

# Robust multigrid preconditioners for cell-centered finite volume discretization of the high-contrast diffusion equation

Burak Aksoylu · Zuhail Yeter

REVISED: MARCH 22, 2010

**Abstract** We study a conservative 5-point cell-centered finite volume discretization of the high-contrast diffusion equation. We aim to construct preconditioners that are robust with respect to the magnitude of the coefficient contrast and the mesh size simultaneously. For that, we prove and numerically demonstrate the robustness of the preconditioner proposed by Aksoylu et al. (2008, *Comput. Vis. Sci.* 11, pp. 319–331) by extending the devised singular perturbation analysis from linear finite element discretization to the above discretization. The singular perturbation analysis is more involved than that of finite element case because all the subblocks in the discretization matrix depend on the diffusion coefficient. However, as the diffusion coefficient approaches infinity, that dependence is eliminated. This allows the same preconditioner to be utilized due to similar limiting behaviours of the submatrices; leading to a narrowing family of preconditioners that can be used for different discretizations. Therefore, we have accomplished a desirable preconditioner design goal. We compare our numerical results to standard cell-centered multigrid implementations and observe that performance of our preconditioner is independent of the utilized smoothers and prolongation operators.

As a side result, we also prove a fundamental qualitative property of solution of the high-contrast diffusion equation. Namely, the solution over the highly-diffusive island becomes constant asymptotically. Integration of this qualitative understanding of the underlying PDE

to our preconditioner is the main reason behind its superior performance.

Diagonal scaling is probably the most basic preconditioner for high-contrast coefficients. Extending the matrix entry based spectral analysis introduced by Graham and Hagger, we rigorously show that the number of small eigenvalues of the diagonally scaled matrix depends on the number of isolated islands comprising the highly-diffusive region. This indicates that diagonal scaling creates a significant clustering of the spectrum, a favorable property for faster convergence of Krylov subspace solvers.

**Keywords** Diffusion equation · high-contrast coefficients · interface problem · discontinuous coefficients · cell-centered multigrid · mass conservative · finite volume · cell-centered discretization · singular perturbation analysis · diagonal scaling · deflation

**Mathematics Subject Classification (2000)** 65F10 · 65N22 · 65N55 · 65F35 · 15A12 · 65N55

## 1 Introduction

We advocate that qualitative understanding of the PDE operators and their dependence on the coefficients is essential for designing preconditioners. Because, the performance, hence the robustness, of a preconditioner depends essentially on the degree to which the preconditioner approximates the properties of the underlying PDE. Therefore, designing preconditioners involves a process that draws heavily upon effective utilization of tools from operator theory as well as singular perturbation analysis (SPA). In the operator theory framework, Aksoylu and Beyer [5,4] have studied the diffusion equation with rough coefficients. The roughness of

---

B. Aksoylu (corresponding author) · Z. Yeter  
Department of Mathematics &  
Center for Computation and Technology,  
Louisiana State University  
216 Johnston Hall, Baton Rouge LA, 70803 USA  
E-mail: burak@cct.lsu.edu, zyeter1@cct.lsu.edu

coefficients creates serious complications. For instance, it was shown in [5] that the standard elliptic regularity in the smooth coefficient case fails to hold. Moreover, the domain of the diffusion operator heavily depends on the regularity of the coefficients.

Roughness of PDE coefficients causes loss of robustness of preconditioners. We aim to establish robustness with respect to the magnitude of the coefficient contrast and the mesh size simultaneously. In that regard, SPA provides valuable insight into the qualitative nature of the underlying PDE. In the case of linear finite element (FE), Aksoylu et al. [7] devised a SPA on the matrix entries to study the robustness of the same preconditioner  $B_{AGKS}$  under consideration in this article. SPA turned out to be an effective tool in analyzing certain behaviors of the discretization matrix  $K$  such as the asymptotic rank, decoupling, low-rank perturbations of the resulting submatrices. This information in turn is exploited to accomplish dramatic computational savings. In [7], we also provided a rigorous convergence analysis of  $B_{AGKS}$ . Hence, with the insights provided by SPA and the operator theory, we are in control of the effectiveness and computational feasibility at the same time. Furthermore, the authors [9] also extended the use of the family of preconditioners  $B_{AGKS}$  for HCT and Morley discretizations of the high-contrast biharmonic plate equation. The broadness of the applicability of  $B_{AGKS}$  has been achieved due to the designed SPA.

The preconditioner  $B_{AGKS}$  originates from the family of robust preconditioners constructed by Aksoylu and Klie [8] for the cell-centered finite volume (FV) discretization of the high-contrast diffusion equation for porous media flow related applications based on the two-point approximation scheme studied in [2]. However, for the original variants of  $B_{AGKS}$ , robustness with respect to the contrast size was the main design feature. Hence, the emphasis in [8] was mostly on deflation methods (that are used as stage-two preconditioners) based on the Krylov subspace solvers. As in [7], in this article, we incorporate (cell-centered) multigrid preconditioners to restore robustness with respect to the mesh size. Hence, the main purpose of this article is to extend the preconditioner  $B_{AGKS}$  and the related analysis to a 5-point conservative cell-centered FV discretization. From the flow application perspective, maintaining the continuity of the flux across the control volume interfaces ensures the highly popular and crucial discretization property of *local mass conservation*. Our interest in flow applications is the main reason behind conducting research in the direction of mass conservative discretizations.

We prove and numerically demonstrate that the very same preconditioner  $B_{AGKS}$  that is used for FE dis-

cretization can also be used with minimal modification for FV discretizations. This was possible by the help of SPA because we have identified similar algebraic features of the discretization matrices between linear FE and FV methods. The same preconditioner can be utilized due to similar limiting behaviours of the submatrices. This observation leads to a narrowing family of preconditioners that can be used for different discretizations. Therefore, we have accomplished to construct a preconditioner that is compatible with and equally effective under different discretizations, and this is a desirable feature in the design and construction of preconditioners. In addition, extension to FV discretization does not spoil  $B_{AGKS}$ 's inherit algebraic nature. The first algebraic phase involves partitioning of the degrees of freedom (DOF) into a set corresponding to a high-diffusivity and a low-diffusivity region. For high enough contrast, we can still obtain the partitioning by examining the diagonal entries of  $K$ . This simple algebraic examination rules out the standard multigrid requirement of the coarsest mesh resolving the boundary of the island.

The diffusion equation with discontinuous coefficients is known as the *interface problem* in the computational fluid dynamics community. There has been intense research activity on the interface problem. It is such a well-established problem that one can find it in the text books dedicated to multigrid methods, for instance, by Wesseling [31] and Trottenberg, Oosterlee, and Schüller [26]. Mohr and Wienands [22] revisited the cell-centered multigrid (CCMG) preconditioner and attributed the pioneering CCMG to Wesseling [30]; see [22] and the references therein for further review of CCMG.

For interface problems, there have been many attempts to construct problem independent prolongation and restriction operators to accommodate the roughness of coefficients. Among the variational approaches, Wesseling [30] and Wesseling and Khalil [32] constructed such prolongations for high-contrast coefficients, whereas Kwak [18] studied medium-contrasts. Kwak and Lee [19] proposed problem dependent prolongations for medium-contrasts. Among the non-variational approaches, Ewing and Shen [10] examined medium-contrast with piecewise constant prolongation and restriction operators. In our CCMG and  $B_{AGKS}$  implementations, we prefer problem independent prolongation operators utilizing Galerkin variational approach based on Wesseling and Khalil [32] and bi-linear interpolation; see [22, Figure 2] or [31, p. 72].

Diagonal scaling is probably the most basic preconditioner for discretizations with high-contrast coefficients. Although diagonal scaling has no effect on the asymptotic behaviour of the condition number, Gra-

ham and Hagger [14] observed, in the case of FE, that spectrum of the diagonally scaled matrix  $A$  enjoys better clustering than that of  $K$ . The spectrum of  $A$  is bounded from above and below except a single eigenvalue in the case of a single isolated highly-diffusive island. On the other hand, the spectrum of  $K$  contains eigenvalues approaching infinity with cardinality depending on the number of DOF contained within highly-diffusive island. For faster convergence, the Krylov subspace solver favors the clustering provided by diagonal scaling. Based on the matrix entry utilizing spectral analysis introduced by Graham and Hagger [14] for linear FE, we extend the spectral analysis to cell-centered FV discretization and obtain a similar spectral result for  $A$ . Namely, the number of small eigenvalues of  $A$  depends on the number of isolated islands comprising the highly-diffusive region. This rigorous analysis is presented in Section 4.

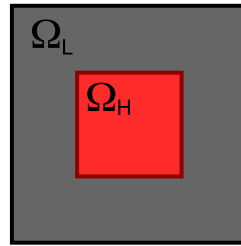
We extend the devised SPA from FE to FV discretization in order to explain the properties of the submatrices related to the stiffness matrix  $K(m)$  defined in (2). In particular, SPA of  $K_{HH}(m)$ , the highly-diffusive block of  $K(m)$  in the decomposition (4), as diffusivity  $m \rightarrow \infty$ , has important implications for the behaviour of the Schur complement  $S(m)$  of  $K_{HH}(m)$  in  $K(m)$ . Namely,

$$S(m) = S^\infty + \mathcal{O}(m^{-1}),$$

where  $S^\infty$  is a low rank perturbation of  $K_{LL}^\infty$  which denotes the limiting matrix of  $K_{LL}(m)$ , the lowly-diffusive block of  $K(m)$  in (4). The rank of the perturbation depends on the number of disconnected components comprising the highly-diffusive region. This special limiting form of  $S(m)$  allows us to build a robust approximation of  $S(m)^{-1}$  by merely using solvers for  $K_{LL}^\infty$  by the help of the Sherman-Morrison-Woodbury formula [13, Eq. (2.1.4)]. In Section 5, by using SPA, the asymptotic behaviour of submatrices is provided and the final convergence proof of  $B_{AGKS}$  is based on that result.

The remainder of the article is structured as follows. In Section 6, we reveal the qualitative nature of the solution of the high-contrast diffusion equation and the resulting decoupling in the solution. This valuable insight is provided by SPA. In Section 7, we highlight the implementation aspects of  $B_{AGKS}$ . In addition, we present how subdomain deflation gives rise to a dramatic computational saving due to reduction to a block diagonal system. Finally, in Section 8, we numerically demonstrate the simultaneous robustness of  $B_{AGKS}$  with respect to magnitude of the coefficient contrast and the mesh size. We also provide comparisons to CCMG preconditioner with different prolongation operators and smoothers.

## 2 The underlying PDE and the linear system



**Fig. 1**  $\Omega = \overline{\Omega}_H \cup \Omega_L$  where  $\Omega_H$  and  $\Omega_L$  are high and low diffusivity regions, respectively.

We study the following high-contrast stationary diffusion equation.

$$\begin{cases} -\nabla \cdot (\alpha \nabla u) = f & \text{in } \Omega \\ u = 0 & \text{on } \partial\Omega \end{cases} \quad (1)$$

where  $\Omega \subset \mathbb{R}^d$ ,  $d = 2, 3$ . The coefficient  $\alpha(x)$  may vary over many orders of magnitude in an unstructured way on  $\Omega$ . Problems with high-contrast coefficients are ubiquitous in porous media flow applications. Many examples of this kind arise in groundwater flow and oil reservoir simulation; see for example the comprehensive overviews [1, 12, 21, 24]. Consequently, development of efficient solvers for high-contrast heterogeneous media has been an active area of research, specifically in the setting of multiscale solvers [3, 14, 15, 16, 25]. In addition, the fictitious domain method and composite materials are also sources of rough coefficients; see the references in [17]. Important current applications deal with composite materials whose components have nearly constant diffusivity, but vary by several orders of magnitude. In composite material applications, it is quite common to idealize the diffusivity by a piecewise constant function. Likewise, we restrict the diffusion process to a *binary regime* (see Figure 1) in which the coefficient  $\alpha$  is a piecewise constant function with the following values:

$$\alpha(x) = \begin{cases} m \gg 1, & x \in \Omega_H, \\ 1, & x \in \Omega_L. \end{cases}$$

Due to the atomistic structure of matter, the physical treatment of diffusion involves regular ( $C^\infty$ -) diffusivity, Aksoylu and Beyer [5] showed that the idealization of diffusivity by piecewise constant coefficients is meaningful by showing a continuous dependence of the solutions on the diffusivity.

Let us denote the linear system arising from the finite volume discretization by:

$$K(m) x(m) = b. \quad (2)$$

Let  $\Omega$  be decomposed with respect to diffusivity value as

$$\Omega = \overline{\Omega}_H \cup \Omega_L, \quad (3)$$

where  $\Omega_H$  and  $\Omega_L$  denote the high and low diffusivity regions, respectively. When  $m$ -dependence is explicitly stated and the discretization system (2) is decomposed with respect to (3), i.e., the magnitude of the coefficient values, we arrive at the following  $2 \times 2$  block system:

$$\begin{bmatrix} K_{HH}(m) & K_{HL}(m) \\ K_{LH}(m) & K_{LL}(m) \end{bmatrix} \begin{bmatrix} x_H(m) \\ x_L(m) \end{bmatrix} = \begin{bmatrix} b_H \\ b_L \end{bmatrix}. \quad (4)$$

Note that all the subblocks in (4) have  $m$ -dependence. This is the main difference from the FE discretization in which only the  $HH$ -subblock has  $m$ -dependence. Hence, the perturbation analysis of the FV discretization becomes more involved than that of the FE discretization.

The exact inverse of  $K$  can be written as:

$$\begin{aligned} K^{-1}(m) &= \begin{bmatrix} I_{HH} & -K_{HH}^{-1}(m)K_{HL}(m) \\ 0 & I_{LL} \end{bmatrix} \\ &\times \begin{bmatrix} K_{HH}^{-1}(m) & 0 \\ 0 & S^{-1}(m) \end{bmatrix} \\ &\times \begin{bmatrix} I_{HH} & 0 \\ -K_{LH}(m)K_{HH}^{-1}(m) & I_{LL} \end{bmatrix}, \end{aligned} \quad (5)$$

where  $I_{HH}$  and  $I_{LL}$  denote the identity matrices of the appropriate dimension and the  $S(m)$  is the Schur complement of  $K_{HH}(m)$  in  $K(m)$  given by

$$S(m) = K_{LL}(m) - K_{LH}(m)K_{HH}^{-1}(m)K_{HL}(m). \quad (6)$$

Let  $\mathcal{N}_{HH}$  denote the Neumann matrix corresponding to the pure Neumann problem for the Laplace operator on  $\Omega_H$ . We write an important decomposition which will be used in the analysis to come:

$$K_{HH}(m) = m\mathcal{N}_{HH} + \Delta(m). \quad (7)$$

### 3 The cell-centered finite volume discretization

#### 3.1 The outline of the discretization scheme

Since the utilized analysis is based on matrix entries, we provide the outline of the cell-centered FV discretization for the 5-point stencil used in this article; for more details see [11]. Let  $\mathcal{T} = V_{i,j}$  with  $i, j = 1, \dots, n^{1/2}$  be the mesh and the control volume be defined by

$$V_{i,j} = [x_{i-1/2}, x_{i+1/2}] \times [y_{j-1/2}, y_{j+1/2}].$$

The FV scheme is constructed by integrating (1) over each control volume  $V_{i,j}$ , which yields

$$\begin{aligned} & - \int_{y_{j-1/2}}^{y_{j+1/2}} \alpha_{i+1/2,j} u_x(x_{i+1/2}, y) dy \\ & + \int_{y_{j-1/2}}^{y_{j+1/2}} \alpha_{i-1/2,j} u_x(x_{i-1/2}, y) dy \\ & + \int_{x_{i-1/2}}^{x_{i+1/2}} \alpha_{i,j-1/2} u_y(x, y_{j-1/2}) dx \\ & - \int_{x_{i-1/2}}^{x_{i+1/2}} \alpha_{i,j+1/2} u_y(x, y_{j+1/2}) dx \\ & = \int_{V_{i,j}} f(x, y) dx dy. \end{aligned}$$

Let  $k_{x_i} = x_{i+1/2} - x_{i-1/2}$  and  $k_{y_j} = y_{j+1/2} - y_{j-1/2}$  be the  $x$ - and  $y$ -size of the control volume  $V_{i,j}$ , respectively. Since a uniform mesh is used,  $k_{x_i} = k_x$  and  $k_{y_j} = k_y$  for all  $i, j = 1, \dots, n^{1/2}$ . Then, the discretization scheme is

$$F_{i+1/2,j} - F_{i-1/2,j} + F_{i,j+1/2} - F_{i,j-1/2} = h_{i,j} f_{i,j},$$

where  $h_{i,j} = k_x k_y$ , and  $f_{i,j}$  is the mean value of  $f$  over  $V_{i,j}$ , and where

$$F_{i+1/2,j} = -\frac{k_y}{k_x} \bar{h}_{\alpha_i} \{u(x_{i+1}, y_j) - u(x_i, y_j)\},$$

$$F_{i,j+1/2} = -\frac{k_x}{k_y} \bar{h}_{\alpha_j} \{u(x_i, y_{j+1}) - u(x_i, y_j)\},$$

and

$$\bar{h}_{\alpha_i} = \frac{2\alpha_{i,j}\alpha_{i+1,j}}{\alpha_{i,j} + \alpha_{i+1,j}}, \quad \bar{h}_{\alpha_j} = \frac{2\alpha_{i,j}\alpha_{i,j+1}}{\alpha_{i,j} + \alpha_{i,j+1}}. \quad (8)$$

Note that  $\bar{h}_{\alpha_i}$  and  $\bar{h}_{\alpha_j}$  are the harmonic means of the diffusion coefficients for the adjacent control volumes in  $x$  and  $y$  directions respectively. The discretization formula can be written explicitly as follows:

$$\begin{aligned} & - \bar{h}_{\alpha_i} u(x_{i+1}, y_j) - \bar{h}_{\alpha_{i-1}} u(x_{i-1}, y_j) \\ & + (\bar{h}_{\alpha_i} + \bar{h}_{\alpha_{i-1}} + \bar{h}_{\alpha_j} + \bar{h}_{\alpha_{j-1}}) u(x_i, y_j) \\ & - \bar{h}_{\alpha_j} u(x_i, y_{j+1}) - \bar{h}_{\alpha_{j-1}} u(x_i, y_{j-1}) = h_{i,j} f_{i,j}. \end{aligned} \quad (9)$$

In our binary diffusivity regime, for notational convenience, we denote the harmonic mean by

$$\bar{h}_m := \frac{2m}{m+1}. \quad (10)$$

The *harmonic mean* is used to ensure the *continuity of the flux* across the control volume interfaces. As a result, this flavor of finite volume discretization enjoys the desirable property of mass conservation.

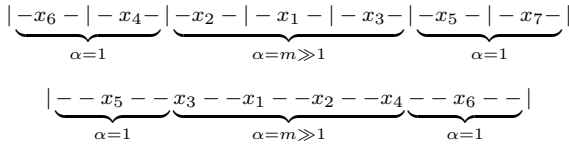
One can write the discretization matrix entries *a priori* due to the formula (9). Hence, in 2D, we explicitly state each contribution of the off-diagonals to the diagonal entry values in the following:

$$[K(m)]_{pp} = \quad (11)$$

$$\begin{cases} m + m + m + m, & p \in I_{\Omega_1}, \\ m + m + m + \bar{h}_m, & p \in \Gamma_{\Omega_1} \text{ and non-corner}, \\ m + m + \bar{h}_m + \bar{h}_m, & p \in \Gamma_{\Omega_1} \text{ and corner}, \\ 1 + 1 + 1 + \bar{h}_m, & p \in \Gamma_{\Omega_2}, \\ 1 + 1 + 1 + 1, & p \in I_{\Omega_2} \text{ and strictly interior}, \\ 5, & p \in I_{\Omega_2} \text{ and non-corner bdry}, \\ 6, & p \in I_{\Omega_2} \text{ and corner bdry}, \end{cases}$$

where bdry denotes boundary.

### 3.2 Comparison of finite element and finite volume discretizations on a 1D example



**Fig. 2** (Top) The finite volume mesh where the cell-centers are denoted by  $x_i$ ,  $i = 1, \dots, 7$ . (Bottom) The finite element mesh where the vertex locations are denoted by  $x_i$ ,  $i = 1, \dots, 6$ .

We explicitly provide the discretization matrix utilizing the FV method in (9) corresponding to (1). The domain is chosen to be  $\Omega := (0, 1)$  with the highly-diffusive island  $\Omega_H := (2/7, 5/7)$ . The mesh consists of 7 cells. The cells and cell-centers are denoted by  $V_1, \dots, V_7$  and  $x_1, \dots, x_7$ , respectively. See Figure 2.

The corresponding submatrices in (4) are given below:

$$K_{HH}(m) = \begin{bmatrix} 2m & -m & -m \\ -m & m + \bar{h}_m & 0 \\ -m & 0 & m + \bar{h}_m \end{bmatrix},$$

$$K_{HL}(m) = \begin{bmatrix} 0 & 0 & 0 \\ -\bar{h}_m & 0 & 0 \\ 0 & -\bar{h}_m & 0 \end{bmatrix} = K_{LH}^T(m),$$

$$K_{LL}(m) = \begin{bmatrix} 1 + \bar{h}_m & 0 & -1 & 0 \\ 0 & 1 + \bar{h}_m & 0 & -1 \\ -1 & 0 & 3 & 0 \\ 0 & -1 & 0 & 3 \end{bmatrix}.$$

Moreover, from (7), we obtain

$$\mathcal{N}_{HH}(m) = \begin{bmatrix} 2m & -m & -m \\ -m & m & 0 \\ -m & 0 & m \end{bmatrix},$$

$$\Delta(m) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & \bar{h}_m & 0 \\ 0 & 0 & \bar{h}_m \end{bmatrix}.$$

We readily see that the  $m$ -dependence of the matrices  $K_{LH}(m), K_{HL}(m), K_{LL}(m)$ , and  $\Delta(m)$  is eliminated asymptotically. For instance,

$$\Delta(m) = \Delta^\infty + \mathcal{O}(m^{-1}),$$

$$= \begin{bmatrix} 0 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{bmatrix} + \mathcal{O}(m^{-1}).$$

By maintaining a similar configuration (see Figure 2) used for the FV case, the corresponding submatrices in (4) for the linear FE discretization are given below. DOF that lie on the interface are always included to the DOF of the highly-diffusive island. Hence, only  $K_{HH}^{FE}$  subblock has  $m$ -dependence.

$$K_{HH}^{FE}(m) = \begin{bmatrix} 2m & -m & -m & 0 \\ -m & 2m & 0 & -m \\ -m & 0 & m+1 & 0 \\ 0 & -m & 0 & m+1 \end{bmatrix},$$

$$K_{HL}^{FE} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ -1 & 0 \\ 0 & -1 \end{bmatrix} = K_{LH}^{FE^T},$$

$$K_{LL}^{FE} = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}.$$

$$m\mathcal{N}_{HH}^{FE} = \begin{bmatrix} 2m & -m & -m & 0 \\ -m & 2m & 0 & -m \\ -m & 0 & m & 0 \\ 0 & -m & 0 & m \end{bmatrix},$$

$$\Delta^{FE} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

### 4 Spectral analysis of the diagonally scaled finite volume discretization matrix

We will analyze the behavior of the spectrum of the symmetrically scaled discretization matrix

$$A(m) := (\text{diag}K(m))^{-1/2} K(m) (\text{diag}K(m))^{-1/2}. \quad (12)$$

**Table 1** A 2D example showing the condition numbers and eigenvalues of the finite volume discretization matrix  $K(m)$  and its diagonally scaled version  $A(m)$  in which the eigenvalues are sorted in ascending order.

$m$	$\kappa(K(m))$	$\lambda_1(K(m))$	$\lambda_{61}(K(m))$	$\lambda_{62}(K(m))$	$\lambda_{64}(K(m))$	$\kappa(A(m))$	$\lambda_1(A(m))$	$\lambda_2(A(m))$	$\lambda_{64}(A(m))$
$10^0$	$2.627 \times 10^1$	$3.045 \times 10^{-1}$	$7.695 \times 10^0$	$7.848 \times 10^0$	$8.000 \times 10^0$	$2.553 \times 10^1$	$7.538 \times 10^{-2}$	$1.800 \times 10^{-1}$	$1.925 \times 10^0$
$10^2$	$1.248 \times 10^3$	$3.237 \times 10^{-1}$	$7.995 \times 10^0$	$2.040 \times 10^2$	$4.040 \times 10^2$	$3.448 \times 10^2$	$5.784 \times 10^{-3}$	$1.362 \times 10^{-1}$	$1.994 \times 10^0$
$10^4$	$1.235 \times 10^5$	$3.240 \times 10^{-1}$	$8.000 \times 10^0$	$2.000 \times 10^4$	$4.000 \times 10^4$	$3.258 \times 10^4$	$6.139 \times 10^{-5}$	$1.346 \times 10^{-1}$	$1.999 \times 10^0$
$10^6$	$1.235 \times 10^7$	$3.240 \times 10^{-1}$	$8.000 \times 10^0$	$2.000 \times 10^6$	$4.000 \times 10^6$	$3.256 \times 10^6$	$6.142 \times 10^{-7}$	$1.346 \times 10^{-1}$	$2.000 \times 10^0$
$10^8$	$1.235 \times 10^9$	$3.240 \times 10^{-1}$	$8.000 \times 10^0$	$2.000 \times 10^8$	$4.000 \times 10^8$	$3.256 \times 10^8$	$6.143 \times 10^{-9}$	$1.346 \times 10^{-1}$	$2.000 \times 10^0$
$10^{10}$	$1.235 \times 10^{11}$	$3.240 \times 10^{-1}$	$8.000 \times 10^0$	$2.000 \times 10^{10}$	$4.000 \times 10^{10}$	$3.256 \times 10^{10}$	$6.143 \times 10^{-11}$	$1.346 \times 10^{-1}$	$2.000 \times 10^0$

In the case of a single highly-diffusive island corresponding to the configuration in Figure 1,  $A(m)$  contains only one small eigenvalue for which the below main result is established:

$$C_1 m^{-1} \leq \lambda_{\min}(A(m)) \leq C_2 m^{-1/2}. \quad (13)$$

The analysis also supports multiple highly-diffusive islands. In fact, the number of small eigenvalues of  $A(m)$  depends on the number of isolated islands comprising the highly-diffusive region.

The required analysis is quite technical and will be presented in the appendix. In this section, we outline only the steps leading to the main result (13). The upper bound in (13) is obtained by an application of Courant-Fischer mini-max Theorem; see Lemma 3.

*Remark 1* Numerically we observe that the smallest eigenvalue is  $\mathcal{O}(m^{-1})$ ; see Table 1. The perturbation expansion leads to  $\mathcal{O}(m^{-1})$  estimate with one exceptional case corresponding to the first ‘‘Case 1’’ in §A.1. Hence,  $\mathcal{O}(m^{-1/2})$  estimate is an artifact of the perturbation expansion. The same artifact has appeared in the FE analysis; see [14, Eq. (5.14)].

The lower bound proof is more involved; see §A.2. For that, first we establish the below estimate for  $K(m)$ :

$$x^T K(m)x \geq x^T K(1)x \geq m^{-1} x^T K(m)x. \quad (14)$$

Then, we establish a similar estimate for  $\text{diag } K(m)$ :

$$\begin{aligned} x^T \text{diag } K(m)x &\geq x^T \text{diag } K(1)x \\ &\geq m^{-1} x^T \text{diag } K(m)x. \end{aligned} \quad (15)$$

Combining (14) and (15), yields the lower bound estimate for the smallest eigenvalue:

$$C_1 m^{-1} \leq \lambda_{\min}(A(m)).$$

We illustrate the spectral effects of diagonal scaling by an example in Table 1. In the example,  $K(m)$  has 3 eigenvalues approaching to  $\infty$  and 61 bounded eigenvalues, whereas,  $A(m)$  has only one eigenvalue approaching to zero. The merit of diagonal scaling becomes apparent after studying the spectral behavior of  $A(m)$ . We observe that the number of the eigenvalues of  $A$  of  $\mathcal{O}(m^{-1})$  depends only on the number of isolated islands. On the other hand, the number of the eigenvalues of  $K(m)$  of  $\mathcal{O}(m)$  depends on the number of DOF of the islands, asymptotically. Note that the reduction in the number of  $m$ -dependent eigenvalues is a desirable feature for fast convergence of Krylov subspace solvers.

## 5 Singular perturbation analysis on matrix entries

### 5.1 Preliminaries on matrix properties

The discretization formula (9) with the harmonic means (8) gives rise to the matrix entries given in (11). Due to harmonic means, the  $m$ -dependence of the matrix entries corresponding to  $HL$  and  $LL$  couplings is asymptotically eliminated. As a result, the submatrices  $K_{HL}(m)$  and  $K_{LL}(m)$  do not have  $m$ -dependence asymptotically:

$$\begin{aligned} K_{HL}(m) &= K_{HL}^\infty + \mathcal{O}(m^{-1}), \\ K_{LH}(m) &= K_{HL}^{\infty T} + \mathcal{O}(m^{-1}), \\ K_{LL}(m) &= K_{LL}^\infty + \mathcal{O}(m^{-1}). \end{aligned} \quad (16)$$

The analysis in this article mainly relies on this crucial property.

To analyze the  $m$ -robustness of preconditioners based on (5), we need to analyze the asymptotic behaviour of the block components  $K_{HH}(m)^{-1}$ ,  $S(m)^{-1}$ , and  $K_{LH}(m)K_{HH}(m)^{-1}$  as  $m \rightarrow \infty$ . This is the purpose

of Lemma 1 below. To prepare for this, we further decompose DOF associated with  $\bar{\Omega}_H$  into a set of interior DOF associated with index  $I$  and boundary DOF with index  $\Gamma$ . This leads to the following further block representation of

$$K_{HH}(m) = \begin{bmatrix} K_{II}(m) & K_{I\Gamma}(m) \\ K_{\Gamma I}(m) & K_{\Gamma\Gamma}(m) \end{bmatrix}. \quad (17)$$

By using (11), we can write a more refined expression for the block  $K_{\Gamma\Gamma}(m)$ :

$$K_{\Gamma\Gamma}(m) = K_{\Gamma\Gamma}^{(H)}(m) + \bar{h}_m D_{\Gamma\Gamma}^{(L)},$$

with

$$D_{\Gamma\Gamma}^{(L)} := \text{diag}(0, \dots, 0, 1, \dots, 1, 2, \dots, 2), \quad (18)$$

where the number of 0, 1, and 2 entries is equal to the cardinality of interior, non-corner interface, and corner interface cell-centers, respectively. Thus, we can characterize the Neumann matrix  $\mathcal{N}_{HH}$  on  $\Omega_H$  as described in (7) as follows:

$$K_{HH}(m) = m\mathcal{N}_{HH} + \Delta(m), \quad (19)$$

$$\Delta(m) = \bar{h}_m \begin{bmatrix} 0 & 0 \\ 0 & D_{\Gamma\Gamma}^{(L)} \end{bmatrix}, \quad (20)$$

$$= \Delta^\infty + \mathcal{O}(m^{-1}), \quad (21)$$

where  $\Delta(m)$  is a diagonal matrix due to (18).  $\mathcal{N}_{HH}$  is a SPSD matrix with a simple zero eigenvalue and associated constant eigenvector. If  $n_H$  denotes the number of DOF in  $\Omega_H$ , a suitable normalized eigenvector is the constant vector with entries  $n_H^{-1/2}$ , which we denote by  $e_H$ . We further write in block form as  $e_H^T = (e_I^T, e_\Gamma^T)$ .

*Remark 2* The Neumann matrix in the FV discretization is completely different than that of the FE case. However, as expected, the simple zero eigenvalue and the associated constant eigenvector are maintained because this is an inherent property of the underlying PDE. In addition,  $\Delta$  in (19) in the FE discretization is not necessarily diagonal and does not have  $m$ -dependence.

Finally we note that the off-diagonal blocks in (4) have the decomposition:

$$K_{LH}(m) = \begin{bmatrix} 0 & K_{L\Gamma}(m) \end{bmatrix} = K_{HL}(m)^T. \quad (22)$$

As we prepare for the proof of our main Lemma, we need to define the following quantity which will also be used for subdomain deflation in (40):

$$\eta(m) := e_H^T K_{HH}(m) e_H. \quad (23)$$

$\eta(m) > 0$  because  $K_{HH}(m)$  is SPD as a diagonal subblock of  $K(m)$ . Moreover, combining (19) and (20), one can reduce the expression in (23) to

$$\eta(m) = \bar{h}_m e_\Gamma^T D_{\Gamma\Gamma}^{(L)} e_\Gamma. \quad (24)$$

In fact, (24) can be expressed explicitly by:

$$\eta(m) = \bar{h}_m \frac{n_{H,nc} + n_{H,c}}{n_H}, \quad (25)$$

where

$$n_{H,nc} := \#\{\text{non-corner interface cell-centers}\}$$

$$n_{H,c} := \#\{\text{corner interface cell-centers}\}.$$

Finally, (25) delivers the asymptotic convergence expression for  $\eta(m)$ :

$$\eta(m) = \eta + \mathcal{O}(m^{-1}), \quad (26)$$

$$\text{where } \eta = 2 \frac{n_{H,nc} + n_{H,c}}{n_H}.$$

## 5.2 The main results on the preconditioner

**Lemma 1** *The asymptotic behaviour of the submatrices in (5) is described by the following:*

- (i)  $K_{HH}(m)^{-1} = e_H \eta^{-1} e_H^T + \mathcal{O}(m^{-1})$ ,
- (ii)  $S(m) = K_{LL}^\infty - (K_{L\Gamma}^\infty e_\Gamma) \eta^{-1} (e_\Gamma^T K_{\Gamma L}^\infty) + \mathcal{O}(m^{-1})$ ,
- (iii)  $K_{LH}(m) K_{HH}(m)^{-1} = (K_{L\Gamma}^\infty e_\Gamma) \eta^{-1} e_H^T + \mathcal{O}(m^{-1})$ .

*Proof* Since  $\mathcal{N}_{HH}$  is symmetric positive semidefinite we have the eigenvalue decomposition:

$$Z^T \mathcal{N}_{HH} Z = \text{diag}(\lambda_1, \dots, \lambda_{n_H-1}, 0), \quad (27)$$

where  $\{\lambda_i : i = 1, \dots, n_H\}$  is a non-increasing sequence of eigenvalues of  $\mathcal{N}_{HH}$  and  $Z$  is orthogonal. Since, the eigenvector corresponding to the zero eigenvalue is constant, we can write  $Z = [\tilde{Z} \mid e_H]$  and so, using (19), we have

$$\begin{aligned} & Z^T K_{HH}(m) Z \\ &= \begin{bmatrix} m \text{diag}(\lambda_1, \dots, \lambda_{n_H-1}) + \tilde{Z}^T \Delta(m) \tilde{Z} & \tilde{Z}^T \Delta(m) e_H \\ e_H^T \Delta(m) \tilde{Z} & e_H^T \Delta(m) e_H \end{bmatrix} \\ &=: \begin{bmatrix} \tilde{\Lambda}(m) & \tilde{\delta}(m) \\ \tilde{\delta}^T(m) & \eta(m) \end{bmatrix}. \end{aligned} \quad (28)$$

To find the limiting form of  $K_{HH}(m)^{-1}$  note that

$$\begin{aligned} \tilde{\Lambda}(m) &= m \text{diag}(\lambda_1, \dots, \lambda_{n_H-1}) + \tilde{Z}^T \Delta(m) \tilde{Z} \\ &= m \text{diag}(\lambda_1, \dots, \lambda_{n_H-1}) \\ &\quad \times \left( \tilde{I} + m^{-1} \text{diag}(\lambda_1^{-1}, \dots, \lambda_{n_H-1}^{-1}) \tilde{Z}^T \Delta(m) \tilde{Z} \right). \end{aligned}$$

Now, let  $C_\lambda := \max_{i < n_H} \lambda_i^{-1}$  and  $C_\Delta$  be the constant in (21), i.e.,  $\|\Delta(m)\|_2 \leq \|\Delta^\infty\|_2 + C_\Delta m^{-1}$ . Then, for sufficiently large  $m$ ,

$$\begin{aligned} \|\tilde{\Lambda}(m)^{-1}\|_2 &\leq \frac{m^{-1} C_\lambda}{1 - m^{-1} C_\lambda \|\tilde{Z}^T \Delta(m) \tilde{Z}\|_2} \\ &\leq \frac{m^{-1} C_\lambda}{C_{\tilde{\Lambda}}} \end{aligned} \quad (29)$$

where

$$\begin{aligned} C_{\tilde{\Lambda}} &:= 1 - m^{-1} C_{\lambda} \|\tilde{Z}^T\|_2 \|\Delta^\infty\|_2 \|\tilde{Z}\|_2 \\ &\quad - m^{-2} C_{\Delta} C_{\lambda} \|\tilde{Z}^T\|_2 \|\tilde{Z}\|_2 \\ &= 1 + \mathcal{O}(m^{-1}). \end{aligned} \quad (30)$$

Hence, combining (29) and (30), we obtain:

$$\tilde{\Lambda}(m)^{-1} = \mathcal{O}(m^{-1}). \quad (31)$$

We proceed with the following inversion:

$$\begin{bmatrix} \tilde{\Lambda}(m) & \tilde{\delta}(m) \\ \tilde{\delta}(m)^T & \eta(m) \end{bmatrix}^{-1} = U(m) V(m) U(m)^T,$$

where

$$\begin{aligned} U(m) &:= \begin{bmatrix} \tilde{I} & -\tilde{\Lambda}(m)^{-1} \tilde{\delta}(m) \\ 0^T & 1 \end{bmatrix}, \\ V(m) &:= \begin{bmatrix} \tilde{\Lambda}(m)^{-1} & 0 \\ 0^T & \left( \eta(m) - \tilde{\delta}(m)^T \tilde{\Lambda}(m)^{-1} \tilde{\delta}(m) \right)^{-1} \end{bmatrix}. \end{aligned}$$

Using (21), one gets

$$\tilde{\delta}(m) = \tilde{Z}^T \Delta^\infty e_H + \mathcal{O}(m^{-1}). \quad (32)$$

Then, (26), (31), and (32) imply that

$$\begin{aligned} \eta(m)^{-1} &= \eta^{-1} + \mathcal{O}(m^{-1}), \\ U(m) &= I + \mathcal{O}(m^{-1}), \\ V(m) &= \begin{bmatrix} O & 0 \\ 0^T & \eta^{-1} \end{bmatrix} + \mathcal{O}(m^{-1}). \end{aligned}$$

Combining the above results, we arrive at

$$\begin{bmatrix} \tilde{\Lambda}(m) & \tilde{\delta}(m) \\ \tilde{\delta}(m)^T & \eta(m) \end{bmatrix}^{-1} = \begin{bmatrix} O & 0 \\ 0^T & \eta^{-1} \end{bmatrix} + \mathcal{O}(m^{-1}), \quad (33)$$

and, by (28), we have

$$\begin{aligned} K_{HH}(m)^{-1} &= Z \begin{bmatrix} O & 0 \\ 0^T & \eta^{-1} \end{bmatrix} Z^T + \mathcal{O}(m^{-1}) \\ &= e_H \left( e_\Gamma^T 2D_{\Gamma\Gamma}^{(L)} e_\Gamma \right)^{-1} e_H^T + \mathcal{O}(m^{-1}), \end{aligned}$$

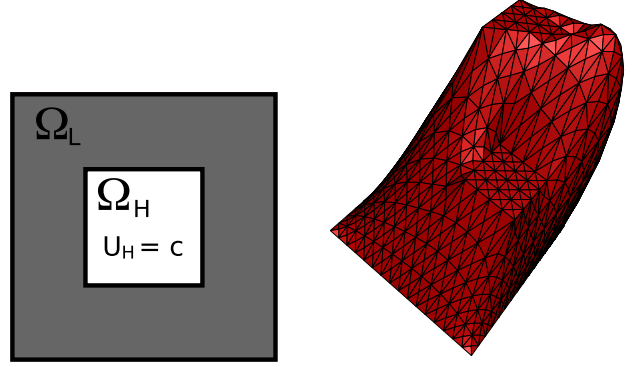
which proves part (i) of the Lemma.

Parts (ii) and (iii) follow from simple substitution, using (6) and (22).  $\square$

We use the following limiting forms in the definition of the proposed preconditioner (36):

$$K_{HH}^{\infty\dagger} := e_H \eta^{-1} e_H^T, \quad (34)$$

$$\begin{aligned} Q_{LH}^\infty &:= K_{LH}^\infty K_{HH}^{\infty\dagger}, \\ S^\infty &:= K_{LL}^\infty - K_{LH}^\infty K_{HH}^{\infty\dagger} K_{HL}^\infty. \end{aligned} \quad (35)$$



**Fig. 3** (Left) The matrix in (38) corresponds to a homogeneous Dirichlet for the Laplacian on  $\Omega$  under the constraint that the solution is constant on  $\Omega_H$ . (Right)  $x_{H_i}(m_i) = c_{H_i} + \mathcal{O}(m_i^{-1})$ ,  $i = 1, 2$  where  $\Omega_{H_1}$  and  $\Omega_{H_2}$  correspond to square and triangle shaped highly-diffusive islands, respectively with  $m_i = 10^6$ .

Based on the above perturbation analysis we propose the following preconditioner:

$$\begin{aligned} B_{AGKS}(m) &:= \begin{bmatrix} I_{HH} & -Q_{LH}^{\infty T} \\ 0 & I_{LL} \end{bmatrix} \begin{bmatrix} K_{HH}(m)^{-1} & 0 \\ 0 & S^{\infty^{-1}} \end{bmatrix} \\ &\quad \times \begin{bmatrix} I_{HH} & 0 \\ -Q_{LH}^\infty & I_{LL} \end{bmatrix}. \end{aligned} \quad (36)$$

The following theorem shows that  $B_{AGKS}$  is an effective preconditioner for  $m \gg 1$ .

**Theorem 1** For  $m$  sufficiently large we have

$$\sigma(B_{AGKS}(m) K(m)) \subset [1 - cm^{-1/2}, 1 + cm^{-1/2}]$$

for some constant  $c$  independent of  $m$ , and therefore

$$\kappa(B_{AGKS}(m) K(m)) = 1 + \mathcal{O}(m^{-1/2}).$$

*Proof* The  $m$ -dependence of  $K_{HL}(m)$ ,  $K_{LH}(m)$ , and  $K_{LL}(m)$  are eliminated asymptotically as in (16). Furthermore,  $m$ -dependence of  $\eta(m)$  is also eliminated as a result of (21). Therefore, the result follows by slightly modifying the proof, i.e., by incorporating the  $m$ -dependencies of the above entities, given in Aksoylu et al. [7] for the FE case.  $\square$

## 6 Qualitative nature of the solution of the high-contrast diffusion equation and decoupling

We advocate the usage of SPA because it is a very effective tool in gaining qualitative insight about the asymptotic behavior of the solution of the underlying PDE. Through SPA, in Lemma 1, we were able to fully reveal the asymptotic behaviour of the submatrices of  $K$  in (5). This information leads to a characterization of the limit of the underlying discretized inverse operator and we studied this in more detail in [4]. We now prove that

asymptotically, the solution over the highly-diffusive island goes to a constant vector. This is probably the most fundamental qualitative feature of the solution of the high-contrast diffusion equation.

**Lemma 2** *Let  $x_H(m)$  be the solution in (4) corresponding the highly-diffusive island and  $e_H$  be the constant vector. Then,*

$$x_H(m) = c_H e_H + \mathcal{O}(m^{-1}), \quad (37)$$

where  $c_H$  is determined by the solution in the lowly-diffusive region.

*Proof* We prove the result by providing an explicit quantification of the limiting process based on Lemma 1:

$$\begin{aligned} x_L(m) &= S^{-1}(m) \{b_L - K_{LH}(m)K_{HH}^{-1}(m)b_H\} \\ &= S^{\infty-1} \{b_L - K_{LH}^{\infty}K_{HH}^{\infty\dagger}b_H\} + \mathcal{O}(m^{-1}) \\ &=: x_L^{\infty} + \mathcal{O}(m^{-1}), \\ x_H(m) &= K_{HH}^{-1}(m) \{b_H - K_{HL}(m)x_L(m)\} \\ &= K_{HH}^{\infty\dagger} \{b_H - K_{HL}^{\infty}x_L^{\infty}\} + \mathcal{O}(m^{-1}) \\ &=: c_H e_H + \mathcal{O}(m^{-1}). \end{aligned}$$

□

SPA helps to reveal a further qualitative property, namely, the decoupling phenomenon. We show how the original algebraic solution strategy decouples into two algebraic problems of different nature and the decoupling is indeed an implication of (37). This observation would be very valuable also for designing preconditioners for different classes of PDEs and discretizations. In [9], the authors gave a solution behaviour result similar to (37) for the high-contrast biharmonic plate equation for HCT and Morley discretizations.

In order to show the decoupling, let us start by noting that  $S^{\infty}$  in Lemma 1 can also be interpreted as the Schur complement of  $c^2 e_{\Gamma}^T K_{\Gamma\Gamma}^{(L)} e_{\Gamma}$  in the matrix

$$KK_{LL}^{\infty} = \begin{bmatrix} c^2 e_{\Gamma}^T K_{\Gamma\Gamma}^{(L)} e_{\Gamma} & c e_{\Gamma}^T K_{\Gamma L} \\ c K_{L\Gamma} e_{\Gamma} & K_{LL} \end{bmatrix},$$

for any nonzero value of  $c$ . In particular, if we choose  $c := n_H^{1/2}$ , then  $ce_{\Gamma} = 1_{\Gamma}$ , the vector of all ones on  $\Gamma$  and, using also (19), we have

$$\begin{aligned} KK_{LL}^{\infty} &:= \begin{bmatrix} 1_{\Gamma}^T K_{\Gamma\Gamma}^{(L)} 1_{\Gamma} & 1_{\Gamma}^T K_{\Gamma L} \\ K_{L\Gamma} 1_{\Gamma} & K_{LL} \end{bmatrix} \\ &= \begin{bmatrix} 1_H^T K_{HH}(1) 1_H & 1_H^T K_{HL} \\ K_{LH} 1_H & K_{LL} \end{bmatrix}. \end{aligned} \quad (38)$$

This is the stiffness matrix for a pure Dirichlet problem for the Laplacian on all of  $\Omega$  with the additional constraint that the solution is constant on  $\Omega_H$ . See Figure 3. Thus, when  $m \gg 1$ , the original problem *decouples almost entirely into a (regularized) Neumann problem* (i.e.  $K_{HH}(m)$ ) for the Laplacian on  $\Omega_H$  (scaled by

$m$ ) and a Dirichlet problem (i.e.  $KK_{LL}^{\infty}$ ) for the Laplacian on all of  $\Omega$ , but under the additional constraint that the solution is constant on  $\Omega_H$ .

Next, we show an additional decoupling. This time the preconditioner decouples into a block diagonal matrix with the help of a deflation method.

## 7 Implementation aspects and the related deflation method

The fact that  $\mathcal{N}_{HH}$  has a simple zero eigenvalue (with the corresponding constant eigenvector  $e_H$ ) and  $\mathcal{N}_{HH}$  is of co-rank 1 imply that  $K_{HH}(m)$  has a single eigenvalue of  $\mathcal{O}(1)$  and  $n_H - 1$  eigenvalues of  $\mathcal{O}(m)$ . For sufficiently large  $m$ ,  $e_H$  can be taken as an approximate eigenvector corresponding to  $\mathcal{N}_{HH}$ 's single smallest eigenvalue. Therefore, in the light of (5),

$$e^T := [e_H^T, 0^T],$$

becomes an approximate eigenvector corresponding to the smallest eigenvalue of the decoupled matrix:

$$\begin{bmatrix} K_{HH}(m) & 0 \\ 0 & S(m) \end{bmatrix}. \quad (39)$$

In order to eliminate the negative effect of the smallest eigenvalue, we utilize a deflation method under the constraint that it gives the decoupling as in (39). If such decoupling occurs, it would provide a large computational advantage. We utilize a deflation method, known as *subdomain deflation* [23, 27, 28, 29] constructed by the following  $K$ -orthogonal projection onto the subspace orthogonal to  $e$ , provides the desired decoupling.

$$\mathcal{P}^T := I - e\eta(m)^{-1}e^T K. \quad (40)$$

We apply  $B_{AGKS}$  within a conjugate gradient algorithm for the deflated system

$$\mathcal{P}Kx^{\perp} = \mathcal{P}b, \quad (41)$$

where  $x^{\perp} := \mathcal{P}^T x$  is the projected solution. The component of  $x$  in the direction of  $e$  is then simply given by

$$x - x^{\perp} = (I - \mathcal{P}^T)x = e\eta(m)^{-1}e^T b.$$

When we rewrite (36) as

$$B_{AGKS}(m) = L^T D(m) L,$$

where

$$\begin{aligned} L &:= \begin{bmatrix} I_{HH} & 0 \\ -Q_{LH}^{\infty} & I_{LL} \end{bmatrix}, \\ D(m) &:= \begin{bmatrix} K_{HH}(m)^{-1} & 0 \\ 0 & S^{\infty-1} \end{bmatrix}. \end{aligned} \quad (42)$$

By observing the following interesting property

$$L\mathcal{P} = \mathcal{P}, \quad (43)$$

the system in (41) after preconditioned by  $B_{AGKS}(m)$  turns into:

$$B_{AGKS}(m) (\mathcal{P}Kx^\perp) = B_{AGKS}(m) (\mathcal{P}b).$$

Then, it reduces to the following block diagonal system:

$$D(m) \mathcal{P}Kx^\perp = D(m) \mathcal{P}b.$$

*Remark 3* The fact that  $e$  becomes an approximate eigenvector corresponding to the smallest eigenvalue of the decoupled matrix in (39) is not necessarily true for the original matrix  $K(m)$  because it is not block diagonal. Therefore, utilizing the above deflation method in the direction of  $e$  will not necessarily provide any further robustness for the underlying preconditioner if that preconditioner uses off-diagonal blocks of  $K(m)$  as well. Multigrid method is one such preconditioner. In fact, numerically we observed that introducing deflation did not improve the multigrid convergence rate at all. If one still wants to improve the convergence rate by the help of deflation, then the approximate eigenvector corresponding to the smallest eigenvalue must be computed in an alternative way rather than the simple usage of  $e$ . Consequently, we can say that  $B_{AGKS}(m)$ , by its design, naturally goes with subdomain deflation. The incorporation of subdomain deflation not only brings robustness with respect to the smallest eigenvalue, but also provides huge computational savings due to reduction to a block diagonal system. This is a significant computational advantage  $B_{AGKS}(m)$  offers.

By exploiting the fact that  $S^\infty$  in (35) is only a low-rank perturbation of  $K^\infty$ , we can build robust preconditioners for  $S^\infty$  in (36) via standard multigrid preconditioners. Combining (34) and (35), we arrive at

$$S^\infty = K_{LL}^\infty - v\eta^{-1}v^T,$$

where  $v := K_{LH}^\infty e_H$ . If  $M_{LL}$  denotes a standard multigrid V-cycle preconditioner for  $K_{LL}$ , we can construct an efficient and robust preconditioner  $\tilde{S}^{-1}$  for  $S^\infty$  using the Sherman-Morrison-Woodbury formula [13, Eq. (2.1.4)], i.e.

$$\tilde{S}^{-1} := M_{LL} + M_{LL}v(1-\eta)^{-1}v^T M_{LL}. \quad (44)$$

Note also that we can precompute and store  $M_{LL}v$  during the setup phase. This means we only need to apply

the multigrid V-cycle  $M_{LL}$  once per iteration. Therefore, the following practical version of preconditioner (36) is used in the implementation:

$$\begin{aligned} \tilde{B}_{AGKS} := & \begin{bmatrix} I_{HH} & -Q_{LH}^{\infty T} \\ 0 & I_{LL} \end{bmatrix} \begin{bmatrix} M_{HH} & 0 \\ 0 & \tilde{S}^{-1} \end{bmatrix} \\ & \times \begin{bmatrix} I_{HH} & 0 \\ -Q_{LH}^\infty & I_{LL} \end{bmatrix}, \end{aligned} \quad (45)$$

where  $M_{HH}$  denotes a multigrid V-cycle preconditioner for  $K_{HH}(m)$ .

## 8 Numerical experiments

The goal of the numerical experiments is to compare the performance of the two preconditioners: AGKS and CCMG. The domain is unit square with a uniform mesh consisting of  $2^\ell \times 2^\ell$ ,  $\ell = 3, \dots, 6$ , cells. The coarsest level mesh contains  $8 \times 8$  cells with a highly-diffusive single island of size  $2 \times 2$  cells centered at the point  $(3/8, 3/8)$ . For the discussion of multiple disconnected islands, refer to [7, Sections 3 and 4].

We denote the norm of the relative residual at iteration  $t$  by  $rr^{(t)}$ :

$$rr^{(t)} := \frac{\|r^{(t)}\|_2}{\|r^{(0)}\|_2},$$

where  $r^{(t)}$  denotes the residual at iteration  $t$  with a stopping criterion of  $rr^{(t)} \leq 10^{-9}$ . In the tables, we report the iteration count and the average reduction factor of the residual which is defined as:

$$\left(rr^{(t)}\right)^{1/t}.$$

We enforce an iteration bound of 60. If the method seems to converge slightly beyond this bound, we denote it by  $60^+$ . Whereas, the divergence is denoted by  $\infty$ . In Tables 2–10, iteration count and the average reduction factor are reported for combinations of preconditioner, prolongation, and smoother types.

We use Galerkin variational approach to construct the coarser level algebraic systems. There are two types of prolongation operators under consideration; Wesseling-Khalil [32] and bi-linear, given by respectively:

**Table 2** Preconditioner = CCMG - V(1,1)-cycle, Prolongation = Wesseling-Khalil, Smoother = ILU

$h \backslash m$	$10^0$	$10^1$	$10^2$	$10^3$	$10^4$	$10^5$	$10^6$	$10^7$	$10^8$	$10^9$
1/8	<b>10</b> , 0.025	<b>29</b> , 0.472	<b>30</b> , 0.500	<b>35</b> , 0.528	<b>36</b> , 0.559	<b>40</b> , 0.593	<b>44</b> , 0.622	<b>52</b> , 0.669	<b>52</b> , 0.669	<b>58</b> , 0.697
1/16	<b>10</b> , 0.121	<b>12</b> , 0.165	<b>15</b> , 0.240	<b>21</b> , 0.315	<b>31</b> , 0.481	<b>38</b> , 0.541	<b>50</b> , 0.644	<b>60</b> <sup>+</sup> , 0.786	<b>60</b> <sup>+</sup> , 0.770	<b>60</b> <sup>+</sup> , 0.922
1/32	<b>8</b> , 0.069	<b>10</b> , 0.093	<b>12</b> , 0.162	<b>14</b> , 0.224	<b>19</b> , 0.309	<b>27</b> , 0.455	<b>52</b> , 0.668	<b>60</b> <sup>+</sup> , 0.829	<b>60</b> <sup>+</sup> , 1.002	<b>60</b> <sup>+</sup> , 1.033
1/64	<b>8</b> , 0.064	<b>10</b> , 0.099	<b>13</b> , 0.160	<b>14</b> , 0.201	<b>19</b> , 0.311	<b>29</b> , 0.462	<b>47</b> , 0.638	<b>60</b> <sup>+</sup> , 0.860	<b>60</b> <sup>+</sup> , 1.014	<b>60</b> <sup>+</sup> , 1.033

**Table 3** Preconditioner = CCMG - V(1,1)-cycle, Prolongation = Wesseling-Khalil, Smoother = sGS

$h \backslash m$	$10^0$	$10^1$	$10^2$	$10^3$	$10^4$	$10^5$	$10^6$	$10^7$	$10^8$	$10^9$
1/8	<b>10</b> , 0.025	<b>29</b> , 0.472	<b>30</b> , 0.500	<b>35</b> , 0.528	<b>36</b> , 0.559	<b>40</b> , 0.593	<b>44</b> , 0.622	<b>52</b> , 0.669	<b>52</b> , 0.669	<b>58</b> , 0.697
1/16	<b>12</b> , 0.166	<b>14</b> , 0.222	<b>19</b> , 0.310	<b>26</b> , 0.409	<b>38</b> , 0.535	<b>52</b> , 0.649	<b>60</b> <sup>+</sup> , 0.777	<b>60</b> <sup>+</sup> , 0.843	<b>60</b> <sup>+</sup> , 0.938	$\infty$ , 1.070
1/32	<b>14</b> , 0.195	<b>15</b> , 0.217	<b>18</b> , 0.294	<b>25</b> , 0.432	<b>39</b> , 0.552	<b>58</b> , 0.698	<b>60</b> <sup>+</sup> , 0.917	$\infty$ , 1.002	$\infty$ , 1.080	$\infty$ , 1.127
1/64	<b>13</b> , 0.197	<b>14</b> , 0.221	<b>17</b> , 0.282	<b>22</b> , 0.362	<b>31</b> , 0.497	<b>48</b> , 0.645	<b>60</b> <sup>+</sup> , 0.793	$\infty$ , 0.954	$\infty$ , 1.097	$\infty$ , 1.120

**Table 4** Preconditioner = CCMG - V(1,1)-cycle, Prolongation = Bi-linear, Smoother = ILU

$h \backslash m$	$10^0$	$10^1$	$10^2$	$10^3$	$10^4$	$10^5$	$10^6$	$10^7$	$10^8$	$10^9$
1/8	<b>10</b> , 0.025	<b>29</b> , 0.472	<b>30</b> , 0.500	<b>35</b> , 0.528	<b>36</b> , 0.559	<b>40</b> , 0.593	<b>44</b> , 0.622	<b>52</b> , 0.669	<b>52</b> , 0.669	<b>58</b> , 0.697
1/16	<b>9</b> , 0.095	<b>12</b> , 0.145	<b>14</b> , 0.225	<b>23</b> , 0.403	<b>32</b> , 0.514	<b>48</b> , 0.637	<b>59</b> , 0.693	<b>60</b> <sup>+</sup> , 0.834	<b>60</b> <sup>+</sup> , 0.924	$\infty$ , 0.990
1/32	<b>7</b> , 0.040	<b>9</b> , 0.074	<b>12</b> , 0.154	<b>60</b> <sup>+</sup> , 1.051	$\infty$ , 1.001	$\infty$ , 1.001	$\infty$ , 1.001	$\infty$ , 1.001	$\infty$ , 1.001	$\infty$ , 1.001
1/64	<b>7</b> , 0.039	<b>8</b> , 0.075	<b>11</b> , 0.127	<b>21</b> , 0.350	$\infty$ , 1.001	$\infty$ , 1.001	$\infty$ , 1.001	$\infty$ , 1.001	$\infty$ , 1.001	$\infty$ , 1.001

$$P^{(WK)} = \frac{1}{4} \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 3 & 2 & 0 \\ & & \cdot & \\ 0 & 2 & 3 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix}_{2h}^h, \quad \text{and} \quad R^{(WK)} = P^{(WK)*},$$

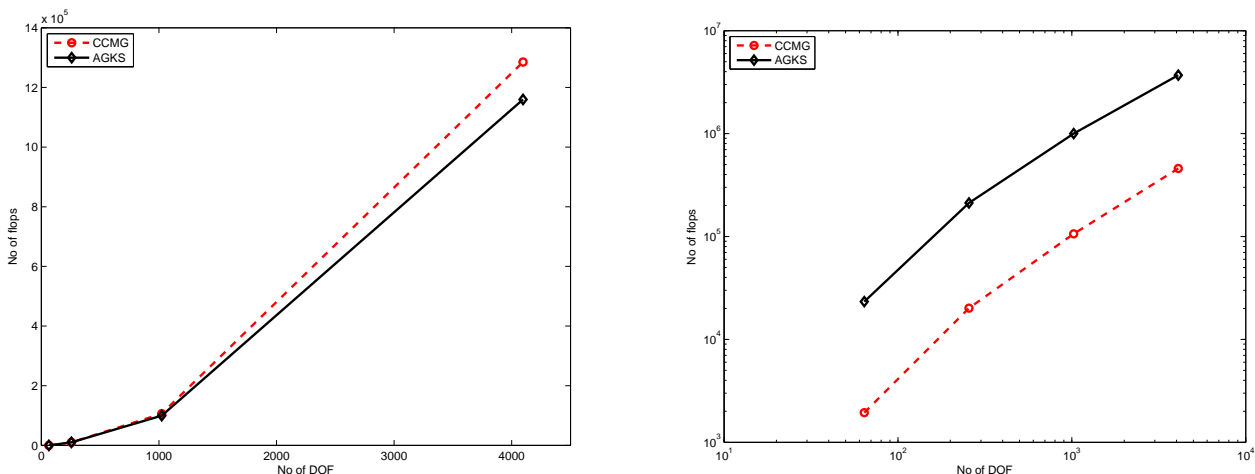
$$P^{(B)} = \frac{1}{16} \begin{bmatrix} 1 & 3 & 3 & 1 \\ 3 & 9 & 9 & 3 \\ & & \cdot & \\ 3 & 9 & 9 & 3 \\ 1 & 3 & 3 & 1 \end{bmatrix}_{2h}^h, \quad \text{and} \quad R^{(B)} = P^{(B)*}.$$

The multigrid preconditioner CCMG is derived from the implementation by Aksoylu, Bond, and Holst [6]. We employ a V(1,1)-cycle with point symmetric Gauss-Seidel (sGS) or ILU smoothers. A direct solver is used for the coarsest level. We construct two different multi-

level hierarchies for multigrid preconditioners  $M_{HH}$  in (45) and  $M_{LL}$  in (44) for DOF corresponding to  $\Omega_H$  and  $\Omega_L$ , respectively. The corresponding prolongation matrices  $P_{HH}$  and  $P_{LL}$  are extracted from the prolongation matrix for the whole domain  $\Omega$  in the fashion following (4):

$$P = \begin{bmatrix} P_{HH} & P_{HL} \\ P_{LH} & P_{LL} \end{bmatrix}. \quad (46)$$

The superior performance of AGKS preconditioner is partially due to employing these two distinct multilevel hierarchies, which is very much in spirit of the aforementioned decoupling in Section 7. In fact, due to decoupling, AGKS technology allows the usage of any ordinary prolongation operator. This operator does not have to be constructed in a sophisticated manner as in the case of Wesseling and Khalil [32] or Kwak [18].



**Fig. 4** (Left) Flop counts for the enforcement of variational conditions. (Right) Flop counts for a single iteration of the preconditioners.

As emphasized in our preceding paper [7], AGKS can be used purely as an algebraic preconditioner. Therefore, the standard multigrid preconditioner constraint that the coarsest level mesh resolves the boundary of the island is automatically eliminated. However, for a fair comparison, we enforce the coarsest level mesh to have that property.

When the discretization matrix is scaled by  $1/m$ , we observe a significant reduction in the iteration count for the AGKS preconditioner, while, CCMG suffers from inconsistent convergence behaviour. That is why, we only report the unscaled case for CCMG. Moreover, for the CCMG preconditioner, we use lexicographic ordering. On the other hand, for AGKS, we follow the standard way of ordering the highly-diffusive after the lowly-diffusive DOF as used in [7].

Note that as the diffusion coefficient  $m$  increases, the CCMG method becomes less effective. For sufficiently large  $m$ , it even diverges. CCMG shows this adverse behaviour for all types of transfer operators, and smoother types for almost all levels; see Tables 2–5. The CCMG preconditioner is more effective than the AGKS preconditioner for  $m < 10^3$ . However, for those  $m$  values, the AGKS preconditioner still manages to converge in less than 30 iterations.

The main issue addressed in our study is the robustness for large values of  $m$ . We observe that the CCMG preconditioner becomes ineffective, even diverges for  $m \geq 10^6$  under both types of smoothers; see the corresponding columns in Tables 2–5. On the other hand, the AGKS preconditioner becomes more effective with increasing  $m$ . Furthermore, as seen in Tables 7–10, AGKS preconditioner demonstrates consistently similar iteration counts and contraction numbers for all types of transfer operators and smoothers. Therefore, AGKS per-

formance is independent of the utilized prolongation operators and smoothers.

CCMG performance heavily depends on the smoother choice. It is well-known that pattern relaxations would not improve CCMG performance [20, p. 112]. In addition, since there is no anisotropy, the line smoothers also do not improve CCMG performance compared to sGS and ILU smoothers; see Llorente and Melson [20, p. 112] for other ordering related smoother complications. Hence, we used neither pattern relaxations nor line smoothers in our experiments. As pointed out by Mohr and Wienands [22], CCMG may need a sophisticated smoother like ILU. The behaviour of ILU was extensively studied by Wittum [33,34], also see [30, pp. 98 and 134]. In our experiments, the specific ILU choice is set to be no-fill-in.

We observe an interesting cut-off  $m$  value for performances of preconditioners. While in general, the CCMG performance starts to deteriorate at around  $m > 10^5$ , the AGKS preconditioner reaches its peak performance and maintains an optimal iteration count for  $m \geq 10^5$ . We reveal several observations. The CCMG preconditioner with sGS smoothing shows an adverse convergence behaviour for all prolongation types. On the other hand, the CCMG preconditioner with ILU smoothing shows a higher performance with Wesseling-Khalil prolongation type. The iteration counts for the CCMG method jumps to 60 at around  $m = 10^5$  and  $m = 10^6$  for sGS and ILU smoothing cases, respectively.

The CCMG preconditioner has been of type V-cycle so far. After observing convergence complications, we decided to try W(1,1)-cycle as an alternative; see the W(1,1)-cycle convergence history in Table 6. W(1,1)-cycle does not show a consistent convergence with respect to the contrast size. For instance, we observe a

**Table 5** Preconditioner = CCMG - V(1,1)-cycle, Prolongation = Bi-linear, Smoother = sGS

$h \backslash m$	$10^0$	$10^1$	$10^2$	$10^3$	$10^4$	$10^5$	$10^6$	$10^7$	$10^8$	$10^9$
1/8	<b>10</b> , 0.025	<b>29</b> , 0.472	<b>30</b> , 0.500	<b>35</b> , 0.528	<b>36</b> , 0.559	<b>40</b> , 0.593	<b>44</b> , 0.622	<b>52</b> , 0.669	<b>52</b> , 0.669	<b>58</b> , 0.697
1/16	<b>11</b> , 0.122	<b>13</b> , 0.194	<b>18</b> , 0.293	<b>24</b> , 0.414	<b>38</b> , 0.564	<b>58</b> , 0.692	<b>60</b> <sup>+</sup> , 0.827	<b>60</b> <sup>+</sup> , 0.931	$\infty$ , 0.995	$\infty$ , 1.094
1/32	<b>8</b> , 0.075	<b>11</b> , 0.121	<b>15</b> , 0.229	<b>22</b> , 0.362	<b>35</b> , 0.545	<b>60</b> <sup>+</sup> , 0.722	<b>60</b> <sup>+</sup> , 0.903	$\infty$ , 1.022	$\infty$ , 1.085	$\infty$ , 1.133
1/64	<b>8</b> , 0.068	<b>10</b> , 0.115	<b>14</b> , 0.216	<b>19</b> , 0.334	<b>27</b> , 0.449	<b>44</b> , 0.600	<b>60</b> <sup>+</sup> , 0.773	$\infty$ , 0.965	$\infty$ , 1.116	$\infty$ , 1.168

**Table 6** Preconditioner = CCMG - W(1,1)-cycle, Prolongation = Wesseling-Khalil, Smoother = ILU

$h \backslash m$	$10^0$	$10^1$	$10^2$	$10^3$	$10^4$	$10^5$	$10^6$	$10^7$	$10^8$	$10^9$
1/8	<b>10</b> , 0.025	<b>29</b> , 0.472	<b>30</b> , 0.500	<b>35</b> , 0.528	<b>36</b> , 0.559	<b>40</b> , 0.593	<b>44</b> , 0.622	<b>52</b> , 0.669	<b>52</b> , 0.669	<b>58</b> , 0.697
1/16	<b>10</b> , 0.121	<b>12</b> , 0.166	<b>15</b> , 0.245	<b>21</b> , 0.315	<b>31</b> , 0.481	<b>38</b> , 0.541	<b>50</b> , 0.644	<b>60</b> <sup>+</sup> , 0.786	<b>60</b> <sup>+</sup> , 0.769	<b>60</b> <sup>+</sup> , 922
1/32	<b>8</b> , 0.070	<b>12</b> , 0.165	<b>17</b> , 0.263	<b>19</b> , 0.328	$\infty$ , 0.968	$\infty$ , 1.110	$\infty$ , 1.127	$\infty$ , 1.164	$\infty$ , 1.169	$\infty$ , 1.184
1/64	<b>8</b> , 0.064	<b>12</b> , 0.137	<b>16</b> , 0.267	<b>18</b> , 0.305	<b>20</b> , 0.345	<b>32</b> , 0.520	$\infty$ , 1.115	$\infty$ , 1.137	$\infty$ , 1.190	$\infty$ , 1.199

**Table 7** Preconditioner = AGKS - V(1,1)-cycle, Prolongation = Wesseling-Khalil, Smoother = ILU

$h \backslash m$	$10^0$	$10^1$	$10^2$	$10^3$	$10^4$	$10^5$	$10^6$	$10^7$	$10^8$	$10^9$	$10^{11}$	$10^{13}$
1/8	<b>22</b> , 0.371	<b>10</b> , 0.115	<b>10</b> , 0.116	<b>9</b> , 0.078	<b>9</b> , 0.056	<b>8</b> , 0.059	<b>8</b> , 0.045	<b>8</b> , 0.039	<b>8</b> , 0.039	<b>8</b> , 0.039	<b>8</b> , 0.039	<b>8</b> , 0.039
1/16	<b>16</b> , 0.240	<b>13</b> , 0.199	<b>11</b> , 0.131	<b>9</b> , 0.098	<b>8</b> , 0.043	<b>7</b> , 0.043	<b>6</b> , 0.018	<b>6</b> , 0.011	<b>6</b> , 0.007	<b>6</b> , 0.010	<b>5</b> , 0.010	<b>5</b> , 0.006
1/32	<b>20</b> , 0.329	<b>19</b> , 0.313	<b>14</b> , 0.192	<b>11</b> , 0.127	<b>9</b> , 0.093	<b>8</b> , 0.035	<b>7</b> , 0.041	<b>6</b> , 0.018	<b>6</b> , 0.012	<b>6</b> , 0.008	<b>6</b> , 0.004	<b>5</b> , 0.008
1/64	<b>29</b> , 0.484	<b>26</b> , 0.435	<b>17</b> , 0.284	<b>12</b> , 0.161	<b>10</b> , 0.092	<b>8</b> , 0.061	<b>8</b> , 0.026	<b>6</b> , 0.031	<b>6</b> , 0.016	<b>6</b> , 0.010	<b>6</b> , 0.006	<b>5</b> , 0.013

jump in the iteration count at around  $m = 10^4$  and  $m = 10^6$  for levels 3 and 4, respectively. Therefore, the performance of the CCMG preconditioners gets worse for  $m > 10^5$  independently from the cycle type. Consequently, this observation indicates that CCMG fails to be robust with respect to the contrast size whereas AGKS maintains its robustness.

We conclude the numerical experiments by reporting the cost of each preconditioner. For variational conditions, the decoupling of  $K_{HH}(m)$  and  $S^\infty$  in (42) causes the AGKS preconditioner to be cheaper than CCMG because it only employs  $P_{HH}$  and  $P_{LL}$  blocks in (46) whereas CCMG employs the whole  $P$  matrix; see the flop counts in Figure 4. When the size of the highly-diffusive island grows, the enforcement of the variational conditions of the AGKS preconditioner be-

comes even less costly than that of the CCMG preconditioner.

Finally, we report the cost per iteration for AGKS and CCMG V(1,1)-cycle preconditioners. AGKS preconditioner in (42) requires inversions of two blocks:  $K_{HH}(m)$  and  $S^\infty$  corresponding to highly- and lowly-diffusive regions, respectively. Therefore, for each iteration of AGKS preconditioner, we utilize a full CCMG method for each block separately. This is exactly the setup that CCMG methods are known to be highly effective because each block corresponds to a discretization of the Laplace equation with homogeneous coefficients. Therefore, one iteration of the AGKS preconditioner is roughly 10 times more costly than that of the CCMG preconditioner; see the flop counts in Figure 4. This additional cost is worthy because AGKS preconditioner results in convergence in a few iterations for

**Table 8** Preconditioner = AGKS - V(1,1)-cycle, Prolongation = Wesseling-Khalil, Smoother = sGS

$h \setminus m$	$10^0$	$10^1$	$10^2$	$10^3$	$10^4$	$10^5$	$10^6$	$10^7$	$10^8$	$10^9$	$10^{11}$	$10^{13}$
1/8	<b>22</b> ,0.371	<b>10</b> ,0.115	<b>10</b> ,0.116	<b>9</b> ,0.078	<b>9</b> ,0.056	<b>8</b> ,0.059	<b>8</b> ,0.045	<b>8</b> ,0.039	<b>8</b> ,0.039	<b>8</b> ,0.039	<b>8</b> ,0.039	<b>8</b> ,0.039
1/16	<b>16</b> ,0.268	<b>13</b> ,0.201	<b>11</b> ,0.131	<b>9</b> ,0.098	<b>8</b> ,0.043	<b>7</b> ,0.043	<b>6</b> ,0.017	<b>6</b> ,0.010	<b>6</b> ,0.007	<b>6</b> ,0.004	<b>5</b> ,0.010	<b>5</b> ,0.005
1/32	<b>20</b> ,0.350	<b>19</b> ,0.317	<b>14</b> ,0.192	<b>11</b> ,0.127	<b>9</b> ,0.093	<b>8</b> ,0.035	<b>7</b> ,0.041	<b>6</b> ,0.016	<b>6</b> ,0.010	<b>6</b> ,0.007	<b>6</b> ,0.008	<b>5</b> ,0.008
1/64	<b>29</b> ,0.483	<b>26</b> ,0.436	<b>17</b> ,0.283	<b>12</b> ,0.162	<b>10</b> ,0.092	<b>8</b> ,0.061	<b>8</b> ,0.030	<b>6</b> ,0.031	<b>6</b> ,0.017	<b>6</b> ,0.011	<b>6</b> ,0.005	<b>5</b> ,0.013

**Table 9** Preconditioner = AGKS - V(1,1)-cycle, Prolongation = Bi-linear, Smoother = ILU

$h \setminus m$	$10^0$	$10^1$	$10^2$	$10^3$	$10^4$	$10^5$	$10^6$	$10^7$	$10^8$	$10^9$	$10^{11}$	$10^{13}$
1/8	<b>22</b> ,0.371	<b>10</b> ,0.115	<b>10</b> ,0.116	<b>9</b> ,0.078	<b>9</b> ,0.056	<b>8</b> ,0.059	<b>8</b> ,0.045	<b>8</b> ,0.039	<b>8</b> ,0.039	<b>8</b> ,0.039	<b>8</b> ,0.039	<b>8</b> ,0.039
1/16	<b>16</b> ,0.237	<b>13</b> ,0.197	<b>11</b> ,0.131	<b>9</b> ,0.098	<b>8</b> ,0.043	<b>7</b> ,0.043	<b>6</b> ,0.018	<b>6</b> ,0.012	<b>6</b> ,0.007	<b>6</b> ,0.011	<b>5</b> ,0.010	<b>5</b> ,0.006
1/32	<b>20</b> ,0.348	<b>19</b> ,0.312	<b>14</b> ,0.192	<b>11</b> ,0.126	<b>9</b> ,0.093	<b>8</b> ,0.035	<b>7</b> ,0.041	<b>6</b> ,0.016	<b>6</b> ,0.010	<b>6</b> ,0.006	<b>6</b> ,0.021	<b>5</b> ,0.008
1/64	<b>29</b> ,0.483	<b>26</b> ,0.431	<b>17</b> ,0.283	<b>12</b> ,0.161	<b>10</b> ,0.092	<b>8</b> ,0.062	<b>8</b> ,0.026	<b>6</b> ,0.030	<b>6</b> ,0.015	<b>6</b> ,0.010	<b>6</b> ,0.006	<b>5</b> ,0.013

**Table 10** Preconditioner = AGKS - V(1,1)-cycle, Prolongation = Bi-linear, Smoother = sGS

$h \setminus m$	$10^0$	$10^1$	$10^2$	$10^3$	$10^4$	$10^5$	$10^6$	$10^7$	$10^8$	$10^9$	$10^{11}$	$10^{13}$
1/8	<b>22</b> ,0.371	<b>10</b> ,0.115	<b>10</b> ,0.116	<b>9</b> ,0.078	<b>9</b> ,0.056	<b>8</b> ,0.059	<b>8</b> ,0.045	<b>8</b> ,0.039	<b>8</b> ,0.039	<b>8</b> ,0.039	<b>8</b> ,0.039	<b>8</b> ,0.039
1/16	<b>16</b> ,0.256	<b>13</b> ,0.197	<b>11</b> ,0.131	<b>9</b> ,0.098	<b>8</b> ,0.043	<b>7</b> ,0.043	<b>6</b> ,0.017	<b>6</b> ,0.010	<b>6</b> ,0.007	<b>6</b> ,0.004	<b>5</b> ,0.010	<b>5</b> ,0.004
1/32	<b>20</b> ,0.352	<b>19</b> ,0.317	<b>14</b> ,0.192	<b>11</b> ,0.127	<b>9</b> ,0.093	<b>8</b> ,0.035	<b>7</b> ,0.041	<b>6</b> ,0.016	<b>6</b> ,0.010	<b>6</b> ,0.007	<b>6</b> ,0.020	<b>5</b> ,0.009
1/64	<b>29</b> ,0.481	<b>26</b> ,0.438	<b>17</b> ,0.281	<b>12</b> ,0.161	<b>10</b> ,0.092	<b>8</b> ,0.061	<b>8</b> ,0.026	<b>6</b> ,0.031	<b>6</b> ,0.018	<b>6</b> ,0.012	<b>6</b> ,0.008	<b>5</b> ,0.014

large values of  $m$ , whereas, the CCMG preconditioner results in a consistent failure.

### A The technical part of the spectral analysis for the diagonally scaled matrix

Let  $\{1, \dots, s\}$  denote the index of islands in the domain. Let  $N_k$ ,  $k = 1, \dots, s$  be the FV discretization matrix of  $-\Delta$  on the  $k$ -th island  $\Omega_k$  with respect to  $\mathcal{T}_i$  with homogeneous pure Neumann boundary condition. Let  $\mathcal{C}$  denote the set of all DOF in  $\Omega$ .

We start by examining the dependence of the  $(p, q)$ -th entry of  $K(m)$  on  $m$ , for each  $p \in \mathcal{C}$ . Let  $\Omega_{s+1}$  denote the outer region of islands, i.e.,

$$\Omega_{s+1} = \Omega \setminus \bigcup_{k=1, \dots, s} \bar{\Omega}_k.$$

We denote the cell-centers that are adjacent to  $\Omega_{s+1}$  and the ones that are interior to  $\Omega_k$ ,  $k = 1, \dots, s$  by  $\Gamma_{\Omega_k}$  and  $I_{\Omega_k}$ , respectively. On the other hand, the cell-centers in the outer region  $\Omega_{s+1}$  that are adjacent to the islands  $\Omega_k$ ,  $k = 1, \dots, s$  are denoted by  $\Gamma_{\Omega_{s+1}}$ .

We define the following index set for  $p, q \in \mathcal{C}$  with  $p \neq q$ :

$$\mathcal{I}(p \wedge q) := \begin{cases} k, & p \text{ and } q \in \Omega_k, k = 1, \dots, s \\ s+1, & p \text{ or } q \in \Omega_{s+1}, \end{cases}$$

$$\mathcal{I}(p) := \begin{cases} k, & p \in I_{\Omega_k}, k = 1, \dots, s \\ s+1, k, & p \in \Gamma_{\Omega_k}, k = 1, \dots, s \\ s+1, & p \in \Omega_{s+1}. \end{cases}$$

Also, we define  $\mathcal{I}(p \wedge p) = \mathcal{I}(p)$ .

Note that the discretization matrix and its entries can be written as follows:

$$K(m) = \sum_{k=1}^s mN_k(1) + N_{s+1}(m)$$

$$[K(m)]_{pq} = \sum_{\ell \in \mathcal{I}(p \wedge q)} \alpha_\ell(m) [N_\ell]_{pq},$$

where

$$\alpha_\ell(m) = \begin{cases} m, & \ell = 1, \dots, s \\ 1, & \ell = s+1, \end{cases} \quad (47)$$

and by abuse of notation, we have defined  $N_\ell := N_\ell(1)$  for  $\ell = 1, \dots, s$  and  $N_{s+1} := N_{s+1}(m)$ . Then,

$$\begin{aligned} & [A(m)]_{pq} \\ &= [K(m)]_{pp}^{-1/2} [K(m)]_{pq} [K(m)]_{qq}^{-1/2}, \\ &= \left\{ \sum_{\ell \in \mathcal{I}(p)} \alpha_\ell(m) [N_\ell]_{pp} \right\}^{-1/2} \sum_{\ell \in \mathcal{I}(p \wedge q)} \alpha_\ell(m) [N_\ell]_{pq} \\ &\quad \times \left\{ \sum_{\ell \in \mathcal{I}(q)} \alpha_\ell(m) [N_\ell]_{qq} \right\}^{-1/2}. \end{aligned}$$

For  $p \in \mathcal{C}$ ,  $[A(m)]_{pq} = 0$  if  $p, q$  are not adjacent cell-centers and  $[A(m)]_{pp} = 1$ . Furthermore note that,  $[A(m)]_{pq}$  is  $m$ -dependent only if either  $p$  or  $q \in \bigcup_{k=1, \dots, s+1} \Gamma_{\Omega_k}$ .

It is sufficient to study only the single island case because single island expression for

$$K(m) = mN_1 + N_2(m) \quad (48)$$

can be simply generalized to

$$K(m) = \sum_{k=1}^s mN_k + N_{s+1}(m).$$

### A.1 Perturbation expansion analysis for the upper bound of the smallest eigenvalue

We devise a perturbation expansion analysis based on  $m$ , in order to study the  $m$ -dependent spectral behavior of  $A(m)$ . Hence, only the matrix entries  $A(m)_{pq}$ ,  $p \neq q$ , that have  $m$ -dependence are considered where cells  $p$  and  $q$  are adjacent. Combining (11) and (48), one can deduce that

$$N_2(m) = N_2^\infty + \mathcal{O}(m^{-1}). \quad (49)$$

We will use (49) in the below analysis. For clarity, we treat the perturbation expansion in full detail for the first case.

**Case 1:**  $p \in \Gamma_{\Omega_1}$  and  $q \in \Omega_2$ ,

$$\begin{aligned} & [A(m)]_{pq} \\ &= \{m[N_1]_{pp} + [N_2(m)]_{pp}\}^{-1/2} [N_2(m)]_{pq} \{[N_2(m)]_{qq}\}^{-1/2} \\ &= m^{-1/2} \{[N_1]_{pp} + m^{-1}[N_2(m)]_{pp}\}^{-1/2} [N_2(m)]_{pq} \\ &\quad \times \{[N_2(m)]_{qq}\}^{-1/2} \\ &= m^{-1/2} \left\{ m^{-1}[N_1]_{pp}^{-1/2} - 1/2[N_1]_{pp}^{-3/2} [N_2(m)]_{pp} \right. \\ &\quad \left. + \mathcal{O}(m^{-2}) \right\} \{[N_2(m)]_{pq}\} \{[N_2(m)]_{qq}\}^{-1/2} \\ &= m^{-1} \left\{ [N_1]_{pp}^{-1/2} + \mathcal{O}(m^{-1}) \right\} \{[N_2^\infty]_{pq} + \mathcal{O}(m^{-1})\} \\ &\quad \times \left\{ m^{-1}[N_2^\infty]_{qq} + \mathcal{O}(m^{-2}) \right\}^{-1/2} \\ &= \mathcal{O}(m^{-1/2}) \end{aligned}$$

**Case 2:**  $p \in \Gamma_{\Omega_1}$  and  $q \in \Gamma_{\Omega_1}$ ,

$$\begin{aligned} & [A(m)]_{pq} \\ &= \{m[N_1]_{pp} + [N_2(m)]_{pp}\}^{-1/2} [mN_1]_{pq} \\ &\quad \times \{m[N_1]_{qq} + [N_2(m)]_{qq}\}^{-1/2} \\ &= \{[N_1]_{pp} + m^{-1}[N_2(m)]_{pp}\}^{-1/2} [N_1]_{pq} \\ &\quad \times \{[N_1]_{qq} + m^{-1}[N_2(m)]_{qq}\}^{-1/2} \\ &= [N_1]_{pp}^{-1/2} [N_1]_{pq} [N_1]_{qq}^{-1/2} + \mathcal{O}(m^{-1}) \end{aligned}$$

**Case 3:**  $p \in \Gamma_{\Omega_1}$  and  $q \in I_{\Omega_1}$ ,

$$\begin{aligned} & [A(m)]_{pq} \\ &= \{m[N_1]_{pp} + [N_2(m)]_{pp}\}^{-1/2} [mN_1]_{pq} \{m[N_1]_{qq}\}^{-1/2} \\ &= \{[N_1]_{pp} + m^{-1}[N_2(m)]_{pp}\}^{-1/2} [N_1]_{pq} \{[N_1]_{qq}\}^{-1/2} \\ &= [N_1]_{pp}^{-1/2} [N_1]_{pq} [N_1]_{qq}^{-1/2} + \mathcal{O}(m^{-1}) \end{aligned}$$

**Case 4:**  $p \in \Gamma_{\Omega_2}$  and  $q \in \Omega_2$ ,

$$\begin{aligned} & [A(m)]_{pq} \\ &= \{[N_2(m)]_{pp}\}^{-1/2} [N_2(m)]_{pq} \{[N_2(m)]_{qq}\}^{-1/2} \\ &= \{m^{-1}[N_2(m)]_{pp}\}^{-1/2} m^{-1}[N_2(m)]_{pq} \\ &\quad \times \{m^{-1}[N_2(m)]_{qq}\}^{-1/2} \\ &= \{m^{-1}[N_2^\infty]_{pp} + \mathcal{O}(m^{-2})\}^{-1/2} \{m^{-1}[N_2^\infty]_{pq} + \mathcal{O}(m^{-2})\} \\ &\quad \times \{m^{-1}[N_2^\infty]_{qq} + \mathcal{O}(m^{-2})\}^{-1/2} \\ &= [N_2^\infty]_{pp}^{-1/2} [N_2^\infty]_{pq} [N_2^\infty]_{qq}^{-1/2} + \mathcal{O}(m^{-1}). \end{aligned}$$

We will use following modification of  $N_2$  for our further analysis:

$$[\tilde{N}_2]_{pq} = \begin{cases} 0 & \text{if } p \in \Gamma_{\Omega_1} \\ [N_2^\infty]_{pq} & \text{otherwise.} \end{cases} \quad (50)$$

Consider the reduced version of (48)

$$\tilde{K}(m) = mN_1 + \tilde{N}_2,$$

and let  $\tilde{A}(m)$  denote the diagonally scaled version of  $\tilde{K}(m)$ . Then  $m$ -independent  $\tilde{A}(m)$  has a single zero

eigenvalue. Next, we proceed with the elementwise analysis of  $A(m) - \tilde{A}(m)$ .

**Case 1:**  $p \in \Gamma_{\Omega_1}$  and  $q \in \Omega_2$ ,

$$[A(m)]_{pq} = m^{-1/2} [N_1]_{pp}^{-1/2} [N_2^\infty]_{pq} [N_2^\infty]_{qq}^{-1/2} + \mathcal{O}(m^{-3/2})$$

and from (50) we get

$$[\tilde{A}(m)]_{pq} = 0.$$

Therefore,

$$[A(m)]_{pq} - [\tilde{A}(m)]_{pq} = \mathcal{O}(m^{-1/2}).$$

**Case 2:**  $p \in \Gamma_{\Omega_1}$  and  $q \in \Omega_1$ ,

$$[A(m)]_{pq} = [N_1]_{pp}^{-1/2} [N_1]_{pq} [N_1]_{qq}^{-1/2} + \mathcal{O}(m^{-1})$$

and from (50) we get

$$[\tilde{A}(m)]_{pq} = [N_1]_{pp}^{-1/2} [N_1]_{pq} [N_1]_{qq}^{-1/2}$$

Thus,

$$[A(m)]_{pq} - [\tilde{A}(m)]_{pq} = \mathcal{O}(m^{-1}).$$

**Case 3:**  $p \in \Gamma_{\Omega_2}$  (nodes of  $\Omega_2$ ),

$$[A(m)]_{pq} = [N_2^\infty]_{pp}^{-1/2} [N_2^\infty]_{pq} [N_2^\infty]_{qq}^{-1/2} + \mathcal{O}(m^{-1})$$

and from (50) we get

$$[\tilde{A}(m)]_{pq} = [N_2^\infty]_{pp}^{-1/2} [N_2^\infty]_{pq} [N_2^\infty]_{qq}^{-1/2}.$$

Thus,

$$[A(m)]_{pq} - [\tilde{A}(m)]_{pq} = \mathcal{O}(m^{-1}).$$

**Case 4:** Otherwise,

$$[A(m)]_{pq} - [\tilde{A}(m)]_{pq} = 0.$$

Therefore,  $\lambda_{\max}(A(m) - \tilde{A}(m)) = \mathcal{O}(m^{-1/2})$ .

**Lemma 3** *Let  $G$  and  $H$  be symmetric matrices of dimension  $n \times n$ . Then, for  $k = 1, \dots, n$ , the following holds:*

$$\lambda_k(G) + \lambda_{\min}(H) \leq \lambda_k(G + H) \leq \lambda_k(G) + \lambda_{\max}(H).$$

*Proof* The result follows from Courant-Fischer min-max Theorem; see [13, Corollary 8.1.3].  $\square$

From Lemma 3, we have

$$\begin{aligned} \lambda_{\min}(A(m)) &\leq \lambda_{\min}(\tilde{A}(m)) + \lambda_{\max}(A(m) - \tilde{A}(m)) \\ &= \lambda_{\max}(A(m) - \tilde{A}(m)) \\ &= \mathcal{O}(m^{-1/2}) \end{aligned}$$

Moreover, we have for all  $k \geq 1$ ,

$$\begin{aligned} \lambda_k(A(m)) &\geq \lambda_k(\tilde{A}(m)) + \lambda_{\min}(A(m) - \tilde{A}(m)) \\ &\geq \lambda_k(\tilde{A}(1)) - \mathcal{O}(m^{-1}) \geq C \end{aligned}$$

for some constant  $C$  independent of  $m$ , asymptotically. Thus,  $A(m)$  has a single eigenvalue approaching to zero while the remaining eigenvalues are bounded away from 0.

## A.2 Lower bound for the smallest eigenvalue

We aim to show the following lower bound for the smallest eigenvalue:

$$\lambda_{\min}(A(m)) \geq C m^{-1}. \quad (51)$$

For that, we will establish the below main estimates in the discussion to follow:

$$x^T K(m)x \geq x^T K(1)x \geq m^{-1} x^T K(m)x. \quad (52)$$

*Remark 4* Establishing the estimate (52) is not as simple as in the FE discretization case due to  $m$ -dependence of  $N_2$  in (48). This requires further detailed matrix analysis.

### A.2.1 $x^T K(m)x \geq x^T K(1)x$ estimate

For the decomposition in (48), it is straightforward to see that  $mN_1 \geq N_1$  for  $m \geq 1$ . Hence, in order to establish (52), we concentrate on the following auxiliary estimate:

$$x^T N_2(m)x \geq x^T N_2(1)x. \quad (53)$$

First, note that  $N_2(m)$  has positive diagonal and negative off-diagonal entries. In addition, due to the discretization formula (9), it has a row sum equal to zero with the exception that the row sums corresponding to cell-centers that are adjacent to the boundary are positive. Hence,  $N_2(m)$  is a diagonally dominant matrix. We further decompose  $N_2(m)$  as follows:

$$N_2(m) = \bar{N}_2(m) + R_2, \quad (54)$$

where  $\bar{N}_2(m)$  is a symmetric matrix with positive diagonal entries and a row sum equal to zero, and  $R_2 := N_2(m) - \bar{N}_2(m)$  is the remainder matrix.

**Lemma 4** *Let  $G$  be a symmetric matrix and have a row sum equal to zero. In addition, let  $[G]_{pp} \geq 0$ , then  $G$  is symmetric positive semi-definite (SPSD).*

*Proof* Let  $\lambda_p$  be the  $p$ -th eigenvalue of  $G$ . Using Gerschgorin's Theorem with the fact that  $G$  has a row sum equal to zero yields:

$$|\lambda_p - [G]_{pp}| \leq \sum_{q \neq p} |[G]_{pq}| = [G]_{pp}.$$

The result follows from  $0 \leq \lambda_p \leq 2[G]_{pp}$ .  $\square$

By Lemma 4,  $\bar{N}_2(m)$  is immediately SPSPD. In addition,  $R_2$  is a diagonally dominant matrix with non-negative diagonal entries, again by Lemma 4,  $R_2$  is SPSPD. Now, we can conclude that  $N_2(m)$  is SPSPD.

From (11), one observes that  $[K(m)]_{pp}$  is monotonically increasing in  $m$ . This important property implies that

$$[\bar{N}_2(m)]_{pp} \geq [\bar{N}_2(1)]_{pp}, \quad m \geq 1. \quad (55)$$

We need the following additional decomposition:

$$\bar{N}_2(m) = \bar{N}_2(1) + \bar{R}_2(m). \quad (56)$$

Combining (55) and (56), we obtain  $[\bar{R}_2(m)]_{pp} \geq 0$ . Hence, Lemma 4 implies that  $\bar{R}_2(m)$  is SPSD. Using this, we now arrive at the auxiliary estimate (53) in the following:

$$\begin{aligned} x^T N_2(m)x &= x^T \bar{N}_2(1)x + x^T \bar{R}_2(m)x + x^T R_2x \\ &\geq x^T \bar{N}_2(1)x + x^T R_2x \\ &= x^T N_2(1)x. \end{aligned}$$

Consequently,

$$\begin{aligned} mx^T K(m)x &= mx^T N_1x + x^T N_2(m)x \\ &\geq x^T N_1x + x^T N_2(m)x \\ &\geq x^T N_1x + x^T N_2(1)x \\ &= x^T K(1)x. \end{aligned}$$

### A.2.2 $mx^T K(1)x \geq x^T K(m)x$ estimate

Combining (48) and (54), we obtain

$$K(m) = mN_1 + \bar{N}_2(m) + R_2,$$

where  $N_1$  and  $R_2$  are SPSD and independent of  $m$ , which yields the following for  $m \geq 1$ :

$$\begin{aligned} mx^T N_1x &\geq x^T N_1x, \\ mx^T R_2x &\geq x^T R_2x, \end{aligned}$$

Thus, in order to establish (52), it is sufficient to establish the auxiliary estimate

$$mx^T \bar{N}_2(1)x \geq x^T \bar{N}_2(m)x. \quad (57)$$

By using 11, one can also show that

$$m[\bar{N}_2(1)]_{pp} \geq [\bar{N}_2(m)]_{pp}. \quad (58)$$

We will use the following decomposition:

$$m\bar{N}_2(1) = \bar{N}_2(m) + \hat{R}_2(m), \quad (59)$$

where  $\hat{R}_2(m)$  is the SPSD remainder matrix. Combining (58) and (59), we obtain  $[\hat{R}_2(m)]_{pp} \geq 0$ , which leads us to the auxiliary estimate (57).

Hence,

$$\begin{aligned} mx^T K(1)x &= mx^T N_1x + mx^T N_2(1)x \\ &\geq mx^T N_1x + x^T N_2(m)x \\ &= x^T K(m)x. \end{aligned}$$

In conclusion, we have obtained the two main estimates in (52). These yield similar estimates for  $\text{diag } K(m)$ .

$$\begin{aligned} x^T \text{diag } K(m)x &\geq x^T \text{diag } K(1)x \\ &\geq m^{-1} x^T \text{diag } K(m)x \end{aligned} \quad (60)$$

From (52) and (60), and letting  $C_1 := \lambda_{\min}(A(1))$ , we get:

$$\begin{aligned} x^T K(m)x &\geq x^T K(1)x \geq C_1 x^T \text{diag } K(1)x \\ &\geq C_1 m^{-1} x^T \text{diag } K(m)x, \end{aligned}$$

yielding

$$C_1 m^{-1} \leq \lambda_{\min}(A(m)) \leq C_2 m^{-1/2}.$$

## References

1. Aarnes, J.E.: Modelling of multiscale structures in flow simulations for petroleum reservoirs. In: Geometric Modelling, Numerical Simulation, and Optimization Applied Mathematics at SINTEF, pp. 307–360. Springer Verlag (2007)
2. Aarnes, J.E., Gimse, T., Lie, K.A.: An introduction to the numerics of flow in porous media using Matlab. In: G. Hasle, K.A. Lie, E. Quak (eds.) Geometrical Modeling, Numerical Simulation and Optimisation: Industrial Mathematics at SINTEF, pp. 265–306. Springer Verlag (2007)
3. Aarnes, J.E., Hou, T.: Multiscale domain decomposition methods for elliptic problems with high aspect ratios. Acta Mathematicae Applicatae Sinica **18**(1), 63–76 (2002)
4. Aksoylu, B., Beyer, H.R.: On the characterization of the asymptotic cases of the diffusion equation with rough coefficients and applications to preconditioning. Numer. Funct. Anal. Optim. **30**, 405–420 (2009)
5. Aksoylu, B., Beyer, H.R.: Results on the diffusion equation with rough coefficients. SIAM J. Math. Anal. **42**(1), 406–426 (2010)
6. Aksoylu, B., Bond, S., Holst, M.: An odyssey into local refinement and multilevel preconditioning III: Implementation and numerical experiments. SIAM J. Sci. Comput. **25**(2), 478–498 (2003)
7. Aksoylu, B., Graham, I.G., Klie, H., Scheichl, R.: Towards a rigorously justified algebraic preconditioner for high-contrast diffusion problems. Comput. Vis. Sci. **11**, 319–331 (2008)
8. Aksoylu, B., Klie, H.: A family of physics-based preconditioners for solving elliptic equations on highly heterogeneous media. Appl. Num. Math. **59**, 1159–1186 (2009)
9. Aksoylu, B., Yeter, Z.: Robust multigrid preconditioners for the high-contrast biharmonic plate equation (2009). Revised, Numer. Linear Algeb. Appl., arXiv 0910.0487
10. Ewing, R.E., Shen, J.: A multigrid algorithm for the cell-centered finite difference scheme. In: The Proceedings of 6-th Copper Mountain conference on Multigrid Methods. NASA Conference Publication 3224 (1993)
11. Eymard, R., Gallout, T., Herbin, R.: The finite volume method. In: P.G. Ciarlet, J. Lions (eds.) Handbook for Numerical Analysis, pp. 715–1022. North Holland (2000)
12. Gerritsen, M., Durllofsky, L.: Modeling fluid flow in oil reservoirs. Annu. Rev. Fluid Mech **37**, 211–238 (2005)
13. Golub, G.H., Loan, C.F.V.: Matrix Computations. Johns Hopkins University Press, Baltimore, MD (1989)
14. Graham, I.G., Hagger, M.J.: Unstructured additive Schwarz-conjugate gradient method for elliptic problems with highly discontinuous coefficients. SIAM J. Sci. Comp. **20**(6), 2041–2066 (1999)

15. Graham, I.G., Lechner, P.O., Scheichl, R.: Domain decomposition for multiscale PDEs. *Numer. Math.* **106**, 589–626 (2007). DOI 10.1007/s00211-007-0074-1
16. Graham, I.G., Scheichl, R.: Robust domain decomposition algorithms for multiscale PDEs. *Numer. Methods Partial Differential Equations* pp. 859–878 (2007). DOI 10.1002/num.20254
17. Knyazev, A., Widlund, O.: Lavrentiev regularization + Ritz approximation = uniform finite element error estimates for differential equations with rough coefficients. *Math. Comp.* **72**(241), 17–40 (2003)
18. Kwak, D.Y.: V-cycle multigrid for cell-centered finite differences. *SIAM J. Sci. Comput.* **21**(2), 552–564 (1999)
19. Kwak, D.Y., Lee, J.S.: Multigrid algorithm for the cell-centered finite difference method ii: Discontinuous coefficient case. *Numer. Meth. Partial Differential Eqs.* **20**(5), 723–741 (2004)
20. Llorente, I.M., Melson, N.D.: Behavior of plane relaxation methods as multigrid smoothers. *Elect. Trans. Numer. Anal.* **10**, 92–114 (2000)
21. Ming, P., Ye, X.: Numerical methods for multiscale elliptic problems. *J. Comp. Phys.* **214**, 421–445 (2006)
22. Mohr, M., Wienands, R.: Cell-centered multigrid revisited. *Comput. Vis. Sci.* **7**, 129–140 (2004)
23. Nicolaidis, R.: Deflation of conjugate gradients with applications to boundary value problems. *SIAM J. Numer. Anal.* **24**, 355–365 (1987)
24. Nøttinger, B., Artus, V., Zargar, G.: The future of stochastic and upscaling methods in hydrogeology. *Hydrogeology Journal* **13**, 184–201 (2005)
25. Scheichl, R., Vainikko, E.: Additive schwarz and aggregation-based coarsening for elliptic problems with highly variable coefficients. *Computing* **80**(4), 319–343 (2007)
26. Trottenberg, U., Oosterlee, C., Schüller, A.: *Multigrid*. Academic Press, London (2001)
27. Vuik, C., Segal, A., Meijerink, J.: An efficient preconditioned CG method for the solution of a class of layered problems with extreme contrasts of coefficients. *J. Comp. Phys.* **152**, 385–403 (1999)
28. Vuik, C., Segal, A., Meijerink, J.: An efficient preconditioned CG method for the solution of a class of layered problems with extreme contrasts of coefficients. *J. Comp. Phys.* **152**, 385–403 (1999)
29. Vuik, C., Segal, A., Meijerink, J., Wijma, G.: The construction of projection vectors for a ICCG method applied to problems with extreme contrasts in the coefficients. *J. Comp. Phys.* **172**, 426–450 (2001)
30. Wesseling, P.: Cell-centered multigrid for interface problems. *J. Comput. Phys.* **79**, 85–91 (1988)
31. Wesseling, P.: *An Introduction to Multigrid Methods*. <http://www.MGNet.org>, Cos Cob, CT (2001). Reprint of the 1992 edition
32. Wesseling, P., Khalil, M.: Vertex-centered and cell-centered multigrid for interface problems. *Journal of Computational Physics* **98**, 1–10 (1992)
33. Wittum, G.: Linear iterations as smoothers in multigrid methods: Theory with applications to incomplete decompositions. *Impact Comput. Sci. Eng.* **1**, 180–215 (1989)
34. Wittum, G.: On the robustness of ILU-smoothing. *SIAM J. Sci. Stat. Comput.* **10**, 699–717 (1989)