HAL open science

A-posteriori-steered and adaptive p-robust multigrid solvers

Ani Miraçi

▶ To cite this version:

Ani Miraçi. A-posteriori-steered and adaptive p-robust multigrid solvers. Numerical Analysis [math.NA]. INRIA Paris; Sorbonne Universite, 2020. English. NNT : . tel-03334270v1

HAL Id: tel-03334270

https://hal.science/tel-03334270v1

Submitted on 25 Feb 2021 (v1), last revised 22 Nov 2021 (v3)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

THÈSE

PRÉSENTÉE À

SORBONNE UNIVERSITÉ

ÉCOLE DOCTORALE: Sciences Mathématiques de Paris Centre (ED 386)

Par **Ani MIRAÇI**

POUR OBTENIR LE GRADE DE DOCTEUR

SPÉCIALITÉ: Mathématiques Appliquées

Solveurs multigrille p-robustes et adaptatifs guidés par des estimateurs a posteriori

---

Directeur de thèse: Martin Vohralík (Inria Paris)

Co-directeurs de thèse: Alexandre Ern (École des Ponts ParisTech)

Soutenue le: 14 décembre 2020

Devant la commission d'examen formée de:

Paola ANTONIETTI	Politecnico di Milano	Examinatrice
Roland BECKER	Université de Pau et des Pays de l'Adour	Examinateur
Alexandre ERN	École des Ponts ParisTech	Co-directeur de thèse
Martin GANDER	Université de Genève	Examinateur
Frédéric NATAF	Sorbonne Université	Examinateur
Dirk PRAETORIUS	Technische Universität Wien	Rapporteur
Marco VERANI	Politecnico di Milano	Rapporteur
Martin VOHRALÍK	Inria Paris	Directeur de thèse

THESIS

PRESENTED AT THE

SORBONNE UNIVERSITY

DOCTORAL SCHOOL: Mathematical Sciences of Central Paris (ED 386)

By **Ani MIRAÇI**

TO OBTAIN THE DEGREE OF DOCTOR OF PHILOSOPHY
SPECIALITY: Applied Mathematics

A-posteriori-steered and adaptive p-robust multigrid solvers

Thesis advisor: Martin Vohralík (Inria Paris)
Thesis co-advisors: Alexandre Ern (École des Ponts ParisTech)

Defended on: December 14th, 2020

In front of the examination committee consisting of:

Paola ANTONIETTI	Politecnico di Milano	Examiner
Roland BECKER	Université de Pau et des Pays de l'Adour	Examiner
Alexandre ERN	École des Ponts ParisTech	Thesis co-advisor
Martin GANDER	Université de Genève	Examiner
Frédéric NATAF	Sorbonne Université	Examiner
Dirk PRAETORIUS	Technische Universität Wien	Reviewer
Marco VERANI	Politecnico di Milano	Reviewer
Martin VOHRALÍK	Inria Paris	Thesis advisor

To my family and friends
for unwaveringly supporting me.

To all my math teachers
for lighting a spark which led me here.

SERENA

This thesis has been carried out within the research team SERENA, a joint project-team between **Inria Paris** and **Ecole des Ponts ParisTech**.

Inria
2 rue Simone Iff
75589 Paris, France

CERMICS
Ecole des Ponts
77455 Marne-la-Vallée, France

Résumé

Dans cette thèse, nous considérons des systèmes d'équations algébriques linéaires provenant de discrétisation d'équations aux dérivées partielles elliptiques de second ordre par des éléments finis de degré polynomial arbitraire.

Dans le Chapitre 1 nous proposons un estimateur a posteriori pour l'erreur algébrique, la construction duquel est intrinsèquement liée à celle d'un solveur multigrille avec zéro pas de pré-lissage et seulement un pas de post-lissage par des méthodes de Schwarz (bloc-Jacobi) avec recouvrement. Les contributions principales de cette approche portent sur les deux résultats suivants ainsi que leur équivalence : le solveur contracte l'erreur algébrique indépendamment du degré polynomial (p-robustesse) ; l'estimateur représente une borne p-robuste supérieure et inférieure de l'erreur algébrique. Les preuves de ces résultats sont valables en un, deux, et trois dimensions d'espace, sous l'hypothèse de régularité minimale H^1 de la solution faible, pour des maillages quasi-uniformes ou bien des maillages issus de raffinements adaptatifs par bissection, et sont indépendantes de la base de l'espace d'éléments finis choisie. Nous introduisons ici un pas optimal (par recherche linéaire) de l'étape de correction d'erreur de multigrille.

Dans le Chapitre 2 nous introduisons des pas optimaux par niveau, ainsi maximisant la réduction d'erreur algébrique à chaque niveau. Sous hypothèse de régularité H^2, nous prouvons ici que la contraction/efficacité p-robuste sont aussi indépendants du nombre de niveaux dans la hiérarchie de maillages. En plus de l'amélioration des performances du solveur, l'utilisation des pas optimaux par niveau conduit également à une formule de Pythagore explicite de la réduction de l'erreur algébrique d'une itération à l'autre. La formule sert alors de fondement pour une stratégie adaptative simple et efficace qui permet au solveur de choisir le nombre nécessaire de pas de post-lissage à chaque niveau.

Dans le Chapitre 3 nous introduisons une stratégie de lissage local adaptatif grâce à notre estimateur efficace a posteriori, qui a la propriété importante d'être localisé par niveaux et par patchs d'éléments. Ainsi, l'estimateur peut détecter et marquer quels patchs d'éléments parmi tous les niveaux contribuent plus qu'un pourcentage prescrit par l'utilisateur de l'erreur algébrique globale (via un critère de type bulk-chasing). Chaque itération du solveur adaptatif est ici composée de deux sous-étapes: après un premier V-cycle non adaptatif, un deuxième V-cycle adaptatif et peu coûteux n'utilise le lissage local que dans les patchs marqués. Nous prouvons que chacune de ces sous-étapes contracte l'erreur algébrique de manière p-robuste.

Pour terminer, dans le Chapitre 4 nous donnons des extensions des résultats ci-dessus au cadre d'éléments finis mixtes en deux dimensions d'espace.

Une variété de tests numériques est présentée pour confirmer les résultats théoriques de cette thèse, ainsi que pour montrer les avantages de nos approches p-robustes et/ou d'adaptivité de solveurs algébriques.

Mots-clés : problème elliptique de deuxième ordre, méthode des éléments finis, solveur algébrique itératif, méthode multigrille, méthode de Schwarz, lisseur bloc-Jacobi, erreur algébrique, estimateur d'erreur a posteriori, p-robustesse, décomposition stable, pas optimaux, recherche linéaire, adaptivité, choix adaptatif du nombre de pas de lissage, lissage local

Abstract

In this thesis, we consider systems of linear algebraic equations arising from discretizations of second-order elliptic partial differential equations using finite elements of arbitrary polynomial degree.

In Chapter 1, we propose an a posteriori estimator for the algebraic error whose construction is inherently interconnected with the design of a multigrid solver with zero pre- and only one post-smoothing step by overlapping Schwarz (block-Jacobi) methods. The main contribution of this approach consists in the two following results and their equivalence: the solver contracts the algebraic error independently of the polynomial degree (p-robustness); the estimator represents a two-sided p-robust bound on the algebraic error. The proofs of these results hold in one, two, and three space dimensions, under the minimal H^1-regularity of the weak solution, for quasi-uniform meshes as well as for possibly highly graded ones, and are independent of the basis of the chosen finite element space. We introduce here an optimal step-size (by line search) in the error correction stage of the multigrid.

In Chapter 2, we introduce level-wise optimal step-sizes, thus maximizing the decrease of the algebraic error on each level. Under the H^2-regularity assumption, we prove here that the p-robust contraction/efficiency also hold independently of the number of mesh levels. Apart from improving the performance of the solver, the use of the level-wise step-sizes also leads to an explicit Pythagorean formula of the decrease of the algebraic error from one iteration to the next. The formula then serves as foundation of a simple and effective adaptive strategy which allows the solver to choose the necessary number of post-smoothing steps on each level.

In Chapter 3, we introduce an adaptive local smoothing strategy thanks to our efficient a posteriori estimator, which has the important property of being localized level-wise and patch-wise. Thus, the estimator can detect and mark which patches of elements among all mesh levels contribute more than a user prescribed percentage to the global algebraic error (via a bulk-chasing criterion). Each iteration of the adaptive solver is here composed of two sub-steps: after a first non-adaptive V-cycle, a second adaptive and inexpensive V-cycle employs local smoothing only in the marked patches. We prove that each of these sub-steps contracts the algebraic error p-robustly.

Finally, in Chapter 4, we provide an extension of the above results to the case of the mixed finite element method discretization in two space dimensions.

A variety of numerical tests is presented to confirm the theoretical findings of this thesis, as well as to show the benefits of our p-robust and/or adaptive solver approaches.

Keywords: second-order elliptic problem, finite element method, algebraic iterative solver, multigrid method, Schwarz method, block-Jacobi smoother, algebraic error, a posteriori error estimate, p-robustness, stable decomposition, optimal step-sizes, line search, adaptivity, adaptive choice of the number of smoothing steps, local smoothing

Acknowledgements

This thesis would not have been possible without the help of many people. First and foremost, my deepest gratitude goes to my supervisors, Martin Vohralík and Alexandre Ern, for trusting me with this project and for the invaluable scientific guidance and kindness they provided me with since day one of this endeavor. Thank you Martin for teaching me so much these last three years, it has been an incredibly rewarding experience. I am deeply grateful for your continuous support and patience throughout the thesis. Thank you Alexandre for all the insightful advices very much needed in the most crucial stages of the thesis.

My profound thanks also go to Jan Papež, from whom I have also learned a great deal. Thank you so much Jan for your tirelessness, all the productive discussions, and your numerous contributions, including the remarkable self-made finite element code, which have helped to shape this work. I am also very grateful for the opportunity of engaging in so many constructive discussions with Ivan Yotov during his visit at Inria Paris.

I wish to express my sincere gratitude to Martin Gander, Dirk Praetorius, and Marco Verani for carefully reading the thesis and the reports they wrote. I am truly honored for your participation in the thesis committee, including Paola Antonietti, Roland Becker, and Frédéric Nataf as examiners. I owe you all many thanks for your investment in the thesis defense and for sparking fascinating discussions.

By serendipity I joined the Serena team (pun intended) three years ago. It has been such a pleasure sharing the work environment with all the colleagues – permanent researchers, research engineers, team assistants, post-docs, Ph.D. students, and interns. I cherish the memories of our internal seminars, scientific discussions, coffee breaks, and holiday gatherings. Being in the team has been as much a scientifically stimulating experience as a humanely nurturing one. Beyond this, I am very grateful for the friendships that were kindled at Inria, including with members of the other research teams.

I cannot help but feel grateful for having always had amazing mathematics teachers, from childhood to the present day, I have been extremely fortunate. It is also thanks to each and everyone of them that this thesis was made possible.

Immense thanks finally go to my family and friends, for their loving support, their gentle encouragements, their honest advice, their unshaken optimism, their good as well as bad jokes. You all bring joy to my life and I am most lucky to have you.

"Together we're invincible."
- Muse

Contents

Résumé **v**

Abstract **vii**

Acknowledgements **ix**

List of Figures **xvii**

List of Tables **xix**

Introduction **1**

i Finite element method 1
ii Linear solvers 2
ii.1 Direct solvers 3
ii.2 Iterative solvers 3
ii.3 Adaptive linear solvers 6
ii.4 Robustness with respect to the polynomial degree p 6
iii Two central building blocks for the results of the thesis 6
iv Model problem and its discretization 7
v A posteriori point of view and goals of the thesis 7
v.1 Multilevel setting 8
v.2 Multilevel procedure for constructing an a posteriori estimator of the algebraic error and a linear solver 9
v.3 Main results: p-robust efficiency of the a posteriori estimator and p-robust solver contraction 12
v.4 Alternative approaches 13
v.5 Extension to the mixed finite element method 14
vi Adaptivity in a-posteriori-steered solvers 15
vi.1 Adaptive number of post-smoothing steps 15
vi.2 Adaptive local smoothing 16
vii Contents and contributions of the thesis 21
vii.1 Chapter 1 21
vii.2 Chapter 2 21
vii.3 Chapter 3 22
vii.4 Chapter 4 22
vii.5 Implementation notes 23
viii Perspectives 23

1 A multilevel algebraic error estimator and the corresponding iterative solver with p-robust behavior 25
1 Introduction . . . 26
2 Setting . . . 29
2.1 Model problem . . . 29
2.2 Finite element discretization . . . 29
2.3 Algebraic system, approximate solution, and algebraic residual 29
2.4 A hierarchy of meshes . . . 30
2.5 A hierarchy of spaces . . . 31
2.6 Two types of patches . . . 31
3 Multilevel lifting of the algebraic residual . . . 32
3.1 Exact algebraic residual lifting . . . 32
3.2 Coarse solve . . . 33
3.3 Multilevel algebraic residual lifting . . . 33
4 An a posteriori estimator on the algebraic error and a multilevel solver 34
4.1 A posteriori estimate on the algebraic error . . . 34
4.2 Multilevel solver . . . 35
5 Main results . . . 36
6 Numerical experiments . . . 38
6.1 Performance of the damped additive Schwarz (dAS) construction of the solver . . . 39
6.2 Performance of the weighted restrictive additive Schwarz (wRAS) construction of the solver . . . 42
6.3 Comparison with other multilevel solvers . . . 43
7 Proofs of the main results . . . 45
7.1 Upper bound on $\|\nabla \rho^i_{J,\mathrm{alg}}\|$ . . . 46
7.2 Lower bound on $(f, \rho^i_{J,\mathrm{alg}}) - (\nabla u^i_J, \nabla \rho^i_{J,\mathrm{alg}})$ . . . 47
7.3 Polynomial-degree-robust multilevel stable decomposition . . . 48
7.4 Upper bound on $\|\nabla \tilde{\rho}^i_{J,\mathrm{alg}}\|$ . . . 51
7.5 Proof of Theorem 5.1 . . . 53
7.6 Proof of Corollary 5.6 . . . 54
8 Conclusions and outlook . . . 54

2 A-posteriori-steered p-robust multigrid with optimal step-sizes and adaptive number of smoothing steps 56
1 Introduction . . . 57
2 Setting . . . 59
2.1 Model problem, finite element discretization, and algebraic system . . . 60
2.2 A hierarchy of meshes and spaces . . . 60
3 Motivation: level-wise orthogonal decomposition of the error . . . 61
4 Multilevel solver . . . 62
5 A posteriori estimator on the algebraic error . . . 64
6 Main results . . . 65
6.1 Setting, mesh, and regularity assumptions . . . 65
6.2 Main results . . . 66
6.3 Additional results . . . 67

7 Adaptive number of smoothing steps . . . 68
8 Complexity of the solver . . . 69
9 Numerical experiments . . . 69
9.1 Performance of the multilevel solver of Definition 4.1 . . . 70
9.2 Adaptive number of smoothing steps using Definition 7.1 . . . 71
9.3 Examples in three space dimensions . . . 74
9.4 Comparison with solvers from literature . . . 74
10 Proof of Theorem 6.6 . . . 77
10.1 Properties of the estimator η^i_{alg} . . . 77
10.2 Properties of the exact residual lifting $\tilde{\rho}^i_{J,\mathrm{alg}}$ . . . 77
10.3 Proof of Theorem 6.6 under the minimal $H^1_0(\Omega)$-regularity assumption . . . 78
10.4 Proof of Theorem 6.6 under the $H^2(\Omega)$-regularity assumption 80
11 Conclusions and future work . . . 85

3 Contractive local adaptive smoothing based on Dörfler's marking in a-posteriori-steered p-robust multigrid solvers 86
1 Introduction . . . 87
2 Setting . . . 89
2.1 Model problem and its finite element discretization . . . 89
2.2 A hierarchy of meshes and spaces . . . 89
3 Adaptive multilevel solver . . . 90
3.1 Algorithmic description of the solver . . . 91
3.2 Mathematical description of the solver . . . 93
4 A posteriori estimator on the algebraic error . . . 96
5 Main results . . . 97
5.1 Mesh assumptions . . . 97
5.2 Main result . . . 98
5.3 Additional results . . . 99
6 Numerical experiments . . . 99
6.1 Can we predict the distribution of the algebraic error? . . . 100
6.2 Does the adaptivity pay off? . . . 101
6.3 Dependence on the marking parameter . . . 105
7 Proofs of the main results . . . 107
7.1 Proof of contraction: full-smoothing substep . . . 107
7.2 Proof of contraction: adaptive-smoothing substep . . . 109
8 Conclusions . . . 112

4 p-robust multilevel and domain decomposition methods with optimal step-sizes for mixed finite element discretizations of elliptic problems 114
1 Introduction . . . 115
2 Model problem and its mixed finite element discretization . . . 116
2.1 Discrete mixed finite element problem . . . 117
3 Multilevel setting . . . 117
3.1 A hierarchy of meshes . . . 117
3.2 A hierarchy of spaces . . . 118

4 An a-posteriori-steered multigrid solver 119
4.1 Multigrid solver 119
4.2 A posteriori estimator on the algebraic error 121
5 An a-posteriori-steered domain decomposition solver 121
5.1 Two-level hierarchy 122
5.2 Two-level iterative solver: overlapping additive Schwarz . . . 122
5.3 A posteriori estimator on the algebraic error 123
6 Main results 124
7 Proofs of the main results 125
7.1 Multilevel setting results 128
7.2 Two-level domain decomposition setting results 128
8 Conclusions 130

List of Figures

1 Illustration of the finite element method in one space dimension. Left: the domain $\Omega = (0,1)$, the mesh $\mathcal{T}_h$, where h is the mesh size (here $h = \frac{1}{8}$). Center: shape functions $\varphi_{h,0}$, $\varphi_{h,1}$, $\ldots$, $\varphi_{h,8}$ (here piecewise affine) that are used to define the discrete solution u_h. The unknown coefficients c_0, c_1, $\ldots$, c_8 (in red) are to be determined so that u_h can approximate the unavailable exact solution u (dotted line) in the discretization points. Right: we obtain the discrete solution u_h (in green) after determining the nine unknown coefficients by solving the algebraic system of nine linear equations obtained from the discrete problem. 2

2 Example of the domain Ω (left) being partitioned into a non-overlapping (center) and overlapping (right) decomposition. The subdomains Ω_1, Ω_2, are illustrated in yellow and blue, respectively. For the overlapping decomposition, the overlapping region shared by both subdomains is illustrated in light green. 4

3 The multigrid approach for a hierarchy of meshes illustrated on the left with $J = 3$ refinements. 5

4 Two types of mesh hierarchies generated by $J = 3$ refinements of a quasi-uniform coarse mesh $\mathcal{T}_0$. Top: Uniform mesh refinement, where each triangle of the previous level is subdivided into four congruent triangles. Bottom: Adaptive mesh refinement by using the newest bisection algorithm, cf. Sewell [1972]. 5

5 Example of a mesh and space hierarchy for a number of refinements $J = 4$. 9

6 Generalization of the multilevel approach of Definiton v.1 used to define a linear solver for a hierarchy of meshes illustrated on the left for $J = 3$. 14

7 Illustration for $J = 3$ of one iteration of the adaptive local smoothing solver: one full-smoothing and one adaptive-smoothing V-cycle substeps 18

8 [Peak test case, $J=2$, $p_0=1$, $p_1=p_2=6$, $\theta=0.95$] Comparing the algebraic error distribution (left) to the local error indicators (right) (levels $j=1$ top, $j=2$ bottom). Voronoi cells correspond to patch values, and the ones with the red border are marked for local smoothing. 18

9 [Different tests, $J = 3$, $p_0 = 1$, $p_1 = 1$, $p_2 = 2$, $p_3 = 3$, $\theta = 0.95$, $\gamma = 0.7$] Adaptive local smoothing: coarsest level marked or not and percentages of patches marked for each level $1 \leq j \leq J$ (Y-axis). Iterations of the adaptive local smoothing solver (X-axis). 20

10 [Different tests, $J=3$, $p_0=1$, $p_1=1$, $p_2=2$, $p_3=3$, $\theta=0.95$, $\gamma=0.7$] Adaptive local smoothing: decrease of the relative energy norm of the algebraic error $\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J-u_J^i)\|/\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla u_J\|$ in the full-smoothing substep and adaptive local smoothing substep in each iteration. . . 20

1.1 Illustration of the set $\mathcal{B}_j$. The mesh $\mathcal{T}_{j-1}$ and its refinement $\mathcal{T}_j$ are given by full and dotted lines, respectively. 31

1.2 Illustration of degrees of freedom ($p' = p = 2$) for the space $V_{j,0}^{\mathbf{b}}$ associated to the "small" patch $\mathcal{T}_{j,0}^{\mathbf{b}}$ (left) and for the space $V_{j,1}^{\mathbf{a}}$ associated to the "large" patch $\mathcal{T}_{j,1}^{\mathbf{a}}$ (right). The mesh $\mathcal{T}_{j-1}$ and its refinement $\mathcal{T}_j$ are defined in bold and dotted lines, respectively. 32

1.3 Sine problem (1.32), w_1=$J(d+1)$, w_2=1: results of the solver (1.27) for $p' = p$ in (1.10a), "small" (left) and "large" (right) patches, and stopping criterion (1.35). Top: error contraction factors $\|\nabla(u_J - u_J^{i+1})\|/\|\nabla(u_J - u_J^i)\|$. Bottom: relative algebraic error $\|\nabla(u_J - u_J^i)\|/\|\nabla u_J\|$. 40

1.4 Peak problem (1.33), $w_1 = 4\sqrt{J}$, $w_2 = \infty$: results of the solver (1.27) for $p' = p$ in (1.10a), "small" (left) and "large" (right) patches, and stopping criterion (1.35). Top: error contraction factors $\|\nabla(u_J - u_J^{i+1})\|/\|\nabla(u_J - u_J^i)\|$. Bottom: relative algebraic error $\|\nabla(u_J - u_J^i)\|/\|\nabla u_J\|$. 40

1.5 L-shape problem (1.34), $w_1 = d + 1$, $w_2 = J$: results of the solver (1.27) for $p' = p$ in (1.10a), "small" (left) and "large" (right) patches, and stopping criterion (1.35). Top: error contraction factors $\|\nabla(u_J - u_J^{i+1})\|/\|\nabla(u_J - u_J^i)\|$. Bottom: relative algebraic error $\|\nabla(u_J - u_J^i)\|/\|\nabla u_J\|$. 41

1.6 Sine problem (1.32), "small" patches, $p' = p$: study of the contraction factor behavior with respect to the number of post-smoothing steps and damping weights for the solver (1.27). 42

2.1 Illustration of degrees of freedom ($p_j = 2$) for the space $V_j^{\mathbf{a}}$ associated to the patch $\mathcal{T}_j^{\mathbf{a}}$. 61

2.2 Illustration of the set $\mathcal{B}_j$; the refinement $\mathcal{T}_j$ (dotted lines) of the mesh $\mathcal{T}_{j-1}$ (full lines). 65

2.3 Graded meshes obtained by the newest-vertex bisection algorithm. Left: L-shape problem and $J = 10$. Right: Checkerboard $\mathrm{O}(10^6)$ and $J = 10$. The regions where the diffusion coefficient is constant are bordered by red lines. 71

2.4 Number of smoothing steps per level for the Checkerboard case, polynomial degree $p = 3$, number of mesh levels $J = 3$, diffusion coefficient jump $\mathcal{J}(\boldsymbol{\mathcal{K}}) = O(10^6)$, and mesh hierarchies with $p_j = 1$ and $p_j = p$, $j \in \{1, \ldots, J-1\}$. 72

2.5 Comparison between a fixed number of (block-Jacobi) smoothing steps ν on all levels (Definition 4.1 and its obvious modification for $\nu \geq 1$) and the adaptive number of smoothing steps of Definition 7.1. Number of iterations i_s, floating point operations given by (2.32) relative with respect to Definition 7.1, and the number of global synchronizations by (2.38). 73
2.6 Cube case: decay of the relative algebraic error (left) and the relative residual (right) for the hierarchy with $p_j = 1$, $j \in \{1, \ldots, J-1\}$, $J = 4$. The solver of Definition 4.1 is stopped at iteration $i = 40$. nDoFs: 5 501 for $p = 1$, 41 337 for $p = 2$, 136 693 for $p = 3$, 320 753 for $p = 4$. 75
2.7 Nested cubes case: decay of the relative algebraic error (left) and the relative residual (right) for the hierarchy with $p_j = 1$, $j \in \{1, \ldots, J-1\}$, $J = 4$. The solver of Definition 4.1 is stopped at iteration $i = 40$. nDoFs: 7 281 for $p = 1$, 55 649 for $p = 2$, 185 041 for $p = 3$, 435 393 for $p = 4$. 75
2.8 Checkers cubes: decay of the relative algebraic error (left) and the relative residual (right) for the hierarchy with $p_j = 1$, $j \in \{1, \ldots, J-1\}$, $J = 4$. The solver of Definition 4.1 is stopped at iteration $i = 40$. nDoFs: 5 425 for $p = 1$, 40 033 for $p = 2$, 131 473 for $p = 3$, 307 393 for $p = 4$. 75

3.1 Illustration of a patch $\mathcal{T}_j^{\mathbf{a}}$, the patch subdomain $\omega_j^{\mathbf{a}}$, and of the degrees of freedom for the space $V_j^{\mathbf{a}}$ with $p_j = 2$. 90
3.2 Illustration of the full-smoothing and adaptive-smoothing V-cycle substeps, $J = 3$. 90
3.3 Illustration of the set $\mathcal{B}_j$; the refinement $\mathcal{T}_j$ (dotted lines) of mesh $\mathcal{T}_{j-1}$ (full lines). 98
3.4 [L-shape, J=2, p_0=1, p_1=p_2=3, θ=0.95, γ=0.7] Comparing algebraic error distribution (left) to local error indicators (right) (levels $j = 1$ top, $j = 2$ bottom). Voronoi cells correspond to patch values, and the ones with the red border are marked for local smoothing. 100
3.5 [Peak, J=2, p_0=1, p_1=p_2=6, θ=0.95, γ=0.7] Comparing algebraic error distribution (left) to local error indicators (right) (levels $j = 1$ top, $j = 2$ bottom). Voronoi cells correspond to patch values, and the ones with the red border are marked for local smoothing. 101
3.6 [All tests, $J = 3$, $p_0 = 1$, $p_1 = 1$, $p_2 = 2$, $p_3 = 3$, $\theta = 0.95$, $\gamma = 0.7$] Convergence of Algorithm 1 in the relative energy norm of the algebraic error $\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla (u_J - u_J^i)\| / \|\boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla u_J\|$. 101
3.7 [Different tests, $J = 3$, $p_0 = 1$, $p_1 = 1$, $p_2 = 2$, $p_3 = 3$, $\theta = 0.95$, $\gamma = 0.7$] Local adaptive smoothing: coarsest level marked or not and percentages of patches marked for each level $1 \leq j \leq J$ (Y-axis). Iterations of Algorithm 1 (X-axis). Results for the L-shape test case are given in the separate Figure 3.8. 102
3.8 [L-shape, $J = 3$, $p_0 = 1$, $p_1 = 1$, $p_2 = 2$, $p_3 = 3$, $\gamma = 0.7$, varying θ] Local adaptive smoothing: coarsest level marked or not and percentages of patches marked for each level $1 \leq j \leq J$ (Y-axis). Iterations of Algorithm 1 (X-axis). 103

3.9 [All tests, $J=3$, $p_0=1$, $p_1=1$, $p_2=2$, $p_3=3$, $\gamma=0.7$, varying θ] Convergence of Algorithm 1 in the relative energy norm of the algebraic error $\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J - u_J^i)\|/\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla u_J\|$. 105

4.1 Illustration of the set $\mathcal{B}_j$; the refinement $\mathcal{T}_j$ (dotted lines) of the mesh $\mathcal{T}_{j-1}$ (full lines). 118

4.2 Illustration of a patch in the two level overlapping additive Schwarz method: coarse grid $\mathcal{T}_H$ (solid line), fine grid $\mathcal{T}_h$ (dashed line), the patch associated with the vertex $\mathbf{a} \in \mathcal{V}_H$ contains four coarse elements of $\mathcal{T}_H$ that share $\mathbf{a}$ and form a subdomain. The coarse (subdomains) patches are discretized with the fine grid $\mathcal{T}_h$. 122

List of Tables

1 Comparison of wRAS solver of Chapter 1 with the standard V-cycle multigrid employing one post-smoothing step with Jacobi iteration for three test problems of the Section 6 in Chapter 1. Number of iterations needed for the ℓ^2-norm of the algebraic residual vector to drop below 10^{-5} times the initial value. 11

2 Number of iterations i_{s} needed for the ℓ^2-norm of the algebraic residual vector to drop below 10^{-5} times the initial value, for different polynomial degrees p, number of mesh levels J, space hierarchies with two different intermediate polynomial degrees p_j, $j \in \{1, \ldots, J-1\}$, and jump in the diffusion coefficient $\mathcal{J}(\boldsymbol{\mathcal{K}})$. 13

3 Number of post-smoothing steps per level in each iteration for a given test case, polynomial degree $p = 3$, number of mesh levels $J = 3$, diffusion coefficient jump $\mathcal{J}(\boldsymbol{\mathcal{K}}) = O(10^6)$, and mesh hierarchies with intermediate polynomial degrees $p_j = 1$ and $p_j = p$, $j \in \{1, \ldots, J-1\}$. 16

4 Checkerboard O(10^6) problem: comparison of iteration numbers i_{s} and CPU times for different solvers. The horizontal/rising arrow denotes whether the polynomial degree per level remains the same/-gradually increases from the coarsest grid $p_0 = 1$ to the finest grid $p_J = p$. The number of pre- and post-smoothing steps are given in parantheses, and the smoothers are given by block-Jacobi (bJ), block Gauss–Seidel (bGS), pointwise Gauss–Seidel (GS), or PCG with incomplete Cholesky preconditioner. The number of iterations is limited to 80. 17

1.1 dAS construction (1.37): problems (1.32)–(1.34), $p' = p$ in (1.10a), "small" and "large" patches. i_{s}: the number of iterations needed to reach the stopping criterion (1.35). $\bar{\alpha}$: average error contraction factor given by (1.36). 41

1.2 Number of iterations needed to reach the stopping criterion (1.35): wRAS construction (1.38), problems (1.32)–(1.34), $p' = p$ in (1.10a), "small" and "large" patches, ν post-smoothing steps, and standard multigrid method with piecewise affine coarse solve (1.15), initialized by the coarse grid solution, no pre-smoothing, one post-smoothing step, and Jacobi (J) and Gauss–Seidel (GS) smoothers. 43

1.3 Comparison of various multilevel solvers (described in Section 6.3) for the L-shape case (1.34), i_{s} is the number of iterations to reach the stopping criterion (1.35). 45

2.1 Number of iterations i_s for different polynomial degrees p, number of mesh levels J, space hierarchies with two different p_j, $j \in \{1, \ldots, J-1\}$, and jump in the diffusion coefficient $\mathcal{J}(\boldsymbol{\mathcal{K}})$ 70
2.2 Number of iterations i_s for different polynomial degrees p, number of mesh levels J, space hierarchies given by $p_j = p$, $j \in \{1, \ldots, J-1\}$; graded mesh hierarchies. 71
2.3 Number of smoothing steps per level in each iteration it for the Checkerboard case, $\theta = 0.2$, polynomial degree $p = 3$, number of mesh levels $J = 3$, diffusion coefficient jump $\mathcal{J}(\boldsymbol{\mathcal{K}}) = O(10^6)$, and mesh hierarchies with $p_j = 1$ and $p_j = p$, $j \in \{1, \ldots, J-1\}$. 72
2.4 Estimated number of floating point operations given by (2.32) and number of iterations i_s for two singular test cases, different polynomial degrees p, number of mesh levels J, and space hierarchies with p_j, $j \in \{0, \ldots, J\}$. 73
2.5 Checkerboard O(10^6) problem: comparison of iteration numbers i_s and CPU times for different solvers. The horizontal/rising arrow denotes whether the polynomial degree per level remains the same/-gradually increases. The number of pre- and post-smoothing steps are given in parantheses, and the smoothers are given by block-Jacobi (bJ), block Gauss–Seidel (bGS), pointwise Gauss–Seidel (GS), or PCG with incomplete Cholesky preconditioner. The number of iterations is limited to 80. 76

3.1 Number of iterations (number of adaptive-smoothing substeps in brackets) for various choices of the parameter γ in (3.14). The marking parameter in (3.13) is set as $\theta = 0.95$ 104
3.2 Number of iterations (number of adaptive-smoothing substeps in brackets) for various choices of marking parameter θ in (3.13). The parameter γ from (3.14) is set as $\gamma = 0.7$ 106
3.3 [L-shape, $J = 3$, $\theta = 0.95$, $\gamma = 0.7$] Study of sensitivity with respect to the shape regularity of the mesh (minimal angle of mesh elements) for the local adaptive smoothing solver. 107

"*Truth is much too complicated to allow anything but approximations.*"
- John Von Neumann

Introduction

We live in a world full of imperfections, but as long as they are *acceptable*, we can carry on with our lives perfectly fine: even though none of our watches are exact, we live our daily lives as if they were. In mathematics, this idea of adopting what is good enough corresponds to the concept of *approximation* and it is often our best solution when tackling physical problems modeled through partial differential equations. Thus, instead of the unavailable exact solution of such a problem, we develop computable mathematically-based approximations. But how to decide what is a *good-enough* approximation? *A posteriori analysis*, which uses the available outcome of the computations, plays an essential role to treat this question by identifying the magnitude but also the source and nature of the error between the unavailable exact solution and its constructed approximation. This information is valuable in order to control the overall error and develop new approaches to improve the approximation as efficiently as possible.

i Finite element method

While the analytic solution of a partial differential equation defined for a physical domain is usually not accessible, one can search for an approximation of the solution in a *finite-dimensional* space. This step is accomplished through *discretization* methods. Thus, we go from a continuous problem with infinitely many unknowns to a discrete problem with a finite number of unknowns, which is suitable to be handled with the help of computers. One of the most popular and versatile discretization methods is the finite element method, see e.g. Ciarlet [1978], Ern and Guermond [2004], or Brenner and Scott [2008]. To define the method, first, the computational domain is partitioned in simple subdomains, for instance simplices, referred to as a *mesh*. Then, simple shape functions, for instance piecewise polynomials, are defined on each subdomain, referred to henceforth as element. The guiding principle of the finite element method is to be able to approximate the exact solution through a combination of these shape functions. This typically leads to the problem of determining the unknown coefficients needed to weigh each of these shape functions before summing them together to obtain our approximation. We illustrate this idea in a simple one-dimensional case in Figure 1.

In particular, the accuracy of the discrete solution depends on how small the mesh size h is chosen as well as on the polynomial degree of approximation p, see e.g. Szabó and Babuška [1991] or Šolín et al. [2004]. However, initially choosing a very fine mesh or high polynomial degree is often not the best approach, since this implies a non-negligible computational cost, even for areas of the domain where

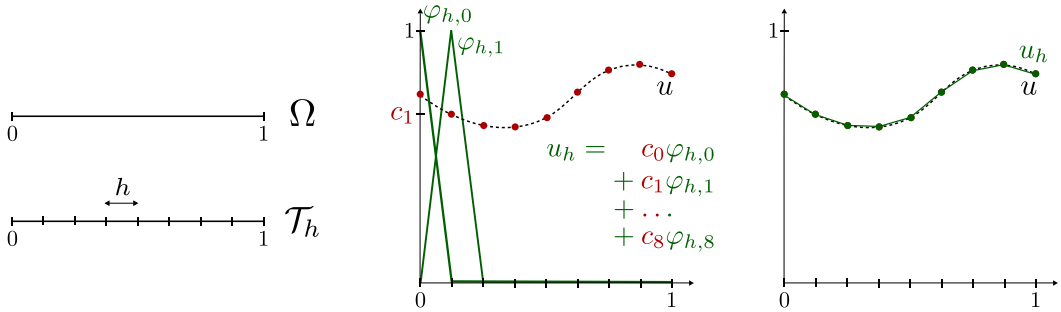


Figure 1: Illustration of the finite element method in one space dimension. Left: the domain $\Omega = (0, 1)$, the mesh $\mathcal{T}_h$, where h is the mesh size (here $h = \frac{1}{8}$). Center: shape functions $\varphi_{h,0}$, $\varphi_{h,1}$, $\dots$, $\varphi_{h,8}$ (here piecewise affine) that are used to define the discrete solution u_h. The unknown coefficients c_0, c_1, $\dots$, c_8 (in red) are to be determined so that u_h can approximate the unavailable exact solution u (dotted line) in the discretization points. Right: we obtain the discrete solution u_h (in green) after determining the nine unknown coefficients by solving the algebraic system of nine linear equations obtained from the discrete problem.

the same accuracy can be achieved with fewer unknowns. There have been many contributions dedicated to the topic, where the idea is to begin by a simple configuration of the mesh and polynomial degree distribution and then adaptively enrich areas of the domain by either refining the mesh and/or increasing the polynomial degree. This approach is referred to as *hp*-adaptive finite element method.

Work on this subject traces back to the pioneering work on the *h*-, *p*-, and *hp*-version of finite element method in Gui and Babuška [1986a,b,c], and interest and development has continued over the years. Closely related is the progress made in a posteriori analysis for finite element methods, see e.g. Babuška and Rheinboldt [1978], Ainsworth and Oden [2000], or Verfürth [2013] for an overview. These approaches can be combined together so that the refinement decision is steered adaptively by a posteriori estimates. To mention a few contributions, see e.g., Carstensen et al. [2014] for axioms of adaptivity in an abstract *h*-refinement setting, Mitchell and McClain [2014] for a comparison of *hp*-adaptive strategies, Morin et al. [2002], Cascón et al. [2008], Becker and Mao [2009], Feischl et al. [2014], Bespalov et al. [2017], and Canuto et al. [2017] for convergence and quasi-optimality/optimality results, Dolejší et al. [2016] for a polynomial-degree-robust a-posteriori-steering, Daniel et al. [2018] for computable and guaranteed error decrease bound, Gantner et al. [2018] for optimal convergence rates when the operator is nonlinear.

ii Linear solvers

The benefits of employing the *hp*-adaptive finite element method, and more generally the accuracy obtained through high-order approximations, lead us to study finite element discretizations of arbitrary polynomial degree p. Since increasing the polynomial degree implies increasing the number of shape functions to better capture the behavior of the unknown exact solution, this also leads to *algebraic* systems of increasing size: a larger number of unknown coefficients needs to be determined from solving a system of linear algebraic equations of the corresponding size.

ii.1 Direct solvers

When the number of unknowns is manageable computationally, we can rely on *direct* solvers to give us the coefficients needed to construct our discrete solution.

The algebraic system can be written in a matrix form and depending on the matrix, certain direct methods are more suitable than others. For instance, the first distinction to be made is whether or not the matrix is *sparse*, i.e. most coefficients of the matrix are zero. This distinction can lead to, when possible, savings in storage. One can then check if the matrix has certain properties, for example, being diagonal/banded/triangular/permutation of a triangular matrix, in order to use simpler and faster solvers (e.g., in the diagonal case it suffices to invert the coefficients of the diagonal, whereas for the triangular case a backward/forward solve suffices). In other cases, when the matrix is symmetric and positive definite, a Cholesky solver can be employed, whereas when the matrix is invertible but not symmetric, an LU solver can be used instead. Many novel approaches such as unifrontal-multifrontal methods have been derived in the past decade, see e.g., Yeralan et al. [2017], Duff et al. [2020] and the references therein. For more details on these methods and others, see e.g. Golub and Van Loan [1996] or Davis et al. [2016].

ii.2 Iterative solvers

Often, due to the size of the algebraic system, it is not possible or it is too time- or memory-consuming in terms of computer resources to employ direct solvers. In this case, one can resort to *iterative linear* solvers: algorithms which produce a sequence of approximations of the unknown coefficients, which should converge to the unknown exact coefficients.

We refer to the linear solvers that only require information from the given matrix as *algebraic* methods. Classic examples include the Jacobi, Gauss–Seidel, and successive overrelaxation (SOR) iterative solvers, see e.g. Kelley [1995], Varga [2000], or Saad [2003]. More powerful algebraic methods have been developed over the time, such as the algebraic multigrid method, see the seminal works of Brandt et al. [1985] and Ruge and Stüben [1987], or recent contributions in, e.g., Napov and Notay [2012] and the references therein.

Other linear solvers that use the given matrix and additionally smaller matrices assembled on sub-meshes and/or other information about the underlying mesh and problem are referred to as *geometric* methods. In this class of solvers, we mention the *domain decomposition* methods and (geometric) *multigrid* solvers which are treated in the following.

ii.2.1 Domain decomposition methods

The strategy of domain decomposition methods can be compared to a "divide and conquer" approach, see e.g. Dryja and Widlund [1990], Quarteroni and Valli [1999], Toselli and Widlund [2005], or Dolean et al. [2015] for an in-depth introduction. The main idea is to subdivide the original computational domain Ω into smaller subdomains (often with simpler geometry), where the associated algebraic systems can be solved directly. Depending on whether the subdomains are chosen to be overlapping

or not, the domain decomposition methods can be overlapping or non-overlapping, see Figure 2 for an illustration. We focus here on overlapping methods, which are based on the alternating Schwarz method introduced in Schwarz [1870]. This method can be summarized, for the historic case of two subdomains, by solving on the first subdomain a smaller-sized problem, then using the obtained approximation to improve the problem which should be solved in the second neighboring subdomain. This procedure continues iteratively, alternating from one subdomain to the other, until the subdomain approximations match in the overlapping region and a global approximation on the whole domain is obtained. To benefit from parallel computing, nowadays, the subdomain problems are typically solved independently, and then the information from the overlapping region is exchanged between subdomains. For an exhaustive presentation of Schwarz methods, see Gander [2008].

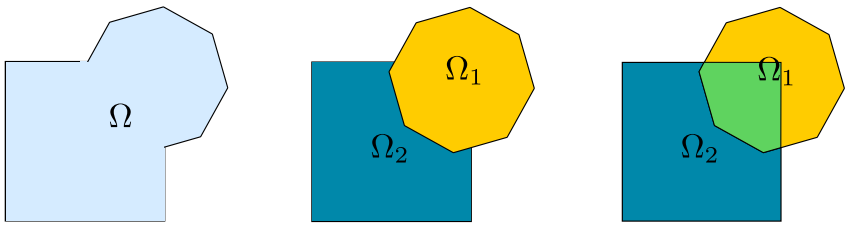


Figure 2: Example of the domain Ω (left) being partitioned into a non-overlapping (center) and overlapping (right) decomposition. The subdomains Ω_1, Ω_2, are illustrated in yellow and blue, respectively. For the overlapping decomposition, the overlapping region shared by both subdomains is illustrated in light green.

ii.2.2 Multigrid solvers

Geometric multigrid solvers, see e.g. Brandt and Livne [2011], Hackbusch [2003] or Briggs et al. [2000], and more generally multilevel solvers, see e.g. Oswald [1994], are amongst the most efficient and versatile linear solvers. The main idea of multigrid solvers is to capture complementary components of the algebraic error through the use of a *hierarchy of meshes*, see Figure 4 for an illustration of two different types of hierarchies, one obtained through uniform mesh refinement and the other from an adaptive mesh refinement approach. At each level of the hierarchy, a number of simple iterations, called *smoothings*, are employed to improve a given approximation. Importantly, at the coarsest mesh, the associated algebraic system is small enough in size to be solved *directly*. In order to convey information from one level to another, interpolation and restriction operators are crucial, which is where the geometric information of the mesh hierarchy is used. Depending on the order in which the levels are visited, an iteration of multigrid is composed of a given cycle. For example, for the V-cycle, the iteration begins at the finest level, then levels are visited from finest to coarsest, then revisited again from coarsest to finest, see Figure 3 for an illustration. We shall refer to the smoothing steps taking place before or after the coarse solve, respectively as pre-smoothing steps, post-smoothing steps. Importanly, one of the main features of multigrid solvers is their intrinsic robustness with respect to the mesh size, i.e. *h-robustness*. This means that the factor by which the algebraic error is divided on each step is *independent* of the mesh size parameter h.

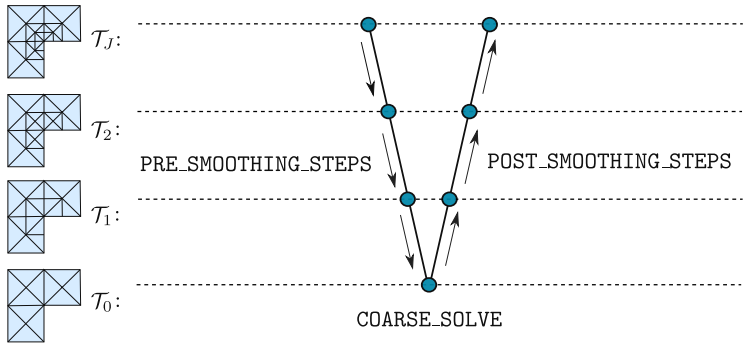


Figure 3: The multigrid approach for a hierarchy of meshes illustrated on the left with $J = 3$ refinements.

There are many similarities between multigrid and domain decomposition methods. In fact both are subspace correction methods, see e.g. Xu [1992]. The gaps between these two methods shrink when a coarse level is introduced in the domain decomposition setting and even more so when Schwarz methods are considered as smoothers in multigrid solvers, see e.g. Loisel et al. [2008].

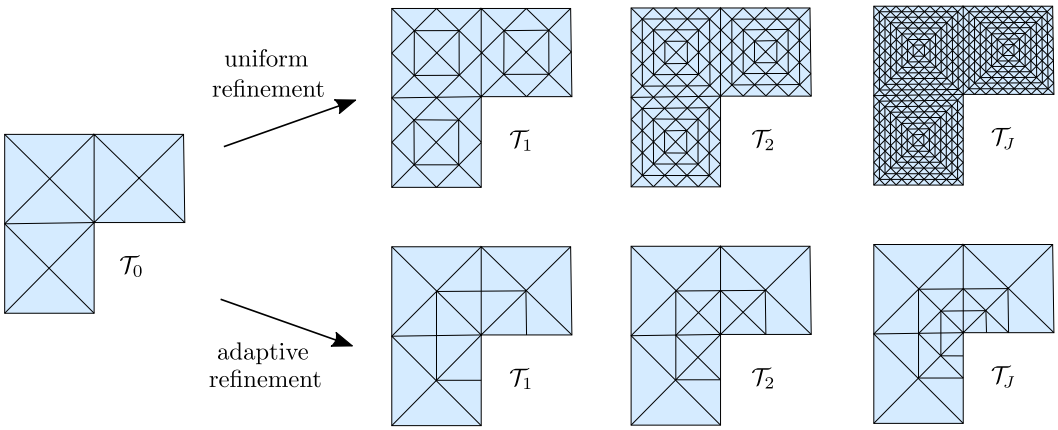


Figure 4: Two types of mesh hierarchies generated by $J = 3$ refinements of a quasi-uniform coarse mesh $\mathcal{T}_0$. Top: Uniform mesh refinement, where each triangle of the previous level is subdivided into four congruent triangles. Bottom: Adaptive mesh refinement by using the newest bisection algorithm, cf. Sewell [1972].

ii.2.3 Preconditioners

The methods we mentioned above, both algebraic and geometric, can be used either as iterative solvers, or their procedure can be modified (if needed) to be symmetric, so that they can serve as *preconditioners* for, e.g., the conjugate gradient algorithm of Hestenes and Stiefel [1952]. Using preconditioning can accelerate the convergence of the solver, see e.g. [Xu, 1992, Proposition 2.2]. For cases when the procedure is not symmetric, one can also resort to the use of the GMRES method introduced in Saad and Schultz [1986], or the BiCGSTAB method developed in van der Vorst [1992]. These methods belong to the same family of Krylov subspace methods. Note, however, that the performance of the original methods degrades when the

mesh size h decreases (they are not h-robust), but this is amended once h-robust preconditioners are used.

ii.3 Adaptive linear solvers

Adaptivity in linear solvers can be understood in various ways depending primarily on the nature of the solver that is being considered. In the family of algebraic multigrid methods, there have been several works which develop adaptive smoothed aggregation to build a coarser linear system, for example by determining near-kernel components, see e.g. Brezina et al. [2006], or by path covers using generalized a posteriori error estimates, see Hu et al. [2019], see also the references therein. Other techniques include the adaptive construction of preconditioners, see e.g., Anciaux-Sedrakian et al. [2020], which relies on a posteriori error estimates of the algebraic error used in, e.g., Papež et al. [2018]. Another recent approach is the adaptive multilevel Krylov method developed in Kehl et al. [2019], where the number of iterations performed on each level is chosen through a theoretically-derived criterion.

ii.4 Robustness with respect to the polynomial degree p

Due to the role of the linear solvers in obtaining a computable approximation of the discrete solution, their behavior with respect to the discretization parameters, mesh size h and polynomial degree p, should be considered. While we mentioned that certain solvers are robust with respect to the mesh size h, e.g. geometric multigrid solvers and certain preconditioners, we now focus on the behavior of solvers with respect to p. Very often the performance of linear solvers degrades with the increase of the polynomial degree, i.e., the algebraic error decreases in each iteration more slowly when p increases. When the solver is immune to this degradation, we refer to it as a *p-robust* solver.

In Quarteroni and Sacchi Landriani [1988], a p-robust domain decompostion method was presented for a specific domain configuration, and later advances on the topic were made in Pavarino [1994] for quadrilateral/hexahedral meshes, where a p-robust domain decomposition method using additive Schwarz was introduced. The generalization of this result for triangular/tetrahedral meshes was achieved by Schöberl et al. [2008] and the additive Schwarz method was used here to construct a p-robust preconditioner. For results for multigrid solvers also covering more general meshes, see e.g., Antonietti et al. [2018] and Antonietti and Pennesi [2019], where p-robustness is achieved when the number of smoothing steps is chosen sufficiently large.

For a computational survey on multigrid solvers for high-order discretizations, see e.g. Sundar et al. [2015].

iii Two central building blocks for the results of the thesis

We wish to acknowledge the important role of the two following works in the development of our theoretical results: 1) The p-robust results of this thesis crucially rely on the p-robust stable decomposition based on local problems of Schöberl et al.

[2008]. In Schöberl et al. [2008] this is developed on one given mesh (one-level), whereas we generalize it here to the multilevel setting. Moreover, the analysis in Schöberl et al. [2008] serves to construct a (symmetric) preconditioner, whereas in our work, the decomposition is a crucial ingredient of a standalone (non-symmetric) solver. 2) The multilevel *piecewise affine* stable decomposition of Xu et al. [2009], in particular for graded meshes generated by bisection. In Xu et al. [2009] the finite element hierarchy only uses order p approximations in the finest level and the estimates are not p-robust, whereas in our work, the hierarchy can be more general as long as the level-wise finite element spaces are nested, and the estimates are p-robust. By combining the above two results together, we obtain a p-robust multilevel stable splitting, essential for our analysis.

iv Model problem and its discretization

In this thesis, we will consider of a second-order elliptic diffusion problem posed over $\Omega \subset \mathbb{R}^d$, $d \in \{1,2,3\}$, an open bounded polytope with a Lipschitz-continuous boundary. Below, $f \in L^2(\Omega)$ denotes the source term and $\boldsymbol{\mathcal{K}} \in [L^\infty(\Omega)]^{d\times d}$ is a bounded tensor-valued diffusion coefficient taking symmetric and uniformly positive definite values. The problem consists in finding $u : \Omega \to \mathbb{R}$ such that

$$\begin{aligned} -\nabla \cdot \big(\boldsymbol{\mathcal{K}} \nabla u\big) &= f \quad \text{in } \Omega, \\ u &= 0 \quad \text{on } \partial\Omega. \end{aligned} \tag{1}$$

The continuous primal weak formulation of problem (1) consists in finding $u \in H^1_0(\Omega)$, such that

$$(\boldsymbol{\mathcal{K}} \nabla u, \nabla v) = (f, v) \quad \forall v \in H^1_0(\Omega), \tag{2}$$

where $(\cdot,\cdot)$ is the $L^2(\Omega)$ or $[L^2(\Omega)]^d$ scalar product. The existence and uniqueness of the solution of (2) follows from the Riesz representation theorem and assumptions on the data.

After fixing a matching simplicial mesh $\mathcal{T}_J$ of Ω and an integer $p \geq 1$, we can introduce the finite element space of continuous piecewise p-degree polynomials

$$V_J^p := \mathbb{P}_p(\mathcal{T}_J) \cap H^1_0(\Omega), \tag{3}$$

where $\mathbb{P}_p(\mathcal{T}_J) := \{v_J \in L^2(\Omega), v_J|_K \in \mathbb{P}_p(K)\ \forall K \in \mathcal{T}_J\}$.

The discretization of problem (2) leads to searching for $u_J \in V_J^p$ such that

$$(\boldsymbol{\mathcal{K}} \nabla u_J, \nabla v_J) = (f, v_J) \quad \forall v_J \in V_J^p. \tag{4}$$

Upon introducing a basis of V_J^p, the discrete problem (4) can be rewritten in a matrix form; however, the newly written problem would then be basis-dependent. Throughout this thesis, we opt to work with the functional *basis-independent* writing (4).

v A posteriori point of view and goals of the thesis

The approach and focus of this thesis is driven by *a posteriori analysis*. In a nutshell, we develop a posteriori error estimates of the *algebraic error*

$$\big\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla (u_J - u_J^i)\big\|,$$

where u_J is the exact (unknown) solution of (4) and $u_J^i \in V_J^p$ is its arbitrary approximation. The a posteriori estimator η_{alg}^i is computable from u_J^i (by a procedure equivalent to a V-cycle multigrid with no pre-smoothing and one post-smoothing step) and yields a *guaranteed lower bound* on the algebraic error:

$$\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J - u_J^i)\right\| \geq \eta_{\text{alg}}^i. \tag{5}$$

A salient feature of our approach is that we simultaneously use the construction of the estimator to define a linear solver. If u_J^i is the current iterate for the iteration counter i, then the linear solver constructs the next iterate u_J^{i+1} from u_J^i as

$$u_J^{i+1} := u_J^i + \text{“update”}, \tag{6}$$

where the *same* procedure which constructs the a posteriori estimator η_{alg}^i also gives us the solver update.

The procedure we develop in this thesis is more precisely designed in a way that links the estimator and the solver as

$$\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J - u_J^{i+1})\right\|^2 = \left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J - u_J^i)\right\|^2 - \left(\eta_{\text{alg}}^i\right)^2, \tag{7}$$

where, recall, u_J is the exact (unknown) solution of the linear system (4). The first important property of our approach, following from (7), is that the a posteriori estimator is a *guaranteed lower bound* of the algebraic error, i.e. (5) holds.

One natural follow-up question is: is this estimator also *efficient*, i.e. is η_{alg}^i also an upper bound, up to a constant, of the algebraic error $\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J - u_J^i)\right\|$?

Owing to the link (7) between the solver and the estimator, this question is in fact equivalent to: does the solver *contract* the algebraic error, i.e. is $\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J - u_J^{i+1})\right\|$ strictly smaller than $\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J - u_J^i)\right\|$? This equivalence can be seen from:

$$\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J - u_J^{i+1})\right\|^2 \leq \alpha^2\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J - u_J^i)\right\|^2 \quad \text{(error contraction)} \tag{8}$$

$$\overset{(7)}{\Leftrightarrow} \left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J - u_J^i)\right\|^2 - \left(\eta_{\text{alg}}^i\right)^2 \leq \alpha^2\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J - u_J^i)\right\|^2$$

$$\Leftrightarrow \left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J - u_J^i)\right\|^2(1-\alpha^2) \leq \left(\eta_{\text{alg}}^i\right)^2 \quad \text{(estimator efficiency)}. \tag{9}$$

We show in this thesis that the answer to the above two equivalent questions is *yes*, and that the factor $0 < \alpha < 1$ is moreover independent of the polynomial degree p, i.e. *p-robust*.

In the upcoming chapters, we will see different ways of defining a posteriori estimators and algebraic solvers that have these desirable properties. We shall also exploit further the properties of the a posteriori estimator by proposing new *adaptive* approaches in algebraic solvers. Due to the inherent connection between estimators and solvers, we refer to these solvers as *a-posteriori-steered*.

v.1 Multilevel setting

In order to define the procedure that constructs both the estimator η_{alg}^i and the solver update in (6), linked by (7), we take a *multilevel* approach. This is necessary when it comes to estimating the algebraic error, as already pointed out in, e.g.,

Rüde [1993], and can also seen in the counterexample of [Papež et al., 2020, Section 2.1]: one-level a posteriori estimators of the algebraic error are ill-suited to the task. In fact, the mentioned numerical counterexample illustrates how the accuracy of the one-level estimator degrades, in particular when the mesh size h decreases.

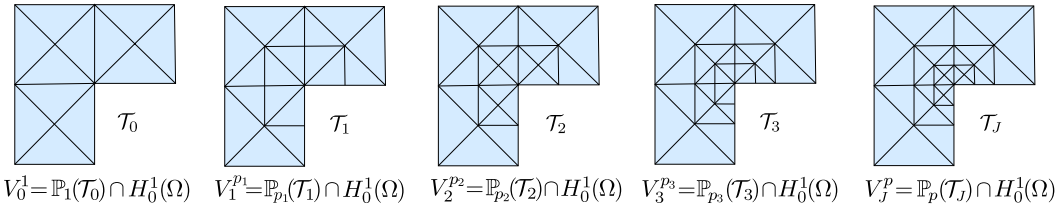


Figure 5: Example of a mesh and space hierarchy for a number of refinements $J = 4$.

For our multilevel setting, we assume we have at our disposal a fixed sequence of nested matching simplicial meshes $\{\mathcal{T}_j\}_{0 \le j \le J}$, $J \ge 1$, where $\mathcal{T}_J$ is the previously introduced finest mesh, and each $\mathcal{T}_j$ is a refinement of $\mathcal{T}_{j-1}$, $1 \le j \le J$. Our meshes in the hierarchy can be quasi-uniform or highly graded; we, however, require that: (i) the initial coarsest mesh $\mathcal{T}_0$ is quasi-uniform; (ii) all the meshes of the hierarchy are shape-regular; (iii) the maximum strength of refinement from one mesh to the next is bounded.

From the hierarchy of meshes, we then introduce a hierarchy of nested finite element spaces. By fixing an increasing sequence of level-wise polynomial degrees $1 = p_0 \le p_1 \le \ldots \le p_{J-1} \le p_J = p$, we can define:

$$\text{for } j = 0: \quad V_0^1 := \mathbb{P}_1(\mathcal{T}_0) \cap H_0^1(\Omega) \quad \text{(lowest-order space)}, \tag{10a}$$

$$\text{for } 0 < j < J: \quad V_j^{p_j} := \mathbb{P}_{p_j}(\mathcal{T}_j) \cap H_0^1(\Omega) \quad \text{(intermediate-order spaces)}, \tag{10b}$$

$$\text{for } j = J: \quad V_J^{p} := \mathbb{P}_{p}(\mathcal{T}_J) \cap H_0^1(\Omega) \quad \text{(highest-order space)}. \tag{10c}$$

An illustration of a possible hierarchy is given in Figure 5.

v.2 Multilevel procedure for constructing an a posteriori estimator of the algebraic error and a linear solver

Throughout this thesis we present several different constructions of an a posteriori estimator and solver based on the following common multilevel procedure, whose goal is to lift the algebraic residual. We first introduce the algebraic residual functional on V_J^p given by

$$v_J \mapsto (f, v_J) - (\boldsymbol{\mathcal{K}} \nabla u_J^i, \nabla v_J) \in \mathbb{R}, \quad v_J \in V_J^p. \tag{11}$$

Definition v.1 (Multilevel lifting of the algebraic residual)**.** *Given an arbitrary $u_J^i \in V_J^p$, perform the following steps:*

1. *Define the global lowest-order algebraic residual lifting $\rho_0^i \in V_0^1$ by*

$$(\boldsymbol{\mathcal{K}} \nabla \rho_0^i, \nabla v_0) = (f, v_0) - (\boldsymbol{\mathcal{K}} \nabla u_J^i, \nabla v_0) \quad \forall v_0 \in V_0^1. \tag{12}$$

   This corresponds to the global residual solve on the coarsest mesh.

   Set $\lambda_0^i := 1$.

2. *Go through the levels $j \in \{1, \dots, J\}$ in ascending order:*

   (a) *Let $\mathcal{V}_j$ be the set of vertices of the mesh $\mathcal{T}_j$. Given a vertex $\mathbf{a} \in \mathcal{V}_j$, let $\omega_j^{\mathbf{a}}$ be the open patch subdomain of all mesh elements of $\mathcal{T}_j$ that share the vertex $\mathbf{a}$. Let the local space $V_j^{\mathbf{a}}$ be defined by*

$$V_j^{\mathbf{a}} := \mathbb{P}_{p_j}(\mathcal{T}_j) \cap H_0^1(\omega_j^{\mathbf{a}}). \tag{13}$$

   (b) *For all $\mathbf{a} \in \mathcal{V}_j$, define $\rho_{j,\mathbf{a}}^i \in V_j^{\mathbf{a}}$ as the solution to the local intermediate or highest-order residual problem:*

$$\begin{aligned}(\boldsymbol{\mathcal{K}}\nabla\rho_{j,\mathbf{a}}^i, \nabla v_{j,\mathbf{a}})_{\omega_j^{\mathbf{a}}} = (f, v_{j,\mathbf{a}})_{\omega_j^{\mathbf{a}}} - (\boldsymbol{\mathcal{K}}\nabla u_J^i, \nabla v_{j,\mathbf{a}})_{\omega_j^{\mathbf{a}}} \\ - \sum_{k=0}^{j-1} \lambda_k^i (\boldsymbol{\mathcal{K}}\nabla\rho_k^i, \nabla v_{j,\mathbf{a}})_{\omega_j^{\mathbf{a}}} \quad \forall v_{j,\mathbf{a}} \in V_j^{\mathbf{a}}.\end{aligned} \tag{14}$$

   (c) *Define the algebraic residual lifting on level j by*

$$\rho_j^i := \sum_{\mathbf{a} \in \mathcal{V}_j} \rho_{j,\mathbf{a}}^i. \tag{15}$$

   If $\rho_j^i \neq 0$, define the optimal step-size on level j by

$$\lambda_j^i := \frac{(f, \rho_j^i) - (\boldsymbol{\mathcal{K}}\nabla(u_J^i + \sum_{k=0}^{j-1} \lambda_k^i \rho_k^i), \nabla\rho_j^i)}{\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_j^i\right\|^2}, \tag{16}$$

   otherwise set $\lambda_j^i := 1$.

One can notice that this multilevel procedure is *parallelizable* on each given level, since the local patch problems are mutually independent. It is, though, not parallelizable level-wise. From an algebraic perspective, the procedure is *additive* patch-wise for each given level and *multiplicative* level-wise.

Definition v.2 (A posteriori estimator of the algebraic error)**.** *Given any arbitrary $u_J^i \in V_J^p$, let the level-wise algebraic residual liftings $\{\rho_j^i\}_{0 \le j \le J}$ and the level-wise optimal step-sizes $\{\lambda_j^i\}_{0 \le j \le J}$ be constructed as in Definition v.1. Define the a posteriori estimator of the algebraic error associated to u_J^i as*

$$\eta_{\text{alg}}^i := \Big(\sum_{j=0}^{J} \big(\lambda_j^i \left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_j^i\right\|\big)^2\Big)^{\frac{1}{2}}. \tag{17}$$

Definition v.3 (A posteriori-steered solver)**.** *Initialize $u_J^0 = 0$ and let $i = 0$. Perform the following steps:*

1. *Construct the level-wise algebraic residual liftings $\{\rho_j^i\}_{0 \le j \le J}$ and the level-wise optimal step-sizes $\{\lambda_j^i\}_{0 \le j \le J}$ from u_J^i as in Definition v.1.*
2. *Update the current approximation* $$u_J^{i+1} := u_J^i + \sum_{j=0}^{J} \lambda_j^i \rho_j^i. \tag{18}$$
3. *If $u_J^{i+1} = u_J^i$, then stop the solver; otherwise increase $i := i+1$ and go to step 1.*

One iteration of the solver of Definition v.3 can be seen as a *geometric multigrid* V-cycle with zero pre- and a single post-smoothing step by *overlapping additive Schwarz* method, i.e. *block-Jacobi*, associated to patches of elements sharing the given vertex. Indeed, this is the interpretation of the local residual problems (14): note that except for the case $p_j = 1$, where there is only one unknown per patch (i.e., the smoothing is actually a simple Jacobi), the solution of the local problems (14) implies inverting sub-matrices (Jacobi blocks) of the original stiffness matrix. This is a central point in our construction which follows Schöberl et al. [2008] and allows us later to prove polynomial-degree robustness.

Note also that the coarse solve is cheaper than that of typical multigrid solvers, since we only employ lowest-order polynomials on the coarsest level.

Moreover, a salient feature of the solver that distinguishes it from classical multigrid approaches is the use of the *level-wise optimal step-sizes* defined in (16). These step-sizes, used previously in, e.g., the work of Heinrichs [1988], are determined through a line search and play a crucial role at the *error correction stage* of the multigrid by minimizing the algebraic error of the current mesh level before moving on to the next. Numerically, the role and importance of optimal step-sizes (even only a global one on level J, as used in Chapter 1) can be seen in, e.g., Table 1 (excerpt of Table 1.2 in Chapter 1): for the case $p = 1$, zero pre- and only one post-smoothing step, the only difference between our solver (denoted wRAS) and the usual V-cycle geometric multigrid with simple Jacobi smoothing (denoted MG(0,1)-J) is the use of the global optimal step-size as in (16), (18) for $j = J$. We see that employing the step-size not only helps the solver to be faster, but even makes it converge when the simple Jacobi iteration fails.

		Sine		Peak		L-shape	
J	p	wRAS	MG(0,1)-J	wRAS	MG(0,1)-J	wRAS	MG(0,1)-J
3	1	21	-	19	68	17	44
	3	15	-	15	-	12	-
	6	13	-	14	-	10	-
	9	13	-	14	-	10	-
4	1	23	-	20	-	18	-
	3	15	-	15	-	12	-
	6	13	-	14	-	10	-
	9	13	-	14	-	9	-
5	1	22	-	20	-	17	-
	3	15	-	15	-	12	-
	6	13	-	14	-	9	-
	9	13	-	13	-	8	-

Table 1: Comparison of wRAS solver of Chapter 1 with the standard V-cycle multigrid employing one post-smoothing step with Jacobi iteration for three test problems of the Section 6 in Chapter 1. Number of iterations needed for the ℓ^2-norm of the algebraic residual vector to drop below 10^{-5} times the initial value.

The level-wise minimization of the algebraic error through the use of optimal step-sizes leads to the following Pythagorean formula of the error decrease, which now gives all necessary details to (7).

Theorem v.4 (Pythagorean error representation of one solver step)**.** *For $u_J^i \in V_J^p$, let $u_J^{i+1} \in V_J^p$ be the next iterate constructed from u_J^i by the solver of Definiton v.3. Then*

$$\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J - u_J^{i+1})\|^2 = \|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J - u_J^i)\|^2 - \sum_{j=0}^{J}\big(\lambda_j^i\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_j^i\|\big)^2. \tag{19}$$

Proof. We go through the levels from finest to coarsest and we use the construction (16) of the level-wise optimal step-sizes λ_j^i:

$$\begin{aligned}
\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J - u_J^{i+1})\|^2 &\overset{(18)}{=} \Big\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\Big(\big(u_J - u_J^i - \sum_{j=0}^{J-1}\lambda_j^i\rho_j^i\big) - \lambda_J^i\rho_J^i\Big)\Big\|^2 \\
&\overset{(4)}{=} \Big\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\big(u_J - u_J^i - \sum_{j=0}^{J-1}\lambda_j^i\rho_j^i\big)\Big\|^2 - 2\lambda_J^i\Big[(f,\rho_J^i) - \Big(\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\big(u_J^i + \sum_{j=0}^{J-1}\lambda_j^i\rho_j^i\big), \nabla\rho_J^i\Big)\Big] \\
&\qquad + \Big(\lambda_J^i\Big\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_J^i\Big\|\Big)^2 \\
&\overset{(16)}{=} \|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J - u_J^i - \sum_{j=0}^{J-1}\lambda_j^i\rho_j^i)\|^2 - \big(\lambda_J^i\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_J^i\|\big)^2 = \dots \\
&= \|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\big(u_J - (u_J^i + \lambda_0^i\rho_0^i)\big)\|^2 - \sum_{j=1}^{J}\big(\lambda_j^i\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_j^i\|\big)^2 \\
&\overset{(12)}{=} \|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J - u_J^i)\|^2 - \sum_{j=0}^{J}\big(\lambda_j^i\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_j^i\|\big)^2 \\
&\overset{(17)}{=} \|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J - u_J^i)\|^2 - \big(\eta_{\mathrm{alg}}^i\big)^2.
\end{aligned} \tag{20}$$

□

v.3 Main results: p-robust efficiency of the a posteriori estimator and p-robust solver contraction

We prove that the a posteriori estimator of the algebraic error is efficient and that the associated a-posteriori-steered algebraic solver contracts the algebraic error at each iteration. These two main results can be presented as follows:

Theorem v.5 (p-robust reliable and efficient bound on the algebraic error)**.** *Let $u_J \in V_J^p$ be the (unknown) finite element solution of (4) and let $u_J^i \in V_J^p$ be arbitrary, $i \geq 0$. Let η_{alg}^i be given by Definition v.2. Then, in addition to $\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J - u_J^i)\| \geq \eta_{\mathrm{alg}}^i$ from (20), there holds*

$$\eta_{\mathrm{alg}}^i \geq \beta\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J - u_J^i)\|, \tag{21}$$

where $0 < \beta < 1$ depends on the space dimension d, the mesh shape regularity parameter, the ratio of the largest and the smallest eigenvalue of the diffusion coefficient $\boldsymbol{\mathcal{K}}$, at most linearly on the number of mesh levels J, and additionally on the mesh hierarchy parameters like the strength of refinement and quasi-uniformity of the coarse mesh (graded bisections) or all meshes (uniform refinement). In particular, β is independent of the polynomial degree p.

Theorem v.6 (p-robust error contraction of the multilevel solver)**.** *Let $u_J \in V_J^p$ be the (unknown) finite element solution of (4) and let $u_J^i \in V_J^p$ be arbitrary, $i \geq 0$. Take u_J^{i+1} to be constructed from u_J^i using one step of the multilevel solver of Definition v.3. There holds*

$$\left\|\mathcal{K}^{\frac{1}{2}}\nabla(u_J - u_J^{i+1})\right\| \leq \alpha\left\|\mathcal{K}^{\frac{1}{2}}\nabla(u_J - u_J^{i})\right\|. \tag{22}$$

where α is given by $\alpha = \sqrt{1-\beta^2}$ with β from (21).

Recall also the equivalence of Theorems v.5 and v.6 by (8)–(9); details can be found in Section 6 of Chapter 2.

The p-robustness results stated above are numerically illustrated in Table 2, see Table 2.1 of Chapter 2. Different test cases with different regularities, polynomial degrees p, and numbers of mesh levels J are considered. As we expect from Theorem v.6, the number of iterations stays in particular stable despite the increase of the polynomial degree p.

			Sine		Peak		L-shape		Checkerboard				Skyscraper			
			$\mathcal{K}=I$		$\mathcal{K}=I$		$\mathcal{K}=I$		$\mathcal{K}=I$		$\mathcal{J}(\mathcal{K})=O(10^6)$		$\mathcal{J}(\mathcal{K})=O(1)$		$\mathcal{J}(\mathcal{K})=O(10^7)$	
p_j			1	p	1	p	1	p	1	p	1	p	1	p	1	p
J	p	DoF	i_s	i_s	i_s	i_s	i_s	i_s	i_s	i_s	i_s	i_s	i_s	i_s	i_s	i_s
3	1	$2e^4$	19	19	19	19	21	21	18	18	18	18	19	19	19	19
	3	$1e^5$	29	13	28	14	29	11	27	11	28	11	31	13	31	13
	6	$6e^5$	30	13	30	14	26	9	24	9	25	10	28	11	28	11
	9	$1e^6$	31	14	30	14	23	9	23	9	23	9	26	10	26	10
4	1	$6e^4$	21	21	20	20	21	21	19	19	19	19	19	19	19	19
	3	$6e^5$	29	13	29	14	28	11	26	11	27	11	30	11	30	11
	6	$2e^6$	31	13	30	14	25	9	24	9	24	9	27	10	27	10
	9	$5e^6$	32	14	31	15	23	9	22	9	23	9	25	9	25	9

Table 2: Number of iterations i_s needed for the ℓ^2-norm of the algebraic residual vector to drop below 10^{-5} times the initial value, for different polynomial degrees p, number of mesh levels J, space hierarchies with two different intermediate polynomial degrees p_j, $j \in \{1, \ldots, J-1\}$, and jump in the diffusion coefficient $\mathcal{J}(\mathcal{K})$.

v.4 Alternative approaches

We can obtain different linear solvers when we modify the multilevel construction of Definition v.1.

For instance, one can decide to modify the definition of "patches" used to fix the local spaces (13) into larger subdomains. The assembly of the solutions of local problems (14) per level can also be modified depending on which smoothing method we employ. For example, in Chapter 1 we consider damped additive Schwarz smoothing, whereas in Chapter 2 and Chapter 4, we use additive Schwarz, and in Chapters 1 and 3 we also study the numerically better performing weighted restricted additive Schwarz.

The level-wise contributions (15) can also be combined differently to set the update of the solver in (18): in Chapter 1, only one global optimal step-size is used, which still enables us to have the desired connection (7) between estimator and

solver. This was improved in the later chapters by the use of the step-sizes (16) that are optimal level by level. In Chapter 1, we also study variants that are additive not only on each mesh but also level-wise, thus being fully parallelizable. Equally in Chapter 1, variants leading to local smoothing are mentioned.

A generalized version of the multilevel construction of Definition v.1 for defining p-robust multilevel solvers is illustrated in Figure 6. The module `COARSE_GLOBAL_LIFTING` refers to the solution of the coarse global problem (12); `SMOOTH_PATCHES` encapsulates the choice of patches used in the local spaces (13), local problems (14), and assembly (15); `COMBINE` refers to the combination of level-wise contributions into the solver update in (18).

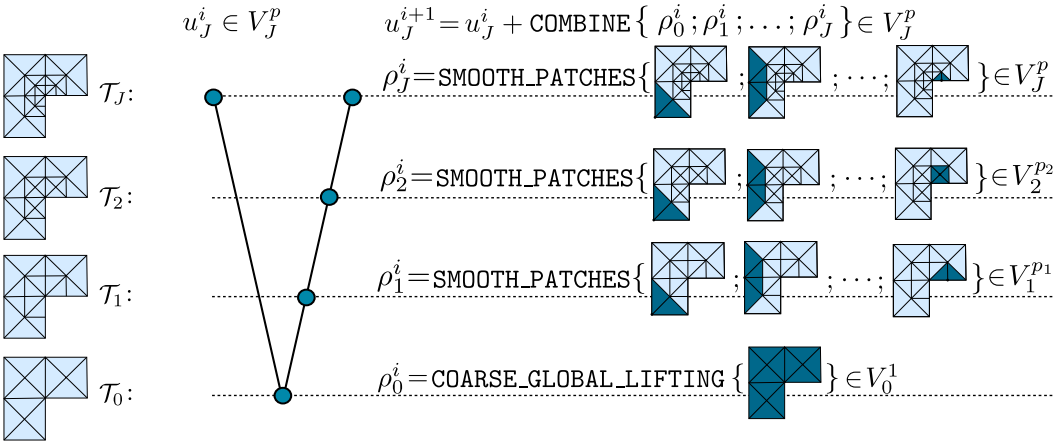


Figure 6: Generalization of the multilevel approach of Definiton v.1 used to define a linear solver for a hierarchy of meshes illustrated on the left for $J = 3$.

v.5 Extension to the mixed finite element method

In case of homogeneous Neumann boundary conditions instead of homogeneous Dirichlet ones, the model problem (1) can be equivalently written in a dual formulation: find $\mathbf{u} \in \mathbf{V}^f$ so that

$$(\boldsymbol{\mathcal{K}}^{-1}\mathbf{u}, \mathbf{v}) = 0 \quad \forall \mathbf{v} \in \mathbf{V}^0, \tag{23}$$

where $\mathbf{V}^f := \{\mathbf{v} \in \mathbf{H}(\mathrm{div}; \Omega),\ \nabla \cdot \mathbf{v} = f,\ \mathbf{v} \cdot \mathbf{n} = 0 \text{ on } \partial\Omega\}$ and $\mathbf{V}^0 := \{\mathbf{v} \in \mathbf{H}(\mathrm{div}, \Omega),\ \nabla \cdot \mathbf{v} = 0,\ \mathbf{v} \cdot \mathbf{n} = 0 \text{ on } \partial\Omega\}$, with the boundary condition to be understood in an appropriate sense. This writing is useful when we are interested in the dual variable $\mathbf{u}$ instead of the primal one, usually motivated by the physical meaning of the variable (often, $\mathbf{u}$ represents a fluid velocity).

There are many options to choose from to discretize the newly written model problem (23), see e.g. Boffi et al. [2013]. For simplicity, let f be a piecewise polynomial. Then, we just denote for now $\mathbf{V}_J^f \in \mathbf{V}^f$ and $\mathbf{V}_J^0 \in \mathbf{V}^0$ "suitable" discrete spaces of a given polynomial order, allowing us to write the discrete problem: find $\mathbf{u}_J \in \mathbf{V}_J^f$ so that

$$(\boldsymbol{\mathcal{K}}^{-1}\mathbf{u}_J, \mathbf{v}_J) = 0 \quad \forall \mathbf{v}_J \in \mathbf{V}_J^0. \tag{24}$$

Like in the primal formulation setting, we shall work with the functional formulation in order to avoid any dependence on the choice of the basis functions.

To tackle problem (24), we take a similar approach as in Ewing and Wang [1992]. The first step we take to define a linear solver to treat (24), is to construct $\mathbf{u}_J^f \in \mathbf{V}_J^f$, for example, as done in [Ewing and Wang, 1992, Theorem 3.1]. Since we have an initial guess which has the correct divergence, i.e., which belongs to $\mathbf{V}_J^f$, we aim to extend the solver of Definition v.3 to the current setting by constructing updates that are divergence-free to approximate $\mathbf{u}_J$.

The parts that need to be adapted from the multilevel construction of Definition v.1 are the coarse problem, local spaces, and local problems. However, the main approach in constructing the multilevel solver remains the same and importantly, we can prove, in two space dimensions, that this solver also contracts the algebraic error independently of the polynomial degree at each iteration. This polynomial-degree-robustness result is the main difference from the work done in Ewing and Wang [1992]; moreover, since we use level-wise optimal step-sizes, no relaxation / damping parameters are needed for the solver and the analysis.

vi Adaptivity in a-posteriori-steered solvers

The second main point of this thesis is the development of *adaptive* approaches in a-posteriori-steered solvers.

Since we are working in the context of geometric multigrid methods, the two adaptive approaches we explore are the following: (i) adaptive choice of the number of smoothing steps per level, (ii) adaptive local smoothing. While the closely related topics of a variable number of smoothing steps per level, see e.g. Bramble and Pasciak [1987] or Thekale et al. [2010], and local smoothing, see e.g. Bai and Brandt [1987], McCormick [1989], Rüde [1993], or Xu et al. [2009], are not new, our novelty here is to explore them through the lens of adaptivity based on *a posteriori estimates* of the *algebraic error*.

We explain now in more detail the adaptive approaches we present in this thesis.

vi.1 Adaptive number of post-smoothing steps

The starting point for this approach is in the *Pythagorean formula* (19) that we restate for the reader's convenience:

$$\begin{aligned}\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J-u_J^{i+1})\right\|^2 &= \left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J-u_J^{i})\right\|^2 - \sum_{j=0}^{J}\left(\lambda_j^i\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_j^i\right\|\right)^2\\ &= \left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J-u_J^{i})\right\|^2 - \left(\eta_{\mathrm{alg}}^i\right)^2.\end{aligned}$$

This important property is satisfied by the solver introduced in Definition v.3 and can be understood in the following way: while applying one step of the solver, we know exactly the contraction of the algebraic error on all previous levels. After employing *one mandatory* post-smoothing step in the current level, we can compare the error decrease with that of previous levels on-the-fly. Thus, we can decide whether or not another post-smoothing step is *needed* based on the computable terms that constitute the a posteriori estimator η_{alg}^i of Definition v.2. If the decrease after the mandatory post-smoothing step is higher than a user-prescribed portion of the decrease made by the previous levels and previous smoothings on the given

level, we decide to employ another post-smoothing step before going to the next level.

This criterion is a greedy-type one, where the goal is to decide when the algebraic error of the given level is important enough to require another post-smoothing step. Note also that since the criterion compares the current level error decrease with the decrease of *all* previous levels, we penalize the finest levels by making it more difficult for them to be smoothed many times. This evokes, for example, the approach of Bramble and Pasciak [1987], where the number of smoothing steps is doubled on successively coarser grids, thus avoiding more smoothing steps in finest levels thereby resulting in a computationally cheaper procedure.

	$p_j = 1$								$p_j = p$					
	it=1	it=2	it=3	it=4	it=5	it=6	it=7	it=8	it=1	it=2	it=3	it=4	it=5	it=6
level 0	1	1	1	1	1	1	1	1	1	1	1	1	1	1
level 1	3	3	3	3	3	3	3	3	3	4	4	4	4	4
level 2	3	3	3	3	3	3	3	3	2	1	1	1	1	1
level 3	3	4	4	4	4	4	4	4	2	2	2	2	2	1

Table 3: Number of post-smoothing steps per level in each iteration for a given test case, polynomial degree $p = 3$, number of mesh levels $J = 3$, diffusion coefficient jump $\mathcal{J}(\boldsymbol{\mathcal{K}}) = O(10^6)$, and mesh hierarchies with intermediate polynomial degrees $p_j = 1$ and $p_j = p$, $j \in \{1, \dots, J-1\}$.

Numerically, the adaptive approach is illustrated in Table 3, Table 2.3 in Chapter 2, where the number of post-smoothing steps on each level for each iteration is presented for a given test problem. We also compare the performance, both in terms of number of iterations and timing, of this approach compared to other multilevel solvers in Table 4, Table 2.5 in Chapter 2. Therein, MG(0,adapt)-bJ(wRAS) denotes the multigrid solver with zero pre- and an adaptive number of post-smoothing steps, that we have just described, equipped with a weighted restricted additive Schwarz (block-Jacobi) smoother. As the numerical results suggest, the solver outperforms the other considered methods, both in number of iterations and in timing, while preserving the p-robust nature.

vi.2 Adaptive local smoothing

If mesh adaptivity consists in *refining* the mesh in areas of the domain where the a posteriori estimator of the *discretization* error is large, the solver adaptivity counterpart is to *smooth* in levels and areas of the hierarchy of meshes where the a posteriori estimator of the *algebraic* error is large. This is what we explore with the adaptive local smoothing approach. Importantly, the approach that we introduce can be used in mesh hierarchies that are graded or uniform, as in e.g. Figure 4, as opposed to local smoothing multigrid which is designed for adaptively refined (graded) meshes and the smoothing is only applied around newly generated vertices.

Once again, we shall rely on the properties of our efficient a posteriori estimator of the algebraic error given in Definition v.2. First, note that

$$\sum_{\mathbf{a}\in\mathcal{V}_j} \|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho^i_{j,\mathbf{a}}\|^2_{\omega^{\mathbf{a}}_j} \overset{(14)}{\underset{(15)}{=}} (f,\rho^i_j) - \Big(\boldsymbol{\mathcal{K}}\nabla\big(u^i_J + \sum_{k=0}^{j-1}\lambda^i_k\rho^i_k\big), \nabla\rho^i_j\Big) \overset{(16)}{=} \lambda^i_j\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho^i_j\|^2, \quad (25)$$

		∼MG(0,1)-bJ $1 \to 1, p$		∼MG(0,1)-bJ $1, p \to p$		∼MG(0,adapt)-bJ $1, p \to p$		∼MG(0,adapt)-bJ (wRAS) $1 \nearrow p$		PCG(MG(3,3)-bJ) $p \to p$		MG(1,1)-PCG(iChol) $1 \nearrow p$		MG(0,1)-bGS $1 \to 1, p$		MG(3,3)-GS $1 \nearrow p$	
J	p	i_s	time	i_s	time	i_s	time	i_s	time	i_s	time	i_s	time	i_s	time	i_s	time
3	1	18	0.05 s	18	0.07 s	8	0.04 s	8	0.04 s	10	0.07 s	6	0.39 s	10	0.04 s	4	0.02 s
	3	28	0.96 s	11	0.50 s	6	0.43 s	6	0.41 s	3	0.57 s	22	3.43 s	11	2.62 s	6	0.34 s
	6	25	9.88 s	10	5.43 s	6	5.24 s	5	2.90 s	2	5.24 s	44	51.38 s	9	7.35 s	11	5.91 s
	9	23	45.87 s	9	27.01 s	6	25.25 s	4	13.86 s	2	36.95 s	80	5.22m	8	32.53 s	11	19.72 s
4	1	19	0.12 s	19	0.12 s	9	0.11 s	9	0.11 s	11	0.20 s	16	0.74 s	11	0.06 s	4	0.05 s
	3	27	3.85 s	11	2.07 s	6	1.89 s	7	1.62 s	3	2.34 s	44	27.48 s	10	9.64 s	5	1.37 s
	6	24	41.79 s	9	20.19 s	6	20.69 s	4	12.54 s	3	38.40 s	80	6.87m	9	34.78 s	6	14.44 s
	9	23	3.63m	9	2.13m	6	2.09m	3	49.84 s	2	2.24m	80	23.08m	8	1.72m	9	1.21m

Table 4: Checkerboard $\mathrm{O}(10^6)$ problem: comparison of iteration numbers i_s and CPU times for different solvers. The horizontal/rising arrow denotes whether the polynomial degree per level remains the same/gradually increases from the coarsest grid $p_0 = 1$ to the finest grid $p_J = p$. The number of pre- and post-smoothing steps are given in parantheses, and the smoothers are given by block-Jacobi (bJ), block Gauss–Seidel (bGS), pointwise Gauss–Seidel (GS), or PCG with incomplete Cholesky preconditioner. The number of iterations is limited to 80.

which means that our estimator is localized not only level-wise but also patch-wise:

$$\left(\eta^i_{\mathrm{alg}}\right)^2 = \sum_{j=0}^{J} \left(\lambda^i_j \|\boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \rho^i_j\|\right)^2 = \|\boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \rho^i_0\|^2 + \sum_{j=1}^{J} \lambda^i_j \left(\lambda^i_j \|\boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \rho^i_j\|^2\right)$$
$$\overset{(25)}{=} \|\boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \rho^i_0\|^2 + \sum_{j=1}^{J} \lambda^i_j \sum_{\mathbf{a} \in \mathcal{V}_j} \|\boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \rho^i_{j,\mathbf{a}}\|^2_{\omega^{\mathbf{a}}_j}.$$

From the efficiency of the a posteriori estimator η^i_{alg}, cf. Theorem v.5, we immediately have equivalence of the algebraic error with the localized version of the a posteriori estimator:

Corollary vi.1 (Equivalence of error–global estimator–local estimators)**.** *Let the assumptions of Theorem v.5 hold. Then*

$$\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla (u_J - u^i_J)\|^2 \approx \left(\eta^i_{\mathrm{alg}}\right)^2 = \|\boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \rho^i_0\|^2 + \sum_{j=1}^{J} \lambda^i_j \sum_{\mathbf{a} \in \mathcal{V}_j} \|\boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \rho^i_{j,\mathbf{a}}\|^2_{\omega^{\mathbf{a}}_j}. \tag{26}$$

Thus, if we employ one step of the solver of Definition v.3, we have at our disposal the efficient localized estimator in (26). Then, we can rely on a bulk-chasing criterion, cf. Dörfler [1996], to detect and mark the patches with increased error on all levels. The next step consists in employing a modified step of the solver which only applies smoothing in these marked patches, if an analysis-driven condition, based on the available a posteriori estimator, holds. The idea of this adaptive solver is illustrated in Figure 7.

We give here an illustration of how the bulk-chasing criterion marks the patches where we estimate the error to be increased. For a user-prescribed parameter $\theta \in (0, 1)$, we sort all patch-wise algebraic error estimators on all levels and select for

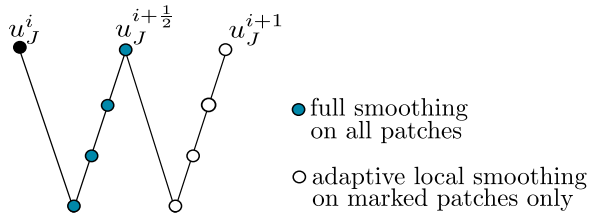


Figure 7: Illustration for $J = 3$ of one iteration of the adaptive local smoothing solver: one full-smoothing and one adaptive-smoothing V-cycle substeps

marking the smallest cardinality set of the coarsest level and vertex indices, $1 \leq j \leq J$, by the following bulk-chasing criterion:

$$\theta^2 \left(\left\| \boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \rho_0^i \right\|^2 + \sum_{j=1}^{J} \lambda_j^i \sum_{\mathbf{a} \in \mathcal{V}_j} \left\| \boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \rho_{j,\mathbf{a}}^i \right\|_{\omega_j^{\mathbf{a}}}^2 \right) \leq \sum_{j \in \mathcal{M}} \lambda_j^i \sum_{\mathbf{a} \in \mathcal{M}_j} \left\| \boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \rho_{j,\mathbf{a}}^i \right\|_{\omega_j^{\mathbf{a}}}^2, \quad (27)$$

where $\left\| \boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \rho_0^i \right\|$ appears on the coarsest level $j = 0$ if it is marked. For a given test case, we present in Figure 8 an example of how the distribution of the estimated algebraic error can faithfully follow the distribution of the true algebraic error. The problematic patches on each level have been marked for adaptive local smoothing (red border).

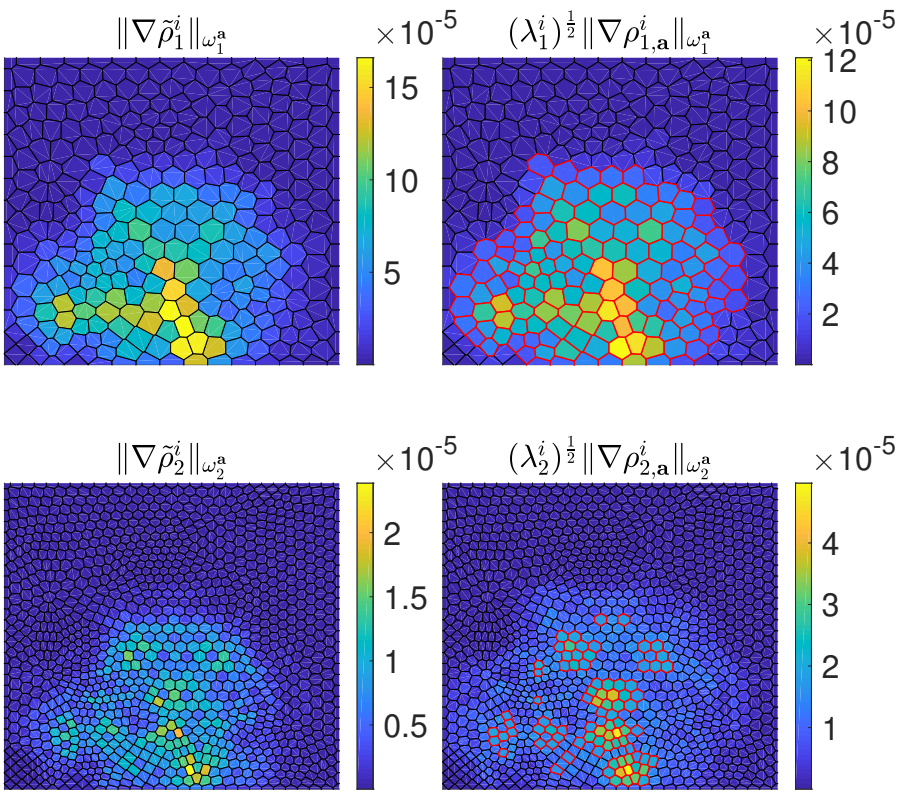


Figure 8: [Peak test case, J=2, p_0=1, p_1=p_2=6, θ=0.95] Comparing the algebraic error distribution (left) to the local error indicators (right) (levels $j = 1$ top, $j = 2$ bottom). Voronoi cells correspond to patch values, and the ones with the red border are marked for local smoothing.

We prove that this adaptive approach leads to a solver that in each of its sub-steps contracts the algebraic error p-robustly cf. Figure 7:

Theorem vi.2 (p-robust error contraction of the adaptive multilevel solver)**.** *Let $u_J \in V_J^p$ be the (unknown) solution of (4) and let $u_J^i \in V_J^p$ be arbitrary, $i \geq 0$. Let $u_J^{i+\frac{1}{2}} \in V_J^p$ be the update at the end of the full-smoothing substep. Then*

$$\big\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\big(u_J - u_J^{i+\frac{1}{2}}\big)\big\| \leq \alpha\big\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\big(u_J - u_J^{i}\big)\big\|. \tag{28}$$

When the following tests are satisfied:

$$\sum_{j\in\mathcal{M}}\lambda_j^i\sum_{\mathbf{a}\in\mathcal{M}_j}\Big(\sum_{k=j}^{J}\lambda_k^i\boldsymbol{\mathcal{K}}\nabla\rho_k^i,\nabla\rho_{j,\mathbf{a}}^i\Big)_{\omega_j^{\mathbf{a}}} \leq \gamma^2\sum_{j\in\mathcal{M}}\lambda_j^i\sum_{\mathbf{a}\in\mathcal{M}_j}\Big\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_{j,\mathbf{a}}^i\Big\|_{\omega_j^{\mathbf{a}}}^2, \tag{29}$$

$$\lambda_j^i \leq 2(d+1) \quad \forall j \in \{0,\dots,J\}, \tag{30}$$

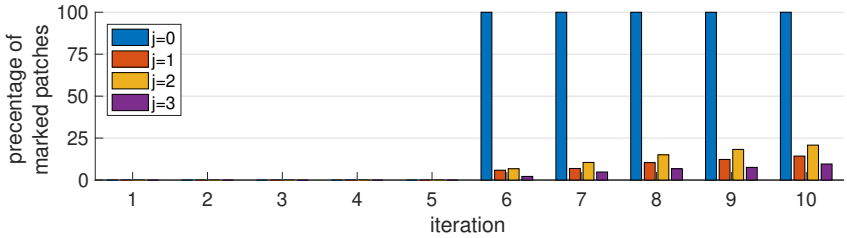
for $\gamma \in (0,1)$ a user-prescribed parameter, let $u_J^{i+1} \in V_J^p$ be the update at the end of the adaptive substep. Then

$$\big\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\big(u_J - u_J^{i+1}\big)\big\| \leq \tilde{\alpha}\big\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\big(u_J - u_J^{i+\frac{1}{2}}\big)\big\|. \tag{31}$$

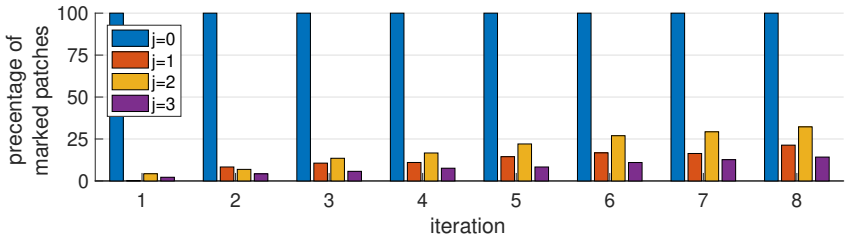
Here $0 < \alpha < 1$, $0 < \tilde{\alpha} < 1$ depend on the space dimension d, the mesh shape regularity parameter $\kappa_{\mathcal{T}}$, the number of mesh levels J, and the ratio of the largest and the smallest eigenvalues of the diffusion coefficient $\boldsymbol{\mathcal{K}}$, and additionally on the mesh hierarchy parameters like the strength of refinement and quasi-uniformity of the coarse mesh (graded bisections) or all meshes (uniform refinement). The dependence of the number of levels J is at most linear for α and cubic for $\tilde{\alpha}$. The factor $\tilde{\alpha}$ depends additionally on the marking parameter θ of (27) and the adaptivity test parameter γ. In particular, both α and $\tilde{\alpha}$ are independent of the polynomial degree p.

Numerically, we see from Figure 9, that even when the marking parameter is as high as $\theta = 0.95$, only a relatively small percentage of patches are marked for smoothing (in the Peak test case during the first 5 iterations, the test (29) is not satisfied, so that no adaptive substep it performed). Thus, it is beneficial to smooth in only this small portion of patches while obtaining a contraction of the algebraic error with a similar quality as smoothing in all the patches. This is seen in Figure 10, where the relative energy norm of the algebraic error with numerically very similar contraction factor in both the full-smoothing sub-steps and the adaptive-smoothing sub-steps.

Peak test case



Skyscraper test case (diff. contrast $O(10^2)$)



Skyscraper test case (diff. contrast $O(10^5)$)

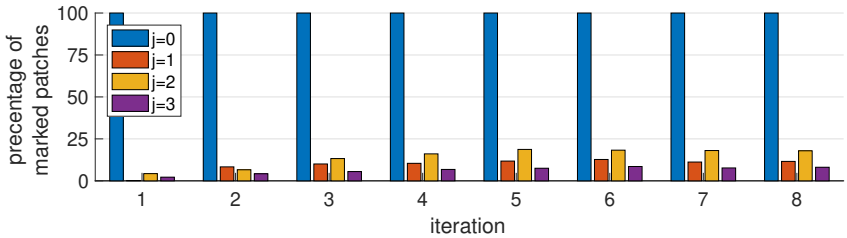


Figure 9: [Different tests, $J = 3$, $p_0 = 1$, $p_1 = 1$, $p_2 = 2$, $p_3 = 3$, $\theta = 0.95$, $\gamma = 0.7$] Adaptive local smoothing: coarsest level marked or not and percentages of patches marked for each level $1 \leq j \leq J$ (Y-axis). Iterations of the adaptive local smoothing solver (X-axis).

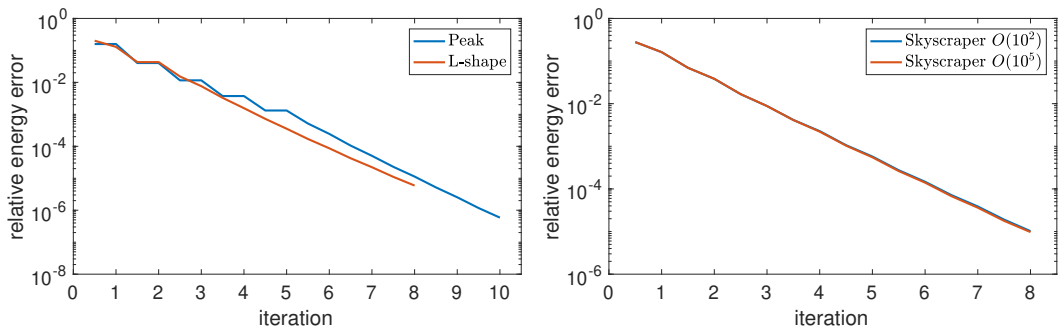


Figure 10: [Different tests, $J = 3$, $p_0 = 1$, $p_1 = 1$, $p_2 = 2$, $p_3 = 3$, $\theta = 0.95$, $\gamma = 0.7$] Adaptive local smoothing: decrease of the relative energy norm of the algebraic error $\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J - u_J^i)\|/\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla u_J\|$ in the full-smoothing substep and adaptive local smoothing substep in each iteration.

vii Contents and contributions of the thesis

The manuscript is constituted of four chapters. We now describe the contributions of each one.

vii.1 Chapter 1

This chapter consists of the article Miraçi et al. [2020], SIAM Journal on Numerical Analysis, 58, 5 (2020), 2856–2884, written with Jan Papež and Martin Vohralík.

In this chapter we lay the groundwork which will be reused also in the subsequent chapters. We develop an algebraic error estimator and a multigrid iterative linear solver, where the multilevel construction is the predecessor of that of Definition v.1. Here, only one optimal step-size is used, at the finest level J, and damping weights appear in the sum of the local problems (14) as well as in the level-wise assembly of contributions (15). These damping weights have to satisfy a compatibility condition and their role is to counterbalance the effect of overlapping patch-wise and level-wise contributions due to the additive Schwarz smoothing. Thus, the smoothing here is *damped additive Schwarz*. We also consider in this chapter two types of patches of different sizes which are used to define the local spaces: those of (13) and one-layer larger patches. Finally, we also introduce here the *weighted restricted additive Schwarz* smoothing which does not require damping weights and seems to perform better numerically.

The main results and novelties of Chapter 1 are: 1) the proof that the a posteriori estimator of the algebraic error we introduce based on the multilevel construction is *p-robustly efficient*; 2) the proof that the associated solver is *p-robustly contractive* at each iteration; 3) the equivalence of these results as in (8)–(9); 4) the proof that the algebraic error is also equivalent to an a posteriori estimator which is localized patch-wise and level-wise in the spirit of (26); 5) the analysis is done for a general nested hierarchy of unstructured and possibly highly graded simplicial meshes in two or three space dimensions; 6) the analysis is done under the minimal H^1-regularity of the weak solution.

To the best of our knowledge, this is the first work that proves p-robust multigrid convergence on triangular/tetrahedral meshes while requiring only one post-smoothing step.

vii.2 Chapter 2

This chapter consists of the article Miraçi et al. [2021a], SIAM Journal on Scientific Computing, DOI 10.1137/20M1349503, written with Jan Papež and Martin Vohralík.

In this chapter we introduce the improved version of the multilevel construction which is described here in Definition v.1. Therein, the *level-wise optimal step-sizes* (given by line search) are introduced to maximize error decrease from one level to the next. Additionally, after introducing the optimal step-sizes per level, we no longer need to use any damping parameters, whose tuning can be cumbersome. A simple and effective *adaptive strategy* which allows to choose the necessary *number of post-smoothing steps* on each level is also presented in this chapter. The advantages of

using optimal step-sizes and the adaptive approach are also seen in our numerical experiments. Finally, we emphasize that the main ideas of optimal step-size per level and adaptive number of post-smoothing steps are flexible approaches that can be used even in other geometric multigrid solvers. Implementation-wise, these ideas are easy to add to existing codes and alleviate the task of choosing the number of smoothing steps arbitrarily.

The main results of Chapter 2 are as: points 1)–3) and 5) presented for Chapter 1. Compared to Chapter 1, the novelties are: the optimal step-sizes lead to the explicit level-by-level error decrease formula (19); the a posteriori estimator we introduce is localized level-wise and patch-wise following (26); the analysis of Chapter 2 gives at most linear dependence on the number of mesh levels J under minimal H^1-regularity and complete independence of J in H^2-regularity setting.

vii.3 Chapter 3

This chapter consists of the article Miraçi et al. [2021b], Computational Methods in Applied Mathematics, DOI 10.1515/cmam-2020-0024, written with Jan Papež and Martin Vohralík.

In this chapter we use the localized a posteriori estimator of the algebraic error (26) of Chapter 2 to develop the new *adaptive local smoothing* strategy. The solver we consider uses one full-smoothing sub-step as the solver in Chapter 2. Then, thanks to the localized a posteriori estimator and a bulk-chasing criterion, we mark for an adaptive-smoothing sub-step only the patches where an increased error is estimated. Numerically, we observe that the adaptive local smoothing gives the same quality of error decrease as smoothing in all patches despite a much lower cost since smoothing is applied in (much) fewer patches.

The main results of Chapter 3 are as: points 1)–3) and 5) presented for Chapter 1 for each of the sub-steps of our solver. Compared to Chapter 2 the novelties are: the development of a new kind of adaptivity that is local in patches with increased algebraic error, whereas the adaptivity in Chapter 2 chooses the number of post-smoothing steps globally per level; the localization in space relying on Dörfler's marking; the proof that the new adaptive sub-step contracts the error p-robustly, despite it only smoothes in marked patches (no convergence proof of the adaptive scheme is given in Chapter 2); the adaptive decision on which smoothing (additive Schwarz or weighted restricted additive Schwarz) variant to employ per level and inclusion of the weighted restricted additive Schwarz in the analysis, which was not done in Chapter 2.

To the best of our knowledge, this is the first work to prove that an adaptive local smoothing multigrid solver contracts the algebraic error p-robustly, while requiring only one post-smoothing step.

vii.4 Chapter 4

This chapter corresponds to an article in preparation. This work is a collaboration with Martin Vohralík and Ivan Yotov.

In this chapter we adapt the main ideas of our a posteriori-steered multigrid solver to the mixed finite element method. In two space dimensions, the theoret-

ical results of the Chapter 2 can be extended to this setting. Depending on the mesh hierarchy, we define the spaces in the multilevel construction and introduce a multigrid solver as well as a domain decomposition one.

The main results and novelties of Chapter 4 include a p-robustly contractive multigrid solver with associated p-robustly efficient a posteriori estimator as well as a p-robustly contractive domain decomposition solver with associated p-robustly efficient a posteriori estimator for a linear system discretized by mixed finite elements. While this work follows the same setting as in Ewing and Wang [1992], we prove that the estimates we develop are p-robust. Moreover, since we use level-wise step-sizes, we do not need to add any relaxation/damping parameters for the analysis and one post-smoothing step is sufficient.

vii.5 Implementation notes

The numerical experiments of this thesis were performed thanks to an in-house MATLAB finite element 2D code developed initially by Jan Papež. The coarse meshes of our multilevel setting are generated from a Delaunay triangulation algorithm using the Partial Differential Equation Toolbox of MATLAB and the refinements are uniform: each triangle is subdivided in four congruent new ones. The polynomial degrees supported in the code vary from 1 to 13, though this can be extended if needed. The main interest of the implementation is academic: since we have access to every single component of the code, it is easier to modify and verify all modules for our specific needs.

The solver modules were gradually added to the code as the results of thesis evolved. Currently, the code handles multigrid solvers with additive Schwarz smoothing, damped additive Schwarz smoothing, weighted restricted additive Schwarz smoothing associated to patch subdomains of different sizes, as well as the two adaptive approaches presented in the thesis. I have added some of these modules on my own, whereas others were added in collaboration with Jan Papež.

viii Perspectives

There are many directions and topics that would be interesting to explore.

One point to be pursued theoretically would be the robustness with respect to the jumps in the diffusion coefficient. While the numerical experiments we have conducted in two space dimensions show no degradation with increasing jumps, we believe that the outcome of tests in three space dimensions (apart from being important in their own right), would give helpful insights if one can hope to prove theoretical robustness.

Another possible subject to be tackled would be the replacement of the coarsest level direct solve in our approach by an inexact solver. We believe that this could be included in the theory upon ensuring, via our a posteriori estimators, that the error committed in the inexact coarse solve does not harm the overall precision.

We would also wish to incorporate our a-posteriori-steered multigrid solver as an inexact solver in a setting where the hierarchy is obtained through hp-refinement. Mesh elements in this case are expected to have different polynomial degrees and

this poses no problem for our analysis: we only need the hierarchy of finite element spaces to be nested.

Further work would also explore how the p-robust theory we presented for mixed finite elements in two space dimensions could be extended to the case of three space dimensions. Discretization by mortar elements is also another direction to be considered.

Finally, we are drawn to see how the developed solvers of this thesis can be used in real world hydrogeological applications with complex geometry such as fluid flow in fractured porous media. This subject is expected to be treated in the near future in the SERENA team.

Chapter 1

A multilevel algebraic error estimator and the corresponding iterative solver with p-robust behavior

This chapter consists of the article Miraçi et al. [2020], *SIAM Journal on Numerical Analysis*, 58, 5 (2020), 2856–2884, written with Jan Papež and Martin Vohralík.

Contents

1 Introduction . . . 26

2 Setting . . . 29

2.1 Model problem . . . 29

2.2 Finite element discretization . . . 29

2.3 Algebraic system, approximate solution, and algebraic residual 29

2.4 A hierarchy of meshes . . . 30

2.5 A hierarchy of spaces . . . 31

2.6 Two types of patches . . . 31

3 Multilevel lifting of the algebraic residual . . . 32

3.1 Exact algebraic residual lifting . . . 32

3.2 Coarse solve . . . 33

3.3 Multilevel algebraic residual lifting . . . 33

4 An a posteriori estimator on the algebraic error and a multilevel solver . . . 34

4.1 A posteriori estimate on the algebraic error . . . 34

4.2 Multilevel solver . . . 35

5 Main results . . . 36

6 Numerical experiments . . . 38

6.1 Performance of the damped additive Schwarz (dAS) construction of the solver . . . 39

6.2 Performance of the weighted restrictive additive Schwarz (wRAS) construction of the solver . . . 42

6.3 Comparison with other multilevel solvers . . . 43

7 Proofs of the main results **45**
7.1 Upper bound on $\|\nabla\rho^i_{J,\mathrm{alg}}\|$ 46
7.2 Lower bound on $(f,\rho^i_{J,\mathrm{alg}}) - (\nabla u^i_J, \nabla\rho^i_{J,\mathrm{alg}})$ 47
7.3 Polynomial-degree-robust multilevel stable decomposition . . . 48
7.4 Upper bound on $\|\nabla\tilde{\rho}^i_{J,\mathrm{alg}}\|$ 51
7.5 Proof of Theorem 5.1 53
7.6 Proof of Corollary 5.6 54
8 Conclusions and outlook **54**

Abstract

In this work, we consider conforming finite element discretizations of arbitrary polynomial degree $p \geq 1$ of the Poisson problem. We propose a multilevel a posteriori estimator of the algebraic error. We prove that this estimator is reliable and efficient (represents a two-sided bound of the error), with a constant independent of the degree p. We next design a multilevel iterative algebraic solver from our estimator and show that this solver contracts the algebraic error on each iteration by a factor bounded independently of p. Actually, we show that these two results are equivalent. The p-robustness results rely on the work of Schöberl et al. [IMA J. Numer. Anal., 28 (2008), pp. 1–24] for one given mesh. We combine this with the design of an algebraic residual lifting constructed over a hierarchy of nested unstructured, possibly highly graded, simplicial meshes. The lifting includes a global coarse-level solve with the lowest polynomial degree one together with local contributions from the subsequent mesh levels. These contributions, of the highest polynomial degree p on the finest mesh, are given as solutions of mutually independent local Dirichlet problems posed over overlapping patches of elements around vertices. The construction of this lifting can be seen as one geometric V-cycle multigrid step with zero pre- and one post-smoothing by (damped) additive Schwarz (block Jacobi). One particular feature of our approach is the optimal choice of the step-size generated from the algebraic residual lifting. Numerical tests are presented to illustrate the theoretical findings.

1 Introduction

The finite element method (FEM) is a widespread approach for discretizing problems given in the form of partial differential equations, and has been used in engineering for more than fifty years. For a thorough overview on the topic, we refer the reader to, e.g., Ciarlet [1978], Ern and Guermond [2004], and Brenner and Scott [2008]. Many iterative methods have been suggested to treat the linear systems arising from finite element discretizations; see e.g., Bramble et al. [1986] and Bramble et al. [1990], Hackbusch [2003], Bank et al. [1988], Brandt et al. [1985], Oswald [1994], or Zhang [1992], and the references therein. A systematic description of iterative solvers is given in Xu [1992]. For convergence results on unstructured and graded meshes, we refer the reader to, e.g., Wu and Chen [2006], Hiptmair et al. [2012], Chen et al. [2012], and Xu et al. [2009]. The convergence of these methods is

typically robust with respect to the size of the mesh (h-robustness). In fact, this is one of the key advantages of multigrid methods. For the conjugate gradient method on the other hand, h-robustness is not intrinsic; this problem can be bypassed with the development of appropriate preconditioners.

If we are to consider methods of arbitrary approximation polynomial degree, an additional question arises: how does the polynomial degree p affect the performance of the method? In this regard, results for p-version FEM include Foresti et al. [1989] for two-dimensional domains, Mandel [1990] for three-dimensional domains, and Babuška et al. [1991] for two-dimensional domains. For the latter, the condition number of the preconditioned system grows at most by $1 + \log^2(p)$, and a generalization of this work for hp-FEM is given by Ainsworth [1996], where the p-dependence is still present. An early version of a polynomial-degree robust (p-robust) solver was introduced by Quarteroni and Sacchi Landriani [1988] for a specific domain configuration (decomposable into rectangles without internal cross points). A notable development on p-robustness was later made by Pavarino [1994] for quadrilateral/hexahedral meshes, where the author introduced a p-robust additive Schwarz method. The generalization of this result for triangular/tetrahedral meshes was achieved by Schöberl et al. [2008], once more by introducing an additive Schwarz preconditioner. More recent works were carried out based on these approaches. In Antonietti et al. [2017] (see also the references therein), the p-robust approach for rectangular/hexahedral meshes was used for high-order discontinuous Galerkin (DG) methods; moreover the spectral bounds of the preconditioner are also robust with respect to the method's penalization coefficient. We also mention the introduction of multilevel preconditioners yielded by block Gauss–Seidel smoothers in Kanschat [2008] for rectangular/hexahedral meshes and DG discretizations. Further multilevel approaches for rectangular/hexahedral meshes using overlapping or nonoverlapping Schwarz smoothers can be found in, e.g., Janssen and Kanschat [2011] and Lucero Lorca and Kanschat [2021]. For a study on more general meshes, see, e.g., Antonietti and Pennesi [2019], where a multigrid approach behaves p-robustly under the condition that the number of smoothing steps (depending on p) is chosen sufficiently large. Another notable contribution is the design of algebraic multigrid methods (AMG) via aggregation techniques; see, e.g., Notay and Napov [2015], Bastian et al. [2012], and the references therein. The numerical results of the latter give a satisfactory indication of p-robustness.

An associated topic is the development of estimates of the algebraic error. In this regard, a posteriori tools have primarily been used to estimate the algebraic error for existing solvers. One particular goal is the development of reliable stopping criteria, allowing one to avoid unnecessary iterations. This is achieved with a combination of a posteriori error estimators for the discretization error. Some contributions on this matter (see also references therein) include Becker et al. [1995] where adaptive error control is achieved for a multigrid solver, Bornemann and Deuflhard [1996], where a one-way multigrid method is presented by integrating an adaptive stopping criterion based on a posteriori tools. Further developments were made in Meidner et al. [2009], where goal-oriented error estimates are used in the framework of the dual weighted residual (DWR) method. In Jiránek et al. [2010] and later in Papež et al. [2018], upper and lower bounds for both the algebraic and total errors are computed, which allow one to derive guaranteed upper and

lower bounds on the discretization error, and consecutively construct safe stopping criteria for iterative algebraic solvers. Arioli et al. [2013] propose practical stopping criteria which guarantee that the considered inexact adaptive FEM algorithm converges for inexact solvers of Krylov subspace type. To the best of the authors' knowledge, though, dedicated proofs of efficiency of a posteriori estimators of the algebraic error have so far not been presented.

In this work, we present an a posteriori algebraic error estimator and a multilevel iterative solver associated to it. The cornerstone of their definitions lies in the multilevel construction of a residual algebraic lifting, motivated partly by the approach of Papež et al. [2020]. The lifting can be seen as an approximation of the algebraic error by continuous piecewise polynomials of degree p, obtained by a V-cycle multigrid method with no pre-smoothing step and a single post-smoothing step. The coarse correction is given by a lowest polynomial degree (piecewise affine) function. Our smoothing is chosen in the family of damped additive Schwarz (block Jacobi) methods applied to overlapping subdomains composed of patches of elements (two options for defining the patches will be given in due time) and corresponds to local Dirichlet problems with the highest p-degree on the finest mesh. Note that additive Schwarz-type smoothing allows for a parallelizable implementation at each level of the mesh hierarchy. Once this lifting is built, the a posteriori estimator is easily derived as a natural guaranteed lower bound on the algebraic error, following Papež et al. [2020] and the references therein. As our first main result, we show that up to a p-robust constant, the estimator is also an upper bound on the error.

Our solver is then defined as a linear iterative method. Because we have at hand the residual lifting, which approximates the algebraic error, we use it as a descent direction (the asymmetric, since no pre-smoothing is used, approach in defining the lifting will not be a problem for the analysis). The step-size is then chosen by a line search in the direction of the lifting. Our choice presents a resemblance to the conjugate gradient method, in that we choose the step-size that ensures the best error contraction in the energy norm at the next iteration. Other precedents of the use of optimal step-sizes include, e.g., Canuto and Quarteroni [1985], and in the multigrid setting Heinrichs [1988]. As our second main result, we prove that this solver contracts the error at each iteration by a p-robust constant. Actually, we also show that the p-robust efficiency of the estimator is equivalent to the p-robust convergence of the solver. All these results are defined for a general hierarchy of nested, unstructured, possibly highly refined (graded) matching simplicial meshes, and no assumption beyond $u \in H_0^1(\Omega)$ is imposed on the weak solution.

The work is structured as follows. In Section 2, we introduce the setting in which we will be working as well as the notation employed throughout the paper. Then, we introduce our multilevel residual lifting construction in Section 3, following Papež et al. [2020]. In Section 4, we present the a posteriori estimator on the algebraic error and the corresponding multilevel solver based on the residual lifting. Our main results are presented in the form of two theorems in Section 5, together with a corollary establishing their equivalence. Another important corollary is the equivalence of the algebraic error with a computable estimator which is localized levelwise as well as patchwise. We provide numerical experiments in Section 6, focusing mainly on showcasing p-robustness, in agreement with our theoretical results, and on a comparison with several existing approaches. We also introduce a

weighted restricted additive Schwarz variant of our solver. The proofs of our main results are given in Section 7. In particular, for the stable decomposition estimate, the p-robust result on one level introduced by Schöberl et al. [2008] is crucial. We also rely on the multilevel stable splitting of Xu et al. [2009] for $p = 1$ to obtain acceptable estimates with respect to the number of levels. Finally, Section 8 brings forth our conclusions and outlook for future work.

2 Setting

We will consider in this work the Poisson problem defined over $\Omega \subset \mathbb{R}^d$, $d \in \{1, 2, 3\}$, an open bounded polytope with a Lipschitz-continuous boundary.

2.1 Model problem

Let $f \in L^2(\Omega)$ be the source term. We consider the following problem: find $u : \Omega \to \mathbb{R}$ such that

$$\begin{aligned} -\Delta u &= f \quad \text{in } \Omega, \\ u &= 0 \quad \text{on } \partial\Omega. \end{aligned} \tag{1.1}$$

In the weak formulation, we search for $u \in H_0^1(\Omega)$ such that

$$(\nabla u, \nabla v) = (f, v) \quad \forall v \in H_0^1(\Omega), \tag{1.2}$$

where $(\cdot, \cdot)$ is the $L^2(\Omega)$ or $[L^2(\Omega)]^d$ scalar product. The existence and uniqueness of the solution of (1.2) follows from the Riesz representation theorem.

2.2 Finite element discretization

Let $\mathcal{T}_J$ be a given simplicial mesh of Ω. Fixing an integer $p \geq 1$, we introduce the finite element space of continuous piecewise p-degree polynomials

$$V_J^p := \mathbb{P}_p(\mathcal{T}_J) \cap H_0^1(\Omega), \tag{1.3}$$

where $\mathbb{P}_p(\mathcal{T}_J) := \{v_J \in L^2(\Omega), v_J|_K \in \mathbb{P}_p(K)\ \forall K \in \mathcal{T}_J\}$. We set $N_J := \dim(V_J^p)$. The discrete problem consists in finding $u_J \in V_J^p$ such that

$$(\nabla u_J, \nabla v_J) = (f, v_J) \quad \forall v_J \in V_J^p. \tag{1.4}$$

2.3 Algebraic system, approximate solution, and algebraic residual

If one introduces ψ_J^l, $1 \leq l \leq N_J$, a basis of V_J^p, then problem (1.4) is equivalent to solving a system of linear algebraic equations. Denoting by $(\mathbb{A}_J)_{lm} := (\nabla\psi_J^m, \nabla\psi_J^l)$ the symmetric, positive definite (stiffness) matrix and by $(\mathrm{F}_J)_l := (f, \psi_J^l)$ the right-hand side (load) vector, one obtains $u_J = \sum_{m=1}^{N_J} (\mathrm{U}_J)_m \psi_J^m$, where $\mathrm{U}_J \in \mathbb{R}^{N_J}$ is the solution of

$$\mathbb{A}_J \mathrm{U}_J = \mathrm{F}_J.$$

For any approximation $\mathrm{U}_J^i \in \mathbb{R}^{N_J}$ of U_J given by an arbitrary algebraic solver at iteration step $i \geq 0$, the associated continuous piecewise polynomial of degree p is

$u_J^i = \sum_{m=1}^{N_J} (\mathrm{U}_J^i)_m \, \psi_J^m \in V_J^p$. The associated algebraic residual vector is given by

$$\mathrm{R}_J^i := \mathrm{F}_J - \mathbb{A}_J \mathrm{U}_J^i.$$

Note, however, that R_J^i depends on the choice of the basis functions ψ_J^l, $1 \le l \le N_J$. To avoid this dependence, we work instead with a residual functional on V_J^p given by

$$v_J \mapsto (f, v_J) - (\nabla u_J^i, \nabla v_J) \in \mathbb{R}, \quad v_J \in V_J^p. \tag{1.5}$$

We emphasize that the forthcoming results are independent of the choice of the basis.

2.4 A hierarchy of meshes

We consider a hierarchy of nested matching simplicial meshes $\{\mathcal{T}_j\}_{0 \le j \le J}$, $J \ge 1$, where $\mathcal{T}_J$ was introduced in Section 2.2, and where $\mathcal{T}_j$ is a refinement of $\mathcal{T}_{j-1}$, $1 \le j \le J$. For any element K on a given mesh, we denote $h_K := \mathrm{diam}(K)$ and by $\mathcal{V}_K$ the set of its vertices. We also denote $h_j := \max_{K \in \mathcal{T}_j} h_K$ for $0 \le j \le J$. Hereafter, we shall always assume that our meshes are shape regular.

Assumption 2.1 (Shape regularity)**.** *There exists $\kappa_{\mathcal{T}} > 0$ such that*

$$\max_{K \in \mathcal{T}_j} \frac{h_K}{\rho_K} \le \kappa_{\mathcal{T}} \text{ for all } 0 \le j \le J, \tag{1.6}$$

where ρ_K denotes the diameter of the largest ball inscribed in K.

Additionally to the above assumption, we will treat below two specific cases. In the first one, we suppose quasi-uniformity of the meshes in the hierarchy and that the strength of refinement is bounded. In the second case, we suppose that the meshes are generated by a series of bisections, e.g., the newest vertex bisection; cf. Sewell [1972].

2.4.1 A hierarchy of quasi-uniform meshes

We assume quasi-uniformity and that the hierarchy of meshes is such that the size of each parent element is comparable to the size of each of its children.

Assumption 2.2 (Maximum refinement strength and mesh quasi-uniformity)**.** *There exists $0 < C_{\mathrm{ref}} \le 1$, a fixed positive real number such that for any $j \in \{1, \dots, J\}$, $\forall K \in \mathcal{T}_{j-1}$, and for any $K^* \in \mathcal{T}_j$ such that $K^* \subset K$, there holds*

$$C_{\mathrm{ref}} h_K \le h_{K^*} \le h_K. \tag{1.7}$$

There further exists C_{qu}, a fixed positive real number such that for any $j \in \{0, \dots, J\}$ and $\forall K \in \mathcal{T}_j$, there holds

$$C_{\mathrm{qu}} h_j \le h_K \le h_j. \tag{1.8}$$

2.4.2 A hierarchy of graded bisection meshes

In the case of graded mesh hierarchies obtained by bisection, one refinement of an edge of $\mathcal{T}_{j-1}$, for $j \in \{1, \dots, J\}$, gives a new finer mesh $\mathcal{T}_j$. We denote by $\mathcal{B}_j \subset \mathcal{V}_j$ the set consisting of the new vertex obtained after the bisection together with its two neighbors on the refinement edge, cf. Figure 1.1 for an illustration when $d = 2$. We denote by $h_{\mathcal{B}_j}$ the maximal diameter of elements having a vertex in $\mathcal{B}_j$. This setting is described by the following.

Assumption 2.3 (Local quasi-uniformity of bisection-generated meshes)**.** *$\mathcal{T}_0$ is a conforming quasi-uniform mesh with parameter C^0_{qu}. The graded conforming mesh $\mathcal{T}_J$ is generated from $\mathcal{T}_0$ by a series of bisections. There exists a fixed positive real number $C^{\mathrm{loc}}_{\mathrm{qu}}$ such that for any $j \in \{1, \dots, J\}$, there holds*

$$C^{\mathrm{loc}}_{\mathrm{qu}} h_{\mathcal{B}_j} \leq h_K \leq h_{\mathcal{B}_j} \quad \forall K \in \mathcal{T}_j \text{ such that a vertex of } K \text{ belongs to } \mathcal{B}_j. \tag{1.9}$$

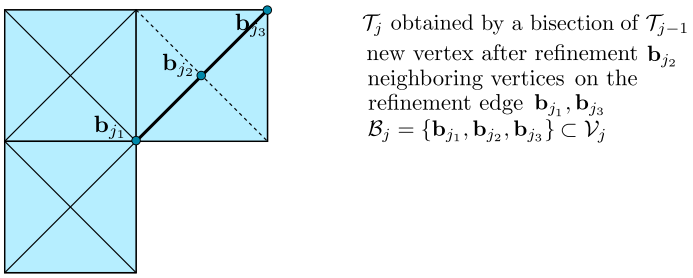


Figure 1.1: Illustration of the set $\mathcal{B}_j$. The mesh $\mathcal{T}_{j-1}$ and its refinement $\mathcal{T}_j$ are given by full and dotted lines, respectively.

2.5 A hierarchy of spaces

In the following, we will need to introduce a hierarchy of finite element spaces associated to the mesh hierarchy. For this purpose, let $p' \in \{1, \dots, p\}$ be a polynomial degree between 1 and p that we employ for the intermediate levels. In particular, let the following hold:

$$\text{for } 1 \leq j \leq J-1, \qquad V_j^{p'} := \mathbb{P}_{p'}(\mathcal{T}_j) \cap H^1_0(\Omega) \quad (p'\text{th order spaces}), \tag{1.10a}$$

$$\text{for } j = 0, \qquad V_0^1 = \mathbb{P}_1(\mathcal{T}_0) \cap H^1_0(\Omega) \quad (\text{lowest-order space}), \tag{1.10b}$$

where $\mathbb{P}_{p'}(\mathcal{T}_j) := \{v_j \in L^2(\Omega), v_j|_K \in \mathbb{P}_{p'}(K) \;\; \forall K \in \mathcal{T}_j\}$ for $1 \leq j \leq J-1$. Note that $V_0^1 \subset V_1^{p'} \subset \cdots \subset V_{J-1}^{p'} \subset V_J^p$. Let $\mathcal{V}_j$ be the set of vertices of the mesh $\mathcal{T}_j$. We denote by $\psi_{j,\mathbf{a}}$ the standard hat function associated to the vertex $\mathbf{a} \in \mathcal{V}_j$, $0 \leq j \leq J$. This is the piecewise affine function with respect to the mesh $\mathcal{T}_j$ that takes value 1 in the vertex $\mathbf{a}$ and vanishes in all other jth level vertices of $\mathcal{V}_j$.

2.6 Two types of patches

For the following, we define two types of patches of elements. In order to facilitate the work with both, we introduce a switching parameter $s \in \{0, 1\}$. First, given a

vertex $\mathbf{a} \in \mathcal{V}_{j-s}$, $j \in \{1, \dots, J\}$, we denote by $\mathcal{T}_{j,s}^{\mathbf{a}}$ the patch formed by all elements of the mesh $\mathcal{T}_{j-s}$ sharing the vertex $\mathbf{a}$, i.e.,

$$\mathcal{T}_{j,s}^{\mathbf{a}} := \{K \in \mathcal{T}_{j-s}, \mathbf{a} \in \mathcal{V}_K\}. \tag{1.11}$$

We also denote by $\omega_{j,s}^{\mathbf{a}}$ the open patch subdomain corresponding to $\mathcal{T}_{j,s}^{\mathbf{a}}$. An illustration is given in Figure 1.2 (left) for "small" patches $s = 0$ and (right) for "large" patches $s = 1$. Then the associated local space $V_{j,s}^{\mathbf{a}}$ is given by

$$V_{j,s}^{\mathbf{a}} := \mathbb{P}_{p'}(\mathcal{T}_j) \cap H_0^1(\omega_{j,s}^{\mathbf{a}}),\ j \in \{1, \dots, J-1\} \text{ and } V_{J,s}^{\mathbf{a}} := \mathbb{P}_p(\mathcal{T}_J) \cap H_0^1(\omega_{J,s}^{\mathbf{a}}). \tag{1.12}$$

Note that $V_{j,s}^{\mathbf{a}}$ are continuous piecewise polynomial spaces with respect to the mesh $\mathcal{T}_j$ for both $s = 0$ and $s = 1$, the support being bigger in the latter case. An illustration is also given in Figure 1.2.

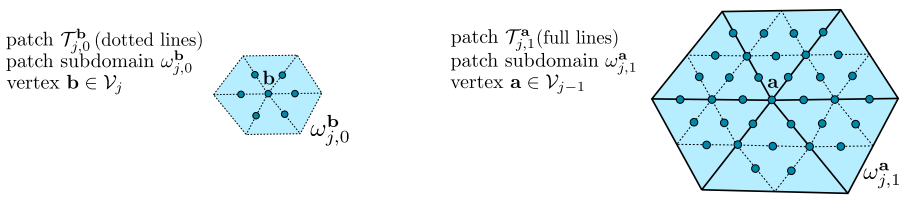


Figure 1.2: Illustration of degrees of freedom ($p' = p = 2$) for the space $V_{j,0}^{\mathbf{b}}$ associated to the "small" patch $\mathcal{T}_{j,0}^{\mathbf{b}}$ (left) and for the space $V_{j,1}^{\mathbf{a}}$ associated to the "large" patch $\mathcal{T}_{j,1}^{\mathbf{a}}$ (right). The mesh $\mathcal{T}_{j-1}$ and its refinement $\mathcal{T}_j$ are defined in bold and dotted lines, respectively.

3 Multilevel lifting of the algebraic residual

In the spirit of Papež et al. [2020], we design a multilevel lifting of the algebraic residual given by (1.5). This lifting will lead to the construction of an a posteriori error estimator; it will also serve as a descent direction for the solver we introduce in the next section.

3.1 Exact algebraic residual lifting

For illustration and theoretical analysis later, we introduce the following definition.

Definition 3.1 (Exact residual lifting)**.** *Let $u_J^i \in V_J^p$ be arbitrary. We introduce $\tilde{\rho}_{J,\mathrm{alg}}^i \in V_J^p$ as the solution of the residual problem*

$$(\nabla \tilde{\rho}_{J,\mathrm{alg}}^i, \nabla v_J) = (f, v_J) - (\nabla u_J^i, \nabla v_J) \quad \forall v_J \in V_J^p, \tag{1.13}$$

so that

$$\tilde{\rho}_{J,\mathrm{alg}}^i = u_J - u_J^i. \tag{1.14}$$

3.2 Coarse solve

The first step of our construction is to solve a global lowest-order problem on the coarsest mesh. Let $u_J^i \in V_J^p$ be given. Recalling that $V_0^1 = \mathbb{P}_1(\mathcal{T}_0) \cap H_0^1(\Omega)$, we define $\rho_0^i \in V_0^1$ by

$$(\nabla \rho_0^i, \nabla v_0) = (f, v_0) - (\nabla u_J^i, \nabla v_0) \quad \forall v_0 \in V_0^1. \tag{1.15}$$

Note that due to (1.15) and (1.13), we have

$$(\nabla \rho_0^i, \nabla v_0) = (\nabla \tilde{\rho}_{J,\mathrm{alg}}^i, \nabla v_0) \quad \forall v_0 \in V_0^1, \tag{1.16}$$

so that ρ_0^i is the orthogonal projection of $\tilde{\rho}_{J,\mathrm{alg}}^i$ onto the coarsest space V_0^1.

3.3 Multilevel algebraic residual lifting

Let us now introduce our hierarchical construction of the algebraic residual lifting $\rho_{J,\mathrm{alg}}^i \in V_J^p$ that is hopefully close to $\tilde{\rho}_{J,\mathrm{alg}}^i$. The construction relies on the use of a coarse solution of (1.15) and on local contributions arising from all the finer mesh levels. These local contributions are defined on patch subdomains $\omega_{j,s}^{\mathbf{a}}$. We denote by $(\cdot,\cdot)_{\omega_{j,s}^{\mathbf{a}}}$ the $L^2(\omega_{j,s}^{\mathbf{a}})$ or $[L^2(\omega_{j,s}^{\mathbf{a}})]^d$ scalar product. Since we consider two definitions of patches with switching parameter $s \in \{0,1\}$ (see Section 2.6), two constructions of $\rho_{J,\mathrm{alg}}^i$ are implied.

Definition 3.2 (Construction of the algebraic residual lifting)**.** *Let $w_1, w_2 \in \mathbb{R} \cup \{\infty\}$ be damping weights satisfying the conditions*

$$1 \le w_1 < 6J(d+1) \quad \textit{and} \quad w_2 \ge \max\left(1, \frac{5J^2(d+1)^2}{w_1(6J(d+1) - w_1)}\right). \tag{1.17}$$

Let $u_J^i \in V_J^p$ be arbitrary. We introduce $\rho_{J,\mathrm{alg}}^i \in V_J^p$ by

$$\rho_{J,\mathrm{alg}}^i := \rho_0^i + \sum_{j=1}^{J} \rho_j^i, \tag{1.18}$$

where $\rho_0^i \in V_0^1$ solves (1.15) and $\rho_j^i \in V_j^{p'}$, for $j \in \{1, \dots, J-1\}$, and $\rho_J^i \in V_J^p$ are given by

$$\rho_j^i := \frac{1}{w_1} \sum_{\mathbf{a} \in \mathcal{V}_{j-s}} \rho_{j,\mathbf{a}}^i, \ 1 \le j \le J, \tag{1.19}$$

with the local contributions $\rho_{j,\mathbf{a}}^i \in V_{j,s}^{\mathbf{a}}$ given by patch problems, $\forall v_{j,\mathbf{a}} \in V_{j,s}^{\mathbf{a}}$

$$(\nabla \rho_{j,\mathbf{a}}^i, \nabla v_{j,\mathbf{a}})_{\omega_{j,s}^{\mathbf{a}}} = (f, v_{j,\mathbf{a}})_{\omega_{j,s}^{\mathbf{a}}} - (\nabla u_J^i, \nabla v_{j,\mathbf{a}})_{\omega_{j,s}^{\mathbf{a}}} - \frac{1}{w_2} \sum_{k=0}^{j-1} (\nabla \rho_k^i, \nabla v_{j,\mathbf{a}})_{\omega_{j,s}^{\mathbf{a}}}. \tag{1.20}$$

Remark 3.3 (Construction of $\rho_{J,\mathrm{alg}}^i$)**.** *The construction (1.18)–(1.20) of $\rho_{J,\mathrm{alg}}^i$ can be seen as an approximation of $\tilde{\rho}_{J,\mathrm{alg}}^i$ from (1.13) by one iteration of a V-cycle multigrid, with no pre-smoothing and a single post-smoothing step, corresponding to a "damped" additive Schwarz iteration, with the damping factor determined by the weights w_1 and w_2. The subdomains for the Schwarz method correspond to the*

patch domains where the local problems in (1.20) *are defined. Two patch options as in Figure 1.2 are considered. In particular, for $p = 1$ and "small" patches, $s = 0$ (Figure 1.2, left), this corresponds to one step of the Jacobi (diagonal) smoother, whereas when $p' = p > 1$, the smoother is block Jacobi. A weighted variant of Definition 3.2 is tested in Section 6.*

Remark 3.4 (Value of the damping parameter)**.** *Condition* (1.17) *is based on the proofs in Section 7 below, where the use of appropriate damping seems crucial. This is what is also indicated numerically to be needed in our approach. Possible combinations of the damping weights satisfying* (1.17) *include, for example,*

$$w_1 = J(d+1) \text{ and } w_2 = 1, \tag{1.21a}$$

$$w_1 = d+1 \text{ and } w_2 = J, \tag{1.21b}$$

$$w_1 = w_2 = \sqrt{J(d+1)}, \tag{1.21c}$$

$$w_1 = 1 \text{ and } w_2 = \infty, \tag{1.21d}$$

$$w_1 = 4\sqrt{J} \text{ and } w_2 = \infty. \tag{1.21e}$$

Examples (1.21a)–(1.21c) *above result in a procedure that is additive patchwise and multiplicative levelwise. Examples* (1.21d)–(1.21e)*, in turn, result in a completely additive patchwise and levelwise procedure, which is fully parallelizable. We also note that when the intermediate polynomial degree is $p' = 1$ and for any choice with $w_2 = 1$, the smoothing resulting from Definition 3.2 is local with respect to mesh $\mathcal{T}_0$ for graded meshes; it is actually only performed there where the meshes $\mathcal{T}_j$, $j \geq 1$, are different from $\mathcal{T}_0$.*

4 An a posteriori estimator on the algebraic error and a multilevel solver

We now present how the residual lifting $\rho^i_{J,\mathrm{alg}}$ of Definition 3.2 can be used to define an a posteriori estimator as well as a multilevel solver.

4.1 A posteriori estimate on the algebraic error

We begin by introducing η^i_{alg}, an a posteriori estimator defined using the residual lifting $\rho^i_{J,\mathrm{alg}}$.

Definition 4.1 (Lower bound algebraic error estimator)**.** *Let $u^i_J \in V^p_J$ be arbitrary and let $\rho^i_{J,\mathrm{alg}}$ be the algebraic residual lifting given by Definition 3.2. If $\rho^i_{J,\mathrm{alg}} = 0$, we define the lower bound algebraic error estimator $\eta^i_{\mathrm{alg}} := 0$. Otherwise, set*

$$\eta^i_{\mathrm{alg}} := \frac{(f, \rho^i_{J,\mathrm{alg}}) - (\nabla u^i_J, \nabla \rho^i_{J,\mathrm{alg}})}{\|\nabla \rho^i_{J,\mathrm{alg}}\|}. \tag{1.22}$$

The estimator η^i_{alg} is immediately a guaranteed lower bound on the algebraic error; cf., e.g., [Papež et al., 2020, Theorem 5.3].

Lemma 4.2 (Guaranteed lower bound on the algebraic error)**.** *There holds*

$$\|\nabla(u_J - u^i_J)\| \geq \eta^i_{\mathrm{alg}}. \tag{1.23}$$

Proof. Note that if $\rho^i_{J,\mathrm{alg}} = 0$, then $\|\nabla(u_J - u^i_J)\| \geq 0 = \eta^i_{\mathrm{alg}}$. Otherwise

$$\|\nabla(u_J - u^i_J)\| = \max_{\substack{v_J \in V^p_J, \\ \|\nabla v_J\| \neq 0}} \frac{(\nabla(u_J - u^i_J), \nabla v_J)}{\|\nabla v_J\|} \geq \frac{(\nabla(u_J - u^i_J), \nabla \rho^i_{J,\mathrm{alg}})}{\|\nabla \rho^i_{J,\mathrm{alg}}\|} \overset{\substack{(1.4)\\(1.22)}}{=} \eta^i_{\mathrm{alg}}.$$

□

4.2 Multilevel solver

We will now reuse the construction of $\rho^i_{J,\mathrm{alg}}$ given in Definition 3.2 to obtain an approximation of u_J on a next step in view of constructing a multilevel solver. Note that for any $u^i_J \in V^p_J$, the lifting $\rho^i_{J,\mathrm{alg}}$ is built to approximate the algebraic error $\tilde{\rho}^i_{J,\mathrm{alg}}$ given in (1.13), where we recall that $u_J = u^i_J + \tilde{\rho}^i_{J,\mathrm{alg}}$. Thus, it seems reasonable to consider a linear iterative solver of the form

$$u^{i+1}_J := u^i_J + \lambda \rho^i_{J,\mathrm{alg}}, \tag{1.24}$$

where $\lambda \in \mathbb{R}$ is a real parameter. The optimal choice of λ is given below.

Lemma 4.3 (Optimal step-size)**.** *Consider a solver of the form* (1.24) *and suppose* $\rho^i_{J,\mathrm{alg}} \neq 0$. *Then the choice* $\lambda := [(f, \rho^i_{J,\mathrm{alg}}) - (\nabla u^i_J, \nabla \rho^i_{J,\mathrm{alg}})]/\|\nabla \rho^i_{J,\mathrm{alg}}\|^2$ *leads to minimal algebraic error with respect to the energy norm.*

Proof. We write the algebraic error of the next iteration as a function of λ,

$$\|\nabla(u_J - u^{i+1}_J)\|^2 = \|\nabla(u_J - u^i_J)\|^2 - 2\lambda(\nabla(u_J - u^i_J), \nabla \rho^i_{J,\mathrm{alg}}) + \lambda^2 \|\nabla \rho^i_{J,\mathrm{alg}}\|^2, \tag{1.25}$$

and realize that this function has a minimum at

$$\lambda_{\min} = \frac{(\nabla(u_J - u^i_J), \nabla \rho^i_{J,\mathrm{alg}})}{\|\nabla \rho^i_{J,\mathrm{alg}}\|^2} \overset{(1.4)}{=} \frac{(f, \rho^i_{J,\mathrm{alg}}) - (\nabla u^i_J, \nabla \rho^i_{J,\mathrm{alg}})}{\|\nabla \rho^i_{J,\mathrm{alg}}\|^2} \overset{(1.22)}{=} \frac{\eta^i_{\mathrm{alg}}}{\|\nabla \rho^i_{J,\mathrm{alg}}\|}. \tag{1.26}$$

□

We are now ready to define our multilevel solver.

Definition 4.4 (Multilevel solver)**.**

1. *Initialize* $u^0_J \in V^1_0$ *as the solution of* $(\nabla u^0_J, \nabla v_0) = (f, v_0)\ \forall v_0 \in V^1_0$.
2. *Let* $i \geq 0$ *be the iteration counter and let* $\rho^i_{J,\mathrm{alg}}$ *be constructed from* u^i_J *following Definition 3.2. When* $\rho^i_{J,\mathrm{alg}} = 0$, *set* $u^{i+1}_J := u^i_J$ *and stop; then actually* $u^{i+1}_J = u^i_J = u_J$. *Otherwise, let*

$$u^{i+1}_J := u^i_J + \frac{(f, \rho^i_{J,\mathrm{alg}}) - (\nabla u^i_J, \nabla \rho^i_{J,\mathrm{alg}})}{\|\nabla \rho^i_{J,\mathrm{alg}}\|^2} \rho^i_{J,\mathrm{alg}}. \tag{1.27}$$

Remark 4.5 (Multilevel solver)**.** *Note that the solver of Definition 4.4 is not initialized randomly but via a coarse solve. The descent direction is the residual lifting* $\rho^i_{J,\mathrm{alg}}$, *constructed via a single V-cycle iteration with no pre-smoothing and one post-smoothing step, and the step-size is optimized via the line search* (1.26). *This minimalist and asymmetrical procedure will not be an issue in the forthcoming analysis.*

Remark 4.6 (Cost of one iteration)**.** *On each iteration of the developed solver, there are costs which correspond to those of standard multigrid methods: coarse solve (here with the lowest polynomial degree) and interlevel transfer operations. The crucial difference is in the smoothing cost. While we prove below that our solver is p-robust and only mildly depends on h (since $J \sim |\log h|$), meaning the number of iterations will not degrade when p increases, the sizes of the local matrices used to solve the local problems* (1.20) *increase (in 2D approximately as p^2). This induces a significant computational, but perfectly parallelizable, cost for higher p. Other cheaper options may be developed to bypass the local problems, for example, in the spirit of* Papež and Vohralík [2019]*. Recall, however, that there is only one smoothing per iteration in our approach.*

5 Main results

In this section, we present the main results concerning our a posteriori estimator η^i_{alg} of Definition 4.1 and our multilevel solver of Definition 4.4. We shall also see how these two main results are related.

For the estimator the following holds.

Theorem 5.1 (p-robust reliable and efficient bound on the algebraic error)**.** *Let Assumption 2.1 hold, together with either Assumption 2.2 or 2.3. Let $u_J \in V^p_J$ be the (unknown) solution of* (1.4) *and let $u^i_J \in V^p_J$ be arbitrary, $i \geq 0$. Let η^i_{alg} be given by Definition 4.1. Then, in addition to $\|\nabla(u_J - u^i_J)\| \geq \eta^i_{\mathrm{alg}}$ of* (1.23)*, there holds*

$$\eta^i_{\mathrm{alg}} \geq \beta \|\nabla(u_J - u^i_J)\|, \tag{1.28}$$

where $0 < \beta < 1$ only depends on the space dimension d, the mesh shape regularity parameter $\kappa_{\mathcal{T}}$, and the number of mesh levels J, as well as on the mesh refinement parameter C_{ref} and the quasi-uniformity parameter C_{qu} if Assumption 2.2 holds, or on the coarse mesh and the local quasi-uniformity parameters C^0_{qu} and $C^{\mathrm{loc}}_{\mathrm{qu}}$ if Assumption 2.3 holds. For all weights satisfying (1.17)*, there holds $\beta \geq J^{-5/2}\beta^*$ with β^* independent of the number of levels J. Better bounds hold for the weights of Remark 3.4; see Example 7.8 below for details.*

The theorem allows us to write η^i_{alg} as a two-sided bound of the algebraic error (up to the generic constant β for the upper bound), meaning that the estimator is robustly efficient with respect to the polynomial degree p. We can also reinterpret this result as follows.

Remark 5.2 (Angle between the error and the descent direction)**.** *Note that if we rewrite* (1.28) *by plugging in the Definition 4.1 of η^i_{alg}, when $u_J - u^i_J \neq 0$ and $\rho^i_{J,\mathrm{alg}} \neq 0$, we have*

$$\frac{\eta^i_{\mathrm{alg}}}{\|\nabla(u_J - u^i_J)\|} \overset{(1.22)}{=} \frac{(f, \rho^i_{J,\mathrm{alg}}) - (\nabla u^i_J, \nabla \rho^i_{J,\mathrm{alg}})}{\|\nabla(u_J - u^i_J)\| \|\nabla \rho^i_{J,\mathrm{alg}}\|} \overset{(1.4)}{=} \frac{(\nabla(u_J - u^i_J), \nabla \rho^i_{J,\mathrm{alg}})}{\|\nabla(u_J - u^i_J)\| \|\nabla \rho^i_{J,\mathrm{alg}}\|} \overset{(1.28)}{\geq} \beta > 0.$$

This can be compared to classical results in line search methods (see, e.g.*, Nocedal and Wright* [Nocedal and Wright, 2006, Chapter 3.2]*) of boundedness away from zero of the cosine of the angle between the vector to be minimized (here $u_J - u^i_J$) and the descent direction (here the lifting $\rho^i_{J,\mathrm{alg}}$).*

For the solver, in turn, we have the following.

Theorem 5.3 (p-robust error contraction of the multilevel solver)**.** *Let Assumption 2.1 hold, together with either Assumption 2.2 or 2.3. Let $u_J \in V_J^p$ be the (unknown) solution of (1.4) and let $u_J^i \in V_J^p$ be arbitrary, $i \geq 0$. Take u_J^{i+1} to be constructed from u_J^i using one step of the multilevel solver of Definition 4.4 by (1.27). Then there holds*

$$\|\nabla(u_J - u_J^{i+1})\| \leq \alpha \|\nabla(u_J - u_J^i)\|, \tag{1.29}$$

where $0 < \alpha < 1$ is given by $\alpha = \sqrt{1-\beta^2}$ with β the constant from (1.28).

In the above theorem, α is a bound on the algebraic error contraction factor at each step i. Looking at the dependencies of α, we see that the solver of Definition 4.4 contracts the algebraic error at each iteration step in a robust way with respect to the polynomial degree p.

Theorems 5.1 and 5.3 are connected as follows.

Corollary 5.4 (Equivalence of the p-robust estimator efficiency and p-robust solver contraction)**.** *Let the assumptions of Theorems 5.1 and 5.3 be satisfied. Then (1.28) holds if and only if (1.29) holds, and $\beta = \sqrt{1-\alpha^2}$.*

Proof. Let $u_J \in V_J^p$ be the solution of (1.4), let $u_J^i \in V_J^p$ be arbitrary, and let $u_J^{i+1} \in V_J^p$ be constructed from u_J^i by our multilevel solver of Definition 4.4. First, we write the relation between the algebraic errors associated to u_J^{i+1} and u_J^i.

Case $\rho_{J,\mathrm{alg}}^i \neq 0$. Using (1.25) and (1.26), we see

$$\|\nabla(u_J - u_J^{i+1})\|^2 = \|\nabla(u_J - u_J^i)\|^2 - (\eta_{\mathrm{alg}}^i)^2. \tag{1.30}$$

Case $\rho_{J,\mathrm{alg}}^i = 0$. By Definitions 4.4 and 4.1, we have $u_J^{i+1} = u_J^i$ and $\eta_{\mathrm{alg}}^i = 0$. In particular, this means that $\|\nabla(u_J - u_J^{i+1})\| = \|\nabla(u_J - u_J^i)\|$, so that (1.30) still holds.

The above observations allow us to write, in any case, starting from (1.29) with $0 < \alpha < 1$,

$$\begin{aligned} & \|\nabla(u_J - u_J^{i+1})\|^2 \leq \alpha^2 \|\nabla(u_J - u_J^i)\|^2 \\ \overset{(1.30)}{\Leftrightarrow} \; & \|\nabla(u_J - u_J^i)\|^2 - (\eta_{\mathrm{alg}}^i)^2 \leq \alpha^2 \|\nabla(u_J - u_J^i)\|^2 \\ \Leftrightarrow \; & \|\nabla(u_J - u_J^i)\|^2 (1-\alpha^2) \leq (\eta_{\mathrm{alg}}^i)^2, \end{aligned}$$

which is (1.28) with $\beta^2 = 1 - \alpha^2$. □

In view of Corollary 5.4, we will prove in Section 7 below only Theorem 5.1.

Importantly, the following also holds.

Corollary 5.5 (Equivalence of vanishing algebraic lifting with the solver reaching the solution)**.** *Let the assumptions of Theorems 5.1 and 5.3 be satisfied. Then $\rho_{J,\mathrm{alg}}^i = 0$ if and only if $u_J^{i+1} = u_J^i = u_J$.*

Finally, by the proofs in Section 7, the algebraic error is also equivalent to a localized a posteriori error estimate.

Corollary 5.6 (p-robust localized reliable and efficient a posteriori estimate on the algebraic error)**.** *Let the assumptions of Theorem 5.1 be satisfied. Let $\rho^i_{J,\mathrm{alg}}$ be the algebraic residual lifting constructed in Definition 3.2. Then*

$$\|\nabla(u_J - u_J^i)\|^2 \leq C_1^2 \left(\|\nabla \rho_0^i\|^2 + \sum_{j=1}^{J} \sum_{\mathbf{a}\in\mathcal{V}_{j-s}} \|\nabla \rho^i_{j,\mathbf{a}}\|^2_{\omega^{\mathbf{a}}_{j,s}} \right) \leq C_2^2 \|\nabla(u_J - u_J^i)\|^2, \tag{1.31}$$

where $C_2 = \frac{1}{\beta}$ and C_1 is identified in Section 7.6.

Equivalence (1.31) gives us an idea where the algebraic error is situated level-wise and patchwise. This information can be exploited to tackle problematic areas adaptively, which is the subject of forthcoming works.

6 Numerical experiments

In this section we report some numerical illustrations of the theoretical results of Section 5. In particular, we focus on the p-robustness. In the following tests, we consider the model problem (1.1) with three different choices of the domain $\Omega \subset \mathbb{R}^2$ and of the exact solution u:

$$\text{Sine:} \quad u(x,y) := \sin(2\pi x)\sin(2\pi y), \ \Omega := (-1,1)^2. \tag{1.32}$$

$$\text{Peak:} \quad u(x,y) := x(x-1)y(y-1)\mathrm{e}^{-100((x-0.5)^2-(y-0.117)^2)}, \ \Omega := (0,1)^2. \tag{1.33}$$

$$\text{L-shape:} \quad u(r,\theta) := r^{2/3}\sin(2\theta/3), \ \Omega := (-1,1)^2 \setminus ([0,1]\times[-1,0]). \tag{1.34}$$

For the L-shape problem (1.34), we impose an inhomogeneous Dirichlet boundary condition corresponding to the exact solution, which is expressed here in polar coordinates. For each of the test cases, we start with an initial Delaunay triangulation of Ω. Then we consider J uniform refinements where all triangles are decomposed into four congruent subtriangles. Implementation-wise, we opt for Lagrange basis functions with nonuniformly distributed nodes because of their better behavior with respect to high-order methods; see Warburton [2006]. Recall that this choice has no influence on the theoretical results of Section 5 as well as presented numerical results (in exact arithmetic). Though it is not the focus of this work, we also remark that our solver can be implemented in a matrix-free way and can also be parallelized.

The contraction factor of the solver of Definition 4.4 on each step i is given by $\|\nabla(u_J - u_J^{i+1})\|/\|\nabla(u_J - u_J^i)\|$, and, as stated in Corollary 5.4, it reveals the efficiency of the a posteriori estimator η^i_{alg} of Definition 4.1. Keeping this in mind, we only focus on the solver and the contraction factor. We will follow a common choice for the stopping criterion, with the notation of Section 2.3:

$$\frac{\|\mathrm{F}_J - \mathbb{A}_J \mathrm{U}_J^{i_\mathrm{s}}\|}{\|\mathrm{F}_J\|} \leq 10^{-5} \frac{\|\mathrm{F}_J - \mathbb{A}_J \mathrm{U}_J^0\|}{\|\mathrm{F}_J\|}. \tag{1.35}$$

We also introduce the average error contraction factor

$$\bar{\alpha} := \frac{1}{i_\mathrm{s}} \sum_{i=0}^{i_\mathrm{s}-1} \frac{\|\nabla(u_J - u_J^{i+1})\|}{\|\nabla(u_J - u_J^i)\|}. \tag{1.36}$$

We expect a p-robust solver to converge in a similar number of iterations and have similar error contraction factors at all iterations for different polynomial degrees p. The tests below cover different numbers of mesh levels $J = 3, 4, 5$, polynomial degrees $p = 1, 3, 6, 9$, and the "small" as well as the "large" patches as in Figure 1.2.

6.1 Performance of the damped additive Schwarz (dAS) construction of the solver

A crucial component in the definition of our a posteriori estimator and multilevel solver is the construction of the residual lifting $\rho^i_{J,\mathrm{alg}}$ of Definition 3.2, where we have used damped additive Schwarz (dAS) to cope with overlapping:

$$\text{dAS:} \quad \rho^i_{J,\mathrm{alg}} := \sum_{j=0}^{J} \rho^i_j \quad \text{and} \quad \rho^i_j := \frac{1}{w_1} \sum_{\mathbf{a}\in\mathcal{V}_{j-s}} \rho^i_{j,\mathbf{a}}, \; 1 \le j \le J, \tag{1.37}$$

$$(\nabla \rho^i_{j,\mathbf{a}}, \nabla v_{j,\mathbf{a}})_{\omega^{\mathbf{a}}_{j,s}} = (f, v_{j,\mathbf{a}})_{\omega^{\mathbf{a}}_{j,s}} - (\nabla u^i_J, \nabla v_{j,\mathbf{a}})_{\omega^{\mathbf{a}}_{j,s}} - \frac{1}{w_2} \sum_{k=0}^{j-1} (\nabla \rho^i_k, \nabla v_{j,\mathbf{a}})_{\omega^{\mathbf{a}}_{j,s}}.$$

For the three test cases we consider three different choices of the damping weights which satisfy condition (1.17) (see Remark 3.4):

$$\begin{aligned}
&\text{for problem (1.32):} && w_1 = J(d+1) \text{ and } w_2 = 1;\\
&\text{for problem (1.33):} && w_1 = 4\sqrt{J} \text{ and } w_2 = \infty;\\
&\text{for problem (1.34):} && w_1 = d+1 \text{ and } w_2 = J.
\end{aligned}$$

Recall that the choice $w_2 = \infty$ means that the construction of the lifting $\rho^i_{J,\mathrm{alg}}$ can be implemented completely in parallel, levelwise as well as patchwise.

The results are presented in Figures 1.3–1.5 and in Table 1.1. They confirm the expected complete independence of the polynomial degree p for our multilevel solver which uses the construction dAS (1.37) of the lifting. Actually, we observe better contraction factors for higher polynomial degrees.

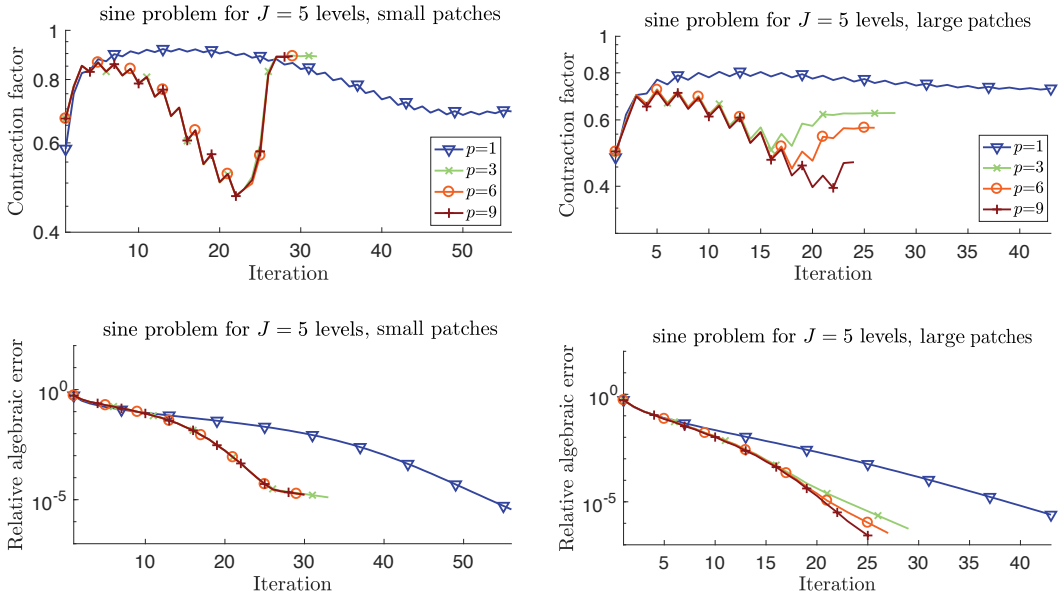


Figure 1.3: Sine problem (1.32), $w_1{=}J(d{+}1)$, $w_2{=}1$: results of the solver (1.27) for $p'{=}p$ in (1.10a), "small" (left) and "large" (right) patches, and stopping criterion (1.35). Top: error contraction factors $\|\nabla(u_J - u_J^{i+1})\|/\|\nabla(u_J - u_J^i)\|$. Bottom: relative algebraic error $\|\nabla(u_J - u_J^i)\|/\|\nabla u_J\|$.

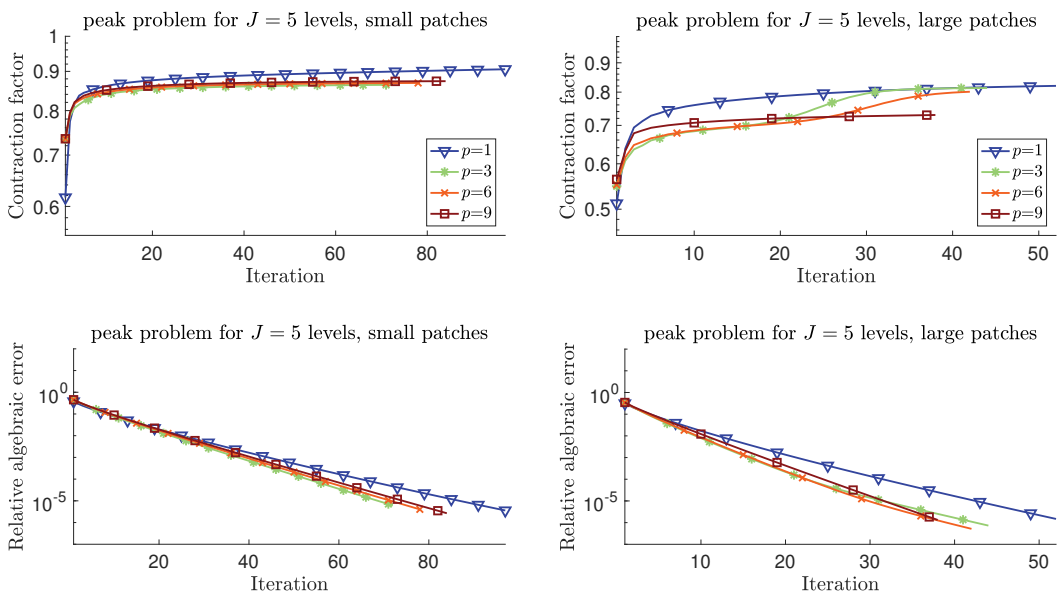


Figure 1.4: Peak problem (1.33), $w_1 = 4\sqrt{J}$, $w_2 = \infty$: results of the solver (1.27) for $p' = p$ in (1.10a), "small" (left) and "large" (right) patches, and stopping criterion (1.35). Top: error contraction factors $\|\nabla(u_J - u_J^{i+1})\|/\|\nabla(u_J - u_J^i)\|$. Bottom: relative algebraic error $\|\nabla(u_J - u_J^i)\|/\|\nabla u_J\|$.

			Sine problem (1.32) $w_1 = J(d+1)$, $w_2 = 1$				Peak problem (1.33) $w_1 = 4\sqrt{J}$, $w_2 = \infty$				L-shape problem (1.34) $w_1 = d+1$, $w_2 = J$			
			"small"		"large"		"small"		"large"		"small"		"large"	
J	p	DoF	i_s	$\bar{\alpha}$	i_s	$\bar{\alpha}$	i_s	$\bar{\alpha}$	i_s	$\bar{\alpha}$	i_s	$\bar{\alpha}$	i_s	$\bar{\alpha}$
3	1	$5e^3$	48	0.79	34	0.70	74	0.85	43	0.75	38	0.75	20	0.56
	3	$4e^4$	23	0.63	24	0.59	60	0.83	36	0.70	28	0.68	18	0.53
	6	$2e^5$	23	0.63	22	0.55	58	0.82	34	0.68	27	0.69	16	0.49
	9	$4e^5$	23	0.63	19	0.50	58	0.82	31	0.65	25	0.69	14	0.46
4	1	$2e^4$	52	0.80	40	0.74	87	0.87	48	0.77	39	0.76	23	0.60
	3	$2e^5$	27	0.68	26	0.60	66	0.84	41	0.72	28	0.70	23	0.60
	6	$1e^6$	26	0.66	24	0.57	68	0.84	38	0.70	29	0.72	20	0.58
	9	$2e^5$	26	0.67	21	0.53	70	0.84	33	0.67	28	0.72	18	0.55
5	1	$1e^5$	56	0.81	43	0.75	97	0.88	52	0.78	40	0.76	25	0.63
	3	$1e^6$	32	0.73	28	0.61	72	0.85	44	0.74	30	0.72	27	0.65
	6	$3e^6$	29	0.71	26	0.58	78	0.86	42	0.72	31	0.74	25	0.63
	9	$6e^6$	29	0.71	24	0.56	84	0.86	38	0.71	30	0.74	21	0.59

Table 1.1: dAS construction (1.37): problems (1.32)–(1.34), $p' = p$ in (1.10a), "small" and "large" patches. i_s: the number of iterations needed to reach the stopping criterion (1.35). $\bar{\alpha}$: average error contraction factor given by (1.36).

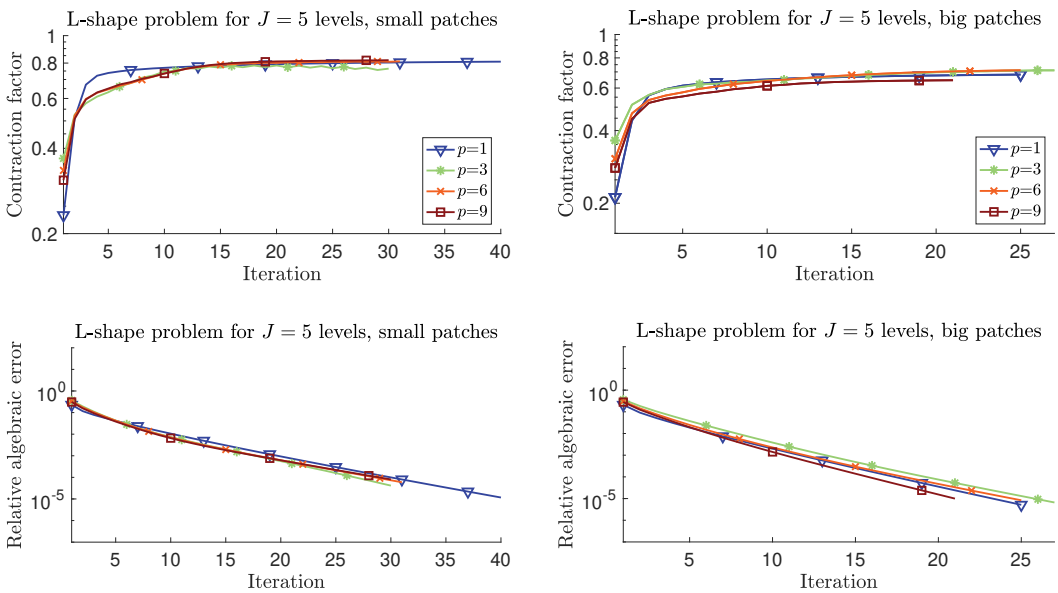


Figure 1.5: L-shape problem (1.34), $w_1 = d+1$, $w_2 = J$: results of the solver (1.27) for $p' = p$ in (1.10a), "small" (left) and "large" (right) patches, and stopping criterion (1.35). Top: error contraction factors $\|\nabla(u_J - u_J^{i+1})\|/\|\nabla(u_J - u_J^i)\|$. Bottom: relative algebraic error $\|\nabla(u_J - u_J^i)\|/\|\nabla u_J\|$.

An inferior quality of the contraction factors for the case of $p = 1$ and the use of damping factors $w_1 = J(d+1)$ and $w_2 = 1$ appears. This is in line with some precedents in literature, where numerically p-robust solvers also perform worse for order 1 approximations; we mention, for example, [Griebel et al., 2005, Table 1] and [Kronbichler and Wall, 2018, Table 1]. Recall that we consider no pre-

smoothing and only one post-smoothing step; an important drop in the number of iterations appears if more smoothing steps are employed, which will be explored below. Another observation is that the number of iterations depends on the number of mesh levels J, in accordance with the theoretical result of Section 7, even though rather mildly.

The behavior of the contraction factor in each iteration in Figures 1.3–1.5 appears quite different. This seems to be related partly to the smoothness of the problem and partly to choice of the damping weights. We explore this in more detail in Figure 1.6 by using different choices of the weights and number of post-smoothing steps ν. In particular the degradation of the contraction factors observed in Figure 1.3 disappears when employing more smoothing steps.

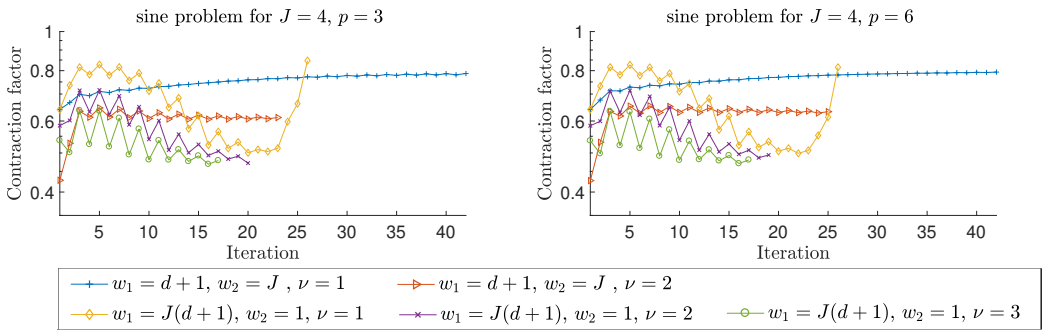


Figure 1.6: Sine problem (1.32), "small" patches, $p' = p$: study of the contraction factor behavior with respect to the number of post-smoothing steps and damping weights for the solver (1.27).

6.2 Performance of the weighted restrictive additive Schwarz (wRAS) construction of the solver

As observed in the literature, replacing the damping with parameter w_1 in (1.19) by hat function weighting via a restrictive additive Schwarz often performs better; cf. Cai and Sarkis [1999], Efstathiou and Gander [2003], or Loisel et al. [2008]. Thus, in addition to the dAS construction (1.37), we now numerically also explore the weighted restricted additive Schwarz (wRAS) construction of the lifting $\rho^i_{J,\mathrm{alg}}$:

$$\text{wRAS:} \quad \rho^i_{J,\mathrm{alg}} := \sum_{j=0}^{J} \rho^i_j \quad \text{and} \quad \rho^i_j := \sum_{\mathbf{a}\in\mathcal{V}_{j-s}} \mathcal{I}^q_j(\psi^{\mathbf{a}}_{j-s}\rho^i_{j,\mathbf{a}}),\ 1 \le j \le J, \tag{1.38}$$

$$(\nabla\rho^i_{j,\mathbf{a}}, \nabla v_{j,\mathbf{a}})_{\omega^{\mathbf{a}}_{j,s}} = (f, v_{j,\mathbf{a}})_{\omega^{\mathbf{a}}_{j,s}} - (\nabla u^i_J, \nabla v_{j,\mathbf{a}})_{\omega^{\mathbf{a}}_{j,s}} - \sum_{k=0}^{j-1}(\nabla\rho^i_k, \nabla v_{j,\mathbf{a}})_{\omega^{\mathbf{a}}_{j,s}}$$

with $q = p'$ except for $j = J$ where $q = p$, we denote by $\mathcal{I}^q_j$ the $\mathbb{P}_q$ Lagrange interpolation operator on the mesh level j, i.e., $\mathcal{I}^q_j : C^0(\overline{\Omega}) \to V^q_j$, $\mathcal{I}^q_j(v)$ preserves the values of v in the nodes corresponding to the Lagrange degrees of freedom. No damping weights are to be chosen here.

We summarize the results obtained for each of the problems (1.32)–(1.34) in Table 1.2. In addition to one post-smoothing step, $\nu = 1$, we also present the results for $\nu = 3$ post-smoothing steps. In both cases, no pre-smoothing has been

employed. In the last two columns for each problem, we present a comparison of our solver of Definition 4.4 employing (1.38) with two standard smoothers for multigrid, namely the Jacobi (J) and the Gauss–Seidel (GS) ones. Here, we employ no pre-smoothing step, one post-smoothing step, and a coarse solve with polynomials of order 1 as in (1.15) to compare with our approach.

The results using the wRAS (1.38) construction of the lifting indicate an improvement in the error contraction factors with respect to dAS (1.37) of Section 6.1 and, moreover, present a complete numerical independence of the number of levels J. Furthermore, the iteration numbers drop by at least half when three post-smoothing steps are employed. In contrast to these results, we see that the multigrid with standard smoothers degrades violently with respect to the polynomial degree p. Note in this respect that for $p = 1$, the only difference between wRAS of (1.38) with small patches and $\nu = 1$ and standard Jacobi lies in the optimally chosen step-size of Lemma 4.3. This gives a spectacular gain in the number of iterations, and makes the method convergent even when the standard Jacobi fails.

		Sine problem (1.32)						Peak problem (1.33)						L-shape problem (1.34)					
		wRAS				MG		wRAS				MG		wRAS				MG	
		"small"		"large"		$\nu = 1$		"small"		"large"		$\nu = 1$		"small"		"large"		$\nu = 1$	
J	p	$\nu = 1$	$\nu = 3$	$\nu = 1$	$\nu = 3$	J	GS	$\nu = 1$	$\nu = 3$	$\nu = 1$	$\nu = 3$	J	GS	$\nu = 1$	$\nu = 3$	$\nu = 1$	$\nu = 3$	J	GS
3	1	21	10	9	4	-	10	19	9	9	4	68	8	17	9	8	4	44	9
	3	15	5	6	3	-	81	15	6	6	3	-	70	12	4	5	3	-	49
	6	13	5	6	3	-	470	14	6	6	3	-	462	10	4	5	2	-	228
	9	13	5	6	3	-	+600	14	6	5	3	-	+600	10	4	5	2	-	586
4	1	23	11	9	4	-	11	20	9	9	4	-	10	18	9	8	4	-	9
	3	15	5	6	3	-	81	15	6	5	3	-	79	12	4	5	3	-	42
	6	13	5	6	3	-	468	14	6	5	3	-	460	10	4	5	2	-	186
	9	13	5	5	3	-	+600	14	6	5	3	-	+600	9	4	5	2	-	454
5	1	22	11	9	4	-	11	20	11	9	4	-	11	17	9	8	4	-	8
	3	15	5	6	3	-	81	15	6	5	3	-	80	12	4	5	3	-	35
	6	13	5	6	3	-	470	14	6	5	3	-	461	9	4	5	2	-	147
	9	13	5	5	3	-	+600	13	6	5	3	-	+600	8	3	4	2	-	333

Table 1.2: Number of iterations needed to reach the stopping criterion (1.35): wRAS construction (1.38), problems (1.32)–(1.34), $p' = p$ in (1.10a), "small" and "large" patches, ν post-smoothing steps, and standard multigrid method with piecewise affine coarse solve (1.15), initialized by the coarse grid solution, no pre-smoothing, one post-smoothing step, and Jacobi (J) and Gauss–Seidel (GS) smoothers.

6.3 Comparison with other multilevel solvers

Some recent comparisons of state-of-the-art solvers for Poisson problems with multigrid methods in the high-order setting include Gholami et al. [2016], Sundar et al. [2015], and Kronbichler and Wall [2018]. In Sundar et al. [2015], it was in particular reported that none of the methods considered behaves fully independently of the polynomial degree. In this subsection, we compare our developments with four well-established methods. We focus on the number of iterations, but we also indicate CPU times of our vectorized MATLAB implementation[1], trusting the reader to

[1]The codes were prepared to benefit as much as possible from MATLAB's fast operations on matrices and vectors. The experiments were run on one **Dell C6220** dual-Xeon E5-2650 node of Inria Sophia Antipolis - Méditerranée "NEF" computation cluster, in a sequential MATLAB script.

understand the trickiness inherent in such implementation- and machine-dependent measurements. The timings below involve the solution time only; i.e., they do not include the assembly time of the matrices. The methods we consider for the comparison are as follows:

wRAS, $\nu = 1$ (∼MG(0,1)-bJ): Definition 4.4, small patches, $p' = p$ in (1.10a) (to illustrate the associated space hierarchy, we write "$1, p \to p$"), wRAS construction (1.38).

wRAS, $\nu = 3$ (∼MG(0,3)-bJ): Definition 4.4, small patches, $p' = p$ in (1.10a) ("$1, p \to p$"), wRAS construction (1.38), three post-smoothing steps employed.

wRAS, $\nu = 1$ (∼MG(0,1)-bJ): Definition 4.4, small patches, $p' = 1$ in (1.10a) ("$1 \to 1, p$"), wRAS construction (1.38).

wRAS, $\nu = 3$ (∼MG(0,3)-bJ): Definition 4.4, small patches, $p' = 1$ in (1.10a) ("$1 \to 1, p$"), wRAS construction (1.38), three post-smoothing steps employed.

PCG(MG(3,3)-bJ): Preconditioned conjugate gradient solver; the preconditioner is multigrid V-cycle(3,3) with weighted restrictive additive Schwarz (block Jacobi) smoother associated to small patches; the space hierarchy relies on order p discretization, including the coarsest space ("$p \to p$"); the iterations start with the zero vector. This choice of solver is motivated by Antonietti and Pennesi [2019], adapted to the conforming finite elements setting.

MG(1,1)-PCG(iChol): Multigrid solver V-cycle(1,1); the smoother is PCG with incomplete zero level fill-in Cholesky preconditioner; the space hierarchy is of increasing order: from order 1 for the coarsest level to order p for the finest level ("$1 \nearrow p$"); the iterations start with the zero vector. This choice of solver is motivated by Botti et al. [2017], adapted for a symmetric setting.

MG(0,1)-bGS: Multigrid solver V-cycle(0,1); the smoother is block Gauss–Seidel associated to small patches; the space hierarchy consists of order 1 for all levels except the finest level, which is of order p ("$1 \to 1, p$"), i.e., as in (1.10a) with p'=1; the iterations start with the zero vector. This choice of the solver is motivated by NGSolve Schöberl [2014]; however, the multigrid is used here as a solver instead of a preconditioner.

MG(0,3)-bGS: Multigrid solver analogous to MG(0,1)-bGS where now three post-smoothing steps are employed.

MG(3,3)-GS: Multigrid solver V-cycle(3,3); the smoother is standard Gauss–Seidel; the space hierarchy is of increasing order: from order 1 for coarse level to order p for the finest level ("$1 \nearrow p$"); the iterations start with the zero vector.

		wRAS $\nu=1$ $1,p\to p$		wRAS $\nu=3$ $1,p\to p$		wRAS $\nu=1$ $1\to 1,p$		wRAS $\nu=3$ $1\to 1,p$		PCG(MG (3,3)-bJ) $p\to p$		MG(1,1)-PCG(iChol) $1\nearrow p$		MG(0,1)-bGS $1\to 1,p$		MG(0,3)-bGS $1\to 1,p$		MG(3,3)-GS $1\nearrow p$	
J	p	i_s	time	i_s	time	i_s	time	i_s	time	i_s	time	i_s	time	i_s	time	i_s	time	i_s	time
3	1	17	0.0 s	9	0.0 s	17	0.0 s	9	0.0 s	7	0.0 s	4	0.0 s	9	0.0 s	4	0.0 s	3	0.0 s
	3	12	0.2 s	4	0.1 s	18	0.2 s	6	0.1 s	3	0.2 s	14	0.6 s	8	0.6 s	4	0.8 s	4	0.1 s
	6	10	1.8 s	4	1.7 s	15	1.9 s	6	2.0 s	2	2.0 s	21	8.6 s	7	1.8 s	4	2.7 s	9	1.5 s
	9	10	9.9 s	4	10.2 s	14	9.7 s	6	11.2 s	2	10.1 s	63	1.2m	6	6.9 s	3	8.7 s	9	5.3 s
4	1	18	0.0 s	9	0.0 s	18	0.0 s	9	0.0 s	8	0.1 s	7	0.1 s	9	0.0 s	4	0.0 s	3	0.0 s
	3	12	0.8 s	4	0.6 s	18	0.8 s	6	0.6 s	3	0.7 s	29	5.6 s	8	2.4 s	4	3.4 s	4	0.3 s
	6	10	7.3 s	4	7.4 s	15	7.8 s	6	7.9 s	3	10.9 s	49	1.2 m	7	8.6 s	3	9.4 s	5	3.5 s
	9	9	34.7 s	4	40.7 s	13	37.2 s	5	37.4 s	2	39.3 s	167	12.5m	6	28.3 s	3	36.7 s	8	20.7 s
5	1	17	0.1 s	9	0.1 s	17	0.1 s	9	0.1 s	8	0.2 s	19	1.2 s	8	0.1 s	4	0.1 s	3	0.1 s
	3	12	3.2 s	4	2.3 s	17	3.4 s	6	2.6 s	3	3.1 s	77	57.7 s	8	10.7 s	4	15.7 s	4	1.5 s
	6	9	27.6 s	4	30.3 s	14	32.0 s	6	33.8 s	3	45.6 s	129	11.6m	7	30.8 s	3	33.7 s	4	12.8 s
	9	8	2.3m	3	2.1m	12	2.3m	5	2.5m	2	3.1m	+200	+1.0 h	6	2.2m	3	2.7m	8	1.3m

Table 1.3: Comparison of various multilevel solvers (described in Section 6.3) for the L-shape case (1.34), i_s is the number of iterations to reach the stopping criterion (1.35).

As one can see from Table 1.3, the presented methods split into two groups: numerically p-robust (wRAS, PCG(MG-bJ), MG-bGS) and not (MG-PCG(iChol), MG-GS). Note that the choice of three pre- and three post-smoothing steps makes every iteration of the methods PCG(MG(3,3)-bJ) and MG(3,3)-GS considerably more expensive than those of the methods wRAS and MG-bGS with $\nu = 1$, where the minimalist (0,1) choice is sufficient. The variants wRAS and MG-bGS with $\nu = 3$ are also cheaper. In addition, in PCG(MG(3,3)-bJ), the coarse grid correction is more expensive as it uses order p approximations. The inversion of the Jacobi blocks in PCG(MG(3,3)-bJ) on the finest level J, corresponds to solving the patch problems of order p as in (1.20), so that its cost is the same as for the local problems of wRAS. As for MG(1,1)-PCG(iChol), we find the method to be quite satisfactory for lower-order approximations and small J, but as soon as p and J increase, the number of iterations degrades considerably. In contrast to wRAS, MG-bGS is a multiplicative Schwarz method and is thus less suitable for parallelization. Finally, the classical MG(3,3)-GS is a combination of h- and p-multigrid and gives the best timings in our experiments. The numbers of pre- and post-smoothing steps, however, remain parameters, and their tuning might not be straightforward in order to get an efficient and numerically robust multigrid solver in general (cf. the poor results of the very similar—up to the different number of pre- and post-smoothing steps and a stronger hierarchy—MG(0,1)-GS version in Table 1.2). The Gauss–Seidel smoother used therein again makes the method harder to parallelize.

7 Proofs of the main results

As shown in Corollary 5.4, the results of Theorems 5.1 and 5.3 are equivalent. Therefore it suffices to prove the first one. Our approach to proving Theorem 5.1 consists in studying the uncomputable exact residual lifting $\tilde{\rho}^i_{J,\mathrm{alg}}$ given by (1.13) and its approximation $\rho^i_{J,\mathrm{alg}}$ given by Definition 3.2. In particular, we will estimate p-robustly the quantities $\|\nabla\tilde{\rho}^i_{J,\mathrm{alg}}\|$, $\|\nabla\rho^i_{J,\mathrm{alg}}\|$, and $(f,\rho^i_{J,\mathrm{alg}}) - (\nabla u^i_J, \nabla\rho^i_{J,\mathrm{alg}})$ by

local contributions $\rho^i_{j,\mathbf{a}}$ of (1.20) used to construct $\rho^i_{J,\mathrm{alg}}$, and we show that

$$\frac{(f,\rho^i_{J,\mathrm{alg}}) - (\nabla u^i_J, \nabla \rho^i_{J,\mathrm{alg}})}{\|\nabla \rho^i_{J,\mathrm{alg}}\|} \geq \beta \|\nabla \tilde{\rho}^i_{J,\mathrm{alg}}\| \qquad \text{when } \rho^i_{J,\mathrm{alg}} \neq 0,$$

$$\tilde{\rho}^i_{J,\mathrm{alg}} = 0 \qquad \text{when } \rho^i_{J,\mathrm{alg}} = 0,$$

which also establishes Corollary 5.5.

7.1 Upper bound on $\|\nabla \rho^i_{J,\mathrm{alg}}\|$

We present here properties of the constructed residual lifting $\rho^i_{J,\mathrm{alg}}$ and its levelwise components ρ^i_j, where $1 \leq j \leq J$.

Lemma 7.1 (Estimate on $\|\nabla \rho^i_{J,\mathrm{alg}}\|$ and $\|\nabla \rho^i_j\|$ by patchwise contributions)**.** *Let $\rho^i_{J,\mathrm{alg}}$ and ρ^i_j for $j \in \{1,\dots,J\}$ be given by Definition 3.2. Then*

$$\|\nabla \rho^i_j\|^2 \leq \frac{d+1}{w_1^2} \sum_{\mathbf{a}\in\mathcal{V}_{j-s}} \|\nabla \rho^i_{j,\mathbf{a}}\|^2_{\omega^{\mathbf{a}}_{j,s}}, \tag{1.39}$$

$$\|\nabla \rho^i_{J,\mathrm{alg}}\|^2 \leq C^2_{\max}(w_1)\left(\|\nabla \rho^i_0\|^2 + \sum_{j=1}^{J}\sum_{\mathbf{a}\in\mathcal{V}_{j-s}} \|\nabla \rho^i_{j,\mathbf{a}}\|^2_{\omega^{\mathbf{a}}_{j,s}}\right), \tag{1.40}$$

where

$$2 \leq C^2_{\max}(w_1) := 2\max\left(1, \frac{J(d+1)}{w_1^2}\right) \leq 2J(d+1). \tag{1.41}$$

Proof. Definition 3.2 and inequality $|\sum_{k=1}^{d+1} a_k|^2 \leq (d+1)\sum_{k=1}^{d+1}|a_k|^2$ lead to

$$\begin{aligned}\|\nabla \rho^i_j\|^2 &= \sum_{K\in\mathcal{T}_{j-s}} \|\nabla \rho^i_j\|^2_K = \sum_{K\in\mathcal{T}_{j-s}} \left\|\frac{1}{w_1}\sum_{\mathbf{a}\in\mathcal{V}_K} \nabla \rho^i_{j,\mathbf{a}}\right\|^2_K \\ &\leq \frac{d+1}{w_1^2}\sum_{K\in\mathcal{T}_{j-s}}\sum_{\mathbf{a}\in\mathcal{V}_K} \|\nabla \rho^i_{j,\mathbf{a}}\|^2_K = \frac{d+1}{w_1^2}\sum_{\mathbf{a}\in\mathcal{V}_{j-s}} \|\nabla \rho^i_{j,\mathbf{a}}\|^2_{\omega^{\mathbf{a}}_{j,s}}.\end{aligned}$$

Note that this allows us to write

$$\left\|\sum_{j=1}^{J} \nabla \rho^i_j\right\|^2 \leq J\sum_{j=1}^{J}\|\nabla \rho^i_j\|^2 \leq \frac{J(d+1)}{w_1^2}\sum_{j=1}^{J}\sum_{\mathbf{a}\in\mathcal{V}_{j-s}} \|\nabla \rho^i_{j,\mathbf{a}}\|^2_{\omega^{\mathbf{a}}_{j,s}}. \tag{1.42}$$

This property together with some simple manipulations, gives the second estimate:

$$\begin{aligned}\|\nabla \rho^i_{J,\mathrm{alg}}\|^2 &\leq 2\|\nabla \rho^i_0\|^2 + 2\left\|\sum_{j=1}^{J}\nabla \rho^i_j\right\|^2 \\ &\overset{(1.42)}{\leq} 2\|\nabla \rho^i_0\|^2 + \frac{2J(d+1)}{w_1^2}\sum_{j=1}^{J}\sum_{\mathbf{a}\in\mathcal{V}_{j-s}} \|\nabla \rho^i_{j,\mathbf{a}}\|^2_{\omega^{\mathbf{a}}_{j,s}} \\ &\leq 2\max\left(1, \frac{J(d+1)}{w_1^2}\right)\left(\|\nabla \rho^i_0\|^2 + \sum_{j=1}^{J}\sum_{\mathbf{a}\in\mathcal{V}_{j-s}} \|\nabla \rho^i_{j,\mathbf{a}}\|^2_{\omega^{\mathbf{a}}_{j,s}}\right).\end{aligned}$$

The bounds (1.41) on $C^2_{\max}(w_1)$ are easily obtained by using $w_1 \geq 1$ requested in Definition 3.2. □

7.2 Lower bound on $(f, \rho^i_{J,\mathrm{alg}}) - (\nabla u^i_J, \nabla \rho^i_{J,\mathrm{alg}})$

While studying the term $(f, \rho^i_{J,\mathrm{alg}}) - (\nabla u^i_J, \nabla \rho^i_{J,\mathrm{alg}})$, the interaction of different level contributions ρ^i_j of the lifting $\rho^i_{J,\mathrm{alg}}$ arises naturally. In order to estimate these terms, the damping parameters w_1, w_2 used in the construction (1.19) of our lifting prove to be essential.

Lemma 7.2 (Estimate on $(f, \rho^i_{J,\mathrm{alg}}) - (\nabla u^i_J, \nabla \rho^i_{J,\mathrm{alg}})$ from below by patchwise contributions)**.** *Let $\rho^i_{J,\mathrm{alg}}$ be given by Definition 3.2. Then*

$$(f, \rho^i_{J,\mathrm{alg}}) - (\nabla u^i_J, \nabla \rho^i_{J,\mathrm{alg}}) \geq C^2_{\min}(w_1, w_2) \left(\|\nabla \rho^i_0\|^2 + \sum_{j=1}^{J} \sum_{\mathbf{a} \in \mathcal{V}_{j-s}} \|\nabla \rho^i_{j,\mathbf{a}}\|^2_{\omega^{\mathbf{a}}_{j,s}} \right), \tag{1.43}$$

where

$$\frac{1}{6J(d+1)} \leq C^2_{\min}(w_1, w_2) := \min\left(\frac{1}{4}\,,\ \frac{1}{w_1} - \frac{J(1+\frac{2}{3w_2})(d+1)}{2w_2 w_1^2} \right) \leq \frac{1}{4}\,. \tag{1.44}$$

Proof. We begin by using the construction of $\rho^i_{J,\mathrm{alg}}$ given in Definition 3.2 to write

$$\begin{aligned}
(f, \rho^i_{J,\mathrm{alg}}) - (\nabla u^i_J, \nabla \rho^i_{J,\mathrm{alg}}) &= (f, \rho^i_0) - (\nabla u^i_J, \nabla \rho^i_0) + \sum_{j=1}^{J} \Big((f, \rho^i_j) - (\nabla u^i_J, \nabla \rho^i_j) \Big) \\
&\overset{(1.15)}{\underset{(1.19)}{=}} \|\nabla \rho^i_0\|^2 + \frac{1}{w_1} \sum_{j=1}^{J} \sum_{\mathbf{a} \in \mathcal{V}_{j-s}} \Big((f, \rho^i_{j,\mathbf{a}})_{\omega^{\mathbf{a}}_{j,s}} - (\nabla u^i_J, \nabla \rho^i_{j,\mathbf{a}})_{\omega^{\mathbf{a}}_{j,s}} \Big) \\
&\overset{(1.20)}{=} \|\nabla \rho^i_0\|^2 + \frac{1}{w_1} \sum_{j=1}^{J} \sum_{\mathbf{a} \in \mathcal{V}_{j-s}} \left(\|\nabla \rho^i_{j,\mathbf{a}}\|^2_{\omega^{\mathbf{a}}_{j,s}} + \frac{1}{w_2} \sum_{k=0}^{j-1} (\nabla \rho^i_k, \nabla \rho^i_{j,\mathbf{a}})_{\omega^{\mathbf{a}}_{j,s}} \right) \\
&\overset{(1.19)}{=} \|\nabla \rho^i_0\|^2 + \frac{1}{w_1} \sum_{j=1}^{J} \sum_{\mathbf{a} \in \mathcal{V}_{j-s}} \|\nabla \rho^i_{j,\mathbf{a}}\|^2_{\omega^{\mathbf{a}}_{j,s}} + \frac{1}{w_2} \sum_{j=1}^{J} \sum_{k=0}^{j-1} (\nabla \rho^i_k, \nabla \rho^i_j).
\end{aligned}$$

The first two terms above are of the right form to prove the result, but one needs to be a bit more careful with the third one. We estimate it using Young's inequality and the sum interchange $\sum_{j=2}^{J} \sum_{k=1}^{j-1} = \sum_{k=1}^{J-1} \sum_{j=k+1}^{J}$; Young's parameter μ is picked later to control the dependence on J of the final estimate. We have

$$\begin{aligned}
&\frac{1}{w_2} \sum_{j=1}^{J} \sum_{k=0}^{j-1} (\nabla \rho^i_k, \nabla \rho^i_j) = \frac{1}{w_2} \left(\sum_{j=2}^{J} \sum_{k=1}^{j-1} (\nabla \rho^i_k, \nabla \rho^i_j) + \sum_{j=1}^{J} (\nabla \rho^i_0, \nabla \rho^i_j) \right) \\
&\geq \frac{1}{w_2} \sum_{j=2}^{J} \sum_{k=1}^{j-1} \left(-\frac{1}{2} \|\nabla \rho^i_k\|^2 - \frac{1}{2} \|\nabla \rho^i_j\|^2 \right) + \frac{1}{w_2} \sum_{j=1}^{J} \left(-\frac{1}{2\mu} \|\nabla \rho^i_0\|^2 - \frac{\mu}{2} \|\nabla \rho^i_j\|^2 \right) \\
&= -\frac{1}{2w_2} \sum_{j=1}^{J} (J-j) \|\nabla \rho^i_j\|^2 - \frac{1}{2w_2} \sum_{j=1}^{J} (j-1) \|\nabla \rho^i_j\|^2 - \frac{J}{2\mu w_2} \|\nabla \rho^i_0\|^2 - \frac{\mu}{2w_2} \sum_{j=1}^{J} \|\nabla \rho^i_j\|^2,
\end{aligned}$$

where we added the terms in the sum corresponding to $k = J$ and $j = 1$ since they are zero, and then renamed the summation index when there is no confusion.

Picking Young's inequality parameter $\mu = \frac{2J}{3w_2}$, a few more manipulations on the right-hand side give us

$$\begin{aligned}\frac{1}{w_2}\sum_{j=1}^{J}\sum_{k=0}^{j-1}(\nabla\rho_k^i,\nabla\rho_j^i) &\geq -\frac{3}{4}\|\nabla\rho_0^i\|^2 - \frac{J-1+\frac{2J}{3w_2}}{2w_2}\sum_{j=1}^{J}\|\nabla\rho_j^i\|^2\\ &\overset{(1.39)}{\geq} -\frac{3}{4}\|\nabla\rho_0^i\|^2 - \frac{J(1+\frac{2}{3w_2})(d+1)}{2w_2w_1^2}\sum_{j=1}^{J}\sum_{\mathbf{a}\in\mathcal{V}_{j-s}}\|\nabla\rho_{j,\mathbf{a}}^i\|^2_{\omega_{j,s}^{\mathbf{a}}}.\end{aligned}$$

We return to the main estimate and obtain the result by using definition (1.44)

$$\begin{aligned}&(f,\rho_{J,\mathrm{alg}}^i) - (\nabla u_J^i,\nabla\rho_{J,\mathrm{alg}}^i)\\ &\geq \frac{1}{4}\|\nabla\rho_0^i\|^2 + \left(\frac{1}{w_1} - \frac{J(1+\frac{2}{3w_2})(d+1)}{2w_2w_1^2}\right)\sum_{j=1}^{J}\sum_{\mathbf{a}\in\mathcal{V}_{j-s}}\|\nabla\rho_{j,\mathbf{a}}^i\|^2_{\omega_{j,s}^{\mathbf{a}}}\\ &\overset{(1.44)}{\geq} C_{\min}^2(w_1,w_2)\left(\|\nabla\rho_0^i\|^2 + \sum_{j=1}^{J}\sum_{\mathbf{a}\in\mathcal{V}_{j-s}}\|\nabla\rho_{j,\mathbf{a}}^i\|^2_{\omega_{j,s}^{\mathbf{a}}}\right).\end{aligned} \tag{1.45}$$

The upper bound on $C_{\min}^2(w_1,w_2)$ in (1.44) is immediate due to the minimum in its expression, while the lower bound is obtained by rewriting condition (1.17) on w_2

$$w_2 \geq \frac{5J^2(d+1)^2}{w_1 6J(d+1)\left(1-\frac{w_1}{6J(d+1)}\right)} \Leftrightarrow \frac{1}{w_1} - \frac{5}{6}\frac{J(d+1)}{w_2w_1^2} \geq \frac{1}{6J(d+1)}. \tag{1.46}$$

As also $w_2 \geq 1$,

$$\begin{aligned}\frac{1}{w_1} - \frac{J(1+\frac{2}{3w_2})(d+1)}{2w_2w_1^2} &\geq \frac{1}{w_1} - \frac{J(1+\frac{2}{3})(d+1)}{2w_2w_1^2} = \frac{1}{w_1} - \frac{5}{6}\frac{J(d+1)}{w_2w_1^2}\\ &\overset{(1.46)}{\geq} \frac{1}{6J(d+1)}.\end{aligned}$$

□

7.3 Polynomial-degree-robust multilevel stable decomposition

Now, we devise a p-robust multilevel stable decomposition. This decomposition relies on the one level p-robust stable decomposition given in [Schöberl et al., 2008, Proof of Theorem 2.1] and the piecewise affine multilevel decomposition in the spirit of [Xu et al., 2009, Theorems 3.1 and 4.3]. These results are presented below in the form of lemmas. Note that in the decomposition, only "small" patches are used, which will be sufficient for our purposes. Recall also the definition of the local spaces (1.12), which will be useful below. Hereafter, we always assume that Assumption 2.1 is satisfied.

By [Schöberl et al., 2008, Proof of Theorem 2.1], we have the following.

Lemma 7.3 (One-level p-robust stable decomposition)**.** *For all $v_J \in V_J^p$, there exists a finest-level decomposition $v_J = v_J^{\#} + \sum_{\mathbf{b}\in\mathcal{V}_J} v_{J,\mathbf{b}}^p$ where $v_J^{\#} \in V_J^1$ and $v_{J,\mathbf{b}}^p \in V_{J,0}^{\mathbf{b}}$, $\mathbf{b}\in\mathcal{V}_J$, and this decomposition is stable in the sense*

$$\|\nabla v_J^{\#}\|^2 + \sum_{\mathbf{b}\in\mathcal{V}_J}\|\nabla v_{J,\mathbf{b}}^p\|^2_{\omega_{J,0}^{\mathbf{b}}} \leq C_{\mathrm{SD}}^2\|\nabla v_J\|^2, \tag{1.47}$$

where $C_{\mathrm{SD}} \geq 1$ only depends on the mesh shape regularity parameter $\kappa_{\mathcal{T}}$ and space dimension d.

Similarly to [Xu et al., 2009, Lemma 3.1 and Theorem 3.1], in the case of quasi-uniform meshes with bounded refinement strength, we have the following.

Lemma 7.4 ($\mathbb{P}_1$-multilevel stable decomposition for quasi-uniform meshes)**.** *For all $v_J^{\#} \in V_J^1$, there exists a multilevel piecewise affine decomposition $v_J^{\#} = v_0^1 + \sum_{j=1}^J \sum_{\mathbf{b}\in\mathcal{V}_j} v_{j,\mathbf{b}}^1$ with $v_0^1 \in V_0^1$ and $v_{j,\mathbf{b}}^1 \in V_{j,0}^{\mathbf{b}} = \mathbb{P}_1(\mathcal{T}_{j,0}^{\mathbf{b}}) \cap H_0^1(\omega_{j,0}^{\mathbf{b}})$. Under Assumption 2.2, this decomposition is stable as*

$$\|\nabla v_0^1\|^2 + \sum_{j=1}^J \sum_{\mathbf{b}\in\mathcal{V}_j} \|\nabla v_{j,\mathbf{b}}^1\|_{\omega_{j,0}^{\mathbf{b}}}^2 \leq C_{\mathrm{MD}}^2 \|\nabla v_J^{\#}\|^2, \tag{1.48}$$

where $C_{\mathrm{MD}} \geq 1$ only depends on the space dimension d, the mesh shape regularity parameter $\kappa_{\mathcal{T}}$, the maximum strength of refinement parameter C_{ref}, and the quasi-uniformity parameter C_{qu}.

Proof. Let $v_J^{\#} \in V_J^1$. We first apply a levelwise decomposition that follows from [Xu et al., 2009, Lemma 3.1] by keeping the gradient on level zero. This gives us $v_J^{\#} = \sum_{j=0}^J v_j^1$ with $v_j^1 \in V_j^1$ such that

$$\|\nabla v_0^1\|^2 + \sum_{j=1}^J h_j^{-2} \|v_j^1\|^2 \leq C_{\mathrm{ml}}^2 \|\nabla v_J^{\#}\|^2, \tag{1.49}$$

where $C_{\mathrm{ml}} \geq 1$ has the same dependencies as C_{MD}.

We further decompose each of the above $v_j^1 \in V_j^1$, $j \geq 1$, into patchwise components. For this purpose, we use the standard nodal decomposition $v_j^1 = \sum_{\mathbf{b}\in\mathcal{V}_j} v_{j,\mathbf{b}}^1$, where $v_{j,\mathbf{b}}^1 = v_j^1(\mathbf{b})\psi_{j,\mathbf{b}}$ belongs to the local space $V_{j,0}^{\mathbf{b}}$ for $p = p' = 1$. By stability of the $\mathbb{P}_1$ nodal decomposition,

$$\sum_{\mathbf{b}\in\mathcal{V}_j} \|v_{j,\mathbf{b}}^1\|_{\omega_{j,0}^{\mathbf{b}}}^2 \leq C_{\mathrm{nd}}^2 \|v_j^1\|^2, \tag{1.50}$$

where C_{nd} only depends on the space dimension d and the mesh shape regularity parameter $\kappa_{\mathcal{T}}$. This can, for instance, be shown by considering a patch $\omega_{j,0}^{\mathbf{b}}$ and an element K contained in the patch. Since $v_{j,\mathbf{b}}^1 = v_j^1(\mathbf{b})\psi_{j,\mathbf{b}} \in V_{j,0}^{\mathbf{b}}$ and by equivalence of norms in finite dimension, we have

$$\begin{aligned}\|v_j^1(\mathbf{b})\psi_{j,\mathbf{b}}\|_{\omega_{j,0}^{\mathbf{b}}} &\approx \|v_j^1(\mathbf{b})\psi_{j,\mathbf{b}}\|_K \leq \|v_j^1(\mathbf{b})\psi_{j,\mathbf{b}}\|_{\infty,K}|K|^{\frac{1}{2}} \\ &\leq \Big\| \sum_{\mathbf{b}\in\mathcal{V}_K} v_j^1(\mathbf{b})\psi_{j,\mathbf{b}} \Big\|_{\infty,K} |K|^{\frac{1}{2}} = \|v_j^1\|_{\infty,K}|K|^{\frac{1}{2}} \lesssim \|v_j^1\|_K. \end{aligned} \tag{1.51}$$

The result (1.50) is obtained by summing both sides over all vertices.

Now, the claim (1.48) follows by using an inverse inequality on patches, the quasi-uniformity of the meshes, the above decompositions as

$$\sum_{j=1}^J \sum_{\mathbf{b}\in\mathcal{V}_j} \|\nabla v_{j,\mathbf{b}}^1\|_{\omega_{j,0}^{\mathbf{b}}}^2 \leq C_{\mathrm{inv}}^2 \sum_{j=1}^J \sum_{\mathbf{b}\in\mathcal{V}_j} h_{\omega_{j,0}^{\mathbf{b}}}^{-2} \|v_{j,\mathbf{b}}^1\|_{\omega_{j,0}^{\mathbf{b}}}^2 \overset{(1.8)}{\leq} C_{\mathrm{inv}}^2 C_{\mathrm{qu}}^{-2} \sum_{j=1}^J h_j^{-2} \sum_{\mathbf{b}\in\mathcal{V}_j} \|v_{j,\mathbf{b}}^1\|_{\omega_{j,0}^{\mathbf{b}}}^2$$

$$\overset{(1.50)}{\leq} C_{\mathrm{inv}}^2 C_{\mathrm{qu}}^{-2} C_{\mathrm{nd}}^2 \sum_{j=1}^{J} h_j^{-2} \|v_j^1\|^2 \overset{(1.49)}{\leq} C_{\mathrm{inv}}^2 C_{\mathrm{qu}}^{-2} C_{\mathrm{nd}}^2 C_{\mathrm{ml}}^2 \|\nabla v_J^{\#}\|^2$$

and by summing the left-hand side with $\|\nabla v_0^1\|^2$, which satisfies a similar bound from (1.49). We set $C_{\mathrm{MD}}^2 := C_{\mathrm{inv}}^2 C_{\mathrm{qu}}^{-2} C_{\mathrm{nd}}^2 C_{\mathrm{ml}}^2 + C_{\mathrm{ml}}^2$ to obtain the result. □

By [Xu et al., 2009, Theorem 4.3], in the case of graded meshes, we have the following.

Lemma 7.5 ($\mathbb{P}_1$-multilevel stable decomposition for graded meshes)**.** *For all $v_J^{\#} \in V_J^1$, there exists a multilevel piecewise affine decomposition $v_J^{\#} = v_0^1 + \sum_{j=1}^{J} \sum_{\mathbf{b} \in \mathcal{V}_j} v_{j,\mathbf{b}}^1$ with $v_0^1 \in V_0^1$ and $v_{j,\mathbf{b}}^1 \in V_{j,0}^{\mathbf{b}} = \mathbb{P}_1(\mathcal{T}_{j,0}^{\mathbf{b}}) \cap H_0^1(\omega_{j,0}^{\mathbf{b}})$. Under Assumption 2.3, this decomposition is stable as*

$$\|\nabla v_0^1\|^2 + \sum_{j=1}^{J} \sum_{\mathbf{b} \in \mathcal{V}_j} \|\nabla v_{j,\mathbf{b}}^1\|_{\omega_{j,0}^{\mathbf{b}}}^2 \leq C_{\mathrm{MD}}^2 \|\nabla v_J^{\#}\|^2, \tag{1.52}$$

where $C_{\mathrm{MD}} \geq 1$ only depends on the space dimension d, the mesh shape regularity parameter $\kappa_{\mathcal{T}}$, the coarse mesh quasi-uniformity parameter C_{qu}^0, and the local quasi-uniformity parameter $C_{\mathrm{qu}}^{\mathrm{loc}}$.

Proof. Let $v_J^{\#} \in V_J^1$. We apply the results on stable decomposition on graded meshes of [Xu et al., 2009, Theorem 4.3]. On the one hand, this gives us the decomposition

$$v_J^{\#} = v_0^1 + \sum_{j=1}^{J} \sum_{\mathbf{b} \in \mathcal{B}_j} v_{j,\mathbf{b}}^1 + \sum_{\mathbf{b} \in \mathcal{V}_J} v_{J,\mathbf{b}}^1,$$

where $v_0^1 \in V_0^1$, $\forall \mathbf{b} \in \mathcal{B}_j$, $v_{j,\mathbf{b}}^1 \in V_{j,0}^{\mathbf{b}}$ for $p = p' = 1$, and $\forall \mathbf{b} \in \mathcal{B}_J$, $v_{J,\mathbf{b}}^1 \in V_{J,0}^{\mathbf{b}}$ for $p = 1$. On the other hand, the result also gives us the following stability inequality:

$$\|\nabla v_0^1\|^2 + \sum_{j=1}^{J} h_{\mathcal{B}_j}^{-2} \Big\| \sum_{\mathbf{b} \in \mathcal{B}_j} v_{j,\mathbf{b}}^1 \Big\|^2 + \sum_{\mathbf{b} \in \mathcal{V}_J} h_{\omega_{J,0}^{\mathbf{b}}}^{-2} \|v_{J,\mathbf{b}}^1\|_{\omega_{J,0}^{\mathbf{b}}}^2 \leq C_{\mathrm{gra}}^2 \|\nabla v_J^{\#}\|^2, \tag{1.53}$$

where C_{gra} has the same dependencies as C_{MD}.

First, since the mesh hierarchy is created via bisections, we have the local quasi-uniformity property (1.9). This, together with an inverse inequality and L^2-stability as in (1.51), gives us

$$\sum_{j=1}^{J} \sum_{\mathbf{b} \in \mathcal{B}_j} \|\nabla v_{j,\mathbf{b}}^1\|_{\omega_{j,0}^{\mathbf{b}}}^2 \leq C_{\mathrm{inv}}^2 \sum_{j=1}^{J} \sum_{\mathbf{b} \in \mathcal{B}_j} h_{\omega_{j,0}^{\mathbf{b}}}^{-2} \|v_{j,\mathbf{b}}^1\|_{\omega_{j,0}^{\mathbf{b}}}^2 \overset{(1.9)}{\leq} C_{\mathrm{inv}}^2 (C_{\mathrm{qu}}^{\mathrm{loc}})^{-2} C_{\mathrm{nd}}^2 \sum_{j=1}^{J} h_{\mathcal{B}_j}^{-2} \Big\| \sum_{\mathbf{b} \in \mathcal{B}_j} v_{j,\mathbf{b}}^1 \Big\|^2.$$

Second, we only need an inverse inequality to obtain

$$\sum_{\mathbf{b} \in \mathcal{V}_J} \|\nabla v_{J,\mathbf{b}}^1\|_{\omega_{J,0}^{\mathbf{b}}}^2 \leq C_{\mathrm{inv}}^2 \sum_{\mathbf{b} \in \mathcal{V}_J} h_{\omega_{J,0}^{\mathbf{b}}}^{-2} \|v_{J,\mathbf{b}}^1\|_{\omega_{J,0}^{\mathbf{b}}}^2.$$

Third, we can sum together the above estimations and use (1.53) to obtain

$$\|\nabla v_0^1\|^2 + \sum_{j=1}^{J} \sum_{\mathbf{b} \in \mathcal{B}_j} \|\nabla v_{j,\mathbf{b}}^1\|_{\omega_{j,0}^{\mathbf{b}}}^2 + \sum_{\mathbf{b} \in \mathcal{V}_J} \|\nabla v_{J,\mathbf{b}}^1\|_{\omega_{J,0}^{\mathbf{b}}}^2 \leq C_{\mathrm{MD}}^2 \|\nabla v_J^{\#}\|^2,$$

where $C_{\mathrm{MD}} := C_{\mathrm{gra}} \cdot \max(C_{\mathrm{inv}}, C_{\mathrm{inv}} C_{\mathrm{nd}} (C_{\mathrm{qu}}^{\mathrm{loc}})^{-1}, 1)$. Finally, since $\mathcal{B}_j \subset \mathcal{V}_j$, we can set $v_{j,\mathbf{b}}^1 := 0$ for $\mathbf{b} \in \mathcal{V}_j \setminus \mathcal{B}_j$ and have a new decomposition $v_J^{\#} = v_0^1 + \sum_{j=1}^J \sum_{\mathbf{b}\in\mathcal{V}_j} v_{j,\mathbf{b}}^1$ (reusing the notation) such that (1.52) holds. □

Proposition 7.6 (p-robust multilevel stable decomposition)**.** *Let $v_J \in V_J^p$. Under either Assumption 2.2 or 2.3, there exists a decomposition*

$$v_J = v_0^1 + \sum_{j=1}^J \sum_{\mathbf{b}\in\mathcal{V}_j} v_{j,\mathbf{b}}, \quad v_0^1 \in V_0^1,\, v_{j,\mathbf{b}} \in V_{j,0}^{\mathbf{b}} \tag{1.54}$$

stable as

$$\|\nabla v_0^1\|^2 + \sum_{j=1}^J \sum_{\mathbf{b}\in\mathcal{V}_j} \|\nabla v_{j,\mathbf{b}}\|^2_{\omega_{j,0}^{\mathbf{b}}} \le C_{\mathrm{SMD}}^2 \|\nabla v_J\|^2, \tag{1.55}$$

where $C_{\mathrm{SMD}} := \sqrt{2} C_{\mathrm{SD}} C_{\mathrm{MD}} \ge 1$ only depends on the space dimension d, the mesh shape regularity parameter $\kappa_{\mathcal{T}}$, and, depending on whether Assumption 2.2 or 2.3 is satisfied, on either the quasi-uniformity parameter C_{qu} and the maximum strength of refinement parameter C_{ref} or the coarse mesh and local quasi-uniformity parameters C_{qu}^0, $C_{\mathrm{qu}}^{\mathrm{loc}}$, respectively.

Proof. Let $v_J \in V_J^p$ and let us begin by applying the decomposition of Lemma 7.3. This gives $v_J = v_J^{\#} + \sum_{\mathbf{b}\in\mathcal{V}_J} v_{J,\mathbf{b}}^p$ with $v_J^{\#} \in V_J^1$ and $v_{J,\mathbf{b}}^p \in V_{J,0}^{\mathbf{b}}$ $\forall \mathbf{b} \in \mathcal{V}_J$. Then, we further decompose $v_J^{\#}$ using either Lemma 7.4 or 7.5, depending on whether Assumption 2.2 or 2.3 is satisfied. We obtain $v_J^{\#} = v_0^1 + \sum_{j=1}^J \sum_{\mathbf{b}\in\mathcal{V}_j} v_{j,\mathbf{b}}^1$, where $v_0^1 \in V_0^1$ and $v_{j,\mathbf{b}}^1 \in V_{j,0}^{\mathbf{b}}$ (actually $V_{j,0}^{\mathbf{b}} \cap \mathbb{P}_1(\mathcal{T}_{j,0}^{\mathbf{b}})$). We set $v_{j,\mathbf{b}} := v_{j,\mathbf{b}}^1$ $\forall \mathbf{b} \in \mathcal{V}_j$, $j \in \{1, \dots, J-1\}$ and $v_{J,\mathbf{b}} := v_{J,\mathbf{b}}^1 + v_{J,\mathbf{b}}^p$. Thus, we have $v_J = v_0^1 + \sum_{j=1}^J \sum_{\mathbf{b}\in\mathcal{V}_j} v_{j,\mathbf{b}}$ with $v_0^1 \in V_0^1$ and $v_{j,\mathbf{b}} \in V_{j,0}^{\mathbf{b}}$. The stable decomposition results presented in the previous lemmas allow us to write

$$\begin{aligned}
\|\nabla v_0^1\|^2 + \sum_{j=1}^J \sum_{\mathbf{b}\in\mathcal{V}_j} \|\nabla v_{j,\mathbf{b}}\|^2_{\omega_{j,0}^{\mathbf{b}}} &\le 2\Big(\|\nabla v_0^1\|^2 + \sum_{j=1}^J \sum_{\mathbf{b}\in\mathcal{V}_j} \|\nabla v_{j,\mathbf{b}}^1\|^2_{\omega_{j,0}^{\mathbf{b}}} + \sum_{\mathbf{b}\in\mathcal{V}_J} \|\nabla v_{J,\mathbf{b}}^p\|^2_{\omega_{J,0}^{\mathbf{b}}}\Big) \\
&\overset{(1.48)\text{ or }(1.52)}{\le} 2C_{\mathrm{MD}}^2 \Big(\|\nabla v_J^{\#}\|^2 + \sum_{\mathbf{b}\in\mathcal{V}_J} \|\nabla v_{J,\mathbf{b}}^p\|^2_{\omega_{J,0}^{\mathbf{b}}}\Big) \overset{(1.47)}{\le} 2C_{\mathrm{SD}}^2 C_{\mathrm{MD}}^2 \|\nabla v_J\|^2.
\end{aligned}$$

□

7.4 Upper bound on $\|\nabla \tilde{\rho}_{J,\mathrm{alg}}^i\|$

Recall that $\tilde{\rho}_{J,\mathrm{alg}}^i$, introduced in (1.13), is the unknown exact algebraic error. We now estimate $\|\nabla \tilde{\rho}_{J,\mathrm{alg}}^i\|$ from above. We introduce some helpful notation first. For all $\mathbf{a} \in \mathcal{V}_{j-s}$, let $\mathcal{I}_{\mathbf{a}} \subset \mathcal{V}_j$ be a set containing (fine-mesh) vertices of the interior of the patch $\omega_{j,s}^{\mathbf{a}}$ such that $\{\mathcal{I}_{\mathbf{a}}\}_{\mathbf{a}\in\mathcal{V}_{j-s}}$ cover $\mathcal{V}_j$ and are mutually disjoint; if $s = 0$, we have $\mathcal{I}_{\mathbf{a}} = \{\mathbf{a}\}$. This allows us to write $\sum_{\mathbf{b}\in\mathcal{V}_j} = \sum_{\mathbf{a}\in\mathcal{V}_{j-s}} \sum_{\mathbf{b}\in\mathcal{I}_{\mathbf{a}}}$. Moreover, since the indices of $\mathcal{I}_{\mathbf{a}}$ are localized in the interior of the patch $\omega_{j,s}^{\mathbf{a}}$, we have $\sum_{\mathbf{b}\in\mathcal{I}_{\mathbf{a}}} v_{j,\mathbf{b}} \in V_{j,s}^{\mathbf{a}}$ when $v_{j,\mathbf{b}} \in V_{j,0}^{\mathbf{b}}$. Writing it this way will help us to apply the results on the p-robust stable decomposition of Lemma 7.6 given for "small" patches only to the "large"-patch setting as well.

Lemma 7.7 (Estimating $\|\nabla\tilde{\rho}^i_{J,\mathrm{alg}}\|$ by local contributions)**.** *Let $\tilde{\rho}^i_{J,\mathrm{alg}} \in V^p_J$ be defined by (1.13). We have*

$$\|\nabla\tilde{\rho}^i_{J,\mathrm{alg}}\|^2 \le C^2_{\max}(w_1,w_2)\left(\|\nabla\rho^i_0\|^2 + \sum_{j=1}^{J}\sum_{\mathbf{a}\in\mathcal{V}_{j-s}}\|\nabla\rho^i_{j,\mathbf{a}}\|^2_{\omega^{\mathbf{a}}_{j,s}}\right), \tag{1.56}$$

where

$$\begin{aligned}2(d+1) \le C^2_{\max}(w_1,w_2) &:= 2(d+1)C^2_{\mathrm{SMD}}\left(1+\frac{J^2}{w_2}\max\left(1,\frac{d+1}{w_1^2}\right)\right)\\ &\le 4(d+1)^2C^2_{\mathrm{SMD}}J^2.\end{aligned} \tag{1.57}$$

Proof. The main ingredient of the proof is to replace locally the uncomputable $\tilde{\rho}^i_{J,\mathrm{alg}} = u_J - u^i_J$ by the constructed local contributions $\rho^i_{j,\mathbf{a}}$ using the problems they solve on patches. We begin by using Proposition 7.6 applied to $\tilde{\rho}^i_{J,\mathrm{alg}} \in V^p_J$, writing $\tilde{\rho}^i_{J,\mathrm{alg}} = e_0 + \sum_{j=1}^J\sum_{\mathbf{b}\in\mathcal{V}_j} e_{j,\mathbf{b}}$. Then

$$\begin{aligned}
\|\nabla\tilde{\rho}^i_{J,\mathrm{alg}}\|^2 &= \Big(\nabla\tilde{\rho}^i_{J,\mathrm{alg}}, \nabla e_0 + \sum_{j=1}^{J}\sum_{\mathbf{b}\in\mathcal{V}_j}\nabla e_{j,\mathbf{b}}\Big)\\
&\overset{(1.16)}{=} \Big(\nabla\rho^i_0,\nabla e_0\Big) + \sum_{j=1}^{J}\sum_{\mathbf{a}\in\mathcal{V}_{j-s}}\Big(\nabla\tilde{\rho}^i_{J,\mathrm{alg}}, \sum_{\mathbf{b}\in\mathcal{I}_{\mathbf{a}}}\nabla e_{j,\mathbf{b}}\Big)_{\omega^{\mathbf{a}}_{j,s}}\\
&\overset{(1.13)}{=} \Big(\nabla\rho^i_0,\nabla e_0\Big) + \sum_{j=1}^{J}\sum_{\mathbf{a}\in\mathcal{V}_{j-s}}\left(\Big(f, \sum_{\mathbf{b}\in\mathcal{I}_{\mathbf{a}}} e_{j,\mathbf{b}}\Big)_{\omega^{\mathbf{a}}_{j,s}} - \Big(\nabla u_J, \sum_{\mathbf{b}\in\mathcal{I}_{\mathbf{a}}}\nabla e_{j,\mathbf{b}}\Big)_{\omega^{\mathbf{a}}_{j,s}}\right)\\
&\overset{(1.20)}{=} \Big(\nabla\rho^i_0,\nabla e_0\Big) + \sum_{j=1}^{J}\sum_{\mathbf{a}\in\mathcal{V}_{j-s}}\left(\Big(\nabla\rho^i_{j,\mathbf{a}}, \sum_{\mathbf{b}\in\mathcal{I}_{\mathbf{a}}}\nabla e_{j,\mathbf{b}}\Big)_{\omega^{\mathbf{a}}_{j,s}} + \frac{1}{w_2}\sum_{k=0}^{j-1}\Big(\nabla\rho^i_k, \sum_{\mathbf{b}\in\mathcal{I}_{\mathbf{a}}}\nabla e_{j,\mathbf{b}}\Big)_{\omega^{\mathbf{a}}_{j,s}}\right)\\
&= \Big(\nabla\rho^i_0,\nabla e_0\Big) + \sum_{j=1}^{J}\sum_{\mathbf{a}\in\mathcal{V}_{j-s}}\Big(\nabla\rho^i_{j,\mathbf{a}}, \sum_{\mathbf{b}\in\mathcal{I}_{\mathbf{a}}}\nabla e_{j,\mathbf{b}}\Big)_{\omega^{\mathbf{a}}_{j,s}} + \frac{1}{w_2}\sum_{j=1}^{J}\sum_{k=0}^{j-1}\Big(\nabla\rho^i_k, \sum_{\mathbf{b}\in\mathcal{V}_j}\nabla e_{j,\mathbf{b}}\Big).
\end{aligned}$$

We will now estimate each of the above three terms using Young's inequality and patch overlap arguments as done in the proof of Lemma 7.1. First, we have

$$\Big(\nabla\rho^i_0,\nabla e_0\Big) \le \frac{C^2_{\mathrm{SMD}}}{2}\|\nabla\rho^i_0\|^2 + \frac{1}{2C^2_{\mathrm{SMD}}}\|\nabla e_0\|^2.$$

For the second term, we similarly obtain

$$\begin{aligned}
&\sum_{j=1}^{J}\sum_{\mathbf{a}\in\mathcal{V}_{j-s}}\Big(\nabla\rho^i_{j,\mathbf{a}}, \sum_{\mathbf{b}\in\mathcal{I}_{\mathbf{a}}}\nabla e_{j,\mathbf{b}}\Big)_{\omega^{\mathbf{a}}_{j,s}}\\
&\le \sum_{j=1}^{J}\sum_{\mathbf{a}\in\mathcal{V}_{j-s}}\left(\frac{2(d+1)C^2_{\mathrm{SMD}}}{2}\|\nabla\rho^i_{j,\mathbf{a}}\|^2_{\omega^{\mathbf{a}}_{j,s}} + \frac{\Big\|\sum_{\mathbf{b}\in\mathcal{I}_{\mathbf{a}}}\nabla e_{j,\mathbf{b}}\Big\|^2_{\omega^{\mathbf{a}}_{j,s}}}{2(2(d+1)C^2_{\mathrm{SMD}})}\right)\\
&\le (d+1)C^2_{\mathrm{SMD}}\sum_{j=1}^{J}\sum_{\mathbf{a}\in\mathcal{V}_{j-s}}\|\nabla\rho^i_{j,\mathbf{a}}\|^2_{\omega^{\mathbf{a}}_{j,s}} + \frac{1}{4C^2_{\mathrm{SMD}}}\sum_{j=1}^{J}\sum_{\mathbf{b}\in\mathcal{V}_j}\|\nabla e_{j,\mathbf{b}}\|^2_{\omega^{\mathbf{b}}_{j,0}}.
\end{aligned}$$

Finally, for the third term we additionally use the property $w_2 \geq 1$, and rename summation indices when there is no confusion

$$\begin{aligned}
&\frac{1}{w_2}\sum_{j=1}^{J}\sum_{k=0}^{j-1}\Big(\nabla\rho_k^i, \sum_{\mathbf{b}\in\mathcal{V}_j}\nabla e_{j,\mathbf{b}}\Big)\\
&\leq \frac{2(d+1)C_{\mathrm{SMD}}^2 J}{2w_2}\sum_{j=1}^{J}\sum_{k=0}^{j-1}\|\nabla\rho_k^i\|^2 + \frac{1}{2w_2(2(d+1)C_{\mathrm{SMD}}^2 J)}\sum_{j=1}^{J}\sum_{k=0}^{j-1}\Big\|\sum_{\mathbf{b}\in\mathcal{V}_j}\nabla e_{j,\mathbf{b}}\Big\|^2\\
&\leq \frac{(d+1)C_{\mathrm{SMD}}^2 J^2}{w_2}\sum_{k=0}^{J}\|\nabla\rho_k^i\|^2 + \frac{1}{4C_{\mathrm{SMD}}^2}\sum_{j=1}^{J}\sum_{\mathbf{b}\in\mathcal{V}_j}\|\nabla e_{j,\mathbf{b}}\|^2_{\omega_{j,0}^{\mathbf{b}}}\\
&\overset{(1.39)}{\leq} \frac{(d+1)C_{\mathrm{SMD}}^2 J^2}{w_2}\Big(\|\nabla\rho_0^i\|^2 + \frac{d+1}{w_1^2}\sum_{j=1}^{J}\sum_{\mathbf{a}\in\mathcal{V}_{j-s}}\|\nabla\rho_{j,\mathbf{a}}^i\|^2_{\omega_{j,s}^{\mathbf{a}}}\Big)\\
&\qquad + \frac{1}{4C_{\mathrm{SMD}}^2}\sum_{j=1}^{J}\sum_{\mathbf{b}\in\mathcal{V}_j}\|\nabla e_{j,\mathbf{b}}\|^2_{\omega_{j,0}^{\mathbf{b}}}.
\end{aligned}$$

Summing these components together, we can now pursue our main estimate:

$$\begin{aligned}
\|\nabla\tilde{\rho}_{J,\mathrm{alg}}^i\|^2 &\overset{(1.57)}{\leq} \frac{C_{\max}^2(w_1,w_2)}{2}\Big(\|\nabla\rho_0^i\|^2 + \sum_{j=1}^{J}\sum_{\mathbf{a}\in\mathcal{V}_{j-s}}\|\nabla\rho_{j,\mathbf{a}}^i\|^2_{\omega_{j,s}^{\mathbf{a}}}\Big)\\
&\qquad + \frac{\|\nabla e_0\|^2 + \sum_{j=1}^{J}\sum_{\mathbf{b}\in\mathcal{V}_j}\|\nabla e_{j,\mathbf{b}}\|^2_{\omega_{j,0}^{\mathbf{b}}}}{2C_{\mathrm{SMD}}^2}\\
&\overset{(1.55)}{\leq} \frac{C_{\max}^2(w_1,w_2)}{2}\Big(\|\nabla\rho_0^i\|^2 + \sum_{j=1}^{J}\sum_{\mathbf{a}\in\mathcal{V}_{j-s}}\|\nabla\rho_{j,\mathbf{a}}^i\|^2_{\omega_{j,s}^{\mathbf{a}}}\Big) + \frac{1}{2}\|\nabla\tilde{\rho}_{J,\mathrm{alg}}^i\|^2.
\end{aligned}$$

After subtracting $\frac{1}{2}\|\nabla\tilde{\rho}_{J,\mathrm{alg}}^i\|^2$ on both sides, we finally obtain the desired result.

The lower bound on $C_{\max}^2(w_1,w_2)$ in (1.57) is obtained by using $C_{\mathrm{SMD}} \geq 1$ from Proposition 7.6. To derive the upper bound, we use the fact that weights of Definition 3.2 satisfy $w_1 \geq 1$, $w_2 \geq 1$. This gives $\frac{J^2}{w_2} \leq J^2$ and $\frac{d+1}{w_1^2} \leq d+1$, leading to the desired result:

$$C_{\max}^2(w_1,w_2) \leq 2(d+1)C_{\mathrm{SMD}}^2\left(1 + J^2(d+1)\right) \leq 2(d+1)C_{\mathrm{SMD}}^2\left(2J^2(d+1)\right).$$

□

7.5 Proof of Theorem 5.1

The results of the previous subsections allow us now to give a concise proof of Theorem 5.1.

Proof of Theorem 5.1. Case $\rho_{J,\mathrm{alg}}^i = 0$. By Definition 4.1 this means $\eta_{\mathrm{alg}}^i = 0$, so that it suffices to show that $u_J = u_J^i$ in this case. We do this by using Lemmas 7.2 and 7.7, which lead to

$$\|\nabla(u_J - u_J^i)\|^2 \overset{(1.14)}{=} \|\nabla\tilde{\rho}_{J,\mathrm{alg}}^i\|^2 \overset{(1.56)}{\leq} C_{\max}^2(w_1,w_2)\left(\|\nabla\rho_0^i\|^2 + \sum_{j=1}^{J}\sum_{\mathbf{a}\in\mathcal{V}_{j-s}}\|\nabla\rho_{j,\mathbf{a}}^i\|^2_{\omega_{j,s}^{\mathbf{a}}}\right)$$

$$\overset{(1.43)}{\leq} \frac{C_{\max}^2(w_1,w_2)}{C_{\min}^2(w_1,w_2)}\Big((f,\rho^i_{J,\mathrm{alg}}) - (\nabla u^i_J, \nabla \rho^i_{J,\mathrm{alg}})\Big) = 0. \tag{1.58}$$

Case $\rho^i_{J,\mathrm{alg}} \neq 0$. In this case, we combine the results of Lemmas 7.1, 7.2, and 7.7

$$\eta^i_{\mathrm{alg}} = \frac{(f,\rho^i_{J,\mathrm{alg}}) - (\nabla u^i_J, \nabla \rho^i_{J,\mathrm{alg}})}{\|\nabla \rho^i_{J,\mathrm{alg}}\|} \overset{\substack{(1.40)\\(1.45)}}{\geq} \frac{C_{\min}^2(w_1,w_2)}{C_{\max}(w_1)}\left(\|\nabla\rho^i_0\|^2 + \sum_{j=1}^{J}\sum_{\mathbf{a}\in\mathcal{V}_{j-s}} \|\nabla\rho^i_{j,\mathbf{a}}\|^2_{\omega^{\mathbf{a}}_{j,s}}\right)^{\frac{1}{2}}$$
$$\overset{(1.56)}{\geq} \frac{C_{\min}^2(w_1,w_2)}{C_{\max}(w_1)C_{\max}(w_1,w_2)}\|\nabla\hat{\rho}^i_{J,\mathrm{alg}}\| \overset{(1.14)}{=} \beta\|\nabla(u_J - u^i_J)\| \tag{1.59}$$

for

$$\frac{1}{12\sqrt{2}C_{\mathrm{SMD}}J^{\frac{5}{2}}(d+1)^{\frac{5}{2}}} \leq \beta := \frac{C_{\min}^2(w_1,w_2)}{C_{\max}(w_1)C_{\max}(w_1,w_2)} \leq \frac{1}{8\sqrt{d+1}}.$$

The bounds on β follow from (1.41), (1.44), and (1.57). □

Example 7.8 (Specific choices of weights)**.** *We illustrate here a bound on the efficiency factor β in* (1.28) *for different choices of the damping weights satisfying the compatibility condition* (1.17) *from Remark 3.4:*

$$w_1 = J(d+1) \text{ and } w_2 = 1: \qquad \frac{1}{12C_{\mathrm{SMD}}J^2\sqrt{2(d+1)^3}} \leq \beta.$$
$$w_1 = d+1 \text{ and } w_2 = J: \qquad \frac{1}{12C_{\mathrm{SMD}}J\sqrt{2(d+1)^3}} \leq \beta.$$
$$w_1 = w_2 = \sqrt{J(d+1)}: \qquad \frac{1}{12\sqrt{2}C_{\mathrm{SMD}}J^{\frac{5}{4}}(d+1)} \leq \beta.$$
$$w_1 = 1 \text{ and } w_2 = \infty: \qquad \frac{1}{8C_{\mathrm{SMD}}\sqrt{J}(d+1)} \leq \beta.$$
$$w_1 = 4\sqrt{J} \text{ and } w_2 = \infty: \qquad \frac{1}{8C_{\mathrm{SMD}}\sqrt{J(d+1)}} \leq \beta.$$

7.6 Proof of Corollary 5.6

If $\rho^i_{J,\mathrm{alg}} = 0$, as a result of (1.58), we have $\|\nabla(u_J - u^i_J)\| = \|\nabla\rho^i_0\|^2 + \sum_{j=1}^{J}\sum_{\mathbf{a}\in\mathcal{V}_{j-s}}\|\nabla\rho^i_{j,\mathbf{a}}\|^2_{\omega^{\mathbf{a}}_{j,s}} = 0$. Otherwise by (1.56) we set $C_1 := C_{\max}(w_1,w_2)$ and by (1.23) and (1.59), $C_2 = \dfrac{1}{\beta} = \dfrac{C_{\max}(w_1)C_{\max}(w_1,w_2)}{C_{\min}^2(w_1,w_2)}$ gives the result.

8 Conclusions and outlook

In this work, we presented a hierarchical construction of the algebraic residual lifting in the spirit of Papež et al. [2020]. This lifting approximates the algebraic error by one iteration of a V-cycle multigrid with no pre-smoothing step, a single damped additive Schwarz post-smoothing step, and a coarse solve of the lowest polynomial degree. The lifting leads us to an a posteriori estimator on the algebraic

error and to a linear iterative solver. We showed that the two following results are equivalent: the (reliable) a posteriori estimator is p-robustly efficient, and the solver contracts p-robustly the error at each iteration. The provided numerical tests agree with these theoretical findings. Moreover, we also presented numerical results for a modified solver corresponding to a weighted restricted additive Schwarz smoothing. In accordance with the literature, this modified solver provides a further speed-up compared to the damped Schwarz smoothing. Although we currently cannot show that our p-robust theoretical result also applies to this construction, the use of high-degree polynomials does not seem to cause a degradation of the solver. So far, our theory involves estimates depending algebraically on the number of mesh levels J, which we do not observe in the numerical results for the weighted restricted variant. In forthcoming works, we plan to develop adaptivity based on the property (1.31), i.e., a computable splitting equivalent to the error and localized not only levelwise but also patchwise. Applications to more involved problems are also on our work list.

Chapter 2

A-posteriori-steered p-robust multigrid with optimal step-sizes and adaptive number of smoothing steps

We present in this chapter the results of the article Miraçi et al. [2021a], *SIAM Journal on Scientific Computing*, DOI 10.1137/20M1349503, written with Jan Papež and Martin Vohralík.

Contents

1 Introduction **57**

2 Setting **59**

2.1 Model problem, finite element discretization, and algebraic system 60

2.2 A hierarchy of meshes and spaces 60

3 Motivation: level-wise orthogonal decomposition of the error **61**

4 Multilevel solver **62**

5 A posteriori estimator on the algebraic error **64**

6 Main results **65**

6.1 Setting, mesh, and regularity assumptions 65

6.2 Main results 66

6.3 Additional results 67

7 Adaptive number of smoothing steps **68**

8 Complexity of the solver **69**

9 Numerical experiments **69**

9.1 Performance of the multilevel solver of Definition 4.1 70

9.2 Adaptive number of smoothing steps using Definition 7.1 . . . 71

9.3 Examples in three space dimensions 74

9.4 Comparison with solvers from literature 74

10 Proof of Theorem 6.6 . . . . . **77**

10.1 Properties of the estimator η^i_{alg} 77

10.2 Properties of the exact residual lifting $\tilde{\rho}^i_{J,\mathrm{alg}}$ 77
10.3 Proof of Theorem 6.6 under the minimal $H^1_0(\Omega)$-regularity assumption 78
10.4 Proof of Theorem 6.6 under the $H^2(\Omega)$-regularity assumption . 80
11 Conclusions and future work 85

Abstract

We develop a multigrid solver steered by an a posteriori estimator of the algebraic error. We adopt the context of a second-order elliptic diffusion problem discretized by conforming finite elements of arbitrary polynomial degree $p \geq 1$. Our solver employs zero pre- and one post-smoothing by the overlapping Schwarz (block-Jacobi) method and features an optimal choice of the step-sizes in the smoothing correction on each level by line search. This leads to a simple Pythagorean formula of the algebraic error in the next step in terms of the current error and level-wise and patch-wise error reductions. We show the two following results and their equivalence: the solver contracts the algebraic error independently of the polynomial degree p; and the estimator represents a two-sided p-robust bound on the algebraic error. The p-robustness results are obtained by carefully applying the results of Schöberl *et al.* [IMA J. Numer. Anal., 28 (2008), pp. 1–24] for one mesh, combined with a multilevel stable decomposition for piecewise affine polynomials of Xu *et al.* [Multiscale, nonlinear and adaptive approximation, Springer, Berlin, 2009, pp. 599–659]. We consider quasi-uniform or graded bisection simplicial meshes and prove mild dependence on the number of mesh levels for minimal H^1-regularity and complete independence for H^2-regularity. We also present a simple and effective way for the solver to adaptively choose the number of post-smoothing steps necessary at each individual level, yielding a yet improved error reduction. Numerical tests confirm p-robustness and show the benefits of the adaptive number of smoothing steps.

1 Introduction

Multilevel (multigrid) methods have shown their versatility as solvers and/or preconditioners of large sparse algebraic linear systems arising from numerical discretizations of partial differential equations. We refer to pioneering works such as Brandt et al. [1985], Bramble et al. [1986], Bank et al. [1988], Ruge and Stüben [1987], or Oswald [1994], as well as to survey works that thoroughly treat subspace correction methods in Xu [1992], robust multigrid methods with respect to non-smooth coefficients in Chan and Wan [2000], multigrid solvers for high-order discretizations in Sundar et al. [2015], and the references therein.

In this work, we develop a *multilevel solver* for algebraic linear systems arising from the discretization using conforming finite elements of arbitrary polynomial degree $p \geq 1$. One iteration of our solver can be seen as a V-cycle employing *zero pre-* and *one post-smoothing step*, where the level-wise smoother is *overlapping additive Schwarz (block-Jacobi)* associated to the patches of elements sharing a common

vertex. A crucial difference to the classical V-cycle is that on each level, we use an *optimal step-size* at the error correction stage, yielding minimal algebraic error in the subsequent level.

The idea of an optimal step-size in the error correction is not new; in fact, a weighting of multigrid error corrections concept appears as early as in Brandt [1977]. Then, this approach is used, e.g., in Canuto and Quarteroni [1985], though not in a multigrid setting. The interest of an optimally-weighted error correction in the context of multigrid has been also pointed out in Heinrichs [1988], where this choice resulted in a better numerical performance of the solver. Another version of multigrid solvers with a changing step-size error correction can be found in the form of a scaled residual in Rüde [1993]. A crucial immediate consequence of our present optimal step-sizes choice is that the error contraction becomes explicitly known. This allows to obtain the following Pythagorean formula representing the *error decrease* from step i to step $i+1$:

$$\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J-u_J^{i+1})\right\|^2=\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J-u_J^{i})\right\|^2-\sum_{j=0}^{J}\left(\lambda_j^i\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_j^i\right\|\right)^2. \tag{2.1}$$

Here, $\boldsymbol{\mathcal{K}}$ is the diffusion tensor, $j\in\{0,\dots,J\}$ is the level counter, u_J is the (unknown) exact algebraic solution, u_J^i denotes the available iterate, u_J^{i+1} is the next iterate, ρ_j^i are the computed level-wise smoothing corrections, and λ_j^i are the level-wise optimal step-sizes.

A salient feature of formula (2.1) is that the computable level-wise terms $\left\{\sum_{j=0}^{J}\left(\lambda_j^i\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_j^i\right\|\right)^2\right\}^{\frac{1}{2}}$ form an a posteriori estimator η_{alg}^i, representing a guaranteed lower bound for the algebraic error $\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J-u_J^i)\right\|$. Thus our solver is actually driven by the information provided by the estimator, making the solver an *a-posteriori-steered* multigrid.

Our main results can be summarized as follows. First, we prove that our multilevel solver *contracts* the error in each iteration. Second, we show that the associated a posteriori estimator η_{alg}^i is *efficient* in that it also represents an upper bound of the error (up to a constant). These two claims are actually equivalent. Third, there holds

$$\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J-u_J^i)\right\|^2\approx\underbrace{\sum_{j=0}^{J}\left(\lambda_j^i\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_j^i\right\|\right)^2}_{\left(\eta_{\mathrm{alg}}^i\right)^2}=\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_0^i\right\|^2+\sum_{j=1}^{J}\lambda_j^i\sum_{\mathbf{a}\in\mathcal{V}_j}\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_{j,\mathbf{a}}^i\right\|_{\omega_j^{\mathbf{a}}}^2,$$

so that the developed a posteriori error estimator actually localizes the algebraic error with respect to mesh levels and also with respect to patches of elements on each level. These results hold for quasi-uniform meshes as well as possibly highly graded ones. Importantly, all the results hold *p-robustly*, i.e. are robust with respect to the polynomial degree p.

Notable previous works in treating p-robustness include Quarteroni and Sacchi Landriani [1988] for a specific domain configuration and Pavarino [1994] for quadrilateral/hexahedral meshes, where the author introduced a p-robust additive Schwarz method. Later, Janssen and Kanschat [2011] and Lucero Lorca and Kanschat [2021] used multilevel preconditioners for rectangular/hexahedral meshes, and

Antonietti and Pennesi [2019] considered more general meshes. Therein, however, more smoothing steps are generally necessary, whereas, we recall, we only rely on a single post-smoothing step. A p-robust stable decompostion on triangular/tetrahedral meshes was presented in Schöberl et al. [2008]. It leads to a (one-mesh) p-robust preconditioner and plays an important part in the analysis of our work.

Compared to our previous work Miraçi et al. [2020], see Chapter 1, we can mention the following improvements: 1) In the solver of Miraçi et al. [2020], a global optimal step-size was used, whereas we use here the level-wise step-sizes. 2) We obtain here the powerful error decrease formula (2.1). 3) The solver proposed in this work does not need any damping, where tuning of the parameters can be cumbersome. 4) The current analysis gives at most linear dependence on the number of mesh levels J under minimal H^1-regularity. 5) The current analysis gives complete independence of J in H^2-regularity setting.

Formula (2.1) is also the foundation of a simple and efficient *adaptive strategy* for the choice of the number of post-smoothing steps per level. The essence and particularity of our strategy relies on *a-posteriori-steered decision-making* of the number of smoothing steps. Following (2.1), after one mandatory smoothing step at each level, if the given decrease $\lambda_j^i \left\| \boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \rho_j^i \right\|$ is higher than a user-prescribed portion of the decrease made by the previous levels, we decide to do another smoothing step before going to the next level. The idea of employing a *variable* number of smoothing steps per level has also been explored e.g. in Bramble and Pasciak [1987], where a generalized V-cycle uses more smoothing steps on coarser grids and fewer on finer ones. This decision is however taken a priori. Closely related to the subject is also the work of Thekale et al. [2010], who suggest a variable number of multigrid cycles per level which optimizes the costs of the full multigrid method by formulating a nonlinear integer programming problem of small enough size to be solved exactly.

This manuscript is organized as follows. In Section 2, we present the multilevel setting and notation we will be working with, and Section 3 develops the motivation leading us to consider our particular multilevel solver. The solver is then presented in Section 4, and the a posteriori error estimator is introduced in Section 5. We collect in Section 6 the main results of the manuscript. In Section 7, we present the solver with the adaptive choice of number of post-smoothing steps, Section 8 presents a simplified cost analysis, and Section 9 collects the results of numerical experiments, which additionally show numerical robustness of our solver with respect to the jumps of the diffusion tensor for uniform mesh refinements. The proofs of our main results are given in Section 10, and we present our concluding remarks in Section 11.

2 Setting

This section presents the model problem and the multilevel setting with which we will be working.

2.1 Model problem, finite element discretization, and algebraic system

We consider a second-order elliptic diffusion problem defined over $\Omega \subset \mathbb{R}^d$, $d \in \{1,2,3\}$, an open bounded polytope with a Lipschitz-continuous boundary. Let $f \in L^2(\Omega)$ be a source term and $\boldsymbol{\mathcal{K}} \in [L^\infty(\Omega)]^{d\times d}$ a symmetric positive definite diffusion coefficient. The weak solution $u \in H^1_0(\Omega)$ is given by

$$(\boldsymbol{\mathcal{K}}\nabla u, \nabla v) = (f, v) \quad \forall v \in H^1_0(\Omega), \tag{2.2}$$

where $(\cdot,\cdot)$ is the $L^2(\Omega)$ or $[L^2(\Omega)]^d$ scalar product.

We discretize the continuous problem (2.2) by fixing $\mathcal{T}_J$, a matching simplicial mesh of Ω, and an integer $p \geq 1$, in order to introduce the finite element space of continuous piecewise p-degree polynomials

$$V^p_J := \mathbb{P}_p(\mathcal{T}_J) \cap H^1_0(\Omega), \tag{2.3}$$

where $\mathbb{P}_p(\mathcal{T}_J) := \{v_J \in L^2(\Omega), v_J|_K \in \mathbb{P}_p(K)\ \forall K \in \mathcal{T}_J\}$. The discrete problem now consists in finding $u_J \in V^p_J$ such that

$$(\boldsymbol{\mathcal{K}}\nabla u_J, \nabla v_J) = (f, v_J) \quad \forall v_J \in V^p_J. \tag{2.4}$$

If one introduces a basis of V^p_J, then the discrete problem is equivalent to solving a system of linear algebraic equations whose matrix is symmetric and positive definite. However, such a linear system depends on the choice of the basis functions. To avoid this dependence, we work instead with a functional description of the problem. In particular, we define the algebraic residual functional on V^p_J, for any $u^i_J \in V^p_J$, by

$$v_J \mapsto (f, v_J) - (\boldsymbol{\mathcal{K}}\nabla u^i_J, \nabla v_J) \in \mathbb{R}, \quad v_J \in V^p_J. \tag{2.5}$$

2.2 A hierarchy of meshes and spaces

We work with a hierarchy of matching simplicial meshes $\{\mathcal{T}_j\}_{0\leq j\leq J}$, $J \geq 1$, where $\mathcal{T}_J$ has been introduced above, and where $\mathcal{T}_j$ is a refinement of $\mathcal{T}_{j-1}$, $1 \leq j \leq J$. We also introduce a hierarchy of finite element spaces associated to the mesh hierarchy. For this purpose, for $j \in \{0,\dots,J\}$, fix p_j, the polynomial degree associated to mesh level j such that $1 = p_0 \leq p_1 \leq \ldots \leq p_{J-1} \leq p_J = p$. In particular, let:

$$\text{for } j = 0: \qquad V^1_0 := \mathbb{P}_1(\mathcal{T}_0) \cap H^1_0(\Omega) \quad \text{(lowest-order space)}, \tag{2.6a}$$

$$\text{for } 1 \leq j \leq J: \qquad V^{p_j}_j := \mathbb{P}_{p_j}(\mathcal{T}_j) \cap H^1_0(\Omega) \quad (p_j\text{-th order spaces}), \tag{2.6b}$$

where $\mathbb{P}_{p_j}(\mathcal{T}_j) := \{v_j \in L^2(\Omega),\ v_j|_K \in \mathbb{P}_{p_j}(K)\ \ \forall K \in \mathcal{T}_j\}$. Note that $V^1_0 \subset V^{p_1}_1 \subset \ldots \subset V^{p_{J-1}}_{J-1} \subset V^{p_J}_J = V^p_J$.

Let $\mathcal{V}_j$, $0 \leq j \leq J$, be the set of vertices of the mesh $\mathcal{T}_j$. For the following, we need to define the notion of patches of elements, illustrated in Figure 2.1. Given a vertex $\mathbf{a} \in \mathcal{V}_j$, we denote by $\mathcal{T}^{\mathbf{a}}_j$ all the mesh elements of $\mathcal{T}_j$ that share the vertex $\mathbf{a}$, $\mathcal{T}^{\mathbf{a}}_j := \{K \in \mathcal{T}_j, \mathbf{a} \in \mathcal{V}_K\}$, where $\mathcal{V}_K$ is the set of vertices of an element K. The corresponding open patch subdomain is denoted by $\omega^{\mathbf{a}}_j$. We also denote by $\psi_{j,\mathbf{a}}$ the standard hat function associated to the vertex $\mathbf{a} \in \mathcal{V}_j$, i.e., the piecewise affine

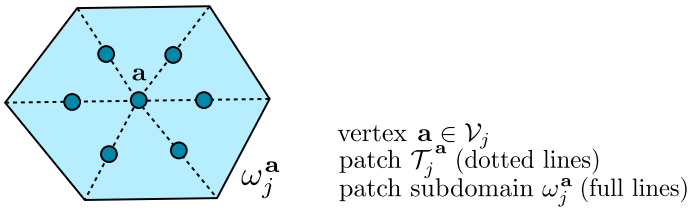


Figure 2.1: Illustration of degrees of freedom ($p_j = 2$) for the space $V_j^{\mathbf{a}}$ associated to the patch $\mathcal{T}_j^{\mathbf{a}}$.

function with respect to $\mathcal{T}_j$ taking value 1 at vertex $\mathbf{a}$ and vanishing in all other vertices of $\mathcal{V}_j$. Note that $\omega_j^{\mathbf{a}}$ is the support of $\psi_{j,\mathbf{a}}$. Finally the local spaces $V_j^{\mathbf{a}}$ are defined by

$$V_j^{\mathbf{a}} := \mathbb{P}_{p_j}(\mathcal{T}_j) \cap H_0^1(\omega_j^{\mathbf{a}}), \tag{2.7}$$

cf. Figure 2.1 for the illustration of degrees of freedom when $p_j = 2$.

3 Motivation: level-wise orthogonal decomposition of the error

It is known that a multilevel construction is required to capture correctly the behavior of the algebraic error, cf., e.g., Rüde [1993], or the counterexample of [Papež et al., 2020, Section 2.1]. Consider, for a given $u_J^i \in V_J^p$, the following (costly for practice but illustrative) hierarchical construction $\tilde{\rho}_{J,\mathrm{alg}}^i \in V_J^p$

$$\tilde{\rho}_{J,\mathrm{alg}}^i := \rho_0^i + \sum_{j=1}^{J} \tilde{\rho}_j^i; \tag{2.8}$$

here, $\rho_0^i = \tilde{\rho}_0^i \in V_0^1$ is the solution to a global lowest-order residual problem on the coarsest mesh

$$(\boldsymbol{\mathcal{K}}\nabla\rho_0^i, \nabla v_0) = (f, v_0) - (\boldsymbol{\mathcal{K}}\nabla u_J^i, \nabla v_0) \quad \forall v_0 \in V_0^1, \tag{2.9}$$

and, moreover, for $j = 1 : J$, $\tilde{\rho}_j^i \in V_j^{p_j}$ are the solutions of

$$(\boldsymbol{\mathcal{K}}\nabla\tilde{\rho}_j^i, \nabla v_j) = (f, v_j) - (\boldsymbol{\mathcal{K}}\nabla u_J^i, \nabla v_j) - \sum_{k=0}^{j-1}(\boldsymbol{\mathcal{K}}\nabla\tilde{\rho}_k^i, \nabla v_j) \quad \forall v_j \in V_j^{p_j}. \tag{2.10}$$

This construction returns the algebraic error, i.e., $\tilde{\rho}_{J,\mathrm{alg}}^i = u_J - u_J^i$, or, equivalently,

$$u_J = u_J^i + \sum_{j=0}^{J} \tilde{\rho}_j^i. \tag{2.11}$$

This, in turn, means that $\tilde{\rho}_{J,\mathrm{alg}}^i$ satisfies

$$(\boldsymbol{\mathcal{K}}\nabla\tilde{\rho}_{J,\mathrm{alg}}^i, \nabla v_J) = (f, v_J) - (\boldsymbol{\mathcal{K}}\nabla u_J^i, \nabla v_J) \quad \forall v_J \in V_J^p. \tag{2.12}$$

Moreover, there holds $(\boldsymbol{\mathcal{K}}\nabla\tilde{\rho}_j^i, \nabla\tilde{\rho}_k^i) = 0$, for $0 \le k, j \le J$, $j \ne k$. These observations altogether lead to the orthogonal decomposition of the error between u_J^i and u_J as

$$\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J - u_J^i)\right\|^2 = \left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\tilde{\rho}_{J,\mathrm{alg}}^i\right\|^2 = \sum_{j=0}^{J}\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\tilde{\rho}_j^i\right\|^2. \tag{2.13}$$

4 Multilevel solver

We introduce now our local constructions inspired by (2.8)–(2.10), producing level-wise approximations of the algebraic error components $\tilde{\rho}_j^i$ of (2.10). The construction relies on the inexpensive coarse residual solve (2.9) and on local contributions, defined on patches of elements on each level, see Figure 2.1. We go through the levels adding gradually level-wise updates $u_{J,j}^i$ to the current approximation u_J^i as described below. Hereafter, $(\cdot,\cdot)_{\omega_j^{\mathbf{a}}}$ stands for the $L^2(\omega_j^{\mathbf{a}})$ or $[L^2(\omega_j^{\mathbf{a}})]^d$ scalar product.

Definition 4.1 (Multilevel solver). *1. Initialize $u_J^0 \in V_J^p$ as the zero function and set $i := 0$.*

2. Perform the following steps (a)–(d)*:*

(a) Define ρ_0^i by (2.9), impose $\lambda_0^i := 1$, and set

$$u_{J,0}^i := u_J^i + \lambda_0^i \rho_0^i.$$

(b) For $j = 1 : J$, define the local contributions $\rho_{j,\mathbf{a}}^i \in V_j^{\mathbf{a}}$ as solutions of patch problems, for all vertices $\mathbf{a} \in \mathcal{V}_j$,

$$(\boldsymbol{\mathcal{K}}\nabla\rho_{j,\mathbf{a}}^i, \nabla v_{j,\mathbf{a}})_{\omega_j^{\mathbf{a}}} = (f, v_{j,\mathbf{a}})_{\omega_j^{\mathbf{a}}} - (\boldsymbol{\mathcal{K}}\nabla u_{J,j-1}^i, \nabla v_{j,\mathbf{a}})_{\omega_j^{\mathbf{a}}} \quad \forall v_{j,\mathbf{a}} \in V_j^{\mathbf{a}}, \tag{2.14}$$

and the descent direction $\rho_j^i \in V_j^{p_j}$ on the level j by

$$\rho_j^i := \sum_{\mathbf{a}\in\mathcal{V}_j} \rho_{j,\mathbf{a}}^i. \tag{2.15}$$

If $\rho_j^i \neq 0$, define the optimal step-size on level j

$$\lambda_j^i := \frac{(f, \rho_j^i) - (\boldsymbol{\mathcal{K}}\nabla u_{J,j-1}^i, \nabla\rho_j^i)}{\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_j^i\right\|^2}, \tag{2.16}$$

otherwise set $\lambda_j^i := 1$. Define the level update by

$$u_{J,j}^i := u_{J,j-1}^i + \lambda_j^i \rho_j^i. \tag{2.17}$$

(c) Set the final update as $u_J^{i+1} := u_{J,J}^i \in V_J^p$.

(d) If $u_J^{i+1} = u_J^i$, then stop the solver. Otherwise increase $i := i+1$ and go to step 2(a)*.*

Note that by definition $\lambda_0^i = 1$ and we thus have for $\rho_0^i \neq 0$,

$$\frac{(f, \rho_0^i) - (\boldsymbol{\mathcal{K}}\nabla u_J^i, \nabla\rho_0^i)}{\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_0^i\right\|^2} \overset{(2.9)}{=} 1 = \lambda_0^i.$$

Remark 4.2 (Compact writing of the iteration update). *Let $u_J^i \in V_J^p$. It is easily noted that the level update (2.17) equivalently writes as*

$$u_{J,j}^i = u_J^i + \sum_{k=0}^{j} \lambda_k^i \rho_k^i. \tag{2.18}$$

Thus, using the conventions $u^i_{J,-1} := u^i_J$ *and* $\frac{0}{0} = 0$*, the new iterate after one step of the solver described in Definition 4.1 is, compare to* (2.11)*,*

$$u^{i+1}_J = u^i_J + \sum_{j=0}^{J} \lambda^i_j \rho^i_j = u^i_J + \sum_{j=0}^{J} \frac{(f, \rho^i_j) - (\boldsymbol{\mathcal{K}} \nabla u^i_{J,j-1}, \nabla \rho^i_j)}{\left\| \boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \rho^i_j \right\|^2} \rho^i_j. \tag{2.19}$$

The lemma below justifies rigorously the choice and use of the step-sizes (2.16).

Lemma 4.3 (Level-wise optimal step-sizes)**.** *Let* $u^i_{J,j-1} \in V^p_J$ *be arbitrary, let* $j \in \{1, \dots, J\}$*, and* ρ^i_j *and* λ^i_j *be given by* (2.15) *and* (2.16)*, respectively. Then*

$$\lambda^i_j = \arg\min_{\lambda \in \mathbb{R}} \left\| \boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \big(u_J - (u^i_{J,j-1} + \lambda \rho^i_j) \big) \right\|.$$

Proof. We write the algebraic error of $u^i_{J,j-1} + \lambda \rho^i_j$ as a function of λ

$$\begin{aligned} \left\| \boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \big(u_J - (u^i_{J,j-1} + \lambda \rho^i_j) \big) \right\|^2 &= \left\| \boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla (u_J - u^i_{J,j-1}) \right\|^2 \\ &\quad - 2\lambda \big(\boldsymbol{\mathcal{K}} \nabla (u_J - u^i_{J,j-1}), \nabla \rho^i_j \big) + \lambda^2 \left\| \boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \rho^i_j \right\|^2. \end{aligned} \tag{2.20}$$

We realize that this function has a minimum, as given by (2.16), at

$$\lambda^i_j = \frac{(\boldsymbol{\mathcal{K}} \nabla (u_J - u^i_{J,j-1}), \nabla \rho^i_j)}{\left\| \boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \rho^i_j \right\|^2} \overset{(2.4)}{=} \frac{(f, \rho^i_j) - (\boldsymbol{\mathcal{K}} \nabla u^i_{J,j-1}, \nabla \rho^i_j)}{\left\| \boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \rho^i_j \right\|^2}.$$

□

Remark 4.4 (Construction of the new iterate)**.** *The construction of* u^{i+1}_J *from* u^i_J *by the solver of Definition 4.1 can be seen as one iteration of a V-cycle multigrid, with no pre- and one post-smoothing step, with an optimal step-size at the error correction stage. The smoother on each level is additive Schwarz associated to patch subdomains where the local problems* (2.14) *are defined. Note that when* $p_j = 1$*,* $j \in \{1, \dots, J\}$*, the smoother is the diagonal Jacobi smoother, whereas when* $p_j > 1$*, the smoother is block-Jacobi. As detailed in [Miraçi et al., 2020, Section 6.2], see Section 6.2 in Chapter 1, employing a weighted restricted additive Schwarz smoothing (wRAS) can offer a further speed-up of the solver, briefly addressed in Section 9.4 below.*

Remark 4.5 (Connection of local contributions with level-wise updates)**.** *Note that for* ρ^i_j *given by* (2.14)–(2.15)*,* $j \in \{1, \dots, J\}$*, we have*

$$\sum_{\mathbf{a} \in \mathcal{V}_j} \left\| \boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \rho^i_{j,\mathbf{a}} \right\|^2_{\omega^{\mathbf{a}}_j} \overset{\substack{(2.14)\\(2.15)}}{=} (f, \rho^i_j) - \big(\boldsymbol{\mathcal{K}} \nabla u^i_{J,j-1}, \nabla \rho^i_j \big) \overset{(2.16)}{=} \lambda^i_j \left\| \boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \rho^i_j \right\|^2. \tag{2.21}$$

Remark 4.6 (Extension of the solver to *hp*-refinement hierarchy)**.** *The multilevel approach we take in this work can be easily extended to a setting where the mesh and space hierarchies are obtained by hp-refinement, since all we require in our multilevel construction of Definition 4.1 is nestedness of the meshes and finite element spaces. To obtain the theoretical results, one would need to adapt the stable decomposition results of Schöberl et al. [2008] from a global fixed polynomial order to a variable one.*

The optimal step-sizes also lead to the following important result, which can be compared to the orthogonal error decomposition (2.13).

Theorem 4.7 (Error representation of one solver step)**.** *For $u_J^i \in V_J^p$, let $u_J^{i+1} \in V_J^p$ be given by Definiton 4.1. Then*

$$\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J - u_J^{i+1})\right\|^2 = \left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J - u_J^i)\right\|^2 - \sum_{j=0}^{J}\left(\lambda_j^i\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_j^i\right\|\right)^2 \tag{2.22a}$$

$$= \left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J - u_J^i)\right\|^2 - \left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_0^i\right\|^2 - \sum_{j=1}^{J}\lambda_j^i\sum_{\mathbf{a}\in\mathcal{V}_j}\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_{j,\mathbf{a}}^i\right\|_{\omega_j^{\mathbf{a}}}^2. \tag{2.22b}$$

Proof. The second line (2.22b) follows immediately upon multiplying (2.21) by λ_j^i on both sides and summing over the mesh levels. We obtain the first line (2.22a) by going through the levels from finest to coarsest and using the relation of each level's update with its associated optimal step-size, similarly to (2.20):

$$\begin{aligned}
\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J - u_J^{i+1})\right\|^2 &\overset{(2.17)}{=} \left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J - (u_{J,J-1}^i + \lambda_J^i\rho_{J,\mathrm{alg}}^i))\right\|^2 \\
&\overset{(2.4)}{\underset{(2.16)}{=}} \left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J - u_{J,J-1}^i)\right\|^2 - \left(\lambda_J^i\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_{J,\mathrm{alg}}^i\right\|\right)^2 = \dots \\
&= \left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\big(u_J - (u_J^i + \lambda_0^i\rho_0^i)\big)\right\|^2 - \sum_{j=1}^{J}\left(\lambda_j^i\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_j^i\right\|\right)^2 \\
&\overset{(2.9)}{=} \left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J - u_J^i)\right\|^2 - \sum_{j=0}^{J}\left(\lambda_j^i\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_j^i\right\|\right)^2.
\end{aligned}$$

□

5 A posteriori estimator on the algebraic error

We now present how the solver introduced in Section 4 induces an a posteriori estimator η_{alg}^i.

Definition 5.1 (Algebraic error estimator)**.** *Let $u_J^i \in V_J^p$ be arbitrary and let $u_J^{i+1} \in V_J^p$ be the update at the end of one step of the solver introduced in Definition 4.1. We define the algebraic error estimator*

$$\eta_{\mathrm{alg}}^i := \left(\sum_{j=0}^{J}\left(\lambda_j^i\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_j^i\right\|\right)^2\right)^{\frac{1}{2}}. \tag{2.23}$$

Following Theorem 4.7, the estimator η_{alg}^i is immediately a guaranteed lower bound on the algebraic error.

Lemma 5.2 (Guaranteed lower bound on the algebraic error)**.** *There holds:*

$$\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J - u_J^i)\right\| \geq \eta_{\mathrm{alg}}^i. \tag{2.24}$$

6 Main results

In this section, we present the main results concerning our multilevel solver of Definition 4.1 and our a posteriori estimator η^i_{alg} of Definition 5.1. As in Miraçi et al. [2020], see Chapter 1, these two results are equivalent. We first collect our assumptions.

6.1 Setting, mesh, and regularity assumptions

For any mesh level $j \in \{1, \dots, J\}$, we denote by $h_K := \text{diam}(K)$ the diameter of the element $K \in \mathcal{T}_j$ and by $h_j = \max_{K \in \mathcal{T}_j} h_K$ the mesh size on level j. We shall always assume that our meshes are shape-regular:

Assumption 6.1 (Mesh shape regularity)**.** *There exists $\kappa_{\mathcal{T}} > 0$ such that*

$$\max_{K \in \mathcal{T}_j} \frac{h_K}{\rho_K} \leq \kappa_{\mathcal{T}} \text{ for all } 0 \leq j \leq J, \tag{2.25}$$

where ρ_K denotes the diameter of the largest ball contained in K.

Below, we work in one of the three following settings. In the first setting, the hierarchy consists of quasi-uniform meshes with a bounded refinement factor between consecutive levels:

Assumption 6.2 (Refinement strength and mesh quasi-uniformity)**.** *There exists a fixed positive real number $0 < C_{\text{ref}} \leq 1$ such that for all $j \in \{1, \dots, J\}$, for all $K \in \mathcal{T}_{j-1}$, and for any $K^* \in \mathcal{T}_j$ such that $K^* \subset K$, there holds*

$$C_{\text{ref}} h_K \leq h_{K^*} \leq h_K. \tag{2.26}$$

There further exists a fixed positive real number $0 < C_{\text{qu}} \leq 1$ such that for all $j \in \{0, \dots, J\}$ and for all $K \in \mathcal{T}_j$, there holds

$$C_{\text{qu}} h_j \leq h_K \leq h_j. \tag{2.27}$$

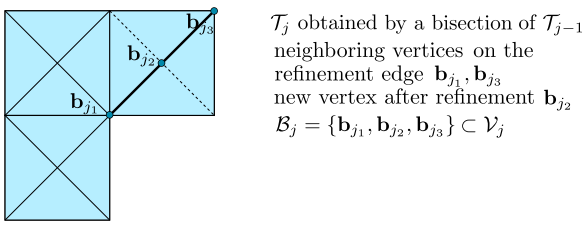


Figure 2.2: Illustration of the set $\mathcal{B}_j$; the refinement $\mathcal{T}_j$ (dotted lines) of the mesh $\mathcal{T}_{j-1}$ (full lines).

In the second setting, we work with a hierarchy generated from a quasi-uniform coarse mesh by a series of bisections, e.g. newest vertex bisection, cf. Sewell [1972] and Mitchell [1991]. In this case, one refinement edge of $\mathcal{T}_{j-1}$, for $j \in \{1, \dots, J\}$, gives us a new finer mesh $\mathcal{T}_j$. We denote by $\mathcal{B}_j \subset \mathcal{V}_j$ the set consisting of the new vertex obtained after the bisection together with its two neighbors on the refinement edge; see Figure 2.2 for $d = 2$. We also denote by $h_{\mathcal{B}_j}$ the maximal diameter of elements having a vertex in the set $\mathcal{B}_j$, for $j \in \{1, \dots, J\}$. Here we assume:

Assumption 6.3 (Local refinement strength and the coarsest mesh quasi-uniformity of bisection-generated meshes)**.** *The coarsest mesh $\mathcal{T}_0$ is a conforming quasi-uniform mesh in the sense of (2.27), with parameter $0 < C^0_{\mathrm{qu}} \leq 1$. The (possibly highly graded) conforming mesh $\mathcal{T}_J$ is generated from $\mathcal{T}_0$ by a series of bisections. There exists a fixed positive real number $0 < C_{\mathrm{loc,qu}} \leq 1$ such that for all $j \in \{1, \dots, J\}$, there holds*

$$C_{\mathrm{loc,qu}} h_{\mathcal{B}_j} \leq h_K \leq h_{\mathcal{B}_j} \quad \forall K \in \mathcal{T}_j \text{ such that a vertex of } K \text{ belongs to } \mathcal{B}_j. \tag{2.28}$$

In the third setting, we assume:

Assumption 6.4 (Refinement strength, mesh quasi-uniformity, and H^2-regularity)**.** *Let Assumption 2.2 hold. Moreover, let for each $g \in L^2(\Omega)$, $w_g \in H^1_0(\Omega)$ such that*

$$(\nabla w_g, \nabla v) = (g, v) \quad \forall v \in H^1_0(\Omega),$$

belong to $H^2(\Omega)$.

6.2 Main results

We now present our main results, the proofs of which are given in Section 10. For the solver it holds:

Theorem 6.5 (p-robust error contraction of the multilevel solver)**.** *Let $u_J \in V^p_J$ be the (unknown) finite element solution of (2.4) and let $u^i_J \in V^p_J$ be arbitrary, $i \geq 0$. Take u^{i+1}_J to be constructed from u^i_J using one step of the multilevel solver of Definition 4.1. Under Assumption 6.1 and either of Assumptions 6.2, 6.3, or 6.4, there holds*

$$\big\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla (u_J - u^{i+1}_J)\big\| \leq \alpha \big\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla (u_J - u^i_J)\big\|. \tag{2.29}$$

Here $0 < \alpha < 1$ depends on the space dimension d, the mesh shape regularity parameter $\kappa_{\mathcal{T}}$, the ratio of the largest and the smallest eigenvalue of the diffusion coefficient $\boldsymbol{\mathcal{K}}$, and additionally on: (i) the parameters C_{ref} and C_{qu} and at most linearly on the number of mesh levels J under Assumption 6.2; (ii) the parameters C^0_{qu} and $C_{\mathrm{loc,qu}}$ and at most linearly on the number of mesh levels J under Assumption 6.3; (iii) the parameters C_{ref} and C_{qu} under Assumption 6.4. In particular, α is independent of the polynomial degree p.

In (2.29), α represents an upper bound on the algebraic error contraction factor at each step i. In particular, this means that the solver of Definition 4.1 contracts the algebraic error at each iteration step robustly with respect to the polynomial degree p. Moreover, under the Assumption 6.4, the contraction is also robust with respect to the number of mesh levels J .

For the estimator, in turn, we have:

Theorem 6.6 (p-robust reliable and efficient bound on the algebraic error)**.** *Let $u_J \in V^p_J$ be the (unknown) finite element solution of (2.4) and let $u^i_J \in V^p_J$ be arbitrary, $i \geq 0$. Let η^i_{alg} be given by Definition 5.1. Let Assumption 6.1 and either of Assumptions 6.2, 6.3, or 6.4 hold. Then, in addition to $\big\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla (u_J - u^i_J)\big\| \geq \eta^i_{\mathrm{alg}}$ of (2.24), there holds*

$$\eta^i_{\mathrm{alg}} \geq \beta \big\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla (u_J - u^i_J)\big\|, \tag{2.30}$$

where $0 < \beta < 1$ *is given by* $\beta = \sqrt{1-\alpha^2}$ *with* α *from* (2.29).

Theorem 6.6 allows to write η^i_{alg} as a two-sided bound of the algebraic error (up to the constant β for the upper bound), meaning that the estimator is reliable and efficient, robustly with respect to the polynomial degree p.

6.3 Additional results

Theorems 6.5 and 6.6 are actually equivalent, similarly to [Miraçi et al., 2020, Corollary 5.4], see Corollary 5.4 in Chapter 1, (we thus only prove Theorem 6.6 in Section 10 below).

Corollary 6.7 (Equivalence of the p-robust solver contraction and p-robust estimator efficiency)**.** *Let the assumptions of Theorems 6.5 and 6.6 be satisfied. Then* (2.29) *holds if and only if* (2.30) *holds, and* $\alpha = \sqrt{1-\beta^2}$.

Proof. We give the proof for completeness. Starting from (2.29), with $0 < \alpha < 1$,

$$\begin{aligned}
&\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J - u_J^{i+1})\right\|^2 \leq \alpha^2 \left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J - u_J^{i})\right\|^2 \\
\overset{(2.22a)}{\Leftrightarrow}\ &\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J - u_J^{i})\right\|^2 - \sum_{j=0}^{J}(\lambda_j^i \left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_j^i\right\|)^2 \leq \alpha^2 \left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J - u_J^{i})\right\|^2 \\
\overset{(2.23)}{\Leftrightarrow}\ &\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J - u_J^{i})\right\|^2 (1-\alpha^2) \leq \left(\eta^i_{\mathrm{alg}}\right)^2.
\end{aligned}$$

□

Finally, the following corollary formulates a three-part equivalence (recall that the step-sizes are given by (2.16) and the local (patch-wise) contributions by (2.14)).

Corollary 6.8 (Equivalence error–estimator–localized contributions)**.** *Let Assumption 6.1 hold, as well as either of Assumptions 6.2, 6.3, or 6.4. Then*

$$\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J - u_J^{i})\right\|^2 \approx \left(\eta^i_{\mathrm{alg}}\right)^2 = \left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_0^i\right\|^2 + \sum_{j=1}^{J}\lambda_j^i \sum_{\mathbf{a}\in\mathcal{V}_j}\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_{j,\mathbf{a}}^i\right\|^2_{\omega_j^{\mathbf{a}}}, \tag{2.31}$$

where the constant hidden in the equivalence is β *from* (2.30).

Proof. Under Assumptions 6.2, 6.3, or 6.4, Theorem 6.6 together with (2.24) gives $\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J - u_J^{i})\right\| \approx \eta^i_{\mathrm{alg}}$. The equality in (2.31) is easily obtained as in Theorem 4.7 upon multiplying (2.21) by λ_j^i on both sides and summing over the mesh levels. □

Remark 6.9. *(Localized a posteriori estimator of the algebraic error) The localization* (2.31) *is over vertex patches as in the a posteriori error estimators of the discretization error in the finite element method, see* e.g.*, Babuška and Rheinboldt [1978] or Verfürth [1996]. Therein, the construction also relies on solving local Dirichlet problems.*

7 Adaptive number of smoothing steps

We consider here a simple and practical way to make the solver described in Definition 4.1 choose autonomously and adaptively the number of smoothing steps on each mesh level. The idea of the adaptive version is to make more post-smoothing steps *if needed* on levels that contribute most to the algebraic error. This is decided relying on the a posteriori error estimate on the algebraic error we have at our disposal, relying on a Dörfler-type condition, cf. Dörfler [1996].

Definition 7.1 (Adaptive multilevel solver)**.** *Let $\nu_{\max} \geq 1$ be a user-specified maximal number of smoothing steps and let $0 < \theta < 1$ be a bulk-chasing parameter.*

1. *Initialize $u_J^0 \in V_J^p$ as the zero function and set $i := 0$.*
2. *Perform the following steps* (a)–(d):
   - (a) *Let ρ_0^i be constructed by (2.9). Set $\rho_{0,1}^i := \rho_0^i$, $\lambda_{0,1}^i := 1$, $\nu_0^i := 1$, and $u_{J,0}^i := u_J^i + \lambda_{0,1}^i \rho_{0,1}^i$.*
   - (b) *For $j = 1 : J$:*
     - i. *Set $\nu := 1$.*
     - ii. *From $u_{J,j-1}^i$, construct ρ_j^i and λ_j^i by (2.14)–(2.16).*
       Set $\rho_{j,\nu}^i := \rho_j^i$, $\lambda_{j,\nu}^i := \lambda_j^i$, $u_{J,j,\nu}^i := u_{J,j-1}^i + \lambda_{j,\nu}^i \rho_{j,\nu}^i$, and
       `while` $\Big[\nu < \nu_{\max}$ `and`
       $\big(\lambda_{j,\nu}^i \|\boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \rho_{j,\nu}^i\|\big)^2 \geq \theta^2 \Big(\sum_{k=0}^{j-1} \sum_{\ell=1}^{\nu_k^i} \big(\lambda_{k,\ell}^i \|\boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \rho_{k,\ell}^i\|\big)^2 + \sum_{\ell=1}^{\nu-1} \big(\lambda_{j,\ell}^i \|\boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \rho_{j,\ell}^i\|\big)^2 \Big)\Big]$
       `do` *Set $\nu := \nu + 1$.*
       From $u_{J,j,\nu-1}^i$, construct $\bar{\rho}_j^i$ and $\bar{\lambda}_j^i$ by (2.14)–(2.16).
       Set $\rho_{j,\nu}^i := \bar{\rho}_j^i$, $\lambda_{j,\nu}^i := \bar{\lambda}_j^i$, $u_{J,j,\nu}^i := u_{J,j,\nu-1}^i + \lambda_{j,\nu}^i \rho_{j,\nu}^i$.
       `endwhile`
     - iii. *Set $\nu_j^i = \nu$ and $u_{J,j}^i := u_{J,j,\nu}^i$.*
   - (c) *Define the final update on step i as $u_J^{i+1} := u_{J,J}^i \in V_J^p$.*
   - (d) *If $u_J^{i+1} = u_J^i$, then stop the solver. Otherwise increase $i := i + 1$ and go to step* 2(a).

Remark 7.2 (Adaptive substep)**.** *Note that if we skip the adaptive substep in* 2(b) *in Definition 7.1 by setting $\nu_{\max} = 1$, we obtain the non-adaptive version of the solver of Definition 4.1.*

Remark 7.3 (Optimal step-sizes and adaptive number of smoothing steps as a general approach)**.** *The main ideas of optimal step-size per level and adaptive number of smoothing steps we use in Definition 7.1 can be used in other geometric multigrid solvers. Implementation-wise, these ideas are easy to add to existing codes and alleviate the task of choosing the number of smoothing steps arbitrarily.*

Remark 7.4 (Adaptivity criterion)**.** *The bulk-chasing (Dörfler's) marking criterion is not crucial above, other criteria like the maximal one can be considered as well. We note that we do not analyze here the influence of the additional adaptive smoothing steps on the convergence speed.*

8 Complexity of the solver

We wish to give some insights into the complexity of the proposed solver here. In particular, a way of estimating the number of floating point operations can be done by the formula

$$\begin{aligned}\text{nflops} := &\frac{|\mathcal{V}_0|^3}{3} + \sum_{j=1}^{J}\sum_{\mathbf{a}\in\mathcal{V}_j}\frac{\text{ndof}(V_j^{\mathbf{a}})^3}{3} + \sum_{i=1}^{i_\text{s}}\left[2|\mathcal{V}_0|^2 + \sum_{j=1}^{J}\nu_j^i\sum_{\mathbf{a}\in\mathcal{V}_j}2\text{ndof}(V_j^{\mathbf{a}})^2\right]\\ &+\sum_{i=1}^{i_\text{s}}\sum_{j=1}^{J}\left[2\,\text{nnz}(\mathcal{I}_{j-1}^j) + 2\,\text{nnz}(\mathcal{I}_j^{j-1}) + 2\nu_j^i\,\text{nnz}(\mathbb{A}_j) + 3\nu_j^i(2\,\text{size}(\mathbb{A}_j))\right].\end{aligned} \tag{2.32}$$

This formula is derived assuming 1) an initial Cholesky decomposition of the local matrices associated to each patch on each level except for the coarsest one, where the global stiffness matrix for piecewise affine functions is factorized (for a matrix of size n, this cost is estimated as $1/3n^3$); 2) local solves by forward and backward substitutions (cost $2n^2$); 3) intergrid operators $\mathcal{I}_{j-1}^j : V_{j-1}^{p_{j-1}} \to V_j^{p_j}$ with the cost estimated by two-times the number of nonzeros of the associated interpolation matrix; and 4) evaluation of the optimal step-sizes λ_j as in formula (2.16) with a cost equal to two-times the number of nonzeros of the stiffness matrix $\mathbb{A}_j$ on the given level and three inner products. Recall that ν_j^i is the number of smoothing steps on level j at iteration i.

We would like to point out that the above estimation (2.32) is a worst-case scenario. In fact, in the case of a structured initial mesh $\mathcal{T}_0$ containing an arbitrary number of simplices, or $\mathcal{T}_0$ only containing a few simplices and uniform or newest vertex bisection graded refinemement, most patches have the same geometry. Then the second (cubic, potentially dominant) term in (2.32) almost vanishes. Moreover, the developed solver and estimator are fully parallelizable on each mesh level and thus the discussion of complexity in floating point operations no longer has the same meaning in a parallel implementation; in particular, all the terms in (2.32) containing the sum over (all) vertices can be fully parallelized. On the other hand, formula (2.32) ignores the operations needed to evaluate the right-hand sides of local problems (2.14). Such evaluation may affect the overall flops count, but this is very dependent on the particular implementation.

9 Numerical experiments

In this section, we first consider three test cases with the diffusion tensor constant in Ω, $\boldsymbol{\mathcal{K}} = I$, where the domains $\Omega \subset \mathbb{R}^2$ and the exact solutions u are given by

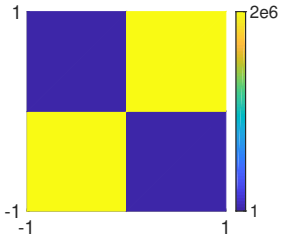
$$\text{Sine:}\quad u(x,y) := \sin(2\pi x)\sin(2\pi y),\ \Omega := (-1,1)^2, \tag{2.33}$$

$$\text{Peak:}\quad u(x,y) := x(x-1)y(y-1)\mathrm{e}^{-100((x-0.5)^2-(y-0.117)^2)},\ \Omega := (0,1)^2, \tag{2.34}$$

$$\text{L-shape:}\quad u(r,\theta) := r^{2/3}\sin(2\theta/3),\ \Omega := (-1,1)^2 \setminus ([0,1]\times[-1,0]). \tag{2.35}$$

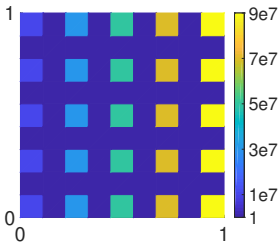
We further consider two tests with piecewise constant diffusion tensor $\boldsymbol{\mathcal{K}} = c(x,y)\cdot I$ on the square domain. For each of these test, we will vary $c(x,y)$ in order to study its influence on the solver's performance. The tests are described by

$$\text{Checkerboard: } u(r,\varphi) = r^{\gamma}\mu(\varphi), \;\; \Omega := (-1,1)^2, \tag{2.36}$$



where $\mu(\varphi)$ is constructed following Kellogg [1975]. We consider the case $\gamma = 1$, $\boldsymbol{\mathcal{K}} = I$, and a singular solution with $\gamma = 0.0009$ and diffusion contrast 2001405.429972. For the latter, $c(x,y)$ varies across the domain as in the figure on the left.

$$\text{Skyscraper: unknown analytic solution, } \Omega := (0,1)^2. \tag{2.37}$$



The variations of $c(x,y)$ are shown in the figure on the left. We take the source term $f = 1$ and Dirichlet boundary condition $u_D(x,y) = \sqrt{x}$ on $\partial\Omega$. We adjust $c(x,y)$ to obtain two tests: one with diffusion contrast proportional to 1, and another proportional to 10^7. An analogous test case is also described and used in [Anciaux-Sedrakian et al., 2020, Section 5.3].

In all tests, the exact solution of the algebraic systems is given by a direct solver.

			Sine $\boldsymbol{\mathcal{K}}=I$		Peak $\boldsymbol{\mathcal{K}}=I$		L-shape $\boldsymbol{\mathcal{K}}=I$		Checkerboard $\boldsymbol{\mathcal{K}}=I$		Checkerboard $\mathcal{J}(\boldsymbol{\mathcal{K}})=O(10^6)$		Skyscraper $\mathcal{J}(\boldsymbol{\mathcal{K}})=O(1)$		Skyscraper $\mathcal{J}(\boldsymbol{\mathcal{K}})=O(10^7)$	
p_j			1	p	1	p	1	p	1	p	1	p	1	p	1	p
J	p	DoF	i_s	i_s	i_s	i_s	i_s	i_s	i_s	i_s	i_s	i_s	i_s	i_s	i_s	i_s
3	1	$2e^4$	19	19	19	19	21	21	18	18	18	18	19	19	19	19
	3	$1e^5$	29	13	28	14	29	11	27	11	28	11	31	13	31	13
	6	$6e^5$	30	13	30	14	26	9	24	9	25	10	28	11	28	11
	9	$1e^6$	31	14	30	14	23	9	23	9	23	9	26	10	26	10
4	1	$6e^4$	21	21	20	20	21	21	19	19	19	19	19	19	19	19
	3	$6e^5$	29	13	29	14	28	11	26	11	27	11	30	11	30	11
	6	$2e^6$	31	13	30	14	25	9	24	9	24	9	27	10	27	10
	9	$5e^6$	32	14	31	15	23	9	22	9	23	9	25	9	25	9

Table 2.1: Number of iterations i_s for different polynomial degrees p, number of mesh levels J, space hierarchies with two different p_j, $j \in \{1,\dots,J-1\}$, and jump in the diffusion coefficient $\mathcal{J}(\boldsymbol{\mathcal{K}})$.

9.1 Performance of the multilevel solver of Definition 4.1

We first consider mesh hierarchies obtained by J uniform refinements of an initial Delaunay triangulation of the domain Ω. We study the solver of Definition 4.1 stopped when the ℓ^2-norm of the algebraic residual vector drops below 10^{-5} times the initial one; then we expect for a p-robust solver that the number of iterations i_s needed to reach it will be similar for different polynomial degrees. We also numerically investigate J-robustness and robustness with respect to the jump in the diffusion coefficient, denoted henceforth by $\mathcal{J}(\boldsymbol{\mathcal{K}})$. Results presented in Table 2.1

L-shape, $\mathcal{K} = I$

J	p	i_s	J	p	i_s
10	1	15	15	1	17
	3	6		3	11
	6	5		6	5
	9	5		9	4

Checkerboard, $\mathcal{J}(\mathcal{K}) = O(10^6)$

J	p	i_s	J	p	i_s
7	1	43	10	1	57
	3	18		3	23
	6	13		6	15
	9	11		9	12

Table 2.2: Number of iterations i_s for different polynomial degrees p, number of mesh levels J, space hierarchies given by $p_j = p$, $j \in \{1, \dots, J-1\}$; graded mesh hierarchies.

confirm perfect p-robustness, as well as numerical $\mathcal{K}$- and J-robustness even in low-regularity cases.

We now present some experiments for graded mesh hierarchies. The meshes were obtained by the newest vertex bisection algorithm, cf. Sewell [1972] and Mitchell [1991], and a Dörfler's bulk-chasing criterion Dörfler [1996] which uses the true discretizaton error. The true discretizaton error is used in the marking for refinement instead of an a posteriori discretization error estimator for the purpose of simplicity and result reproducibility: our main goal is to test the solver of Definition 4.1 in graded meshes that satisfy Assumption 6.3. The resulting meshes are depicted in Figure 2.3 for two different test cases, and the results are given in Table 2.2. We observe perfect p-robustness behavior of the solver of Definition 4.1, which is in agreement with our theoretical results also covering graded mesh hierarchies.

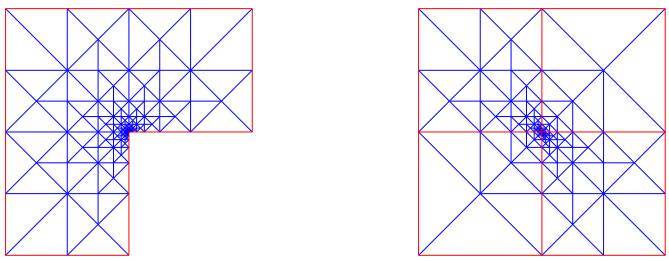


Figure 2.3: Graded meshes obtained by the newest-vertex bisection algorithm. Left: L-shape problem and $J = 10$. Right: Checkerboard $\mathrm{O}(10^6)$ and $J = 10$. The regions where the diffusion coefficient is constant are bordered by red lines.

9.2 Adaptive number of smoothing steps using Definition 7.1

Now we will study the behavior of the solver described in Definition 7.1, where we set the maximum number of smoothing steps $\nu_{\max} = 5$. In order to do a comparison study of the solver's performance in different settings, we will use the estimated number of floating point operations (2.32) and we also introduce the number of global synchronizations

$$\text{sync} := i_s + \sum_{i=1}^{i_s} \sum_{j=1}^{J} \nu_j^i. \tag{2.38}$$

9.2.1 Dependence on θ

In Figure 2.4 we report the cumulated number of the smoothing steps employed at each level for different choices of θ. The non-adaptive variant of solver of Definition 4.1 ($\nu_{\max} = 1$) is also plotted for comparison. Recall that this employs just one post-smoothing step, and may lead to an increased number of iterations, whereas the solver of Definition 7.1 makes more smoothing steps and typically cuts the number of iterations. If in Figure 2.4 we find for a given θ that all numbers are consistently low for all levels, then this results in a cheaper procedure and gives us an idea of the best candidates for θ. Table 2.3 then gives the detailed numbers of smoothing steps per level and iteration for $\theta = 0.2$.

In Table 2.4 more results are presented together with the estimated costs in order to compare the performance of the solver for different values of θ. Most often, the costs are very close for different choices of θ and in practice the choice $\theta = 0.2$ is quite satisfactory. It typically brings the number of iterations down to 5-8 from 9-28, upon usually performing 2-4 post-smoothing steps on each level instead of just one. Note also that choosing θ in our setting somehow differs from typical bulk-chasing criteria, where larger θ means including more elements. Here instead, smaller θ make the condition of the while loop of Definition 7.1 more likely to be satisfied, thus leading to more smoothing steps and overall smaller iteration numbers, as seen in Table 2.4.

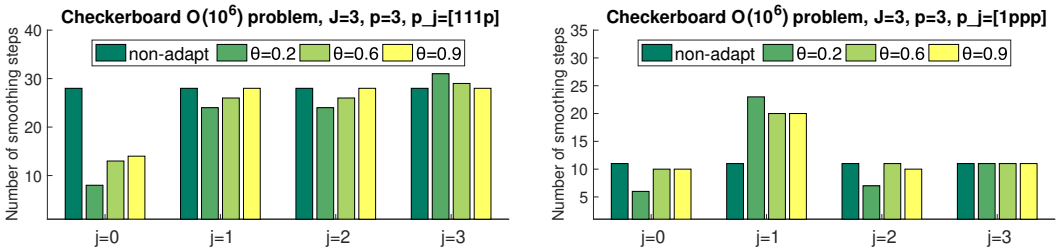


Figure 2.4: Number of smoothing steps per level for the Checkerboard case, polynomial degree $p = 3$, number of mesh levels $J = 3$, diffusion coefficient jump $\mathcal{J}(\mathcal{K}) = O(10^6)$, and mesh hierarchies with $p_j = 1$ and $p_j = p$, $j \in \{1, \dots, J-1\}$.

	$p_j = 1$								$p_j = p$					
	it=1	it=2	it=3	it=4	it=5	it=6	it=7	it=8	it=1	it=2	it=3	it=4	it=5	it=6
level 0	1	1	1	1	1	1	1	1	1	1	1	1	1	1
level 1	3	3	3	3	3	3	3	3	3	4	4	4	4	4
level 2	3	3	3	3	3	3	3	3	2	1	1	1	1	1
level 3	3	4	4	4	4	4	4	4	2	2	2	2	2	1

Table 2.3: Number of smoothing steps per level in each iteration it for the Checkerboard case, $\theta = 0.2$, polynomial degree $p = 3$, number of mesh levels $J = 3$, diffusion coefficient jump $\mathcal{J}(\mathcal{K}) = O(10^6)$, and mesh hierarchies with $p_j = 1$ and $p_j = p$, $j \in \{1, \dots, J-1\}$.

		L-shape test case								Checkerboard $O(10^6)$							
		non-adapt		$\theta = 0.2$		$\theta = 0.6$		$\theta = 0.9$		non-adapt		$\theta = 0.2$		$\theta = 0.6$		$\theta = 0.9$	
J	p_j	i_s	nflops	i_s	nflops	i_s	nflops	i_s	nflops	i_s	nflops	i_s	nflops	i_s	nflops	i_s	nflops
3	1111	21	$2.17e^7$	7	$1.57e^7$	11	$1.75e^7$	11	$1.67e^7$	18	$2.01e^7$	8	$1.76e^7$	12	$1.91e^7$	11	$1.72e^7$
	1113	29	$6.05e^8$	7	$5.28e^8$	12	$5.75e^8$	15	$5.84e^8$	28	$6.05e^8$	8	$6.01e^8$	13	$5.80e^8$	14	$5.66e^8$
	1116	26	$1.20e^{10}$	7	$1.28e^{10}$	11	$1.22e^{10}$	13	$1.19e^{10}$	25	$1.21e^{10}$	8	$1.38e^{10}$	12	$1.23e^{10}$	13	$1.23e^{10}$
	1119	23	$9.08e^{10}$	6	$9.22e^{10}$	10	$9.23e^{10}$	12	$9.23e^{10}$	23	$9.39e^{10}$	7	$1.00e^{11}$	12	$9.54e^{10}$	12	$9.54e^{10}$
	1333	11	$3.90e^8$	6	$3.61e^8$	10	$4.07e^8$	10	$3.86e^8$	11	$4.04e^8$	6	$3.52e^8$	10	$4.04e^8$	10	$3.99e^8$
	1666	9	$9.49e^9$	6	$1.00e^{10}$	8	$9.53e^9$	8	$9.45e^9$	10	$1.03e^{10}$	6	$9.71e^9$	9	$1.04e^{10}$	8	$9.77e^9$
	1999	9	$9.18e^{10}$	6	$9.31e^{10}$	8	$9.21e^{10}$	8	$9.17e^{10}$	9	$9.48e^{10}$	6	$9.45e^{10}$	8	$9.51e^{10}$	8	$9.47e^{10}$
4	11111	21	$7.24e^7$	8	$5.61e^7$	11	$5.66e^7$	12	$6.00e^7$	19	$6.83e^7$	9	$6.29e^7$	11	$5.71e^7$	12	$5.92e^7$
	11113	28	$2.34e^9$	7	$2.04e^9$	12	$2.30e^9$	14	$2.19e^9$	27	$2.33e^9$	8	$2.40e^9$	12	$2.17e^9$	14	$2.26e^9$
	11116	25	$4.69e^{10}$	7	$5.00e^{10}$	11	$4.77e^{10}$	13	$4.78e^{10}$	24	$4.72e^{10}$	7	$5.04e^{10}$	12	$4.93e^{10}$	13	$4.93e^{10}$
	11119	23	$3.65e^{11}$	7	$3.97e^{11}$	10	$3.64e^{11}$	12	$3.71e^{11}$	23	$3.77e^{11}$	7	$4.03e^{11}$	11	$3.76e^{11}$	12	$3.83e^{11}$
	13333	11	$1.59e^9$	6	$1.43e^9$	9	$1.50e^9$	10	$1.61e^9$	11	$1.64e^9$	6	$1.48e^9$	9	$1.55e^9$	10	$1.59e^9$
	16666	9	$3.88e^{10}$	5	$3.65e^{10}$	8	$3.85e^{10}$	8	$3.81e^{10}$	9	$4.00e^{10}$	6	$3.99e^{10}$	9	$4.19e^{10}$	8	$3.94e^{10}$
	19999	9	$3.74e^{11}$	5	$3.64e^{11}$	8	$3.73e^{11}$	8	$3.71e^{11}$	9	$3.87e^{11}$	6	$3.78e^{11}$	8	$3.86e^{11}$	8	$3.83e^{11}$

Table 2.4: Estimated number of floating point operations given by (2.32) and number of iterations i_s for two singular test cases, different polynomial degrees p, number of mesh levels J, and space hierarchies with p_j, $j \in \{0, \dots, J\}$.

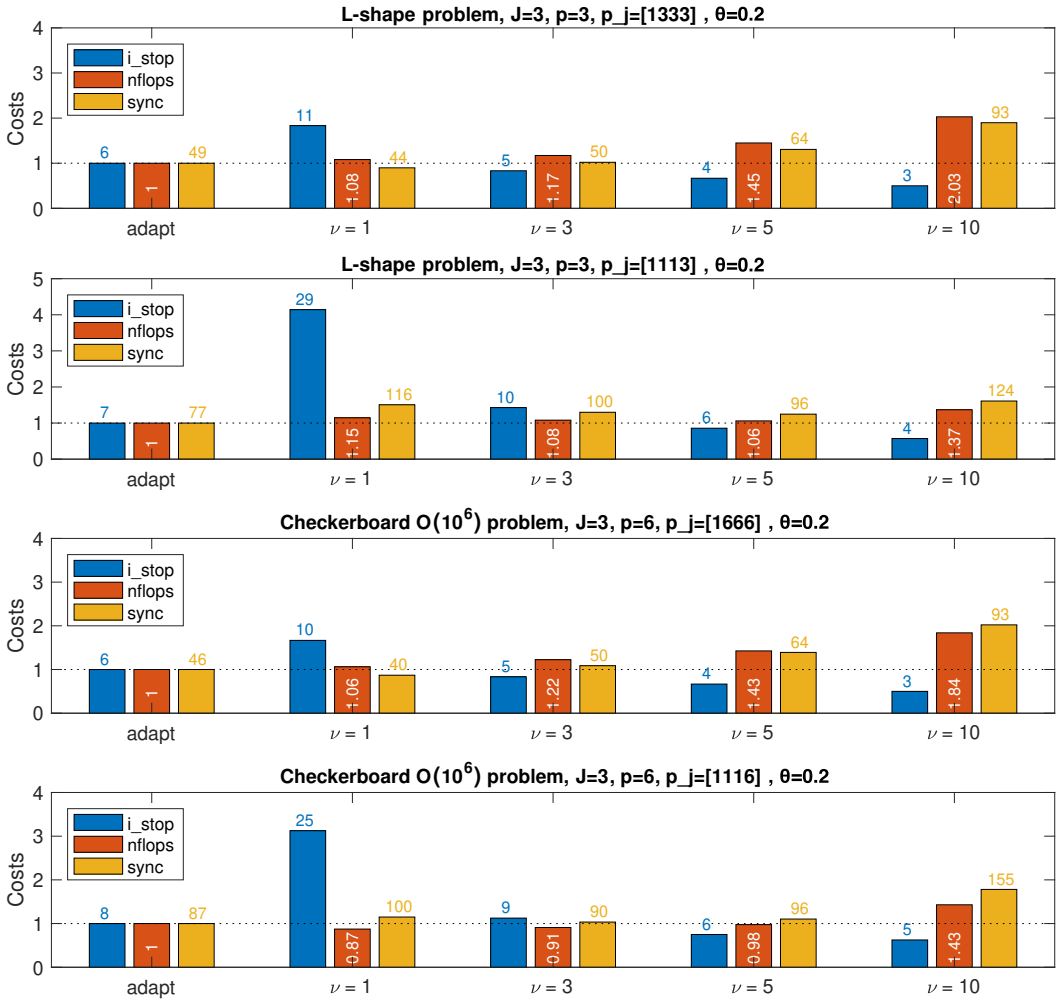


Figure 2.5: Comparison between a fixed number of (block-Jacobi) smoothing steps ν on all levels (Definition 4.1 and its obvious modification for $\nu \geq 1$) and the adaptive number of smoothing steps of Definition 7.1. Number of iterations i_s, floating point operations given by (2.32) relative with respect to Definition 7.1, and the number of global synchronizations by (2.38).

9.2.2 Performance of the adaptive solver of Definition 7.1

In Figure 2.5, we fix $\theta = 0.2$ and we compare our adaptive number of smoothing steps approach with the number of smoothing steps $\nu \geq 1$ being fixed to the same value on each mesh level. Relative to the adaptive approach, the solver using a fixed number of smoothing steps, whatever it is, is typically more costly, both in terms of nflops computed by (2.32) and of sync computed by (2.38). Note also that when using a fixed number of smoothing steps, the simplistic solver of Definition 4.1 ($\nu_{\max} = 1$) is often the cheapest to employ, although its number of iterations may seem rather increased at first sight. As for the adaptive solver, we also point out that the maximum number of smoothing steps $\nu_{\max} = 5$ is hardly ever reached in our experiments, endorsing our adaptive approach in two ways: a fixed number of smoothing steps is not the best way to take advantage of a multigrid solver; the criterion used for the while loop in Definition 7.1 successfully identifies the levels in which more smoothing steps are necessary, without over-smoothing.

9.3 Examples in three space dimensions

We consider now three test cases where $\Omega \subset \mathbb{R}^3$, $\boldsymbol{\mathcal{K}} = I$ except in areas of the domain explicitly specified below, and, when available, exact solution u:

$$\text{Cube:} \quad u(x,y,z) := x(x-1)y(y-1)z(z-1), \ \Omega := (0,1)^3, \tag{2.39}$$

$$\text{Nested cubes:} \quad \text{unknown analytic solution}, \ \Omega := (-1,1)^3, \quad \boldsymbol{\mathcal{K}} = 10^5 * I \text{ in } (-0.5,0.5)^3, \tag{2.40}$$

$$\text{Checkers cubes:} \quad \text{unknown analytic solution}, \ \Omega := (0,1)^3, \quad \boldsymbol{\mathcal{K}} = 10^6 * I \text{ in } (0,0.5)^3 \cup (0.5,1)^3. \tag{2.41}$$

In the case of nested cubes and checkers cubes, the source term is given by $f = 1$ in Ω and zero Dirichlet boundary conditions are prescribed on $\partial\Omega$.

We employ our solver of Definition 4.1 for polynomial degrees $p = 1,2,3,4$, number of mesh levels $J = 4$, and hierarchies given by $p_j = 1$, $j \in \{1,\dots,J-1\}$. The coarse mesh in all these test cases is unstructured and the hierarchy is obtained by uniform refinement, where each tetrahedron is refined into eight new tetrahedra using the midpoints of edges in the initial tetrahedron. In Figures 2.6–2.8, we present the decay of the relative energy norm of the algebraic error and the relative ℓ^2-norm of the residual vector with respect to the iterations. Even in three space dimensions, in accordance with our theory, we see that the results are p-robust and in agreement with the more in-depth experiments of two space dimensions. Moreover, similarly to the previous tests in two space dimensions, we numerically observe that the behavior of our solver is not influenced by the magnitude of the diffusion coefficient jump. The implementation of the experiments in this section is done with NGSolve, Schöberl [2014].

9.4 Comparison with solvers from literature

In Table 2.5, we compare our solver of Definition 4.1 (denoted as ∼MG(0,1)-bJ due to the similarity with the multigrid using only one post-smoothing step by block-Jacobi, the only difference being the use of the optimal step-size per level in

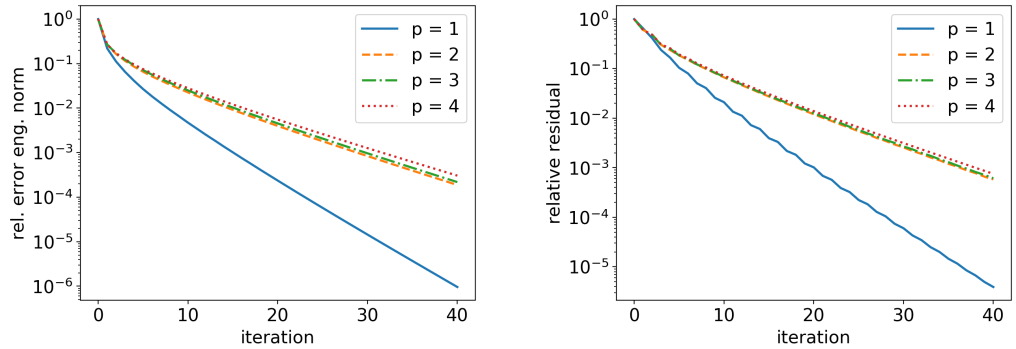


Figure 2.6: Cube case: decay of the relative algebraic error (left) and the relative residual (right) for the hierarchy with $p_j = 1$, $j \in \{1, \ldots, J-1\}$, $J = 4$. The solver of Definition 4.1 is stopped at iteration $i = 40$. nDoFs: 5 501 for $p = 1$, 41 337 for $p = 2$, 136 693 for $p = 3$, 320 753 for $p = 4$.

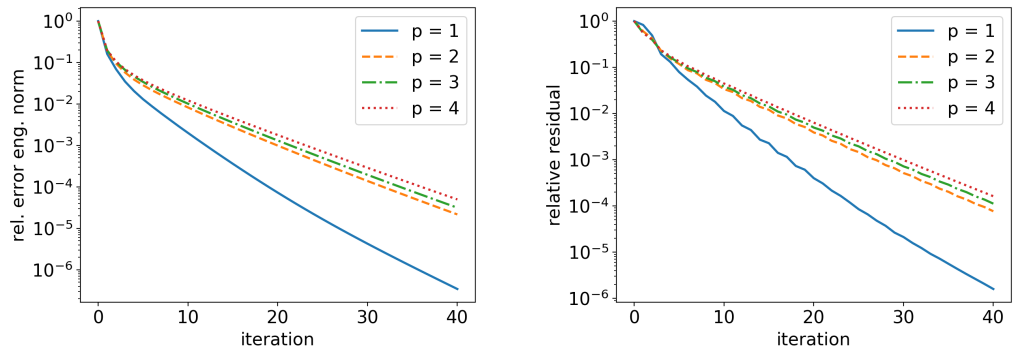


Figure 2.7: Nested cubes case: decay of the relative algebraic error (left) and the relative residual (right) for the hierarchy with $p_j = 1$, $j \in \{1, \ldots, J-1\}$, $J = 4$. The solver of Definition 4.1 is stopped at iteration $i = 40$. nDoFs: 7 281 for $p = 1$, 55 649 for $p = 2$, 185 041 for $p = 3$, 435 393 for $p = 4$.

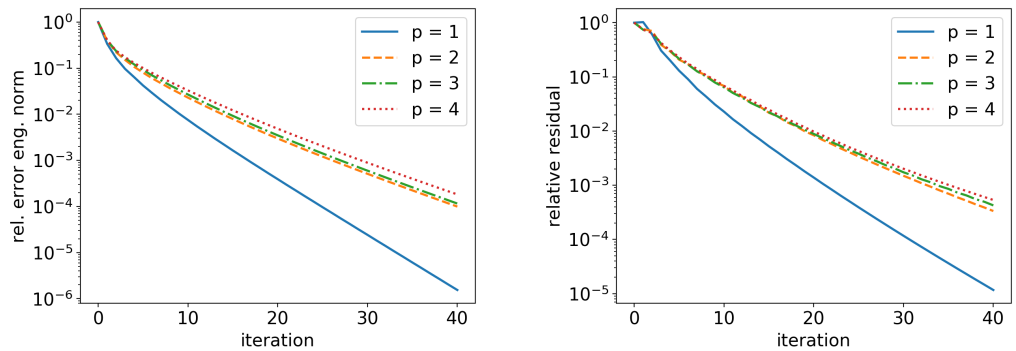


Figure 2.8: Checkers cubes: decay of the relative algebraic error (left) and the relative residual (right) for the hierarchy with $p_j = 1$, $j \in \{1, \ldots, J-1\}$, $J = 4$. The solver of Definition 4.1 is stopped at iteration $i = 40$. nDoFs: 5 425 for $p = 1$, 40 033 for $p = 2$, 131 473 for $p = 3$, 307 393 for $p = 4$.

the error correction stage), with different multigrid solvers used in the literature as solvers or preconditioners, see Miraçi et al. [2020], Chapter 1, for a more detailed discussion on these methods. The test case we choose has a poor regularity, and as we see both in terms of CPU timing[1] and iteration numbers, our solver performs well comparing with the other methods, despite the more simplistic one post-smoothing step, while having the advantage of being naturally parallelizable on each level as the smoother is block-Jacobi. Importantly, note that other block smoothing methods, namely PCG(MG-bJ) which uses a symmetric multigrid with block-Jacobi smoothing as preconditioner, and MG-bGS, the multigrid using block Gauss–Seidel as smoother, also exhibit numerical p-robustness, whereas the classical MG-GS does not. In addition to the solver of Definition 7.1 with $\theta = 0.2$ and $\nu_{\max} = 5$ (denoted as ∼MG(0,adapt)-bJ), we also introduce its weighted restrictive additive Schwarz (wRAS) smoother variant, which outperforms the other methods while preserving numerical p-robustness. Smoothing by wRAS, see details in [Miraçi et al., 2020, Section 6.2], see Section 6.2 in Chapter 1, only differs from the additive Schwarz smoothing used in Definitions 4.1 and 7.1 by summing in (2.15) local contributions $\rho^i_{j,\mathbf{a}}$ weighted by the corresponding hat functions $\psi_{j,\mathbf{a}}$ and then interpolated to the local spaces $V^{\mathbf{a}}_j$. Another important observation, as proven in e.g. Bramble et al. [1991], is that multigrid methods can perform robustly with respect to the diffusion contrast in two space dimensions. This is reflected by most methods of Table 2.5, having rather low iteration numbers for all diffusion tensors.

		∼MG(0,1)-bJ $1 \to 1, p$		∼MG(0,1)-bJ $1, p \to p$		∼MG(0,adapt)-bJ $1, p \to p$		∼MG(0,adapt)-bJ (wRAS) $1 \nearrow p$		PCG(MG(3,3)-bJ) $p \to p$		MG(1,1)-PCG(iChol) $1 \nearrow p$		MG(0,1)-bGS $1 \to 1, p$		MG(3,3)-GS $1 \nearrow p$	
J	p	i_s	time	i_s	time	i_s	time	i_s	time	i_s	time	i_s	time	i_s	time	i_s	time
3	1	18	0.05 s	18	0.07 s	8	0.04 s	8	0.04 s	10	0.07 s	6	0.39 s	10	0.04 s	4	0.02 s
	3	28	0.96 s	11	0.50 s	6	0.43 s	6	0.41 s	3	0.57 s	22	3.43 s	11	2.62 s	6	0.34 s
	6	25	9.88 s	10	5.43 s	6	5.24 s	5	2.90 s	2	5.24 s	44	51.38 s	9	7.35 s	11	5.91 s
	9	23	45.87 s	9	27.01 s	6	25.25 s	4	13.86 s	2	36.95 s	>80	>5.22m	8	32.53 s	11	19.72 s
4	1	19	0.12 s	19	0.12 s	9	0.11 s	9	0.11 s	11	0.20 s	16	0.74 s	11	0.06 s	4	0.05 s
	3	27	3.85 s	11	2.07 s	6	1.89 s	7	1.62 s	3	2.34 s	44	27.48 s	10	9.64 s	5	1.37 s
	6	24	41.79 s	9	20.19 s	6	20.69 s	4	12.54 s	3	38.40 s	>80	>6.87m	9	34.78 s	6	14.44 s
	9	23	3.63m	9	2.13m	6	2.09m	3	49.84 s	2	2.24m	>80	>23.08m	8	1.72m	9	1.21m

Table 2.5: Checkerboard $O(10^6)$ problem: comparison of iteration numbers i_s and CPU times for different solvers. The horizontal/rising arrow denotes whether the polynomial degree per level remains the same/gradually increases. The number of pre- and post-smoothing steps are given in parantheses, and the smoothers are given by block-Jacobi (bJ), block Gauss–Seidel (bGS), pointwise Gauss–Seidel (GS), or PCG with incomplete Cholesky preconditioner. The number of iterations is limited to 80.

[1] The codes were prepared to benefit as much as possible from Matlab's fast operations on matrices and vectors. The timings cover the solution time only, without the preparation phase of matrices assembly. The experiments were run on one **Dell C6220** dual-Xeon E5-2650 node of Inria Sophia Antipolis - Méditerranée "NEF" computation cluster, in a sequential Matlab script.

10 Proof of Theorem 6.6

Our approach to proving Theorem 6.6 consists in studying level-wise the contributions $\tilde{\rho}^i_j$ of (2.10) of the uncomputable exact residual lifting $\tilde{\rho}^i_{J,\mathrm{alg}}$ given by (2.8). The polynomial-degree-robust stable decomposition result of Schöberl et al. [2008] then allows us to exploit the similarities of the local computable contributions $\rho^i_{j,\mathbf{a}}$ (2.14) to the global inaccessible ones $\tilde{\rho}^i_j$ (2.10).

We will first present the proof of p-robust efficiency of the estimator stated in Theorem 6.6 under Assumption 6.2 or 6.3. Then we give the proof of p-robust and J-robust efficiency under Assumption 6.4. Let us start with some generalities.

10.1 Properties of the estimator η^i_{alg}

We first present some general properties of the estimator η^i_{alg} of Definition (5.1) needed for the proof.

Lemma 10.1 (Estimation of $\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho^i_j\|$ by the local contributions)**.** *Let $\rho^i_{j,\mathbf{a}}$ and ρ^i_j for $j\in\{1,\dots,J\}$, $\mathbf{a}\in\mathcal{V}_j$, be given by (2.14) and (2.15). Then there holds*

$$\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho^i_j\|^2 \le (d+1)\sum_{\mathbf{a}\in\mathcal{V}_j}\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho^i_{j,\mathbf{a}}\|^2_{\omega^{\mathbf{a}}_j}. \tag{2.42}$$

Proof. Since $\rho^i_j=\sum_{\mathbf{a}\in\mathcal{V}_j}\rho^i_{j,\mathbf{a}}$, the inequality $\left|\sum_{k=1}^{d+1}a_k\right|^2\le(d+1)\sum_{k=1}^{d+1}|a_k|^2$ leads to

$$\begin{aligned}\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho^i_j\|^2&=\sum_{K\in\mathcal{T}_j}\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho^i_j\|^2_K=\sum_{K\in\mathcal{T}_j}\Big\|\sum_{\mathbf{a}\in\mathcal{V}_K}\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho^i_{j,\mathbf{a}}\Big\|^2_K\\&\le(d+1)\sum_{K\in\mathcal{T}_j}\sum_{\mathbf{a}\in\mathcal{V}_K}\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho^i_{j,\mathbf{a}}\|^2_K=(d+1)\sum_{\mathbf{a}\in\mathcal{V}_j}\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho^i_{j,\mathbf{a}}\|^2_{\omega^{\mathbf{a}}_j}.\end{aligned}$$

□

Remark 10.2 (Lower bound on the optimal step-sizes)**.** *Note that (2.21) together with (2.42) and the definition $\lambda^i_j=1$ when $\rho^i_j=0$ or $j=0$ lead to*

$$\lambda^i_j\ge\frac{1}{d+1}\quad 0\le j\le J. \tag{2.43}$$

10.2 Properties of the exact residual lifting $\tilde{\rho}^i_{J,\mathrm{alg}}$

Hereafter, we use two crucial properties of the level-wise error contributions of (2.10) $\tilde{\rho}^i_j$, $j\in\{1,\dots,J\}$: the orthogonality of $\tilde{\rho}^i_j$ with respect to previous levels and local properties of $\tilde{\rho}^i_j$ on level j.

Lemma 10.3 (Inter-level properties of $\tilde{\rho}^i_j$)**.** *Consider the hierarchical construction of the error $\tilde{\rho}^i_{J,\mathrm{alg}}$ given in (2.8). For $j\in\{1,\dots,J\}$, there holds*

$$(\boldsymbol{\mathcal{K}}\nabla\tilde{\rho}^i_j,\nabla v_k)=0\quad\forall v_k\in V^{p_k}_k,\quad 0\le k<j. \tag{2.44}$$

Proof. Take $v_k \in V_k^{p_k}$. Note that since $k \leq j-1$ and by nestedness of the spaces, we have $v_k \in V_{j-1}^{p_{j-1}} \subset V_j^{p_j}$. The definition given in (2.10) applied to $\tilde{\rho}_j^i$ and $\tilde{\rho}_{j-1}^i$, allows us to write

$$\begin{aligned}(\boldsymbol{\mathcal{K}}\nabla\tilde{\rho}_j^i, \nabla v_k) &= (f, v_k) - (\boldsymbol{\mathcal{K}}\nabla u_J^i, \nabla v_k) - \sum_{l=0}^{j-2}(\boldsymbol{\mathcal{K}}\nabla\tilde{\rho}_l^i, \nabla v_k) - (\boldsymbol{\mathcal{K}}\nabla\tilde{\rho}_{j-1}^i, \nabla v_k)\\ &= (\boldsymbol{\mathcal{K}}\nabla\tilde{\rho}_{j-1}^i, \nabla v_k) - (\boldsymbol{\mathcal{K}}\nabla\tilde{\rho}_{j-1}^i, \nabla v_k) = 0.\end{aligned}$$

□

Now, we present the relation between $\tilde{\rho}_j^i$ and ρ_j^i locally on patches, more precisely when tested against functions of the local spaces $V_j^{\mathbf{a}}$ given by (2.7).

Lemma 10.4 (Local relation between $\tilde{\rho}_j^i$ and $\rho_{j,\mathbf{a}}^i$)**.** *Let $j \in \{1, \dots, J\}$. Let $\tilde{\rho}_j^i$, $\rho_{j,\mathbf{a}}^i$, ρ_j^i be respectively given by (2.10), (2.14), and (2.15). For all vertices $\mathbf{a} \in \mathcal{V}_j$ and all functions $v_{j,\mathbf{a}} \in V_j^{\mathbf{a}}$, we have*

$$(\boldsymbol{\mathcal{K}}\nabla\tilde{\rho}_j^i, \nabla v_{j,\mathbf{a}})_{\omega_j^{\mathbf{a}}} = (\boldsymbol{\mathcal{K}}\nabla\rho_{j,\mathbf{a}}^i, \nabla v_{j,\mathbf{a}})_{\omega_j^{\mathbf{a}}} - \sum_{k=1}^{j-1}(\boldsymbol{\mathcal{K}}\nabla(\tilde{\rho}_k^i - \lambda_k^i\rho_k^i), \nabla v_{j,\mathbf{a}})_{\omega_j^{\mathbf{a}}}. \tag{2.45}$$

We use the convention that the sum in the relation above is zero when $j = 1$.

Proof. We take $v_{j,\mathbf{a}} \in V_j^{\mathbf{a}}$. This implies that $v_{j,\mathbf{a}}$ is zero on the boundary of the patch domain $\omega_j^{\mathbf{a}}$. Since $v_{j,\mathbf{a}} \in V_J^p$, we can use it as a test function in the definition of $\tilde{\rho}_j^i$ in (2.10) as well as in the definition of $\rho_{j,\mathbf{a}}^i$ in (2.14). We conclude by using (2.18) and subtracting the two following identities

$$\begin{aligned}(\boldsymbol{\mathcal{K}}\nabla\tilde{\rho}_j^i, \nabla v_{j,\mathbf{a}})_{\omega_j^{\mathbf{a}}} &= (f, v_{j,\mathbf{a}})_{\omega_j^{\mathbf{a}}} - (\boldsymbol{\mathcal{K}}\nabla u_J^i, \nabla v_{j,\mathbf{a}})_{\omega_j^{\mathbf{a}}} - \sum_{k=0}^{j-1}(\boldsymbol{\mathcal{K}}\nabla\tilde{\rho}_k^i, \nabla v_{j,\mathbf{a}})_{\omega_j^{\mathbf{a}}},\\ (\boldsymbol{\mathcal{K}}\nabla\rho_{j,\mathbf{a}}^i, \nabla v_{j,\mathbf{a}})_{\omega_j^{\mathbf{a}}} &= (f, v_{j,\mathbf{a}})_{\omega_j^{\mathbf{a}}} - (\boldsymbol{\mathcal{K}}\nabla u_J^i, \nabla v_{j,\mathbf{a}})_{\omega_j^{\mathbf{a}}} - \sum_{k=0}^{j-1}\lambda_k^i(\boldsymbol{\mathcal{K}}\nabla\rho_k^i, \nabla v_{j,\mathbf{a}})_{\omega_j^{\mathbf{a}}}.\end{aligned}$$

□

10.3 Proof of Theorem 6.6 under the minimal $H_0^1(\Omega)$-regularity assumption

We begin by presenting here a result given in [Miraçi et al., 2020, Proposition 7.6], see Proposition 7.6 in Chapter 1, obtained by a combination of a one-level p-robust stable decomposition proven in Schöberl et al. [2008] and a multilevel stable decomposition for piecewise linear functions given in Xu et al. [2009].

Lemma 10.5 (p-robust multilevel stable decomposition)**.** *Let $v_J \in V_J^p$. Under Assumption 6.1 and either of Assumptions 6.2 or Assumption 6.3, there exists a decomposition*

$$v_J = v_0 + \sum_{j=1}^{J}\sum_{\mathbf{a}\in\mathcal{V}_j} v_{j,\mathbf{a}}, \quad v_0 \in V_0^1,\ v_{j,\mathbf{a}} \in V_j^{\mathbf{a}}, \tag{2.46}$$

stable as

$$\|\nabla v_0\|^2 + \sum_{j=1}^{J} \sum_{\mathbf{a}\in\mathcal{V}_j} \|\nabla v_{j,\mathbf{a}}\|^2_{\omega_j^{\mathbf{a}}} \leq C_{\mathrm{S}}^2 \|\nabla v_J\|^2, \tag{2.47}$$

where $C_{\mathrm{S}} \geq 1$ only depends on the space dimension d, the mesh shape regularity parameter $\kappa_{\mathcal{T}}$, and on the maximum strength of refinement parameter C_{ref} and quasi-uniformity parameter C_{qu} when Assumption 6.2 is satisfied, or on the coarse and local quasi-uniformity parameters C_{qu}^0, $C_{\mathrm{loc,qu}}$ when Assumption 6.3 is satisfied.

The previous results and properties allow us now to give concise proofs.

Proof of Theorem 6.6. (p-robust estimator efficiency under Assumption 6.2 or 6.3). Note that by (2.13), we have $\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J - u_J^i)\right\| = \left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\tilde{\rho}^i_{J,\mathrm{alg}}\right\|$. Thus, we work with the exact algebraic residual lifting $\tilde{\rho}^i_{J,\mathrm{alg}}$. We begin by applying Lemma 10.5 to $\tilde{\rho}^i_{J,\mathrm{alg}}$, which allows to decompose it as

$$\tilde{\rho}^i_{J,\mathrm{alg}} = \tilde{c}_0^i + \sum_{j=1}^{J} \sum_{\mathbf{a}\in\mathcal{V}_j} \tilde{\rho}^i_{j,\mathbf{a}}, \quad \tilde{c}_0^i \in V_0^1,\ \tilde{\rho}^i_{j,\mathbf{a}} \in V_j^{\mathbf{a}}, \tag{2.48}$$

$$\|\nabla\tilde{c}_0^i\|^2 + \sum_{j=1}^{J} \sum_{\mathbf{a}\in\mathcal{V}_j} \|\nabla\tilde{\rho}^i_{j,\mathbf{a}}\|^2_{\omega_j^{\mathbf{a}}} \leq C_{\mathrm{S}}^2 \|\nabla\tilde{\rho}^i_{J,\mathrm{alg}}\|^2. \tag{2.49}$$

Taking into account the variations of the diffusion coefficient $\boldsymbol{\mathcal{K}}$, we have

$$\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\tilde{c}_0^i\|^2 + \sum_{j=1}^{J} \sum_{\mathbf{a}\in\mathcal{V}_j} \left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\tilde{\rho}^i_{j,\mathbf{a}}\right\|^2_{\omega_j^{\mathbf{a}}} \leq C_{\mathrm{S},\boldsymbol{\mathcal{K}}}^2 \left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\tilde{\rho}^i_{J,\mathrm{alg}}\right\|^2, \tag{2.50}$$

where the constant $C_{\mathrm{S},\boldsymbol{\mathcal{K}}}^2$ additionally depends on the ratio of the largest and the smallest eigenvalue of the diffusion coefficient $\boldsymbol{\mathcal{K}}$. Since $\max\left(1, C_{\mathrm{S},\boldsymbol{\mathcal{K}}}^2\right)$ also satisfies (2.50), we can assume $C_{\mathrm{S},\boldsymbol{\mathcal{K}}} \geq 1$. We use this decomposition to develop

$$\begin{aligned}
&\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\tilde{\rho}^i_{J,\mathrm{alg}}\right\|^2 \\
&\overset{(2.48)}{=} \left(\boldsymbol{\mathcal{K}}\nabla\tilde{\rho}^i_{J,\mathrm{alg}}, \nabla\tilde{c}_0^i + \sum_{j=1}^{J}\sum_{\mathbf{a}\in\mathcal{V}_j} \nabla\tilde{\rho}^i_{j,\mathbf{a}}\right) \\
&\overset{(2.9)}{=} \left(\boldsymbol{\mathcal{K}}\nabla\rho_0^i, \nabla\tilde{c}_0^i\right) + \sum_{j=1}^{J}\sum_{\mathbf{a}\in\mathcal{V}_j} \left(\boldsymbol{\mathcal{K}}\nabla\tilde{\rho}^i_{J,\mathrm{alg}}, \nabla\tilde{\rho}^i_{j,\mathbf{a}}\right)_{\omega_j^{\mathbf{a}}} \\
&\overset{(1.13)}{=} \left(\boldsymbol{\mathcal{K}}\nabla\rho_0^i, \nabla\tilde{c}_0^i\right) + \sum_{j=1}^{J}\sum_{\mathbf{a}\in\mathcal{V}_j} \left(\left(f, \tilde{\rho}^i_{j,\mathbf{a}}\right)_{\omega_j^{\mathbf{a}}} - \left(\boldsymbol{\mathcal{K}}\nabla u_J^i, \nabla\tilde{\rho}^i_{j,\mathbf{a}}\right)_{\omega_j^{\mathbf{a}}}\right) \\
&\overset{(2.14)}{\overset{(2.18)}{=}} \left(\boldsymbol{\mathcal{K}}\nabla\rho_0^i, \nabla\tilde{c}_0^i\right) + \sum_{j=1}^{J}\sum_{\mathbf{a}\in\mathcal{V}_j} \left(\left(\boldsymbol{\mathcal{K}}\nabla\rho^i_{j,\mathbf{a}}, \nabla\tilde{\rho}^i_{j,\mathbf{a}}\right)_{\omega_j^{\mathbf{a}}} + \sum_{k=0}^{j-1}\left(\lambda_k^i\boldsymbol{\mathcal{K}}\nabla\rho_k^i, \nabla\tilde{\rho}^i_{j,\mathbf{a}}\right)_{\omega_j^{\mathbf{a}}}\right) \\
&= \left(\boldsymbol{\mathcal{K}}\nabla\rho_0^i, \nabla\tilde{c}_0^i\right) + \sum_{j=1}^{J}\sum_{\mathbf{a}\in\mathcal{V}_j} \left(\boldsymbol{\mathcal{K}}\nabla\rho^i_{j,\mathbf{a}}, \nabla\tilde{\rho}^i_{j,\mathbf{a}}\right)_{\omega_j^{\mathbf{a}}} + \sum_{j=1}^{J}\sum_{k=0}^{j-1}\left(\lambda_k^i\boldsymbol{\mathcal{K}}\nabla\rho_k^i, \sum_{\mathbf{a}\in\mathcal{V}_j}\nabla\tilde{\rho}^i_{j,\mathbf{a}}\right).
\end{aligned}$$

We will now estimate each of the above three terms using Young's inequality and patch overlap arguments as done in the proof of Lemma 10.1. First, we have, using the fact that $\lambda_0^i = 1$,

$$\left(\boldsymbol{\mathcal{K}}\nabla\rho_0^i, \nabla\tilde{c}_0^i\right) \leq \frac{C_{\mathrm{S},\boldsymbol{\mathcal{K}}}^2}{2}\left(\lambda_0^i\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_0^i\|\right)^2 + \frac{1}{2C_{\mathrm{S},\boldsymbol{\mathcal{K}}}^2}\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\tilde{c}_0^i\|^2.$$

For the second term, we similarly obtain

$$\begin{aligned}
&\sum_{j=1}^{J}\sum_{\mathbf{a}\in\mathcal{V}_j}\left(\boldsymbol{\mathcal{K}}\nabla\rho_{j,\mathbf{a}}^i, \nabla\tilde{\rho}_{j,\mathbf{a}}^i\right)_{\omega_j^{\mathbf{a}}} \\
&\leq C_{\mathrm{S},\boldsymbol{\mathcal{K}}}^2\sum_{j=1}^{J}\sum_{\mathbf{a}\in\mathcal{V}_j}\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_{j,\mathbf{a}}^i\|_{\omega_j^{\mathbf{a}}}^2 + \frac{1}{4C_{\mathrm{S},\boldsymbol{\mathcal{K}}}^2}\sum_{j=1}^{J}\sum_{\mathbf{a}\in\mathcal{V}_j}\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\tilde{\rho}_{j,\mathbf{a}}^i\|_{\omega_j^{\mathbf{a}}}^2 \\
&\overset{\substack{(2.21)\\(2.43)}}{\leq} C_{\mathrm{S},\boldsymbol{\mathcal{K}}}^2(d+1)\sum_{j=1}^{J}\left(\lambda_j^i\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_j^i\|\right)^2 + \frac{1}{4C_{\mathrm{S},\boldsymbol{\mathcal{K}}}^2}\sum_{j=1}^{J}\sum_{\mathbf{a}\in\mathcal{V}_j}\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\tilde{\rho}_{j,\mathbf{a}}^i\|_{\omega_j^{\mathbf{a}}}^2.
\end{aligned}$$

Finally, for the third term, we have

$$\begin{aligned}
&\sum_{j=1}^{J}\sum_{k=0}^{j-1}\left(\lambda_k^i\boldsymbol{\mathcal{K}}\nabla\rho_k^i, \sum_{\mathbf{a}\in\mathcal{V}_j}\nabla\tilde{\rho}_{j,\mathbf{a}}^i\right) \\
&\leq \frac{2(d+1)C_{\mathrm{S},\boldsymbol{\mathcal{K}}}^2 J}{2}\sum_{j=1}^{J}\sum_{k=0}^{j-1}\left(\lambda_k^i\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_k^i\|\right)^2 + \frac{\sum_{j=1}^{J}\sum_{k=0}^{j-1}\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\sum_{\mathbf{a}\in\mathcal{V}_j}\nabla\tilde{\rho}_{j,\mathbf{a}}^i\right\|^2}{2(2(d+1)C_{\mathrm{S},\boldsymbol{\mathcal{K}}}^2 J)} \\
&\leq (d+1)C_{\mathrm{S},\boldsymbol{\mathcal{K}}}^2 J^2\sum_{k=0}^{J}\left(\lambda_k^i\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_k^i\|\right)^2 + \frac{1}{4C_{\mathrm{S},\boldsymbol{\mathcal{K}}}^2}\sum_{j=1}^{J}\sum_{\mathbf{a}\in\mathcal{V}_j}\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\tilde{\rho}_{j,\mathbf{a}}^i\|_{\omega_j^{\mathbf{a}}}^2.
\end{aligned}$$

Summing these components together, we can now pursue our main estimate

$$\begin{aligned}
&\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\tilde{\rho}_{J,\mathrm{alg}}^i\|^2 \\
&\leq 2(d+1)C_{\mathrm{S},\boldsymbol{\mathcal{K}}}^2 J^2\sum_{j=0}^{J}\left(\lambda_j^i\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_j^i\|\right)^2 + \frac{1}{2C_{\mathrm{S}}^2\boldsymbol{\mathcal{K}}}\left(\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\tilde{c}_0^i\|^2 + \sum_{j=1}^{J}\sum_{\mathbf{a}\in\mathcal{V}_j}\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\tilde{\rho}_{j,\mathbf{a}}^i\|_{\omega_j^{\mathbf{a}}}^2\right) \\
&\overset{\substack{(2.23)\\(2.50)}}{\leq} 2(d+1)C_{\mathrm{S},\boldsymbol{\mathcal{K}}}^2 J^2\left(\eta_{\mathrm{alg}}^i\right)^2 + \frac{1}{2}\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\tilde{\rho}_{J,\mathrm{alg}}^i\|^2.
\end{aligned}$$

After subtracting $\frac{1}{2}\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\tilde{\rho}_{J,\mathrm{alg}}^i\|^2$ on both sides, we finally obtain the desired result

$$\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\tilde{\rho}_{J,\mathrm{alg}}^i\|^2 \leq 4(d+1)C_{\mathrm{S},\boldsymbol{\mathcal{K}}}^2 J^2\left(\eta_{\mathrm{alg}}^i\right)^2. \tag{2.51}$$

□

10.4 Proof of Theorem 6.6 under the $H^2(\Omega)$-regularity assumption

Under Assumption 6.4, we now prove that the result of Theorem 6.6 holds not only p-robustly but also J-robustly. For this, we exhibit a different level-wise stable

decomposition from that of Section 10.3. Here, we will define the piecewise linear component of the stable decomposition via a H^1-orthogonal projection and then use a duality-type argument.

Definition 10.6 (H^1-orthogonal lowest-order projection of error components)**.** *For any $j \in \{1, \dots, J\}$, let $\tilde{\rho}_j^i$ be given by (2.10). Then let $c_j^i \in V_j^1$ be the solution of*

$$(\nabla c_j^i, \nabla v_j) = (\nabla \tilde{\rho}_j^i, \nabla v_j) \quad \forall v_j \in V_j^1. \tag{2.52}$$

Remark 10.7 (Orthogonality properties of c_j^i)**.** *For any $j \in \{1, \dots, J\}$, c_j^i satisfies the following orthogonality with piecewise affine functions of previous levels*

$$(\nabla c_j^i, \nabla v_k) \overset{(2.52)}{=} (\nabla \tilde{\rho}_j^i, \nabla v_k) \overset{(2.44)}{=} 0, \quad \forall v_k \in V_k^1, \quad \forall 0 \le k < j. \tag{2.53}$$

Lemma 10.8 (H^2-regularity result)**.** *Under Assumption 6.4, for any c_j^i given by Definition 10.6, $j \in \{1, \dots, J\}$, there holds*

$$\|\tilde{\rho}_j^i\| \le \frac{C_{\text{app}}}{C_{\text{qu}} C_{\text{ref}}} h_j \|\nabla \tilde{\rho}_j^i\|, \tag{2.54}$$

$$\|c_j^i\| \le \frac{C_{\text{app}}}{C_{\text{qu}} C_{\text{ref}}} h_j \|\nabla c_j^i\|, \tag{2.55}$$

where the constant C_{app} depends on the space dimension d and the mesh shape regularity parameter $\kappa_{\mathcal{T}}$, and C_{qu} and C_{ref} are the quasi-uniformity and refinement strength parameters from Assumption 6.2.

Proof. To prove the first result, we proceed by a standard duality argument. We consider the following problem: find $\xi_j \in H_0^1(\Omega)$ such that

$$(\nabla \xi_j, \nabla v) = (\tilde{\rho}_j^i, v) \quad \forall v \in H_0^1(\Omega). \tag{2.56}$$

Following [Grisvard, 1985, Theorem 4.3.1.4], under Assumption 6.4, $\xi_j \in H^2(\Omega)$, and we have

$$|\xi_j|_{H^2(\Omega)} = \|\Delta \xi_j\| = \|\tilde{\rho}_j^i\|. \tag{2.57}$$

Consider $\mathrm{I}_{j-1}^1(\xi_j)$ the $\mathbb{P}^1$-Lagrange interpolation of ξ_j on mesh level $j-1$. Since $\xi_j \in H^2(\Omega)$, following, e.g., [Ern and Guermond, 2004, Corollary 1.110], we obtain

$$\|\nabla(\xi_j - \mathrm{I}_{j-1}^1(\xi_j))\| \le C_{\text{app}} h_{j-1} |\xi_j|_{H^2(\Omega)}. \tag{2.58}$$

In particular: $\mathrm{I}_{j-1}^1(\xi_j) \in V_{j-1}^{p_{j-1}}$, so by the orthogonality relation (2.44)

$$(\nabla \mathrm{I}_{j-1}^1(\xi_j), \nabla \tilde{\rho}_j^i) = 0. \tag{2.59}$$

We have now all the elements to conclude

$$\|\tilde{\rho}_j^i\|^2 \overset{(2.56)}{=} (\nabla \xi_j, \nabla \tilde{\rho}_j^i) \overset{(2.59)}{=} (\nabla(\xi_j - \mathrm{I}_{j-1}^1(\xi_j)), \nabla \tilde{\rho}_j^i) \le \|\nabla(\xi_j - \mathrm{I}_{j-1}^1(\xi_j))\| \|\nabla \tilde{\rho}_j^i\|$$
$$\overset{(2.58)}{\le} C_{\text{app}} h_{j-1} |\xi_j|_{H^2(\Omega)} \|\nabla \tilde{\rho}_j^i\| \overset{(2.57)}{=} C_{\text{app}} h_{j-1} \|\tilde{\rho}_j^i\| \|\nabla \tilde{\rho}_j^i\| \overset{\substack{(2.26)\\(2.27)}}{\le} \frac{C_{\text{app}}}{C_{\text{qu}} C_{\text{ref}}} h_j \|\tilde{\rho}_j^i\| \|\nabla \tilde{\rho}_j^i\|,$$

which gives us (2.54). To obtain (2.55), the same argument is used once the right-hand side of the dual problem (2.56) is modified to (c_j^i, v), and we replace the orthogonality relation (2.59) by (2.53). Note that at this point, it is important that $\mathrm{I}_{j-1}^1(\xi_j) \in V_{j-1}^1$. □

We can now present the stable decomposition used in the proof of Theorem 6.6.

Lemma 10.9 (Stable decomposition of the error level-wise components)**.** *For $\tilde{\rho}_j^i$ given by (2.10), c_j^i given by Definition 10.6, $j \in \{1, ..., J\}$, there exist $\tilde{\tilde{\rho}}_{j,\mathbf{a}}^i \in V_j^{\mathbf{a}}$, so that*

$$\tilde{\rho}_j^i = c_j^i + \sum_{\mathbf{a}\in\mathcal{V}_j} \tilde{\tilde{\rho}}_{j,\mathbf{a}}^i, \tag{2.60}$$

$$\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla c_j^i\right\|^2 + \sum_{\mathbf{a}\in\mathcal{V}_j} \left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\tilde{\tilde{\rho}}_{j,\mathbf{a}}^i\right\|_{\omega_j^{\mathbf{a}}}^2 \leq C_{\mathrm{SD},\boldsymbol{\mathcal{K}}}^2 \|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\tilde{\rho}_j^i\|^2, \tag{2.61}$$

where $C_{\mathrm{SD},\boldsymbol{\mathcal{K}}}^2 \geq 1$ only depends on the space dimension d, the mesh shape regularity parameter $\kappa_{\mathcal{T}}$, the quasi-uniformity parameter C_{qu}, the strength refinement parameter C_{ref}, and the ratio of the largest and the smallest eigenvalue of the diffusion coefficient $\boldsymbol{\mathcal{K}}$.

Proof. We now rely on the stable decomposition result of Schöberl et al. [2008] for one-level setting. We will first show, as in [Schöberl et al., 2008, Lemma 3.1], that the coarse contribution c_j^i satisfies

$$\|\nabla c_j^i\|^2 + \|\nabla(\tilde{\rho}_j^i - c_j^i)\|^2 + \sum_{K\in\mathcal{T}_j} h_K^{-2}\|(\tilde{\rho}_j^i - c_j^i)\|_K^2 \leq \left(5 + \left(\frac{2C_{\mathrm{app}}}{C_{\mathrm{ref}}C_{\mathrm{qu}}^2}\right)^2\right)\|\nabla\tilde{\rho}_j^i\|^2. \tag{2.62}$$

Then, one can construct local contributions $\tilde{\tilde{\rho}}_{j,\mathbf{a}}^i \in V_j^{\mathbf{a}}$ as in [Schöberl et al., 2008, Section 3], which by [Schöberl et al., 2008, Proof of Theorem 2.1] gives us

$$\|\nabla c_j^i\|^2 + \sum_{\mathbf{a}\in\mathcal{V}_j} \|\nabla\tilde{\tilde{\rho}}_{j,\mathbf{a}}^i\|_{\omega_j^{\mathbf{a}}}^2 \leq C_{\mathrm{SD}}^2 \|\nabla\tilde{\rho}_j^i\|^2,$$

and the claim (2.61) follows by taking into consideration the variations of $\boldsymbol{\mathcal{K}}$.

To show (2.62), we first use Definition 10.6 of c_j^i

$$\|\nabla c_j^i\|^2 = (\nabla c_j^i, \nabla c_j^i) \overset{(2.52)}{=} (\nabla c_j^i, \nabla\tilde{\rho}_j^i) \leq \|\nabla c_j^i\|\|\nabla\tilde{\rho}_j^i\|. \tag{2.63}$$

This allows to estimate the first and second term (after using the triangle inequality) of (2.62). The third term is then estimated by

$$\sum_{K\in\mathcal{T}_j} h_K^{-2}\|(\tilde{\rho}_j^i - c_j^i)\|_K^2 \overset{(2.27)}{\leq} C_{\mathrm{qu}}^{-2} h_j^{-2} \sum_{K\in\mathcal{T}_j} \|(\tilde{\rho}_j^i - c_j^i)\|_K^2 \leq 2C_{\mathrm{qu}}^{-2} h_j^{-2}\left(\|\tilde{\rho}_j^i\|^2 + \|c_j^i\|^2\right)$$
$$\overset{\substack{(2.54)\\(2.55)}}{\leq} 2\left(\frac{C_{\mathrm{app}}}{C_{\mathrm{ref}}C_{\mathrm{qu}}^2}\right)^2 (\|\nabla\tilde{\rho}_j^i\|^2 + \|\nabla c_j^i)\|^2) \overset{(2.63)}{\leq} 4\left(\frac{C_{\mathrm{app}}}{C_{\mathrm{ref}}C_{\mathrm{qu}}^2}\right)^2 \|\nabla\tilde{\rho}_j^i\|^2.$$

□

Remark 10.10 (Localized writing of level-wise components)**.** *Note that since $\tilde{\rho}_j^i = c_j^i + \sum_{\mathbf{a}\in\mathcal{V}_j} \tilde{\tilde{\rho}}_{j,\mathbf{a}}^i$, we can decompose the piecewise linear $c_j^i \in V_j^1$ using the nodal basis functions. We can then write*

$$\tilde{\rho}_j^i = c_j^i + \sum_{\mathbf{a}\in\mathcal{V}_j} \tilde{\tilde{\rho}}_{j,\mathbf{a}}^i = \sum_{\mathbf{a}\in\mathcal{V}_j} (c_{j,\mathbf{a}}^i \psi_{j,\mathbf{a}} + \tilde{\tilde{\rho}}_{j,\mathbf{a}}^i), \tag{2.64}$$

where $c_{j,\mathbf{a}}^i$ is the nodal value on vertex $\mathbf{a} \in \mathcal{V}_j$ of c_j^i, and $c_{j,\mathbf{a}}^i \psi_{j,\mathbf{a}} + \tilde{\tilde{\rho}}_{j,\mathbf{a}}^i \in V_j^{\mathbf{a}}$.

Lemma 10.11 (L^2-stability of nodal decomposition)**.** *For all $j \in \{1,\dots,J\}$ and all $v_j \in V_j^1$ decomposed into the hat functions $v_j = \sum_{\mathbf{a}\in\mathcal{V}_j} v_{j,\mathbf{a}}\psi_{j,\mathbf{a}}$, we have*

$$\|v_j\|^2 \le (d+1)\sum_{\mathbf{a}\in\mathcal{V}_j}\|v_{j,\mathbf{a}}\psi_{j,\mathbf{a}}\|^2_{\omega_j^{\mathbf{a}}}, \quad \text{and} \quad \sum_{\mathbf{a}\in\mathcal{V}_j}\|v_{j,\mathbf{a}}\psi_{j,\mathbf{a}}\|^2_{\omega_j^{\mathbf{a}}} \le C_{\mathrm{nd}}^2\|v_j\|^2, \tag{2.65}$$

where $C_{\mathrm{nd}} \ge 1$ only depends on the space dimension d and the mesh shape regularity parameter $\kappa_{\mathcal{T}}$.

Proof. For the first estimate, we apply the usual overlapping argument as done for (2.42). As for the second estimate, consider a patch $\omega_j^{\mathbf{a}}$ and element K contained in the patch. Since $v_j \in V_j^1$ and by mesh shape regularity and equivalence of norms in finite dimension, we have

$$\begin{aligned}\|v_{j,\mathbf{a}}\psi_{j,\mathbf{a}}\|_{\omega_j^{\mathbf{a}}} &\le C_{\kappa_{\mathcal{T}}}\|v_{j,\mathbf{a}}\psi_{j,\mathbf{a}}\|_K \le C_{\kappa_{\mathcal{T}}}\|v_{j,\mathbf{a}}\psi_{j,\mathbf{a}}\|_\infty |K|^{\frac12}\\ &\le C_{\kappa_{\mathcal{T}}}\Big\|\sum_{\mathbf{a}\in\mathcal{V}_K} v_{j,\mathbf{a}}\psi_{j,\mathbf{a}}\Big\|_\infty |K|^{\frac12} = C_{\kappa_{\mathcal{T}}}\|v_j\|_{\infty,K}|K|^{\frac12} \le C_{\kappa_{\mathcal{T}}}\widetilde{C}_{\kappa_{\mathcal{T}}}\|v_j\|_K,\end{aligned}$$

where $C_{\kappa_{\mathcal{T}}} \ge 1$ and $\widetilde{C}_{\kappa_{\mathcal{T}}} \ge 1$ only depend on the mesh shape regularity parameter $\kappa_{\mathcal{T}}$. The result is obtained by summing both sides over all vertices. □

Lemma 10.12 (Level-wise estimation of c_j^i)**.** *Let $j \in \{1,\dots,J\}$ and let $c_j^i = \sum_{\mathbf{a}\in\mathcal{V}_j} c_{j,\mathbf{a}}^i\psi_{j,\mathbf{a}}$ be given by Definition 10.6. Then there holds*

$$\sum_{\mathbf{a}\in\mathcal{V}_j}\big\|\boldsymbol{\mathcal{K}}^{\frac12}c_{j,\mathbf{a}}^i\nabla\psi_{j,\mathbf{a}}\big\|^2_{\omega_j^{\mathbf{a}}} \le C^2_{\mathrm{stab},\boldsymbol{\mathcal{K}}}\big\|\boldsymbol{\mathcal{K}}^{\frac12}\nabla c_j^i\big\|^2, \tag{2.66}$$

where $C_{\mathrm{stab},\boldsymbol{\mathcal{K}}} \ge 1$ only depends on the space dimension d, the mesh shape regularity parameter $\kappa_{\mathcal{T}}$, the quasi-uniformity parameter C_{qu}, the strength refinement parameter C_{ref}, and the ratio of the largest and the smallest eigenvalue of the diffusion coefficient $\boldsymbol{\mathcal{K}}$.

Proof. We start by using an inverse inequality, denoting by $h_{\omega_j^{\mathbf{a}}}$ the diameter of patch $\omega_j^{\mathbf{a}}$ and then use the quasi-uniformity assumption (2.27)

$$\begin{aligned}&\sum_{\mathbf{a}\in\mathcal{V}_j}\|\boldsymbol{\mathcal{K}}^{\frac12}c_{j,\mathbf{a}}^i\nabla\psi_{j,\mathbf{a}}\|^2_{\omega_j^{\mathbf{a}}} \le C^2_{\boldsymbol{\mathcal{K}}}\sum_{\mathbf{a}\in\mathcal{V}_j}\|c_{j,\mathbf{a}}^i\nabla\psi_{j,\mathbf{a}}\|^2_{\omega_j^{\mathbf{a}}} \le C^2_{\boldsymbol{\mathcal{K}}}C^2_{\mathrm{inv}}\sum_{\mathbf{a}\in\mathcal{V}_j}h^{-2}_{\omega_j^{\mathbf{a}}}\|c_{j,\mathbf{a}}^i\psi_{j,\mathbf{a}}\|^2_{\omega_j^{\mathbf{a}}}\\ &\le C^2_{\boldsymbol{\mathcal{K}}}C^{-2}_{\mathrm{qu}}C^2_{\mathrm{inv}}h_j^{-2}\sum_{\mathbf{a}\in\mathcal{V}_j}\|c_{j,\mathbf{a}}^i\psi_{j,\mathbf{a}}\|^2_{\omega_j^{\mathbf{a}}} \overset{(2.65)}{\le} C^2_{\boldsymbol{\mathcal{K}}}C^{-2}_{\mathrm{qu}}C^2_{\mathrm{inv}}C^2_{\mathrm{nd}}h_j^{-2}\|c_j^i\|^2\\ &\overset{(2.55)}{\le} \frac{C^2_{\boldsymbol{\mathcal{K}}}C^2_{\mathrm{inv}}C^2_{\mathrm{nd}}C^2_{\mathrm{app}}}{C^4_{\mathrm{qu}}C^2_{\mathrm{ref}}}\|\nabla c_j^i\|^2 \le \frac{C^2_{\boldsymbol{\mathcal{K}}}C^2_{\mathrm{inv}}C^2_{\mathrm{nd}}C^2_{\mathrm{app}}}{C^4_{\mathrm{qu}}C^2_{\mathrm{ref}}c^2_{\boldsymbol{\mathcal{K}}}}\|\boldsymbol{\mathcal{K}}^{\frac12}\nabla c_j^i\|^2 = C^2_{\mathrm{stab},\boldsymbol{\mathcal{K}}}\|\boldsymbol{\mathcal{K}}^{\frac12}\nabla c_j^i\|^2,\end{aligned}$$

where $c^2_{\boldsymbol{\mathcal{K}}}$, $C^2_{\boldsymbol{\mathcal{K}}}$ are respectively constants that depend on the smallest and the largest eigenvalue of the diffusion coefficient $\boldsymbol{\mathcal{K}}$. Note that the resulting constant $C_{\mathrm{stab},\boldsymbol{\mathcal{K}}}$ can be safely assumed to be greater than 1, otherwise replace it with $\max(1, C_{\mathrm{stab},\boldsymbol{\mathcal{K}}})$. □

Lemma 10.13 (p-robust level-wise error estimation)**.** *Let $j \in \{1,\dots,J\}$ and let $\tilde\rho_j^i$ and ρ_j^i be defined by (2.10) and (2.15), respectively. Then there holds*

$$\big\|\boldsymbol{\mathcal{K}}^{\frac12}\nabla\tilde\rho_j^i\big\|^2 \le 2C^2_{\mathrm{SD},\boldsymbol{\mathcal{K}}}C^2_{\mathrm{stab},\boldsymbol{\mathcal{K}}}\sum_{\mathbf{a}\in\mathcal{V}_j}\big\|\boldsymbol{\mathcal{K}}^{\frac12}\nabla\rho_{j,\mathbf{a}}^i\big\|^2_{\omega_j^{\mathbf{a}}}. \tag{2.67}$$

Proof. We begin by using the splitting (2.60) in the form (2.64), which gives

$$
\begin{aligned}
\big\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\tilde{\rho}^i_j\big\|^2 &\overset{(2.64)}{=} \sum_{\mathbf{a}\in\mathcal{V}_j}\big(\boldsymbol{\mathcal{K}}\nabla\tilde{\rho}^i_j,\nabla(c^i_{j,\mathbf{a}}\psi_{j,\mathbf{a}}+\tilde{\tilde{\rho}}^i_{j,\mathbf{a}})\big)_{\omega^{\mathbf{a}}_j}\\
&\overset{(2.45)}{=}\sum_{\mathbf{a}\in\mathcal{V}_j}\Big(\big(\boldsymbol{\mathcal{K}}\nabla\rho^i_{j,\mathbf{a}},\nabla(c^i_{j,\mathbf{a}}\psi_{j,\mathbf{a}}+\tilde{\tilde{\rho}}^i_{j,\mathbf{a}})\big)_{\omega^{\mathbf{a}}_j}-\sum_{k=1}^{j-1}\big(\boldsymbol{\mathcal{K}}\nabla(\tilde{\rho}^i_k-\lambda^i_k\rho^i_k),\nabla(c^i_{j,\mathbf{a}}\psi_{j,\mathbf{a}}+\tilde{\tilde{\rho}}^i_{j,\mathbf{a}})\big)_{\omega^{\mathbf{a}}_j}\Big)\\
&\overset{(2.64)}{=}\sum_{\mathbf{a}\in\mathcal{V}_j}\big(\boldsymbol{\mathcal{K}}\nabla\rho^i_{j,\mathbf{a}},\nabla(c^i_{j,\mathbf{a}}\psi_{j,\mathbf{a}}+\tilde{\tilde{\rho}}^i_{j,\mathbf{a}})\big)_{\omega^{\mathbf{a}}_j}-\sum_{k=1}^{j-1}\big(\boldsymbol{\mathcal{K}}\nabla(\tilde{\rho}^i_k-\lambda^i_k\rho^i_k),\nabla\tilde{\rho}^i_j\big)\\
&\overset{(2.44)}{=}\sum_{\mathbf{a}\in\mathcal{V}_j}\big(\boldsymbol{\mathcal{K}}\nabla\rho^i_{j,\mathbf{a}},\nabla\big(c^i_{j,\mathbf{a}}\psi_{j,\mathbf{a}}+\tilde{\tilde{\rho}}^i_{j,\mathbf{a}}\big)\big)_{\omega^{\mathbf{a}}_j}-0\\
&\le C^2_{\mathrm{SD},\boldsymbol{\mathcal{K}}}C^2_{\mathrm{stab},\boldsymbol{\mathcal{K}}}\sum_{\mathbf{a}\in\mathcal{V}_j}\big\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho^i_{j,\mathbf{a}}\big\|^2_{\omega^{\mathbf{a}}_j}+\frac{\sum_{\mathbf{a}\in\mathcal{V}_j}\big\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(c^i_{j,\mathbf{a}}\psi_{j,\mathbf{a}}+\tilde{\tilde{\rho}}^i_{j,\mathbf{a}})\big\|^2_{\omega^{\mathbf{a}}_j}}{4C^2_{\mathrm{SD},\boldsymbol{\mathcal{K}}}C^2_{\mathrm{stab},\boldsymbol{\mathcal{K}}}}\\
&\le C^2_{\mathrm{SD},\boldsymbol{\mathcal{K}}}C^2_{\mathrm{stab},\boldsymbol{\mathcal{K}}}\sum_{\mathbf{a}\in\mathcal{V}_j}\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho^i_{j,\mathbf{a}}\|^2_{\omega^{\mathbf{a}}_j}+\frac{\sum_{\mathbf{a}\in\mathcal{V}_j}\big(\big\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}c^i_{j,\mathbf{a}}\nabla\psi_{j,\mathbf{a}}\big\|^2_{\omega^{\mathbf{a}}_j}+\big\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\tilde{\tilde{\rho}}^i_{j,\mathbf{a}}\big\|^2_{\omega^{\mathbf{a}}_j}\big)}{2C^2_{\mathrm{SD},\boldsymbol{\mathcal{K}}}C^2_{\mathrm{stab},\boldsymbol{\mathcal{K}}}}\\
&\overset{(2.66)}{\le} C^2_{\mathrm{SD},\boldsymbol{\mathcal{K}}}C^2_{\mathrm{stab},\boldsymbol{\mathcal{K}}}\sum_{\mathbf{a}\in\mathcal{V}_j}\big\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho^i_{j,\mathbf{a}}\big\|^2_{\omega^{\mathbf{a}}_j}+\frac{C^2_{\mathrm{stab},\boldsymbol{\mathcal{K}}}\big\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla c^i_j\big\|^2+\sum_{\mathbf{a}\in\mathcal{V}_j}\big\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\tilde{\tilde{\rho}}^i_{j,\mathbf{a}}\big\|^2_{\omega^{\mathbf{a}}_j}}{2C^2_{\mathrm{SD},\boldsymbol{\mathcal{K}}}C^2_{\mathrm{stab},\boldsymbol{\mathcal{K}}}}\\
&\overset{(2.61)}{\le} C^2_{\mathrm{SD},\boldsymbol{\mathcal{K}}}C^2_{\mathrm{stab},\boldsymbol{\mathcal{K}}}\sum_{\mathbf{a}\in\mathcal{V}_j}\big\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho^i_{j,\mathbf{a}}\big\|^2_{\omega^{\mathbf{a}}_j}+\frac{1}{2}\big\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\tilde{\rho}^i_j\big\|^2,
\end{aligned}
$$

which leads to the assertion (2.67). □

We can now give a concise proof of Theorem 6.6.

Proof of Theorem 6.6. (p- and J-robust estimator efficiency under Assumption 6.4) To estimate the algebraic error, we use the level-wise decomposition (2.13). Each level's contribution was estimated in Lemma 10.13. Summing over different levels,

$$
\begin{aligned}
\big\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J-u^i_J)\big\|^2 &\overset{(2.13)}{=}\sum_{j=0}^{J}\big\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\tilde{\rho}^i_j\big\|^2\\
&\overset{(2.67)}{\le}\big\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho^i_0\big\|^2+2C^2_{\mathrm{SD},\boldsymbol{\mathcal{K}}}C^2_{\mathrm{stab},\boldsymbol{\mathcal{K}}}\sum_{j=1}^{J}\sum_{\mathbf{a}\in\mathcal{V}_j}\big\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho^i_{j,\mathbf{a}}\big\|^2_{\omega^{\mathbf{a}}_j}\\
&\overset{(2.21)}{\underset{(2.43)}{\le}} 2C^2_{\mathrm{SD},\boldsymbol{\mathcal{K}}}C^2_{\mathrm{stab},\boldsymbol{\mathcal{K}}}(d+1)\Big(\big(\lambda^i_0\big\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho^i_0\big\|\big)^2+\sum_{j=1}^{J}\big(\lambda^i_j\big\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho^i_j\big\|\big)^2\Big)\\
&\overset{(2.23)}{=} 2C^2_{\mathrm{SD},\boldsymbol{\mathcal{K}}}C^2_{\mathrm{stab},\boldsymbol{\mathcal{K}}}(d+1)\big(\eta^i_{\mathrm{alg}}\big)^2.
\end{aligned}
$$

Thus we have showed $\eta^i_{\mathrm{alg}}\ge\beta\big\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J-u^i_J)\big\|$ for $\beta:=\frac{1}{\sqrt{2(d+1)}C_{\mathrm{SD},\boldsymbol{\mathcal{K}}}C_{\mathrm{stab},\boldsymbol{\mathcal{K}}}}>0$. □

11 Conclusions and future work

In this work we presented a multilevel algebraic solver whose construction is inherently interconnected with an a posteriori estimator of the algebraic error. The solver can be seen as a geometric multigrid relying on V-cycles with zero pre- and one post-smoothing, where the smoother is additive Schwarz associated to patches of elements (block-Jacobi). A crucial difference compared to classic multigrid solvers is the use of an optimal step-size in the error correction stage on each level of the mesh hierarchy. This significantly improves the behavior of the solver and conveniently enough, makes the analysis easier leading in particular to the Pythagorean error decrease formula (2.22a). We also presented a simple and efficient way for the solver to automatically increase the number of post-smoothing steps on each level to the amount needed, based on the a posteriori estimator of the algebraic error. We showed that the non-adaptive version of the solver (with only one post-smoothing step) contracts the error in each iteration robustly with respect to the polynomial degree p of the underlying finite element discretization; this result is equivalent to showing p-robust efficiency of the a posteriori error estimate. If we additionally assume H^2-regularity in the sense of Assumption 6.4, we can show that these results are also robust with respect to the number of mesh levels J. An interesting side property is that the error estimator is equivalent to a sum of level- and patchwise-localized computable contributions by formula (2.31). Future work will explore how to incorporate this information in the solver so that it adaptively tackles only problematic regions contributing most to the algebraic error (local smoothing). Finally, numerical results indicate that even for singular test cases, the solver behaves robustly with respect to the polynomial degree p, number of levels J, as well as the diffusion coefficient $\mathcal{K}$ (for quasi-uniform meshes).

Chapter 3

Contractive local adaptive smoothing based on Dörfler's marking in a-posteriori-steered p-robust multigrid solvers

We present in this chapter the results of the article Miraçi et al. [2021b], *Computational Methods in Applied Mathematics*, DOI 10.1515/cmam-2020-0024, written with Jan Papež and Martin Vohralík.

Contents

1 Introduction **87**

2 Setting **89**

2.1 Model problem and its finite element discretization 89

2.2 A hierarchy of meshes and spaces 89

3 Adaptive multilevel solver **90**

3.1 Algorithmic description of the solver 91

3.2 Mathematical description of the solver 93

4 A posteriori estimator on the algebraic error **96**

5 Main results **97**

5.1 Mesh assumptions 97

5.2 Main result 98

5.3 Additional results 99

6 Numerical experiments **99**

6.1 Can we predict the distribution of the algebraic error? 100

6.2 Does the adaptivity pay off? 101

6.3 Dependence on the marking parameter 105

7 Proofs of the main results **107**

7.1 Proof of contraction: full-smoothing substep 107

7.2 Proof of contraction: adaptive-smoothing substep 109

8 Conclusions **112**

Abstract

In this work, we study a local adaptive smoothing algorithm for a-posteriori-steered p-robust multigrid methods. The solver tackles a linear system which is generated by the discretization of a second-order elliptic diffusion problem using conforming finite elements of polynomial order $p \geq 1$. After one V-cycle ("full-smoothing" substep) of the solver of Miraçi, Papež, and Vohralík [*SIAM J. Sci. Comput.* DOI 10.1137/20M1349503], we dispose of a reliable, efficient, and localized estimation of the algebraic error. We use this existing result to develop our new adaptive algorithm: thanks to the information of the estimator and based on a bulk-chasing criterion, cf. Dörfler [SIAM J. Numer. Anal., 33 (1996), pp. 1106–1124], we mark patches of elements with increased estimated error on all levels. Then, we proceed by a modified and cheaper V-cycle ("adaptive-smoothing" substep), which only applies smoothing in the marked regions. The proposed adaptive multigrid solver picks autonomously and adaptively the optimal step-size per level as in our previous work but also the type of smoothing per level (weighted restricted additive or additive Schwarz) and concentrates smoothing to marked regions with high error. We prove that, under a numerical condition that we verify in the algorithm, each substep (full and adaptive) contracts the error p-robustly, which is confirmed by numerical experiments. Moreover, the proposed algorithm behaves numerically robustly with respect to the number of levels as well as to the diffusion coefficient jump for a uniformly-refined hierarchy of meshes.

1 Introduction

The finite element method is a widespread and versatile discretization method for partial differential equations, see e.g. Ciarlet [1978], Ern and Guermond [2004], or Brenner and Scott [2008]. In particular, the use of high-order methods has shown numerous advantages in terms of accuracy, see e.g. Szabó and Babuška [1991], Bernardi and Maday [1997], Šolín et al. [2004], and the references therein. The implementation of these methods, however, leads to a linear system that is abundantly bigger than for low-order discretizations. Moreover, since the conditioning degrades with increasing order, commonly used solvers begin to suffer. Amongst the most efficient solvers we mention multigrid solvers, see e.g. Hackbusch [2003], Briggs et al. [2000], more generally multilevel methods e.g. Zhang [1992], Oswald [1994], Griebel and Oswald [1995], and the closely related domain decomposition methods, e.g. Quarteroni and Valli [1999] or Dolean et al. [2015]. Note that the above methods can be used in their own right as iterative solvers, or as a preconditioner (possibly after making them symmetric).

The idea of defining an *adaptive* algebraic solver is rather old. On the subject of local smoothing methods, we refer, e.g., to Bai and Brandt [1987], McCormick [1989], Rüde [1993], Lötzbeyer and Rüde [1997], and more recently Xu et al. [2009], Janssen and Kanschat [2011], or Chen et al. [2012]. Here, the smoothing is typically localized to parts where the adaptive mesh refinement was performed (to newly added elements only), but it is not adaptive per se. Adaptive smoothed aggregation aiming at building a coarser linear system by determining near-kernel components was proposed in the context of algebraic multigrid, see e.g. Brezina et al. [2006]

and the references therein. More recently, an aggregation based on path covers was proposed by Hu et al. [2019]. Another interesting approach consists in applying an adaptive construction of preconditioners, see, e.g., the recent work of Anciaux-Sedrakian et al. [2020], where the adaptivity relies on a posteriori error estimates of the algebraic error, cf. Papež et al. [2020, 2018], combined with a bulk-chasing criterion in the spirit of Dörfler [1996]. To the best of the authors' knowledge, this is the first time a bulk-chasing criterion is used in an algebraic solver adaptivity (and not mesh refinement) setting. However, the results in Anciaux-Sedrakian et al. [2020] are mainly numerical, whereas mathematical analysis is not really developed.

The subject of this work is to propose a multigrid solver with *local adaptive smoothing* based on rigorous a posteriori error estimates of the algebraic error and a bulk-chasing criterion, and to prove its convergence. We rely on the polynomial-degree-robust solver introduced in Miraçi et al. [2021a], see Chapter 2, which is a geometric multigrid whose iteration consists of a V-cycle with zero pre- and one post-smoothing step, where the smoothing is overlapping additive Schwarz (block-Jacobi) associated to patches of elements. This solver already contains a first adaptive step, since the error correction update from one level to the next, in contrast to a standard multigrid, picks the optimal (adaptive) step-size that reduces the algebraic error in the best possible way. The results of Miraçi et al. [2021a] also give us a *reliable* and *efficient a posteriori estimator* on the algebraic error and an equivalence of the algebraic error with *localized* (levelwise/patchwise) computable estimators that serve as a starting point for our current contribution.

In this work, after implementing one step of the original solver of Miraçi et al. [2021a] (one full-smoothing V-cycle), we obtain a fairly good indication of where (levelwise/patchwise) the algebraic error is concentrated. We then use a bulk-chasing criterion to mark the highest contribution patches, and then perform a cheaper step (one adaptive-smoothing V-cycle) only smoothing in these problematic regions. Additionally, based on numerical performance and literature results, see, e.g., Cai and Sarkis [1999], Efstathiou and Gander [2003], or Loisel et al. [2008], we give the solver the option to pick adaptively the type of smoothing, be it additive Schwarz or (the typically better performing) weighted restricted additive Schwarz. We focus on quasi-uniform meshes, but our theory also applies to possibly highly graded bisection grids.

We prove that the presented algorithm contracts the error in *each of the sub-steps*, the full-smoothing and the adaptive-smoothing, *robustly* with respect to the *polynomial degree* p of the underlying finite element discretization. The results on the full-smoothing substep rely on Miraçi et al. [2021a], where a p-robust stable decomposition for one level by Schöberl et al. [2008], and a multilevel stable decomposition for piecewise affine polynomials on quasi-uniform/bisection grids by Xu et al. [2009] are crucial. Numerically, for a hierarchy of meshes obtained through uniform refinement, we additionally observe robustness with respect to the number of levels in the mesh hierarchy as well as the jumps in the diffusion coefficient.

Compared to Miraçi et al. [2021a], the novelties of this work are: 1) Development of a new kind of adaptivity that is local in patches with increased algebraic error, whereas the adaptivity in Miraçi et al. [2021a] chooses the number of post-smoothing steps globally per level. 2) Localization in space relying on Dörfler's marking. 3) Proof that the new adaptive sub-step contracts the error p-robustly, despite it

only smoothes in marked patches provided that a numerical condition is verified (no convergence proof of the adaptive scheme is given in Miraçi et al. [2021a]). 4) Adaptive decision on which smoothing (additive Schwarz or weighted restricted additive Schwarz) variant to employ per level and inclusion of the weighted restricted additive Schwarz in the analysis, which was not done in Miraçi et al. [2021a].

The manuscript is organized as follows. In Section 2, we introduce the model problem and the notation we will be working with. Section 3 presents in detail the algorithmic description of the solver with each of its modules, as well as the rigorous mathematical definition of the solver. In Section 4, we define the algebraic error estimator. The main results are collected in Section 5, and the numerical tests are showcased in Section 6. Section 7 gives the proofs of our main results. Finally, some concluding remarks are given in Section 8.

2 Setting

In this section we present the model problem we will be studying and the notation needed for the multilevel setting we work with.

2.1 Model problem and its finite element discretization

We work with a second-order elliptic problem defined over $\Omega \subset \mathbb{R}^d$, $d \in \{1, 2, 3\}$, an open bounded polytope with a Lipschitz-continuous boundary. In the weak formulation, we search for $u \in H_0^1(\Omega)$ such that

$$(\boldsymbol{\mathcal{K}} \nabla u, \nabla v) = (f, v) \quad \forall v \in H_0^1(\Omega), \tag{3.1}$$

where $f \in L^2(\Omega)$ is a source term and $\boldsymbol{\mathcal{K}} \in [L^\infty(\Omega)]^{d \times d}$ is a symmetric and positive definite diffusion coefficient.

Let $\mathcal{T}_J$ be a matching simplicial mesh of Ω. Fixing an integer $p \geq 1$, we introduce the finite element space of piecewise continuous polynomials of degree p

$$V_J^p := \mathbb{P}_p(\mathcal{T}_J) \cap H_0^1(\Omega), \tag{3.2}$$

where $\mathbb{P}_p(\mathcal{T}_J) := \{v_J \in L^2(\Omega), v_J|_K \in \mathbb{P}_p(K) \ \forall K \in \mathcal{T}_J\}$. The discrete problem consists in finding $u_J \in V_J^p$ such that

$$(\boldsymbol{\mathcal{K}} \nabla u_J, \nabla v_J) = (f, v_J) \quad \forall v_J \in V_J^p. \tag{3.3}$$

2.2 A hierarchy of meshes and spaces

We rely in this contribution on a hierarchy of matching simplicial meshes $\{\mathcal{T}_j\}_{0 \leq j \leq J}$, $J \geq 1$, where $\mathcal{T}_J$ has been introduced in Section 2.1, and where $\mathcal{T}_j$ is a refinement of $\mathcal{T}_{j-1}$, $1 \leq j \leq J$. We also introduce a hierarchy of finite element spaces associated to the mesh hierarchy. For this purpose, fix p_j, the polynomial degree associated to mesh level $j \in \{1, \ldots, J\}$, such that $1 = p_0 \leq p_1 \leq \ldots \leq p_{J-1} \leq p_J = p$. We then introduce

$$\text{for } j = 0: \qquad V_0^1 := \mathbb{P}_1(\mathcal{T}_0) \cap H_0^1(\Omega) \quad \text{(lowest-order space)}, \tag{3.4a}$$

$$\text{for } 1 \le j \le J-1: \qquad V_j^{p_j} := \mathbb{P}_{p_j}(\mathcal{T}_j) \cap H_0^1(\Omega) \quad (p_j\text{-th order spaces}), \tag{3.4b}$$

where $\mathbb{P}_{p_j}(\mathcal{T}_j) := \{v_j \in L^2(\Omega), v_j|_K \in \mathbb{P}_{p_j}(K) \quad \forall K \in \mathcal{T}_j\}$. Note that $V_0^1 \subset V_1^{p_1} \subset \ldots \subset V_{J-1}^{p_{J-1}} \subset V_J^p$, so that the spaces are nested. Let $\mathcal{V}_j$ be the set of vertices of the mesh $\mathcal{T}_j$. We denote by $\psi_{j,\mathbf{a}}$ the standard hat function associated to the vertex $\mathbf{a} \in \mathcal{V}_j$, $0 \le j \le J$; this is the piecewise affine function with respect to the mesh $\mathcal{T}_j$ that takes value 1 in the vertex $\mathbf{a}$ and vanishes in all other vertices of $\mathcal{V}_j$.

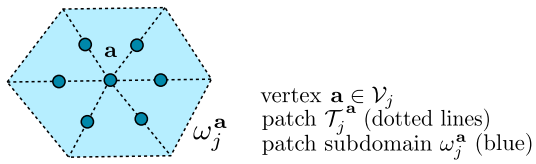


Figure 3.1: Illustration of a patch $\mathcal{T}_j^{\mathbf{a}}$, the patch subdomain $\omega_j^{\mathbf{a}}$, and of the degrees of freedom for the space $V_j^{\mathbf{a}}$ with $p_j = 2$.

For the following, we need to define the notion of patches of elements, illustrated in Figure 3.1. Let $j \in \{1, \ldots, J\}$. For any element $K \in \mathcal{T}_j$, we denote by $\mathcal{V}_K$ the set of its vertices. Then, given an arbitrary vertex $\mathbf{a} \in \mathcal{V}_j$, we denote by $\mathcal{T}_j^{\mathbf{a}}$ the patch formed by all elements of the mesh $\mathcal{T}_j^{\mathbf{a}}$ sharing the vertex $\mathbf{a}$, i.e., $\mathcal{T}_j^{\mathbf{a}} := \{K \in \mathcal{T}_j, \mathbf{a} \in \mathcal{V}_K\}$. Then we denote by $\omega_j^{\mathbf{a}}$ the open patch subdomain corresponding to $\mathcal{T}_j^{\mathbf{a}}$. Finally, the associated local space $V_j^{\mathbf{a}}$ is defined by

$$V_j^{\mathbf{a}} := \mathbb{P}_{p_j}(\mathcal{T}_j) \cap H_0^1(\omega_j^{\mathbf{a}}), \; j \in \{1, \ldots, J\}. \tag{3.5}$$

Larger subdomains can also be considered, cf. Miraçi et al. [2020]. Finally, denote by $\mathcal{I}_j^{p_j}$ the $\mathbb{P}^{p_j}$ Lagrange interpolation operator on the mesh level j, i.e. $\mathcal{I}_j^{p_j} : C^0(\overline{\Omega}) \to V_j^{p_j}$, $\mathcal{I}_j^{p_j}(v)$ preseves the values of v in the nodes corresponding to the Lagrange degrees of freedom. This will play an important role in the adaptive choice of smoothing of the solver presented below in Section 3.

3 Adaptive multilevel solver

The basic idea of our adaptive solver is illustrated in Figure 3.2. In Section 3.1, we give an algorithmic description of the solver, followed by the explanation of its constituting modules. Then in Section 3.2, we provide a mathematical description of the solver, lengthier but better suited for the forthcoming theoretical analysis.

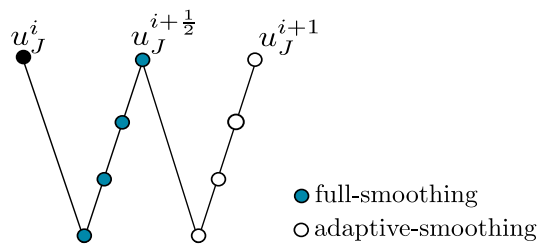


Figure 3.2: Illustration of the full-smoothing and adaptive-smoothing V-cycle substeps, $J = 3$.

3.1 Algorithmic description of the solver

The adaptive solver we propose can be written in an algorithmic description:

Algorithm 1: A-posteriori-steered multigrid with local adaptive smoothing

Input: [polynomial degrees $\{p_j\}_{0\le j\le J}$, mesh hierarchy $\{\mathcal{T}_j\}_{0\le j\le J}$, bulk-chasing parameter θ, adaptivity-decision parameter γ, requested tolerance tol]

```
i := 0; u_J^i := 0; η_alg^i := 10tol;
while η_alg^i ≥ tol do
    i := i + 1; u_J^i := u_J^{i-1};
    ρ_0^i := COARSE_SOLVE; u_J^i := u_J^i + ρ_0^i; (η_alg^i)^2 := ‖K^{1/2} ∇ρ_0^i‖^2;
    for j = 1, ..., J do
        for a ∈ V_j do
            ρ_{j,a}^i := LOCAL_SOLVE(j, a);
        end
        ρ_j^i :=ADAPT_SMOOTH(j, V_j); λ_j^i := OPTIMAL_STEPSIZE(ρ_j^i);
        u_J^i := u_J^i + λ_j^i ρ_j^i; (η_alg^i)^2 := (η_alg^i)^2 + (λ_j^i ‖K^{1/2} ∇ρ_j^i‖)^2;
    end
    if η_alg^i < tol break while loop;
    (M, {a ∈ M_j}_{j∈M}) := DÖRFLER_MARKING(ρ_0^i, {{ρ_{j,a}^i}_{j=1}^J}_{a∈V_j}, θ);
    if [ TEST_ADAPT(γ) ] then
        if 0 ∈ M then
            ρ_0^i := COARSE_SOLVE; u_J^i := u_J^i + ρ_0^i ; (η_alg^i)^2:= ‖K^{1/2} ∇ρ_0^i‖^2;
        end
        for j ∈ M \ {0} do
            for a ∈ M_j do
                ρ_{j,a}^i := LOCAL_SOLVE(j, a);
            end
            ρ_j^i := ADAPT_SMOOTH(j, M_j); λ_j^i := OPTIMAL_STEPSIZE(ρ_j^i);
            u_J^i := u_J^i + λ_j^i ρ_j^i; (η_alg^i)^2 := (η_alg^i)^2 + (λ_j^i ‖K^{1/2} ∇ρ_j^i‖)^2;
        end
    end
end
i_stop := i;
```

Output: $[\, u_J^{i_{\text{stop}}},\ \eta_{\text{alg}}^{i_{\text{stop}}} \,]$

3.1.1 Module COARSE_SOLVE (coarse grid residual solve)

Input: - ; Output: global $\mathbb{P}_1$-lifting ρ_0^i of the current algebraic residual.

Given the latest approximation $u_J^i \in V_J^p$, define $\rho_0^i \in V_0^1$ by

$$(\mathcal{K}\nabla\rho_0^i, \nabla v_0) = (f, v_0) - (\mathcal{K}\nabla u_J^i, \nabla v_0) \quad \forall v_0 \in V_0^1.$$

3.1.2 Module `LOCAL_SOLVE` (block-Jacobi smoother)

Input: level j, vertex $\mathbf{a}$; Output: local $\mathbb{P}_{p_j}$-lifting $\rho^i_{j,\mathbf{a}}$ of the current algebraic residual.

Given the latest approximation $u^i_J \in V^p_J$, define the local contribution $\rho^i_{j,\mathbf{a}} \in V^{\mathbf{a}}_j$ by

$$(\boldsymbol{\mathcal{K}}\nabla\rho^i_{j,\mathbf{a}}, \nabla v_{j,\mathbf{a}})_{\omega^{\mathbf{a}}_j} = (f, v_{j,\mathbf{a}})_{\omega^{\mathbf{a}}_j} - (\boldsymbol{\mathcal{K}}\nabla u^i_J, \nabla v_{j,\mathbf{a}})_{\omega^{\mathbf{a}}_j} \quad \forall v_{j,\mathbf{a}} \in V^{\mathbf{a}}_j.$$

3.1.3 Module `ADAPT_SMOOTH` (descent direction)

Input: level j, set of vertices $\mathcal{V}(j)$; Output: descent direction ρ^i_j.

The following test verifies if the weighted restricted additive Schwarz smoothing is compatible with the convergence analysis of the solver.

Given the latest approximation $u^i_J \in V^p_J$, if the two following conditions hold

- $\displaystyle\sum_{\mathbf{a}\in\mathcal{V}(j)} \mathcal{I}^{p_j}_j(\psi_{j,\mathbf{a}}\rho^i_{j,\mathbf{a}}) \neq 0,$
- $\displaystyle\left(\frac{\sum_{\mathbf{a}\in\mathcal{V}(j)} \left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho^i_{j,\mathbf{a}}\right\|^2_{\omega^{\mathbf{a}}_j}}{d+1}\right)^{\frac{1}{2}} \leq \frac{\sum_{\mathbf{a}\in\mathcal{V}(j)} \left((f, \mathcal{I}^{p_j}_j(\psi_{j,\mathbf{a}}\rho^i_{j,\mathbf{a}}))_{\omega^{\mathbf{a}}_j} - (\boldsymbol{\mathcal{K}}\nabla u^i_J, \nabla\mathcal{I}^{p_j}_j(\psi_{j,\mathbf{a}}\rho^i_{j,\mathbf{a}}))_{\omega^{\mathbf{a}}_j}\right)}{\left\|\sum_{\mathbf{a}\in\mathcal{V}(j)} \boldsymbol{\mathcal{K}}\nabla\mathcal{I}^{p_j}_j(\psi_{j,\mathbf{a}}\rho^i_{j,\mathbf{a}})\right\|},$

and, if the module is in the full-smoothing substep additionally, if

- $\displaystyle\sum_{\mathbf{a}\in\mathcal{V}(j)} \left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\mathcal{I}^{p_j}_j(\psi_{j,\mathbf{a}}\rho^i_{j,\mathbf{a}})\right\|^2_{\omega^{\mathbf{a}}_j} \leq \sum_{\mathbf{a}\in\mathcal{V}(j)} \left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho^i_{j,\mathbf{a}}\right\|^2_{\omega^{\mathbf{a}}_j},$

then the solver employs weighted restricted additive Schwarz smoothing, by defining the descent direction on level j, $\rho^i_j \in V^{p_j}_j$, as

$$\rho^i_j := \sum_{\mathbf{a}\in\mathcal{V}(j)} \mathcal{I}^{p_j}_j(\psi_{j,\mathbf{a}}\rho^i_{j,\mathbf{a}}).$$

Otherwise, additive Schwarz smoothing is employed and

$$\rho^i_j := \sum_{\mathbf{a}\in\mathcal{V}(j)} \rho^i_{j,\mathbf{a}}.$$

3.1.4 Module `OPTIMAL_STEPSIZE` (optimal level step-size)

Input: descent direction ρ^i_j on level j; Output: optimal step-size λ^i_j on level j.

Given the latest approximation $u^i_J \in V^p_J$, if $\rho^i_j = 0$, set $\lambda^i_j := 1$, otherwise define the optimal step-size on level j, as

$$\lambda^i_j := \frac{(f, \rho^i_j) - (\boldsymbol{\mathcal{K}}\nabla u^i_J, \nabla\rho^i_j)}{\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho^i_j\right\|^2}.$$

3.1.5 Module DÖRFLER_MARKING (bulk choice of levels/patches for smoothing)

Input: liftings ρ_0^i, $\rho_{j,\mathbf{a}}^i$ for $1 \le j \le J$, $\mathbf{a} \in \mathcal{V}_j$, bulk-chasing parameter θ;
Output: set of marked levels $\mathcal{M}$, set of marked vertices per level $\mathcal{M}_j$, $j \in \mathcal{M}$.

For $\theta \in (0,1)$, we sort all patchwise contributions on all levels and select for marking the smallest cardinality set of the coarsest level and vertex indices, $1 \le j \le J$, by the following bulk-chasing criterion, cf. Dörfler [1996],

$$\theta^2 \left(\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \rho_0^i\right\|^2 + \sum_{j=1}^{J} \lambda_j^i \sum_{\mathbf{a}\in\mathcal{V}_j} \left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \rho_{j,\mathbf{a}}^i\right\|_{\omega_j^{\mathbf{a}}}^2 \right) \le \sum_{j\in\mathcal{M}} \lambda_j^i \sum_{\mathbf{a}\in\mathcal{M}_j} \left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \rho_{j,\mathbf{a}}^i\right\|_{\omega_j^{\mathbf{a}}}^2,$$

where only $\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \rho_0^i\right\|^2$ appears on the coarsest level $j = 0$ if it is marked, $0 \in \mathcal{M}$.

Here and below, we will always use the shorthand notation "$j \in \mathcal{M}$" for accessing the set $\mathcal{M}$ in ascending order.

3.1.6 Module TEST_ADAPT (deciding whether adaptivity will pay-off)

Input: User-prescribed parameter γ; Output: bool.

For $\gamma \in (0,1)$, if the following (analysis-driven) conditions hold, the solver will proceed to the adaptive-smoothing substep.

- $\displaystyle\sum_{j\in\mathcal{M}} \lambda_j^i \sum_{\mathbf{a}\in\mathcal{M}_j} \left(\sum_{k=j}^{J} \lambda_k^i \boldsymbol{\mathcal{K}} \nabla \rho_k^i, \nabla \rho_{j,\mathbf{a}}^i \right)_{\omega_j^{\mathbf{a}}} \le \gamma^2 \sum_{j\in\mathcal{M}} \lambda_j^i \sum_{\mathbf{a}\in\mathcal{M}_j} \left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \rho_{j,\mathbf{a}}^i\right\|_{\omega_j^{\mathbf{a}}}^2,$
- $\lambda_j^i \le 2(d+1) \quad \forall j \in \{0,\dots,J\}$.

In practice, one needs to verify the first condition, whereas the second one seems always satisfied.

3.2 Mathematical description of the solver

We now present the adaptive solver in a rigorous mathematical notation. This notation will be used for the remainder of the manuscript. Below, we describe in detail one iteration of the adaptive solver. The initialization is given by $u_J^0 := 0 \in V_J^p$.

1. **Full-smoothing substep**

   (a) Define $\rho_0^i \in V_0^1$ by

   $$(\boldsymbol{\mathcal{K}} \nabla \rho_0^i, \nabla v_0) = (f, v_0) - (\boldsymbol{\mathcal{K}} \nabla u_J^i, \nabla v_0) \quad \forall v_0 \in V_0^1 \tag{3.6}$$

   and set $\lambda_0^i := 1$ and $u_{J,0}^i := u_J^i + \lambda_0^i \rho_0^i$.

   (b) For all $j \in \{1,\dots,J\}$, $\mathbf{a} \in \mathcal{V}_j$, define the local contributions $\rho_{j,\mathbf{a}}^i \in V_j^{\mathbf{a}}$ by

   $$(\boldsymbol{\mathcal{K}} \nabla \rho_{j,\mathbf{a}}^i, \nabla v_{j,\mathbf{a}})_{\omega_j^{\mathbf{a}}} = (f, v_{j,\mathbf{a}})_{\omega_j^{\mathbf{a}}} - (\boldsymbol{\mathcal{K}} \nabla u_{J,j-1}^i, \nabla v_{j,\mathbf{a}})_{\omega_j^{\mathbf{a}}} \quad \forall v_{j,\mathbf{a}} \in V_j^{\mathbf{a}}. \tag{3.7}$$

   i. **Test (adaptive smoothing choice)**: If the following conditions hold

$$\sum_{\mathbf{a}\in\mathcal{V}_j} \mathcal{I}_j^{p_j}\left(\psi_{j,\mathbf{a}}\rho_{j,\mathbf{a}}^i\right) \neq 0, \tag{3.8a}$$

$$\left(\frac{\sum_{\mathbf{a}\in\mathcal{V}_j}\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_{j,\mathbf{a}}^i\right\|_{\omega_j^{\mathbf{a}}}^2}{d+1}\right)^{\frac{1}{2}} \leq \frac{\sum_{\mathbf{a}\in\mathcal{V}_j}\left[\left(f,\mathcal{I}_j^{p_j}\left(\psi_{j,\mathbf{a}}\rho_{j,\mathbf{a}}^i\right)\right)_{\omega_j^{\mathbf{a}}} - \left(\boldsymbol{\mathcal{K}}\nabla u_{J,j-1}^i,\nabla\mathcal{I}_j^{p_j}\left(\psi_{j,\mathbf{a}}\rho_{j,\mathbf{a}}^i\right)\right)_{\omega_j^{\mathbf{a}}}\right]}{\left\|\sum_{\mathbf{a}\in\mathcal{V}_j}\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\mathcal{I}_j^{p_j}\left(\psi_{j,\mathbf{a}}\rho_{j,\mathbf{a}}^i\right)\right\|}, \tag{3.8b}$$

$$\sum_{\mathbf{a}\in\mathcal{V}_j}\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\mathcal{I}_j^{p_j}(\psi_{j,\mathbf{a}}\rho_{j,\mathbf{a}}^i)\right\|_{\omega_j^{\mathbf{a}}}^2 \leq \sum_{\mathbf{a}\in\mathcal{V}_j}\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_{j,\mathbf{a}}^i\right\|_{\omega_j^{\mathbf{a}}}^2, \tag{3.8c}$$

then define the level j descent direction $\rho_j^i \in V_j^{p_j}$ as

$$\rho_j^i := \sum_{\mathbf{a}\in\mathcal{V}_j}\mathcal{I}_j^{p_j}(\psi_{j,\mathbf{a}}\rho_{j,\mathbf{a}}^i); \tag{3.9}$$

otherwise define

$$\rho_j^i := \sum_{\mathbf{a}\in\mathcal{V}_j}\rho_{j,\mathbf{a}}^i. \tag{3.10}$$

If $\rho_j^i = 0$, set $\lambda_j^i := 1$, otherwise define the optimal step-size on level j

$$\lambda_j^i := \frac{(f,\rho_j^i) - (\boldsymbol{\mathcal{K}}\nabla u_{J,j-1}^i,\nabla\rho_j^i)}{\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_j^i\right\|^2}. \tag{3.11}$$

The level update is given by

$$u_{J,j}^i := u_{J,j-1}^i + \lambda_j^i\rho_j^i. \tag{3.12}$$

(c) The update after the full-smoothing substep is $u_J^{i+\frac{1}{2}} := u_{J,J}^i \in V_J^p$.

2. **Marking** We mark the patches and/or the coarse level by the following bulk-chasing criterion Dörfler [1996], for a parameter $\theta \in (0,1)$

$$\theta^2\left(\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_0^i\right\|^2 + \sum_{j=1}^{J}\lambda_j^i\sum_{\mathbf{a}\in\mathcal{V}_j}\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_{j,\mathbf{a}}^i\right\|_{\omega_j^{\mathbf{a}}}^2\right) \leq \sum_{j\in\mathcal{M}}\lambda_j^i\sum_{\mathbf{a}\in\mathcal{M}_j}\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_{j,\mathbf{a}}^i\right\|_{\omega_j^{\mathbf{a}}}^2, \tag{3.13}$$

with the convention that if $0 \in \mathcal{M}$, we write $\sum_{\mathbf{a}\in\mathcal{M}_0}\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_{0,\mathbf{a}}^i\right\|_{\omega_0^{\mathbf{a}}}^2$ to mean $\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_0^i\right\|^2$.

3. **Test (adaptive substep):** If the two following conditions are satisfied, proceed to the adaptive-smoothing substep:

$$\sum_{j\in\mathcal{M}}\lambda_j^i\sum_{\mathbf{a}\in\mathcal{M}_j}\left(\sum_{k=j}^{J}\lambda_k^i\boldsymbol{\mathcal{K}}\nabla\rho_k^i,\nabla\rho_{j,\mathbf{a}}^i\right)_{\omega_j^{\mathbf{a}}} \leq \gamma^2\sum_{j\in\mathcal{M}}\lambda_j^i\sum_{\mathbf{a}\in\mathcal{M}_j}\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_{j,\mathbf{a}}^i\right\|_{\omega_j^{\mathbf{a}}}^2, \tag{3.14}$$

$$\lambda_j^i \leq 2(d+1) \quad \forall j \in \{0,\dots,J\}, \tag{3.15}$$

where $\gamma \in (0,1)$ is a user-prescribed parameter. If these conditions do not hold, then let $u_J^{i+1} := u_J^{i+\frac{1}{2}}$ and ignore the adaptive-smoothing substep.

Conditions (3.14), (3.15) are needed in the analysis below. One might possibly prove (3.14) by a strengthened Cauchy–Schwarz analysis under some circumstances, but this condition is sometimes numerically not satisfied. Condition (3.15) was always satisfied in our numerical experiments and the proof that (3.15) holds could possibly be accomplished via a p-robust multilevel stable decomposition.

4. **Adaptive-smoothing substep**

(a) If $0 \notin \mathcal{M}$, then define $\rho_0^{i+\frac{1}{2}} := 0$ and $\lambda_0^{i+\frac{1}{2}} := 0$.
Otherwise, when $0 \in \mathcal{M}$, set $\lambda_0^{i+\frac{1}{2}} := 1$ and define $\rho_0^{i+\frac{1}{2}} \in V_0^1$ by

$$(\boldsymbol{\mathcal{K}}\nabla\rho_0^{i+\frac{1}{2}}, \nabla v_0) = (f, v_0) - (\boldsymbol{\mathcal{K}}\nabla u_J^{i+\frac{1}{2}}, \nabla v_0) \quad \forall v_0 \in V_0^1. \tag{3.16}$$

Define the coarsest level update $u_{J,0}^{i+\frac{1}{2}} := u_J^{i+\frac{1}{2}} + \lambda_0^{i+\frac{1}{2}}\rho_0^{i+\frac{1}{2}}$.

(b) Let $j \in \{1,\dots,J\}$. If j is not a marked level ($j \notin \mathcal{M}$), then define $\rho_j^{i+\frac{1}{2}} := 0$, $\lambda_j^{i+\frac{1}{2}} := 0$, and $u_{J,j}^{i+\frac{1}{2}} := u_{J,j-1}^{i+\frac{1}{2}}$. Otherwise, when j is a marked level ($j \in \mathcal{M}$), then define $\rho_{j,\mathbf{a}}^{i+\frac{1}{2}} \in V_j^{\mathbf{a}}$ for all marked vertices $\mathbf{a} \in \mathcal{M}_j$ by

$$(\boldsymbol{\mathcal{K}}\nabla\rho_{j,\mathbf{a}}^{i+\frac{1}{2}}, \nabla v_{j,\mathbf{a}})_{\omega_j^{\mathbf{a}}} = (f, v_{j,\mathbf{a}})_{\omega_j^{\mathbf{a}}} - (\boldsymbol{\mathcal{K}}\nabla u_{J,j-1}^{i+\frac{1}{2}}, \nabla v_{j,\mathbf{a}})_{\omega_j^{\mathbf{a}}} \quad \forall v_{j,\mathbf{a}} \in V_j^{\mathbf{a}}. \tag{3.17}$$

i. **Test (adaptive smoothing choice)**: If the following conditions hold

$$\sum_{\mathbf{a}\in\mathcal{M}_j} \mathcal{I}_j^{p_j}\left(\psi_{j,\mathbf{a}}\rho_{j,\mathbf{a}}^{i+\frac{1}{2}}\right) \neq 0, \tag{3.18a}$$

$$\left(\frac{\sum_{\mathbf{a}\in\mathcal{M}_j} \|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_{j,\mathbf{a}}^{i+\frac{1}{2}}\|_{\omega_j^{\mathbf{a}}}^2}{d+1}\right)^{\frac{1}{2}} \leq \frac{\sum_{\mathbf{a}\in\mathcal{M}_j}\left[\left(f, \mathcal{I}_j^{p_j}\left(\psi_{j,\mathbf{a}}\rho_{j,\mathbf{a}}^{i+\frac{1}{2}}\right)\right)_{\omega_j^{\mathbf{a}}} - \left(\boldsymbol{\mathcal{K}}\nabla u_{J,j-1}^{i+\frac{1}{2}}, \nabla\mathcal{I}_j^{p_j}\left(\psi_{j,\mathbf{a}}\rho_{j,\mathbf{a}}^{i+\frac{1}{2}}\right)\right)_{\omega_j^{\mathbf{a}}}\right]}{\left\|\sum_{\mathbf{a}\in\mathcal{M}_j}\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\mathcal{I}_j^{p_j}\left(\psi_{j,\mathbf{a}}\rho_{j,\mathbf{a}}^{i+\frac{1}{2}}\right)\right\|}, \tag{3.18b}$$

then define the level j descent direction $\rho_j^{i+\frac{1}{2}} \in V_j^{p_j}$ as

$$\rho_j^{i+\frac{1}{2}} := \sum_{\mathbf{a}\in\mathcal{M}_j} \mathcal{I}_j^{p_j}\left(\psi_{j,\mathbf{a}}\rho_{j,\mathbf{a}}^{i+\frac{1}{2}}\right); \tag{3.19}$$

otherwise define

$$\rho_j^{i+\frac{1}{2}} := \sum_{\mathbf{a}\in\mathcal{M}_j} \rho_{j,\mathbf{a}}^{i+\frac{1}{2}}. \tag{3.20}$$

If $\rho_j^{i+\frac{1}{2}} = 0$, set $\lambda_j^{i+\frac{1}{2}} := 1$, otherwise define the optimal step-size on level j

$$\lambda_j^{i+\frac{1}{2}} := \frac{(f, \rho_j^{i+\frac{1}{2}}) - (\boldsymbol{\mathcal{K}}\nabla u_{J,j-1}^i, \nabla\rho_j^{i+\frac{1}{2}})}{\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_j^{i+\frac{1}{2}}\right\|^2}. \tag{3.21}$$

The level update is given by

$$u_{J,j}^{i+\frac{1}{2}} := u_{J,j-1}^{i+\frac{1}{2}} + \lambda_j^{i+\frac{1}{2}}\rho_j^{i+\frac{1}{2}}. \tag{3.22}$$

(c) The final update is $u_J^{i+1} := u_{J,J}^{i+\frac{1}{2}} \in V_J^p$.

Remark 3.1 (Compact writing of the iteration updates)**.** *Let $u_J^i \in V_J^p$. After the full-smoothing substep of the solver introduced above, we have*

$$u_J^{i+\frac{1}{2}} = u_J^i + \sum_{j=0}^{J} \lambda_j^i \rho_j^i, \tag{3.23}$$

and after the adaptive-smoothing substep we have

$$u_J^{i+1} = u_J^{i+\frac{1}{2}} + \sum_{j\in\mathcal{M}} \lambda_j^{i+\frac{1}{2}}\rho_j^{i+\frac{1}{2}}. \tag{3.24}$$

Analogously to [Miraçi et al., 2021a, Theorem 4.7], see Theorem 4.7 in Chapter 2, due to the optimal step-sizes (3.11),(3.21), the error after each substep of the solver can be represented conveniently:

Lemma 3.2 (Error representation of each substep of the solver)**.** *For $u_J^i \in V_J^p$, let $u_J^{i+\frac{1}{2}} \in V_J^p$, $u_J^{i+1} \in V_J^p$ be constructed from u_J^i by the full-smoothing and the adaptive-smoothing substep of the solver of Section 3, respectively. Then*

$$\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\big(u_J - u_J^{i+\frac{1}{2}}\big)\right\|^2 = \left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\big(u_J - u_J^i\big)\right\|^2 - \sum_{j=0}^{J}\Big(\lambda_j^i\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_j^i\right\|\Big)^2, \tag{3.25}$$

$$\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\big(u_J - u_J^{i+1}\big)\right\|^2 = \left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\big(u_J - u_J^{i+\frac{1}{2}}\big)\right\|^2 - \sum_{j\in\mathcal{M}}\Big(\lambda_j^{i+\frac{1}{2}}\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_j^{i+\frac{1}{2}}\right\|\Big)^2. \tag{3.26}$$

4 A posteriori estimator on the algebraic error

The solver we introduced in Section 3 is inherently linked to an a posteriori estimator η_{alg}^i for the full-smoothing substep and $\eta_{\text{alg}}^{i+\frac{1}{2}}$ for the adaptive-smoothing substep.

Definition 4.1 (Algebraic error estimator)**.** *Let $u_J^i \in V_J^p$ be arbitrary, let $u_J^{i+\frac{1}{2}} \in V_J^p$ be the update at the end of the full-smoothing substep, and let $u_J^{i+1} \in V_J^p$ be the update at the end of the adaptive substep. We define the algebraic error estimators*

$$\eta_{\text{alg}}^i := \left(\sum_{j=0}^{J} \left(\lambda_j^i \left\| \boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \rho_j^i \right\| \right)^2 \right)^{\frac{1}{2}}, \tag{3.27}$$

$$\eta_{\text{alg}}^{i+\frac{1}{2}} := \left(\sum_{j \in \mathcal{M}} \left(\lambda_j^{i+\frac{1}{2}} \left\| \boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \rho_j^{i+\frac{1}{2}} \right\| \right)^2 \right)^{\frac{1}{2}}. \tag{3.28}$$

The following result is immediate from Lemma 3.2:

Lemma 4.2 (Guaranteed lower bound on the algebraic error per substep)**.** *Under the assumptions of Lemma 3.2 and Definition 4.1, the estimators are guaranteed lower bounds on the algebraic error for the respective substeps of the solver*

$$\left\| \boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \left(u_J - u_J^i \right) \right\| \geq \eta_{\text{alg}}^i, \tag{3.29}$$

$$\left\| \boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \left(u_J - u_J^{i+\frac{1}{2}} \right) \right\| \geq \eta_{\text{alg}}^{i+\frac{1}{2}}. \tag{3.30}$$

5 Main results

We present here our main result for the solver introduced in Section 3. Similarly to Miraçi et al. [2020], Miraçi et al. [2021a], see Chapters 1 and 2, we show for each substep that the error contraction of the solver is equivalent to the efficiency of the associated a posteriori error estimator.

5.1 Mesh assumptions

For $j \in \{1, \ldots, J\}$, we denote in the following $h_K := \operatorname{diam}(K)$ for $K \in \mathcal{T}_j$ and $h_j = \max_{K \in \mathcal{T}_j} h_K$. We shall always assume that our meshes are shape-regular:

Assumption 5.1 (Shape regularity)**.** *There exists $\kappa_{\mathcal{T}} > 0$ such that*

$$\max_{K \in \mathcal{T}_j} \frac{h_K}{\rho_K} \leq \kappa_{\mathcal{T}} \text{ for all } 0 \leq j \leq J, \tag{3.31}$$

where ρ_K denotes the diameter of the largest ball contained in K.

We mainly work with a hierarchy of quasi-uniform meshes with a bounded refinement factor between consecutive levels. This setting is described by:

Assumption 5.2 (Refinement strength and mesh quasi-uniformity)**.** *There exists $0 < C_{\text{ref}} \leq 1$, a fixed positive real number such that for any $j \in \{1, \ldots, J\}$, for all $K \in \mathcal{T}_{j-1}$, and for any $K^* \in \mathcal{T}_j$ such that $K^* \subset K$, there holds*

$$C_{\text{ref}} h_K \leq h_{K^*} \leq h_K. \tag{3.32}$$

There further exists C_{qu}, a fixed positive real number such that for any $j \in \{0, \ldots, J\}$ and for all $K \in \mathcal{T}_j$, there holds

$$C_{\text{qu}} h_j \leq h_K \leq h_j. \tag{3.33}$$

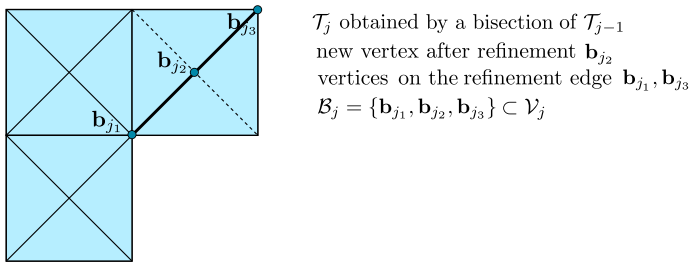


Figure 3.3: Illustration of the set $\mathcal{B}_j$; the refinement $\mathcal{T}_j$ (dotted lines) of mesh $\mathcal{T}_{j-1}$ (full lines).

The forthcoming main result also covers the setting of graded bisection grids, e.g. the newest vertex bisection, cf. Sewell [1972] and Mitchell [1991], that we present here for completeness. In this case, one refinement of an edge of $\mathcal{T}_{j-1}$, for $j \in \{1, \dots, J\}$, gives us a new finer mesh $\mathcal{T}_j$. We denote by $\mathcal{B}_j \subset \mathcal{V}_j$ the set consisting of the new vertex obtained after the bisection together with its two neighbors on the refinement edge, cf. Figure 3.3 for an illustration when $d = 2$. We denote by $h_{\mathcal{B}_j}$ the maximal diameter of elements having a vertex in $\mathcal{B}_j$. This setting is described by:

Assumption 5.3 (Local quasi-uniformity of bisection-generated meshes)**.** *$\mathcal{T}_0$ is a conforming quasi-uniform mesh with parameter C^0_{qu}. The graded conforming mesh $\mathcal{T}_J$ is generated from $\mathcal{T}_0$ by a series of bisections. There exists a fixed positive real number $C_{\mathrm{loc,qu}}$ such that for any $j \in \{1, \dots, J\}$, there holds*

$$C_{\mathrm{loc,qu}} h_{\mathcal{B}_j} \le h_K \le h_{\mathcal{B}_j} \quad \forall K \in \mathcal{T}_j \text{ such that a vertex of } K \text{ belongs to } \mathcal{B}_j. \tag{3.34}$$

5.2 Main result

We now present the main result of this manuscript.

Theorem 5.4 (p-robust error contraction of the adaptive multilevel solver)**.** *Let Assumption 5.1 hold, and let either Assumption 5.2 or Assumption 5.3 be satisfied. Let $u_J \in V_J^p$ be the (unknown) solution of (3.3) and let $u_J^i \in V_J^p$ be arbitrary, $i \ge 0$. Let $u_J^{i+\frac{1}{2}} \in V_J^p$ be the update at the end of the full-smoothing substep of the solver described in Section 3. Then*

$$\big\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \big(u_J - u_J^{i+\frac{1}{2}}\big)\big\| \le \alpha \big\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \big(u_J - u_J^{i}\big)\big\|. \tag{3.35}$$

When tests (3.14)–(3.15) are satisfied, let $u_J^{i+1} \in V_J^p$ be the update at the end of the adaptive substep. Then

$$\big\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \big(u_J - u_J^{i+1}\big)\big\| \le \tilde{\alpha} \big\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \big(u_J - u_J^{i+\frac{1}{2}}\big)\big\|. \tag{3.36}$$

Here $0 < \alpha < 1$, $0 < \tilde{\alpha} < 1$ depend on the space dimension d, the mesh shape regularity parameter $\kappa_{\mathcal{T}}$, the number of mesh levels J, and the ratio of the largest and the smallest eigenvalues of the diffusion coefficient $\boldsymbol{\mathcal{K}}$, as well as on the mesh refinement parameter C_{ref} and quasi-uniformity parameter C_{qu} if Assumption 5.2 holds, or the coarse grid/local quasi-uniformity parameters C^0_{qu} and $C_{\mathrm{loc,qu}}$ if Assumption 5.3 holds. The dependence on the number of levels J is at most linear for

α and cubic for $\tilde{\alpha}$. The factor $\tilde{\alpha}$ depends additionally on the marking parameter θ and the adaptivity test parameter γ from (3.14).

Tests (3.14)–(3.15) are analysis-driven checks, that, if satisfied, ensure at the end of the full-smoothing substep that the adaptive-smoothing substep will also contract the error.

5.3 Additional results

There is a strong link between the solver defined in Section 3 and the a posteriori estimators defined in Section 4. Similarly to Miraçi et al. [2020], Miraçi et al. [2021a], see Chapters 1 and 2, we have the following theorem (recall also Lemma 4.2).

Theorem 5.5 (Equivalence estimator efficiency–solver contraction)**.** *Let the assumptions of Theorem 5.4 be satisfied. Then* (3.35) *holds if and only if*

$$\eta^i_{\mathrm{alg}} \geq \beta \big\| \boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \big(u_J - u^i_J \big) \big\| \tag{3.37}$$

holds with $\beta = \sqrt{1-\alpha^2}$. Similarly, (3.36) *holds if and only if*

$$\eta^{i+\frac{1}{2}}_{\mathrm{alg}} \geq \tilde{\beta} \big\| \boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \big(u_J - u^{i+\frac{1}{2}}_J \big) \big\| \tag{3.38}$$

holds with $\tilde{\beta} = \sqrt{1-\tilde{\alpha}^2}$.

The following result can be seen as the main motivation for our adaptive algorithm.

Corollary 5.6 (Equivalence error–estimator–localized contributions)**.** *Let the assumptions of Theorem 5.4 be satisfied. Then, at the end of the full-smoothing substep, there holds*

$$\big\| \boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \big(u_J - u^i_J \big) \big\|^2 \approx \Big(\eta^i_{\mathrm{alg}} \Big)^2 \approx \big\| \boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \rho^i_0 \big\|^2 + \sum_{j=1}^{J} \lambda^i_j \sum_{\mathbf{a} \in \mathcal{V}_j} \big\| \boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \rho^i_{j,\mathbf{a}} \big\|^2_{\omega^{\mathbf{a}}_j}, \tag{3.39}$$

where the constants involved in the equivalences "$\approx$" have the same dependency as α in (3.35)*, see* (3.45) *below for details.*

6 Numerical experiments

We consider four test cases: "Peak" (smooth solution with source term dominating in a part of a square domain), "L-shape" (problem with a singularity due to the L-shaped domain with a re-entrant corner), and "Skyscraper" (a problem we consider in two variants: with diffusion tensor having a jump of order 10^2 and 10^5), described in detail in [Miraçi et al., 2021a, Section 9], see Section 9 in Chapter 2. The hierarchy of meshes we consider here is obtained through uniform refinement. We point out that test (3.15) is always satisfied in practice, whereas (3.14) is not always satisfied. In order to see numerical evidence of p-robustness, the stopping criterion is given by the relative residual dropping below 10^{-5}.

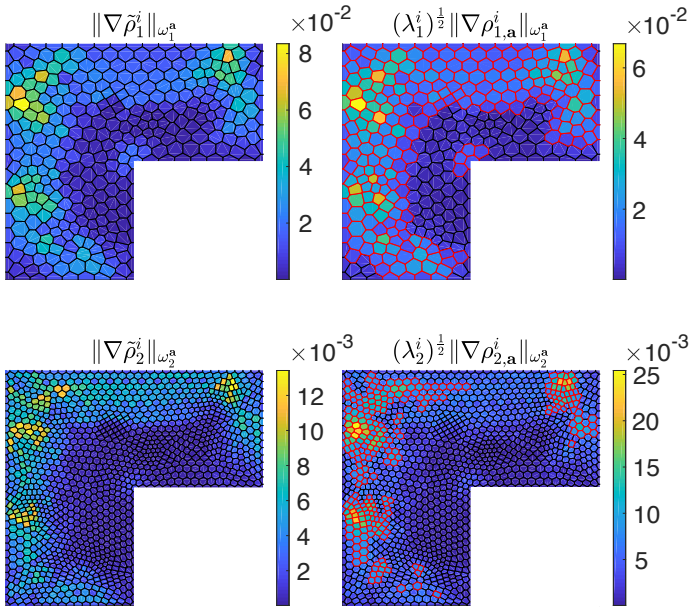


Figure 3.4: [L-shape, J=2, p_0=1, p_1=p_2=3, θ=0.95, γ=0.7] Comparing algebraic error distribution (left) to local error indicators (right) (levels $j = 1$ top, $j = 2$ bottom). Voronoi cells correspond to patch values, and the ones with the red border are marked for local smoothing.

6.1 Can we predict the distribution of the algebraic error?

We provide in Figures 3.4–3.5 an illustration on how the distribution of the algebraic error $\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\left(u_J - u_J^i\right)\right\|$ is locally estimated using our algebraic error indicators. For this purpose, we consider the L-shape and Peak problems on a mesh hierarchy with $J = 2$ and $p_1 = p_2 = 3$, respectively $p_1 = p_2 = 6$ (recall that $p_0 = 1$ in our setting). In the figures, we compare, for a single iteration ($i = 3$ for L-shape, $i = 4$ for Peak), our algebraic error indicators $\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_{j,\mathbf{a}}^i\|_{\omega_j^{\mathbf{a}}}$ with the local algebraic error distribution $\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\tilde{\rho}_j^i\|_{\omega_j^{\mathbf{a}}}$, where $\tilde{\rho}_j^i \in V_j^{p_j}$ is the levelwise orthogonal decomposition of the algebraic error with $\tilde{\rho}_0^i = \rho_0^i$ and, for $j \in \{1, \dots, J\}$,

$$(\boldsymbol{\mathcal{K}}\nabla\tilde{\rho}_j^i, \nabla v_j) = (f, v_j) - (\boldsymbol{\mathcal{K}}\nabla u_J^i, \nabla v_j) - \sum_{k=0}^{j-1}(\boldsymbol{\mathcal{K}}\nabla\tilde{\rho}_k^i, \nabla v_j) \qquad \forall v_j \in V_j^{p_j},$$

see, e.g., [Miraçi et al., 2021a, Section 3], see Section 3 in Chapter 2. We highlight by a red border the patches marked for smoothing in the adaptive-smoothing substep, with the choice of the Dörfler's marking parameter $\theta = 0.95$ in (3.13).

One can see that the local error indicators provide indeed a quite accurate information about the error distribution over the levels and patches in these tests. We note that one obtains similar results also for the other test cases, higher number of mesh levels J, different polynomial degrees p, and different choices of the marking parameter θ. Thus the considered adaptivity indeed targets the problematic regions. It is important to note that the region with increased error could be dynamically changing from iteration to iteration. Our localized a posteriori estimator is designed in such a way that it will dynamically adjust to the new region with increased error. In all our experiments, the regions of increased algebraic error were rather stable,

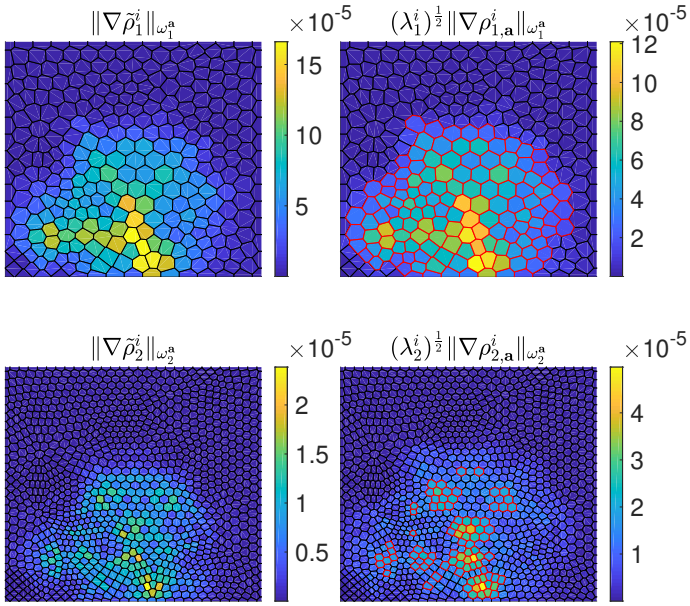


Figure 3.5: [Peak, J=2, p_0=1, p_1=p_2=6, θ=0.95, γ=0.7] Comparing algebraic error distribution (left) to local error indicators (right) (levels $j=1$ top, $j=2$ bottom). Voronoi cells correspond to patch values, and the ones with the red border are marked for local smoothing.

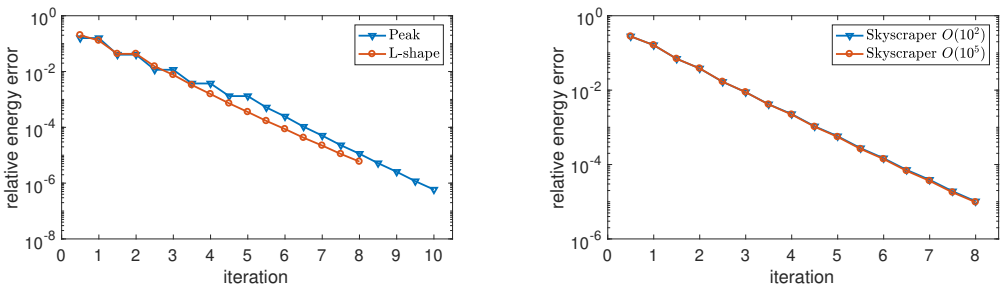


Figure 3.6: [All tests, $J=3$, $p_0=1$, $p_1=1$, $p_2=2$, $p_3=3$, $\theta=0.95$, $\gamma=0.7$] Convergence of Algorithm 1 in the relative energy norm of the algebraic error $\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J-u_J^i)\|/\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla u_J\|$.

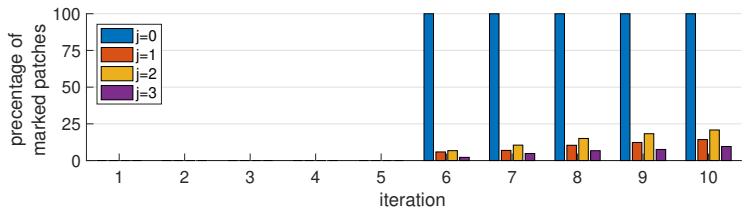
but we note that when periodic flipping occurs, the overall efficiency of the adaptive local smoothing in Algorithm 1 may be spoiled.

6.2 Does the adaptivity pay off?

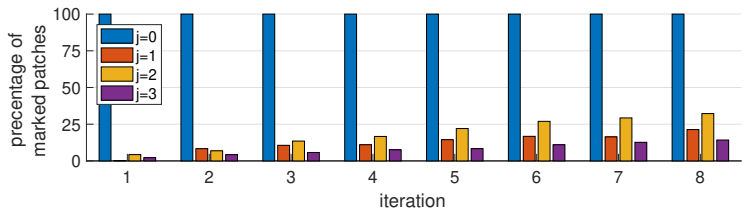
Next, we investigate the performance of the adaptive Algorithm 1. We focus on convergence in the energy norm of the algebraic error during the iterations and the percentage of the patches marked for local adaptive smoothing. For this purpose, we consider the four test cases and $J=3$, $p_j=1,1,2,3$, $\gamma=0.7$, and the marking parameter θ fixed to 0.95; one obtains similar results also for other polynomial degrees. The results are summarized in Figure 3.6. One can see the decrease in each full-smoothing substep and that the adaptive substeps indeed also yield a decrease of the energy norm of the error; the adaptive-smoothing substeps actually

yield nearly the same decrease as the full substeps – the convergence curve is nearly affine (in log scale) in the iterations where the adaptive smoothing is performed (note a stagnation in the iterations where condition (3.14) was not satisfied and hence the adaptive-smoothing substep was not performed). Figures 3.7–3.8 then confirm that only a small portion of patches is marked for local adaptive smoothing, which suggest that Algorithm 1 may also be computationally beneficial.

Peak test case



Skyscraper test case (diff. contrast $O(10^2)$)



Skyscraper test case (diff. contrast $O(10^5)$)

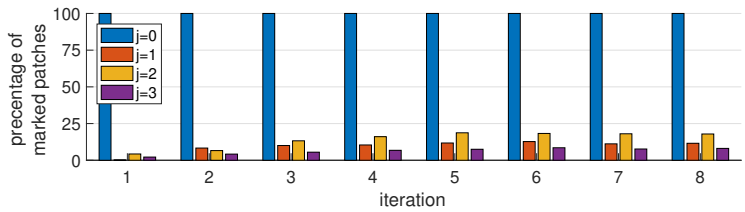


Figure 3.7: [Different tests, $J = 3$, $p_0 = 1$, $p_1 = 1$, $p_2 = 2$, $p_3 = 3$, $\theta = 0.95$, $\gamma = 0.7$] Local adaptive smoothing: coarsest level marked or not and percentages of patches marked for each level $1 \leq j \leq J$ (Y-axis). Iterations of Algorithm 1 (X-axis). Results for the L-shape test case are given in the separate Figure 3.8.

Next, we test if the adaptive substeps provide a speed-up with respect to the variant without the adaptive substep. In Table 3.1, we compare, for varying polynomial degrees and number of levels, the results of Algorithm 1 when varying the parameter γ from test (3.14). We consider choices $\gamma = 0$, which corresponds to not using the adaptive substep at all, $\gamma = 0.7$, and, formally, $\gamma = \infty$, which stands for skipping the evaluation of (3.14), (3.15) and using the adaptive substep in every iteration. The latter choice is motivated by the fact that one would want to avoid evaluating the terms in test (3.14) if possible.

In Table 3.1, we in particular provide the number of iterations i with the number of adaptive-smoothing substeps in the brackets. For example "6(4)" means that the solver took 6 iterations to reach the stopping criterion, and the tests (3.14)–(3.15) were passed four times, i.e., 4 adaptive-smoothing substeps were performed

L-shape test case

$\theta = 0.7$

$\theta = 0.9$

$\theta = 0.95$

$\theta = 0.99$

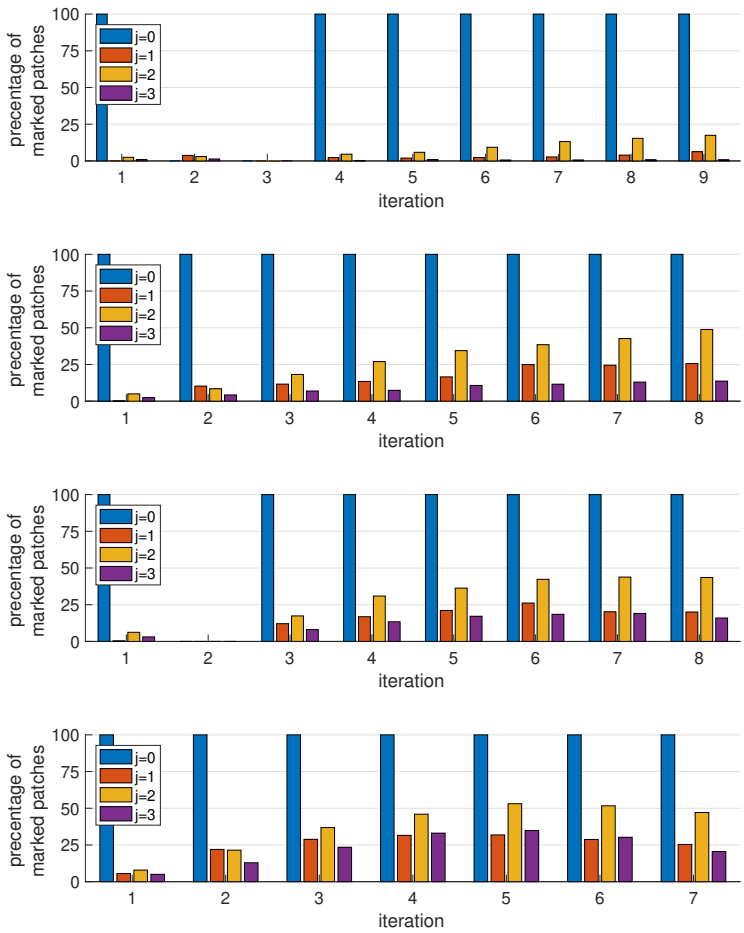


Figure 3.8: [L-shape, $J = 3$, $p_0 = 1$, $p_1 = 1$, $p_2 = 2$, $p_3 = 3$, $\gamma = 0.7$, varying θ] Local adaptive smoothing: coarsest level marked or not and percentages of patches marked for each level $1 \leq j \leq J$ (Y-axis). Iterations of Algorithm 1 (X-axis).

in addition to the 6 full-smoothing substeps. For $p = 1$, test (3.14) is typically not verified, but otherwise Algorithm 1 with $\gamma = 0.7$ usually passes the adaptivity test (3.14) and leads to a reduction of the total number of iterations for the price of only employing a few local-adaptive-smoothing substeps. By always employing the adaptive substep ($\gamma = \infty$), we may cut the iteration count by nearly a half also for $p = 1$.

For comparison of the associated computational cost, we also provide, as in Miraçi et al. [2021a], see Chapter 2, an estimated number of floating point operations. This number is given by the formula

$$\text{nflops} := \frac{|\mathcal{V}_0|^3}{3} + \sum_{j=1}^{J} \sum_{\mathbf{a}\in\mathcal{V}_j} \frac{\text{ndof}(V_j^{\mathbf{a}})^3}{3} + \sum_{i=1}^{i_s} \left[2\delta_0^i |\mathcal{V}_0|^2 + \sum_{j\in\mathcal{M}\setminus\{0\}} \sum_{\mathbf{a}\in\mathcal{M}_j} 2\text{ndof}(V_j^{\mathbf{a}})^2 \right]$$
$$+ \sum_{i=1}^{i_s} \sum_{j=1}^{J} \left[2\,\text{nnz}(\mathcal{I}_{j-1}^{j}) + 2\,\text{nnz}(\mathcal{I}_{j}^{j-1}) + 2\,\text{nnz}(\mathbb{A}_j) + 3(2\,\text{size}(\mathbb{A}_j)) \right].$$

			$\gamma = 0$		$\gamma = 0.7$		$\gamma = \infty$	
	J	p_j	niter	nflops	niter	nflops	niter	nflops
Peak test case	3	1 1 1 1	19(0)	2.11×10^{7}	19(0)	2.11×10^{7}	11(11)	2.22×10^{7}
		1 1 2 3	15(0)	4.26×10^{8}	10(5)	3.70×10^{8}	8(8)	3.63×10^{8}
		1 2 4 6	12(0)	8.81×10^{9}	9(4)	8.15×10^{9}	7(7)	7.74×10^{9}
		1 3 6 9	13(0)	8.17×10^{10}	9(7)	7.69×10^{10}	8(8)	7.54×10^{10}
	4	1 1 1 1 1	20(0)	7.17×10^{7}	20(0)	7.17×10^{7}	12(12)	8.20×10^{7}
		1 1 2 2 3	13(0)	1.51×10^{9}	10(4)	1.43×10^{9}	8(8)	1.46×10^{9}
		1 2 3 5 6	11(0)	3.78×10^{10}	9(4)	3.68×10^{10}	7(7)	3.52×10^{10}
		1 3 5 7 9	13(0)	3.46×10^{11}	9(7)	3.28×10^{11}	8(8)	3.21×10^{11}
			$\gamma = 0$		$\gamma = 0.7$		$\gamma = \infty$	
	J	p_j	niter	nflops	niter	nflops	niter	nflops
L-shape test case	3	1 1 1 1	21(0)	2.17×10^{7}	21(0)	2.17×10^{7}	11(11)	2.11×10^{7}
		1 1 2 3	13(0)	3.63×10^{8}	8(7)	3.43×10^{8}	7(7)	3.19×10^{8}
		1 2 4 6	8(0)	7.02×10^{9}	5(5)	6.50×10^{9}	5(5)	6.50×10^{9}
		1 3 6 9	8(0)	6.94×10^{10}	5(5)	6.59×10^{10}	5(5)	6.59×10^{10}
	4	1 1 1 1 1	21(0)	7.24×10^{7}	21(0)	7.24×10^{7}	11(11)	7.29×10^{7}
		1 1 2 2 3	9(0)	1.06×10^{9}	8(5)	1.24×10^{9}	6(6)	1.10×10^{9}
		1 2 3 5 6	7(0)	2.95×10^{10}	5(5)	2.92×10^{10}	5(5)	2.92×10^{10}
		1 3 5 7 9	6(0)	2.75×10^{11}	5(5)	2.78×10^{11}	5(5)	2.78×10^{11}
			$\gamma = 0$		$\gamma = 0.7$		$\gamma = \infty$	
	J	p_j	niter	nflops	niter	nflops	niter	nflops
Skyscraper test case diff. contrast $O(10^2)$	3	1 1 1 1	19(0)	1.90×10^{7}	19(0)	1.90×10^{7}	12(12)	2.18×10^{7}
		1 1 2 3	15(0)	4.10×10^{8}	8(8)	3.50×10^{8}	8(8)	3.50×10^{8}
		1 2 4 6	9(0)	7.36×10^{9}	6(6)	6.94×10^{9}	6(6)	6.94×10^{9}
		1 3 6 9	9(0)	7.11×10^{10}	6(6)	6.80×10^{10}	6(6)	6.80×10^{10}
	4	1 1 1 1 1	19(0)	6.31×10^{7}	19(0)	6.31×10^{7}	12(12)	7.61×10^{7}
		1 1 2 2 3	11(0)	1.26×10^{9}	8(7)	1.35×10^{9}	7(7)	1.25×10^{9}
		1 2 3 5 6	8(0)	3.11×10^{10}	6(6)	3.15×10^{10}	6(6)	3.15×10^{10}
		1 3 5 7 9	8(0)	2.91×10^{11}	5(5)	2.77×10^{11}	5(5)	2.77×10^{11}
			$\gamma = 0$		$\gamma = 0.7$		$\gamma = \infty$	
	J	p_j	niter	nflops	niter	nflops	niter	nflops
Skyscraper test case diff. contrast $O(10^5)$	3	1 1 1 1	19(0)	1.90×10^{7}	19(0)	1.90×10^{7}	13(13)	2.33×10^{7}
		1 1 2 3	15(0)	4.10×10^{8}	8(8)	3.48×10^{8}	8(8)	3.48×10^{8}
		1 2 4 6	9(0)	7.36×10^{9}	6(6)	6.93×10^{9}	6(6)	6.93×10^{9}
		1 3 6 9	9(0)	7.11×10^{10}	6(6)	6.79×10^{10}	6(6)	6.79×10^{10}
	4	1 1 1 1 1	19(0)	6.31×10^{7}	19(0)	6.31×10^{7}	12(12)	7.60×10^{7}
		1 1 2 2 3	11(0)	1.26×10^{9}	8(7)	1.35×10^{9}	7(7)	1.25×10^{9}
		1 2 3 5 6	8(0)	3.11×10^{10}	6(6)	3.15×10^{10}	6(6)	3.15×10^{10}
		1 3 5 7 9	8(0)	2.91×10^{11}	5(5)	2.77×10^{11}	5(5)	2.77×10^{11}

Table 3.1: Number of iterations (number of adaptive-smoothing substeps in brackets) for various choices of the parameter γ in (3.14). The marking parameter in (3.13) is set as $\theta = 0.95$

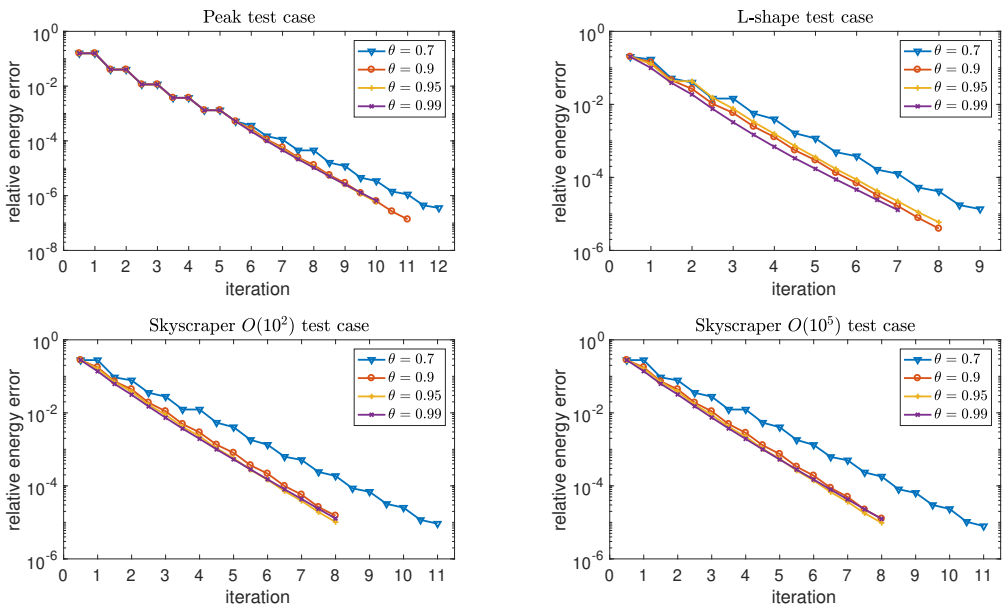


Figure 3.9: [All tests, $J = 3$, $p_0 = 1$, $p_1 = 1$, $p_2 = 2$, $p_3 = 3$, $\gamma = 0.7$, varying θ] Convergence of Algorithm 1 in the relative energy norm of the algebraic error $\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J - u_J^i)\|/\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla u_J\|$.

This formula is derived assuming 1) an initial Cholesky decomposition of local matrices associated to each patch on each level except for the coarsest one, where the global stiffness matrix for piecewise affine functions is factorized (for a matrix of size n, this cost is estimated as $1/3n^3$); 2) local solves by forward and backward substitutions (cost $2n^2$); 3) $\mathcal{I}_{j-1}^j : V_{j-1}^{p_{j-1}} \to V_j^{p_j}$ with the cost estimated by two-times the number of nonzeros of the associated interpolation matrix; and 4) evaluation of the optimal step-sizes λ_j as in formulas (3.11), (3.21) involving multiplication with the stiffness matrix $\mathbb{A}_j$ on the given level (cost equal to two-times the number of nonzeros) and three inner products. From the above tests, we see that adaptivity is of interest. Not only does it provide error contraction on the adaptive substep of almost the same quality as the full-smoothing substep with just local smoothing in a relatively small percentage of marked patches, cf. Figures 3.6–3.8, but in numerous cases, the adaptive variant is cheaper than the non-adaptive one in terms of the above nflops formula. Note that the nflops only represent *one* way of estimating the costs and the interest in adaptivity is not solely determined by it. Please note that if the coarsest mesh has O(1) elements, the first, cubic term has a minor influence only. The second, also cubic, can then be treated fully in parallel, see [Miraçi et al., 2021a, Section 8] for details, see Section 8 in Chapter 2.

6.3 Dependence on the marking parameter

We finally vary the Dörfler's marking parameter θ from (3.13), setting $\theta = 0.7, 0.9, 0.95, 0.99$. The results are given in Figure 3.9 and in Table 3.2, where we consider $\gamma = 0.7$.

One can see that the choice $\theta = 0.7$ is often not sufficiently efficient. For this choice, the number of iterations is not reduced sufficiently and the cost of intergrid

Peak test case									
		$\theta = 0.7$		$\theta = 0.9$		$\theta = 0.95$		$\theta = 0.99$	
J	p_j	niter	nflops	niter	nflops	niter	nflops	niter	nflops
4	1 1 1 1 1	20(0)	7.17×10^{7}	20(0)	7.17×10^{7}	20(0)	7.17×10^{7}	20(0)	7.17×10^{7}
	1 1 2 2 3	12(2)	1.52×10^{9}	11(3)	1.47×10^{9}	10(4)	1.43×10^{9}	10(4)	1.44×10^{9}
	1 2 3 5 6	11(0)	3.78×10^{10}	10(3)	3.80×10^{10}	9(4)	3.68×10^{10}	8(4)	3.52×10^{10}
	1 3 5 7 9	12(8)	3.57×10^{11}	10(8)	3.39×10^{11}	9(7)	3.28×10^{11}	8(6)	3.17×10^{11}

L-shape test case									
		$\theta = 0.7$		$\theta = 0.9$		$\theta = 0.95$		$\theta = 0.99$	
J	p_j	niter	nflops	niter	nflops	niter	nflops	niter	nflops
4	1 1 1 1 1	21(0)	7.24×10^{7}	21(0)	7.24×10^{7}	21(0)	7.24×10^{7}	21(0)	7.24×10^{7}
	1 1 2 2 3	9(4)	1.28×10^{9}	8(5)	1.24×10^{9}	8(5)	1.24×10^{9}	6(5)	1.06×10^{9}
	1 2 3 5 6	6(3)	2.97×10^{10}	6(4)	3.03×10^{10}	5(5)	2.92×10^{10}	4(4)	2.70×10^{10}
	1 3 5 7 9	6(6)	2.90×10^{11}	5(5)	2.78×10^{11}	5(5)	2.78×10^{11}	4 (4)	2.68×10^{11}

Skyscraper test case (diff. contrast $O(10^2)$)									
		$\theta = 0.7$		$\theta = 0.9$		$\theta = 0.95$		$\theta = 0.99$	
J	p_j	niter	nflops	niter	nflops	niter	nflops	niter	nflops
4	1 1 1 1 1	19(0)	6.31×10^{7}	19(0)	6.31×10^{7}	19(0)	6.31×10^{7}	19(0)	6.31×10^{7}
	1 1 2 2 3	10(4)	1.38×10^{9}	8(7)	1.34×10^{9}	8(7)	1.35×10^{9}	6(6)	1.10×10^{9}
	1 2 3 5 6	8(4)	3.38×10^{10}	6(6)	3.15×10^{10}	6(6)	3.15×10^{10}	5(5)	2.92×10^{10}
	1 3 5 7 9	7(7)	2.99×10^{11}	6(6)	2.88×10^{11}	5(5)	2.77×10^{11}	5(5)	2.77×10^{11}

Skyscraper test case (diff. contrast $O(10^5)$)									
		$\theta = 0.7$		$\theta = 0.9$		$\theta = 0.95$		$\theta = 0.99$	
J	p_j	niter	nflops	niter	nflops	niter	nflops	niter	nflops
4	1 1 1 1 1	19(0)	6.31×10^{7}	19(0)	6.31×10^{7}	19(0)	6.31×10^{7}	19(0)	6.31×10^{7}
	1 1 2 2 3	11(5)	1.53×10^{9}	8(7)	1.34×10^{9}	8(7)	1.35×10^{9}	7(7)	1.26×10^{9}
	1 2 3 5 6	8(4)	3.38×10^{10}	6(6)	3.15×10^{10}	6(6)	3.15×10^{10}	5(5)	2.91×10^{10}
	1 3 5 7 9	7(7)	2.99×10^{11}	6(6)	2.88×10^{11}	5(5)	2.77×10^{11}	5(5)	2.77×10^{11}

Table 3.2: Number of iterations (number of adaptive-smoothing substeps in brackets) for various choices of marking parameter θ in (3.13). The parameter γ from (3.14) is set as $\gamma = 0.7$

operation then dominates over the cost of local smoothings. The best choice of θ seems to differ, but $\theta = 0.95$ reveals quite satisfactory in most of the cases.

Remark 6.1 (Dependence on the shape regularity parameter)**.** *We would like to point out how the performance of the solver depends on the parameters of the Assumptions 5.1–5.3. As an example, we present in Table 3.3 the number of iterations required when the shape regularity parameter $\kappa_{\mathcal{T}}$ degrades. One can see an overall degradation, but the polynomial degree robustness is preserved as expected.*

		minimal angle: 32.1°	minimal angle: 21.4°	minimal angle: 12.0°
p_j	DoF	niter	niter	niter
1 1 2 3	1e5	8(7)	9(9)	17(17)
1 2 4 6	6e5	5(5)	6(6)	11(11)
1 3 6 9	1e6	5(5)	6(6)	10(10)

Table 3.3: [L-shape, $J = 3$, $\theta = 0.95$, $\gamma = 0.7$] Study of sensitivity with respect to the shape regularity of the mesh (minimal angle of mesh elements) for the local adaptive smoothing solver.

7 Proofs of the main results

In this section, we present the proofs of the results stated in Section 5. We start with noting that Theorem 5.5 can be proven exactly along the lines of [Miraçi et al., 2021a, Corollary 6.7], see Corollary 6.7 in Chapter 2.

7.1 Proof of contraction: full-smoothing substep

We start with a generalization of the properties given in Miraçi et al. [2021a], see Chapter 2, covering the test (3.8), in order to extend the results from the case of additive Schwarz smoothing to the case of weighted restricted additive Schwarz smoothing.

Lemma 7.1 (Lower bound on levelwise updates by patchwise contributions)**.** *Let $u_J^i \in V_J^p$ be arbitrary. Let $j \in \{1, \dots, J\}$ and let ρ_j^i, λ_j^i be constructed from u_J^i by the full-smoothing substep of the solver described in Section 3. Then*

$$\sum_{\mathbf{a} \in \mathcal{V}_j} \left\| \boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \rho_{j,\mathbf{a}}^i \right\|_{\omega_j^{\mathbf{a}}}^2 \leq (d+1) \left(\lambda_j^i \left\| \boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \rho_j^i \right\| \right)^2 \quad \forall 1 \leq j \leq J, \tag{3.40}$$

where for each vertex $\mathbf{a} \in \mathcal{V}_j$, $\rho_{j,\mathbf{a}}^i$ is the solution of the local problem (3.7).

Proof. Depending if test (3.8) of the solver in Section 3 is satisfied or not, ρ_j^i will be constructed differently. We show that (3.40) holds for either outcome of test (3.8).

Case test (3.8) is satisfied: Then ρ_j^i is constructed by (3.9) and the outcome of Test (3.8a),(3.8b) ensures on the one hand that $\rho_j^i \neq 0$ and on the other hand that

$$\left(\frac{\sum_{\mathbf{a} \in \mathcal{V}_j} \left\| \boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \rho_{j,\mathbf{a}}^i \right\|_{\omega_j^{\mathbf{a}}}^2}{d+1} \right)^{\frac{1}{2}} \leq \frac{(f, \rho_j^i) - (\boldsymbol{\mathcal{K}} \nabla u_{J,j-1}^i, \nabla \rho_j^i)}{\left\| \boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \rho_j^i \right\|}.$$

Using (3.11), this leads to: $\left(\sum_{\mathbf{a}\in\mathcal{V}_j}\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho^i_{j,\mathbf{a}}\|^2_{\omega^{\mathbf{a}}_j}\right)^{\frac{1}{2}} \leq \sqrt{d+1}\lambda^i_j\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho^i_j\|$.

Case test (3.8) *is not satisfied:* Then ρ^i_j is constructed by (3.10). First, note that

$$\sum_{\mathbf{a}\in\mathcal{V}_j}\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho^i_{j,\mathbf{a}}\|^2_{\omega^{\mathbf{a}}_j} \overset{(3.7),(3.10)}{=} (f,\rho^i_j)-(\boldsymbol{\mathcal{K}}\nabla u^i_{J,j-1},\nabla\rho^i_j). \tag{3.41}$$

Thus, if $\rho^i_j=0$, then the result (3.40) holds trivially. To treat the remaining case $\rho^i_j\neq 0$, we use the expression of λ^i_j together with [Miraçi et al., 2021a, Lemma 9.1], see Lemma 10.1 in Chapter 2, to obtain

$$\sum_{\mathbf{a}\in\mathcal{V}_j}\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho^i_{j,\mathbf{a}}\|^2_{\omega^{\mathbf{a}}_j} \overset{(3.11)}{=} \lambda^i_j\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho^i_j\|^2 \leq \lambda^i_j\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho^i_j\|\left((d+1)\sum_{\mathbf{a}\in\mathcal{V}_j}\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho^i_{j,\mathbf{a}}\|^2_{\omega^{\mathbf{a}}_j}\right)^{\frac{1}{2}}.$$

□

The second important property we will need is given below.

Lemma 7.2 (Upper bound on levelwise updates by patchwise contributions)**.** *Let $u^i_J\in V^p_J$ be arbitrary. Let $j\in\{1,\dots,J\}$ and let ρ^i_j, λ^i_j be constructed from u^i_J by the full-smoothing substep of the solver described in Section 3. Then*

$$\left(\lambda^i_j\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho^i_j\|\right)^2 \leq \lambda^i_j\sum_{\mathbf{a}\in\mathcal{V}_j}\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho^i_{j,\mathbf{a}}\|^2_{\omega^{\mathbf{a}}_j} \quad \forall 1\leq j\leq J, \tag{3.42}$$

where for each vertex $\mathbf{a}\in\mathcal{V}_j$, $\rho^i_{j,\mathbf{a}}$ is the solution of the local problem (3.7)*.*

Proof. We only need to show (3.42) when $\rho^i_j\neq 0$, otherwise the result is trivial.

Case test (3.8) *is satisfied:* Then ρ^i_j is constructed by (3.9) and by using Young's inequality together with test (3.8c), we obtain

$$\begin{aligned}
(f,\rho^i_j)-(\boldsymbol{\mathcal{K}}\nabla u^i_{J,j-1},\nabla\rho^i_j) &= \sum_{\mathbf{a}\in\mathcal{V}_j}\left(\left(f,\mathcal{I}^{p_j}_j(\psi_{j,\mathbf{a}}\rho^i_{j,\mathbf{a}})\right)_{\omega^{\mathbf{a}}_j}-\left(\boldsymbol{\mathcal{K}}\nabla u^i_{J,j-1},\nabla\mathcal{I}^{p_j}_j(\psi_{j,\mathbf{a}}\rho^i_{j,\mathbf{a}})\right)_{\omega^{\mathbf{a}}_j}\right)\\
&\overset{(3.7)}{=}\sum_{\mathbf{a}\in\mathcal{V}_j}\left(\boldsymbol{\mathcal{K}}\nabla\rho^i_{j,\mathbf{a}},\nabla\mathcal{I}^{p_j}_j(\psi_{j,\mathbf{a}}\rho^i_{j,\mathbf{a}})\right)_{\omega^{\mathbf{a}}_j} \leq \sum_{\mathbf{a}\in\mathcal{V}_j}\frac{1}{2}\left(\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho^i_{j,\mathbf{a}}\|^2_{\omega^{\mathbf{a}}_j}+\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\mathcal{I}^{p_j}_j(\psi_{j,\mathbf{a}}\rho^i_{j,\mathbf{a}})\|^2_{\omega^{\mathbf{a}}_j}\right)\\
&\overset{(3.8c)}{\leq}\sum_{\mathbf{a}\in\mathcal{V}_j}\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho^i_{j,\mathbf{a}}\|^2_{\omega^{\mathbf{a}}_j}
\end{aligned}$$

Case test (3.8) *is not satisfied:* The above estimate is in fact an equality, by (3.41).

As we see, for both possible outcomes of test (3.8), we obtain the desired result

$$\left(\lambda^i_j\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho^i_j\|\right)^2 \overset{(3.11)}{=} \lambda^i_j\frac{(f,\rho^i_j)-(\boldsymbol{\mathcal{K}}\nabla u^i_{J,j-1},\nabla\rho^i_j)}{\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho^i_j\|^2}\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho^i_j\|^2 \leq \lambda^i_j\sum_{\mathbf{a}\in\mathcal{V}_j}\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho^i_{j,\mathbf{a}}\|^2_{\omega^{\mathbf{a}}_j}.$$

□

Remark 7.3 (Lower bound on the optimal step-sizes)**.** *As in* [Miraçi et al., 2021a, Remark 10.2]*, see Remark 10.2 in Chapter 2, by putting together the results of Lemmas 7.1, 7.2, and since $\lambda^i_j=1$ when $\rho^i_j=0$ or $j=0$, we have*

$$\lambda^i_j \geq \frac{1}{d+1} \quad 0\leq j\leq J. \tag{3.43}$$

We can now present the proof of contraction of the solver for the full-smoothing substep. The proof follows in the same way as the proof of [Miraçi et al., 2021a, Theorem 6.6], see Theorem 6.6 in Chapter 2.

Proof of part 1 of Theorem 5.4. Even though the results in Miraçi et al. [2021a], see Chapter 2, are given for the case of additive Schwarz smoothing only, we will use here the three main estimates established in the proof of [Miraçi et al., 2021a, Theorem 6.6] under minimal H^1-regularity, see Theorem 6.6 in Chapter 2. This is possible because the estimates only use the levelwise and patchwise contributions $\rho^i_{j,\mathbf{a}}$ which are constructed in the same way here, allowing us to extend the proof for case of the weighted restricted additive Schwarz smoothing. This yields $C_{\mathrm{S},1} := \sqrt{2(d+1)}C_{\mathrm{S},\boldsymbol{\mathcal{K}}}J$, $C_{\mathrm{S},2} := \sqrt{2(d+1)}C_{\mathrm{S},\boldsymbol{\mathcal{K}}}$, for $C_{\mathrm{S},\boldsymbol{\mathcal{K}}} \geq 1$ of Miraçi et al. [2021a] having the same dependencies as α, such that

$$\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\left(u_J - u^i_J\right)\right\|^2 \leq C^2_{\mathrm{S},1}\left(\eta^i_{\mathrm{alg}}\right)^2 + C^2_{\mathrm{S},2}\sum_{j=1}^{J}\sum_{\mathbf{a}\in\mathcal{V}_j}\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho^i_{j,\mathbf{a}}\right\|^2_{\omega^{\mathbf{a}}_j} \overset{(3.40)}{\leq} C^2_{\mathrm{S}}\left(\eta^i_{\mathrm{alg}}\right)^2, \tag{3.44}$$

with $C^2_{\mathrm{S}} := 2\max(C_{\mathrm{S},1}, (d+1)C_{\mathrm{S},2})$.

By Theorem 5.5, this is equivalent to (3.35) with $\alpha = \sqrt{1 - C^2_{\mathrm{S}}}$. □

Proof of Corollary 5.6. First, note that this result extends [Miraçi et al., 2021a, Corollary 6.8], see Corollary 6.8 in Chapter 2, to the weighted restricted additive Schwarz smoothing case. In the case when additive Schwarz smoothing is employed, the second equivalence in (3.39) is in fact an equality as given in [Miraçi et al., 2021a, Remark 4.5]. We obtain the desired equivalences in a closed chain of estimates

$$\begin{aligned}
\left(\eta^i_{\mathrm{alg}}\right)^2 &\overset{(3.29)}{\leq} \left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\left(u_J - u^i_J\right)\right\|^2 \\
&\overset{(3.44)}{\underset{(3.42)}{\leq}} C^2_{\mathrm{S}}\Big(\big(\lambda^i_0\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho^i_0\|\big)^2 + \sum_{j=1}^{J}\lambda^i_j\sum_{\mathbf{a}\in\mathcal{V}_j}\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho^i_{j,\mathbf{a}}\right\|^2_{\omega^{\mathbf{a}}_j}\Big) \\
&\overset{(3.15)}{\leq} 2(d+1)C^2_{\mathrm{S}}\Big(\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho^i_0\right\|^2 + \sum_{j=1}^{J}\sum_{\mathbf{a}\in\mathcal{V}_j}\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho^i_{j,\mathbf{a}}\right\|^2_{\omega^{\mathbf{a}}_j}\Big) \\
&\overset{(3.40)}{\leq} 2(d+1)^2C^2_{\mathrm{S}}\left(\eta^i_{\mathrm{alg}}\right)^2.
\end{aligned} \tag{3.45}$$

□

7.2 Proof of contraction: adaptive-smoothing substep

Let the tests (3.14)–(3.15) be satisfied. We introduce the notation $\delta_j = 1$ if the level j is marked (when $j \in \mathcal{M}$), otherwise $\delta_j = 0$. First, we present the generalization of Lemma 7.1, obtained by only working with the marked vertices.

Lemma 7.4 (Lower bound on levelwise updates by patchwise contributions)**.** *Let $u^i_J \in V^p_J$ be arbitrary. Let $j \in \mathcal{M}\setminus\{0\}$, and let $\rho^{i+\frac{1}{2}}_j$, $\lambda^{i+\frac{1}{2}}_j$ be constructed from $u^{i+\frac{1}{2}}_J$ by the adaptive-smoothing substep of the solver described in Section 3. There holds*

$$\sum_{\mathbf{a}\in\mathcal{M}_j}\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho^{i+\frac{1}{2}}_{j,\mathbf{a}}\right\|^2_{\omega^{\mathbf{a}}_j} \leq (d+1)\Big(\lambda^{i+\frac{1}{2}}_j\left\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho^{i+\frac{1}{2}}_j\right\|\Big)^2 \quad \forall 1 \leq j \leq J, \tag{3.46}$$

where for each vertex $\mathbf{a} \in \mathcal{V}_j$, $\rho_{j,\mathbf{a}}^{i+\frac{1}{2}}$ *is the solution of a local problem* (3.17).

Summing over all mesh levels and since $d+1 \geq 1$ (on $j=0$), estimate (3.46) gives:

Corollary 7.5 (Lower bound on the estimator by localized contributions)**.** *There holds*

$$\sum_{j\in\mathcal{M}} \sum_{\mathbf{a}\in\mathcal{M}_j} \|\boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \rho_{j,\mathbf{a}}^{i+\frac{1}{2}}\|_{\omega_j^{\mathbf{a}}}^2 \leq (d+1)\Big(\eta_{\mathrm{alg}}^{i+\frac{1}{2}}\Big)^2. \tag{3.47}$$

The following result is crucial in the proof of contraction of the adaptive-smoothing substep. Since the marking takes place at the end of the full-smoothing substep, which determines where the adaptive-smoothing takes place, a connection between the two substeps is needed. This is the goal of the tests (3.14)–(3.15).

Lemma 7.6 (Link between full- and adaptive-smoothing substeps)**.** *Under the adaptivity tests* (3.14)–(3.15)*, we have*

$$\sum_{j\in\mathcal{M}} \lambda_j^i \sum_{\mathbf{a}\in\mathcal{M}_j} \|\boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \rho_{j,\mathbf{a}}^{i}\|_{\omega_j^{\mathbf{a}}}^2 \leq \frac{4(d+1)^2(|\mathcal{M}|^2+1)}{(1-\gamma^2)^2}\Big(\eta_{\mathrm{alg}}^{i+\frac{1}{2}}\Big)^2. \tag{3.48}$$

Proof. We first make the connection between the two substeps, then we arrange together the terms given by the adaptive substep. The remaining full-smoothing substep terms are then treated by (3.14) and finally, we apply Young's inequality. The main term we want to estimate can be split in the two quantities below

$$\sum_{j\in\mathcal{M}} \lambda_j^i \sum_{\mathbf{a}\in\mathcal{M}_j} \|\boldsymbol{\mathcal{K}}^{\frac{1}{2}} \nabla \rho_{j,\mathbf{a}}^{i}\|_{\omega_j^{\mathbf{a}}}^2 = \delta_0(\boldsymbol{\mathcal{K}}\nabla\rho_0^i, \nabla\rho_0^i) + \sum_{j\in\mathcal{M}\setminus\{0\}} \lambda_j^i \sum_{\mathbf{a}\in\mathcal{M}_j} (\boldsymbol{\mathcal{K}}\nabla\rho_{j,\mathbf{a}}^i, \nabla\rho_{j,\mathbf{a}}^i)_{\omega_j^{\mathbf{a}}}.$$

First,

$$\begin{aligned}
\delta_0(\boldsymbol{\mathcal{K}}\nabla\rho_0^i, \nabla\rho_0^i) &\overset{(3.6),(3.23)}{=} \delta_0\Big((f,\rho_0^i) - (\boldsymbol{\mathcal{K}}\nabla u_J^{i+\frac{1}{2}}, \nabla\rho_0^i) + \sum_{j=0}^{J} \lambda_j^i (\boldsymbol{\mathcal{K}}\nabla\rho_j^i, \nabla\rho_0^i)\Big) \\
&\overset{(3.16)}{=} \delta_0\Big((\boldsymbol{\mathcal{K}}\nabla\rho_0^{i+\frac{1}{2}}, \nabla\rho_0^i) + \sum_{j=0}^{J} \lambda_j^i (\boldsymbol{\mathcal{K}}\nabla\rho_j^i, \nabla\rho_0^i)\Big) \\
&\leq \delta_0 \frac{1}{2(1-\gamma^2)} \|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_0^{i+\frac{1}{2}}\|^2 + \delta_0 \frac{1-\gamma^2}{2} \|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_0^i\|^2 + \delta_0 \sum_{j=0}^{J} \lambda_j^i (\boldsymbol{\mathcal{K}}\nabla\rho_j^i, \nabla\rho_0^i).
\end{aligned}$$

Second,

$$\begin{aligned}
\sum_{j\in\mathcal{M}\setminus\{0\}} \lambda_j^i \sum_{\mathbf{a}\in\mathcal{M}_j} (\boldsymbol{\mathcal{K}}\nabla\rho_{j,\mathbf{a}}^i, \nabla\rho_{j,\mathbf{a}}^i)_{\omega_j^{\mathbf{a}}} &\overset{(3.7)}{=} \sum_{j\in\mathcal{M}\setminus\{0\}} \lambda_j^i \sum_{\mathbf{a}\in\mathcal{M}_j} \Big((f,\rho_{j,\mathbf{a}}^i)_{\omega_j^{\mathbf{a}}} - (\boldsymbol{\mathcal{K}}\nabla u_{J,j-1}^i, \nabla\rho_{j,\mathbf{a}}^i)_{\omega_j^{\mathbf{a}}}\Big) \\
&\overset{(3.12)}{=} \sum_{j\in\mathcal{M}\setminus\{0\}} \lambda_j^i \sum_{\mathbf{a}\in\mathcal{M}_j} \Big((f,\rho_{j,\mathbf{a}}^i)_{\omega_j^{\mathbf{a}}} - (\boldsymbol{\mathcal{K}}\nabla u_J^i, \nabla\rho_{j,\mathbf{a}}^i)_{\omega_j^{\mathbf{a}}} - \sum_{k=0}^{j-1} \lambda_k^i (\boldsymbol{\mathcal{K}}\nabla\rho_k^i, \nabla\rho_{j,\mathbf{a}}^i)_{\omega_j^{\mathbf{a}}}\Big) \\
&\overset{(3.23)}{=} \sum_{j\in\mathcal{M}\setminus\{0\}} \lambda_j^i \sum_{\mathbf{a}\in\mathcal{M}_j} \Big((f,\rho_{j,\mathbf{a}}^i)_{\omega_j^{\mathbf{a}}} - (\boldsymbol{\mathcal{K}}\nabla u_J^{i+1}, \nabla\rho_{j,\mathbf{a}}^i)_{\omega_j^{\mathbf{a}}}
\end{aligned}$$

$$
\begin{aligned}
&+\sum_{k=0}^{J}\lambda_k^i(\boldsymbol{\mathcal{K}}\nabla\rho_k^i,\nabla\rho_{j,\mathbf{a}}^i)_{\omega_j^{\mathbf{a}}}-\sum_{k=0}^{j-1}\lambda_k^i(\boldsymbol{\mathcal{K}}\nabla\rho_k^i,\nabla\rho_{j,\mathbf{a}}^i)_{\omega_j^{\mathbf{a}}}\Big)\\
&\overset{(3.22)}{=}\sum_{j\in\mathcal{M}\setminus\{0\}}\lambda_j^i\sum_{\mathbf{a}\in\mathcal{M}_j}\Big((f,\rho_{j,\mathbf{a}}^i)_{\omega_j^{\mathbf{a}}}-(\boldsymbol{\mathcal{K}}\nabla u_{J,j-1}^{i+\frac{1}{2}},\nabla\rho_{j,\mathbf{a}}^i)_{\omega_j^{\mathbf{a}}}\\
&\quad+\sum_{\substack{l=0\\ l\in\mathcal{M}}}^{j-1}\lambda_l^{i+\frac{1}{2}}(\boldsymbol{\mathcal{K}}\nabla\rho_l^{i+\frac{1}{2}},\nabla\rho_{j,\mathbf{a}}^i)_{\omega_j^{\mathbf{a}}}+\sum_{k=j}^{J}\lambda_k^i(\boldsymbol{\mathcal{K}}\nabla\rho_k^i,\nabla\rho_{j,\mathbf{a}}^i)_{\omega_j^{\mathbf{a}}}\Big)\\
&\overset{(3.17)}{=}\sum_{j\in\mathcal{M}\setminus\{0\}}\lambda_j^i\sum_{\mathbf{a}\in\mathcal{M}_j}\Big((\boldsymbol{\mathcal{K}}\nabla\rho_{j,\mathbf{a}}^{i+\frac{1}{2}},\nabla\rho_{j,\mathbf{a}}^i)_{\omega_j^{\mathbf{a}}}\\
&\quad+\sum_{l=0}^{j-1}\delta_l\lambda_l^{i+\frac{1}{2}}(\boldsymbol{\mathcal{K}}\nabla\rho_l^{i+\frac{1}{2}},\nabla\rho_{j,\mathbf{a}}^i)_{\omega_j^{\mathbf{a}}}+\sum_{k=j}^{J}\lambda_k^i(\boldsymbol{\mathcal{K}}\nabla\rho_k^i,\nabla\rho_{j,\mathbf{a}}^i)_{\omega_j^{\mathbf{a}}}\Big)\\
&\le\sum_{j\in\mathcal{M}\setminus\{0\}}\lambda_j^i\sum_{\mathbf{a}\in\mathcal{M}_j}\Big(\frac{1}{1-\gamma^2}\big\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_{j,\mathbf{a}}^{i+\frac{1}{2}}\big\|_{\omega_j^{\mathbf{a}}}^2+\frac{1-\gamma^2}{4}\big\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_{j,\mathbf{a}}^i\big\|_{\omega_j^{\mathbf{a}}}^2\\
&+\frac{1}{1-\gamma^2}\Big\|\sum_{l=0}^{j-1}\delta_l\lambda_l^{i+\frac{1}{2}}\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_l^{i+\frac{1}{2}}\Big\|_{\omega_j^{\mathbf{a}}}^2+\frac{1-\gamma^2}{4}\big\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_{j,\mathbf{a}}^i\big\|_{\omega_j^{\mathbf{a}}}^2+\sum_{k=j}^{J}\lambda_k^i(\boldsymbol{\mathcal{K}}\nabla\rho_k^i,\nabla\rho_{j,\mathbf{a}}^i)_{\omega_j^{\mathbf{a}}}\Big)
\end{aligned}
$$

We return to the main estimate by summing the two estimates and using the result of Test (3.14)

$$
\begin{aligned}
\sum_{j\in\mathcal{M}}\lambda_j^i\sum_{\mathbf{a}\in\mathcal{M}_j}\big\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_{j,\mathbf{a}}^i\big\|_{\omega_j^{\mathbf{a}}}^2 &\le \frac{1}{1-\gamma^2}\sum_{j\in\mathcal{M}}\lambda_j^i\sum_{\mathbf{a}\in\mathcal{M}_j}\big\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_{j,\mathbf{a}}^{i+\frac{1}{2}}\big\|_{\omega_j^{\mathbf{a}}}^2\\
&+\frac{1-\gamma^2}{2}\sum_{j\in\mathcal{M}}\lambda_j^i\sum_{\mathbf{a}\in\mathcal{M}_j}\big\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_{j,\mathbf{a}}^i\big\|_{\omega_j^{\mathbf{a}}}^2+\frac{1}{1-\gamma^2}\sum_{j\in\mathcal{M}\setminus\{0\}}\lambda_j^i\sum_{\mathbf{a}\in\mathcal{M}_j}\Big\|\sum_{\substack{l=0\\ l\in\mathcal{M}}}^{j-1}\lambda_l^{i+\frac{1}{2}}\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_l^{i+\frac{1}{2}}\Big\|_{\omega_j^{\mathbf{a}}}^2\\
&+\sum_{j\in\mathcal{M}\setminus\{0\}}\lambda_j^i\sum_{\mathbf{a}\in\mathcal{M}_j}\Big(\sum_{k=j}^{J}\lambda_k^i(\boldsymbol{\mathcal{K}}\nabla\rho_k^i,\nabla\rho_{j,\mathbf{a}}^i)_{\omega_j^{\mathbf{a}}}\Big)+\delta_0\sum_{j=0}^{J}\lambda_j^i(\boldsymbol{\mathcal{K}}\nabla\rho_j^i,\nabla\rho_0^i)
\end{aligned}
$$

$$
\begin{aligned}
&\overset{(3.14)}{\le}\frac{1}{1-\gamma^2}\sum_{j\in\mathcal{M}}\lambda_j^i\sum_{\mathbf{a}\in\mathcal{M}_j}\big\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_{j,\mathbf{a}}^{i+\frac{1}{2}}\big\|_{\omega_j^{\mathbf{a}}}^2+\frac{1-\gamma^2}{2}\sum_{j\in\mathcal{M}}\lambda_j^i\sum_{\mathbf{a}\in\mathcal{M}_j}\big\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_{j,\mathbf{a}}^i\big\|_{\omega_j^{\mathbf{a}}}^2\\
&\quad+\frac{1}{1-\gamma^2}\sum_{j\in\mathcal{M}\setminus\{0\}}\lambda_j^i\sum_{\mathbf{a}\in\mathcal{M}_j}\Big\|\sum_{\substack{l=0\\ l\in\mathcal{M}}}^{j-1}\lambda_l^{i+\frac{1}{2}}\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_l^{i+\frac{1}{2}}\Big\|_{\omega_j^{\mathbf{a}}}^2+\gamma^2\sum_{j\in\mathcal{M}}\lambda_j^i\sum_{\mathbf{a}\in\mathcal{M}_j}\big\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_{j,\mathbf{a}}^i\big\|_{\omega_j^{\mathbf{a}}}^2.
\end{aligned}
$$

Rearranging the terms, we have

$$
\begin{aligned}
\frac{1-\gamma^2}{2}\sum_{j\in\mathcal{M}}\lambda_j^i\sum_{\mathbf{a}\in\mathcal{M}_j}\big\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_{j,\mathbf{a}}^i\big\|_{\omega_j^{\mathbf{a}}}^2 &\le \frac{1}{1-\gamma^2}\sum_{j\in\mathcal{M}}\lambda_j^i\sum_{\mathbf{a}\in\mathcal{M}_j}\big\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_{j,\mathbf{a}}^{i+\frac{1}{2}}\big\|_{\omega_j^{\mathbf{a}}}^2\\
&+\frac{1}{1-\gamma^2}\sum_{j\in\mathcal{M}\setminus\{0\}}\lambda_j^i\sum_{\mathbf{a}\in\mathcal{M}_j}\Big\|\sum_{\substack{l=0\\ l\in\mathcal{M}}}^{j-1}\lambda_l^{i+\frac{1}{2}}\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_l^{i+\frac{1}{2}}\Big\|_{\omega_j^{\mathbf{a}}}^2,
\end{aligned}
$$

leading to

$$\sum_{j\in\mathcal{M}}\lambda_j^i\sum_{\mathbf{a}\in\mathcal{M}_j}\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_{j,\mathbf{a}}^i\|_{\omega_j^{\mathbf{a}}}^2 \overset{(3.15)}{\underset{(3.47)}{\leq}} \frac{4(d+1)^2}{(1-\gamma^2)^2}\Big(\Big(\eta_{\mathrm{alg}}^{i+\frac{1}{2}}\Big)^2+\sum_{j\in\mathcal{M}\setminus\{0\}}\Big\|\sum_{\substack{l=0\\ l\in\mathcal{M}}}^{j-1}\lambda_l^{i+\frac{1}{2}}\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_l^{i+\frac{1}{2}}\Big\|^2\Big)$$

$$\leq \frac{4(d+1)^2}{(1-\gamma^2)^2}\Big(\Big(\eta_{\mathrm{alg}}^{i+\frac{1}{2}}\Big)^2+\sum_{j\in\mathcal{M}\setminus\{0\}}|\mathcal{M}|\sum_{\substack{l=0\\ l\in\mathcal{M}}}^{j-1}\Big\|\lambda_l^{i+\frac{1}{2}}\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_l^{i+\frac{1}{2}}\Big\|^2\Big)$$

$$\leq \frac{4(d+1)^2}{(1-\gamma^2)^2}\Big(\Big(\eta_{\mathrm{alg}}^{i+\frac{1}{2}}\Big)^2+|\mathcal{M}|^2\sum_{l\in\mathcal{M}}\Big(\lambda_l^{i+\frac{1}{2}}\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_l^{i+\frac{1}{2}}\|\Big)^2\Big)\overset{(3.28)}{=}\frac{4(d+1)^2(|\mathcal{M}|^2+1)}{(1-\gamma^2)^2}\Big(\eta_{\mathrm{alg}}^{i+\frac{1}{2}}\Big)^2.$$

where $|\mathcal{M}|$ denotes the number of marked levels. □

We can now prove the contraction of the adaptive-smoothing substep below.

Proof of part 2 of Theorem 5.4. We divide the proof into two steps.

Step 1. We prove that there holds:

$$\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J-u_J^{i+\frac{1}{2}})\|^2\leq\tilde{\beta}^2\Big(\eta_{\mathrm{alg}}^{i+\frac{1}{2}}\Big)^2. \tag{3.49}$$

By Theorem 5.5, the efficiency of the estimator η_{alg}^i is equivalent to error contraction of the full-smoothing substep. Using the equivalence error–localized contributions of Corollary 5.6/(3.45), the bulk-chasing criterion (3.13), and the result of Lemma 7.6,

$$\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J-u_J^{i+1})\|^2 \overset{\text{Theorem 5.5}}{\leq} \alpha^2\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla(u_J-u_J^i)\|^2$$

$$\overset{(3.45)}{\leq} \alpha^2C_{\mathrm{S}}^2\Big(\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_0^i\|^2+\sum_{j=1}^{J}\lambda_j^i\sum_{\mathbf{a}\in\mathcal{V}_j}\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_{j,\mathbf{a}}^i\|_{\omega_j^{\mathbf{a}}}^2\Big)$$

$$\overset{(3.13)}{\leq}\frac{\alpha^2C_{\mathrm{S}}^2}{\theta^2}\sum_{j\in\mathcal{M}}\lambda_j^i\sum_{\mathbf{a}\in\mathcal{M}_j}\|\boldsymbol{\mathcal{K}}^{\frac{1}{2}}\nabla\rho_{j,\mathbf{a}}^i\|_{\omega_j^{\mathbf{a}}}^2\overset{(3.48)}{\leq}\frac{4\alpha^2C_{\mathrm{S}}^2(d+1)^2(|\mathcal{M}|^2+1)}{\theta^2(1-\gamma^2)^2}\Big(\eta_{\mathrm{alg}}^{i+\frac{1}{2}}\Big)^2,$$

giving the desired result with

$$\tilde{\beta}^2=\frac{4\alpha^2C_{\mathrm{S}}^2(d+1)^2(|\mathcal{M}|^2+1)}{\theta^2(1-\gamma^2)^2}.$$

Thus, the estimator $\eta_{\mathrm{alg}}^{i+\frac{1}{2}}$ (guaranteed lower bound by (3.30)), is p-robustly efficient.

Step 2. By Theorem 5.5, (3.49) is equivalent to (3.36) with $\tilde{\alpha}=\sqrt{1-\tilde{\beta}^2}$. □

8 Conclusions

In this work, we have presented an adaptive multilevel solver whose adaptive process is supervised by an a posteriori estimator of the algebraic error. We showed that both full-smoothing and adaptive-smoothing substeps of the solver contract the error robustly with respect to the polynomial degree of approximation p, under the decision tests (3.14)–(3.15) for the latter. To the best of the authors' knowledge,

this is the first work where adaptive smoothing not necessarily everywhere in the meshes is proven to contract the algebraic error, and moreover does so in a p-robust way. Numerical experiments indicate that the adaptivity can provide an interesting speed-up and is worth considering in practice. Furthermore, for a hierarchy of meshes obtained through uniform refinement, the solver appears numerically robust with respect to the number of levels in the hierarchy as well as the jump in the diffusion coefficient. Further work would explore how this can be rigorously proven.

Acknowledgement

We would like to thank the anonymous referees for their useful suggestions concerning this manuscript.

Chapter 4

p-robust multilevel and domain decomposition methods with optimal step-sizes for mixed finite element discretizations of elliptic problems

We present in this chapter the results of an article in preparation. This work is a collaboration with Martin Vohralík and Ivan Yotov.

Contents

1 Introduction . . . 115

2 Model problem and its mixed finite element discretization 116

2.1 Discrete mixed finite element problem . . . 117

3 Multilevel setting . . . 117

3.1 A hierarchy of meshes . . . 117

3.2 A hierarchy of spaces . . . 118

4 An a-posteriori-steered multigrid solver . . . 119

4.1 Multigrid solver . . . 119

4.2 A posteriori estimator on the algebraic error . . . 121

5 An a-posteriori-steered domain decomposition solver . . . 121

5.1 Two-level hierarchy . . . 122

5.2 Two-level iterative solver: overlapping additive Schwarz . . . 122

5.3 A posteriori estimator on the algebraic error . . . 123

6 Main results . . . 124

7 Proofs of the main results . . . 125

7.1 Multilevel setting results . . . 128

7.2 Two-level domain decomposition setting results . . . 128

8 Conclusions . . . 130

Abstract

In this work, we develop algebraic solvers for linear systems arising from the discretization of second-order elliptic problems by mixed finite elements method of arbirary polynomial degree $k \geq 1$. We present a multigrid and a two-level domain decomposition approach, which are guided by their respective a posteriori estimators of the algebraic error. First, we extend the results given in [HAL Preprint 02494538, 2020] to a mixed finite element setting. Extending the multigrid procedure itself is rather natural, however extending the theoretical results requires a k-robust multilevel stable decomposition of the velocity space. For this, we use the fact that in two space dimensions, the velocity space is the curl of a stream-function space, for which the previous results apply. This allows us to prove, in two space dimensions, that our multigrid solver contracts the algebraic error at each iteration k-robustly and that the associated a posteriori estimator is k-robustly efficient. Next, we use this multilevel methodology to define a two-level domain decomposition method, where the subdomains consist in the overlapping patches of coarse level elements sharing a common coarse level vertex. We prove, in two space dimensions, that the domain decomposition method also contracts the algebraic error k-robustly in each iteration and that the associated a posteriori estimator is k-robustly efficient.

1 Introduction

In many physical problems studying fluid flows, the main focus is on the velocity variable. While different discretization methods can be used to approximate the fluid velocity, the mixed finite element method, see e.g. Boffi et al. [2013], has been one of the most attractive approaches because of the accuracy, robustness, and instantaneous local mass conservation it provides. In order to benefit from these advantages, suitable iterative linear solvers should be also considered. One difficulty is that the saddle-point problem obtained from the discretization leads to an indefinite linear system, see e.g. Benzi et al. [2005] or Brenner [2009]. Some references proposing and studying multilevel and domain decomposition methods arising from mixed discretizations include, e.g., Glowinski and Wheeler [1988], Brenner [1992], Ewing and Wang [1992], Mathew [1993], Ewing and Wang [1994], Cowsar et al. [1995], Chen [1996], Brenner et al. [2018].

It is in particular possible rewrite the problem such that, if one first constructs a suitable initial approximation of the velocity which satisfies a divergence constraint, then only a symmetric and positive definite divergence-free problem remains. This approach has in particular been taken in Ewing and Wang [1992], Mathew [1993], and Ewing and Wang [1994], and we follow it here.

In this work, we further rely on previous work Miraçi et al. [2021a], done for a conforming finite element discretization and extend it to the present mixed finite element setting. We first present a geometric *multigrid* solver, whose iteration consists in a V-cycle with zero pre- and one post-smoothing step with additive Schwarz (block-Jacobi) as a smoother and *optimal level-wise step-sizes* given by line search at the error correction stage. Next, we use the methodology developed for multigrid to define a two-level *domain decomposition* method. In particular, our methods do

not need any damping or relaxation parameters which might require tuning. Associated to these solvers, we present a posteriori estimates for the algebraic error which are easily defined from the same construction that provides the solvers. In this sense, we refer to the presented linear solvers as *a-posteriori-steered*.

We prove that in two space dimensions, the introduced multigrid solver and the domain decomposition method contract the algebraic error at each iteration independently of the polynomial degree k, i.e. *k-robustly*. Moreover, we show that the associated a posteriori estimators are k-robustly efficient. In fact, proving the k-robust error contraction of our solvers is equivalent to proving that the associated a posteriori estimators are k-robustly efficient. A crucial ingredient needed for our analysis is a polynomial-degree-robust *multilevel stable splitting*, which is given in Miraçi et al. [2020] by combining the k-robust one-level stable splitting achieved in Schöberl et al. [2008] and a multilevel piecewise affine stable splitting from Xu et al. [2009]. In order to adapt this result in our case, we use the connection in two space dimensions between discrete stream-function spaces and discrete velocity spaces, see e.g. [Boffi et al., 2013, Corollary 2.3.2]. Finally, we point out that the extension of the proofs for the solver contraction to three space dimensions is possible by using the stable decomposition as in, e.g., [Cai et al., 2003, Lemma 5.1], however with this approach we cannot theoretically prove the k-robustness of the solver.

This work is organized as follows. In Section 2 we present the model problem and its mixed finite element discretization. The multilevel setting and assumptions used in our theory are given in Section 3. In Section 4 we present the a-posteriori-steered multigrid solver with its associated a posteriori estimator of the algebraic error and in Section 5 we similarly present the domain decomposition method with the associated a posteriori estimator. Our main results are summarized in Section 6 and the proofs are given in Section 7. Finally, we present some concluding remarks in Section 8.

2 Model problem and its mixed finite element discretization

We consider an elliptic partial differential equation in a mixed form modeling single phase flow in porous media. Let $\Omega \subset \mathbb{R}^d$, $d = 2, 3$, be a polytopal domain with Lipschitz boundary. The governing equations are of the same type as in (1), but we impose here a homogeneous Neumann boundary condition and write the problem as

$$\mathbf{u} = -\boldsymbol{\mathcal{K}}\nabla p, \quad \nabla \cdot \mathbf{u} = f \quad \text{in } \Omega, \quad \mathbf{u} \cdot \mathbf{n} = 0 \quad \text{on } \partial\Omega, \tag{4.1}$$

where p is the fluid pressure, $\mathbf{u}$ is the Darcy velocity, $\boldsymbol{\mathcal{K}}$ is a symmetric and positive definite tensor representing the rock permeability divided by the fluid viscosity, f is the source term such that $(f, 1) = 0$ in Ω, and $\mathbf{n}$ is the normal vector on $\partial\Omega$. Let

$$\begin{aligned} \mathbf{V} &= \mathbf{H}_0(\mathrm{div};\Omega) := \{\mathbf{v} \in \mathbf{H}(\mathrm{div};\Omega), \mathbf{v} \cdot \mathbf{n} = 0 \text{ on } \partial\Omega \text{ in appropriate sense}\}, \\ W &= L^2_0(\Omega) := \{w \in L^2(\Omega), (w, 1) = 0 \text{ in } \Omega\}. \end{aligned}$$

Let $(\cdot,\cdot)_S$ and $\|\cdot\|_S$, $S \subset \mathbb{R}^d$, be the $L^2(S)$ inner product and norm, respectively, where we omit the subscript if $S = \Omega$. The weak formulation of (4.1), see e.g. Boffi

et al. [2013], is: find $\mathbf{u} \in \mathbf{V}$ and $p \in W$ such that

$$(\boldsymbol{\mathcal{K}}^{-1}\mathbf{u}, \mathbf{v}) - (p, \nabla \cdot \mathbf{v}) = 0 \quad \forall\, \mathbf{v} \in \mathbf{V}, \tag{4.2a}$$

$$(\nabla \cdot \mathbf{u}, w) = (f, w) \quad \forall\, w \in W. \tag{4.2b}$$

For $g \in L^2(\Omega)$, let $\mathbf{V}^g = \{\mathbf{v} \in \mathbf{V} : \nabla \cdot \mathbf{v} = g\}$. Problem (4.2) can be written equivalently as: find $\mathbf{u} \in \mathbf{V}^f$ such that

$$(\boldsymbol{\mathcal{K}}^{-1}\mathbf{u}, \mathbf{v}) = 0 \quad \forall\, \mathbf{v} \in \mathbf{V}^0. \tag{4.3}$$

2.1 Discrete mixed finite element problem

In order to discretize the model problem (4.2), we first introduce a shape-regular simplicial mesh $\mathcal{T}_h$ partitioning Ω. Then, we fix an integer $k \geq 1$ which denotes the polynomial degree used in our mixed finite element (MFE) spaces $\mathbf{V}_h \times W_h \subset \mathbf{V} \times W$. For our setting, we shall work with Raviart–Thomas (**RT**) spaces, see Raviart and Thomas [1977], though Brezzi–Douglas–Marini (**BDM**) spaces, see Brezzi et al. [1986], can also be employed. Define

$$\mathbf{V}_h := \{\mathbf{v}_h \in \mathbf{V}, \mathbf{v}|_K \in \mathbf{RT}_k(K)\ \forall K \in \mathcal{T}_h\}, \tag{4.4}$$

$$W_h := \{w_h \in W,\ w_h|_K \in \mathbb{P}_k(K)\ \forall K \in \mathcal{T}_h\}. \tag{4.5}$$

We search for $\mathbf{u}_h \in \mathbf{V}_h$ and $p_h \in W_h$ such that

$$(\boldsymbol{\mathcal{K}}^{-1}\mathbf{u}_h, \mathbf{v}_h) - (p_h, \nabla \cdot \mathbf{v}_h) = 0 \quad \forall\, \mathbf{v}_h \in \mathbf{V}_h, \tag{4.6a}$$

$$(\nabla \cdot \mathbf{u}_h, w_h) = (f, w_h) \quad \forall\, w_h \in W_h. \tag{4.6b}$$

Denoting $\mathbf{V}_h^g = \{\mathbf{v}_h \in \mathbf{V}_h : (\nabla \cdot \mathbf{v}_h, w_h) = (g, w_h)\ \forall\, w_h \in W_h\}$, the method (4.6) can again be written equivalently as: find $\mathbf{u}_h \in \mathbf{V}_h^f$ such that

$$(\boldsymbol{\mathcal{K}}^{-1}\mathbf{u}_h, \mathbf{v}_h) = 0 \quad \forall\, \mathbf{v}_h \in \mathbf{V}_h^0. \tag{4.7}$$

3 Multilevel setting

We introduce here the assumptions on the hierarchy of meshes and the associated hierarchy of spaces used in this manuscript.

3.1 A hierarchy of meshes

We consider a hierarchy of nested matching simplicial meshes of Ω, $\{\mathcal{T}_j\}_{0 \leq j \leq J}$, $J \geq 1$, where $\mathcal{T}_j$ is a refinement of $\mathcal{T}_{j-1}$, $1 \leq j \leq J$, and $\mathcal{T}_J = \mathcal{T}_h$. For any element $K \in \mathcal{T}_j$, we denote $h_K := \operatorname{diam}(K)$. We always work with meshes that are shape-regular:

Assumption 3.1 (Mesh shape regularity)**.** *There exists $\kappa_{\mathcal{T}} > 0$ such that*

$$\max_{K \in \mathcal{T}_j} \frac{h_K}{\rho_K} \leq \kappa_{\mathcal{T}} \text{ for all } 0 \leq j \leq J, \tag{4.8}$$

where ρ_K denotes the diameter of the largest ball inscribed in K.

Next, we will work in one of the two settings corresponding to the two assumptions below. In the first setting, we assume:

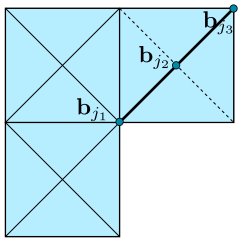
Assumption 3.2 (Refinement strength and mesh quasi-uniformity)**.** *There exists a fixed positive real number* $0 < C_{\text{ref}} \leq 1$*, such that for all* $j \in \{1, \ldots, J\}$*, for all* $K \in \mathcal{T}_{j-1}$ *and* $K^* \in \mathcal{T}_j$ *such that* $K^* \subset K$*, there holds*

$$C_{\text{ref}} h_K \leq h_{K^*} \leq h_K. \tag{4.9}$$

There further exists a fixed positive real number $0 < C_{\text{qu}} \leq 1$*, such that for all* $j \in \{0, \ldots, J\}$ *and for all* $K \in \mathcal{T}_j$*, there holds*

$$C_{\text{qu}} h_j \leq h_K \leq h_j. \tag{4.10}$$

In the second setting, we assume that our hierarchy is generated from a quasi-uniform coarse mesh by a series of bisections, e.g. newest vertex bisection, cf. Sewell [1972]. In this case, refining one edge of $\mathcal{T}_{j-1}$, for $j \in \{1, \ldots, J\}$, leads to a new finer mesh $\mathcal{T}_j$. We denote by $\mathcal{B}_j \subset \mathcal{V}_j$ the set consisting of the new vertex obtained after the bisection together with its two neighbors on the refinement edge; see Figure 4.1. We also denote by $h_{\mathcal{B}_j}$ the maximal diameter of elements having a vertex in the set $\mathcal{B}_j$, for $j \in \{1, \ldots, J\}$. We assume:



$\mathcal{T}_j$ obtained by a bisection of $\mathcal{T}_{j-1}$
neighboring vertices on the refinement edge $\mathbf{b}_{j_1}, \mathbf{b}_{j_3}$
new vertex after refinement $\mathbf{b}_{j_2}$
$\mathcal{B}_j = \{\mathbf{b}_{j_1}, \mathbf{b}_{j_2}, \mathbf{b}_{j_3}\} \subset \mathcal{V}_j$

Figure 4.1: Illustration of the set $\mathcal{B}_j$; the refinement $\mathcal{T}_j$ (dotted lines) of the mesh $\mathcal{T}_{j-1}$ (full lines).

Assumption 3.3 (Local refinement strength and the coarsest mesh quasi-uniformity of bisection-generated meshes)**.** *The coarsest mesh* $\mathcal{T}_0$ *is a conforming quasi-uniform mesh in the sense of* (4.10)*, with parameter* $0 < C^0_{\text{qu}} \leq 1$*. The (possibly highly graded) conforming mesh* $\mathcal{T}_J$ *is generated from* $\mathcal{T}_0$ *by a series of bisections. There exists a fixed positive real number* $0 < C_{\text{loc,qu}} \leq 1$ *such that for all* $j \in \{1, \ldots, J\}$*, there holds*

$$C_{\text{loc,qu}} h_{\mathcal{B}_j} \leq h_K \leq h_{\mathcal{B}_j} \quad \forall K \in \mathcal{T}_j \text{ such that a vertex of } K \text{ belongs to } \mathcal{B}_j. \tag{4.11}$$

3.2 A hierarchy of spaces

We consider a hierarchy of nested mixed finite element spaces associated to the nested meshes. First, fix a sequence of increasing polynomial degrees $0 = k_0 \leq k_1 \leq \ldots \leq k_J = k$. Then, for $0 \leq j \leq J$, define the level-wise mixed finite element space

$$\mathbf{V}_j := \{\mathbf{v}_j \in \mathbf{V},\ \mathbf{v}_j|_K \in \mathbf{RT}_{k_j}(K)\ \forall K \in \mathcal{T}_j\}. \tag{4.12}$$

We also define, for $1 \leq j \leq J$, the level-wise divergence-free discrete spaces

$$\mathbf{V}^0_j := \{\mathbf{v}_j \in \mathbf{V},\ \nabla \cdot \mathbf{v}_j = 0,\ \mathbf{v}_j|_K \in \mathbf{RT}_{k_j}(K)\ \forall K \in \mathcal{T}_j\}. \tag{4.13}$$

4 An a-posteriori-steered multigrid solver

Thanks to the multilevel setting we introduced, we can now develop a multigrid solver to tackle the discrete problem (4.6).

4.1 Multigrid solver

The solver we develop involves solving in each iteration a coarse grid problem and local problems on patches of elements of the other levels of the hierarchy. We begin with the definition of the patches. Let $\mathcal{V}_j$ be the set of vertices of the mesh $\mathcal{T}_j$ and let $\mathcal{V}_K$ be the set of vertices of an element K of $\mathcal{T}_j$. Given a vertex $\mathbf{a} \in \mathcal{V}_j$, $j \in \{0, \dots, J\}$, we denote the patch associated to $\mathbf{a}$ by $\mathcal{T}_j^{\mathbf{a}}$, defined by

$$\mathcal{T}_j^{\mathbf{a}} := \{K \in \mathcal{T}_j, \mathbf{a} \in \mathcal{V}_K\}. \tag{4.14}$$

Denote the corresponding open patch subdomain by $\omega_j^{\mathbf{a}}$. Define the local MFE spaces on $\omega_j^{\mathbf{a}}$ associated with $\mathcal{T}_j$ as

$$\mathbf{V}_j^{\mathbf{a}} := \{\mathbf{v}_j \in \mathbf{V}_j|_{\omega_j^{\mathbf{a}}},\ \mathbf{v}_j \cdot \mathbf{n} = 0 \text{ on } \partial\omega_j^{\mathbf{a}}\}. \tag{4.15}$$

Finally, define

$$\mathbf{V}_j^{\mathbf{a},0} = \{\mathbf{v}_{j,\mathbf{a}} \in \mathbf{V}_j^{\mathbf{a}},\ \nabla \cdot \mathbf{v}_{j,\mathbf{a}} = 0\}. \tag{4.16}$$

Remark 4.1 (Other patches)**.** *Other types of patches can also be considered. For example, in* Miraçi et al. [2020]*, see Chapter 1, larger patches, obtained by combining all elements in the coarser mesh $\mathcal{T}_{j-1}$ that share a vertex in $\mathcal{V}_{j-1}$, are also studied. The trade-off is that there are fewer of these larger patches. The theoretical results also apply in this case. For simplicity we limit the presentation here to the smaller patches.*

We now proceed with the definition of the iterative solver.

Algorithm 4.2 (Multigrid solver)**.**

1. *Initialize $\mathbf{u}_h^0 \in \mathbf{V}_h^f$, e.g., as done in* [Ewing and Wang, 1992, Theorem 3.1]*, and let $i := 0$.*

2. *Perform the following steps (a)–(d):*

   (a) *Solve the coarse grid global residual problem: find $\boldsymbol{\rho}_0^i \in \mathbf{V}_0^0$ such that*

   $$(\boldsymbol{\mathcal{K}}^{-1}\boldsymbol{\rho}_0^i, \mathbf{v}_0) = -(\boldsymbol{\mathcal{K}}^{-1}\mathbf{u}_h^i, \mathbf{v}_0) \quad \forall \mathbf{v}_0 \in \mathbf{V}_0^0 \tag{4.17}$$

   Set $\lambda_0^i := 1$ and $\mathbf{u}_0^i := \mathbf{u}_h^i + \lambda_0^i \boldsymbol{\rho}_0^i$.

   (b) *For $1 \le j \le J$:*

   Compute local updates $\boldsymbol{\rho}_{j,\mathbf{a}}^i \in \mathbf{V}_j^{\mathbf{a},0}$ as solutions of the patch local residual problems, for all $\mathbf{a} \in \mathcal{V}_j$,

   $$(\boldsymbol{\mathcal{K}}^{-1}\boldsymbol{\rho}_{j,\mathbf{a}}^i, \mathbf{v}_{j,\mathbf{a}})_{\omega_j^{\mathbf{a}}} = -(\boldsymbol{\mathcal{K}}^{-1}\mathbf{u}_{j-1}^i, \mathbf{v}_{j,\mathbf{a}})_{\omega_j^{\mathbf{a}}} \quad \forall \mathbf{v}_{j,\mathbf{a}} \in \mathbf{V}_j^{\mathbf{a},0}. \tag{4.18}$$

Define $\boldsymbol{\rho}_j^i \in \mathbf{V}_j^0$ *by*

$$\boldsymbol{\rho}_j^i := \sum_{\mathbf{a}\in\mathcal{V}_j} \boldsymbol{\rho}_{j,\mathbf{a}}^i. \tag{4.19}$$

If $\boldsymbol{\rho}_j^i \neq 0$*, define the level step size by the line search*

$$\lambda_j^i := -\frac{(\boldsymbol{\mathcal{K}}^{-1}\mathbf{u}_{j-1}^i, \boldsymbol{\rho}_j^i)}{\left\|\boldsymbol{\mathcal{K}}^{-1/2}\boldsymbol{\rho}_j^i\right\|^2}, \tag{4.20}$$

otherwise set $\lambda_j^i := 1$*. Define the level update*

$$\mathbf{u}_j^i := \mathbf{u}_{j-1}^i + \lambda_j^i \boldsymbol{\rho}_j^i. \tag{4.21}$$

(c) *Set the next iterate* $\mathbf{u}_h^{i+1} := \mathbf{u}_h^i$.

(d) *If* $\mathbf{u}_h^{i+1} = \mathbf{u}_J^i$*, then stop the solver. Otherwise set* $i := i+1$ *and go to step 2(a).*

Remark 4.3 (Compact formulas)**.** *The update for the new iterate can be written in the compact form*

$$\mathbf{u}_h^{i+1} = \mathbf{u}_h^i + \sum_{j=0}^{J} \lambda_j^i \boldsymbol{\rho}_j^i \overset{(4.19)}{=} \mathbf{u}_h^i + \lambda_0^i \boldsymbol{\rho}_0^i + \sum_{j=1}^{J} \lambda_j^i \sum_{\mathbf{a}\in\mathcal{V}_j} \boldsymbol{\rho}_{j,\mathbf{a}}^i. \tag{4.22}$$

It is also easy to see that the local updates satisfy for $j \in \{1,\dots,J\}$

$$(\boldsymbol{\mathcal{K}}^{-1}\boldsymbol{\rho}_{j,\mathbf{a}}^i, \mathbf{v}_{j,\mathbf{a}})_{\omega_j^{\mathbf{a}}} = -(\boldsymbol{\mathcal{K}}^{-1}\mathbf{u}_h^i, \mathbf{v}_{j,\mathbf{a}})_{\omega_j^{\mathbf{a}}} - \sum_{k=0}^{j-1} \lambda_k^i (\boldsymbol{\mathcal{K}}^{-1}\boldsymbol{\rho}_k^i, \mathbf{v}_{j,\mathbf{a}})_{\omega_j^{\mathbf{a}}} \quad \forall\, \mathbf{v}_{j,\mathbf{a}} \in \mathbf{V}_j^{\mathbf{a},0}. \tag{4.23}$$

Now, we explain through the following lemma the benefits of using optimal level-wise step-sizes, as also seen and used in, e.g., Heinrichs [1988]: this leads to the best possible decrease of the algebraic error along the direction given by $\boldsymbol{\rho}_j^i$.

Lemma 4.4 (Optimal step-sizes)**.** *For* $j \in \{1,\dots,J\}$*, the step size* λ_j^i *defined in* (4.20) *satisfies*

$$\lambda_j^i = \operatorname*{argmin}_{\lambda} \left\|\boldsymbol{\mathcal{K}}^{-1/2}(\mathbf{u}_h - (\mathbf{u}_{j-1}^i + \lambda \boldsymbol{\rho}_j^i))\right\|. \tag{4.24}$$

Proof. The result follows from determining the minimum of the quadratic function

$$\left\|\boldsymbol{\mathcal{K}}^{-1/2}(\mathbf{u}_h - (\mathbf{u}_{j-1}^i + \lambda\boldsymbol{\rho}_j^i))\right\|^2 = \left\|\boldsymbol{\mathcal{K}}^{-1/2}(\mathbf{u}_h - \mathbf{u}_{j-1}^i)\right\|^2 - 2\lambda(\boldsymbol{\mathcal{K}}^{-1}(\mathbf{u}_h - \mathbf{u}_{j-1}^i), \boldsymbol{\rho}_j^i) + \lambda^2\left\|\boldsymbol{\mathcal{K}}^{-1/2}\boldsymbol{\rho}_j^i\right\|^2,$$

which is

$$\lambda_{\min} = \frac{(\boldsymbol{\mathcal{K}}^{-1}(\mathbf{u}_h - \mathbf{u}_{j-1}^i), \boldsymbol{\rho}_j^i)}{\left\|\boldsymbol{\mathcal{K}}^{-1/2}\boldsymbol{\rho}_j^i\right\|^2} \overset{(4.7)}{=} -\frac{(\boldsymbol{\mathcal{K}}^{-1}\mathbf{u}_{j-1}^i, \boldsymbol{\rho}_j^i)}{\left\|\boldsymbol{\mathcal{K}}^{-1/2}\boldsymbol{\rho}_j^i\right\|^2}.$$

□

Lemma 4.5 (Connection of the local contributions with the level-wise updates)**.** *Note that for* $\boldsymbol{\rho}_j^i$ *given by* (4.18)–(4.19)*,* $j \in \{1,\dots,J\}$*, we have*

$$\sum_{\mathbf{a}\in\mathcal{V}_j} \left\|\boldsymbol{\mathcal{K}}^{-1/2}\boldsymbol{\rho}_{j,\mathbf{a}}^i\right\|_{\omega_j^{\mathbf{a}}}^2 \overset{\substack{(4.18)\\(4.19)}}{=} -(\boldsymbol{\mathcal{K}}^{-1}\mathbf{u}_{j-1}^i, \boldsymbol{\rho}_j^i) \overset{(4.20)}{=} \lambda_j^i \left\|\boldsymbol{\mathcal{K}}^{-1/2}\boldsymbol{\rho}_j^i\right\|^2. \tag{4.25}$$

The next result on the error reduction immediately follows from [Miraçi et al., 2021a, Theorem 4.6], see Theorem 4.7 in Chapter 2, and the proof is thus omitted here.

Theorem 4.6 (Error representation of one solver step)**.** *For Algorithm 4.2, it holds that*

$$\left\|\boldsymbol{\mathcal{K}}^{-1/2}(\mathbf{u}_h - \mathbf{u}_h^{i+1})\right\|^2 = \left\|\boldsymbol{\mathcal{K}}^{-1/2}(\mathbf{u}_h - \mathbf{u}_h^{i})\right\|^2 - \sum_{j=0}^{J}(\lambda_j^i \|\boldsymbol{\mathcal{K}}^{-1/2}\boldsymbol{\rho}_j^i\|)^2 \tag{4.26a}$$

$$= \left\|\boldsymbol{\mathcal{K}}^{-1/2}(\mathbf{u}_h - \mathbf{u}_h^{i})\right\|^2 - (\lambda_0^i \|\boldsymbol{\mathcal{K}}^{-1/2}\boldsymbol{\rho}_0^i\|)^2 - \sum_{j=1}^{J}\lambda_j^i \sum_{\mathbf{a}\in\mathcal{V}_j} \|\boldsymbol{\mathcal{K}}^{-1/2}\boldsymbol{\rho}_{j,\mathbf{a}}^i\|_{\omega_j^{\mathbf{a}}}^2. \tag{4.26b}$$

4.2 A posteriori estimator on the algebraic error

The same procedure used in Algorithm 4.2 to define the update from one iteration to the next can also be used to define an a posteriori estimator of the algebraic error.

Definition 4.7 (Multilevel a posteriori estimator)**.** *Let $\mathbf{u}_h^i \in \mathbf{V}_h$ be arbitrary. Let λ_j^i, $\boldsymbol{\rho}_j^i$, $0 \le j \le J$, be given by one step of the solver of Algorithm 4.2. Define the a posteriori estimator of the algebraic error as*

$$\eta_{\text{alg}}^i := \left(\sum_{j=0}^{J}(\lambda_j^i \|\boldsymbol{\mathcal{K}}^{-1/2}\boldsymbol{\rho}_j^i\|)^2\right)^{1/2}. \tag{4.27}$$

Note that thanks to the construction used in the a posteriori estimator and solver, the latter can be interpreted as being *a-posteriori-steered*. It is also important to note that the estimator of Definiton 4.7 provides a guaranteed lower bound on the algebraic error. Indeed, it follows immediately from (4.26a):

Lemma 4.8 (Guaranteed lower bound on the algebraic error)**.** *There holds:*

$$\left\|\boldsymbol{\mathcal{K}}^{-1/2}(\mathbf{u}_h - \mathbf{u}_h^{i})\right\| \ge \eta_{\text{alg}}^i. \tag{4.28}$$

Moreover, thanks to (4.25), the estimator is localized patch-wise as well as level-wise.

Corollary 4.9 (Patch-wise and level-wise localized writing of the estimator)**.** *There holds:*

$$\left(\eta_{\text{alg}}^i\right)^2 = (\lambda_0^i \|\boldsymbol{\mathcal{K}}^{-1/2}\boldsymbol{\rho}_0^i\|)^2 + \sum_{j=1}^{J}\lambda_j^i \sum_{\mathbf{a}\in\mathcal{V}_j} \|\boldsymbol{\mathcal{K}}^{-1/2}\boldsymbol{\rho}_{j,\mathbf{a}}^i\|_{\omega_j^{\mathbf{a}}}^2. \tag{4.29}$$

5 An a-posteriori-steered domain decomposition solver

In this section we present how to adapt the multigrid methodology developed in Section 4 to a domain decomposition one.

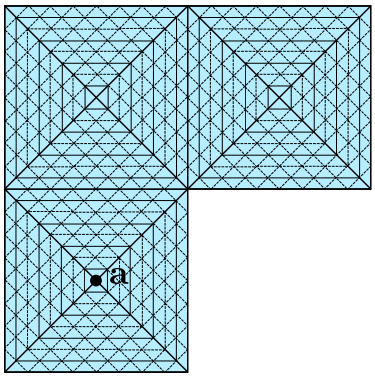


Figure 4.2: Illustration of a patch in the two level overlapping additive Schwarz method: coarse grid $\mathcal{T}_H$ (solid line), fine grid $\mathcal{T}_h$ (dashed line), the patch associated with the vertex $\mathbf{a} \in \mathcal{V}_H$ contains four coarse elements of $\mathcal{T}_H$ that share $\mathbf{a}$ and form a subdomain. The coarse (subdomains) patches are discretized with the fine grid $\mathcal{T}_h$.

5.1 Two-level hierarchy

We consider a hierarchy of *two* nested matching meshes of Ω, denoted by $\mathcal{T}_H$ and $\mathcal{T}_h$, where the mesh $\mathcal{T}_h$ is obtained from $\mathcal{T}_H$ by a *sequence* of refinements. More precisely, we assume that there is a multilevel mesh hierarchy $\{\mathcal{T}_j\}_{0\leq j\leq J}$, as given in Section 3.1, where $J \geq 1$ and $\mathcal{T}_0 = \mathcal{T}_H$, $\mathcal{T}_J = \mathcal{T}_h$. Only the levels 0 and J will be used in the algorithm for the domain decomposition, whereas all the levels $0 \leq j \leq J$ will be used in the forthcoming analysis. The associated mixed finite element spaces are denoted by $\mathbf{V}_H \times W_H \subset \mathbf{V}_h \times W_h$, and are given as in Section 3.2 by $\mathbf{V}_H = \mathbf{V}_0$, $W_H = W_0$, $\mathbf{V}_h = \mathbf{V}_J$, and $W_h = W_J$.

5.2 Two-level iterative solver: overlapping additive Schwarz

We denote by $\mathcal{V}_H$ the set of vertices of the coarse mesh $\mathcal{T}_H$ and by $\mathcal{V}_K$ the set of vertices of an element K of $\mathcal{T}_H$. For each coarse vertex $\mathbf{a} \in \mathcal{V}_H$, the associated patch of coarse elements sharing $\mathbf{a}$ is $\mathcal{T}_H^{\mathbf{a}} := \{K \in \mathcal{T}_H, \mathbf{a} \in \mathcal{V}_K\}$. Next, we denote the open patch subdomain corresponding to $\mathcal{T}_H^{\mathbf{a}}$ by $\omega_H^{\mathbf{a}}$; they will be used as *subdomains* for the overlapping domain decomposition method, Figure 4.2 gives an illustration. Define the local MFE space as

$$\mathbf{V}_h^{\mathbf{a}} := \{\mathbf{v}_h \in \mathbf{V}_h|_{\omega_H^{\mathbf{a}}}\ \mathbf{v}_h \cdot \mathbf{n} = 0 \text{ on } \partial\omega_H^{\mathbf{a}}\}. \tag{4.30}$$

Remark that these spaces are restrictions of the fine-mesh MFE space $\mathbf{V}_h$ on the subdomains $\omega_H^{\mathbf{a}}$ with homogeneous Neumann boundary conditions on the whole boundary of the subdomain. Finally, define

$$\mathbf{V}_h^{\mathbf{a},0} = \{\mathbf{v}_{h,\mathbf{a}} \in \mathbf{V}_h^{\mathbf{a}},\ \nabla \cdot \mathbf{v}_{h,\mathbf{a}} = 0\}. \tag{4.31}$$

The iterative solver is similar to the multilevel Algorithm 4.2 in the case of two levels only ($J = 1$). It reads:

Algorithm 5.1 (Additive Schwarz domain decomposition solver)**.**

1. *Initialize* $\mathbf{u}_h^0 \in \mathbf{V}_h^f$ *as in Algorithm 4.2 and let* $i := 0$.

2. *Perform the following steps (a)–(d):*

   (a) *Solve the coarse grid problem (4.17) to compute the global residual lifting* $\boldsymbol{\rho}_H^i \in \mathbf{V}_H^0$. *Set* $\lambda_H^i := 1$ *and* $\mathbf{u}_H^i := \mathbf{u}_h^i + \lambda_H^i \boldsymbol{\rho}_H^i$.

   (b) *Compute the local updates* $\boldsymbol{\rho}_{h,\mathbf{a}}^i \in \mathbf{V}_h^{\mathbf{a},0}$ *as solutions of the patch problems, for all* $\mathbf{a} \in \mathcal{V}_H$,

$$(\boldsymbol{\mathcal{K}}^{-1}\boldsymbol{\rho}_{h,\mathbf{a}}^i, \mathbf{v}_{h,\mathbf{a}})_{\omega_H^{\mathbf{a}}} = -(\boldsymbol{\mathcal{K}}^{-1}\mathbf{u}_H^i, \mathbf{v}_{h,\mathbf{a}})_{\omega_H^{\mathbf{a}}} \quad \forall \mathbf{v}_{h,\mathbf{a}} \in \mathbf{V}_h^{\mathbf{a},0}. \tag{4.32}$$

   Define $\boldsymbol{\rho}_h^i \in \mathbf{V}_h^0$ *by*

$$\boldsymbol{\rho}_h^i := \sum_{\mathbf{a}\in\mathcal{V}_H} \boldsymbol{\rho}_{h,\mathbf{a}}^i. \tag{4.33}$$

   If $\boldsymbol{\rho}_h^i \neq 0$, *define the level step size by*

$$\lambda_h^i := -\frac{(\boldsymbol{\mathcal{K}}^{-1}\mathbf{u}_H^i, \boldsymbol{\rho}_h^i)}{\left\|\boldsymbol{\mathcal{K}}^{-1/2}\boldsymbol{\rho}_h^i\right\|^2}. \tag{4.34}$$

   and otherwise set $\lambda_h^i := 1$.

   (c) *Set the next iterate*

$$\mathbf{u}_h^{i+1} := \mathbf{u}_H^i + \lambda_h^i \boldsymbol{\rho}_h^i. \tag{4.35}$$

   (d) *If* $\mathbf{u}_h^{i+1} = \mathbf{u}_h^i$, *then stop the solver. Otherwise set* $i := i+1$ *and go to step 2(a).*

Similarly to (4.25) and Theorem 4.6, we also have here

Lemma 5.2 (Connection of the local contributions with the level-wise updates)**.** *For* $\boldsymbol{\rho}_h^i$ *given by* (4.32)–(4.33), *we have the following link between the local and global contributions*

$$\sum_{\mathbf{a}\in\mathcal{V}_H} \left\|\boldsymbol{\mathcal{K}}^{-1/2}\boldsymbol{\rho}_{h,\mathbf{a}}^i\right\|_{\omega_H^{\mathbf{a}}}^2 \overset{\substack{(4.32)\\(4.33)}}{=} -(\boldsymbol{\mathcal{K}}^{-1}\mathbf{u}_H^i, \boldsymbol{\rho}_h^i) \overset{(4.34)}{=} \lambda_h^i \left\|\boldsymbol{\mathcal{K}}^{-1/2}\boldsymbol{\rho}_h^i\right\|^2. \tag{4.36}$$

Moreover, the error decrease in the two-level setting is given by

Corollary 5.3 (Error representation of one solver step)**.** *For Algorithm 5.1, it holds that*

$$\begin{aligned}
&\left\|\boldsymbol{\mathcal{K}}^{-1/2}(\mathbf{u}_h - \mathbf{u}_h^{i+1})\right\|^2 \\
&= \left\|\boldsymbol{\mathcal{K}}^{-1/2}(\mathbf{u}_h - \mathbf{u}_h^{i})\right\|^2 - (\lambda_H^i \left\|\boldsymbol{\mathcal{K}}^{-1/2}\boldsymbol{\rho}_H^i\right\|)^2 - (\lambda_h^i \left\|\boldsymbol{\mathcal{K}}^{-1/2}\boldsymbol{\rho}_h^i\right\|)^2 && (4.37)\\
&\overset{(4.36)}{=} \left\|\boldsymbol{\mathcal{K}}^{-1/2}(\mathbf{u}_h - \mathbf{u}_h^{i})\right\|^2 - (\lambda_H^i \left\|\boldsymbol{\mathcal{K}}^{-1/2}\boldsymbol{\rho}_H^i\right\|)^2 - \lambda_h^i \sum_{\mathbf{a}\in\mathcal{V}_H} \left\|\boldsymbol{\mathcal{K}}^{-1/2}\boldsymbol{\rho}_{h,\mathbf{a}}^i\right\|_{\omega_H^{\mathbf{a}}}^2. && (4.38)
\end{aligned}$$

5.3 A posteriori estimator on the algebraic error

Similarly to the multilevel case, we can define also here a two-level a posteriori estimator of the algebraic error.

Definition 5.4 (Two-level a posteriori estimator)**.** *Let $\mathbf{u}_h^i \in \mathbf{V}_h$ be arbitrary. Let λ_H^i, $\boldsymbol{\rho}_H^i$, λ_h^i, $\boldsymbol{\rho}_h^i$ be given by one step of the solver of Algorithm 5.1. Define the a posteriori estimator of the algebraic error as*

$$\eta_{\mathrm{alg}}^i = \Big((\lambda_H^i \|\boldsymbol{\mathcal{K}}^{-1/2} \boldsymbol{\rho}_H^i\|)^2 + (\lambda_h^i \|\boldsymbol{\mathcal{K}}^{-1/2} \boldsymbol{\rho}_h^i\|)^2 \Big)^{1/2}. \tag{4.39}$$

This estimator is a guaranteed lower bound on the algebraic error as well. Similarly to the multilevel case, it follows from the error decrease formula (4.37):

Lemma 5.5 (Guaranteed lower bound on the algebraic error)**.** *There holds:*

$$\|\boldsymbol{\mathcal{K}}^{-1/2}(\mathbf{u}_h - \mathbf{u}_h^i)\| \geq \eta_{\mathrm{alg}}^i. \tag{4.40}$$

From (4.36), the estimator can be written in a localized form

Corollary 5.6 (Patch-wise and level-wise localized writing of the estimator)**.** *There holds:*

$$\left(\eta_{\mathrm{alg}}^i\right)^2 = \left(\lambda_H^i \|\boldsymbol{\mathcal{K}}^{-1/2} \boldsymbol{\rho}_H^i\|\right)^2 + \lambda_h^i \sum_{\mathbf{a} \in \mathcal{V}_H} \|\boldsymbol{\mathcal{K}}^{-1/2} \boldsymbol{\rho}_{h,\mathbf{a}}^i\|_{\omega_H^{\mathbf{a}}}^2. \tag{4.41}$$

6 Main results

We now present here our main results of k-robust error contraction of the multigrid solver of Algorithm 4.2 and the domain decomposition method of Algorithm 5.1, and k-robust efficiency of their associated a posteriori estimators. One crucial ingredient to obtain such results is the following proposition.

For the a posteriori estimators we introduced, there holds:

Theorem 6.1 (k-robust efficiency of η_{alg}^i)**.** *Let d=2 and Assumption 3.1 hold as well as either Assumption 3.2 or Assumption 3.3. Let $\mathbf{u}_h^i \in \mathbf{V}_h$ be arbitrary. Let η_{alg}^i be constructed from $\mathbf{u}_h^i$ by either Definition 4.7 or Definition 5.4. Then, in addition to $\|\boldsymbol{\mathcal{K}}^{-1/2}(\mathbf{u}_h - \mathbf{u}_h^i)\| \geq \eta_{\mathrm{alg}}^i$, there holds*

$$\eta_{\mathrm{alg}}^i \geq \beta \|\boldsymbol{\mathcal{K}}^{-1/2}(\mathbf{u}_h - \mathbf{u}_h^i)\|, \tag{4.42}$$

where $0 < \beta < 1$ depends on the mesh shape regularity parameter $\kappa_{\mathcal{T}}$, the ratio of the largest and the smallest eigenvalue of the diffusion coefficient $\boldsymbol{\mathcal{K}}$, at most linearly on the number of mesh levels J, and additionally on the parameters C_{ref} and C_{qu} when Assumption 3.2 is satisfied, or on the parameters C_{qu}^0 and $C_{\mathrm{loc,qu}}$ when Assumption 3.3 is satisfied. In particular, C_{SD} is independent of the polynomial degree k.

For the algebraic solvers we presented, there holds:

Theorem 6.2 (k-robust error contraction)**.** *Let d=2 and Assumption 3.1 hold as well as either Assumption 3.2 or Assumption 3.3. Let $\mathbf{u}_h^i \in \mathbf{V}_h$ be arbitrary. Let $\mathbf{u}_h^{i+1} \in \mathbf{V}_h$ be constructed from $\mathbf{u}_h^i$ by one step of either Algorithm 4.2 or Algorithm 5.1. Then, there holds*

$$\|\boldsymbol{\mathcal{K}}^{-1/2}(\mathbf{u}_h - \mathbf{u}_h^{i+1})\| \leq \alpha \|\boldsymbol{\mathcal{K}}^{-1/2}(\mathbf{u}_h - \mathbf{u}_h^i)\|, \tag{4.43}$$

where $0 < \alpha < 1$ is given by $\alpha = \sqrt{1 - \beta^2}$ with β the constant from (4.42).

Proof. The proof follows from the equivalence of (4.42) and (4.43), similarly to [Miraçi et al., 2021a, Corollary 6.7], see Corollary 6.7 in Chapter 2. We present it here for completeness, starting from (4.43) with $0 < \alpha < 1$:

$$\begin{aligned}
&\left\|\mathcal{K}^{-1/2}(\mathbf{u}_h - \mathbf{u}_h^{i+1})\right\|^2 \leq \alpha^2 \left\|\mathcal{K}^{-1/2}(\mathbf{u}_h - \mathbf{u}_h^{i})\right\|^2 \\
&\overset{(4.26a)}{\Leftrightarrow} \left\|\mathcal{K}^{-1/2}(\mathbf{u}_h - \mathbf{u}_h^{i})\right\|^2 - \sum_{j=0}^{J} (\lambda_j^i \|\mathcal{K}^{-1/2}\boldsymbol{\rho}_j^i\|)^2 \leq \alpha^2 \left\|\mathcal{K}^{-1/2}(\mathbf{u}_h - \mathbf{u}_h^{i})\right\|^2 \\
&\overset{(4.27)}{\Leftrightarrow} \left\|\mathcal{K}^{-1/2}(\mathbf{u}_h - \mathbf{u}_h^{i})\right\|^2 - (\eta_{\text{alg}}^i)^2 \leq \alpha^2 \left\|\mathcal{K}^{-1/2}(\mathbf{u}_h - \mathbf{u}_h^{i})\right\|^2 \\
&\Leftrightarrow \quad (1-\alpha^2)\left\|\mathcal{K}^{-1/2}(\mathbf{u}_h - \mathbf{u}_h^{i})\right\|^2 \leq (\eta_{\text{alg}}^i)^2.
\end{aligned}$$

□

Remark 6.3 (Extension to three space dimensions)**.** *We would like to emphasize that the above results can also be extended to three space dimensions. The proofs in the general case were generalized to three dimensions following the approach in* [Cai et al., 2003, Lemma 5.1]*. However, we do not have a proof as of yet of the estimates being k-robust, which represents the main focus of this work.*

7 Proofs of the main results

Since Theorem 6.2 is easily proved from Theorem 6.1, it remains to prove the latter. We begin by introducing below some tools that will be useful for the proofs.

Remark 7.1 (Multilevel orthogonal decomposition of the algebraic error)**.** *In order to apply the results of previous work, we introduce here a multilevel orthogonal decomposition of the algebraic error, similarly to* Miraçi et al. [2021a]*, see Chapter 2. Define* $\tilde{\boldsymbol{\rho}}^i \in \mathbf{V}_h^0$*:*

$$\tilde{\boldsymbol{\rho}}^i := \boldsymbol{\rho}_0^i + \sum_{j=1}^{J} \tilde{\boldsymbol{\rho}}_j^i, \tag{4.44}$$

where $\boldsymbol{\rho}_0^i \in \mathbf{V}_0^0$ *solves the coarse grid problem* (4.17) *and, for* $j \in \{1, \dots, J\}$*,* $\tilde{\boldsymbol{\rho}}_j^i \in \mathbf{V}_j^0$ *solves*

$$(\mathcal{K}^{-1}\tilde{\boldsymbol{\rho}}_j^i, \mathbf{v}_j) = -(\mathcal{K}^{-1}\mathbf{u}_h^i, \mathbf{v}_j) - \sum_{k=0}^{j-1} (\mathcal{K}^{-1}\tilde{\boldsymbol{\rho}}_k^i, \mathbf{v}_j), \quad \forall\, \mathbf{v}_j \in \mathbf{V}_j^0, \tag{4.45}$$

using the convention $\tilde{\boldsymbol{\rho}}_0^i = \boldsymbol{\rho}_0^i$*. From* (4.44)–(4.45)*, one can derive that* $\tilde{\boldsymbol{\rho}}^i = \mathbf{u}_h - \mathbf{u}_h^i$ *and the following orthogonal decomposition holds*

$$\left\|\mathcal{K}^{-1/2}(\mathbf{u}_h - \mathbf{u}_h^i)\right\|^2 = \left\|\mathcal{K}^{-1/2}\tilde{\boldsymbol{\rho}}^i\right\|^2 = \sum_{j=0}^{J} \left\|\mathcal{K}^{-1/2}\tilde{\boldsymbol{\rho}}_j^i\right\|^2. \tag{4.46}$$

Remark 7.2 (Lower bound on the optimal step-sizes)**.** *Similarly to* [Miraçi et al., 2021a, Lemma 9.1]*, see Lemma 10.1 in Chapter 2, we can use* (4.19) *to obtain*

$$\left\|\mathcal{K}^{-1/2}\boldsymbol{\rho}_j^i\right\|^2 \leq (d+1) \sum_{\mathbf{a} \in \mathcal{V}_j} \left\|\mathcal{K}^{-1/2}\boldsymbol{\rho}_{j,\mathbf{a}}^i\right\|_{\omega_j^{\mathbf{a}}}^2. \tag{4.47}$$

A direct consequence of (4.47) *and* (4.25)*, together with the definition of* $\lambda_j^i = 1$ *when* $j = 0$ *or* $\boldsymbol{\rho}_j^i = 0$*, is*

$$\lambda_j^i \geq \frac{1}{d+1} \quad 0 \leq j \leq J. \tag{4.48}$$

A crucial component in our analysis is the existance of a suitable stable decomposition of the space $\mathbf{V}_h^0$:

Proposition 7.3 (Multilevel k-robust stable decomposition)**.** *Let d=2 and Assumption 3.1 hold as well as either Assumption 3.2 or Assumption 3.3. For any* $\mathbf{v}_h \in \mathbf{V}_h^0$*, there exist* $\mathbf{v}_0 \in \mathbf{V}_0^0$ *and* $\mathbf{v}_{j,\mathbf{a}} \in \mathbf{V}_j^{\mathbf{a},0}$*,* $1 \leq j \leq J$*,* $\mathbf{a} \in \mathcal{V}_j$*, such that*

$$\mathbf{v}_h = \mathbf{v}_0 + \sum_{j=1}^{J} \sum_{\mathbf{a}\in\mathcal{V}_j} \mathbf{v}_{j,\mathbf{a}} \tag{4.49a}$$

stable as

$$\left\|\boldsymbol{\mathcal{K}}^{-1/2}\mathbf{v}_0\right\|^2 + \sum_{j=1}^{J} \sum_{\mathbf{a}\in\mathcal{V}_j} \left\|\boldsymbol{\mathcal{K}}^{-1/2}\mathbf{v}_{j,\mathbf{a}}\right\|_{\omega_j^{\mathbf{a}}}^2 \leq C_{\mathrm{SD}}^2 \left\|\boldsymbol{\mathcal{K}}^{-1/2}\mathbf{v}_h\right\|^2, \tag{4.49b}$$

where the constant C_{SD} *depends on the mesh shape regularity parameter* $\kappa_{\mathcal{T}}$*, the ratio of the largest and the smallest eigenvalue of the diffusion coefficient* $\boldsymbol{\mathcal{K}}$*, and additionally on the parameters* C_{ref} *and* C_{qu} *when Assumption 3.2 is satisfied, or on the parameters* C_{qu}^0 *and* $C_{\mathrm{loc,qu}}$ *when Assumption 3.3 is satisfied. In particular,* C_{SD} *is independent of the polynomial degree* k*.*

Proof of Proposition 7.3. The construction of the stable decomposition uses the property that, in two space dimensions, there exists a discrete stream function space S_h such that

$$\mathbf{V}_h^0 = \operatorname{curl} S_h. \tag{4.50}$$

In the case of the spaces $\mathbf{RT}_k$, $k \geq 0$, on triangles, the space S_h consists of continuous piecewise polynomials in P_{k+1}, see e.g. [Boffi et al., 2013, Corollary 2.3.2]: since

$$\mathbf{V}_h^0 = \{\mathbf{v}_h \in \mathbf{V},\ \mathbf{v}_h|_K \in \mathbf{RT}_k(K)\ \forall K \in \mathcal{T}_h,\ \nabla\cdot\mathbf{v}_h = 0\},$$

the associated discrete stream function space such that (4.50) holds is

$$S_h = \mathbb{P}_{k+1}(\mathcal{T}_h) \cap H_0^1(\Omega). \tag{4.51}$$

Actually, the same holds for true also for the **BDM** spaces. Here, see e.g. [Boffi et al., 2013, Remark 2.1.5], we use the notation

$$\operatorname{curl} s = \begin{pmatrix} s_y \\ -s_x \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \begin{pmatrix} s_x \\ s_y \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \nabla s. \tag{4.52}$$

Since for all $\mathbf{v}_h, \boldsymbol{\xi}_h \in \mathbf{V}_h^0$, by (4.50), there is $s_h, \theta_h \in S_h$ such that $\mathbf{v}_h = \operatorname{curl} s_h$ and $\boldsymbol{\xi}_h = \operatorname{curl}\theta_h$, we obtain from (4.52) that

$$(\boldsymbol{\mathcal{K}}^{-1}\mathbf{v}_h, \boldsymbol{\xi}_h) = (\boldsymbol{\mathcal{K}}^{-1}\operatorname{curl} s_h, \operatorname{curl}\theta_h) = (\boldsymbol{\mathcal{A}}\nabla s_h, \nabla\theta_h) \tag{4.53}$$

for $\boldsymbol{\mathcal{A}} := \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} \boldsymbol{\mathcal{K}}^{-1} \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$. In particular, note that $(\boldsymbol{\mathcal{K}}^{-1}\mathbf{v}_h, \boldsymbol{\xi}_h) = (\nabla s_h, \nabla \theta_h)$ when $\boldsymbol{\mathcal{K}} = I$. Similar properties obviously hold on all patches as well.

We state now for the reader's convenience the multilevel k-robust stable decomposition for any function of the stream-function space $S_h = \mathbb{P}_{k+1}(\mathcal{T}_h) \cap H_0^1(\Omega)$ in the $H_0^1(\Omega)$-norm $\|\nabla \cdot\|$, shown in [Miraçi et al., 2020, Proposition 7.6], see Proposition 7.6 in Chapter 1: Under either Assumption 3.2 or Assumption 3.3, for any $v_h \in S_h = \mathbb{P}_{k+1}(\mathcal{T}_h) \cap H_0^1(\Omega)$, there exists a decomposition

$$v_h = v_0 + \sum_{j=1}^{J} \sum_{\mathbf{a} \in \mathcal{V}_j} v_{j,\mathbf{a}}, \tag{4.54a}$$

with $v_0 \in S_0^1 := \mathbb{P}_1(\mathcal{T}_0) \cap H_0^1(\Omega)$ and $v_{j,\mathbf{a}} \in S_j^{\mathbf{a}} := \mathbb{P}_{k+1}(\mathcal{T}_j) \cap H_0^1(\omega_j^{\mathbf{a}})$, stable as

$$\|\nabla v_0\|^2 + \sum_{j=1}^{J} \sum_{\mathbf{a} \in \mathcal{V}_j} \|\nabla v_{j,\mathbf{a}}\|_{\omega_j^{\mathbf{a}}}^2 \leq C_{\mathrm{SD}}^2 \|\nabla v_h\|^2, \tag{4.54b}$$

with C_{SD} depending on the mesh shape regularity parameter $\kappa_{\mathcal{T}}$ and additionally on the parameters C_{ref} and C_{qu} when Assumption 3.2 is satisfied, or on the parameters C_{qu}^0 and $C_{\mathrm{loc,qu}}$ when Assumption 3.3 is satisfied. In particular, C_{SD} is independent of the polynomial degree k. Recall that this result was obtained by combining the results of Schöberl et al. [2008] and Xu et al. [2009].

Let us now show (4.49). Let $\mathbf{v}_h \in \mathbf{V}_h^0$. By (4.50), there is $v_h \in S_h$ such that

$$\mathbf{v}_h = \operatorname{curl} v_h. \tag{4.55}$$

We can now use the stable decomposition (4.54a) for v_h: there exist $v_0 \in S_0^1$ and $v_{j,\mathbf{a}} \in S_j^{\mathbf{a}}$, $1 \leq j \leq J$, $\mathbf{a} \in \mathcal{V}_j$, such that $v_h = v_0 + \sum_{j=1}^{J} \sum_{\mathbf{a} \in \mathcal{V}_j} v_{j,\mathbf{a}}$ and (4.54a) holds. Define

$$\mathbf{v}_0 := \operatorname{curl} v_0 \quad \text{and} \quad \mathbf{v}_{j,\mathbf{a}} := \operatorname{curl} v_{j,\mathbf{a}}. \tag{4.56}$$

Since $\mathbf{V}_0^0 = \operatorname{curl} S_0^1$ and $\mathbf{V}_j^{\mathbf{a},0} = \operatorname{curl} S_j^{\mathbf{a}}$, $1 \leq j \leq J$, $\mathbf{a} \in \mathcal{V}_j$, we have $\mathbf{v}_0 \in \mathbf{V}_0^0$ and $\mathbf{v}_{j,\mathbf{a}} \in \mathbf{V}_j^{\mathbf{a},0}$. Note that by taking the curl on both sides of (4.54a), we have

$$\mathbf{v}_h = \mathbf{v}_0 + \sum_{j=1}^{J} \sum_{\mathbf{a} \in \mathcal{V}_j} \mathbf{v}_{j,\mathbf{a}}, \quad \mathbf{v}_0 \in \mathbf{V}_0^0, \ \mathbf{v}_{j,\mathbf{a}} \in \mathbf{V}_j^{\mathbf{a},0},$$

which is the first part (4.49a) of the result we wanted to prove. Next, note that

$$\begin{aligned} \|\mathbf{v}_0\|^2 + \sum_{j=1}^{J} \sum_{\mathbf{a} \in \mathcal{V}_j} \|\mathbf{v}_{j,\mathbf{a}}\|_{\omega_j^{\mathbf{a}}}^2 &\overset{(4.56)}{=} \|\operatorname{curl} v_0\|^2 + \sum_{j=1}^{J} \sum_{\mathbf{a} \in \mathcal{V}_j} \|\operatorname{curl} v_{j,\mathbf{a}}\|_{\omega_j^{\mathbf{a}}}^2 \\ &\overset{(4.52)}{=} \|\nabla v_0\|^2 + \sum_{j=1}^{J} \sum_{\mathbf{a} \in \mathcal{V}_j} \|\nabla v_{j,\mathbf{a}}\|_{\omega_j^{\mathbf{a}}}^2 \\ &\overset{(4.54b)}{\leq} C_{\mathrm{SD}}^2 \|\nabla v_h\|^2 \overset{(4.52)}{=} C_{\mathrm{SD}}^2 \|\operatorname{curl} v_h\|^2 \overset{(4.55)}{=} C_{\mathrm{SD}}^2 \|\mathbf{v}_h\|^2. \end{aligned}$$

Finally, the result (4.49b) follows once we take into account the variations of the diffusion coefficient $\boldsymbol{\mathcal{K}}$.

□

7.1 Multilevel setting results

We can now present the proof of Theorem 6.1 for the estimator given by Defintion 4.7.

Proof of Theorem 6.1 (multilevel setting). The proof follows closely the proof of [Miraçi et al., 2021a, Theorem 6.6], see Theorem 6.6 in Chapter 2, thus only the main steps are presented here. From (4.23) and (4.45) we get, for $j \in \{1, \dots, J\}$ and $\mathbf{a} \in \mathcal{V}_j$,

$$(\boldsymbol{\mathcal{K}}^{-1}\tilde{\boldsymbol{\rho}}^i_j, \mathbf{v}_{j,\mathbf{a}})_{\omega^{\mathbf{a}}_j} = (\boldsymbol{\mathcal{K}}^{-1}\boldsymbol{\rho}^i_{j,\mathbf{a}}, \mathbf{v}_{j,\mathbf{a}})_{\omega^{\mathbf{a}}_j} - \sum_{k=0}^{j-1}(\boldsymbol{\mathcal{K}}^{-1}(\tilde{\boldsymbol{\rho}}^i_k - \lambda^i_k \boldsymbol{\rho}^i_{k,\mathbf{a}})\mathbf{v}_{j,\mathbf{a}})_{\omega^{\mathbf{a}}_j}, \quad \forall \mathbf{v}_{j,\mathbf{a}} \in \mathbf{V}^{\mathbf{a},0}_j. \tag{4.57}$$

Using the stable decomposition (4.49a)–(4.49b) applied to $\tilde{\boldsymbol{\rho}}^i$ and (4.57), one can show, in the same spirit of [Miraçi et al., 2021a, Proof of Theorem 6.6], see Theorem 6.6 in Chapter 2, that there holds

$$\|\boldsymbol{\mathcal{K}}^{-1/2}\tilde{\boldsymbol{\rho}}^i\|^2 \le C^2\Big(\|\boldsymbol{\mathcal{K}}^{-1/2}\boldsymbol{\rho}^i_0\|^2 + \sum_{j=1}^{J}\sum_{\mathbf{a}\in\mathcal{V}_j}\|\boldsymbol{\mathcal{K}}^{-1/2}\boldsymbol{\rho}^i_{j,\mathbf{a}}\|^2_{\omega^{\mathbf{a}}_j}\Big), \tag{4.58}$$

where C^2 depends on C_{SD} of (4.49b) and at most linearly on the number of mesh levels J. Finally, we have the desired result

$$\|\boldsymbol{\mathcal{K}}^{-1/2}(\mathbf{u}_h - \mathbf{u}^i_h)\|^2 \overset{(4.46)}{=} \|\boldsymbol{\mathcal{K}}^{-1/2}\tilde{\boldsymbol{\rho}}^i\|^2 \overset{(4.58)}{\le} C^2\Big(\|\boldsymbol{\mathcal{K}}^{-1/2}\boldsymbol{\rho}^i_0\|^2 + \sum_{j=1}^{J}\sum_{\mathbf{a}\in\mathcal{V}_j}\|\boldsymbol{\mathcal{K}}^{-1/2}\boldsymbol{\rho}^i_{j,\mathbf{a}}\|^2_{\omega^{\mathbf{a}}_j}\Big)$$

$$\overset{(4.25)}{\underset{(4.48)}{\le}} C^2(d+1)\Big((\lambda^i_0\|\boldsymbol{\mathcal{K}}^{-1/2}\boldsymbol{\rho}^i_0\|)^2 + \sum_{j=1}^{J}(\lambda^i_j\|\boldsymbol{\mathcal{K}}^{-1/2}\boldsymbol{\rho}^i_j\|)^2\Big) \overset{(4.27)}{=} C^2(d+1)(\eta^i_{\mathrm{alg}})^2.$$

□

7.2 Two-level domain decomposition setting results

We now proceed to the proof of Theorem 6.1 for the estimator given by Defintion 5.4. First, we present a few preparatory steps that will allow us to re-use the results we presented in the multilevel setting. In this section, we will use interchangeably the subscripts H and 0 as well as h and J, i.e. $\mathcal{T}_H = \mathcal{T}_0$, $\mathcal{T}_h = \mathcal{T}_J$, $\mathbf{V}_H = \mathbf{V}_0$, $\mathbf{V}_h = \mathbf{V}_J$.

Our goal is to write a multilevel presentation of the residual $\boldsymbol{\rho}^i_{h,\mathbf{a}} \in \mathbf{V}^{\mathbf{a},0}_h$ computed by solving the patch problem (4.32) on $\omega^{\mathbf{a}}_H$. Recall that $\omega^{\mathbf{a}}_H$ is the open patch subdomain corresponding to the coarse grid-related patch $\mathcal{T}^{\mathbf{a}}_0$. For the purpose of analysis, define the local MFE space on $\omega^{\mathbf{a}}_H$ associated with the intermediate mesh levels $\mathcal{T}_j$, $1 \le j \le J$, as

$$\mathbf{V}^{\mathbf{a}}_{j,0} := \{\mathbf{v}_j \in \mathbf{V}_j|_{\omega^{\mathbf{a}}_H},\ \mathbf{v}_j \cdot \mathbf{n} = 0 \text{ on } \partial\omega^{\mathbf{a}}_H\}. \tag{4.59}$$

In contrast to the spaces $\mathbf{V}^{\mathbf{a}}_j \times W^{\mathbf{a}}_j$ from Section 4.1, which are defined on the j-level (small) patches $\omega^{\mathbf{a}}_j$, we stress that the spaces $\mathbf{V}^{\mathbf{a}}_{j,0}$ are defined on the subdomains (large patches) $\omega^{\mathbf{a}}_H$ and their fine meshes $\mathcal{T}_j$; there are much fewer spaces $\mathbf{V}^{\mathbf{a}}_{j,0}$ than

the spaces $\mathbf{V}_j^{\mathbf{a}}$, but the spaces $\mathbf{V}_{j,0}^{\mathbf{a}}$ have much higher dimension than $\mathbf{V}_j^{\mathbf{a}}$. Finally, define

$$\mathbf{V}_{j,0}^{\mathbf{a},0} = \{\mathbf{v}_{h,\mathbf{a}} \in \mathbf{V}_{j,0}^{\mathbf{a}},\ \nabla \cdot \mathbf{v}_{h,\mathbf{a}} = 0\}. \tag{4.60}$$

We consider an orthogonal multilevel decomposition of $\boldsymbol{\rho}_{h,\mathbf{a}}^i \in \mathbf{V}_h^{\mathbf{a},0}$ on the patch $\omega_H^{\mathbf{a}}$. For $1 \leq j \leq J$, define for (analysis purposes, not constructed in practice) $\boldsymbol{\rho}_{j,\mathbf{a}}^i \in \mathbf{V}_{j,0}^{\mathbf{a},0}$ as the solution to

$$(\boldsymbol{\mathcal{K}}^{-1}\boldsymbol{\rho}_{j,\mathbf{a}}^i, \mathbf{v}_{j,\mathbf{a}})_{\omega_H^{\mathbf{a}}} = -(\boldsymbol{\mathcal{K}}^{-1}\mathbf{u}_0^i, \mathbf{v}_{j,\mathbf{a}})_{\omega_H^{\mathbf{a}}} - \sum_{k=1}^{j-1}(\boldsymbol{\mathcal{K}}^{-1}\boldsymbol{\rho}_{k,\mathbf{a}}^i, \mathbf{v}_{j,\mathbf{a}})_{\omega_H^{\mathbf{a}}} \quad \forall\, \mathbf{v}_{j,\mathbf{a}} \in \mathbf{V}_{j,0}^{\mathbf{a},0}. \tag{4.61}$$

It follows that

$$\sum_{j=1}^{J}(\boldsymbol{\mathcal{K}}^{-1}\boldsymbol{\rho}_{j,\mathbf{a}}^i, \mathbf{v}_{h,\mathbf{a}})_{\omega_H^{\mathbf{a}}} = -(\boldsymbol{\mathcal{K}}^{-1}\mathbf{u}_0^i, \mathbf{v}_{h,\mathbf{a}})_{\omega_H^{\mathbf{a}}} \quad \forall \mathbf{v}_{h,\mathbf{a}} \in \mathbf{V}_h^{\mathbf{a},0},$$

which, together with (4.32), implies that

$$\boldsymbol{\rho}_{h,\mathbf{a}}^i = \sum_{j=1}^{J} \boldsymbol{\rho}_{j,\mathbf{a}}^i. \tag{4.62}$$

It is easy to see that the above decomposition is orthogonal:

$$(\boldsymbol{\mathcal{K}}^{-1}\boldsymbol{\rho}_{j,\mathbf{a}}^i, \boldsymbol{\rho}_{k,\mathbf{a}}^i)_{\omega_H^{\mathbf{a}}} = 0, \quad 0 \leq j,k \leq J,\ j \neq k, \tag{4.63}$$

and

$$\left\|\boldsymbol{\mathcal{K}}^{-1/2}\boldsymbol{\rho}_{h,\mathbf{a}}^i\right\|_{\omega_H^{\mathbf{a}}}^2 = \sum_{j=1}^{J}\left\|\boldsymbol{\mathcal{K}}^{-1/2}\boldsymbol{\rho}_{j,\mathbf{a}}^i\right\|_{\omega_H^{\mathbf{a}}}^2. \tag{4.64}$$

The convergence analysis is a modification of the analysis presented in the previous Section 7.1. The key point is that the multilevel representation (4.62) allows us to utilize the k-robust stable decomposition from Proposition 7.3. We present how to achieve this below.

Proof of Theorem 6.1 (two-level domain decomposition setting). The proof here is more similar to [Miraçi et al., 2020, Lemma 7.6], see Lemma 7.7 in Chapter 1, since the patches we are considering are bigger than the ones used in the stable decomposition of Proposition 7.3. Thus, we first define $\mathcal{I}_{\mathbf{a}} \subset \mathcal{V}_J$, for all $\mathbf{a} \in \mathcal{V}_0$, as a set containing vertices in $\mathcal{T}_h$ of the interior of the patch $\omega_H^{\mathbf{a}}$ such that $\{\mathcal{I}_{\mathbf{a}}\}_{\mathbf{a}\in\mathcal{V}_0}$ cover $\mathcal{V}_J$ and are mutually disjoint. This allows us to write $\sum_{\mathbf{a}\in\mathcal{V}_J} = \sum_{\mathbf{a}\in\mathcal{V}_0}\sum_{\mathbf{b}\in\mathcal{I}_{\mathbf{a}}}$. Moreover, since the indices of $\mathcal{I}_{\mathbf{a}}$ are localized in the interior of the patch $\omega_H^{\mathbf{a}}$, we have $\sum_{\mathbf{b}\in\mathcal{I}_{\mathbf{a}}} v_{j,\mathbf{b}} \in \mathbf{V}_h^{\mathbf{a},0}$ when $v_{j,\mathbf{b}} \in \mathbf{V}_j^{\mathbf{b},0}$.

Next, note that from (4.45) and (4.61) we get, for $j \in \{1,\dots,J\}$ and $\mathbf{a} \in \mathcal{V}_0$,

$$(\boldsymbol{\mathcal{K}}^{-1}\tilde{\boldsymbol{\rho}}_j^i, \mathbf{v}_{j,\mathbf{a}})_{\omega_H^{\mathbf{a}}} = (\boldsymbol{\mathcal{K}}^{-1}\boldsymbol{\rho}_{j,\mathbf{a}}^i, \mathbf{v}_{j,\mathbf{a}})_{\omega_H^{\mathbf{a}}} - \sum_{k=0}^{j-1}(\boldsymbol{\mathcal{K}}^{-1}(\tilde{\boldsymbol{\rho}}_k^i - \boldsymbol{\rho}_{k,\mathbf{a}}^i), \mathbf{v}_{j,\mathbf{a}})_{\omega_H^{\mathbf{a}}} \ \forall \mathbf{v}_{j,\mathbf{a}} \in \mathbf{V}_{j,0}^{\mathbf{a},0}. \tag{4.65}$$

Using the stable decomposition (4.49a)–(4.49b) applied to $\tilde{\boldsymbol{\rho}}^i$ and (4.65), one can show by following the approach of [Miraçi et al., 2020, Lemma 7.6], see Lemma 7.7 in Chapter 1, that there holds

$$\|\boldsymbol{\mathcal{K}}^{-1/2}\tilde{\boldsymbol{\rho}}^i\|^2 \leq \widetilde{C}^2\Big(\|\boldsymbol{\mathcal{K}}^{-1/2}\boldsymbol{\rho}_0^i\|^2 + \sum_{j=1}^{J}\sum_{\mathbf{a}\in\mathcal{V}_0}\|\boldsymbol{\mathcal{K}}^{-1/2}\boldsymbol{\rho}_{j,\mathbf{a}}^i\|^2_{\omega_H^{\mathbf{a}}}\Big), \tag{4.66}$$

where $\widetilde{C}^2$ depends on C_{SD} of (4.49b) and at most linearly on the number of mesh levels J. Finally, we obtain the result

$$\begin{aligned}
\|\boldsymbol{\mathcal{K}}^{-1/2}(\mathbf{u}_h - \mathbf{u}_h^i)\|^2 &\overset{(4.46)}{=} \|\boldsymbol{\mathcal{K}}^{-1/2}\tilde{\boldsymbol{\rho}}^i\|^2 \overset{(4.66)}{\leq} \widetilde{C}^2\Big(\|\boldsymbol{\mathcal{K}}^{-1/2}\boldsymbol{\rho}_0^i\|^2 + \sum_{j=1}^{J}\sum_{\mathbf{a}\in\mathcal{V}_0}\|\boldsymbol{\mathcal{K}}^{-1/2}\boldsymbol{\rho}_{j,\mathbf{a}}^i\|^2_{\omega_H^{\mathbf{a}}}\Big)\\
&\overset{(4.64)}{=} \widetilde{C}^2\Big(\|\boldsymbol{\mathcal{K}}^{-1/2}\boldsymbol{\rho}_0^i\|^2 + \sum_{\mathbf{a}\in\mathcal{V}_0}\|\boldsymbol{\mathcal{K}}^{-1/2}\boldsymbol{\rho}_{h,\mathbf{a}}^i\|^2_{\omega_H^{\mathbf{a}}}\Big)\\
&\overset{\substack{(4.36)\\(4.48)}}{\leq} \widetilde{C}^2(d+1)\Big((\lambda_0^i\|\boldsymbol{\mathcal{K}}^{-1/2}\boldsymbol{\rho}_0^i\|)^2 + (\lambda_h^i\|\boldsymbol{\mathcal{K}}^{-1/2}\boldsymbol{\rho}_h^i\|)^2\Big) \overset{(4.39)}{=} \widetilde{C}^2(d+1)(\eta_{\mathrm{alg}}^i)^2.
\end{aligned}$$

□

8 Conclusions

In this work we presented an a-posteriori-steered multigrid solver and an a-posteriori-steered two-level domain decomposition method to solve iteratively an algebraic system originating from a mixed finite element discretization of a second-order elliptic problem. These solvers use information of associated a posteriori estimates of the algebraic error in order to decrease the error as efficiently as possible. We proved that, in two space dimensions, the a posteriori estimators are efficient independently of the polynomial degree k used in the discretization. This leads to the associated solvers contracting the algebraic error at each iteration, also independently of k. Future work would explore and focus on the extension of these theoretical results in three space dimensions.

Bibliography

M. Ainsworth. A preconditioner based on domain decomposition for h-p finite-element approximation on quasi-uniform meshes. *SIAM J. Numer. Anal.*, 33(4): 1358–1376, 1996. doi: 10.1137/S0036142993258221. URL https://doi.org/10.1137/S0036142993258221.

M. Ainsworth and J. T. Oden. *A posteriori error estimation in finite element analysis*. Pure and Applied Mathematics (New York). Wiley-Interscience [John Wiley & Sons], New York, 2000. doi: 10.1002/9781118032824. URL https://doi.org/10.1002/9781118032824.

A. Anciaux-Sedrakian, L. Grigori, Z. Jorti, J. Papež, and S. Yousef. Adaptive solution of linear systems of equations based on a posteriori error estimators. *Numerical Algorithms*, 84(1):331–364, 2020. doi: 10.1007/s11075-019-00757-z. URL https://doi.org/10.1007/s11075-019-00757-z.

P. F. Antonietti and G. Pennesi. V-cycle multigrid algorithms for discontinuous Galerkin methods on non-nested polytopic meshes. *J. Sci. Comput.*, 78(1):625–652, 2019. doi: 10.1007/s10915-018-0783-x. URL https://doi.org/10.1007/s10915-018-0783-x.

P. F. Antonietti, M. Sarti, M. Verani, and L. T. Zikatanov. A uniform additive Schwarz preconditioner for high-order discontinuous Galerkin approximations of elliptic problems. *J. Sci. Comput.*, 70(2):608–630, 2017. doi: 10.1007/s10915-016-0259-9. URL https://doi.org/10.1007/s10915-016-0259-9.

P. F. Antonietti, L. Mascotto, and M. Verani. A multigrid algorithm for the p-version of the virtual element method. *ESAIM Math. Model. Numer. Anal.*, 52 (1):337–364, 2018. doi: 10.1051/m2an/2018007. URL https://doi.org/10.1051/m2an/2018007.

M. Arioli, E. H. Georgoulis, and D. Loghin. Stopping criteria for adaptive finite element solvers. *SIAM J. Sci. Comput.*, 35(3):A1537–A1559, 2013. doi: 10.1137/120867421. URL http://dx.doi.org/10.1137/120867421.

I. Babuška and W. C. Rheinboldt. Error estimates for adaptive finite element computations. *SIAM J. Numer. Anal.*, 15(4):736–754, 1978. doi: 10.1137/0715049. URL https://doi.org/10.1137/0715049.

I. Babuška, A. Craig, J. Mandel, and J. Pitkäranta. Efficient preconditioning for the p-version finite element method in two dimensions. *SIAM J. Numer. Anal.*,

28(3):624–661, 1991. doi: 10.1137/0728034. URL https://doi.org/10.1137/0728034.

D. Bai and A. Brandt. Local mesh refinement multilevel techniques. *SIAM J. Sci. Statist. Comput.*, 8(2):109–134, 1987. doi: 10.1137/0908025. URL http://dx.doi.org/10.1137/0908025.

R. E. Bank, T. F. Dupont, and H. Yserentant. The hierarchical basis multigrid method. *Numer. Math.*, 52(4):427–458, 1988. doi: 10.1007/BF01462238. URL https://doi.org/10.1007/BF01462238.

P. Bastian, M. Blatt, and R. Scheichl. Algebraic multigrid for discontinuous Galerkin discretizations of heterogeneous elliptic problems. *Numer. Linear Algebra Appl.*, 19(2):367–388, 2012. doi: 10.1002/nla.1816. URL https://doi.org/10.1002/nla.1816.

R. Becker and S. Mao. Convergence and quasi-optimal complexity of a simple adaptive finite element method. *M2AN Math. Model. Numer. Anal.*, 43(6):1203–1219, 2009. doi: 10.1051/m2an/2009036. URL http://dx.doi.org/10.1051/m2an/2009036.

R. Becker, C. Johnson, and R. Rannacher. Adaptive error control for multigrid finite element methods. *Computing*, 55(4):271–288, 1995. doi: 10.1007/BF02238483. URL http://dx.doi.org/10.1007/BF02238483.

M. Benzi, G. H. Golub, and J. Liesen. Numerical solution of saddle point problems. *Acta Numer.*, 14:1–137, 2005. doi: 10.1017/S0962492904000212. URL https://doi.org/10.1017/S0962492904000212.

C. Bernardi and Y. Maday. Spectral methods. In *Handbook of numerical analysis, Vol. V*, Handb. Numer. Anal., V, pages 209–485. North-Holland, Amsterdam, 1997. doi: 10.1016/S1570-8659(97)80003-8. URL https://doi.org/10.1016/S1570-8659(97)80003-8.

A. Bespalov, A. Haberl, and D. Praetorius. Adaptive FEM with coarse initial mesh guarantees optimal convergence rates for compactly perturbed elliptic problems. *Comput. Methods Appl. Mech. Engrg.*, 317:318–340, 2017. doi: 10.1016/j.cma.2016.12.014. URL https://doi.org/10.1016/j.cma.2016.12.014.

D. Boffi, F. Brezzi, and M. Fortin. *Mixed finite element methods and applications*, volume 44 of *Springer Series in Computational Mathematics*. Springer, Heidelberg, 2013. doi: 10.1007/978-3-642-36519-5. URL https://doi.org/10.1007/978-3-642-36519-5.

F. A. Bornemann and P. Deuflhard. The cascadic multigrid method for elliptic problems. *Numer. Math.*, 75(2):135–152, 1996. doi: 10.1007/s002110050234. URL http://dx.doi.org/10.1007/s002110050234.

L. Botti, A. Colombo, and F. Bassi. h-multigrid agglomeration based solution strategies for discontinuous Galerkin discretizations of incompressible flow problems. *J. Comput. Phys.*, 347:382–415, 2017. doi: 10.1016/j.jcp.2017.07.002. URL https://doi.org/10.1016/j.jcp.2017.07.002.

J. H. Bramble and J. E. Pasciak. New convergence estimates for multigrid algorithms. *Math. Comp.*, 49(180):311–329, 1987. doi: 10.2307/2008314. URL https://doi.org/10.2307/2008314.

J. H. Bramble, J. E. Pasciak, and A. H. Schatz. The construction of preconditioners for elliptic problems by substructuring. I. *Math. Comp.*, 47(175):103–134, 1986. doi: 10.2307/2008084. URL http://dx.doi.org/10.2307/2008084.

J. H. Bramble, J. E. Pasciak, and J. Xu. Parallel multilevel preconditioners. In *Numerical analysis 1989 (Dundee, 1989)*, volume 228 of *Pitman Res. Notes Math. Ser.*, pages 23–39. Longman Sci. Tech., Harlow, 1990.

J. H. Bramble, J. E. Pasciak, J. P. Wang, and J. Xu. Convergence estimates for multigrid algorithms without regularity assumptions. *Math. Comp.*, 57(195):23–45, 1991. doi: 10.2307/2938661. URL https://doi.org/10.2307/2938661.

A. Brandt. Multi-level adaptive solutions to boundary-value problems. *Math. Comp.*, 31(138):333–390, 1977. doi: 10.2307/2006422. URL https://doi.org/10.2307/2006422.

A. Brandt and O. E. Livne. *Multigrid techniques—1984 guide with applications to fluid dynamics*, volume 67 of *Classics in Applied Mathematics*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2011. doi: 10.1137/1.9781611970753. URL https://doi.org/10.1137/1.9781611970753. Revised edition of the 1984 original [MR0772748].

A. Brandt, S. McCormick, and J. Ruge. Algebraic multigrid (AMG) for sparse matrix equations. In *Sparsity and its applications (Loughborough, 1983)*, pages 257–284. Cambridge Univ. Press, Cambridge, 1985.

S. C. Brenner. A multigrid algorithm for the lowest-order Raviart-Thomas mixed triangular finite element method. *SIAM J. Numer. Anal.*, 29(3):647–678, 1992. doi: 10.1137/0729042. URL http://dx.doi.org/10.1137/0729042.

S. C. Brenner. Fast solvers for mixed finite element methods. In *Mixed finite element technologies*, volume 509 of *CISM Courses and Lect.*, pages 57–88. SpringerWienNewYork, Vienna, 2009. doi: 10.1007/978-3-211-99094-0_2. URL https://doi.org/10.1007/978-3-211-99094-0_2.

S. C. Brenner and L. R. Scott. *The mathematical theory of finite element methods*, volume 15 of *Texts in Applied Mathematics*. Springer, New York, third edition, 2008. doi: 10.1007/978-0-387-75934-0. URL http://dx.doi.org/10.1007/978-0-387-75934-0.

S. C. Brenner, D.-S. Oh, and L.-Y. Sung. Multigrid methods for saddle point problems: Darcy systems. *Numer. Math.*, 138(2):437–471, 2018. doi: 10.1007/s00211-017-0911-9. URL https://doi.org/10.1007/s00211-017-0911-9.

M. Brezina, R. Falgout, S. MacLachlan, T. Manteuffel, S. McCormick, and J. Ruge. Adaptive algebraic multigrid. *SIAM J. Sci. Comput.*, 27(4):1261–1286, 2006. doi: 10.1137/040614402. URL http://dx.doi.org/10.1137/040614402.

F. Brezzi, J. Douglas, Jr., and L. D. Marini. Recent results on mixed finite element methods for second order elliptic problems. In *Vistas in applied mathematics*, Transl. Ser. Math. Engrg., pages 25–43. Optimization Software, New York, 1986.

W. L. Briggs, V. E. Henson, and S. F. McCormick. *A multigrid tutorial*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, second edition, 2000. doi: 10.1137/1.9780898719505. URL https://doi.org/10.1137/1.9780898719505.

X.-C. Cai and M. Sarkis. A restricted additive Schwarz preconditioner for general sparse linear systems. *SIAM J. Sci. Comput.*, 21(2):792–797, 1999. doi: 10.1137/S106482759732678X. URL https://doi.org/10.1137/S106482759732678X.

Z. Cai, R. R. Parashkevov, T. F. Russell, J. D. Wilson, and X. Ye. Domain decomposition for a mixed finite element method in three dimensions. *SIAM J. Numer. Anal.*, 41(1):181–194, 2003. doi: 10.1137/S0036142996296935. URL https://doi.org/10.1137/S0036142996296935.

C. Canuto and A. Quarteroni. Preconditioned minimal residual methods for Chebyshev spectral calculations. *J. Comput. Phys.*, 60(2):315–337, 1985. doi: 10.1016/0021-9991(85)90010-5. URL https://doi.org/10.1016/0021-9991(85)90010-5.

C. Canuto, R. H. Nochetto, R. Stevenson, and M. Verani. Convergence and optimality of **hp-afem**. *Numer. Math.*, 135(4):1073–1119, 2017. doi: 10.1007/s00211-016-0826-x. URL https://doi.org/10.1007/s00211-016-0826-x.

C. Carstensen, M. Feischl, M. Page, and D. Praetorius. Axioms of adaptivity. *Comput. Math. Appl.*, 67(6):1195–1253, 2014. doi: 10.1016/j.camwa.2013.12.003. URL http://dx.doi.org/10.1016/j.camwa.2013.12.003.

J. M. Cascón, C. Kreuzer, R. H. Nochetto, and K. G. Siebert. Quasi-optimal convergence rate for an adaptive finite element method. *SIAM J. Numer. Anal.*, 46(5):2524–2550, 2008. doi: 10.1137/07069047X. URL http://dx.doi.org/10.1137/07069047X.

T. F. Chan and W. L. Wan. Robust multigrid methods for nonsmooth coefficient elliptic linear systems. *J. Comput. Appl. Math.*, 123(1-2):323–352, 2000. doi: 10.1016/S0377-0427(00)00411-8. URL https://doi.org/10.1016/S0377-0427(00)00411-8.

L. Chen, R. H. Nochetto, and J. Xu. Optimal multilevel methods for graded bisection grids. *Numer. Math.*, 120(1):1–34, 2012. doi: 10.1007/s00211-011-0401-4. URL http://dx.doi.org/10.1007/s00211-011-0401-4.

Z. Chen. Equivalence between and multigrid algorithms for nonconforming and mixed methods for second-order elliptic problems. *East-West J. Numer. Math.*, 4(1):1–33, 1996.

P. G. Ciarlet. *The finite element method for elliptic problems*. North-Holland Publishing Co., Amsterdam-New York-Oxford, 1978. Studies in Mathematics and its Applications, Vol. 4.

L. C. Cowsar, J. Mandel, and M. F. Wheeler. Balancing domain decomposition for mixed finite elements. *Math. Comp.*, 64(211):989–1015, 1995. URL https://doi.org/10.2307/2153480.

P. Daniel, A. Ern, I. Smears, and M. Vohralík. An adaptive *hp*-refinement strategy with computable guaranteed bound on the error reduction factor. *Comput. Math. Appl.*, 76(5):967–983, 2018. doi: 10.1016/j.camwa.2018.05.034. URL https://doi.org/10.1016/j.camwa.2018.05.034.

T. A. Davis, S. Rajamanickam, and W. M. Sid-Lakhdar. A survey of direct methods for sparse linear systems. *Acta Numer.*, 25:383–566, 2016. doi: 10.1017/S0962492916000076. URL https://doi.org/10.1017/S0962492916000076.

V. Dolean, P. Jolivet, and F. Nataf. *An introduction to domain decomposition methods*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2015. doi: 10.1137/1.9781611974065.ch1. URL https://doi.org/10.1137/1.9781611974065.ch1.

V. Dolejší, A. Ern, and M. Vohralík. *hp*-adaptation driven by polynomial-degree-robust a posteriori error estimates for elliptic problems. *SIAM J. Sci. Comput.*, 38 (5):A3220–A3246, 2016. doi: 10.1137/15M1026687. URL http://dx.doi.org/10.1137/15M1026687.

W. Dörfler. A convergent adaptive algorithm for Poisson's equation. *SIAM J. Numer. Anal.*, 33(3):1106–1124, 1996. doi: 10.1137/0733054. URL http://dx.doi.org/10.1137/0733054.

M. Dryja and O. B. Widlund. Towards a unified theory of domain decomposition algorithms for elliptic problems. In *Third International Symposium on Domain Decomposition Methods for Partial Differential Equations (Houston, TX, 1989)*, pages 3–21. SIAM, Philadelphia, PA, 1990.

I. Duff, J. Hogg, and F. Lopez. A new sparse LDL^T solver using a posteriori threshold pivoting. *SIAM J. Sci. Comput.*, 42(2):C23–C42, 2020. doi: 10.1137/18M1225963. URL https://doi.org/10.1137/18M1225963.

E. Efstathiou and M. J. Gander. Why restricted additive Schwarz converges faster than additive Schwarz. *BIT*, 43(suppl.):945–959, 2003. doi: 10.1023/B:BITN.0000014563.33622.1d. URL https://doi.org/10.1023/B:BITN.0000014563.33622.1d.

A. Ern and J.-L. Guermond. *Theory and practice of finite elements*, volume 159 of *Applied Mathematical Sciences*. Springer-Verlag, New York, 2004. doi: 10.1007/978-1-4757-4355-5. URL https://doi.org/10.1007/978-1-4757-4355-5.

R. E. Ewing and J. Wang. Analysis of the Schwarz algorithm for mixed finite elements methods. *RAIRO Modél. Math. Anal. Numér.*, 26(6):739–756, 1992. doi: 10.1051/m2an/1992260607391. URL https://doi.org/10.1051/m2an/1992260607391.

R. E. Ewing and J. Wang. Analysis of multilevel decomposition iterative methods for mixed finite element methods. *RAIRO Modél. Math. Anal. Numér.*, 28(4): 377–398, 1994. doi: 10.1051/m2an/1994280403771. URL https://doi.org/10.1051/m2an/1994280403771.

M. Feischl, M. Page, and D. Praetorius. Convergence and quasi-optimality of adaptive FEM with inhomogeneous Dirichlet data. *J. Comput. Appl. Math.*, 255:481–501, 2014. doi: 10.1016/j.cam.2013.06.009. URL https://doi.org/10.1016/j.cam.2013.06.009.

S. Foresti, G. Brussino, S. Hassanzadeh, and V. Sonnad. Multilevel solution method for the p-version of finite elements. *Computer Physics Communications*, 53(1): 349 – 355, 1989. doi: https://doi.org/10.1016/0010-4655(89)90172-0. URL http://www.sciencedirect.com/science/article/pii/0010465589901720.

M. J. Gander. Schwarz methods over the course of time. *Electron. Trans. Numer. Anal.*, 31:228–255, 2008.

G. Gantner, A. Haberl, D. Praetorius, and B. Stiftner. Rate optimal adaptive FEM with inexact solver for nonlinear operators. *IMA J. Numer. Anal.*, 38(4):1797–1831, 2018. doi: 10.1093/imanum/drx050. URL https://doi.org/10.1093/imanum/drx050.

A. Gholami, D. Malhotra, H. Sundar, and G. Biros. FFT, FMM, or multigrid? A comparative study of state-of-the-art Poisson solvers for uniform and nonuniform grids in the unit cube. *SIAM J. Sci. Comput.*, 38(3):C280–C306, 2016. doi: 10.1137/15M1010798. URL https://doi.org/10.1137/15M1010798.

R. Glowinski and M. F. Wheeler. Domain decomposition and mixed finite element methods for elliptic problems. In *First International Symposium on Domain Decomposition Methods for Partial Differential Equations (Paris, 1987)*, pages 144–172. SIAM, Philadelphia, 1988.

G. H. Golub and C. F. Van Loan. *Matrix computations*. Johns Hopkins Studies in the Mathematical Sciences. Johns Hopkins University Press, Baltimore, MD, third edition, 1996.

M. Griebel and P. Oswald. On the abstract theory of additive and multiplicative Schwarz algorithms. *Numer. Math.*, 70(2):163–180, 1995. doi: 10.1007/s002110050115. URL https://doi.org/10.1007/s002110050115.

M. Griebel, P. Oswald, and M. A. Schweitzer. A particle-partition of unity method. VI. A p-robust multilevel solver. In *Meshfree methods for partial differential equations II*, volume 43 of *Lect. Notes Comput. Sci. Eng.*, pages 71–92. Springer, Berlin, 2005. doi: 10.1007/3-540-27099-X_5. URL https://doi.org/10.1007/3-540-27099-X_5.

P. Grisvard. *Elliptic problems in nonsmooth domains*, volume 24 of *Monographs and Studies in Mathematics*. Pitman (Advanced Publishing Program), Boston, MA, 1985.

W. Gui and I. Babuška. The h, p and h-p versions of the finite element method in 1 dimension. I. The error analysis of the p-version. *Numer. Math.*, 49(6): 577–612, 1986a. doi: 10.1007/BF01389733. URL https://doi.org/10.1007/BF01389733.

W. Gui and I. Babuška. The h, p and h-p versions of the finite element method in 1 dimension. II. The error analysis of the h- and h-p versions. *Numer. Math.*, 49(6):613–657, 1986b. doi: 10.1007/BF01389734. URL https://doi.org/10.1007/BF01389734.

W. Gui and I. Babuška. The h, p and h-p versions of the finite element method in 1 dimension. III. The adaptive h-p version. *Numer. Math.*, 49(6):659–683, 1986c. doi: 10.1007/BF01389735. URL https://doi.org/10.1007/BF01389735.

W. Hackbusch. *Multi-grid methods and applications*, volume 4 of *Springer Series in Computational Mathematics*. Springer, Berlin, 2003.

W. Heinrichs. Line relaxation for spectral multigrid methods. *J. Comput. Phys.*, 77 (1):166–182, 1988. doi: 10.1016/0021-9991(88)90161-1. URL https://doi.org/10.1016/0021-9991(88)90161-1.

M. R. Hestenes and E. Stiefel. Methods of conjugate gradients for solving linear systems. *J. Research Nat. Bur. Standards*, 49:409–436, 1952.

R. Hiptmair, H. Wu, and W. Zheng. Uniform convergence of adaptive multigrid methods for elliptic problems and Maxwell's equations. *Numer. Math. Theory Methods Appl.*, 5(3):297–332, 2012. doi: 10.4208/nmtma.2012.m1128. URL http://dx.doi.org/10.4208/nmtma.2012.m1128.

X. Hu, J. Lin, and L. T. Zikatanov. An adaptive multigrid method based on path cover. *SIAM J. Sci. Comput.*, 41(5):S220–S241, 2019. doi: 10.1137/18M1194493. URL https://doi.org/10.1137/18M1194493.

B. Janssen and G. Kanschat. Adaptive multilevel methods with local smoothing for H^1- and H^{curl}-conforming high order finite element methods. *SIAM J. Sci. Comput.*, 33(4):2095–2114, 2011. doi: 10.1137/090778523. URL https://doi.org/10.1137/090778523.

P. Jiránek, Z. Strakoš, and M. Vohralík. A posteriori error estimates including algebraic error and stopping criteria for iterative solvers. *SIAM J. Sci. Comput.*, 32(3):1567–1590, 2010. doi: 10.1137/08073706X. URL http://dx.doi.org/10.1137/08073706X.

G. Kanschat. Robust smoothers for high-order discontinuous Galerkin discretizations of advection-diffusion problems. *J. Comput. Appl. Math.*, 218(1):53–60, 2008. doi: 10.1016/j.cam.2007.04.032. URL https://doi.org/10.1016/j.cam.2007.04.032.

R. Kehl, R. Nabben, and D. B. Szyld. Adaptive multilevel Krylov methods. *Electron. Trans. Numer. Anal.*, 51:512–528, 2019. doi: 10.1553/etna_vol51s512. URL https://doi.org/10.1553/etna_vol51s512.

C. T. Kelley. *Iterative methods for linear and nonlinear equations*, volume 16 of *Frontiers in Applied Mathematics*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1995. doi: 10.1137/1.9781611970944. URL https://doi.org/10.1137/1.9781611970944.

R. B. Kellogg. On the Poisson equation with intersecting interfaces. *Appl. Anal.*, 4:101–129, 1975. doi: 10.1080/00036817408839086. URL https://doi.org/10.1080/00036817408839086.

M. Kronbichler and W. A. Wall. A performance comparison of continuous and discontinuous Galerkin methods with fast multigrid solvers. *SIAM J. Sci. Comput.*, 40(5):A3423–A3448, 2018. doi: 10.1137/16M110455X. URL https://doi.org/10.1137/16M110455X.

S. Loisel, R. Nabben, and D. B. Szyld. On hybrid multigrid-Schwarz algorithms. *J. Sci. Comput.*, 36(2):165–175, 2008. doi: 10.1007/s10915-007-9183-3. URL https://doi.org/10.1007/s10915-007-9183-3.

H. Lötzbeyer and U. Rüde. Patch-adaptive multilevel iteration. *BIT*, 37(3):739–758, 1997. doi: 10.1007/BF02510250. URL https://doi.org/10.1007/BF02510250. Direct methods, linear algebra in optimization, iterative methods (Toulouse, 1995/1996).

J. P. Lucero Lorca and G. Kanschat. Multilevel Schwarz preconditioners for singularly perturbed symmetric reaction-diffusion systems. *Electron. Trans. Numer. Anal.*, 54:89–107, 2021. doi: 10.1553/etna_vol54s89. URL https://doi-org.libraryproxy.ist.ac.at/10.1553/etna_vol54s89.

J. Mandel. Two-level domain decomposition preconditioning for the p-version finite element method in three dimensions. *Internat. J. Numer. Methods Engrg.*, 29(5): 1095–1108, 1990. doi: 10.1002/nme.1620290513. URL https://doi.org/10.1002/nme.1620290513.

T. P. Mathew. Schwarz alternating and iterative refinement methods for mixed formulations of elliptic problems. II. Convergence theory. *Numer. Math.*, 65(4): 469–492, 1993. doi: 10.1007/BF01385763. URL https://doi.org/10.1007/BF01385763.

S. F. McCormick. *Multilevel adaptive methods for partial differential equations*, volume 6 of *Frontiers in Applied Mathematics*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1989. doi: 10.1137/1.9781611971026. URL https://doi.org/10.1137/1.9781611971026.

D. Meidner, R. Rannacher, and J. Vihharev. Goal-oriented error control of the iterative solution of finite element equations. *J. Numer. Math.*, 17(2):143–172, 2009. doi: 10.1515/JNUM.2009.009. URL http://dx.doi.org/10.1515/JNUM.2009.009.

A. Miraçi, J. Papež, and M. Vohralík. A multilevel algebraic error estimator and the corresponding iterative solver with p-robust behavior. *SIAM J. Numer. Anal.*, 58(5):2856–2884, 2020. doi: 10.1137/19M1275929. URL https://doi.org/10.1137/19M1275929.

A. Miraçi, J. Papež, and M. Vohralík. A-posteriori-steered p-robust multigrid with optimal step-sizes and adaptive number of smoothing steps. *SIAM J. Sci. Comput.*, 2021a. DOI 10.1137/20M1349503.

A. Miraçi, J. Papež, and M. Vohralík. Contractive local adaptive smoothing based on Dörfler marking in a-posteriori-steered p-robust multigrid solvers. *Comput. Methods Appl. Math.*, 2021b. DOI 10.1515/cmam-2020-0024.

W. F. Mitchell. Adaptive refinement for arbitrary finite-element spaces with hierarchical bases. *J. Comput. Appl. Math.*, 36(1):65–78, 1991. ISSN 0377-0427. doi: 10.1016/0377-0427(91)90226-A. URL https://doi.org/10.1016/0377-0427(91)90226-A.

W. F. Mitchell and M. A. McClain. A comparison of hp-adaptive strategies for elliptic partial differential equations. *ACM Trans. Math. Software*, 41(1):Art. 2, 39, 2014. doi: 10.1145/2629459. URL https://doi.org/10.1145/2629459.

P. Morin, R. H. Nochetto, and K. G. Siebert. Convergence of adaptive finite element methods. *SIAM Rev.*, 44(4):631–658, 2002. doi: 10.1137/S0036144502409093. URL https://doi.org/10.1137/S0036144502409093.

A. Napov and Y. Notay. An algebraic multigrid method with guaranteed convergence rate. *SIAM J. Sci. Comput.*, 34(2):A1079–A1109, 2012. doi: 10.1137/100818509. URL http://dx.doi.org/10.1137/100818509.

J. Nocedal and S. J. Wright. *Numerical optimization*. Springer Series in Operations Research and Financial Engineering. Springer, New York, second edition, 2006.

Y. Notay and A. Napov. A massively parallel solver for discrete Poisson-like problems. *J. Comput. Phys.*, 281:237–250, 2015. doi: 10.1016/j.jcp.2014.10.043. URL https://doi.org/10.1016/j.jcp.2014.10.043.

P. Oswald. *Multilevel finite element approximation*. Teubner Skripten zur Numerik. B. G. Teubner, Stuttgart, 1994. doi: 10.1007/978-3-322-91215-2. URL https://doi.org/10.1007/978-3-322-91215-2.

J. Papež and M. Vohralík. Inexpensive guaranteed and efficient upper bounds on the algebraic error in finite element discretizations. HAL preprint 02422851, 2019. URL https://hal.inria.fr/hal-02422851.

J. Papež, Z. Strakoš, and M. Vohralík. Estimating and localizing the algebraic and total numerical errors using flux reconstructions. *Numer. Math.*, 138(3):681–721, 2018. doi: 10.1007/s00211-017-0915-5. URL https://link.springer.com/article/10.1007%2Fs00211-017-0915-5.

J. Papež, U. Rüde, M. Vohralík, and B. Wohlmuth. Sharp algebraic and total a posteriori error bounds for h and p finite elements via a multilevel approach. *Comput. Methods Appl. Mech. Engrg.*, 371:113243, 2020. doi: 10.1016/j.cma.2020.113243. URL https://doi.org/10.1016/j.cma.2020.113243.

L. F. Pavarino. Additive Schwarz methods for the p-version finite element method. *Numer. Math.*, 66(4):493–515, 1994. doi: 10.1007/BF01385709. URL https://doi.org/10.1007/BF01385709.

A. Quarteroni and G. Sacchi Landriani. Domain decomposition preconditioners for the spectral collocation method. *J. Sci. Comput.*, 3(1):45–76, 1988. doi: 10.1007/BF01066482. URL https://doi.org/10.1007/BF01066482.

A. Quarteroni and A. Valli. *Domain decomposition methods for partial differential equations*. Numerical Mathematics and Scientific Computation. The Clarendon Press Oxford University Press, New York, 1999. Oxford Science Publications.

P.-A. Raviart and J.-M. Thomas. A mixed finite element method for 2nd order elliptic problems. In *Mathematical aspects of finite element methods (Proc. Conf., Consiglio Naz. delle Ricerche (C.N.R.), Rome, 1975)*, pages 292–315. Lecture Notes in Math., Vol. 606. Springer, Berlin, 1977.

U. Rüde. *Mathematical and computational techniques for multilevel adaptive methods*, volume 13 of *Frontiers in Applied Mathematics*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1993. doi: 10.1137/1.9781611970968. URL http://dx.doi.org/10.1137/1.9781611970968.

J. W. Ruge and K. Stüben. Algebraic multigrid. In *Multigrid methods*, volume 3 of *Frontiers Appl. Math.*, pages 73–130. SIAM, Philadelphia, PA, 1987.

Y. Saad. *Iterative methods for sparse linear systems*. Society for Industrial and Applied Mathematics, Philadelphia, PA, second edition, 2003.

Y. Saad and M. H. Schultz. GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM J. Sci. Statist. Comput.*, 7(3):856–869, 1986. doi: 10.1137/0907058. URL http://dx.doi.org/10.1137/0907058.

J. Schöberl. C++11 Implementation of Finite Elements in NGsolve. Technical report, ASC Report 30/2014, Institute for Analysis and Scientific Computing, Vienna University of Technology, 2014. URL https://ngsolve.org.

J. Schöberl, J. M. Melenk, C. Pechstein, and S. Zaglmayr. Additive Schwarz preconditioning for p-version triangular and tetrahedral finite elements. *IMA J. Numer. Anal.*, 28(1):1–24, 2008. doi: 10.1093/imanum/drl046. URL https://doi.org/10.1093/imanum/drl046.

H. A. Schwarz. Über einen Grenzübergang durch alternierendes Verfahren. *Vierteljahrsschrift der Naturforschenden Gesellschaft in Zürich,* 15: 272–286, 1870.

E. G. Sewell. *Automatic generation of triangulations for piecewise polynomial approximation*. ProQuest LLC, Ann Arbor, MI, 1972. Thesis (Ph.D.)–Purdue University.

P. Šolín, K. Segeth, and I. Doležel. *Higher-order finite element methods*. Studies in Advanced Mathematics. Chapman & Hall/CRC, Boca Raton, FL, 2004.

H. Sundar, G. Stadler, and G. Biros. Comparison of multigrid algorithms for high-order continuous finite element discretizations. *Numer. Linear Algebra Appl.*, 22(4):664–680, 2015. doi: 10.1002/nla.1979. URL https://doi.org/10.1002/nla.1979.

B. Szabó and I. Babuška. *Finite element analysis*. A Wiley-Interscience Publication. John Wiley & Sons Inc., New York, 1991.

A. Thekale, T. Gradl, K. Klamroth, and U. Rüde. Optimizing the number of multigrid cycles in the full multigrid algorithm. *Numer. Linear Algebra Appl.*, 17(2-3):199–210, 2010. doi: 10.1002/nla.697. URL https://doi.org/10.1002/nla.697.

A. Toselli and O. Widlund. *Domain decomposition methods—algorithms and theory*, volume 34 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, 2005. doi: 10.1007/b137868. URL https://doi.org/10.1007/b137868.

H. A. van der Vorst. Bi-CGSTAB: a fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems. *SIAM J. Sci. Statist. Comput.*, 13(2):631–644, 1992.

R. S. Varga. *Matrix iterative analysis*, volume 27 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, expanded edition, 2000. doi: 10.1007/978-3-642-05156-2. URL https://doi.org/10.1007/978-3-642-05156-2.

R. Verfürth. *A review of a posteriori error estimation and adaptive mesh-refinement techniques*. Teubner-Wiley, Stuttgart, 1996.

R. Verfürth. *A posteriori error estimation techniques for finite element methods*. Numerical Mathematics and Scientific Computation. Oxford University Press, Oxford, 2013. doi: 10.1093/acprof:oso/9780199679423.001.0001. URL https://doi.org/10.1093/acprof:oso/9780199679423.001.0001.

T. Warburton. An explicit construction of interpolation nodes on the simplex. *J. Engrg. Math.*, 56(3):247–262, 2006. doi: 10.1007/s10665-006-9086-6. URL https://doi.org/10.1007/s10665-006-9086-6.

H. Wu and Z. Chen. Uniform convergence of multigrid V-cycle on adaptively refined finite element meshes for second order elliptic problems. *Sci. China Ser. A*, 49 (10):1405–1429, 2006. doi: 10.1007/s11425-006-2005-5. URL http://dx.doi.org/10.1007/s11425-006-2005-5.

J. Xu. Iterative methods by space decomposition and subspace correction. *SIAM Rev.*, 34(4):581–613, 1992. doi: 10.1137/1034116. URL https://doi.org/10.1137/1034116.

J. Xu, L. Chen, and R. H. Nochetto. Optimal multilevel methods for $H(\mathrm{grad})$, $H(\mathrm{curl})$, and $H(\mathrm{div})$ systems on graded and unstructured grids. In *Multiscale, nonlinear and adaptive approximation*, pages 599–659. Springer, Berlin, 2009. doi: 10.1007/978-3-642-03413-8_14. URL http://dx.doi.org/10.1007/978-3-642-03413-8_14.

S. N. Yeralan, A. Davis, T, W. M. Sid-Lakhdar, and S. Ranka. Algorithm 980: sparse QR factorization on the GPU. *ACM Trans. Math. Software*, 44(2):Art. 17, 29, 2017. doi: 10.1145/3065870. URL https://doi.org/10.1145/3065870.

X. Zhang. Multilevel Schwarz methods. *Numer. Math.*, 63(4):521–539, 1992. doi: 10.1007/BF01385873. URL https://doi.org/10.1007/BF01385873.