 4. Numerical experiments are discussed in Section 5. Finally, we summarize the main conclusions in Section 6. For more details, we refer to our technical report [19].

2. Methods and assumptions

This section specifies the methods and assumptions that we consider in this paper. Section 2.1 discusses the diffusion model under consideration and discretizes it by means of the SIPG method. Section 2.2 considers two two-level preconditioning strategies for solving the resulting linear system by means of the preconditioned CG method.

2.1. SIPG discretization for diffusion problems

Model problem. We study the following diffusion problem on the d-dimensional domain Ω with source term f, scalar diffusion coefficient K (bounded below and above by positive constants), and a combination of Dirichlet and Neumann boundary conditions (specified by g_D on $\partial\Omega_D \neq \emptyset$ and g_N on $\partial\Omega_N$ with outward normal $\mathbf{n}$ respectively):

$$\begin{aligned} -\nabla\cdot(K\nabla u) &= f, \quad \text{in } \Omega,\\ u &= g_D, \quad \text{on } \partial\Omega_D \neq \emptyset,\\ K\nabla u\cdot\mathbf{n} &= g_N, \quad \text{on } \partial\Omega_N. \end{aligned} \tag{1}$$

We assume that the model parameters are chosen such that a weak solution of this model problem exists (cf. [20] for specific sufficient conditions). Furthermore, we assume that Ω is either an interval ($d = 1$), polygon ($d = 2$) or polyhedron ($d = 3$).

Mesh. To discretize the model problem (1), we subdivide Ω into mesh elements $E_1, \ldots, E_N$ with maximum element diameter h.

We assume that each mesh element E_i is affine-equivalent with a certain reference element E_0
that is an interval/polygon/polyhedron (independent of h) with mutually affine-equivalent edges. (2)

Note that all meshes consisting entirely of either intervals, triangles, tetrahedrons, parallelograms, or parallelepipeds satisfy this property. The mesh does not need to be conforming.

Furthermore, we assume that the mesh is regular in the sense of [21, p. 124]. To specify this property, for all $i = 0, \ldots, N$, let h_i and ρ_i denote the diameter of E_i, and the diameter of the largest ball contained in E_i respectively. We can now define regularity as[1]:

$$\frac{h_i}{\rho_i} \lesssim 1, \quad \forall i = 1, \ldots, N. \tag{3}$$

[1] Throughout this paper, we use the symbol $\lesssim$ in expressions of the form "$F(x) \lesssim G(x)$ for all $x \in X$" to indicate that there exists a constant $C > 0$, independent of the variable x and the maximum mesh element diameter h (or the number of mesh elements), such that $F(x) \leq CG(x)$ for all $x \in X$. The symbol $\gtrsim$ is defined similarly.

Finally, we assume that the mesh is quasi-uniform in the following sense:

$$\frac{h}{h_i} \lesssim 1, \quad \forall i = 1, \ldots, N. \tag{4}$$

SIPG method. Now that we have specified the mesh, we can construct an SIPG approximation for our model problem (1). To this end, define the test space V that contains each function that is a polynomial of degree p or lower within each mesh element, and that may be discontinuous at the element boundaries. The SIPG approximation u_h is now defined as the unique element in this test space that satisfies the relation $B(u_h, v) = L(v)$ for all test functions $v \in V$, where B and L are certain (bi)linear forms that are specified hereafter.

To specify these forms, we use the following notation: let Γ_h denote the collection of all edges $e = \partial E_i \cap \partial E_j$ in the interior. Similarly, let Γ_D and Γ_N denote the collection of all edges (points/lines/polygons) at the Dirichlet and Neumann boundary respectively. Additionally, for all edges e, let h_e denote the length of the largest mesh element adjacent to e for one-dimensional problems, the length of e for two-dimensional problems, and the square root of the surface area of e for three-dimensional problems. Finally, we introduce the usual trace operators for jumps and averages at the mesh element boundaries: in the interior, we define at $\partial E_i \cap \partial E_j$: $[\mathbf{v}] = \mathbf{v}_i \cdot \mathbf{n}_i + \mathbf{v}_j \cdot \mathbf{n}_j$, and $\{\mathbf{v}\} = \frac{1}{2}(\mathbf{v}_i + \mathbf{v}_j)$, where $\mathbf{v}_i$ denotes the trace of the (scalar or vector-valued) function $\mathbf{v}$ along the side of E_i with outward normal $\mathbf{n}_i$. Similarly, at the domain boundary, we define at $\partial E_i \cap \partial \Omega$: $[\mathbf{v}] = \mathbf{v}_i \cdot \mathbf{n}_i$, and $\{\mathbf{v}\} = \mathbf{v}_i$. Using this notation, the forms B and L can be defined as follows (for one-dimensional problems, the boundary integrals below should be interpreted as function evaluations of the integrand):

$$\begin{aligned}
&B_\Omega(u_h, v) = \sum_{i=1}^{N} \int_{E_i} K\nabla u_h \cdot \nabla v, \qquad B_\sigma(u_h, v) = \sum_{e \in \Gamma_h \cup \Gamma_D} \int_e \frac{\sigma}{h_e} [u_h] \cdot [v], \\
&B_r(u_h, v) = - \sum_{e \in \Gamma_h \cup \Gamma_D} \int_e \Big(\{K\nabla u_h\} \cdot [v] + [u_h] \cdot \{K\nabla v\} \Big), \\
&B(u_h, v) = B_\Omega(u_h, v) + B_\sigma(u_h, v) + B_r(u_h, v), \\
&L(v) = \int_\Omega f v - \sum_{e \in \Gamma_D} \int_e \left([K\nabla v] + \frac{\sigma}{h_e} v \right) g_D + \sum_{e \in \Gamma_N} \int_e v g_N,
\end{aligned} \tag{5}$$

where $\sigma \geq 0$ is the so-called penalty parameter.

This parameter penalizes the inter-element jumps to enforce weak continuity required for convergence. Its value may vary throughout the domain, and we assume that it is bounded below and above by positive constants (independent of the maximum element diameter h). Furthermore, we assume that the scheme is coercive in the sense that:

$$0 < B_\Omega(v, v) + B_\sigma(v, v) \lesssim B(v, v), \quad \forall v \in V. \tag{6}$$

For a large class of problems, it has been shown in [20, pp. 38–40] that this condition is satisfied as long as the penalty parameter is sufficiently large.

Linear system. In order to compute the SIPG approximation, we choose a basis for the test space V: for each mesh element E_i, we set the basis function $\phi_1^{(i)}$ equal to zero in the entire domain, except in E_i, where it is equal to one. Similarly, we define higher-order basis functions $\phi_2^{(i)}, \ldots, \phi_m^{(i)}$, which are a higher-order polynomial in E_i and zero elsewhere. Note that, e.g., for one-dimensional problems, we have $m = p + 1$ basis functions.

More specifically, the basis functions are constructed as follows. For all $i = 1, \ldots, N$, let $F_i : E_i \to E_0$ denote an invertible affine mapping (which exists by assumption (2)). Furthermore, let the functions $\phi_k^{(0)} : E_0 \to \mathbb{R}$ (with $k = 1, \ldots, m$) form a basis for the space of all polynomials of degree p and lower on the reference element (setting $\phi_1^{(0)} = 1$). Using this basis on the reference element, the basis function $\phi_k^{(i)}$ is zero in the entire domain, except in the mesh element E_i, where it reads $\phi_k^{(i)} = \phi_k^{(0)} \circ F_i$.

Now that we have defined the basis functions, we can express u_h as a linear combination of these functions: $u_h = \sum_{i=1}^{N} \sum_{k=1}^{m} u_k^{(i)} \phi_k^{(i)}$. The new unknowns $u_k^{(i)}$ in this expression can be determined by solving a linear system $A\mathbf{u} = \mathbf{b}$ of following form (also cf. e.g. [20]):

$$\begin{bmatrix} A_{11} & A_{12} & \cdots & A_{1N} \\ A_{21} & A_{22} & & \vdots \\ \vdots & & \ddots & \\ A_{N1} & \cdots & & A_{NN} \end{bmatrix} \begin{bmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \vdots \\ \mathbf{u}_N \end{bmatrix} = \begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \\ \vdots \\ \mathbf{b}_N \end{bmatrix}, \tag{7}$$

where the blocks all have dimension m, and where, for all $i, j = 1, \dots, N$:

$$A_{ji} = \begin{bmatrix} B(\phi_1^{(i)}, \phi_1^{(j)}) & B(\phi_2^{(i)}, \phi_1^{(j)}) & \cdots & B(\phi_m^{(i)}, \phi_1^{(j)}) \\ B(\phi_1^{(i)}, \phi_2^{(j)}) & B(\phi_2^{(i)}, \phi_2^{(j)}) & & \vdots \\ \vdots & & \ddots & \\ B(\phi_1^{(i)}, \phi_m^{(j)}) & & \cdots & B(\phi_m^{(i)}, \phi_m^{(j)}) \end{bmatrix}, \qquad \mathbf{u}_i = \begin{bmatrix} u_1^{(i)} \\ u_2^{(i)} \\ \vdots \\ u_m^{(i)} \end{bmatrix}, \qquad \mathbf{b}_j = \begin{bmatrix} L(\phi_1^{(j)}) \\ L(\phi_2^{(j)}) \\ \vdots \\ L(\phi_m^{(j)}) \end{bmatrix}. \tag{8}$$

Note that A is Symmetric and Positive-Definite (SPD), as the bilinear form B is coercive (6). This can be seen by observing that, for any nonzero vector $\mathbf{x}$, there exists a nonzero function $v \in V$ (specifically, $v = \sum_{i=1}^{N} \sum_{k=1}^{m} x_{(i-1)m+k}\, \phi_k^{(i)}$) such that $\mathbf{x}^T A \mathbf{x} = B(v, v) \overset{(6)}{\gtrsim} B_\Omega(v, v) + B_\sigma(v, v) > 0$.

2.2. Two-level preconditioning and deflation

In the previous section, we obtained a linear SIPG system of the form $A\mathbf{x} = \mathbf{b}$, where A is SPD. To solve this system, we consider the preconditioned CG method. In particular, we focus on a two-level preconditioner introduced by Dobrev et al. [15], and the corresponding BNN-deflation variant. In this section, both variants are discussed.

Preconditioning variant. The two-level preconditioner is defined in terms of a coarse correction operator $Q \approx A^{-1}$ that switches from the original DG test space to a coarse subspace, then performs a correction that is now simple in this coarse space, and finally switches back to the original DG test space. In this paper, we study the coarse subspace that is based on the piecewise constant basis functions.

More specifically, the coarse correction operator Q is defined as follows. Let R denote the so-called restriction operator such that $A_0 := RAR^T$ is the SIPG matrix for polynomial degree $p = 0$. In other words, R is a matrix of size $N \times Nm$ such that $R_{i,(i-1)m+1} = 1$ for all $i = 1, \dots, N$, and all other entries are zero. Using this notation, the coarse correction operator is defined as $Q := R^T A_0^{-1} R$.

The two-level preconditioner combines this operator with a nonsingular smoother $M_{\text{prec}}^{-1} \approx A^{-1}$ with the property

$$M_{\text{prec}} + M_{\text{prec}}^T - A \text{ is SPD.} \tag{9}$$

It can be seen that this requirement is satisfied for (block) Gauss–Seidel smoothing (in that case, $M_{\text{prec}} + M_{\text{prec}}^T - A$ is the (block) diagonal of A). Furthermore, it will be shown in Section 4.4 that this requirement is satisfied for block Jacobi smoothing. The result $\mathbf{y} = P_{\text{prec}}^{-1}\mathbf{r}$ of applying the two-level preconditioner to a vector $\mathbf{r}$ can now be computed in three steps:

$$\begin{aligned} \mathbf{y}^{(1)} &= M_{\text{prec}}^{-1}\mathbf{r} \quad \text{(pre-smoothing)}, \\ \mathbf{y}^{(2)} &= \mathbf{y}^{(1)} + Q(\mathbf{r} - A\mathbf{y}^{(1)}) \quad \text{(coarse correction)}, \\ \mathbf{y} &= \mathbf{y}^{(2)} + M_{\text{prec}}^{-T}(\mathbf{r} - A\mathbf{y}^{(2)}) \quad \text{(post-smoothing)}. \end{aligned} \tag{10}$$

The operator P_{prec}^{-1} is SPD assuming that (9) is satisfied [22, p. 66].

BNN deflation variant. Basically, the BNN deflation variant is obtained by turning (10) inside out, and using an SPD smoother $M_{\text{defl}}^{-1} \approx A^{-1}$ (such as block Jacobi). We do not impose a condition of the form (9) at this point. The result $\mathbf{y} = P_{\text{defl}}^{-1}\mathbf{r}$ of applying the BNN deflation technique to a vector $\mathbf{r}$ can now be computed as:

$$\begin{aligned} \mathbf{y}^{(1)} &:= Q\mathbf{r} \quad \text{(pre-coarse correction)}. \\ \mathbf{y}^{(2)} &:= \mathbf{y}^{(1)} + M_{\text{defl}}^{-1}(\mathbf{r} - A\mathbf{y}^{(1)}) \quad \text{(smoothing)}, \\ \mathbf{y} &:= \mathbf{y}^{(2)} + Q(\mathbf{r} - A\mathbf{y}^{(2)}) \quad \text{(post-coarse correction)}. \end{aligned} \tag{11}$$

The operator P_{defl}^{-1} is SPD for any SPD smoother M_{defl}^{-1}, as can be shown using the more abstract analysis in [22, p. 66].

Finally, we stress that the BNN deflation variant can be implemented more efficiently in a CG algorithm by using the so-called ADEF2 deflation variant. The latter is obtained by skipping the pre-coarse correction step in (11). ADEF2 yields the same iterates as BNN, as long as the starting vector $\mathbf{x}_0$ is pre-processed according to: $\mathbf{x}_0 \mapsto Q\mathbf{b} + (I - AQ)^T\mathbf{x}_0$ [17]. Furthermore, ADEF2 requires only 2 mat–vecs and 1 smoothing step per iteration, whereas the preconditioning variant (10) requires 3 mat–vecs and 2 smoothing steps. In Section 5, we will compare the overall numerical efficiency of these two methods. However, for the theoretical purposes in this paper, it is more convenient to study BNN than ADEF2.

Additional smoother requirements. To derive the theoretical results in this paper, we require additional assumptions on the smoothers. To specify these, for any SPD matrix M, let $\pi_M := R^T(RMR^T)^{-1}RM$ denote the projection onto the coarse space Range(R^T) that yields the best approximation in the M-norm (cf. [18]). Additionally, for any nonsingular matrix M such that $M + M^T - A$ is SPD, define the symmetrization $\widetilde{M} := M^T(M + M^T - A)^{-1}M$.

Using this notation, we can now specify the additional smoother requirements:

$$2M_{\text{defl}} - A \text{ is SPD}, \tag{12}$$

$$h^{2-d}\mathbf{v}^T\widetilde{M}_{\text{prec}}\mathbf{v} \lesssim \mathbf{v}^T\mathbf{v}, \quad \forall \mathbf{v} \in \text{Range}(I - \pi_I), \tag{13}$$

$$h^{2-d}\mathbf{v}^T M_{\text{defl}}\mathbf{v} \gtrsim \mathbf{v}^T\mathbf{v}, \quad \forall \mathbf{v} \in \text{Range}(I - \pi_I). \tag{14}$$

It will be shown in Section 4.4 that these requirements are satisfied for (damped) block Jacobi smoothing (a similar strategy can be used to show (13) for block Gauss–Seidel smoothing with blocks of size $m \times m$).

The conditions (9) and (12) imply that "the smoother iteration is a contraction in the A-norm" [18, p. 473]. The main idea behind the conditions (13) and (14) is that the smoother should scale with h^{2-d} in the same way that A does, and that $\widetilde{M}$ is an efficient preconditioner for A in the space orthogonal to the coarse space Range(R^T) [22, p. 78]. A slightly stronger version of (13) is also used in [15] to establish scalable convergence.

3. Auxiliary result: regularity on the diagonal of A

We want to show that the linear solvers discussed in the previous section are scalable (independent of h). To this end, we require an auxiliary result that roughly states that the diagonal blocks of A all behave in a similar manner in the space orthogonal to the coarse space. This section derives this property: after discussing two intermediate results in Sections 3.1 and 3.2 respectively, the desired outcome is given in Theorem 1 in Section 3.3.

3.1. Intermediate result I: using regularity

The first intermediate result is a rather abstract property of the mesh. To state this result, recall the mapping $F_i : E_i \to E_0$ (cf. Section 2). Because this mapping is invertible and affine by assumption (2), there exists an invertible matrix $G_i \in \mathbb{R}^{d\times d}$ and a vector $\mathbf{g}_i \in \mathbb{R}^d$ such that $F_i(\mathbf{x}) = G_i\mathbf{x} + \mathbf{g}_i$ for all $\mathbf{x} \in E_i$. Next, let $|G_i^{-1}|$ denote the determinant of G_i^{-1}, and define $Z_i := |G_i^{-1}|G_i^T G_i$. We now have the following result[2]:

Lemma 1. *The eigenvalues of the matrix Z_i above satisfies the following relation:*

$$1 \lesssim \lambda_{\min}(h^{2-d}Z_i) \le \lambda_{\max}(h^{2-d}Z_i) \lesssim 1, \quad \forall i = 1, \dots, N. \tag{15}$$

To show this result, we use that the mesh is regular (3) and quasi-uniform (4). Furthermore, we use the following relations [21, pp. 120–122][3]:

$$|G_i^{-1}| = \frac{\text{meas}(E_i)}{\text{meas}(E_0)}, \qquad \|G_i\|_2 \le \frac{h_0}{\rho_i}, \qquad \|G_i^{-1}\|_2 \le \frac{h_i}{\rho_0}. \tag{16}$$

We can now prove Lemma 1:

Proof of Lemma 1. Because $Z_i := |G_i^{-1}|G_i^T G_i$, and G_i is invertible, it follows from the relations in (16), and the fact that $\rho_i^d \lesssim \text{meas}(E_i) \lesssim h_i^d$ for all $i = 1, \dots, N$:

$$\lambda_{\max}(h^{2-d}Z_i) = h^{2-d}|G_i^{-1}|\,\|G_i\|_2^2 \lesssim \frac{\text{meas}(E_i)}{h^d}\left(\frac{h}{\rho_i}\right)^2 \lesssim \left(\frac{h_i}{h}\right)^d\left(\frac{h}{\rho_i}\right)^2,$$

$$\lambda_{\min}(h^{2-d}Z_i) = h^{2-d}|G_i^{-1}|\frac{1}{\|G_i^{-1}\|_2^2} \gtrsim \frac{\text{meas}(E_i)}{h^d}\left(\frac{h}{h_i}\right)^2 \gtrsim \left(\frac{\rho_i}{h}\right)^d\left(\frac{h}{h_i}\right)^2.$$

Here, we have also used that meas(E_0), h_0 and ρ_0 do not depend on h. Hence, the proof is completed if we can show that $1 \le \frac{h}{h_i} \le \frac{h}{\rho_i} \lesssim 1$ for all $i = 1, \dots, N$. The first two inequalities in this relation follow from the fact that $\rho_i \le h_i \le h$. The last inequality follows from (3) and (4). Hence, we obtain (15), which completes the proof. ■

3.2. Intermediate result II: the desired result in terms of local bilinear forms

The second intermediate result concerns the individual diagonal blocks of A in terms of local bilinear forms. To state this result, we require the following notation: let V_0 denote the space of all polynomials of degree p and lower defined on the

[2] Throughout this paper, $\lambda_{\min}$ and $\lambda_{\max}$ denote the smallest and the largest eigenvalue of a matrix with real eigenvalues respectively.

[3] Throughout this paper, meas(.) denotes the Lebesgue measure.

reference element E_0. Additionally, let Γ_i denote the set of all edges of E_i that are either in the interior or at the Dirichlet boundary. Furthermore, let Γ_0 denote the set of all edges of the reference element E_0. Next, define the following bilinear forms[4]:

$$\begin{aligned} B_\Omega^{(i)}(v,w) &= \int_{E_i} K\nabla(v\circ F_i)\cdot\nabla(w\circ F_i), \qquad & B_\Omega^{(0)}(v,w) &= \int_{E_0}\nabla v\cdot\nabla w, \\ B_\sigma^{(i)}(v,w) &= \sum_{e\in\Gamma_i}\int_e \frac{\sigma}{h_e}[v\circ F_i]\cdot[w\circ F_i], \qquad & B_\sigma^{(0)}(v,w) &= \sum_{e\in\Gamma_0}\int_e [v]\cdot[w], \end{aligned} \tag{17}$$

for all $v, w \in V_0$ and $i = 1, \dots, N$. Using this notation, we now have the following result:

Lemma 2. *The bilinear forms above satisfy the following relations:*

$$B_\Omega^{(0)}(w,w) \lesssim h^{2-d}B_\Omega^{(i)}(w,w) \lesssim B_\Omega^{(0)}(w,w), \quad \forall w\in V_0,\ \forall i=1,\dots,N. \tag{18}$$

$$0 \le h^{2-d}B_\sigma^{(i)}(w,w) \lesssim B_\sigma^{(0)}(w,w), \quad \forall w\in V_0,\ \forall i=1,\dots,N. \tag{19}$$

To show this result, we use the mesh properties (2)–(4), and the assumption that the diffusion coefficient and the penalty parameter are bounded above and below by positive constants (independent of h). We discuss the proof of both relations individually hereafter.

Proof of (18) in Lemma 2. Because the diffusion coefficient K is bounded below and above by positive constants (independent of h), we may write (both displayed relations below are for all $w \in V_0$ and for all $i = 1, \dots, N$):

$$\int_{E_i}\nabla(w\circ F_i)\cdot\nabla(w\circ F_i) \lesssim B_\Omega^{(i)}(w,w) \lesssim \int_{E_i}\nabla(w\circ F_i)\cdot\nabla(w\circ F_i). \tag{20}$$

To rewrite the integrals, we apply the chain rule (using that the Jacobian of F_i is equal to G_i), and a change of variables (from $\mathbf{x} \in E_i$ to $F_i(\mathbf{x}) \in E_0$, which introduces a factor $|G_i^{-1}|$). Multiplication with h^{2-d} then yields (recall $Z_i = |G_i^{-1}|G_i^TG_i$):

$$\int_{E_0}(\nabla w)^T(h^{2-d}Z_i)(\nabla w) \lesssim h^{2-d}B_\Omega^{(i)}(w,w) \lesssim \int_{E_0}(\nabla w)^T(h^{2-d}Z_i)(\nabla w).$$

Application of Lemma 1 completes the proof of (18). ■

Proof of (19) in Lemma 2. We restrict ourselves to the three-dimensional case. The proof for one- and two-dimensional problems follows similarly [19]. Because the penalty parameter σ is bounded below and above by positive constants (independent of h), and because $B_\sigma^{(i)}$ is Symmetric and Positive-SemiDefinite (SPSD), it follows using the notation in (17) that (all displayed relations below are for all $w \in V_0$ and for all $i = 1, \dots, N$):

$$0 \le h^{2-d}B_\sigma^{(i)}(w,w) \lesssim \sum_{e\in\Gamma_i}\int_e \frac{h^{2-d}}{h_e}[w\circ F_i]\cdot[w\circ F_i]. \tag{21}$$

For three-dimensional problems, the faces e are polygons and $h_e = \sqrt{\text{meas}(e)}$ (as defined in Section 2.1), i.e. the square root of the surface area of e. Because all faces are mutually affine-equivalent (2), for all e, there exists an invertible affine mapping $r_e : D \to e$ for some polygon $D \subset \mathbb{R}^2$ (independent of h). By definition of the surface integral over e, we may now rewrite (21) as (using $d = 3$):

$$0 \le h^{-1}B_\sigma^{(i)}(w,w) \lesssim \sum_{e\in\Gamma_i}\frac{1}{h\,h_e}\int_D [w\circ F_i\circ r_e(u,v)]\cdot[w\circ F_i\circ r_e(u,v)]\left|\frac{\partial r_e}{\partial u}\times\frac{\partial r_e}{\partial v}\right|\,\mathrm{d}u\,\mathrm{d}v.$$

Because $r_e(u, v)$ is affine, and $\int_e 1 = \int_D \left|\frac{\partial r}{\partial u}\times\frac{\partial r}{\partial v}\right|\,\mathrm{d}u\,\mathrm{d}v$, it follows that $\left|\frac{\partial r}{\partial u}\times\frac{\partial r}{\partial v}\right| = \frac{\text{meas}(e)}{\text{meas}(D)}$. Hence, using $\text{meas}(e) = h_e^2$ and observing $\frac{h_e}{h} \lesssim 1$ for all edges e:

$$0 \le h^{-1}B_\sigma^{(i)}(w,w) \lesssim \sum_{e\in\Gamma_i}\frac{1}{\text{meas}(D)}\int_D [w\circ F_i\circ r_e(u,v)]\cdot[w\circ F_i\circ r_e(u,v)]\,\mathrm{d}u\,\mathrm{d}v. \tag{22}$$

[4] Here, the trace operators are defined as before at the domain boundary, by extending the function to be zero outside E_0 and E_i. In particular, $[v] = v \cdot n_e$ for $v \in V_0$ and $e \in \Gamma_0$.

Next, consider a single $e \in \Gamma_i$: note that $F_i \circ r_e(D) = F_i(e) =: e_0 \subset \partial E_0$, and define the invertible affine mapping $r_{e_0} := F_i \circ r_e$. As above, we have that $\left| \frac{\partial r_{e_0}}{\partial u} \times \frac{\partial r_{e_0}}{\partial v} \right| = \frac{\text{meas}(e_0)}{\text{meas}(D)}$. By definition of the surface integral over e_0, we may now write (using that meas(e_0) does not depend on h):

$$\frac{1}{\text{meas}(D)} \int_D [w \circ F_i \circ r_e(u, v)] \cdot [w \circ F_i \circ r_e(u, v)] \, \mathrm{d}u \, \mathrm{d}v \lesssim \int_{e_0} [w] \cdot [w] \, .$$

Next, we apply this strategy for all $e \in \Gamma_i$, which yield different (disjunct) $e_0 \subset \partial E_0$, although the entire boundary of E_0 is not reached in the presence of Neumann boundary conditions. Substitution of the results into (22) then yields

$$0 \le h^{-1} B_\sigma^{(i)}(w, w) \lesssim \sum_{e \in \Gamma_i} \int_{F_i(e)} [w] \cdot [w] \le \sum_{\hat{e} \in \Gamma_0} \int_{\hat{e}} [w] \cdot [w] = B_\sigma^{(0)}(w, w).$$

This completes the proof for three-dimensional problems ($d = 3$). ■

3.3. Final auxiliary result: regularity on the diagonal of A

Using the intermediate results in the previous sections, we can now show the final result of this section, which roughly states that the diagonal blocks of A all behave in a similar manner in the space orthogonal to the coarse space. To state this result, we require the following notation: suppose that A_Ω results from the bilinear form B_Ω in the same way that A results from the bilinear form B: this is established by substituting A_Ω for A and B_Ω for B in (7) and (8). Similarly, suppose that the matrices A_σ and A_r result from the bilinear forms B_σ and B_r respectively. Altogether, we may write $A = A_\Omega + A_\sigma + A_r$. Finally, let D_σ be the result of extracting the diagonal blocks of size $m \times m$ from A_σ. Using this notation, we now have the following result:

Theorem 1. *The matrices A_Ω and D_σ above satisfy the following relations:*

$$\mathbf{v}^T \mathbf{v} \lesssim h^{2-d} \mathbf{v}^T A_\Omega \mathbf{v}, \quad \forall \mathbf{v} \in \text{Range}(I - \pi_I), \tag{23}$$

$$h^{2-d} \mathbf{v}^T A_\Omega \mathbf{v} \lesssim \mathbf{v}^T \mathbf{v}, \quad \forall \mathbf{v} \in \mathbb{R}^{mN}, \tag{24}$$

$$0 \le h^{2-d} \mathbf{v}^T D_\sigma \mathbf{v} \lesssim \mathbf{v}^T \mathbf{v}, \quad \forall \mathbf{v} \in \mathbb{R}^{mN}. \tag{25}$$

To show this result, we use the mesh properties (2)–(4), and the assumption that the diffusion coefficient and the penalty parameter are bounded above and below by positive constants (independent of h).

The main idea is to observe that A_Ω is an $N \times N$ block diagonal matrix with blocks of size $m \times m$, where the first row and column in every diagonal block is zero: this follows from the fact that $B_\Omega(\phi_k^i, \phi_\ell^{(j)}) = 0$ for $i \neq j$, and that the gradient of the piecewise constant basis function $\phi_1^{(j)}$ is (piecewise) zero. As a consequence, we can treat the diagonal blocks individually using Lemma 2, and then combine the results (a similar strategy is used for the block diagonal D_σ).

To show (23), we also use the nature of $\pi_I = R^T R$, which is the projection operator (as defined in Section 2.2, using $M = I$ and $RR^T = I$) onto the coarse space Range(R^T) that yields the best approximation in the (Euclidean) 2-norm. As a result, the space Range$(I - \pi_I)$ is orthogonal to Range(R^T), where the latter corresponds to the piecewise constant basis functions. In particular, any $\mathbf{v} \in \text{Range}(I - \pi_I) \subset \mathbb{R}^{Nm}$ is of the form:

$$\mathbf{v} = \begin{bmatrix} 0 & \mathbf{v}_1^T \mid 0 & \mathbf{v}_2^T \mid \ldots \mid 0 & \mathbf{v}_N^T \end{bmatrix}^T, \tag{26}$$

with $\mathbf{v}_1, \ldots \mathbf{v}_N \in \mathbb{R}^{m-1}$. Using these ideas, we can now show Theorem 1:

Proof of Theorem 1. Let $A_\Omega^{(i)}$ denote the result of deleting the first row and column in the ith diagonal block in A_Ω. In other words:

$$\left(A_\Omega^{(i)}\right)_{\ell-1,k-1} = B_\Omega^{(i)}(\phi_k^{(0)}, \phi_\ell^{(0)}), \tag{27}$$

for all $k, \ell = 2, \ldots, m$. Next, observe that $B_\Omega^{(0)}$ is independent of h and symmetric. Furthermore, for all higher-order polynomials $v \in \text{span}\{\phi_2^{(0)}, \ldots, \phi_m^{(0)}\} \setminus \{0\}$, the gradient of v is nonzero, which implies that $B_\Omega^{(0)}(v, v) > 0$. In other words, $B_\Omega^{(0)}$ is even positive-definite for the subspace $\text{span}\{\phi_2^{(0)}, \ldots, \phi_m^{(0)}\} \setminus \{0\}$. As a consequence, combining Lemma 2 with (27), we obtain a result similar to (23), but then for the individual diagonal blocks:

$$\mathbf{w}^T \mathbf{w} \lesssim h^{2-d} \mathbf{w}^T A_\Omega^{(i)} \mathbf{w}, \quad \forall \mathbf{w} \in \mathbb{R}^{m-1}, \ \forall i = 1, \ldots, N.$$

Using the notation in (26), these relations hold in particular for $\mathbf{w} = \mathbf{v}_i$, for all $i = 1, \ldots, N$. Summing over all i and using the structure of $\mathbf{v}$ in (26) once more then yields (23). The relations (24) and (25) follow in a similar manner from Lemma 2 (without deleting the first row and column in each diagonal block). This completes the proof of Theorem 1. ■

4. Main result: scalability of both two-level methods

Using the auxiliary property obtained in the previous section, we can now show the main result of this paper: both two-level methods yield scalable convergence of the preconditioned CG method (independent of the mesh element diameter). After discussing two intermediate results in Sections 4.1 and 4.2 respectively, the scalable convergence is finally established in Theorem 2 in Section 4.3. This result is particularly valid for block Jacobi smoothing, as is demonstrated in Section 4.4.

4.1. Intermediate result I: using the error iteration matrix

In this section, we consider the condition number of the preconditioned system for arbitrary SPD matrices A and a certain class of SPD preconditioners P^{-1}. Specifically, each preconditioner P^{-1} in this class is such that, for some SPD matrix M, the so-called error iteration matrix $I - P^{-1}A$ has the same eigenvalues as (recall the notation in Section 2.2):

$$T_M := (I - \pi_A)(I - M^{-1}A)(I - \pi_A).$$

In Section 4.2 hereafter, we will see that both two-level methods are in this class for certain specific choices of M, i.e. both P^{-1}_{prec} and P^{-1}_{defl} (as defined in Section 2.2) are in the larger class of preconditioners P^{-1} considered here. Defining

$$K_M := \sup_{\mathbf{v}\neq 0} \frac{\|(I - \pi_M)\mathbf{v}\|_M^2}{\|\mathbf{v}\|_A^2}, \tag{28}$$

we now have the following result:

Lemma 3. *The condition number (in the 2-norm) of the preconditioned system $P^{-1}A$ above can be bounded as follows:*

$$\kappa_2(P^{-1}A) \leq \lambda_{\max}(M^{-1}A)K_M. \tag{29}$$

Additionally, if[5] $M - A \geq 0$, *then,*

$$\kappa_2(P^{-1}A) = K_M. \tag{30}$$

To show this, we use that T_M has real eigenvalues (as $A^{\frac{1}{2}}T_M A^{-\frac{1}{2}}$ is symmetric), and that [23, Theorem 2.1 and Corollary 2.1]:

$$T_M \text{ has } m \text{ times eigenvalue } 0, \tag{31}$$

$$\lambda_{\min}(T_M) \geq 1 - \lambda_{\max}(M^{-1}A), \tag{32}$$

$$\lambda_{\max}(T_M) = 1 - \frac{1}{\lambda_{\max}(A^{-1}M(I - \pi_M))}. \tag{33}$$

Proof of Lemma 3. First, note that $P^{-1}A$ and $P^{-\frac{1}{2}}AP^{-\frac{1}{2}}$ have the same positive eigenvalues and singular values. Hence, we may express the condition number as:

$$\begin{aligned}
\kappa_2(P^{-1}A) &= \frac{\lambda_{\max}(P^{-1}A)}{\lambda_{\min}(P^{-1}A)} = \frac{1 - \lambda_{\min}(T_M)}{1 - \lambda_{\max}(T_M)} \\
&\overset{(33)}{=} \frac{1 - \lambda_{\min}(T_M)}{1 - \left(1 - \frac{1}{\lambda_{\max}(A^{-1}M(I-\pi_M))}\right)} \\
&= \big(1 - \lambda_{\min}(T_M)\big)\lambda_{\max}\big(A^{-1}M(I - \pi_M)\big).
\end{aligned} \tag{34}$$

Because $I - \pi_M = (I - \pi_M)^2$ is a projection and $M(I - \pi_M)$ is symmetric, it follows that:

$$\begin{aligned}
\lambda_{\max}\big(A^{-1}M(I - \pi_M)\big) &= \lambda_{\max}\big(A^{-1}M(I - \pi_M)^2\big) \\
&= \lambda_{\max}\big(A^{-1}(I - \pi_M)^T M(I - \pi_M)\big) \\
&= \lambda_{\max}\big(A^{-\frac{1}{2}}(I - \pi_M)^T M(I - \pi_M)A^{-\frac{1}{2}}\big) \\
&= \sup_{\mathbf{v}\neq 0} \frac{\|(I - \pi_M)\mathbf{v}\|_M^2}{\|\mathbf{v}\|_A^2} \overset{(28)}{=} K_M.
\end{aligned}$$

[5] Throughout this paper, for symmetrical matrices $M_1, M_2 \in \mathbb{R}^{n\times n}$, we write $M_1 \leq M_2$ to indicate that $\mathbf{v}^T M_1 \mathbf{v} \leq \mathbf{v}^T M_2 \mathbf{v}$ for all vectors $\mathbf{v} \in \mathbb{R}^n$; the notation $\geq$, $<$, and $>$ is used similarly.

Substitution into (34) yields:

$$\kappa_2(P^{-1}A) = \big(1 - \lambda_{\min}(T_M)\big)K_M. \tag{35}$$

Application of (32) now completes the proof of (29). To show (30), assume that $M - A \geq 0$, which implies that $I - A^{\frac{1}{2}}M^{-1}A^{\frac{1}{2}} \geq 0$:

$$\begin{aligned}
&M \geq A\\
&M^{-1} \overset{[24,\text{ pp. }398,\,471]}{\leq} A^{-1}\\
&A^{\frac{1}{2}}M^{-1}A^{\frac{1}{2}} \leq I\\
&I - A^{\frac{1}{2}}M^{-1}A^{\frac{1}{2}} \geq 0.
\end{aligned}$$

Hence, defining the *symmetric* projection $\bar{\pi}_A := A^{\frac{1}{2}}\pi_A A^{-\frac{1}{2}}$, it follows that the eigenvalues of T_M are non-negative:

$$A^{\frac{1}{2}}T_M A^{-\frac{1}{2}} = (I - \bar{\pi}_A)\left(I - A^{\frac{1}{2}}M^{-1}A^{\frac{1}{2}}\right)(I - \bar{\pi}_A) \geq 0.$$

As a result, (31) implies that $\lambda_{\min}(T_M) = 0$. Substitution into (35) yields (30), which then completes the proof. ■

4.2. Intermediate result II: implications for the two-level methods

Next, we apply the result in the previous section to analyze the condition number of the preconditioned system for both the two-level preconditioner P_{prec}^{-1} and the corresponding BNN deflation variant P_{defl}^{-1} (as specified in Section 2.2). For the two-level preconditioner, it is well-known that

$$\kappa_2(P_{\text{prec}}^{-1}A) \leq K_{\widetilde{M}_{\text{prec}}}. \tag{36}$$

This follows as a special case from [18] (also cf. [22, pp. 70–73]), and relies on assumption (9). Below, we obtain similar bounds for the deflation variant, assuming (12). Furthermore we observe that the theory in [23] implies (via Lemma 3) that (36) remains true if we replace the inequality by equality (see also the error propagation bounds for successive subspace correction algorithms shown by Zikatanov [25]). This latter derivation is also included to indicate the similarities of the two methods. Altogether, we have the following result, which applies for any SPD matrix A:

Lemma 4. *Suppose that A is SPD and let P_{prec}^{-1} and P_{defl}^{-1} be the two-level operators specified in Section 2.2. Then, assuming (9), the condition number (in the 2-norm) of the preconditioned system $P_{\text{prec}}^{-1}A$ can be expressed as follows:*

$$\kappa_2(P_{\text{prec}}^{-1}A) = K_{\widetilde{M}_{\text{prec}}}. \tag{37}$$

Additionally, assuming (12), we have for deflation:

$$\kappa_2(P_{\text{defl}}^{-1}A) \leq \lambda_{\max}(M_{\text{defl}}^{-1}A)K_{M_{\text{defl}}} < 2K_{M_{\text{defl}}}, \tag{38}$$

and, under the stronger assumption $M_{\text{defl}} - A \geq 0$:

$$\kappa_2(P_{\text{defl}}^{-1}A) = K_{M_{\text{defl}}}. \tag{39}$$

To show this result, we apply Lemma 3, using (σ denotes the spectrum):

$$\sigma\left(I - P_{\text{prec}}^{-1}A\right) = \sigma\left(T_{\widetilde{M}_{\text{prec}}}\right), \tag{40}$$

$$\sigma\left(I - P_{\text{defl}}^{-1}A\right) = \sigma\left(T_{M_{\text{defl}}}\right). \tag{41}$$

These relations follow similar to [26, p. 1730]. Finally, we use that, for any nonsingular M [22, Proposition 3.8]:

$$M + M^T - A > 0 \Rightarrow \widetilde{M} - A \geq 0. \tag{42}$$

Proof of Lemma 4. Combining (9) and (42) gives $\widetilde{M}_{\text{prec}} - A \geq 0$. Using this with (40) in Lemma 3 yields (37). Similarly, (39) follows from Lemma 3 using (41) and the assumption $M_{\text{defl}} - A \geq 0$. To show (38), note that the first inequality results from Lemma 3 and (41), while the second inequality follows from observing that (12) implies that $\lambda_{\max}(M_{\text{defl}}^{-1}A) < 2$. This completes the proof. ■

Remark 1 (*Comparing Preconditioning and Deflation*)**.** If $M_{\text{prec}} = M_{\text{defl}} =: M$ SPD with $M - A \geq 0$, then, using Lemma 4, it can be shown [27] that both methods are related in the following sense:

$$\frac{1}{2}\kappa_2(P_{\text{defl}}^{-1}A) \leq \kappa_2(P_{\text{prec}}^{-1}A) \leq \kappa_2(P_{\text{defl}}^{-1}A). \tag{43}$$

At the same time, the error of the jth CG iterate $\mathbf{u}_j$ in the A-norm for both methods can be bounded as follows [28]:

$$\|\mathbf{u} - \mathbf{u}_j\|_A \leq 2\|\mathbf{u} - \mathbf{u}_0\|_A \left(\frac{\kappa_2(P^{-1}A) - 1}{\kappa_2(P^{-1}A) + 1}\right)^{j+1}, \quad P^{-1} = P_{\text{prec}}^{-1}, P_{\text{defl}}^{-1}.$$

Hence, if $M - A \geq 0$, so that (43) implies that $\kappa_2(P_{\text{prec}}^{-1}A) \leq \kappa_2(P_{\text{defl}}^{-1}A)$, then the error $\|\mathbf{u} - \mathbf{u}_j\|_A$ can be bounded by a smaller value for the preconditioner than for the deflation variant. It should be stressed that this relates upper bounds for the errors of the two methods, not the errors themselves. In general, we may have $2M - A > 0$ rather than the stronger assumption $M - A \geq 0$, in which case the analysis above (and the corresponding relation between the upper bounds for the errors) no longer applies. Altogether, (43) provides insight in the way both two-level methods are related, but does not imply that preconditioning is always better. □

4.3. Final main result: scalability of both two-level methods

Using the intermediate results in the previous sections, we can now show the main result of this paper: both two-level methods yield scalable convergence of the CG method, that is, independent of the mesh element diameter (the influence of the polynomial degree and the diffusion coefficient is not considered here). This result has been shown by Dobrev et al. [15] for the preconditioning variant for $p = 1$. In this section, we use a similar strategy to extend these results for $p \geq 1$ and for the deflation variant (in the next section, we show that the theorem below is particularly true for block Jacobi smoothing).

Theorem 2 (*Main Result*)**.** *Let A be the discretization matrix resulting from an SIPG scheme with $p \geq 1$ (cf. Section 2.1). Suppose that the conditions (2)–(4) and (6) are satisfied, and that the diffusion coefficient and the penalty parameter are bounded above and below by positive constants (independent of h). Let P_{prec}^{-1} and P_{defl}^{-1} denote the two-level preconditioner and BNN deflation variant respectively (cf. Section 2.2). Suppose that the smoother M_{prec} is nonsingular and M_{defl} is SPD, and that the conditions (9) and (12)–(14) are satisfied. Then, both two-level methods yield scalable CG convergence in the sense that the condition number κ_2 (in the 2-norm) of the preconditioned system can be bounded independently of the maximum mesh element diameter h:*

$$\kappa_2(P_{\text{prec}}^{-1}A) \lesssim 1, \qquad \kappa_2(P_{\text{defl}}^{-1}A) \lesssim 1. \tag{44}$$

To show Theorem 2, the main idea is to consider Lemma 4:

$$\kappa_2(P_{\text{prec}}^{-1}A) \leq K_{\widetilde{M}_{\text{prec}}}, \qquad \kappa_2(P_{\text{defl}}^{-1}A) \leq 2K_{M_{\text{defl}}}. \tag{45}$$

The proof is then completed by showing that $K_{\widetilde{M}_{\text{prec}}}, K_{M_{\text{defl}}} \lesssim 1$, for any smoother that satisfies the criteria above. This is established using coercivity (6) and the auxiliary result Theorem 1. Altogether, Theorem 2 can now be shown as follows.

Proof of Theorem 2. First, we will show that $K_{\widetilde{M}_{\text{prec}}} \lesssim 1$. For ease of notation, we will write $\widetilde{M}$ for $\widetilde{M}_{\text{prec}}$. The main idea is to show that $\|(I - \pi_{\widetilde{M}})\mathbf{v}\|_{\widetilde{M}} \lesssim \|\mathbf{v}\|_A$ for all $\mathbf{v}$: because $\pi_{\widetilde{M}}$ is a projection onto the coarse space Range(R^T) that yields the best approximation in the $\widetilde{M}$-norm, we can replace $\pi_{\widetilde{M}}$ by the suboptimal projection π_I, and then combine the properties established so far:

$$\begin{aligned}
\|(I - \pi_{\widetilde{M}})\mathbf{v}\|_{\widetilde{M}}^2 &\leq \|(I - \pi_I)\mathbf{v}\|_{\widetilde{M}}^2 \overset{(13)}{\lesssim} h^{d-2}\|(I - \pi_I)\mathbf{v}\|_2^2 \\
&\overset{(23)}{\lesssim} \|(I - \pi_I)\mathbf{v}\|_{A_\Omega}^2 \overset{\text{Section 3.3}}{\lesssim} \|\mathbf{v}\|_{A_\Omega}^2 \\
&\overset{B_\sigma\ \text{SPSD}}{\lesssim} \|\mathbf{v}\|_{A_\Omega + A_\sigma}^2 \overset{(6)}{\lesssim} \|\mathbf{v}\|_A^2, \quad \forall \mathbf{v} \in \mathbb{R}^{Nm}.
\end{aligned}$$

Substitution of this relation into the definition of $K_{\widetilde{M}}$ (introduced at the beginning of Section 4.1) yields $K_{\widetilde{M}} \lesssim 1$. A similar strategy, using (14) instead of (13), yields $K_{M_{\text{defl}}} \lesssim 1$. Substitution of $K_{\widetilde{M}_{\text{prec}}}, K_{M_{\text{defl}}} \lesssim 1$ into (45) now yields (44), which completes the proof of Theorem 2. ■

4.4. Special case: block Jacobi smoothing

This section demonstrates that Theorem 2 is valid for (damped) block Jacobi smoothing. To specify this result, suppose that M_{BJ} is the block Jacobi smoother with blocks of size $m \times m$. Next, consider the specific choices $M_{\text{prec}}, M_{\text{defl}} = \omega^{-1}M_{\text{BJ}}$

with damping parameter $\omega > 0$ (independent of h). We assume that $\omega \leq 1$, with $\omega < 1$ strictly for the preconditioning variant. Additionally, we assume that there exists a permutation matrix P such that A can be permuted as:

$$PAP^T = \Delta - L - L^T, \tag{46}$$

with

$$\Delta = \begin{pmatrix} \Delta_1 & & & \\ & \Delta_2 & & \\ & & \ddots & \\ & & & \Delta_q \end{pmatrix}, \qquad -L = \begin{pmatrix} 0 & & & \\ L_1 & 0 & & \\ & \ddots & 0 & \\ & & L_{q-1} & 0 \end{pmatrix},$$

for some block-diagonal matrices $\Delta_1, \ldots, \Delta_q$ with blocks of size $m \times m$, matrices $L_1, \ldots, L_{q-1}$, and integer $q \leq N$. Note that this assumption implies that the matrix A has property A^π in the sense of [29, Definition 6.7]. Moreover, we remark that (46) is satisfied if the mesh can be colored by two colors[6] (in that case, we can choose $q = 2$, and Δ_1 and Δ_2 each correspond to one of the two colors). In particular, structured rectangular meshes can be colored by two colors and thus satisfy (46).

Altogether, assuming[7] (46), we can now show that all smoother requirements for Theorem 2 are satisfied for (damped) block Jacobi smoothing:

Corollary 1. *Suppose that the mesh can be colored by two colors. Then, Theorem 2 applies for the damped block Jacobi smoothers M_{prec} and M_{defl} above, i.e. both two-level methods yield scalable CG convergence in the sense that the condition number κ_2 (in the 2-norm) of the preconditioned system can be bounded independently of the maximum mesh element diameter h:*

$$\kappa_2(P_{\text{prec}}^{-1}A) \lesssim 1, \qquad \kappa_2(P_{\text{defl}}^{-1}A) \lesssim 1.$$

This result follows immediately from Theorem 2 once we have verified that the conditions (9), (12), (13), and (14) are satisfied for the (SPD) damped block Jacobi smoothers under consideration. In other words, writing $M := \omega^{-1}M_{\text{BJ}}$, we need to show:

$$2M - A > 0, \tag{47}$$

$$h^{2-d}\mathbf{v}^T M\mathbf{v} \lesssim \mathbf{v}^T\mathbf{v}, \quad \forall \mathbf{v} \in \text{Range}(I - \pi_I), \tag{48}$$

$$h^{2-d}\mathbf{v}^T \widetilde{M}\mathbf{v} \lesssim \mathbf{v}^T\mathbf{v}, \quad \forall \mathbf{v} \in \text{Range}(I - \pi_I), \tag{49}$$

for all $\omega \leq 1$, with $\omega < 1$ strictly for (49). We treat each relation separately.

To show (47), we use that (ρ denotes the spectral radius):

$$\rho(B) < 1, \quad B := \Delta^{-1}(L + L^T), \tag{50}$$

which follows from [29, Theorem 6.38] using (46) and the fact that A and M are SPD.

Proof of (47). Without loss of generality, assume that $\omega = 1$. Next, observe that $PMP^T = \Delta$. As a result, it can be shown that $P(2M - A)P^T = \Delta(I + B)$. Hence, $\lambda_{\min}(2M - A) = \lambda_{\min}(I + \Delta^{\frac{1}{2}}B\Delta^{-\frac{1}{2}})$. Since (50) implies that $\rho(B) = \rho(\Delta^{\frac{1}{2}}B\Delta^{-\frac{1}{2}}) < 1$, it follows that $2M - A > 0$. This completes the proof. ■

To show (48), the main idea is to use Theorem 1 and the following property (cf. [15, p. 760] and [30, p. 4]):

$$0 < B(v, v) \lesssim B_\Omega(v, v) + B_\sigma(v, v), \quad \forall \mathbf{v} \in \mathbb{R}^{Nm}. \tag{51}$$

Proof of (48). Without loss of generality, we may assume that $\omega = 1$. Next, recall the notation introduced in the beginning of Section 3.3. Additionally, similar to D_σ, let D_r be the result of extracting the diagonal blocks of size $m \times m$ from A_r. Using this notation, and the fact that A_Ω is a block diagonal matrix with blocks of size $m \times m$, we may write $M = A_\Omega + D_\sigma + D_r$. Next, consider (51) in matrix form, and note that the relation is also true when considering the diagonal blocks only:

$$\mathbf{v}^T M\mathbf{v} \lesssim \mathbf{v}(A_\Omega + D_\sigma)\mathbf{v}, \quad \forall \mathbf{v} \in \mathbb{R}^{Nm}. \tag{52}$$

Application of Theorem 1 now yields (48), which completes the proof. ■

To show (49), we combine the previous results (47) and (48):

Proof of (49). Using (47), and the fact that $\omega < 1$ strictly, it can be shown [19] that $\widetilde{M} \leq \frac{1}{2(1-\omega)}M$. Combining this relation with (48) yields (49), which then completes the proof. ■

[6] That is, the mesh can be represented by a graph whose vertices can be colored such that connected vertices do not have the same color.

[7] Alternatively, we could assume that the damping parameter ω is sufficiently small. This option is not considered further in this paper.

4.5. Influence of damping and the penalty parameter for block Jacobi smoothing

This section studies the influence of damping and the penalty parameter on the constants in Corollary 1, where the same damped block Jacobi smoother is used for both two-level methods.

Regarding damping, it can be shown (the proof is given at the end of this section):

$$\kappa_2(P_{\text{defl}}^{-1}A) \leq \frac{2}{\omega} K_{M_{\text{BJ}}}, \qquad \kappa_2(P_{\text{prec}}^{-1}A) < \frac{1}{2\omega(1-\omega)} K_{M_{\text{BJ}}}. \tag{53}$$

Although these upper bounds may not be optimal, the fact that the upper bound for the preconditioner blows up as ω tends to 1 is in line with our numerical observation in Section 5 later on that the preconditioning variant performs better for ω safely away from 1.

To study the influence of the penalty parameter, let $\sigma_{\max}^{(i)}$ denote the largest value that the penalty parameter σ attains at the edges of mesh element E_i, and let $K_{\min}^{(i)}$ denote the smallest value that the diffusion coefficient K attains within E_i (for all $i = 1, \ldots, N$). We can now bound $K_{M_{\text{BJ}}}$ in terms of the local ratio between the penalty parameter and the diffusion coefficient, assuming[8] $A_\sigma + A_r \geq 0$ (the proof is given at the end of this section):

$$K_{M_{\text{BJ}}} \leq C_1 \max_{i=1,\ldots,N} \frac{\sigma_{\max}^{(i)}}{K_{\min}^{(i)}} + C_2, \tag{54}$$

for some positive constants C_1 and C_2 that are independent of the mesh element diameter h *and* the penalty parameter σ (but possibly dependent on the diffusion coefficient K). The result of substituting (54) into (53) is in line with our numerical observation in Section 5 later on that the penalty parameter can best be chosen dependent on local values of the diffusion coefficient.

We end this section with the proofs of (53) and (54). To this end, we use that, for any SPD matrix M (Not05, Eq. (45)):

$$\frac{1}{2}M \leq \widetilde{M} \leq \frac{1}{2 - \lambda_{\max}(M^{-1}A)} M. \tag{55}$$

Furthermore, for any SPD matrices M, N and scalar $\alpha > 0$ [27]:

$$M \leq \alpha N \Rightarrow K_M \leq \alpha K_N. \tag{56}$$

Proof of (53). The first inequality follows from (45), (56) and the fact that $M_{\text{defl}} = \omega^{-1} M_{\text{BJ}}$. To show the second inequality, we use that (47) implies that $\lambda_{\max}(M_{\text{prec}}^{-1}A) < 2\omega$:

$$\begin{aligned}
\kappa_2(P_{\text{prec}}^{-1}A) &\overset{(45)}{=} K_{\widetilde{M}_{\text{prec}}} \\
&\overset{(56),(55)}{\leq} \frac{1}{2 - \lambda_{\max}(M_{\text{prec}}^{-1}A)} K_{M_{\text{prec}}} \\
&\overset{\lambda_{\max}(M_{\text{prec}}^{-1}A) < 2\omega}{<} \frac{1}{2(1-\omega)} K_{M_{\text{prec}}} \\
&\overset{(56), M_{\text{prec}} = \omega^{-1} M_{\text{BJ}}}{\leq} \frac{1}{2\omega(1-\omega)} K_{M_{\text{BJ}}}.
\end{aligned}$$

This completes the proof of (53). ■

Proof of (54). By definition,

$$K_{M_{\text{BJ}}} \overset{(28)}{=} \sup_{\mathbf{v} \neq 0} \frac{\|(I - \pi_{M_{\text{BJ}}})\mathbf{v}\|_{M_{\text{BJ}}}^2}{\|\mathbf{v}\|_A^2}.$$

As in the proof of Theorem 2, we may replace $\pi_{M_{\text{BJ}}}$ by the suboptimal projection π_I:

$$K_{M_{\text{BJ}}} \leq \sup_{\mathbf{v} \neq 0} \frac{\|(I - \pi_I)\mathbf{v}\|_{M_{\text{BJ}}}^2}{\|\mathbf{v}\|_A^2}.$$

[8] This condition seems closely related to coercivity (6). How either can be guaranteed in practice (for problems with strong contrasts in the coefficients) is left for future research.

Using the notation in Section 3.3 and (52), we can rewrite this as:

$$K_{M_{\mathrm{BJ}}} \leq \sup_{\mathbf{v}\neq 0} \frac{\|(I-\pi_I)\mathbf{v}\|^2_{A_\Omega + D_\sigma + D_r}}{\|\mathbf{v}\|^2_{A_\Omega + A_\sigma + A_r}}.$$

Next, we use the assumption $A_\sigma + A_r \geq 0$:

$$\begin{aligned} K_{M_{\mathrm{BJ}}} &\overset{A_\sigma + A_r \geq 0}{\leq} \sup_{\mathbf{v}\neq 0} \frac{\|(I-\pi_I)\mathbf{v}\|^2_{A_\Omega + D_\sigma + D_r}}{\|\mathbf{v}\|^2_{A_\Omega}} \\ &\overset{\text{Section 3.3}}{=} \sup_{\mathbf{v}\neq 0} \frac{\|(I-\pi_I)\mathbf{v}\|^2_{A_\Omega + D_\sigma + D_r}}{\|(I-\pi_I)\mathbf{v}\|^2_{A_\Omega}} \\ &= 1 + \sup_{\mathbf{v}\in \mathrm{Range}(I-\pi_I)} \frac{\|\mathbf{v}\|^2_{D_\sigma + D_r}}{\|\mathbf{v}\|^2_{A_\Omega}}. \end{aligned}$$

Next, consider the notation for $A_\Omega^{(i)}$ in Section 3.3 and, similarly, let $D_\sigma^{(i)}$ and $D_r^{(i)}$ denote the result of removing the first row and column from diagonal block i in D_σ and D_r respectively. Then, we may write, using the notation for the components of $\mathbf{v}$ in Section 3.3:

$$K_{M_{\mathrm{BJ}}} \leq 1 + \sup_{\mathbf{v}\in \mathrm{Range}(I-\pi_I)} \frac{\sum_{i=1}^{N} \mathbf{v}_i^T (D_\sigma^{(i)} + D_r^{(i)}) \mathbf{v}_i^T}{\sum_{i=1}^{N} \mathbf{v}_i^T A_\Omega^{(i)} \mathbf{v}_i}.$$

At the same time, it can be shown (similar to Section 3) that there exist positive constants C_Ω, C_σ, C_r independent of h and σ, with C_Ω, C_σ also independent of K, such that, for all $\mathbf{w} \in \mathbb{R}^{m-1}$:

$$\begin{aligned} &h^{2-d}\mathbf{w}^T A_\Omega^{(i)} \mathbf{w} \geq C_\Omega K_{\mathrm{min}}^{(i)} \mathbf{w}^T \mathbf{w}, \\ &h^{2-d}\mathbf{w}^T D_\sigma^{(i)} \mathbf{w} \leq C_\sigma \sigma_{\mathrm{max}}^{(i)} \mathbf{w}^T \mathbf{w}, \\ &h^{2-d}\mathbf{w}^T D_r^{(i)} \mathbf{w} \leq C_r \mathbf{w}^T \mathbf{w}. \end{aligned}$$

Combining these relations gives:

$$\mathbf{v}_i^T (D_\sigma^{(i)} + D_r^{(i)}) \mathbf{v}_i^T \leq \frac{C_\sigma \sigma_{\mathrm{max}}^{(i)} + C_r}{C_\Omega K_{\mathrm{min}}^{(i)}} \mathbf{v}_i^T A_\Omega^{(i)} \mathbf{v}_i.$$

Using the latter relation, we may now write:

$$\begin{aligned} K_{M_{\mathrm{BJ}}} &\leq 1 + \sup_{\mathbf{v}\in \mathrm{Range}(I-\pi_I)} \frac{\sum_{i=1}^{N} \left(\frac{C_\sigma \sigma_{\mathrm{max}}^{(i)} + C_r}{C_\Omega K_{\mathrm{min}}^{(i)}} \mathbf{v}_i^T A_\Omega^{(i)} \mathbf{v}_i \right)}{\sum_{i=1}^{N} \mathbf{v}_i^T A_\Omega^{(i)} \mathbf{v}_i} \\ &\leq 1 + \max_{i=1,\ldots,N} \left\{ \frac{C_\sigma \sigma_{\mathrm{max}}^{(i)}}{C_\Omega K_{\mathrm{min}}^{(i)}} \right\} + \max_{i=1,\ldots,N} \left\{ \frac{C_r}{C_\Omega K_{\mathrm{min}}^{(i)}} \right\}. \end{aligned}$$

This can be rewritten as (54), which then completes the proof. ■

5. Numerical results

The previous section demonstrated theoretically that both two-level methods yield mesh-independent convergence of the CG method. In this section, we extend the numerical support in [16] for this result by studying test problems with strong variations in the coefficients.

Test cases. We consider several diffusion problems of the form (1) on the domain $[0, 1]^2$, as illustrated in Fig. 1 (if we subdivide the domain into 10 × 10 equally sized squares, the diffusion coefficient is constant within each square). The first problem is a bubbly flow problem with large jumps in the coefficients, inspired by [26]. We also consider this problem with reversed coefficients, i.e. where the diffusion coefficient $K = 1$ inside the 'bubbles' and K is a small constant outside the bubbles (we consider different values $K = 10^{-1}, 10^{-3}, 10^{-5}$). The final problem is challenging due to homogeneous Neumann boundary conditions (indicated by the black lines in Fig. 1). For all problems, the Dirichlet boundary conditions and the source term

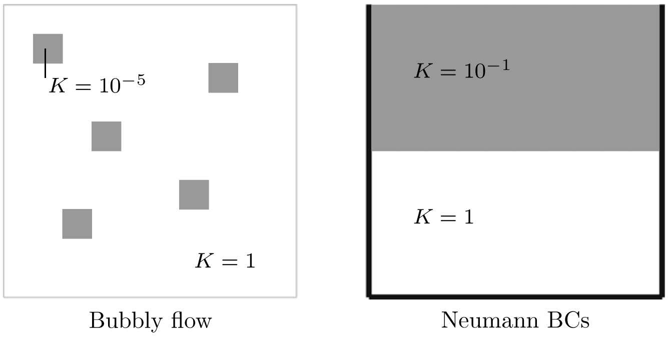

Fig. 1. Illustration of the test cases.

Table 1
Bubbly flow with reversed coefficients ($K_{\max} = 1, K_{\min} = 10^{-5}$) and penalty $\sigma = 20K$: SIPG convergence in the L^2-norm.

Mesh	$p = 2$		$p = 3$	
	Error	Order	Error	Order
$N = 20^2$	6.25e−02	–	8.36e−03	–
$N = 40^2$	6.90e−03	3.18	4.69e−04	4.16
$N = 80^2$	6.53e−04	3.40	2.73e−05	4.10
$N = 160^2$	6.86e−05	3.25	1.66e−06	4.04

f are chosen such that the exact solution reads $u(x, y) = \cos(10\pi x)\cos(10\pi y)$. We stress that this choice does not impact the matrix or the performance of the linear solver, as we use random start vectors (see below).

Experimental setup. All model problems are discretized by means of the SIPG method as discussed in Section 2.1. We use a uniform Cartesian mesh with $N = n \times n$ elements with $n = 40, 80, 160, 320$, and monomial basis functions with polynomial degree $p = 2, 3$ (results for $p = 1$ are similar though). As a result, the largest problems have over 10^6 degrees of freedom.

The penalty parameter[9] is chosen diffusion-dependent, $\sigma = 20K$, as motivated by [16,27]: such a strategy was first proposed in [32]. A common alternative approach is to choose the penalty parameter constant, e.g. $\sigma = 20$ [15,33]. Scalable convergence is obtained in both cases [19]. However, a diffusion-dependent penalty parameter yields significantly faster results for problems with strong variations in the coefficients, as well as faster SIPG convergence. This was demonstrated numerically in [16,27], and computational evidence for the improved SIPG convergence is also provided below. For this reason, we consider $\sigma = 20K$ here. Where K is discontinuous, we use the largest of the two trace values (in our experience, for the applications under consideration, this strategy is more suitable than the alternative of using harmonic means [32,34], while the SIPG convergence remains practically the same [27]).

The resulting linear systems are solved by means of the preconditioned CG method, combined with either the two-level preconditioner or the corresponding ADEF2 deflation variant, as discussed in Section 2.2. Furthermore, we use (damped) block Jacobi smoothing (cf. Section 4.4). For the damping parameter we consider $\omega = 1$ for the deflation variant, and both $\omega = 1$ and $\omega = 0.7$ for the preconditioner, as motivated by [16]. Diagonal scaling is applied as a pre-processing step in all cases, and the same random start vector $\mathbf{x}_0$ is used for all problems of the same size. For the stopping criterion we use: $\frac{\|r_k\|_2}{\|b\|_2} \leq 10^{-6}$, where r_k is the residual after the kth iteration. Coarse systems, involving the SIPG matrix A_0 with polynomial degree $p = 0$, are solved directly. However, a more efficient strategy has been studied in [16]. In any case, the coarse matrix A_0 is quite similar to a central difference matrix, for which very efficient solvers are readily available.

Results. Before showing the performance of the two-level methods, we first illustrate that our choice for the penalty parameter ($\sigma = 20K$) benefits the SIPG convergence. This can be seen by comparing Table 1 (for $\sigma = 20K$) and Table 2 (for a constant $\sigma = 20$) for the bubbly flow problem with reversed coefficients ($K_{\max} = 1, K_{\min} = 10^{-5}$). For $\sigma = 20K$ the L^2-convergence is of order $p + 1$. This is an improvement over the constant penalty case, both in terms of the order and the values of the errors. Results for the other test problems are similar.

Next, we consider the two-level methods: Tables 3–7 display the results in terms of the number of CG iterations required for convergence. The corresponding computational times are provided in Tables 8–12. It can be seen that both two-level

[9] The penalty parameter needs to be sufficiently large to ensure that the SIPG scheme is convergent and the coefficient matrix is SPD. At the same time, a larger penalty parameter typically yields a larger condition number of the coefficient matrix. For certain one-dimensional problems, it suffices to choose $\sigma \geq 2p^2 \frac{k_1^2}{k_0}$, where k_0 and k_1 are the global lower and upper bound respectively of the diffusion coefficient K [31]. Here, we apply similar values in a local fashion by replacing both k_1 and k_0 by local values of K. Specifically, in this paper, we use $\sigma = 20K$ (which is roughly equal to $2p^2K$ for $p = 3$), although smaller values of $\sigma \geq 2p^2K$ yield faster (mesh-independent) results.

Table 2
Bubbly flow with reversed coefficients ($K_{\max} = 1, K_{\min} = 10^{-5}$) and fixed penalty parameter $\sigma = 20$: SIPG convergence in the L^2-norm.

Mesh	$p = 2$		$p = 3$	
	Error	Order	Error	Order
$N = 20^2$	1.96e−01	–	2.47e−02	–
$N = 40^2$	5.22e−02	1.91	3.13e−03	2.98
$N = 80^2$	1.37e−02	1.93	3.93e−04	2.99
$N = 160^2$	3.43e−03	2.00	4.77e−05	3.04

Table 3
Bubbly flow: # CG iterations.

Degree	$p = 2$				$p = 3$			
Mesh	$N = 40^2$	$N = 80^2$	$N = 160^2$	$N = 320^2$	$N = 40^2$	$N = 80^2$	$N = 160^2$	$N = 320^2$
Degrees of freedom	9 600	38 400	153 600	614 400	16 000	64 000	256 000	1 024 000
Prec., 2x BJ	41	42	43	44	55	56	57	58
Prec., 2x BJ ($\omega = 0.7$)	31	31	32	32	33	34	35	35
Defl., 1x BJ	41	39	40	41	45	45	45	46

Table 4
Bubbly flow with reversed coefficients ($K_{\max} = 1, K_{\min} = 10^{-5}$): # CG iterations.

Degree	$p = 2$				$p = 3$			
Mesh	$N = 40^2$	$N = 80^2$	$N = 160^2$	$N = 320^2$	$N = 40^2$	$N = 80^2$	$N = 160^2$	$N = 320^2$
Degrees of freedom	9 600	38 400	153 600	614 400	16 000	64 000	256 000	1 024 000
Prec., 2x BJ	48	57	60	61	58	67	74	77
Prec., 2x BJ ($\omega = 0.7$)	37	41	44	45	43	46	47	48
Defl., 1x BJ	49	55	59	60	55	62	66	67

Table 5
Bubbly flow with reversed coefficients ($K_{\max} = 1, K_{\min} = 10^{-3}$): # CG iterations.

Degree	$p = 2$				$p = 3$			
Mesh	$N = 40^2$	$N = 80^2$	$N = 160^2$	$N = 320^2$	$N = 40^2$	$N = 80^2$	$N = 160^2$	$N = 320^2$
Degrees of freedom	9 600	38 400	153 600	614 400	16 000	64 000	256 000	1 024 000
Prec., 2x BJ	50	53	53	54	63	65	68	69
Prec., 2x BJ ($\omega = 0.7$)	38	40	41	41	43	44	44	45
Defl., 1x BJ	52	54	54	57	56	59	60	61

Table 6
Bubbly flow with reversed coefficients ($K_{\max} = 1, K_{\min} = 10^{-1}$): # CG iterations.

Degree	$p = 2$				$p = 3$			
Mesh	$N = 40^2$	$N = 80^2$	$N = 160^2$	$N = 320^2$	$N = 40^2$	$N = 80^2$	$N = 160^2$	$N = 320^2$
Degrees of freedom	9 600	38 400	153 600	614 400	16 000	64 000	256 000	1 024 000
Prec., 2x BJ	46	46	46	46	59	60	60	61
Prec., 2x BJ ($\omega = 0.7$)	35	36	36	36	38	38	38	39
Defl., 1x BJ	46	47	47	48	49	50	52	52

Table 7
Neumann BCs: # CG iterations.

Degree	$p = 2$				$p = 3$			
Mesh	$N = 40^2$	$N = 80^2$	$N = 160^2$	$N = 320^2$	$N = 40^2$	$N = 80^2$	$N = 160^2$	$N = 320^2$
Degrees of freedom	9 600	38 400	153 600	614 400	16 000	64 000	256 000	1 024 000
Prec., 2x BJ	45	45	45	45	59	59	60	60
Prec., 2x BJ ($\omega = 0.7$)	34	35	36	36	36	37	37	38
Defl., 1x BJ	47	47	47	47	49	49	50	50

methods yield fast and mesh-independent convergence. Without damping, deflation is the most efficient. When a suitable damping value is known, the preconditioning variant performs comparable to deflation.

Factors that influence the convergence rate. There are several factors that can affect the convergence rates in Tables 3–7.

Table 8
Bubbly flow: CPU time in seconds.

Degree	$p = 2$				$p = 3$			
Mesh	$N = 40^2$	$N = 80^2$	$N = 160^2$	$N = 320^2$	$N = 40^2$	$N = 80^2$	$N = 160^2$	$N = 320^2$
Degrees of freedom	9 600	38 400	153 600	614 400	16 000	64 000	256 000	1 024 000
Prec., 2x BJ	0.05	0.28	1.44	7.58	0.18	0.83	3.85	17.89
Prec., 2x BJ ($\omega = 0.7$)	0.04	0.20	1.08	5.57	0.11	0.51	2.39	10.96
Defl., 1x BJ	0.04	0.19	1.08	6.13	0.10	0.47	2.27	11.09

Table 9
Bubbly flow with reversed coefficients ($K_{\max} = 1$, $K_{\min} = 10^{-5}$): CPU time in seconds.

Degree	$p = 2$				$p = 3$			
Mesh	$N = 40^2$	$N = 80^2$	$N = 160^2$	$N = 320^2$	$N = 40^2$	$N = 80^2$	$N = 160^2$	$N = 320^2$
Degrees of freedom	9 600	38 400	153 600	614 400	16 000	64 000	256 000	1 024 000
Prec., 2x BJ	0.06	0.37	2.00	10.21	0.19	0.98	4.92	23.72
Prec., 2x BJ ($\omega = 0.7$)	0.04	0.27	1.48	7.59	0.14	0.68	3.15	14.96
Defl., 1x BJ	0.04	0.27	1.57	8.89	0.12	0.66	3.25	15.96

Table 10
Bubbly flow with reversed coefficients ($K_{\max} = 1$, $K_{\min} = 10^{-3}$): CPU time in seconds.

Degree	$p = 2$				$p = 3$			
Mesh	$N = 40^2$	$N = 80^2$	$N = 160^2$	$N = 320^2$	$N = 40^2$	$N = 80^2$	$N = 160^2$	$N = 320^2$
Degrees of freedom	9 600	38 400	153 600	614 400	16 000	64 000	256 000	1 024 000
Prec., 2x BJ	0.06	0.33	1.73	9.32	0.21	0.97	4.54	21.21
Prec., 2x BJ ($\omega = 0.7$)	0.05	0.25	1.35	7.12	0.14	0.67	2.97	14.02
Defl., 1x BJ	0.04	0.27	1.44	8.43	0.13	0.61	2.97	14.51

Table 11
Bubbly flow with reversed coefficients ($K_{\max} = 1$, $K_{\min} = 10^{-1}$): CPU time in seconds.

Degree	$p = 2$				$p = 3$			
Mesh	$N = 40^2$	$N = 80^2$	$N = 160^2$	$N = 320^2$	$N = 40^2$	$N = 80^2$	$N = 160^2$	$N = 320^2$
Degrees of freedom	9 600	38 400	153 600	614 400	16 000	64 000	256 000	1 024 000
Prec., 2x BJ	0.06	0.30	1.52	7.98	0.20	0.89	4.00	18.80
Prec., 2x BJ ($\omega = 0.7$)	0.04	0.24	1.20	6.29	0.13	0.57	2.56	12.16
Defl., 1x BJ	0.04	0.23	1.26	7.17	0.11	0.52	2.56	12.45

Table 12
Neumann BCs: CPU time in seconds.

Degree	$p = 2$				$p = 3$			
Mesh	$N = 40^2$	$N = 80^2$	$N = 160^2$	$N = 320^2$	$N = 40^2$	$N = 80^2$	$N = 160^2$	$N = 320^2$
Degrees of freedom	9 600	38 400	153 600	614 400	16 000	64 000	256 000	1 024 000
Prec., 2x BJ	0.06	0.29	1.49	7.78	0.20	0.87	4.04	18.64
Prec., 2x BJ ($\omega = 0.7$)	0.04	0.23	1.20	6.27	0.12	0.56	2.52	11.92
Defl., 1x BJ	0.04	0.24	1.27	7.07	0.11	0.53	2.49	12.05

First, a larger value of the penalty parameter σ typically leads to a larger condition number of the coefficient matrix, and thus more CG iterations. This motivates the use of a diffusion-dependent penalty parameter, as its local approach (cf. footnote 9) yields significantly smaller (yet effective) penalties. At the same time, we stress that using $\sigma = 20K$ is not the smallest possible value, and that smaller choices for σ yield faster (mesh-independent) results. For instance, for the reversed bubbly flow problem with $K_{\min} = 10^{-5}$, $p = 2$ and $N = 320^2$, using $\sigma = 1.66p^2K = 6.64K$ resulted in 27 (Prec., $\omega = 0.7$) and 38 (Defl.) iterations, rather than the 45 and 60 iterations observed in Table 4.

Furthermore, the number of iterations seems to increase slightly with the polynomial degree p (e.g. for the bubbly flow problem with $p = 1$ and $N = 80^2$, we observed 35 (Prec.), 30 (Prec., $\omega = 0.7$) and 39 (Defl.) CG iterations).

Moreover, a larger jump in the coefficients can somewhat reduce the convergence speed. For the reversed bubbly flow problem, this can be seen in Tables 4–6. At the same time, for a Poisson problem with $p = 3$ and $N = 160^2$, we observed 54 (Prec.), and 38 (Defl.) CG iterations: compared to the results in Tables 3–7, the differences are quite small.

Finally, the effectiveness of our approach depends on the applied smoother and the underlying discretization. For example, using standard Jacobi or a constant penalty parameter can lead to poor convergence [27]. Helenbrook et al. [7] have also observed that using $p = 0$ at the coarse level can yield poor convergence for certain LDG discretizations and the scheme of Bassi et al. (cf. [7]) (and propose to use continuous basis functions at the coarse level instead). We speculate that the mesh-independent results in Tables 3 and 7 can be explained by an appropriate combination of low- and higher-order information: the coarse matrix (with $p = 0$) is quite similar to the (low-order) central difference matrix for this problem (due to the diffusion-dependent penalty parameter). The higher-order information (regarding the higher-order basis functions) is re-introduced via the block Jacobi smoother. Altogether, the effectiveness of using $p = 0$ at the coarse level in linear solvers for DG matrices depends on the specific underlying DG discretization scheme and the applied smoother. Further investigation of this topic is left for future research.

Performance per iteration. Aside from the number of iterations (and the factors that influence it), the overall performance of the two-level methods is also affected by the costs per iteration. These depend on the following three aspects.

The first aspect is the coarse correction operator. A popular choice is to make use of continuous linear basis functions [15, 14]. In this work, we use a coarse space that is based on the piecewise constant DG basis functions. The corresponding restriction (and prolongation) operator can be applied to a vector by simply extracting elements (and inserting zeros), so these costs are negligible. Furthermore, the coarse matrix A_0 is quite similar to a central difference matrix, for which very efficient solvers are readily available. These two properties of our coarse correction operator limit the costs per iteration and, as such, contribute to the overall efficiency.

The second aspect is the smoother. Examples of choices include pre- and post-smoothing with symmetric Gauss–Seidel [15] and with block Gauss–Seidel [14]. Here, we observe that the cheaper block Jacobi smoother is sufficient to complement our coarse correction operator. As the smoother is applied in each iteration, and even twice for the preconditioning variant, the high efficiency of our smoother positively affects the overall performance of the scheme.

The third aspect is the number of smoothing operations per iteration. Two-level methods that take the form of a preconditioner require two smoothing steps, while some deflation variants, including the ADEF2 method considered in this paper, require only one smoothing step per iteration. This is why the costs per iteration are lower for deflation than for preconditioning in our study (cf. [16] for a detailed comparison in terms of floating point operations). For more expensive smoothers, this benefit of deflation can be expected to be even more relevant.

Altogether, aside from mesh-independent convergence in terms of the number of CG iterations, a significant advantage of the two-level methods considered in this paper is that the costs per iteration are relatively low.

6. Conclusion

This paper is focused on a two-level preconditioner proposed in [15] and the corresponding BNN (ADEF2) deflation variant for linear SIPG systems. For both two-level methods, we have found that the condition number of the preconditioned system can be bounded independently of the mesh element diameter, implying scalable CG convergence. This result is valid for any polynomial degree $p \geq 1$. We have verified that the restrictions on the smoother are satisfied for block Jacobi smoothing. Numerical experiments with strong variations in the coefficients illustrate our main result. Future research could focus on a theoretical *comparison* of both two-level methods and on more advanced, larger scale numerical test cases. Furthermore, the performance of both two-level methods could be compared to that of other preconditioning strategies.

Acknowledgment

We would like thank Yvan Notay for his useful suggestions for generalizing the theory in this paper.

References

[1] P. Castillo, Performance of discontinuous Galerkin methods for elliptic PDEs, SIAM J. Sci. Comput. 24 (2) (2002) 524–547.

[2] S.J. Sherwin, R.M. Kirby, J. Peiró, R.L. Taylor, O.C. Zienkiewicz, On 2D elliptic discontinuous Galerkin methods, Internat. J. Numer. Methods Engrg. 65 (5) (2006) 752–784.

[3] J. Xu, Iterative methods by space decomposition and subspace correction, SIAM Rev. 34 (4) (1992) 581–613.

[4] P. Hemker, W. Hoffman, M.v. Raalte, Two-level Fourier analysis of a multigrid approach for discontinuous Galerkin discretization, SIAM J. Sci. Comput. 25 (3) (2003) 1018–1041.

[5] S.C. Brenner, J. Zhao, Convergence of multigrid algorithms for interior penalty methods, Appl. Numer. Anal. Comput. Math. 2 (1) (2005) 3–18.

[6] J. Gopalakrishnan, G. Kanschat, A multilevel discontinuous Galerkin method, Numer. Math. 95 (3) (2003) 527–550.

[7] B.T. Helenbrook, H.L. Atkins, Solving discontinuous Galerkin formulations of Poisson's equation using geometric and *p*-multigrid, Numer. Linear Algebra Appl. 46 (9) (2008) 894–902.

[8] K.J. Fidkowski, T.A. Oliver, J. Lu, D.L. Darmofal, *p*-Multigrid solution of high-order discontinuous Galerkin discretizations of the compressible Navier–Stokes equations, J. Comput. Phys. 207 (1) (2005) 92–113.

[9] P.-O. Persson, J. Peraire, Newton-GMRES preconditioning for discontinuous Galerkin discretizations of the Navier–Stokes equations, SIAM J. Sci. Comput. 30 (6) (2008) 2709–2733.

[10] F. Prill, M. Lukáčová-Medviďová, R. Hartmann, Smoothed aggregation multigrid for the discontinuous Galerkin method, SIAM J. Sci. Comput. 31 (5) (2009) 3503–3528.

[11] Y. Saad, B. Suchomel, ARMS: an algebraic recursive multilevel solver for general sparse linear systems, Numer. Linear Algebra Appl. 9 (5) (2002) 359–378.

[12] P.F. Antonietti, B. Ayuso, Schwarz domain decomposition preconditioners for discontinuous Galerkin approximations of elliptic problems: non-overlapping case, M2AN Math. Model. Numer. Anal. 41 (1) (2007) 21–54.

[13] X. Feng, O.A. Karakashian, Two-level additive Schwarz methods for a discontinuous Galerkin approximation of second order elliptic problems, SIAM J. Numer. Anal. 39 (4) (2001) 1343–1365 (electronic).

[14] P. Bastian, M. Blatt, R. Scheichl, Algebraic multigrid for discontinuous Galerkin discretizations of heterogeneous elliptic problems, Numer. Linear Algebra Appl. 19 (2) (2012) 367–388.

[15] V.A. Dobrev, R.D. Lazarov, P.S. Vassilevski, L.T. Zikatanov, Two-level preconditioning of discontinuous Galerkin approximations of second-order elliptic equations, Numer. Linear Algebra Appl. 13 (9) (2006) 753–770.

[16] P. van Slingerland, C. Vuik, Fast linear solver for diffusion problems with applications to pressure computation in layered domains, Comput. Geosci. (2014) http://dx.doi.org/10.1007/s10596-014-9400-8.

[17] J.M. Tang, R. Nabben, C. Vuik, Y.A. Erlangga, Comparison of two-level preconditioners derived from deflation, domain decomposition and multigrid methods, J. Sci. Comput. 39 (3) (2009) 340–370.

[18] R.D. Falgout, P.S. Vassilevski, L.T. Zikatanov, On two-grid convergence estimates, Numer. Linear Algebra Appl. 12 (5–6) (2005) 471–494.

[19] P. van Slingerland, C. Vuik, Scalable two-level preconditioning and deflation basd on a piecewise constant subspace for (SIP)DG systems, Tech. Rep. 12-11, Delft University of Technology, 2012.

[20] B. Rivière, Discontinuous Galerkin Methods for Solving Elliptic and Parabolic Equations: Theory and Implementation, in: Frontiers in Applied Mathematics, vol. 35, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2008.

[21] P.G. Ciarlet, The Finite Element Method for Elliptic Problems, in: Classics in Applied Mathematics, vol. 40, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2002.

[22] P.S. Vassilevski, Multilevel Block Factorization Preconditioners: Matrix-Based Analysis and Algorithms for Solving Finite Element Equations, Springer, New York, 2008.

[23] Y. Notay, Algebraic analysis of two-grid methods: the nonsymmetric case, Numer. Linear Algebra Appl. 17 (1) (2010) 73–96.

[24] R. Horn, C. Johnson, Matrix Analysis, Cambridge University Press, Cambridge, 1988.

[25] L. Zikatanov, Two-sided bounds on the convergence rate of two-level methods, Numer. Linear Algebra Appl. 15 (2008) 439–454.

[26] J.M. Tang, S.P. MacLachlan, R. Nabben, C. Vuik, A comparison of two-level preconditioners based on multigrid and deflation, SIAM J. Matrix Anal. Appl. 31 (4) (2009–2010) 1715–1739.

[27] P. van Slingerland, Discontinuous Galerkin methods: linear systems and hidden accuracy (Ph.D. thesis) Delft University of Technology, 2013.

[28] J. Tang, Two-level preconditioned conjugate gradient methods with applications to bubbly flow problems (Ph.D. thesis) Delft University of Technology, 2008.

[29] O. Axelsson, Iterative Solution Methods, Cambridge University Press, Cambridge, 1994.

[30] K. Johannsen, A symmetric smoother for the nonsymmetric interior penalty discontinuous Galerkin discretization, Tech. Rep. ICES Report 05-23, University of Texas at Austin, 2005.

[31] Y. Epshteyn, B. Rivière, Estimation of penalty parameters for symmetric interior penalty Galerkin methods, J. Comput. Appl. Math. 206 (2) (2007) 843–872.

[32] M. Dryja, On discontinuous Galerkin methods for elliptic problems with discontinuous coefficients, Comput. Methods Appl. Math. 3 (1) (2003) 76–85. (electronic).

[33] J. Proft, B. Rivière, Discontinuous Galerkin methods for convection–diffusion equations for varying and vanishing diffusivity, Int. J. Numer. Anal. Model. 6 (4) (2009) 533–561.

[34] A. Ern, A.F. Stephansen, P. Zunino, A discontinuous Galerkin method with weighted averages for advection–diffusion equations with locally small and anisotropic diffusivity, IMA J. Numer. Anal. 29 (2) (2009) 235–256.