FACULTY
OF MATHEMATICS
AND PHYSICS
Charles University

DOCTORAL THESIS

Petr Vacek

Multigrid methods for large-scale problems: approximate coarsest-level solves and mixed precision computation

Department of Numerical Mathematics

Supervisor of the doctoral thesis: Erin Claire Carson, Ph.D.
Study programme: Computational Mathematics

Prague 2024

I declare that I carried out this doctoral thesis on my own, and only with the cited sources, literature and other professional sources. I understand that my work relates to the rights and obligations under the Act No. 121/2000 Sb., the Copyright Act, as amended, in particular the fact that the Charles University has the right to conclude a license agreement on the use of this work as a school work pursuant to Section 60 subsection 1 of the Copyright Act.

In .... date ....

Author's signature

I would like to start by thanking my supervisor Dr. Erin Carson. Her guidance and support during my PhD studies helped me become a better mathematician and researcher. I also appreciate her patience, kindness and optimism. I am happy that I could be a member of your group.

I am also grateful to prof. Zdeněk Strakoš and Dr. Jan Papež, for their mentoring and continuous help since my bachelor studies.

I would like to further thank professors Kirk Soodhalter, Ulrich Rüde, and Hartwig Anzt for giving me the opportunity to visit their research groups in Ireland and Germany. All these stays were really valuable experiences for me from both a professional and a personal standpoint.

This thesis concludes not only my PhD studies but also my time at Charles University, where I have started eleven years ago. Throughout this time many people at the Department of Numerical Mathematics have helped me to learn and overcome many challenges. I was fortunate enough to share this time with my fellow PhD students, to whom I am grateful.

Most of all, I would like to thank my family, my girlfriend and my friends for being a part of my life, for always believing in me, supporting me and for making my life happier.

This work was supported by Charles University PRIMUS project PRIMUS/19/SCI/11, grant SVV-2023-260711, grant SVV-2023-260711, Exascale Computing Project (17-SC-20-SC), a collaborative effort of the U.S. Department of Energy Office of Science and the National Nuclear Security Administration, and by the European Union (ERC, inEXASCALE, 101075632). Views and opinions expressed are those of the authors only and do not necessarily reflect those of the European Union or the European Research Council. Neither the European Union nor the granting authority can be held responsible for them.

Title: Multigrid methods for large-scale problems: approximate coarsest-level solves and mixed precision computation

Author: Petr Vacek

Department: Department of Numerical Mathematics

Supervisor: Erin Claire Carson, Ph.D., Department of Numerical Mathematics

Abstract: The development of new computational hardware components opens possibilities to solve larger and larger problems. It, however, also brings new challenges. In this thesis we study multigrid methods for solving large-scale systems of linear equations. The multigrid approach relies on having a hierarchy of problems, ranging from the smallest (coarsest-level) problem to the original (finest-level) problem. We focus on settings where even the problem on the coarsest-level is large and can be solved only approximately. Such hierarchies arise, for example, when solving problems on domains with complicated geometry or when computing in parallel. We present an approach for analyzing the effects of approximate coarsest-level solves on the convergence of the multigrid V-cycle scheme and derive new coarsest-level stopping criteria tailored to multigrid methods. The multigrid hierarchy can be also used to construct residual-based a posteriori error estimates. We present a new approximation of the term associated with the coarsest level, which results in effective and robust estimates. Finally, we present a new formulation of a mixed precision V-cycle method and provide its finite precision analysis. We apply the analysis to understand how to choose the finite precisions inside a V-cycle scheme with smoothing based on incomplete Cholesky factorization.

Keywords: multigrid, coarsest-level stopping criteria, multilevel residual-based error estimator, mixed precision, smoother based on incomplete Cholesky factorization

Název práce: Víceúrovňové metody pro řešení velkých problémů: přibližné řešení na nejhrubší síti, počítání ve smíšené přesnosti

Autor: Petr Vacek

Katedra: Katedra numerické matematiky

Vedoucí disertační práce: Erin Claire Carson, Ph.D., Katedra numerické matematiky

Abstrakt: Vývoj nového výpočetního hardwaru otevírá možnosti řešení větších a větších problémů. Přináší ale také nové výzvy. V této práci se zabýváme víceúrovňovými metodami pro řešení velkých soustav lineárních rovnic. Víceúrovňové metody využívají hierarchii problémů s různými velikostmi, od nejmenšího problému (nejhrubší úroveň) až po původní problém (nejjemnější úroveň). V této práci uvažujeme hierarchie, kde i problém na nejhrubší úrovni je velký a jeho řešení lze spočítat pouze přibližně. Takové hierarchie vznikají například při řešení problémů na oblastech se složitou geometrií nebo při paralelních výpočtech. Jedním z hlavních výsledků této práce je nový přístup k analýze vlivu přibližného řešení na nejhrubší úrovni na konvergenci V-cycle schématu a odvození nového zastavovacího kritéria pro řešení problému na nejhrubší úrovni. Víceúrovňovou hierarchii je možné využít také ke konstrukci a posteriori odhadu chyby na základě rezidua. Dalším hlavním výsledkem této práce je nový postup aproximace členu odpovídajícímu nejhrubší úrovni, který vede k efektivním a robustním odhadům. Na závěr formulujeme V-cycle schéma využívající počítání v aritmetikách s různými konečnými přesnostmi a odvodíme odhad na chybu způsobenou počítáním v aritmetikách s konečnou přesností. Tyto výsledky následně používáme, abychom zjistili, jak volit aritmetiky s konečnou přesností ve V-cycle schématu se zhlazováním založeným na neúplném Choleského rozkladu.

Klíčová slova: víceúrovňové metody, zastavovací kritérium na nejhrubší síti, víceúrovňový odhad chyby založený na residuu, počítání ve smíšené přesnosti, zhlazovač založený na neúplném Choleského rozkladu

Contents

List of Publications

Journals

[J1] P. Vacek, E. Carson, and K.M. Soodhalter. “The Effect of Approximate Coarsest-Level Solves on the Convergence of Multigrid V-Cycle Methods”, In: *SIAM Journal on Scientific Computing*, 46:4 (2024), pp. A2634-A2659, `https://doi.org/10.1137/23M1578255`.

Introduction

Numerical simulations play an important role in many areas of scientific research and in many industrial applications, such as medicine, weather forecasting, physics, and engineering design optimization. The continuous development of more powerful hardware components opens possibilities to run simulations of larger and larger sizes. Utilizing the full potential of the new hardware is, however, not always straightforward nor easy. Existing numerical methods have to be redesigned or optimized to better exploit the hardware characteristics.

Numerical simulations frequently involve models described using partial differential equations (PDEs). Solving PDEs typically consists of a discretization of the continuous problem and the subsequent solution of a discrete algebraic problem. The associated algebraic problem may be large and hard to solve, especially when an accurate approximation to the continuous solution of the PDEs is needed. Successful design and implementation of numerical methods requires understanding of all stages of the computational process and their interactions. The individual components thus should not be studied separately, but rather as parts of the whole process.

In this thesis we study multigrid methods for solving systems of linear equations. Our primary focus is on systems coming from the discretization of PDEs, but some of the presented results are valid in more general cases. For an introduction to multigrid methods we refer to, e.g., [3, 16], or the author's master's thesis [18]. The multigrid approach relies on having a hierarchy of problems. The hierarchy can be constructed in two ways: in *geometric multigrid* a continuous problem is discretized on multiple nested meshes, whereas in *algebraic multigrid* the construction is based on the properties of the system matrix. The levels in a multigrid hierarchy are referred to as coarse or fine, with the coarsest level containing the problem of the smallest size, and the finest level containing the original problem we aim to solve. The approximate solution is computed using so called smoothing on fine levels and by solving a system of linear equations on the coarsest level. The intermediate results are transferred between the levels using prolongation and restriction operators. There are various multigrid schemes (V-cycle, W-cycle, full multigrid) differing in the pattern in which the individual levels are visited during the computation.

Smoothing typically consists of applying few iterations of a stationary iterative method such as the Richardson, Jacobi, or Gauss-Seidel methods. The solver on the coarsest-level is chosen based on the size and difficulty of the coarsest-level problem. If the size permits, direct methods based on Cholesky or LU factorization are usually applied. Multigrid methods are in practice also used with hierarchies where the coarsest-level problem is large and can be solved only approximately; see e.g., [4, 8]. This arises, for example, when solving problems on complicated domains or when running large-scale simulations on parallel computers. Frequently used approximate coarsest-level solvers are iterative Krylov subspace methods or direct methods based on block-low rank (BLR) approximation. Application of these solvers requires additional specifications, e.g., choosing a stopping criteria for the iterative solvers or setting a value of the low-rank threshold parameter for the BLR solver. The accuracy of the computed approximation then depends on the concrete setting, which is usually chosen based on the experience of the users with the problems and multigrid methods.

Convergence analysis of multigrid methods is typically done under the assumption that the coarsest-level problem is solved exactly; see, e.g., [23, 21]. This assumption is, however, not satisfied in practical computation either due to the use of approximate solvers, or due to the finite precision errors, or both. There are papers (e.g., [12, 22]) allowing more general coarsest-level solvers, but they do not cover some of the frequently used solvers in practice, for example a Krylov subspace method stopped with a residual-based criterion. This leads to the first two research questions considered in this thesis:

a) Can we analytically describe how the accuracy of the coarsest-level solver affects the convergence behavior of the multigrid method?

b) Can we design effective stopping criteria for an iterative coarsest-level solver such that the multigrid method converges in nearly the same number of iterations as its variant with an exact coarsest-level solver?

We focus on these questions in Chapter 1, which is based on the paper [19].

The multigrid hierarchy can also be used to construct residual-based a posteriori error estimates on total and algebraic errors; see, e.g., [2, 6], [13, Section 2.6], and [8, Sections 4.1–4.3]. The estimates presented in the literature, however, require the computation of an error term associated with the coarsest-level. When using multigrid hierarchies with large coarsest-level problems, this term can be in practice computed only approximately. This leads to our next research question:

c) Consider the residual-based multilevel a posteriori error estimates such as in [13, Section 2.6]. Is it possible to compute the term associated with the coarsest-level approximately while preserving the efficiency and accuracy of the estimate?

We address this question in Chapter 2, which is based on the paper [20].

Modern parallel computers support computing in multiple precisions; see, e.g., the list of 500 most powerful commercially available computer systems `https://top500.org/`. There is extensive ongoing research on numerical methods exploiting this hardware feature; see, e.g., the surveys [1, 7]. Some methods utilizing computation in multiple precisions are able to achieve the same overall accuracy as their uniform precision counterparts, in a smaller amount of time, requiring less memory and consuming less energy.

Mixed precision variants of multigrid methods have been implemented and tested on various problems; see, e.g., [17, 24]. Finite precision error analysis of multigrid method was presented in the series of papers [10, 14, 11]. In these works, the authors present a formulation of the multigrid V-cycle method allowing using up to three different precisions on the finest level and a different precision on each coarse level. They assume that the application of a smoothing routine on a concrete level is done in one precision associated with the level. The authors discuss basic smoothing routines, such as routines based on the Richardson and Jacobi methods.

In practice, more computationally intensive smoothers are also used. Examples of these are methods based on incomplete Cholesky (IC) or LU factorizations, used, e.g., when solving PDEs with high anisotropy; see, e.g., [9, 5, 15]. Using IC smoothing requires precomputing the IC factorization once, and solving triangular systems with the IC factor and its transpose when the smoother is applied. These operations may be computationally intensive in comparison to other parts of the V-cycle scheme. This leads us to our fourth research question:

d) Can the execution time of the mixed precision V-cycle method with IC smoothers be reduced by introducing additional precisions for the applications of the smoothers? For example, using different precisions for storing the IC factors or solving the triangular systems. Can we analytically describe the requirements on these individual precisions?

We focus on this in Chapter 3.

The thesis closes with a conclusion including the formulation of open problems. We note that each chapter introduces its own notation.

Bibliography

[1] A. Abdelfattah et al. "A survey of numerical linear algebra methods utilizing mixed-precision arithmetic". In: *The International Journal of High Performance Computing Applications* 35.4 (2021), pp. 344–369. DOI: 10.1177/10943420211003313.

[2] R. Becker, C. Johnson, and R. Rannacher. "Adaptive error control for multigrid finite element methods". In: *Computing* 55.4 (1995), pp. 271–288. DOI: 10.1007/BF02238483.

[3] W. L. Briggs, V. E. Henson, and S. F. McCormick. *A Multigrid Tutorial.* Second. Philadelphia, PA: SIAM, 2000, pp. xii+193. DOI: 10.1137/1.9780898719505.

[4] A. Buttari et al. "Block low-rank single precision coarse grid solvers for extreme scale multigrid methods". In: *Numerical Linear Algebra with Applications* 29.1 (2022), e2407. DOI: 10.1002/nla.2407.

[5] D. Drzisga, A. Wagner, and B. Wohlmuth. "A Matrix-Free ILU Realization Based on Surrogates". In: *SIAM Journal on Scientific Computing* 45.6 (2023), pp. C304–C329. DOI: 10.1137/22M1529415.

[6] H. Harbrecht and R. Schneider. "A note on multilevel based error estimation". In: *Comput. Methods Appl. Math.* 16.3 (2016), pp. 447–458. DOI: 10.1515/cmam-2016-0013.

[7] N. J. Higham and T. Mary. "Mixed precision algorithms in numerical linear algebra". In: *Acta Numerica* 31 (2022), pp. 347–414. DOI: 10.1017/S0962492922000022.

[8] M. Huber. "Massively parallel and fault-tolerant multigrid solvers on petascale systems". PhD thesis. Technical University of Munich, Germany, 2019. URL: http://www.dr.hut-verlag.de/978-3-8439-3917-1.html.

[9] R. Kettler and P. Wesseling. "Aspects of multigrid methods for problems in three dimensions". In: *Applied Mathematics and Computation* 19.1 (1986), pp. 159–168. DOI: 10.1016/0096-3003(86)90102-5.

[10] S. F. McCormick, J. Benzaken, and R. Tamstorf. “Algebraic Error Analysis for Mixed-Precision Multigrid Solvers”. In: *SIAM Journal on Scientific Computing* 43.5 (2021), S392–S419. DOI: 10.1137/20M1348571.

[11] S. F. McCormick and R. Tamstorf. “Rounding-Error Analysis of Multigrid V-Cycles”. In: *SIAM Journal on Scientific Computing* (2024), S88–S95. DOI: 10.1137/23M1582898.

[12] Y. Notay. “Convergence analysis of perturbed two-grid and multigrid methods”. In: *SIAM Journal on Numerical Analysis* 45.3 (2007), pp. 1035–1044. DOI: 10.1137/060652312.

[13] U. Rüde. *Mathematical and computational techniques for multilevel adaptive methods*. Philadelphia, PA: SIAM, 1993.

[14] R. Tamstorf, J. Benzaken, and S. F. McCormick. “Discretization -Error-Accurate Mixed-Precision Multigrid Solvers”. In: *SIAM Journal on Scientific Computing* 43.5 (2021), S420–S447. DOI: 10.1137/20M1349230.

[15] S. Thomas et al. “Scaled ILU smoothers for Navier–Stokes pressure projection”. In: *International Journal for Numerical Methods in Fluids* 96.4 (2024), pp. 537–560. DOI: 10.1002/fld.5254.

[16] U. Trottenberg, C. W. Oosterlee, and A. Schuller. *Multigrid*. London: Academic Press, 2001.

[17] Y.-H. M. Tsai, N. Beams, and H. Anzt. “Mixed Precision Algebraic Multigrid on GPUs”. In: *Parallel Processing and Applied Mathematics*. Ed. by R. Wyrzykowski et al. Cham: Springer International Publishing, 2023, pp. 113–125. DOI: 10.1007/978-3-031-30442-2_9.

[18] P. Vacek. “Multilevel methods”. Master’s thesis. Charles University, 2020. URL: http://hdl.handle.net/20.500.11956/116819.

[19] P. Vacek, E. Carson, and K. M. Soodhalter. “The Effect of Approximate Coarsest-Level Solves on the Convergence of Multigrid V-Cycle Methods”. In: *SIAM Journal on Scientific Computing* 46.4 (2024), A2634–A2659. DOI: 10.1137/23M1578255.

[20] P. Vacek, J. Papež, and Z. Strakoš. *A posteriori error estimates based on multilevel decompositions with large problems on the coarsest level*. 2024. arXiv: 2405.06532 [math.NA].

[21] J. Xu. “Iterative methods by space decomposition and subspace correction”. In: *SIAM Review* 34.4 (1992), pp. 581–613. DOI: 10.1137/1034116.

[22] X. Xu and C.-S. Zhang. “Convergence Analysis of Inexact Two-Grid Methods: A Theoretical Framework”. In: *SIAM Journal on Numerical Analysis* 60.1 (2022), pp. 133–156. DOI: 10.1137/20M1356075.

[23] H. Yserentant. “Old and new convergence proofs for multigrid methods”. In: *Acta Numerica* 2 (1993), pp. 285–326.

[24] Y. Zong et al. “FP16 Acceleration in Structured Multigrid Preconditioner for Real-World Applications”. In: *Proceedings of the 53rd International Conference on Parallel Processing*. ICPP ’24. Gotland, Sweden: Association for Computing Machinery, 2024, pp. 52–62. DOI: 10.1145/3673038.3673040.

1 The effect of approximate coarsest-level solves on the convergence of multigrid V-cycle methods

In this chapter, we focus on the first two questions stated in the introduction:

a) Can we analytically describe how the accuracy of the coarsest-level solver affects the convergence behavior of the multigrid method?

b) Can we design effective stopping criteria for an iterative coarsest-level solver such that the multigrid method converges in nearly the same number of iterations as its variant with an exact coarsest-level solver?

Motivated by these questions, we propose an approach to algebraically analyze the effect of approximate coarsest-level solves in the multigrid V-cycle method for symmetric positive definite (SPD) problems. We design new coarsest-level stopping criteria tailored to multigrid methods and discuss the convergence of methods with frequently used criteria in practice, e.g., criteria based on the Euclidean norm of the relative residual.

This chapter contains a pre-copyedited version of the paper: P. Vacek, E. Carson, and K.M. Soodhalter. "The Effect of Approximate Coarsest-Level Solves on the Convergence of Multigrid V-Cycle Methods", In: *SIAM Journal on Scientific Computing*, 46:4 (2024), pp. A2634-A2659, `https://doi.org/10.1137/23M1578255`.

1.1 Introduction

Multigrid methods [3, 4, 21, 10] are frequently used when solving systems of linear equations, and can be applied either as standalone solvers or as preconditioners for iterative methods. There are two types of multigrid; *geometric*: wherein the hierarchy of systems is obtained by discretizations of an infinite dimensional problem on a sequence of nested meshes; and *algebraic*: wherein the coarse systems are assembled based on the algebraic properties of the matrix. Within each multigrid cycle, the approximation is computed using smoothing on fine levels and solving a system of linear equations on the coarsest level. Smoothing on the fine levels is typically done via a few iterations of a stationary iterative method. The particular solver used for the problem on the coarsest level depends on its size and difficulty. If the size of the problem permits, it is typical to use a direct solver based on LU or Cholesky decomposition.

In this text, we focus on settings where the problem on the coarsest level is large and the use of direct solvers based on LU or Cholesky decomposition may be ineffective or impossible to realize. Such settings may arise, for example, when using geometric multigrid methods to solve problems on complicated domains. The mesh associated with the coarsest level must resolve the domain with certain accuracy. This can yield a large number of degrees of freedom. One possible solution to this issue is to solve the coarsest-level problem using algebraic multigrid, which can introduce additional coarse levels that are not related to the geometry of the problem.

Another setting where large coarsest-level problems may be present is when we use multigrid methods on parallel computers. In parallel computing, the degrees of freedom are assigned to different processors or accelerators. The computation is done in parallel on the individual processors and the results are communicated between them. A challenge for effective parallel implementation of multigrid methods is that the amount of computation on coarse levels decreases at a faster rate than the amount of communication; see e.g., the discussion in the introduction of [5]. One possible solution is to treat this issue by redistribution of the coarse-level problems to a smaller number of processors; see e.g., [8, 14, 20]. Another solution may be to use communication-avoiding methods on the coarse levels; see e.g., [22].

In this paper, we instead consider treating the still large-scale coarsest-level problem by solving *inexactly.* Frequently used solvers for large scale coarsest-level problems include Krylov subspace methods and direct approximate solvers; see, e.g., [12], where the author considers the preconditioned conjugate gradient method, or [5], where the authors study the use of a block low-rank (BLR) low precision direct solver. These solvers approximate the coarsest-level solution to an accuracy which is determined by the choice of a stopping criteria or affected by the choice of the low-rank threshold and finite precision. These parameters are often chosen in practice based on the experience of the user with concrete problems and methods with the goal of balancing the cost of the coarsest-level solve and the total number of V-cycles required for convergence. In Section 1.2.1 we present a motivating numerical experiments, which illustrate how the choice of the accuracy of the coarsest-level solver may affect the convergence of the multigrid V-cycle method.

A general analysis of the effects of the accuracy of the coarsest-level solver on the convergence behaviour of multilevel methods is, to our knowledge, not present in the literature. Multigrid methods are typically analyzed under the assumption that the problem on the coarsest level is solved exactly; see, e.g., [25, 23]. An algebraic analysis of perturbed two grids methods and its application to the analysis of other multigrid schemes with approximate coarsest-level solvers can be found in [19, 24]. The authors derive estimates of the worst-case convergence rate of the methods. The results are, however, obtained under the assumption that the action of the solver on the coarsest level can be expressed using a symmetric positive definite matrix. This is not true for frequently used solvers, e.g., for a Krylov subspace method stopped using a relative residual stopping criterion. A more general setting is considered in the paper [15], which presents the first analysis of mixed precision multigrid solvers. The authors assume that the action of the solver on the coarsest level can be expressed using a non-singular matrix.

In this paper, we propose an approach to algebraically analyze the effect of approximate coarsest-level solves in the multigrid V-cycle method for symmetric positive definite (SPD) problems. The main methodology of our approach is to view the inexact V-cycle (inV-cycle) method as a perturbation of the exact V-cycle (exV-cycle) method in the following sense. We express the error of the approximation computed by one V-cycle with an approximate coarsest-level solver as the error of the approximation computed by one V-cycle with an exact coarsest-level solver plus the difference of the two approximations. We show that the difference can be expressed as a matrix times the error of the coarsest-level solver. The matrix describes how the error from the coarsest level is propagated to the finest level. Moreover, we consider two assumptions on the accuracy of the coarsest-level solver: a *relative* assumption, where the error of the coarsest-level solver is less than a factor of the error of the previous finest-level approximation, and an *absolute* assumption, where the error of the coarsest-level solver is less than a certain constant. Based on the relative assumption we derive an estimate on the convergence rate of the inV-cycle method and discuss its uniform convergence. Utilizing the absolute assumption we get an estimate on the difference between the approximation computed by the inV-cycle method and the exV-cycle method after a number of V-cycle iterations. The analysis is done assuming exact arithmetic computations, aside from the computation of the coarsest level solutions. The model is agnostic about what coarsest-level solver is used; we only assume that the error on the coarsest level satisfies certain assumptions.

The paper is organized as follows. In Section 1.2 we establish the notation, state the V-cycle method and present a motivating numerical experiments, which illustrate that the choice of the accuracy of the coarsest-level solver can significantly affect the convergence of the V-cycle method. In Section 1.3 we present an analysis of the V-cycle method with an approximate coarsest-level solver. The results are applied to describe the possible effects of the choice of the tolerance in a coarsest-level relative residual stopping criterion in Section 1.4. New stopping criteria based on the absolute coarsest-level accuracy assumption are derived in Section 1.5. Finally, we present a series of numerical experiments illustrating the obtained results in Section 1.6. The text closes with conclusions and discussion of open problems in Section 1.7.

1.2 Notation and motivating experiments

We study the multigrid V-cycle method for finding an approximate solution of the following problem. Given an SPD matrix $\mathbf{A} \in \mathbb{R}^{n\times n}$ and a right-hand side vector $\mathbf{b} \in \mathbb{R}^n$ find the vector $\mathbf{x} \in \mathbb{R}^n$ such that

$$\mathbf{A}\mathbf{x} = \mathbf{b}.$$

We consider a hierarchy of $J+1$ levels numbered from zero to J, where level zero is the coarsest level and level J the finest level. Each level contains a system matrix $\mathbf{A}_j \in \mathbb{R}^{n_j\times n_j}$, with $\mathbf{A}_J = \mathbf{A}$. Information is transferred between the $(j-1)$th level and the jth level using a full rank prolongation matrix $\mathbf{P}_j \in \mathbb{R}^{n_j \times n_{j-1}}$, respectively its transpose. We assume that the system matrices and the prolongation matrices satisfy the so called *Galerkin condition*, i.e.,

$$\mathbf{A}_{j-1} = \mathbf{P}_j^\top \mathbf{A}_j \mathbf{P}_j, \quad j = 1\ldots, J. \tag{1.1}$$

We use the notation $\mathbf{A}_{0:j}$, for the sequence of matrices $\mathbf{A}_0, \ldots, \mathbf{A}_j$. Let $\|\cdot\|$ denote the Euclidean vector norm and let $\|\cdot\|_{\mathbf{A}_j} = \|\mathbf{A}_j^{\frac{1}{2}} \cdot \|$ denote the $\mathbf{A}_j$ vector norm, also called the energy norm. We use the same notation for the matrix norms generated by the associated vector norms. Let $\mathbf{I}_j \in \mathbb{R}^{n_j \times n_j}$ denote the identity matrix on the jth level.

We assume that the pre- and post- smoothing on levels $j = 1, \ldots, J$ can be expressed in the form

$$\mathbf{v}_j = \mathbf{v}_j + \mathbf{M}_j(\mathbf{f}_j - \mathbf{A}_j\mathbf{v}_j) \quad \text{and} \quad \mathbf{v}_j = \mathbf{v}_j + \mathbf{N}_j(\mathbf{f}_j - \mathbf{A}_j\mathbf{v}_j),$$

respectively, where $\mathbf{v}_j$ and $\mathbf{f}_j$ are an approximation and a right-hand side on the jth level and $\mathbf{M}_j \in \mathbb{R}^{n_j \times n_j}$ and $\mathbf{N}_j \in \mathbb{R}^{n_j \times n_j}$ are non-singular matrices satisfying

$$\|\mathbf{I}_j - \mathbf{M}_j\mathbf{A}_j\|_{\mathbf{A}_j} < 1 \quad \text{and} \quad \|\mathbf{I}_j - \mathbf{N}_j\mathbf{A}_j\|_{\mathbf{A}_j} < 1. \tag{1.2}$$

This assumption yields monotone convergence of the smoothers as standalone solvers in the $\mathbf{A}_j$-norms. Frequently used smoothers, e.g., a few iterations of a classic stationary iterative method such as damped Jacobi or Gauss-Seidel, typically satisfy these assumptions; see, e.g., the discussion in [25, p. 293] or [23]. We also consider multilevel schemes, where either pre- or post- smoothing is not used, i.e., where formally either $\mathbf{M}_j$, $j = 1, \ldots, J$ or $\mathbf{N}_j$, $j = 1, \ldots, J$, are zero matrices.

Given an approximation $\mathbf{x}^{\text{prev}}$ to the solution $\mathbf{x}$, the approximation after one iteration of the V-cycle method is computed by calling Algorithm 1.1 as (see, e.g., [21, pp. 47–48])

$$\mathbf{x}^{\text{new}} = \mathbf{V}(\mathbf{A}_{0:J}, \mathbf{M}_{1:J}, \mathbf{N}_{1:J}, \mathbf{P}_{1:J}, \mathbf{b}, \mathbf{x}^{\text{prev}}, J).$$

We distinguish between the *exV-cycle method* and the *inV-cycle method* based on whether the coarsest-level problem is solved exactly or not.

Algorithm 1.1 V-cycle scheme, $\mathbf{V}(\mathbf{A}_{0:j}, \mathbf{M}_{1:j}, \mathbf{N}_{1:j}, \mathbf{P}_{1:j}, \mathbf{f}_j, \mathbf{v}_j^{[0]}, j)$.

```
if $j \neq 0$ then
  $\mathbf{v}_j^{[1]} = \mathbf{v}_j^{[0]} + \mathbf{M}_j(\mathbf{f}_j - \mathbf{A}_j\mathbf{v}_j^{[0]})$ {pre-smoothing}
  $\mathbf{f}_{j-1} = \mathbf{P}_j^\top(\mathbf{f}_j - \mathbf{A}_j\mathbf{v}_j^{[1]})$ {restriction}
  $\mathbf{v}_{j-1}^{[2]} = \mathbf{V}(\mathbf{A}_{0:j-1}, \mathbf{M}_{1:j-1}, \mathbf{N}_{1:j-1}, \mathbf{P}_{1:j-1}, \mathbf{f}_{j-1}, \mathbf{0}, j-1)$
  $\mathbf{v}_j^{[3]} = \mathbf{v}_j^{[1]} + \mathbf{P}_j\mathbf{v}_{j-1}^{[2]}$ {coarse grid correction}
  $\mathbf{v}_j^{[4]} = \mathbf{v}_j^{[3]} + \mathbf{N}_j(\mathbf{f}_j - \mathbf{A}_j\mathbf{v}_j^{[3]})$ {post-smoothing}
  return $\mathbf{v}_j^{[4]}$
else
  return (approximate) solution of the problem $\mathbf{A}_0\mathbf{v}_0 = \mathbf{f}_0$
end if
```

1.2.1 Motivating experiments

We illustrate the relevance of the forthcoming analysis with numerical experiments, which demonstrate how the choice of the accuracy of the coarsest-level solve affects the convergence of the V-cycle method.

We consider a second order elliptic PDE of the form

$$-\nabla \cdot (k(x)\nabla u) = f \quad \text{in } \Omega, \qquad u = 0 \quad \text{on } \partial\Omega,$$

where $f \equiv 1$ and $\Omega = (0,1) \times (0,1)$. We consider two variants of the problem based on the coefficient function $k : \Omega \to \mathbb{R}$, “Poisson” with $k \equiv 1$ and “jump-1024” with

$$k(x) = \begin{cases} 1024, & x \in \left(0, \frac{1}{2}\right) \times \left(0, \frac{1}{2}\right) \cup \left(\frac{1}{2}, 1\right) \times \left(\frac{1}{2}, 1\right), \\ 1, & x \in \left(0, \frac{1}{2}\right) \times \left(\frac{1}{2}, 1\right) \cup \left(\frac{1}{2}, 1\right) \times \left(0, \frac{1}{2}\right). \end{cases}$$

The problems are discretized using the Galerkin finite element (FE) method with continuous piecewise affine functions on a hierarchy of nested triangulations obtained from the initial triangulation by uniform refinement. The triangulations are aligned with the line segments where the jumps in the coefficients take place.

We consider a geometric multigrid V-cycle method with 6 levels to solve the discrete problems on the finest level. We generate the sequence of stiffness matrices $\mathbf{A}_{0:J}$, by discretizing the problems on each level of the hierarchy. The sizes of the stiffness matrices are the same for both the Poisson and the jump-1024 problems. The size of the finest-level problems is $1.64 \cdot 10^6$ degrees of freedom (DoF). The size of the coarsest level problems is 1521 DoF. We use the standard prolongation matrices associated with the finite element spaces. The restriction matrices are transposes of the prolongation matrices.

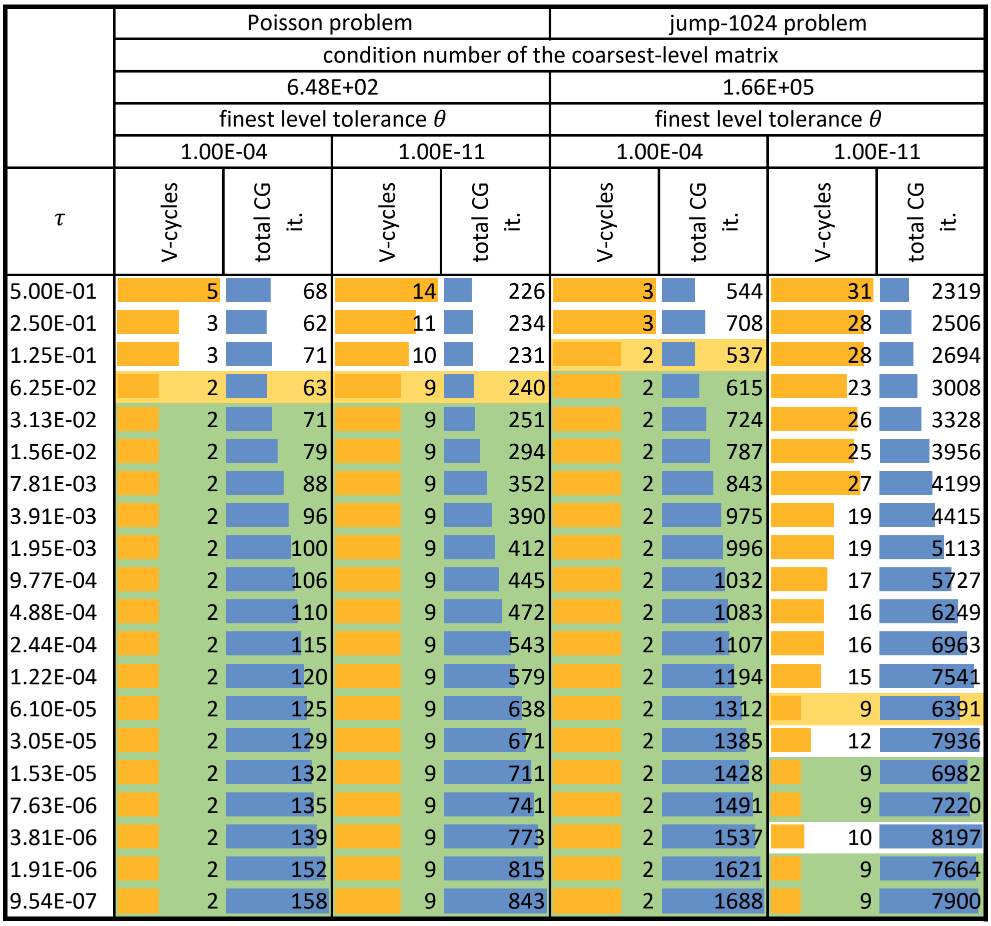
The stiffness and prolongation matrices are generated in the FE software FEniCS (version 2019.1.0) [2, 13]. In FEniCS the stiffness matrix is assembled using all nodes of the mesh. The homogeneous Dirichlet boundary condition is then applied by setting to zero all non-diagonal elements in rows and columns which correspond to nodes on the boundary and setting to zero the corresponding elements in the right-hand side vector. We modify the stiffness matrices, the prolongation matrices and the right-hand side vector so that the Galerkin condition (1.1) is satisfied. The computation is done in MATLAB 2023a. The codes for all experiments presented in this paper can be found at `https://doi.org/10.5281/zenodo.11178544`.

Pre-smoothing and post-smoothing in the V-cycle method are each accomplished via one iteration of the symmetric Gauss-Seidel method. We consider the symmetric Gauss-Seidel smoother in the experiments in this paper since we are able to numerically approximate the convergence rate of the exV-cycle method in the **A**-norm in this setting; see the discussion Section 1.6.1 and Appendix 1.8.1. The theoretical results stated in the paper, however, does not assume symmetry of the smoothing operators.

We consider two variants of the coarsest-level solver: the MATLAB backslash operator and the conjugate gradient method (CG) [11]. CG is stopped using a relative residual stopping criterion; i.e., for a chosen tolerance τ it is stopped when $\|\mathbf{f}_0 - \mathbf{A}_0\mathbf{v}_{0,\text{in}}\|/\|\mathbf{f}_0\| \leq \tau$. We consider various choices of the tolerance $\tau = 2^{-i}$, $i = 1, \ldots, 20$.

We run the V-cycle methods starting with a zero initial approximation and stop when the **A**-norm of the error is (approximately) lower than a tolerance θ, i.e., $\|\mathbf{x} - \mathbf{x}_{\text{in}}^{(n)}\|_{\mathbf{A}} \leq \theta$. We consider two choices of the tolerance $\theta = 10^{-4}$ and $\theta = 10^{-11}$. To approximate the **A**-norm of the error on the finest level, we compute the solution using the MATLAB backslash operator.

For both problems the variant with MATLAB backslash operator as the coarsest-level solver requires 2 and 9 V-cycle iterations to reach the desired finest-level accuracy 10^{-4} and 10^{-11}, respectively. The results of the variants with CG as the coarsest-level solver are summarized in Figure 1.1.



	Poisson problem				jump-1024 problem			
	condition number of the coarsest-level matrix							
	6.48E+02				1.66E+05			
	finest level tolerance θ				finest level tolerance θ			
	1.00E-04		1.00E-11		1.00E-04		1.00E-11	
τ	V-cycles	total CG it.	V-cycles	total CG it.	V-cycles	total CG it.	V-cycles	total CG it.
5.00E-01	5	68	14	226	3	544	31	2319
2.50E-01	3	62	11	234	3	708	28	2506
1.25E-01	3	71	10	231	2	537	28	2694
6.25E-02	2	63	9	240	2	615	23	3008
3.13E-02	2	71	9	251	2	724	26	3328
1.56E-02	2	79	9	294	2	787	25	3956
7.81E-03	2	88	9	352	2	843	27	4199
3.91E-03	2	96	9	390	2	975	19	4415
1.95E-03	2	100	9	412	2	996	19	5113
9.77E-04	2	106	9	445	2	1032	17	5727
4.88E-04	2	110	9	472	2	1083	16	6249
2.44E-04	2	115	9	543	2	1107	16	6963
1.22E-04	2	120	9	579	2	1194	15	7541
6.10E-05	2	125	9	638	2	1312	9	6391
3.05E-05	2	129	9	671	2	1385	12	7936
1.53E-05	2	132	9	711	2	1428	9	6982
7.63E-06	2	135	9	741	2	1491	9	7220
3.81E-06	2	139	9	773	2	1537	10	8197
1.91E-06	2	152	9	815	2	1621	9	7664
9.54E-07	2	158	9	843	2	1688	9	7900

Figure 1.1 Comparison of inV-cycle methods with CG as the coarsest-level solver with various choices of relative residual tolerance τ. The bright yellow and green color highlight variants that converge in the same number of V-cycles as the variant with MATLAB backslash operator on the coarsest-level. The bright yellow variants achieve this in the least total number of CG iterations on the coarsest-level.

Let us first focus on the results for the Poisson problem and finest-level tolerance $\theta = 10^{-4}$. The variants with CG with high coarsest-level tolerances $\tau = 2^{-i}$ $(i = 1, 2, 3)$ converge in a higher number of V-cycles than the variant with MATLAB backslash operator. The stricter the tolerance τ is the smaller the delay. The variants with tolerances $\tau = 6.25 \cdot 10^{-2}$ and smaller converge in the same number of V-cycles as the method with MATLAB backslash. The variant with tolerance $\tau = 6.25 \cdot 10^{-2}$ achieves this in the least total number of CG iterations on the coarsest level; this variant is in the figure highlighted by a bright yellow color. Using stricter tolerance than $\tau = 6.25 \cdot 10^{-2}$ is in this setting not beneficial since it does not yield a lower number of V-cycles but it requires more computational work on the coarsest level. We see analogous behavior for the Poisson problem and finest-level tolerance $\theta = 10^{-11}$. The bright yellow highlighted variant has the same coarsest-level tolerance.

Let us now focus on the results for the jump-1024 problem. The coarsest-level problem used when solving the jump-1024 problem has higher condition number than the one used for solving the Poisson problem. The total number of coarsest-level CG iterations is for all variants significantly higher than for the corresponding variants for the Poisson problem. We again see that the variants with high tolerances converge in a higher number of V-cycles than the variants with MATLAB backslash operator and that this delay becomes smaller for a lower coarsest-level tolerances and eventually vanishes if the tolerance is sufficiently small. It, however, does not strictly hold that lowering the tolerance results in faster converge. This can be seen for example when comparing the variants with tolerance $\tau = 3.05 \cdot 10^{-5}$ and $\tau = 6.10 \cdot 10^{-5}$ in the setting with $\theta = 10^{-11}$. In contrast to the methods for the Poisson problem (where the values of the tolerance of the bright yellow highlighted variants are the same for the two different finest-level tolerances θ) in the setting with the jump-1024 problem these values changes significantly - in order to reach the higher finest-level accuracy in the same number of V-cycles as the variant with MATLAB backslash solver the coarsest-level tolerance has to be significantly lower.

These experiments demonstrate that the choice of coarsest-level solver accuracy can significantly affect the convergence behavior of the V-cycle method and the overall amount of work that has to be done. This relationship is not yet well understood. This leads us to pose the following questions, which drive the work in this paper.

1. Can we analytically describe how the accuracy of the solver on the coarsest level affects the convergence behavior of the V-cycle method?

2. Can we define coarsest-level stopping criteria that would yield a computed V-cycle approximation “close” to the V-cycle approximation which would be obtained by solving the coarsest-level problems exactly?

1.3 Convergence analysis of inV-cycle method

We start by stating a few results and assumptions on the convergence of the exV-cycle method. Let $\mathbf{x}_{\mathrm{ex}}^{\mathrm{new}}$ be an approximation computed by one iteration of the exV-cycle method starting with an approximation $\mathbf{x}^{\mathrm{prev}}$. The error of the approximation $\mathbf{x}_{\text{ex}}^{\text{new}}$ can be written as the error of the previous approximation $\mathbf{x}^{\text{prev}}$ times the error propagation matrix[1] $\mathbf{E}$, i.e.,

$$\mathbf{x} - \mathbf{x}_{\text{ex}}^{\text{new}} = \mathbf{E}(\mathbf{x} - \mathbf{x}^{\text{prev}}).$$

We assume that the error propagation matrix $\mathbf{E}$ corresponds to an operator which is a contraction with respect to the $\mathbf{A}$-norm, i.e., $\|\mathbf{E}\|_{\mathbf{A}} < 1$. Proofs of this property for geometric multigrid methods can be found, e.g., in [23], [25]. The contraction property implies that each iteration of the exV-cycle method reduces the $\mathbf{A}$-norm of the error by at least a factor $\|\mathbf{E}\|_{\mathbf{A}}$, i.e.,

$$\|\mathbf{x} - \mathbf{x}_{\text{ex}}^{\text{new}}\|_{\mathbf{A}} \leq \|\mathbf{E}\|_{\mathbf{A}} \|\mathbf{x} - \mathbf{x}^{\text{prev}}\|_{\mathbf{A}} \quad \forall \mathbf{x}^{\text{prev}}.$$

We remark that this is a worst-case scenario analysis. The actual rate of convergence depends on the right-hand side and the current approximation and cannot be accurately described by a one-number characteristic.

In contrast to the exV-cycle method, the error of the approximation computed after one iteration of the inV-cycle method might not be able to be written as an error propagation matrix times the previous error. This is due to the fact that we consider a general solver on the coarsest level, whose application might not be able to be expressed as a matrix times vector. To obtain insight into the convergence behavior of the inV-cycle method, we view it as a perturbation of the exV-cycle method.

Let $\mathbf{x}_{\text{in}}^{\text{new}}$ denote the approximation computed after one iteration of the inV-cycle method starting with $\mathbf{x}^{\text{prev}}$. The error of the inV-cycle approximation can be written as the error of the approximation $\mathbf{x}_{\text{ex}}^{\text{new}}$ computed after one iteration of the exV-cycle method starting with the same $\mathbf{x}^{\text{prev}}$ plus the difference of the two approximations, i.e.,

$$\mathbf{x} - \mathbf{x}_{\text{in}}^{\text{new}} = \mathbf{x} - \mathbf{x}_{\text{ex}}^{\text{new}} + \mathbf{x}_{\text{ex}}^{\text{new}} - \mathbf{x}_{\text{in}}^{\text{new}} = \mathbf{E}(\mathbf{x} - \mathbf{x}^{\text{prev}}) + \mathbf{x}_{\text{ex}}^{\text{new}} - \mathbf{x}_{\text{in}}^{\text{new}}. \tag{1.3}$$

Taking $\mathbf{A}$-norms on the left and right sides, using the triangle inequality and the norm of $\mathbf{E}$ yields

$$\|\mathbf{x} - \mathbf{x}_{\text{in}}^{\text{new}}\|_{\mathbf{A}} \leq \|\mathbf{E}\|_{\mathbf{A}} \|\mathbf{x} - \mathbf{x}^{\text{prev}}\|_{\mathbf{A}} + \|\mathbf{x}_{\text{ex}}^{\text{new}} - \mathbf{x}_{\text{in}}^{\text{new}}\|_{\mathbf{A}}. \tag{1.4}$$

We turn our focus to the difference $\mathbf{x}_{\text{ex}}^{\text{new}} - \mathbf{x}_{\text{in}}^{\text{new}}$. When applying one step of the inV-cycle method or one step of the exV-cycle method, all intermediate results $\mathbf{v}_j^{[1]}$, $j = 1, \ldots, J$, $\mathbf{f}_j$, $j = 0, \ldots, J$ are the same until the coarsest level is reached. In the exV-cycle method, the exact solution $\mathbf{v}_0$ of the problem on the coarsest level is used, while in the inV-cycle method its computed approximation $\mathbf{v}_{0,\text{in}}$ is used. Writing down the difference $\mathbf{x}_{\text{ex}}^{\text{new}} - \mathbf{x}_{\text{in}}^{\text{new}}$ using the individual steps in Algorithm 1.1 yields (the subscripts "ex" and "in" indicate that the term corresponds to the

[1] The error propagation matrix for a two-level exV-cycle method can be expressed as

$$\mathbf{E} = (\mathbf{I}_1 - \mathbf{N}_1\mathbf{A}_1)(\mathbf{I}_1 - \mathbf{P}_1\mathbf{A}_0^{-1}\mathbf{P}_1^\top\mathbf{A}_1)(\mathbf{I}_1 - \mathbf{M}_1\mathbf{A}_1).$$

A recursive expression for the error propagation matrix for an exV-cycle method with a higher number of levels can be found, e.g., in [21, Theorem 2.4.1].

exV-cycle method and the inV-cycle method, respectively)

$$\begin{aligned}
\mathbf{x}_{\text{ex}}^{\text{new}} - \mathbf{x}_{\text{in}}^{\text{new}} &= \mathbf{v}_{J,\text{ex}}^{[4]} - \mathbf{v}_{J,\text{in}}^{[4]} \\
&= \mathbf{v}_{J,\text{ex}}^{[3]} + \mathbf{N}_J(\mathbf{f}_J - \mathbf{A}_J\mathbf{v}_{J,\text{ex}}^{[3]}) - (\mathbf{v}_{J,\text{in}}^{[3]} + \mathbf{N}_J(\mathbf{f}_J - \mathbf{A}_J\mathbf{v}_{J,\text{in}}^{[3]})) \\
&= (\mathbf{I}_J - \mathbf{N}_J\mathbf{A}_J)(\mathbf{v}_{J,\text{ex}}^{[3]} - \mathbf{v}_{J,\text{in}}^{[3]}) \\
&= (\mathbf{I}_J - \mathbf{N}_J\mathbf{A}_J)(\mathbf{v}_J^{[1]} + \mathbf{P}_J\mathbf{v}_{J-1,\text{ex}}^{[2]} - (\mathbf{v}_J^{[1]} + \mathbf{P}_J\mathbf{v}_{J-1,\text{in}}^{[2]})) \\
&= (\mathbf{I}_J - \mathbf{N}_J\mathbf{A}_J)\mathbf{P}_J(\mathbf{v}_{J-1,\text{ex}}^{[2]} - \mathbf{v}_{J-1,\text{in}}^{[2]}) \\
&= (\mathbf{I}_J - \mathbf{N}_J\mathbf{A}_J)\mathbf{P}_J(\mathbf{v}_{J-1,\text{ex}}^{[4]} - \mathbf{v}_{J-1,\text{in}}^{[4]}) \\
&= (\mathbf{I}_J - \mathbf{N}_J\mathbf{A}_J)\mathbf{P}_J \dots (\mathbf{I}_1 - \mathbf{N}_1\mathbf{A}_1)\mathbf{P}_1(\mathbf{v}_0 - \mathbf{v}_{0,\text{in}}).
\end{aligned}$$

Denoting by $\mathbf{S}$ the matrix

$$\mathbf{S} = (\mathbf{I}_J - \mathbf{N}_J\mathbf{A}_J)\mathbf{P}_J \dots (\mathbf{I}_1 - \mathbf{N}_1\mathbf{A}_1)\mathbf{P}_1 \in \mathbb{R}^{n_J \times n_0} \tag{1.5}$$

gives

$$\mathbf{x}_{\text{ex}}^{\text{new}} - \mathbf{x}_{\text{in}}^{\text{new}} = \mathbf{S}(\mathbf{v}_0 - \mathbf{v}_{0,\text{in}}). \tag{1.6}$$

We have expressed the difference of the inV-cycle and exV-cycle approximation as a matrix $\mathbf{S}$ times the error of the coarsest-level solver. The matrix $\mathbf{S}$ describes how the error is propagated to the finest level. Let $\|\mathbf{S}\|_{\mathbf{A}_0,\mathbf{A}}$ denote the norm of $\mathbf{S}$ generated by the vector norms $\|\cdot\|_{\mathbf{A}_0}$ and $\|\cdot\|_{\mathbf{A}}$, i.e.,

$$\|\mathbf{S}\|_{\mathbf{A}_0,\mathbf{A}} = \max_{\mathbf{v}\in\mathbb{R}^{n_0},\mathbf{v}\neq\mathbf{0}} \frac{\|\mathbf{S}\mathbf{v}\|_{\mathbf{A}}}{\|\mathbf{v}\|_{\mathbf{A}_0}}. \tag{1.7}$$

We derive a bound on the norm $\|\mathbf{S}\|_{\mathbf{A}_0,\mathbf{A}}$. Denoting by $\mathbf{S}_j$, $j = 2,\dots,J-1$, the matrix

$$\mathbf{S}_j = (\mathbf{I}_j - \mathbf{N}_j\mathbf{A}_j)\mathbf{P}_j \dots (\mathbf{I}_1 - \mathbf{N}_1\mathbf{A}_1)\mathbf{P}_1 \in \mathbb{R}^{n_j \times n_0},$$

and using the definition of $\|\mathbf{S}\|_{\mathbf{A}_0,\mathbf{A}}$ leads to

$$\begin{aligned}
\|\mathbf{S}\|_{\mathbf{A}_0,\mathbf{A}} &= \max_{\mathbf{v}\in\mathbb{R}^{n_0},\mathbf{v}\neq\mathbf{0}} \frac{\|(\mathbf{I}_J - \mathbf{N}_J\mathbf{A}_J)\mathbf{P}_J\mathbf{S}_{J-1}\mathbf{v}\|_{\mathbf{A}}}{\|\mathbf{v}\|_{\mathbf{A}_0}} \\
&= \max_{\mathbf{v}\in\mathbb{R}^{n_0},\mathbf{v}\neq\mathbf{0}} \frac{\|(\mathbf{I}_J - \mathbf{N}_J\mathbf{A}_J)\mathbf{P}_J\mathbf{S}_{J-1}\mathbf{v}\|_{\mathbf{A}}}{\|\mathbf{P}_J\mathbf{S}_{J-1}\mathbf{v}\|_{\mathbf{A}}} \frac{\|\mathbf{P}_J\mathbf{S}_{J-1}\mathbf{v}\|_{\mathbf{A}}}{\|\mathbf{v}\|_{\mathbf{A}_0}} \\
&\leq \max_{\mathbf{v}\in\mathbb{R}^{n_0},\mathbf{v}\neq\mathbf{0}} \|\mathbf{I}_J - \mathbf{N}_J\mathbf{A}_J\|_{\mathbf{A}} \frac{\|\mathbf{P}_J\mathbf{S}_{J-1}\mathbf{v}\|_{\mathbf{A}}}{\|\mathbf{v}\|_{\mathbf{A}_0}} \\
&= \|\mathbf{I}_J - \mathbf{N}_J\mathbf{A}_J\|_{\mathbf{A}} \max_{\mathbf{v}\in\mathbb{R}^{n_0},\mathbf{v}\neq\mathbf{0}} \frac{\|\mathbf{S}_{J-1}\mathbf{v}\|_{\mathbf{A}_{J-1}}}{\|\mathbf{v}\|_{\mathbf{A}_0}} \qquad (1.8) \\
&\leq \prod_{j=1}^{J} \|\mathbf{I}_j - \mathbf{N}_j\mathbf{A}_j\|_{\mathbf{A}_j} \max_{\mathbf{v}\in\mathbb{R}^{n_0},\mathbf{v}\neq\mathbf{0}} \frac{\|\mathbf{I}_0\mathbf{v}\|_{\mathbf{A}_0}}{\|\mathbf{v}\|_{\mathbf{A}_0}} \\
&= \prod_{j=1}^{J} \|\mathbf{I}_j - \mathbf{N}_j\mathbf{A}_j\|_{\mathbf{A}_j},
\end{aligned}$$

where we have used the Galerkin condition (1.1) to obtain (1.8). The monotone convergence of the post-smoothers (1.2) in the $\mathbf{A}_j$-norms implies that $\|\mathbf{S}\|_{\mathbf{A}_0,\mathbf{A}} < 1$. If post-smoothing is not used, i.e., $\mathbf{N}_j = \mathbf{0}$, then $\|\mathbf{S}\|_{\mathbf{A}_0,\mathbf{A}} = 1$.

The relation (1.6) implies

$$\|\mathbf{x}_{\text{ex}}^{\text{new}} - \mathbf{x}_{\text{in}}^{\text{new}}\|_{\mathbf{A}} \leq \|\mathbf{S}\|_{\mathbf{A}_0,\mathbf{A}} \|\mathbf{v}_0 - \mathbf{v}_{0,\text{in}}\|_{\mathbf{A}_0}. \tag{1.9}$$

Returning back to the estimate of the **A**-norm of the error of the inV-cycle approximation, using (1.4) and (1.9) we have

$$\|\mathbf{x} - \mathbf{x}_{\text{in}}^{\text{new}}\|_{\mathbf{A}} \leq \|\mathbf{E}\|_{\mathbf{A}} \|\mathbf{x} - \mathbf{x}^{\text{prev}}\|_{\mathbf{A}} + \|\mathbf{S}\|_{\mathbf{A}_0,\mathbf{A}} \|\mathbf{v}_0 - \mathbf{v}_{0,\text{in}}\|_{\mathbf{A}_0}, \quad \forall \mathbf{x}^{\text{prev}}. \tag{1.10}$$

We consider two different assumptions on the $\mathbf{A}_0$-norm of the error of the approximate coarsest-level solver $\|\mathbf{v}_0 - \mathbf{v}_{0,\text{in}}\|_{\mathbf{A}_0}$:

- A *relative* assumption, where the $\mathbf{A}_0$-norm of the error of the coarsest-level solver is less than a factor of the **A**-norm of the error of the previous approximation on the finest level, i.e., there is a constant $\gamma > 0$ such that

$$\|\mathbf{v}_0 - \mathbf{v}_{0,\text{in}}\|_{\mathbf{A}_0} \leq \gamma \|\mathbf{x} - \mathbf{x}^{\text{prev}}\|_{\mathbf{A}}, \quad \forall \mathbf{x}^{\text{prev}}. \tag{1.11}$$

- An *absolute* assumption, where the $\mathbf{A}_0$-norm of the error of the coarsest-level solver is less than a constant, i.e., there is a constant $\epsilon > 0$ such that

$$\|\mathbf{v}_0 - \mathbf{v}_{0,\text{in}}\|_{\mathbf{A}_0} \leq \epsilon, \quad \forall \mathbf{x}^{\text{prev}}. \tag{1.12}$$

We first analyze the inV-cycle method under the relative assumption and then under the absolute assumption. We comment on verification of the assumptions later in Sections 1.4 and 1.5.

1.3.1 Relative coarsest-level accuracy

Combining (1.9) and (1.11) yields an estimate on the **A**-norm of the relative difference of the exV-cycle and inV-cycle approximations after one V-cycle iteration

$$\frac{\|\mathbf{x}_{\text{ex}}^{\text{new}} - \mathbf{x}_{\text{in}}^{\text{new}}\|_{\mathbf{A}}}{\|\mathbf{x} - \mathbf{x}^{\text{prev}}\|_{\mathbf{A}}} \leq \|\mathbf{S}\|_{\mathbf{A}_0,\mathbf{A}} \gamma.$$

For the **A**-norm of the error of the inV-cycle approximation, we have using (1.10) and (1.11)

$$\|\mathbf{x} - \mathbf{x}_{\text{in}}^{\text{new}}\|_{\mathbf{A}} \leq \left(\|\mathbf{E}\|_{\mathbf{A}} + \|\mathbf{S}\|_{\mathbf{A}_0,\mathbf{A}} \gamma\right) \|\mathbf{x} - \mathbf{x}^{\text{prev}}\|_{\mathbf{A}}. \tag{1.13}$$

Assuming that the error of the coarsest-level solver satisfies estimate (1.11) with γ such that

$$\|\mathbf{E}\|_{\mathbf{A}} + \|\mathbf{S}\|_{\mathbf{A}_0,\mathbf{A}} \gamma < 1,$$

the inV-cycle method converges and we have a bound on its convergence rate in terms of the bound on the rate of convergence of the exV-cycle method and $\|\mathbf{S}\|_{\mathbf{A}_0,\mathbf{A}} \gamma$.

We summarize the results in the following theorem.

Theorem 1.1. *Let* $\mathbf{x}_{\text{ex}}^{\text{new}}$ *be the approximation of* $\mathbf{x} = \mathbf{A}^{-1}\mathbf{b}$ *computed after one iteration of the exV-cycle method with error propagation matrix* $\mathbf{E}$, $\|\mathbf{E}\|_{\mathbf{A}} < 1$, *starting with an approximation* $\mathbf{x}^{\text{prev}}$. *Let* $\mathbf{x}_{\text{in}}^{\text{new}}$ *be an approximation of* $\mathbf{x} = \mathbf{A}^{-1}\mathbf{b}$ *computed after one iteration of the inV-cycle method starting with the same* *approximation* $\mathbf{x}^{\text{prev}}$*, and assume the error of the coarsest-level solver* $\mathbf{v}_0 - \mathbf{v}_{0,\text{in}}$ *satisfies*

$$\|\mathbf{v}_0 - \mathbf{v}_{0,\text{in}}\|_{\mathbf{A}_0} \leq \gamma \|\mathbf{x} - \mathbf{x}^{\text{prev}}\|_{\mathbf{A}}, \tag{1.14}$$

for some constant $\gamma > 0$*. Then the following estimate on the* $\mathbf{A}$*-norm of the relative difference of the exV-cycle and inV-cycle approximations after one V-cycle iteration holds:*

$$\frac{\|\mathbf{x}_{\text{ex}}^{\text{new}} - \mathbf{x}_{\text{in}}^{\text{new}}\|_{\mathbf{A}}}{\|\mathbf{x} - \mathbf{x}^{\text{prev}}\|_{\mathbf{A}}} \leq \|\mathbf{S}\|_{\mathbf{A}_0,\mathbf{A}} \gamma, \tag{1.15}$$

where $\mathbf{S}$ *is the matrix defined in* (1.5) *satisfying* $\|\mathbf{S}\|_{\mathbf{A}_0,\mathbf{A}} \leq 1$*. Moreover,*

$$\|\mathbf{x} - \mathbf{x}_{\text{in}}^{\text{new}}\|_{\mathbf{A}} \leq \left(\|\mathbf{E}\|_{\mathbf{A}} + \|\mathbf{S}\|_{\mathbf{A}_0,\mathbf{A}} \gamma\right) \|\mathbf{x} - \mathbf{x}^{\text{prev}}\|_{\mathbf{A}}, \tag{1.16}$$

and if the error of the coarsest-level solver satisfies (1.14) *with* γ *such that*

$$\|\mathbf{E}\|_{\mathbf{A}} + \|\mathbf{S}\|_{\mathbf{A}_0,\mathbf{A}} \gamma < 1,$$

the inV-cycle method converges.

A multigrid method is said to be uniformly convergent if there exist a bound on the rate of convergence which is independent of the number of levels and of the size of the problem on the coarsest level; see e.g., [23, 25]. If we assume that the exV-cycle method converges uniformly and the error of the coarsest-level solver in the inV-cycle method satisfies (1.14) with γ such that $\|\mathbf{E}\|_{\mathbf{A}} + \gamma < 1$ holds and γ is independent of the number of levels and the size of the problem on the coarsest level, inequality (1.16) and the fact that $\|\mathbf{S}\|_{\mathbf{A}_0,\mathbf{A}} < 1$ yield that the inV-cycle method converges uniformly.

We use the results presented in this section to discuss what may be the effect of the choice of tolerance in a relative residual coarsest-level stopping criterion on the convergence of the V-cycle method in Section 1.4. We present numerical experiments testing the accuracy of the estimates (1.15) and (1.16) in Section 1.6.1.

1.3.2 Absolute coarsest-level accuracy

We further focus on the analysis of the inV-cycle method under the assumption on the absolute coarsest-level accuracy (1.12). The following development is inspired by [7, Section 4], where the authors analyze the inexact Richarson method.

Let $\mathbf{x}_{\text{in}}^{(n)}$ be an approximation computed after n iterations of the inV-cycle method, starting with an initial approximation $\mathbf{x}^{(0)}$, and assume the errors of the coarsest-level solver satisfy (1.12) with a constant $\epsilon > 0$. Using (1.3) and (1.6), the error of the kth approximation $\mathbf{x}_{\text{in}}^{(k)}$, $k = 1, \ldots, n$, can be written as

$$\mathbf{x} - \mathbf{x}_{\text{in}}^{(k)} = \mathbf{E}(\mathbf{x} - \mathbf{x}_{\text{in}}^{(k-1)}) + \mathbf{g}^{(k)}, \quad k = 1, \ldots, n,$$

where $\mathbf{g}^{(k)} = \mathbf{S}(\mathbf{v}_0^{(k)} - \mathbf{v}_{0,\text{in}}^{(k)})$ and $\mathbf{v}_0^{(k)} - \mathbf{v}_{0,\text{in}}^{(k)}$ is the error of the coarsest-level solver when computing $\mathbf{x}_{\text{in}}^{(k)}$. Let $\mathbf{x}_{\text{ex}}^{(n)}$ be an approximation computed after n iterations of the exV-cycle method starting with the same initial approximation $\mathbf{x}^{(0)}$. The difference $\mathbf{x}_{\mathrm{ex}}^{(n)} - \mathbf{x}_{\mathrm{in}}^{(n)}$ can be rewritten using the terms $\mathbf{g}^{(k)}$ as

$$\begin{aligned}
\mathbf{x}_{\mathrm{ex}}^{(n)} - \mathbf{x}_{\mathrm{in}}^{(n)} &= (\mathbf{x} - \mathbf{x}_{\mathrm{in}}^{(n)}) - (\mathbf{x} - \mathbf{x}_{\mathrm{ex}}^{(n)}) \\
&= \mathbf{E}(\mathbf{x} - \mathbf{x}_{\mathrm{in}}^{(n-1)}) + \mathbf{g}^{(n)} - \mathbf{E}^n(\mathbf{x} - \mathbf{x}^{(0)}) \\
&= \mathbf{E}(\mathbf{E}(\mathbf{x} - \mathbf{x}_{\mathrm{in}}^{(n-2)}) + \mathbf{g}^{(n-1)}) + \mathbf{g}^{(n)} - \mathbf{E}^n(\mathbf{x} - \mathbf{x}^{(0)}) \\
&= \mathbf{E}^2(\mathbf{x} - \mathbf{x}_{\mathrm{in}}^{(n-2)}) + \mathbf{E}\mathbf{g}^{(n-1)} + \mathbf{g}^{(n)} - \mathbf{E}^n(\mathbf{x} - \mathbf{x}^{(0)}) \\
&= \mathbf{E}^n(\mathbf{x} - \mathbf{x}^{(0)}) + \sum_{k=1}^{n} \mathbf{E}^{n-k}\mathbf{g}^{(k)} - \mathbf{E}^n(\mathbf{x} - \mathbf{x}^{(0)}) \\
&= \sum_{k=1}^{n} \mathbf{E}^{n-k}\mathbf{g}^{(k)}.
\end{aligned}$$

Taking the $\mathbf{A}$-norm of both sides, using the triangle inequality and the multiplicativity of the matrix norm $\|\cdot\|_{\mathbf{A}}$ we obtain

$$\|\mathbf{x}_{\mathrm{ex}}^{(n)} - \mathbf{x}_{\mathrm{in}}^{(n)}\|_{\mathbf{A}} = \|\sum_{k=1}^{n} \mathbf{E}^{n-k}\mathbf{g}^{(k)}\|_{\mathbf{A}} \leq \sum_{k=1}^{n} \|\mathbf{E}\|_{\mathbf{A}}^{n-k}\|\mathbf{g}^{(k)}\|_{\mathbf{A}}. \tag{1.17}$$

Using that $\mathbf{g}^{(k)} = \mathbf{S}(\mathbf{v}_0^{(k)} - \mathbf{v}_{0,\mathrm{in}}^{(k)})$ and the norm of $\mathbf{S}$ (1.7) leads to

$$\begin{aligned}
\|\mathbf{x}_{\mathrm{ex}}^{(n)} - \mathbf{x}_{\mathrm{in}}^{(n)}\|_{\mathbf{A}} &\leq \sum_{k=1}^{n} \|\mathbf{E}\|_{\mathbf{A}}^{n-k}\|\mathbf{S}(\mathbf{v}_0^{(k)} - \mathbf{v}_{0,\mathrm{in}}^{(k)})\|_{\mathbf{A}} \\
&\leq \sum_{k=1}^{n} \|\mathbf{E}\|_{\mathbf{A}}^{n-k}\|\mathbf{S}\|_{\mathbf{A}_0,\mathbf{A}}\|\mathbf{v}_0^{(k)} - \mathbf{v}_{0,\mathrm{in}}^{(k)}\|_{\mathbf{A}_0}.
\end{aligned}$$

This bound provides information on how the accuracy of the solver on the coarsest level during the individual solves affects the $\mathbf{A}$-norm of the difference of the approximations $\mathbf{x}_{\mathrm{ex}}^{(n)}$ and $\mathbf{x}_{\mathrm{in}}^{(n)}$.

Using the assumption (1.12) and the bound for a sum of a geometric series we have

$$\|\mathbf{x}_{\mathrm{ex}}^{(n)} - \mathbf{x}_{\mathrm{in}}^{(n)}\|_{\mathbf{A}} \leq \sum_{k=1}^{n} \|\mathbf{E}\|_{\mathbf{A}}^{n-k}\|\mathbf{S}\|_{\mathbf{A}_0,\mathbf{A}}\epsilon < \|\mathbf{S}\|_{\mathbf{A}_0,\mathbf{A}}\epsilon \sum_{\ell=0}^{+\infty} \|\mathbf{E}\|_{\mathbf{A}}^{\ell} \leq \frac{\epsilon\|\mathbf{S}\|_{\mathbf{A}_0,\mathbf{A}}}{1 - \|\mathbf{E}\|_{\mathbf{A}}}.$$

Using the triangle inequality yields

$$\|\mathbf{x} - \mathbf{x}_{\mathrm{in}}^{(n)}\|_{\mathbf{A}} \leq \|\mathbf{x} - \mathbf{x}_{\mathrm{ex}}^{(n)}\|_{\mathbf{A}} + \frac{\epsilon\|\mathbf{S}\|_{\mathbf{A}_0,\mathbf{A}}}{1 - \|\mathbf{E}\|_{\mathbf{A}}};$$

i.e., the $\mathbf{A}$-norm of the error after n V-cycle iterations is less than the $\mathbf{A}$-norm of the error of the exV-cycle approximation computed after n V-cycles plus the term $\frac{\epsilon\|\mathbf{S}\|_{\mathbf{A}_0,\mathbf{A}}}{1-\|\mathbf{E}\|_{\mathbf{A}}}$.

We summarize the results of this section in the following theorem.

Theorem 1.2. *Let $\mathbf{x}_{\mathrm{ex}}^{(n)}$ be the approximation of $\mathbf{x} = \mathbf{A}^{-1}\mathbf{b}$ computed after n iterations of the exV-cycle method with error propagation matrix $\mathbf{E}$, $\|\mathbf{E}\|_{\mathbf{A}} < 1$, starting with an approximation $\mathbf{x}^{(0)}$. Let $\mathbf{x}_{\mathrm{in}}^{(n)}$ be an approximation of $\mathbf{x} = \mathbf{A}^{-1}\mathbf{b}$ computed after n iterations of the inV-cycle method, starting with the same approximation, and assume the errors of the coarsest-level solver $\mathbf{v}_0^{(k)} - \mathbf{v}_{0,\mathrm{in}}^{(k)}$ satisfy*

$$\|\mathbf{v}_0^{(k)} - \mathbf{v}_{0,\mathrm{in}}^{(k)}\|_{\mathbf{A}_0} \leq \epsilon, \quad k = 1, \ldots, n, \tag{1.18}$$

for a constant $\epsilon > 0$*. Then the following estimate on the A-norm of the difference of* $\mathbf{x}_{\mathrm{ex}}^{(n)}$ *and* $\mathbf{x}_{\mathrm{in}}^{(n)}$ *holds:*

$$\|\mathbf{x}_{\mathrm{ex}}^{(n)} - \mathbf{x}_{\mathrm{in}}^{(n)}\|_{\mathbf{A}} \leq \frac{\epsilon \|\mathbf{S}\|_{\mathbf{A}_0,\mathbf{A}}}{1 - \|\mathbf{E}\|_{\mathbf{A}}}, \tag{1.19}$$

where $\mathbf{S}$ *is the matrix defined in* (1.5) *and* $\|\mathbf{S}\|_{\mathbf{A}_0,\mathbf{A}} \leq 1$*. Moreover,*

$$\|\mathbf{x} - \mathbf{x}_{\mathrm{in}}^{(n)}\|_{\mathbf{A}} \leq \|\mathbf{x} - \mathbf{x}_{\mathrm{ex}}^{(n)}\|_{\mathbf{A}} + \frac{\epsilon \|\mathbf{S}\|_{\mathbf{A}_0,\mathbf{A}}}{1 - \|\mathbf{E}\|_{\mathbf{A}}}.$$

We derive a coarsest-level stopping criteria based on these results in Section 1.5 and perform numerical experiments studying the behavior of an inV-cycle method with the assumption on an absolute coarsest-level accuracy in Section 1.6.3.

1.4 Effects of the choice of the tolerance in relative residual stopping criterion

Stopping an iterative coarsest-level solver based on the size of the relative residual is frequently done both in the literature and in practice. One chooses a tolerance τ and stops the solver when

$$\frac{\|\mathbf{f}_0 - \mathbf{A}_0\mathbf{v}_{0,\mathrm{in}}\|}{\|\mathbf{f}_0\|} \leq \tau. \tag{1.20}$$

In this section we use the results from Section 1.3.1 to analyze the effect of the choice of the tolerance τ on the convergence of the inV-cycle method. We show that if inequality (1.20) holds then inequality (1.14) holds with a certain γ depending on the tolerance τ, and consequently we may use the results from Theorem 1.1.

We start by showing that the Euclidean norm of the right-hand side on the coarsest level can be bounded by the Euclidean norm of the residual of the previous approximation on the finest level. Rewriting $\mathbf{f}_0$ using the individual steps in Algorithm 1.1, we have (note that $\mathbf{v}_j^{[0]} = \mathbf{0}$, $j = 1, \ldots, J-1$)

$$\begin{aligned} \mathbf{f}_0 &= \mathbf{P}_1^\top(\mathbf{f}_1 - \mathbf{A}_1\mathbf{v}_1^{[1]}) = \mathbf{P}_1^\top(\mathbf{f}_1 - \mathbf{A}_1(\mathbf{v}_1^{[0]} + \mathbf{M}_1(\mathbf{f}_1 - \mathbf{A}_1\mathbf{v}_1^{[0]})) \\ &= \mathbf{P}_1^\top(\mathbf{I}_1 - \mathbf{A}_1\mathbf{M}_1)\mathbf{f}_1 = \prod_{j=1}^{J-1} \mathbf{P}_j^\top(\mathbf{I}_j - \mathbf{A}_j\mathbf{M}_j)\mathbf{f}_{J-1}. \end{aligned} \tag{1.21}$$

The vector $\mathbf{f}_{J-1}$ can be expressed as

$$\begin{aligned} \mathbf{f}_{J-1} &= \mathbf{P}_J^\top(\mathbf{b} - \mathbf{A}\mathbf{v}_J^{[1]}) = \mathbf{P}_J^\top(\mathbf{b} - \mathbf{A}(\mathbf{x}^{\mathrm{prev}} + \mathbf{M}_J(\mathbf{b} - \mathbf{A}\mathbf{x}^{\mathrm{prev}}))) \\ &= \mathbf{P}_J^\top(\mathbf{I}_J - \mathbf{A}\mathbf{M}_J)(\mathbf{b} - \mathbf{A}\mathbf{x}^{\mathrm{prev}}). \end{aligned} \tag{1.22}$$

Denoting by $\mathbf{T}$ the matrix

$$\mathbf{T} = \prod_{j=1}^{J} \mathbf{P}_j^\top(\mathbf{I}_j - \mathbf{A}_j\mathbf{M}_j),$$

and combining (1.21) and (1.22), we have $\mathbf{f}_0 = \mathbf{T}\left(\mathbf{b} - \mathbf{A}\mathbf{x}^{\mathrm{prev}}\right)$. The matrix $\mathbf{T}$ describes how the residual from the finest level is propagated to the coarsest level. Based on this relation, we can estimate the Euclidean norm of $\mathbf{f}_0$ as

$$\|\mathbf{f}_0\| \leq \|\mathbf{T}\|\|\mathbf{b} - \mathbf{A}\mathbf{x}^{\mathrm{prev}}\|. \tag{1.23}$$

The norm of $\mathbf{T}$ can be bounded as

$$\|\mathbf{T}\| \leq \prod_{j=1}^{J} \|\mathbf{P}_j^{\top}\|\|\mathbf{I}_j - \mathbf{A}_j\mathbf{M}_j\|,$$

by a procedure analogous to that used in bounding the norm of $\|\mathbf{S}\|_{\mathbf{A}_0,\mathbf{A}}$; see Section 1.3.

Utilizing (1.23) to bound the term $\|\mathbf{f}_0\|$ in (1.20), we obtain

$$\frac{\|\mathbf{f}_0 - \mathbf{A}_0\mathbf{v}_{0,\mathrm{in}}\|}{\|\mathbf{T}\|\|\mathbf{b} - \mathbf{A}\mathbf{x}^{\mathrm{prev}}\|} \leq \tau.$$

Using that the Euclidean norm of the coarsest-level residual can be bounded from below by the $\mathbf{A}_0$-norm of the coarsest-level error as (see Appendix 1.8.2)

$$\|\mathbf{A}_0^{-1}\|^{-\frac{1}{2}}\|\mathbf{v}_0 - \mathbf{v}_{0,\mathrm{in}}\|_{\mathbf{A}_0} \leq \|\mathbf{f}_0 - \mathbf{A}_0\mathbf{v}_{0,\mathrm{in}}\|, \tag{1.24}$$

and that the Euclidean norm of the finest-level residual can be bounded from above by $\mathbf{A}$-norm of the error as (see Appendix 1.8.2)

$$\|\mathbf{b} - \mathbf{A}\mathbf{x}^{\mathrm{prev}}\| \leq \|\mathbf{A}\|^{\frac{1}{2}}\|\mathbf{x} - \mathbf{x}^{\mathrm{prev}}\|_{\mathbf{A}}, \tag{1.25}$$

we get

$$\frac{\|\mathbf{A}_0^{-1}\|^{-\frac{1}{2}}\|\mathbf{v}_0 - \mathbf{v}_{0,\mathrm{in}}\|_{\mathbf{A}_0}}{\|\mathbf{T}\|\|\mathbf{A}\|^{\frac{1}{2}}\|\mathbf{x} - \mathbf{x}^{\mathrm{prev}}\|_{\mathbf{A}}} \leq \tau, \tag{1.26}$$

i.e., the inequality (1.14) holds with $\gamma = \tau\|\mathbf{T}\|\|\mathbf{A}\|^{\frac{1}{2}}\|\mathbf{A}_0^{-1}\|^{\frac{1}{2}}$. Using the results from Theorem 1.1, we have an answer to the question of how the choice of the tolerance in the relative residual stopping criterion for the coarsest-level solver affects the convergence of the V-cycle method.

We note that since (1.26) was derived using the estimates (1.24)-(1.25), which may be a large overestimate, the resulting estimates may be loose and the actual quantities much smaller. We carry out numerical experiments investigating the accuracy of the estimates for the methods used in the motivating numerical experiment in Section 1.6.2.

1.5 Absolute coarsest-level stopping criteria

In this section, we focus on the second question formulated after the motivational experiment; that is:

> “Can we define coarsest-level stopping criteria that would yield a computed V-cycle approximation “close” to the V-cycle approximation which would be obtained by solving the coarsest-level problems exactly?”

We present a new stopping criteria motivated by the assumption on an absolute accuracy of the coarsest-level solver and the results in Theorem 1.2. The inequality (1.18) in the assumption on an absolute accuracy of the coarsest-level solver can not be directly used in practice as a coarsest-level stopping criterion since it involves the $\mathbf{A}_0$-norm of the coarsest-level error, which is not available. We may, however, formulate coarsest-level stopping criteria using estimates of the $\mathbf{A}_0$-norm of the error. Let $\eta(\mathbf{v}_{0,\mathrm{in}}^{(k)})$ be an upper bound on the $\mathbf{A}_0$-norm of the error of the coarsest-level solver in the kth V-cycle iteration, i.e.,

$$\|\mathbf{v}_0^{(k)} - \mathbf{v}_{0,\mathrm{in}}^{(k)}\|_{\mathbf{A}_0} \leq \eta(\mathbf{v}_{0,\mathrm{in}}^{(k)}), \quad k = 1, \ldots, n. \tag{1.27}$$

We formulate a stopping criterion with a parameter $\epsilon > 0$, which is chosen by the user, as

$$\eta(\mathbf{v}_{0,\mathrm{in}}^{(k)}) \leq \epsilon, \quad k = 1, \ldots, n. \tag{1.28}$$

If (1.28) holds then (1.18) holds and from Theorem 1.2 we know that the $\mathbf{A}$-norm of the difference of the inV-cycle and exV-cycle approximations after n V-cycle iterations is bounded according to

$$\|\mathbf{x}_{\mathrm{ex}}^{(n)} - \mathbf{x}_{\mathrm{in}}^{(n)}\|_{\mathbf{A}} \leq \frac{\epsilon}{1 - \|\mathbf{E}\|_{\mathbf{A}}}; \tag{1.29}$$

here we have bounded $\|\mathbf{S}\|_{\mathbf{A}_0,\mathbf{A}}$ by one from above. We note that the accuracy of this estimate is influenced by the accuracy of the estimates (1.27). The term $\|\mathbf{E}\|_{\mathbf{A}}$ is in general unknown. It is, however, included here in the form $1/(1 - \|\mathbf{E}\|_{\mathbf{A}})$. If we assume that $\|\mathbf{E}\|_{\mathbf{A}} < \alpha$, (where, e.g., $\alpha = 1/2$ or $\alpha = 2/3$) we get

$$\|\mathbf{x}_{\mathrm{ex}}^{(n)} - \mathbf{x}_{\mathrm{in}}^{(n)}\|_{\mathbf{A}} \leq \frac{\epsilon}{1 - \alpha}. \tag{1.30}$$

Due to the structure of the term $1/(1-\|\mathbf{E}\|_{\mathbf{A}})$ this is not a significant overestimation even if the actual value of $\|\mathbf{E}\|_{\mathbf{A}}$ is much smaller than α. We note that assuming that $\|\mathbf{E}\|_{\mathbf{A}} < 1/2$ or $\|\mathbf{E}\|_{\mathbf{A}} < 2/3$ is a valid assumption for a well set up V-cycle methods.

The stopping criterion (1.28) thus enable us to control the difference of the inV-cycle and exV-cycle approximations after n V-cycles and consequently also the accuracy of the inV-cycle approximation. If we want to compute an inV-cycle approximation whose $\mathbf{A}$-norm of the error is approximately at the level θ (where e.g., $\theta = 10^{-4}$ or $\theta = 10^{-11}$) we may set ϵ as $\epsilon = (1 - \alpha)\theta$. Using the triangle inequality and (1.30) the $\mathbf{A}$-norm of the error of the inV-cycle approximation is bounded as

$$\|\mathbf{x} - \mathbf{x}_{\mathrm{in}}^{(n)}\|_{\mathbf{A}} \leq \|\mathbf{x} - \mathbf{x}_{\mathrm{ex}}^{(n)}\|_{\mathbf{A}} + \|\mathbf{x}_{\mathrm{ex}}^{(n)} - \mathbf{x}_{\mathrm{in}}^{(n)}\|_{\mathbf{A}} \leq \|\mathbf{x} - \mathbf{x}_{\mathrm{ex}}^{(n)}\|_{\mathbf{A}} + \theta.$$

If we perform sufficiently many V-cycle iterations such that the $\mathbf{A}$-norm of the exV-cycle approximation (i.e., $\|\mathbf{x} - \mathbf{x}_{\mathrm{ex}}^{(n)}\|_{\mathbf{A}}$) would be approximately at the level of θ, than the error of the inV-cycle approximation, $\|\mathbf{x} - \mathbf{x}_{\mathrm{in}}^{(n)}\|_{\mathbf{A}}$, is approximately at the level of θ.

The coarsest-level stopping criterion does not provide a finest-level stopping criterion for the inV-cycle method. We comment on a heuristic finest-level stopping indicator when discussing the results of numerical experiments in Sections 1.6.3 and 1.6.4.

We further comment on the choice of the estimate η on the $\mathbf{A}_0$-norm of the error on the coarsest-level. We may use the residual based estimate on the $\mathbf{A}_0$-norm of the error (1.24); i.e.,

$$\eta(\mathbf{v}_{0,\mathrm{in}}) = \|\mathbf{A}_0^{-1}\|^{-\frac{1}{2}}\|\mathbf{f}_0 - \mathbf{A}_0\mathbf{v}_{0,\mathrm{in}}\|. \tag{1.31}$$

The term $\|\mathbf{A}_0^{-1}\|$, i.e., the reciprocal value of the smallest eigenvalue of $\mathbf{A}_0$, has to be in practical computations estimated or computed approximately.

When we are using the conjugate gradient method or the preconditioned conjugate gradient method, we may use some of the upper bounds on the $\mathbf{A}_0$-norm of the error described e.g., in [9] and the references therein, as well as in [6, 17, 16, 18]. Most of these estimates are derived based on the interpretation of CG as a procedure for computing a Gauss quadrature approximation to a Riemann-Stieltjes integral.

We test the accuracy of estimate (1.30) and the performance of the stopping criterion in numerical experiments in Section 1.6.4.

1.6 Numerical experiments

In this section we present numerical experiments illustrating some of the key results derived in this paper. We consider the same model problems and analogous V-cycle methods as in the motivating experiments in Section 1.2.1. To approximate the errors on the finest and coarsest level we compute the solutions using the MATLAB backslash operator. We simulate the exV-cycle method by using MATLAB backslash operator as the solver on the coarsest level.

1.6.1 inV-cycle method satisfying the relative coarsest-level accuracy assumption

In this experiment, we study the behavior of the inV-cycle method with a coarsest-level solver which is stopped when the assumption on a relative coarsest-level accuracy is satisfied and examine the accuracy of the estimates presented in Theorem 1.1.

We consider the same problems and analogous V-cycle methods as in the motivational experiments in Section 1.2.1. The only difference is that we stop CG on the coarsest level when inequality (1.14) (approximately) holds, i.e., when

$$\|\mathbf{v}_0 - \mathbf{v}_{0,\mathrm{in}}\|_{\mathbf{A}_0} \leq \gamma\|\mathbf{x} - \mathbf{x}^{\mathrm{prev}}\|_{\mathbf{A}}.$$

We consider three choices of the constant γ, $\gamma = 0.3$, $\gamma = 10^{-3}$, and $\gamma = 10^{-4}$. We run the V-cycle method starting with a zero initial approximate solution and stop when the $\mathbf{A}$-norm of the error on the finest-level is (approximately) lower than 10^{-11}.

The results are summarized in Figure 1.2. After each V-cycle iteration we compute the $\mathbf{A}$-norms of the relative difference of the exV-cycle and inV-cycle approximations after one V-cycle iteration, i.e.,

$$\frac{\|\mathbf{x}_{\mathrm{ex}}^{\mathrm{new}} - \mathbf{x}_{\mathrm{in}}^{\mathrm{new}}\|_{\mathbf{A}}}{\|\mathbf{x} - \mathbf{x}^{\mathrm{prev}}\|_{\mathbf{A}}}, \tag{1.32}$$

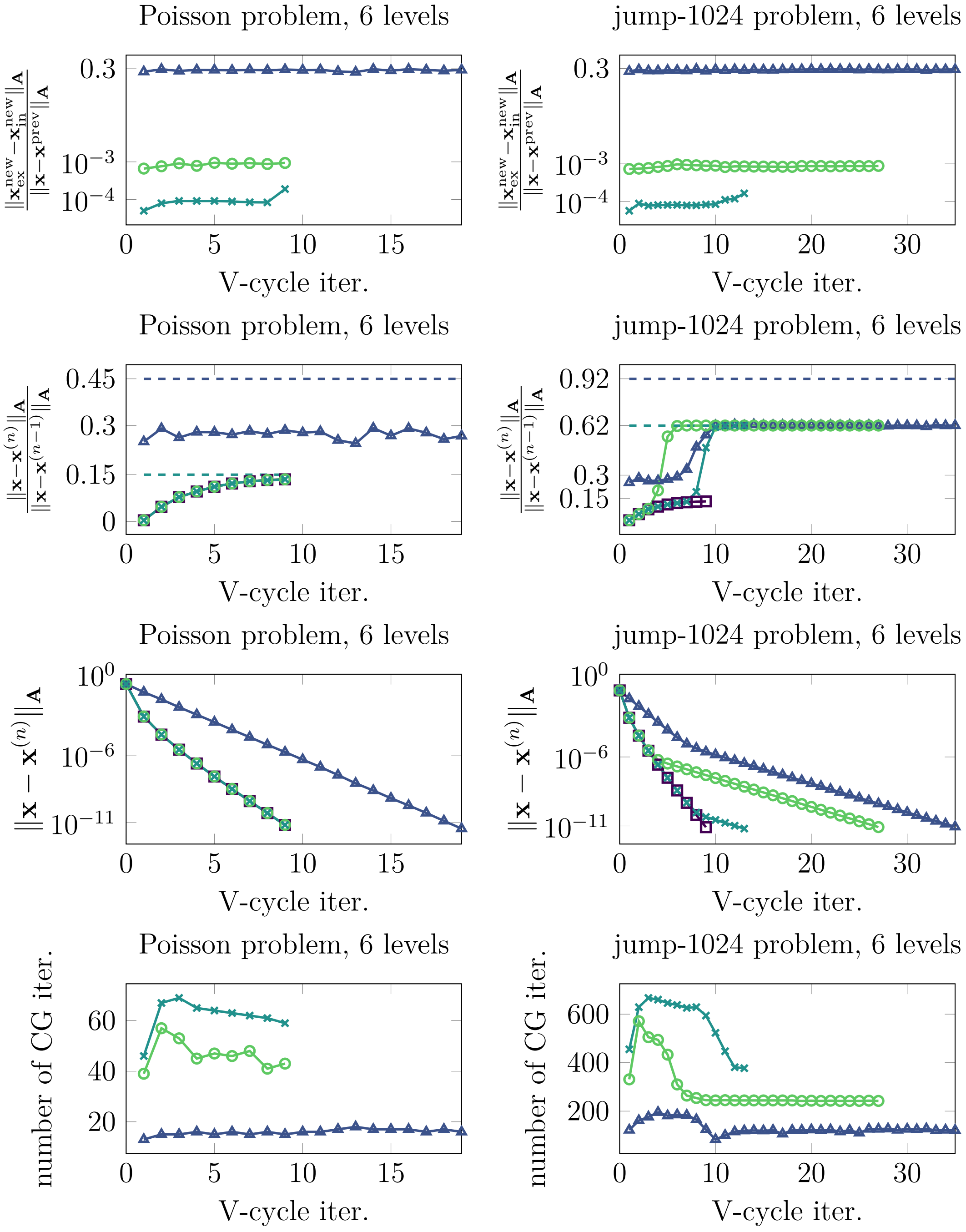


Figure 1.2 Properties of inV-cycle methods with CG as the solver on the coarsest level, which is stopped when the assumption on the relative coarsest-level accuracy (1.14) is satisfied with $\gamma = 0.3$ (—▲—), $\gamma = 10^{-3}$ (—○—), or $\gamma = 10^{-4}$ (—×—). The dashed lines corresponds to the estimates $\|\mathbf{E}\|_{\mathbf{A}} + \gamma$. For comparison we also include results of the exV-cycle method (—□—).

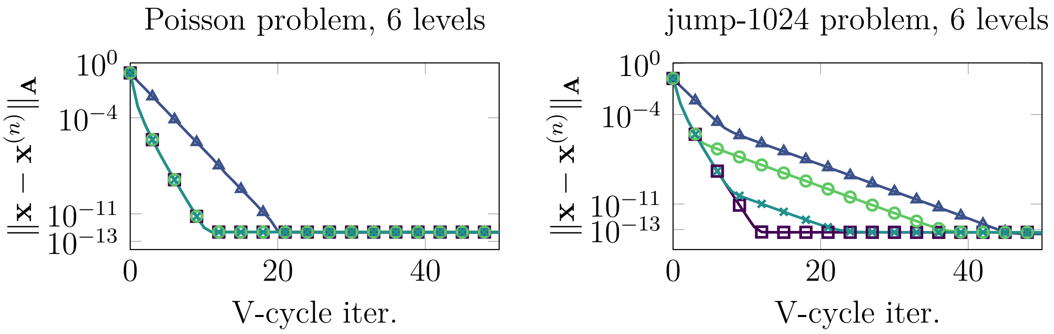


Figure 1.3 **A**-norm of the error of the inV-cycle methods with CG as the solver on the coarsest level, which is stopped when the assumption on the relative coarsest-level accuracy (1.14) is satisfied with $\gamma = 0.3$ (—▲—), $\gamma = 10^{-3}$ (—○—), or $\gamma = 10^{-4}$ (—×—). For comparison we also include the **A**-norm of the error of the exV-cycle method (—□—). Every third point is marked.

for $\mathbf{x}^{\text{prev}} = \mathbf{x}_{\text{in}}^{(k)}$, $k = 0, 1, \ldots$. According to the estimate (1.15) from Theorem 1.1, the relative difference (1.32) should be less than $\gamma\|\mathbf{S}\|_{\mathbf{A}_0,\mathbf{A}}$, where $\|\mathbf{S}\|_{\mathbf{A}_0,\mathbf{A}} \leq 1$. Looking at the results we see that all values (1.32) are slightly less than γ besides the ones computed after the last few V-cycle iterations of the variants with $\gamma = 10^{-4}$. We strongly believe that these outlier are caused by the effects of finite precision arithmetic. Dividing the computed values (1.32) (besides the mentioned outliers) by γ and finding the maximum we get a lower bound on $\|\mathbf{S}\|_{\mathbf{A}_0,\mathbf{A}}$, which is 0.95 and 0.97 for the variant with the Poisson and the jump-1024 problem, respectively.

We also compute the convergence rate in the **A**-norm, after each V-cycle iteration, i.e.,

$$\frac{\|\mathbf{x} - \mathbf{x}^{(n)}\|_{\mathbf{A}}}{\|\mathbf{x} - \mathbf{x}^{(n-1)}\|_{\mathbf{A}}}, \quad n = 1, 2, \ldots, \quad . \tag{1.33}$$

According to the estimate (1.16), the convergence rate (1.33) is bounded by $\|\mathbf{E}\|_{\mathbf{A}} + \gamma$; we have used that $\|\mathbf{S}\|_{\mathbf{A}_0,\mathbf{A}} \leq 1$. We approximate the term $\|\mathbf{E}\|_{\mathbf{A}}$ by a procedure described in Appendix 1.8.1. It is approximately 0.15 and 0.62 for the variant with the Poisson and the jump-1024 problem, respectively. Looking at the results we see that all the computed values of (1.33) are less than the corresponding bounds.

Let us first comment on the result for the Poisson problem. The convergence rates of the variants with $\gamma = 10^{-3}$ and $\gamma = 10^{-4}$ are approximately the same as the convergence rate of the exV-cycle method. The rates are significantly lower than its bounds in the first few V-cycle iterations, but they gradually deteriorate to approximately the value of the bound in the last V-cycle iterations. The convergence rate of the variant with $\gamma = 0.3$ is approximately constant 0.3. Here we don't see the usual deterioration of the convergence rate after the first V-cycle iterations. The bound for this variant is approximately 0.45.

Let us focus on the results for the jump-1024 problem. The convergence rate of the exV-cycle method doesn't deteriorate to the value of its approximate bound 0.62, but it stays under 0.15. This is an interesting behaviour since 0.15 is approximately the value of the bound on the rate of convergence of the exV-cycle method for the Poisson problem. The convergence rates of the variants with

$\gamma = 10^{-3}$, $\gamma = 10^{-4}$, are in the first V-cycle iterations approximately the same as the rate of the exV-cycle method. They, however, eventually deteriorate to the expected bounds. The deterioration happens sooner for the variant with $\gamma = 10^{-3}$.

The convergence rate of the variant with $\gamma = 0.3$ is approximately 0.3 in the first few iterations then it deteriorates to 0.62. This is another interesting behaviour since 0.62 is the value of the bound on the convergence rate of the exV-cycle method. The bound on the convergence rate of the inV-cycle method with $\gamma = 0.3$ is 0.92.

In these experiments we see that the estimate of the rate of convergence of the inV-cycle method with the assumption on a relative coarsest level accuracy is an accurate estimate of the worst-case convergence rate if γ is smaller than $\|\mathbf{E}\|_{\mathbf{A}}$.

We also plot the $\mathbf{A}$-norm of the error and the number of CG iterations on the coarsest level. We see that the number of CG iterations performed in the variants with the jump-1024 problem is significantly higher than in the variants with the Poisson problem.

To find out whether the inV-cycle methods reach the same level of attainable accuracy as the exV-cycle methods, we perform an experiment, where we stop the V-cycle method on the finest level after 50 V-cycle iterations. The results are summarized in Figure 1.3. We see that the considered inV-cycle methods reach the same level of attainable accuracy as the exV-cycle methods.

1.6.2 Accuracy of the estimates for inV-cycle methods with a relative residual coarsest-level stopping criterion

In this experiment, we study the accuracy of the results for a inV-cycle methods with a relative residual coarsest-level stopping criterion discussed in Section 1.4.

We consider the same problems and analogous V-cycle methods as in the motivational experiments in Section 1.2.1. We stop CG on the coarsest level using the relative residual stopping criterion (1.20), i.e., when

$$\frac{\|\mathbf{f}_0 - \mathbf{A}_0\mathbf{v}_{0,\mathrm{in}}\|}{\|\mathbf{f}_0\|} \leq \tau,$$

and choose $\tau = 10^{-4}\|\mathbf{T}\|^{-1}\|\mathbf{A}\|^{-\frac{1}{2}}\|\mathbf{A}_0^{-1}\|^{-\frac{1}{2}}$. We approximate the terms $\|\mathbf{T}\|$, $\|\mathbf{A}\|$, $\|\mathbf{A}_0^{-1}\|$ using MATLAB function `eigs`.

We run the V-cycle method starting with a zero initial approximate solution and stop when the $\mathbf{A}$-norm of the error on the finest level is (approximately) lower than 10^{-11}. In order to find out whether the results are substantially affected by the use of the finite precision arithmetic, we run the computation both in the standard MATLAB double precision and also in a simulated quad precision using the Advanpix toolbox [1].

After each V-cycle iteration we compute the $\mathbf{A}$-norm of the relative difference of the exV-cycle and inV-cycle approximations after one V-cycle iteration (1.32). The V-cycle methods for both problems reach the desired accuracy in 9 V-cycle iterations. The results are summarized in Figure 1.4. According to the discussion in Section 1.4 the relative difference (1.32) should be less than

$$\tau\|\mathbf{T}\|\|\mathbf{A}\|^{\frac{1}{2}}\|\mathbf{A}_0^{-1}\|^{\frac{1}{2}}\|\mathbf{S}\|_{\mathbf{A}_0,\mathbf{A}}.$$

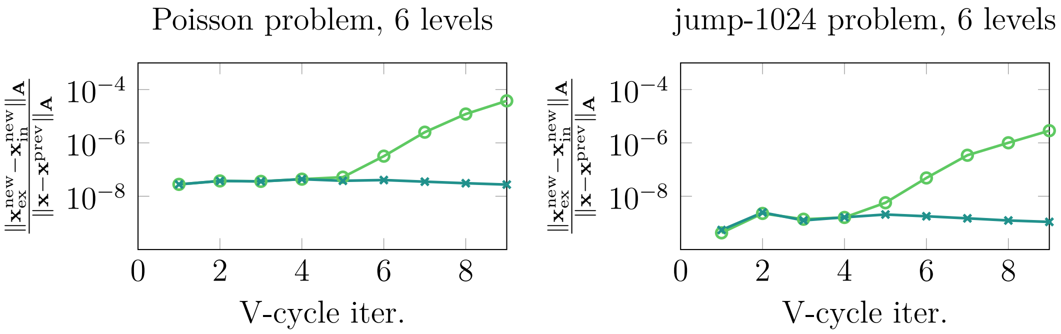


Figure 1.4 Testing accuracy of the estimate discussed in Section 1.4. We consider the V-cycle method with CG as the solver on the coarsest level. CG is stopped using the relative residual stopping criterion (1.20) with $\tau = 10^{-4}\|\mathbf{T}\|^{-1}\|\mathbf{A}\|^{-\frac{1}{2}}\|\mathbf{A}_0^{-1}\|^{-\frac{1}{2}}$. The computation is done in standard MATLAB double precision (—○—) and in simulated quad precision using the Advapix toolbox (—×—).

Bounding $\|\mathbf{S}\|_{\mathbf{A}_0,\mathbf{A}}$ by one from above and considering our choice of τ, we get that the relative difference (1.32) should be less than 10^{-4}. We see that this is true for all of the computed values. The computed values are however significantly smaller than the estimate. This may be a consequence of the usage of the estimates (1.24) and (1.25) in the derivation of the estimates in Section 1.4.

We see that the relative difference (1.32) for the variant computed in double precision starts increasing after the 5th V-cycle iterations, whereas the relative difference for the variant computed in the simulated quad precision stay approximately at the same level. We thus strongly believe that the increase of the values computed in double is caused by the use of the finite precision arithmetic.

1.6.3 inV-cycle method satisfying the absolute coarsest-level accuracy assumption

In this experiment we study the behavior of the inV-cycle method with a coarsest-level solver that is stopped when the assumption on an absolute coarsest-level accuracy is satisfied and examine the accuracy of estimates presented in Theorem 1.2.

We consider the same problems and analogous V-cycle methods as in the motivational experiments in Section 1.2.1. The only difference is that we stop CG on the coarsest level when inequality (1.18) (approximately) holds, i.e., when

$$\|\mathbf{v}_0 - \mathbf{v}_{0,\rm in}\|_{\mathbf{A}_0} \leq \epsilon.$$

We choose $\epsilon = \theta(1 - \|\mathbf{E}\|_{\mathbf{A}})$, where $\theta = 10^{-4}$ or $\theta = 10^{-11}$. We approximate $\|\mathbf{E}\|_{\mathbf{A}}$ as in the experiments in Section 1.6.1. We run the V-cycle method starting with a zero initial approximate solution and stop after 15 V-cycle iterations.

The results are summarized in Figure 1.5. After each V-cycle iteration we compute the $\mathbf{A}$-norm of the difference of the exV-cycle and inV-cycle approximations after n V-cycle iterations, i.e.,

$$\|\mathbf{x}_{\rm ex}^{(n)} - \mathbf{x}_{\rm in}^{(n)}\|_{\mathbf{A}}, \quad n = 1, 2, \dots, \quad . \tag{1.34}$$

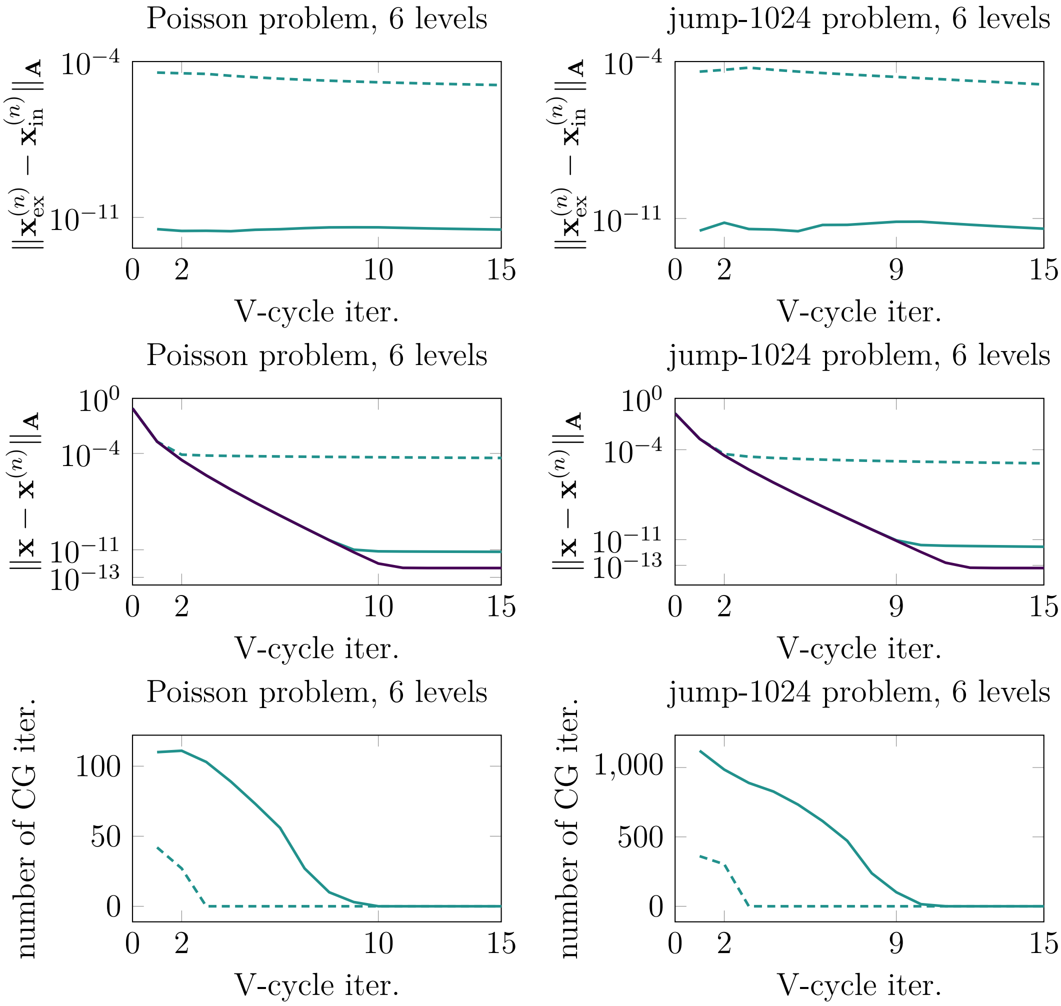


Figure 1.5 Properties of inV-cycle methods with CG as the solver on the coarsest level, which is stopped when the assumption on the absolute coarsest-level accuracy (1.18) (approximately) holds with $\epsilon = \theta(1 - \|\mathbf{E}\|_{\mathbf{A}})$, where $\theta = 10^{-4}$ (- - - -) or $\theta = 10^{-11}$ (——). For comparison we also include the $\mathbf{A}$-norm of the error of the exV-cycle method (——).

According to estimate (1.19) from Theorem 1.2, the norm of the difference (1.34) should be less than

$$\frac{\epsilon \|\mathbf{S}\|_{\mathbf{A}_0,\mathbf{A}}}{1 - \|\mathbf{E}\|_{\mathbf{A}}}.$$

Bounding $\|\mathbf{S}\|_{\mathbf{A}_0,\mathbf{A}}$ from above by one and considering our choice of ϵ, we get that the difference (1.34) should be less than θ. Looking at the results, we see that the computed values (1.34) are slightly less than θ. The estimate (1.19) is accurate for these numerical experiments.

The convergence of the inV-cycle and exV-cycle methods are approximately the same until they reach the level θ. The $\mathbf{A}$-norm of the error of the inV-cycle method then starts decreasing with a significantly slower rate. At this point the stopping criterion on the coarsest-level is automatically satisfied and the coarsest-level solver is not used. The method perform only smoothing on the fine levels.

We see that the choice of ϵ, respectively θ, determines the finest-level accuracy of the inV-cycle approximation. If we look at the number of coarsest-level solver iterations they are decreasing with each V-cycle iteration until they reach zero. The number of CG iterations performed for the variant with $\theta = 10^{-4}$ is significantly smaller than for the variant with $\theta = 10^{-11}$.

The behaviour is analogous for the two problems, the method for the jump-1024 requires significantly more coarsest-level iterations.

1.6.4 inV-cycle method with absolute coarsest-level stopping criteria

In this experiment we study the behaviour of inV-cycle methods with an absolute coarsest-level stopping criteria based on upper bounds of the $\mathbf{A}_0$-norm of the errors.

We run analogous numerical experiments as in Section 1.6.3. The only difference is that we stop CG on the coarsest-level using the stopping criterion (1.28), i.e., when

$$\eta(\mathbf{v}_{0,\mathrm{in}}) \leq \epsilon,$$

where η is an upper bound on the $\mathbf{A}_0$-norm of the error of the coarsest-level solver. We again choose $\epsilon = \theta(1 - \|\mathbf{E}\|_{\mathbf{A}})$, where $\theta = 10^{-4}$ or $\theta = 10^{-11}$. We consider two choices of η. First, the residual based upper bound (1.31). We label this variant as RES. We approximate the term $\|\mathbf{A}_0^{-1}\|$ using the MATLAB function `eigs`. Second, the Gauss-Radau upper bound on the $\mathbf{A}_0$-norm of the error in CG stated in [18, second inequality in (3.5) with updating formula for a coefficient (3.3)]. This upper bound is based on the interpretation of CG as a procedure for computing a Gauss-Radau quadrature approximation to a Riemann-Stieltjes integral. To compute this upper bound we need an lower bound on the smallest eigenvalue of the matrix $\mathbf{A}_0$. We approximate the smallest eigenvalue of $\mathbf{A}_0$ using the MATLAB `eigs` function and use its $1 - 10^{-3}$ multiple as the lower bound. We label this variant as GR. For comparison we include in the plots the results computed in Section 1.6.3 where CG is stopped on the coarsest-level when inequality (1.18) (approximately) holds. We label this variant as ERR.

We run the V-cycle method starting with a zero initial approximate solution and stop after 15 V-cycle iterations. The results are summarized in Figure 1.6.

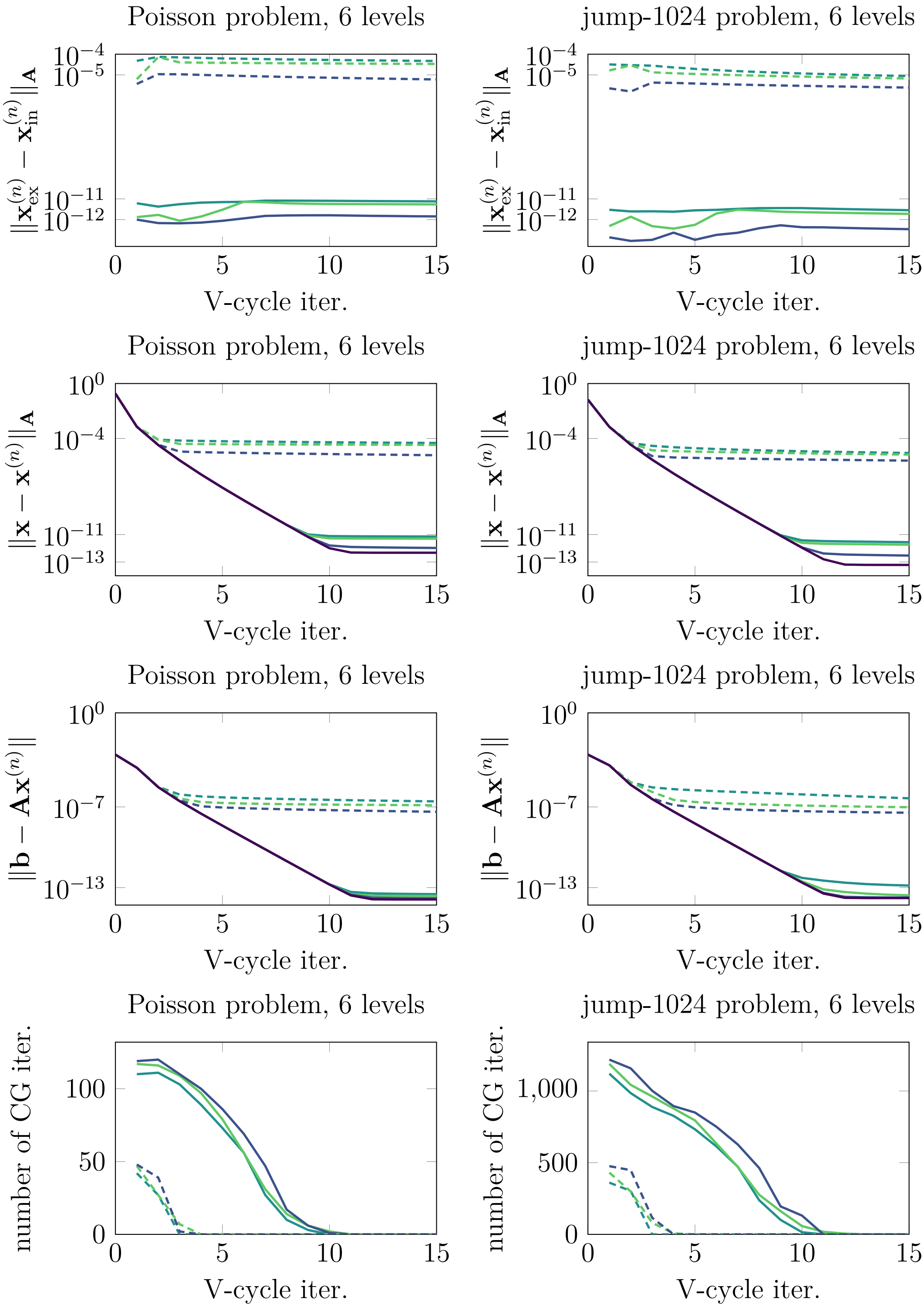


Figure 1.6 Properties of inV-cycle methods with CG as the solver on the coarsest level, which is stopped by an absolute criterion based on upper bounds of the $\mathbf{A}_0$-norm of the errors; variant ERR with $\theta = 10^{-4}$ (----) or $\theta = 10^{-11}$ (——), variant GR with $\theta = 10^{-4}$ (----) or $\theta = 10^{-11}$ (——), variant RES with $\theta = 10^{-4}$ (----) or $\theta = 10^{-11}$ (——). For comparison we also include the $\mathbf{A}$-norm of the error and Euclidean norm of residual of the exV-cycle method (——).

After each V-cycle iteration we compute the $\mathbf{A}$-norm of the difference of the exV-cycle and inV-cycle approximations after n V-cycle iterations (1.34). According to the discussion in Section 1.5 and the choice of ϵ, the norm of the difference (1.34) should be less than θ. Looking at the results we see that all values (1.34) are lower than the corresponding θ. We see that estimate (1.29) is the most accurate for the variant ERR and the loosest for the variants RES. When performing the experiments we observed that the Gauss-Radau upper bound on the $\mathbf{A}_0$-norm of the error used in the GR variants is more accurate than the residual based estimate (1.31) used in the RES variants. The more accurate the upper bound on the $\mathbf{A}_0$-norm of the error on the coarsest-level is used in the stopping criterion the more accurate estimate (1.29) is and the less CG iterations on the coarsest-level are performed.

Looking at the $\mathbf{A}$-norms of the error, we see that the variants GR and RES with stopping criteria based on the upper bounds of the $\mathbf{A}_0$-norm of the coarsest-level errors have analogous convergence behavior as the variant ERR with stopping criteria based on the $\mathbf{A}_0$-norm of the coarsest-level errors.

Based on these experiments, we believe that automatic satisfaction of the coarsest-level criteria can be used as a heuristic indicator that the $\mathbf{A}$-norm of the error on the finest level is at the level of θ. Another heuristic indicator that we reached the desired finest-level accuracy might be a stagnation of the norm of the finest-level residual.

1.6.5 Performance of inV-cycle methods with absolute coarsest-level stopping criteria

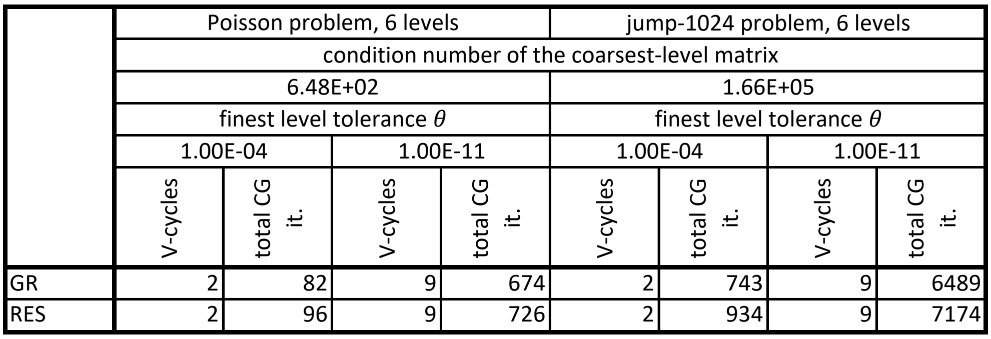
In this experiment, we evaluate the performance of inV-cycle methods with an absolute coarsest-level stopping criteria considered in Section 1.6.4.

We consider the same problems and analogous V-cycle methods. The only difference is that we don't use a computed approximation of $\|\mathbf{E}\|_\mathbf{A}$ but assume that $\|\mathbf{E}\|_\mathbf{A} < 2/3$ for both problems. The assumption $\|\mathbf{E}\|_\mathbf{A} < 2/3$ should be a valid assumption for most of the well set up V-cycle methods. For difficult problems it may be safer to consider it closer to one. Our goal is to compute approximations whose $\mathbf{A}$-norm of the error is approximately at the level of 10^{-4} and 10^{-11}, respectively. According to the discussion in Section 1.5 we choose $\epsilon = (1 - 2/3)\theta$, where $\theta = 10^{-4}$ and $\theta = 10^{-11}$.

We run the V-cycle method starting with a zero initial approximate solution and stop when the $\mathbf{A}$-norm of the error is (approximately) lower than 10^{-4} and 10^{-11} for the variants with $\theta = 10^{-4}$ and $\theta = 10^{-11}$, respectively. For both problems the exV-cycle method requires 2 and 9 V-cycle iterations to reach the desired finest-level accuracy 10^{-4} and 10^{-11}, respectively. The results of the inV-cycle methods are summarized in Figure 1.7.

We see that the inV-cycle methods converge to the desired accuracy in the same number of V-cycle iterations as the exV-cycle methods. The goal of the coarsest-level stopping strategy is thus satisfied. The methods works well for both problems with the same choice of the parameter ϵ. The variants RES, require more CG iterations on the coarsest level than the variants GR.

We may compare the total number of CG iterations in the variants GR and RES with the total number of CG iterations in the variants with a relative residual



	Poisson problem, 6 levels				jump-1024 problem, 6 levels			
	condition number of the coarsest-level matrix							
	6.48E+02				1.66E+05			
	finest level tolerance θ				finest level tolerance θ			
	1.00E-04		1.00E-11		1.00E-04		1.00E-11	
	V-cycles	total CG it.	V-cycles	total CG it.	V-cycles	total CG it.	V-cycles	total CG it.
GR	2	82	9	674	2	743	9	6489
RES	2	96	9	726	2	934	9	7174

Figure 1.7 Properties of inV-cycle methods with CG as the solver on the coarsest level, which is stopped by the absolute criteria based on upper bounds of the $\mathbf{A}_0$-norm of the errors.

stopping criterion in Figure 1.1. We see that the number of total CG iterations in the GR and RES variants are not the lowest possible, such that an inV-cycle method converges to the desired accuracy in the same number of V-cycles as the exV-cycle method, but they also aren't substantially high.

To see how the coarsest-level stopping strategy may be affected by the change of the size of the coarsest-level problem and the change of the number of levels in the V-cycle method we run experiments where we consider the same problem on the finest level, but just three level V-cycle methods. The size of the coarsest-level problems is 101761 DoFs. The results are summarized in Figure 1.8.

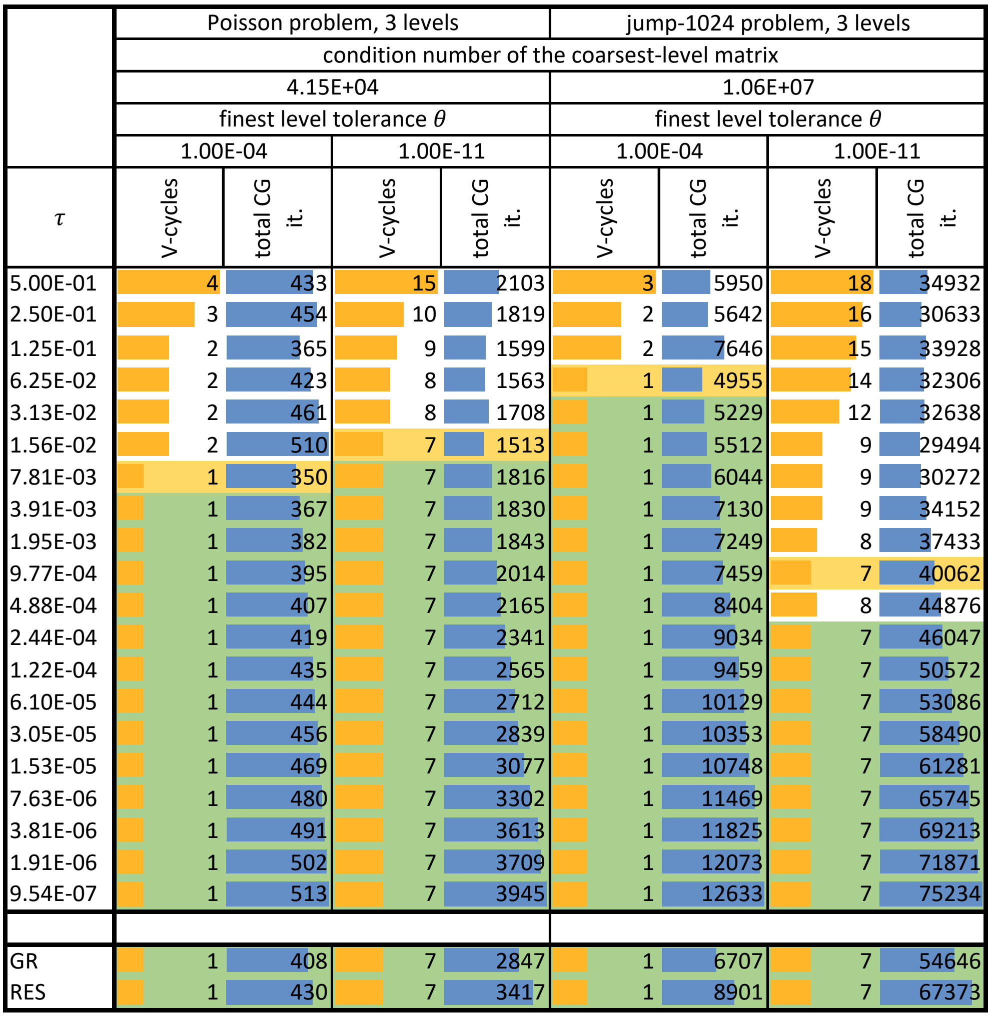
We see analogous behavior as in the experiment with six level V-cycle methods. The variants GR and RES converge to the desired accuracy in the same number of V-cycle iterations as the exV-cycle methods.

The main benefit of the stopping strategy is that we don't have to try different parameters for different problems or when we want to reach different finest-level tolerances or when the size of the coarsest-level problem changes. The parameter θ is chosen the same as the finest-level tolerance we are aiming for.

1.7 Conclusions and open problems

In this paper we present an approach to analyzing the effects of approximate coarsest-level solves on the convergence of the V-cycle method for SPD problems. We use the results to give an answer to the question of how the choice of tolerance in the relative residual stopping criterion for the coarsest-level solver may affect the convergence of the V-cycle method. We present novel coarsest-level stopping criterion which we may use to control the difference between the computed approximation and the approximation which would be computed by the exV-cycle method. This coarsest-level stopping criterion may thus be set up such that the method converges to a chosen finest-level accuracy in (nearly) the same number of V-cycle iterations as the exV-cycle method. The stopping strategy achieves this goal in various numerical experiments. In a future work we would like to test this coarsest-level stopping strategy within the algebraic multigrid methods.

In this work we focus on the use of multigrid methods as a standalone solver. Multigrid methods are, however, also frequently used as a preconditioner for a Krylov subspace method. It would be interesting to investigate how the results



	Poisson problem, 3 levels				jump-1024 problem, 3 levels			
	condition number of the coarsest-level matrix							
	4.15E+04				1.06E+07			
	finest level tolerance θ				finest level tolerance θ			
	1.00E-04		1.00E-11		1.00E-04		1.00E-11	
τ	V-cycles	total CG it.	V-cycles	total CG it.	V-cycles	total CG it.	V-cycles	total CG it.
5.00E-01	4	433	15	2103	3	5950	18	34932
2.50E-01	3	454	10	1819	2	5642	16	30633
1.25E-01	2	365	9	1599	2	7646	15	33928
6.25E-02	2	423	8	1563	1	4955	14	32306
3.13E-02	2	461	8	1708	1	5229	12	32638
1.56E-02	2	510	7	1513	1	5512	9	29494
7.81E-03	1	350	7	1816	1	6044	9	30272
3.91E-03	1	367	7	1830	1	7130	9	34152
1.95E-03	1	382	7	1843	1	7249	8	37433
9.77E-04	1	395	7	2014	1	7459	7	40062
4.88E-04	1	407	7	2165	1	8404	8	44876
2.44E-04	1	419	7	2341	1	9034	7	46047
1.22E-04	1	435	7	2565	1	9459	7	50572
6.10E-05	1	444	7	2712	1	10129	7	53086
3.05E-05	1	456	7	2839	1	10353	7	58490
1.53E-05	1	469	7	3077	1	10748	7	61281
7.63E-06	1	480	7	3302	1	11469	7	65745
3.81E-06	1	491	7	3613	1	11825	7	69213
1.91E-06	1	502	7	3709	1	12073	7	71871
9.54E-07	1	513	7	3945	1	12633	7	75234
GR	1	408	7	2847	1	6707	7	54646
RES	1	430	7	3417	1	8901	7	67373

Figure 1.8 Properties of inV-cycle methods with CG as the solver on the coarsest level, which is stopped by a relative residual criterion with various tolerance τ, or by an absolute criterion based on upper bounds of the $\mathbf{A}_0$-norm of the errors; variants GR and RES. The bright yellow and green color highlight variants that converge in the same number of V-cycles as the exV-cycle method. The bright yellow variants achieve this in the least total number of CG iterations on the coarsest-level.

obtained in this paper could be utilized in this setting. In general an inV-cycle method would have to be applied as a flexible preconditioner.

Other open problems include the generalization to non-symmetric problems or to other multigrid schemes such as the W-cycle scheme or the full multigrid scheme.

1.8 Appendix

1.8.1 Numerical approximation of $\|\mathbf{E}\|_{\mathbf{A}}$

In this section we describe a procedure for numerical approximation of the **A**-norm of the error propagation matrix **E** of the exV-cycle scheme. We consider an exV-cycle scheme where the pre- and post- smoothing is each accomplished by one iteration of the symmetric Gauss-Seidel method. Thanks to the use of the symmetric Gauss-Seidel smoother the matrix **E** is symmetric and there exist a symmetric matrix **B** such that $\mathbf{E} = \mathbf{I} - \mathbf{B}^{-1}\mathbf{A}$; see, e.g., [23]. Then

$$\|\mathbf{E}\|_{\mathbf{A}} = \|\mathbf{I} - \mathbf{B}^{-1}\mathbf{A}\|_{\mathbf{A}} = \|\mathbf{A}^{\frac{1}{2}}(\mathbf{I} - \mathbf{B}^{-1}\mathbf{A})\mathbf{A}^{-\frac{1}{2}}\| = \|\mathbf{I} - \mathbf{A}^{\frac{1}{2}}\mathbf{B}^{-1}\mathbf{A}^{\frac{1}{2}}\|.$$

Since the matrices $\mathbf{A}^{\frac{1}{2}}\mathbf{B}^{-1}\mathbf{A}^{\frac{1}{2}}$ and $\mathbf{B}^{-1}\mathbf{A}$ have the same eigenvalues there holds

$$\|\mathbf{I} - \mathbf{A}^{\frac{1}{2}}\mathbf{B}^{-1}\mathbf{A}^{\frac{1}{2}}\| = \|\mathbf{I} - \mathbf{B}^{-1}\mathbf{A}\|,$$

and consequently $\|\mathbf{E}\|_{\mathbf{A}} = \|\mathbf{E}\|$. We compute it using MATLAB function `eigs` (with the largest eigenvalue option) applied to the function

$$\mathbf{x} \mapsto \mathbf{x} - \mathbf{V}(\mathbf{A}_{0:J}, \mathbf{M}_{1:J}, \mathbf{N}_{1:J}, \mathbf{P}_{1:J}, \mathbf{A}\mathbf{x}, \mathbf{0}, J).$$

1.8.2 Derivation of inequalities (1.24) and (1.25)

In this section we present derivations of inequalities (1.24) and (1.25) used in Section 1.4, i.e.,

$$\begin{aligned} \|\mathbf{A}_0^{-1}\|^{-\frac{1}{2}}\|\mathbf{v}_0 - \mathbf{v}_{0,\text{in}}\|_{\mathbf{A}_0} &\leq \|\mathbf{f}_0 - \mathbf{A}_0\mathbf{v}_{0,\text{in}}\|, \\ \|\mathbf{b} - \mathbf{A}\mathbf{x}^{\text{prev}}\| &\leq \|\mathbf{A}\|^{\frac{1}{2}}\|\mathbf{x} - \mathbf{x}^{\text{prev}}\|_{\mathbf{A}}. \end{aligned}$$

Using that $\mathbf{A}_0\mathbf{v}_0 = \mathbf{f}_0$ and that $\mathbf{A}_0$ is SPD we have

$$\begin{aligned} \|\mathbf{v}_0 - \mathbf{v}_{0,\text{in}}\|_{\mathbf{A}_0}^2 &= (\mathbf{v}_0 - \mathbf{v}_{0,\text{in}})^\top \mathbf{A}_0(\mathbf{v}_0 - \mathbf{v}_{0,\text{in}}) \\ &= (\mathbf{A}_0^{-1}(\mathbf{f}_0 - \mathbf{A}_0\mathbf{v}_{0,\text{in}}))^\top \mathbf{A}_0(\mathbf{A}_0^{-1}(\mathbf{f}_0 - \mathbf{A}_0\mathbf{v}_{0,\text{in}})) \\ &= (\mathbf{f}_0 - \mathbf{A}_0\mathbf{v}_{0,\text{in}})^\top \mathbf{A}_0^{-1}\mathbf{A}_0\mathbf{A}_0^{-1}(\mathbf{f}_0 - \mathbf{A}_0\mathbf{v}_{0,\text{in}}) \\ &= (\mathbf{f}_0 - \mathbf{A}_0\mathbf{v}_{0,\text{in}})^\top \mathbf{A}_0^{-1}(\mathbf{f}_0 - \mathbf{A}_0\mathbf{v}_{0,\text{in}}) \leq \|\mathbf{A}_0^{-1}\|\|\mathbf{f}_0 - \mathbf{A}_0\mathbf{v}_{0,\text{in}}\|^2, \end{aligned}$$

which yields the first inequality. The second inequality can be derived using that $\mathbf{A}\mathbf{x} = \mathbf{b}$ and that **A** is SPD

$$\begin{aligned} \|\mathbf{b} - \mathbf{A}\mathbf{x}^{\text{prev}}\|^2 &= (\mathbf{b} - \mathbf{A}\mathbf{x}^{\text{prev}})^\top(\mathbf{b} - \mathbf{A}\mathbf{x}^{\text{prev}}) = (\mathbf{A}(\mathbf{x} - \mathbf{x}^{\text{prev}}))^\top((\mathbf{A}(\mathbf{x} - \mathbf{x}^{\text{prev}})) \\ &= (\mathbf{x} - \mathbf{x}^{\text{prev}})^\top \mathbf{A}^{\frac{1}{2}}\mathbf{A}\mathbf{A}^{\frac{1}{2}}(\mathbf{x} - \mathbf{x}^{\text{prev}}) \\ &\leq \|\mathbf{A}\|(\mathbf{x} - \mathbf{x}^{\text{prev}})^\top \mathbf{A}(\mathbf{x} - \mathbf{x}^{\text{prev}}) = \|\mathbf{A}\|\|\mathbf{x} - \mathbf{x}^{\text{prev}}\|_{\mathbf{A}}^2. \end{aligned}$$

Acknowledgments

The authors wish to thank Petr Tichý for his useful comments on error estimation in CG and Jaroslav Hron for his suggestions when generating the system matrices in FEniCS. The authors acknowledge the support of the Erasmus+ program that enabled Petr Vacek to spend the Winter semester 2021-2022 at Trinity College Dublin. During this visit the basis of the paper was developed.

Bibliography

[1] *Advanpix Multiprecision Computing Toolbox for MATLAB ver. 5.1.0.15432*. Yokohama, Japan: Advanpix LLC. URL: `https://www.advanpix.com/`.

[2] M. S. Alnaes, J. Blechta, J. Hake, et al. "The FEniCS Project Version 1.5". In: *Archive of Numerical Software* 3 (2015). DOI: `10.11588/ans.2015.100.20553`.

[3] A. Brandt. *Multigrid Techniques 1984 Guide with Applications to Fluid Dynamics Revised Edition*. Philadelphia, PA: SIAM, 2011. DOI: `10.1137/1.9781611970753`.

[4] W. L. Briggs, V. E. Henson, and S. F. McCormick. *A Multigrid Tutorial*. Second. Philadelphia, PA: SIAM, 2000, pp. xii+193. DOI: `10.1137/1.9780898719505`.

[5] A. Buttari et al. "Block low-rank single precision coarse grid solvers for extreme scale multigrid methods". In: *Numerical Linear Algebra with Applications* 29.1 (2022), e2407. DOI: `10.1002/nla.2407`.

[6] D. Calvetti et al. "Computable error bounds and estimates for the conjugate gradient method". In: *Numerical Algorithms* 25.1-4 (2000), pp. 75–88. DOI: `10.1023/A:1016661024093`.

[7] J. van den Eshof and G. L. G. Sleijpen. "Inexact Krylov subspace methods for linear systems". eng. In: *SIAM Journal on Matrix Analysis and Applications* 26.1 (2004), pp. 125–153. DOI: `10.1137/S0895479802403459`.

[8] H. Gahvari et al. "Systematic Reduction of Data Movement in Algebraic Multigrid Solvers". In: *2013 IEEE International Symposium on Parallel & Distributed Processing, Workshops and Phd Forum*. 2013, pp. 1675–1682. DOI: `10.1109/IPDPSW.2013.164`.

[9] G. H. Golub and G. Meurant. *Matrices, moments and quadrature with applications*. USA: Princeton University Press, 2010, pp. xxx+698.

[10] W. Hackbusch. *Iterative solution of large sparse systems of equations*. Second. Vol. 95. Applied Mathematical Sciences. Cham: Springer, 2016, pp. xxiii+509. DOI: `10.1007/978-3-319-28483-5`.

[11] M. R. Hestenes and E. Stiefel. "Methods of conjugate gradients for solving linear systems". In: *Journal of Research of the National Bureau of Standards* 49.6 (1952), pp. 409–436. DOI: `10.6028/jres.049.044`.

[12] M. Huber. "Massively parallel and fault-tolerant multigrid solvers on petascale systems". PhD thesis. Technical University of Munich, Germany, 2019. URL: `http://www.dr.hut-verlag.de/978-3-8439-3917-1.html`.

[13] A. Logg et al. *Automated Solution of Differential Equations by the Finite Element Method*. Springer, 2012. DOI: 10.1007/978-3-642-23099-8.

[14] D. A. May et al. “Extreme-scale multigrid components within PETSc”. In: *Proceedings of the Platform for Advanced Scientific Computing Conference*. 2016, pp. 1–12.

[15] S. F. McCormick, J. Benzaken, and R. Tamstorf. “Algebraic Error Analysis for Mixed-Precision Multigrid Solvers”. In: *SIAM Journal on Scientific Computing* 43.5 (2021), S392–S419. DOI: 10.1137/20M1348571.

[16] G. Meurant, J. Papež, and P. Tichý. “Accurate error estimation in CG”. In: *Numerical Algorithms* 88.3 (2021), pp. 1337–1359. DOI: 10.1007/s11075-021-01078-w.

[17] G. Meurant and P. Tichý. “Approximating the extreme Ritz values and upper bounds for the A-norm of the error in CG”. In: *Numerical Algorithms* 82.3 (2019), pp. 937–968. DOI: 10.1007/s11075-018-0634-8.

[18] G. Meurant and P. Tichý. “The behaviour of the Gauss-Radau upper bound of the error norm in CG”. In: *Numerical Algorithms* 94 (2023), pp. 847–876. DOI: 10.1007/s11075-023-01522-z.

[19] Y. Notay. “Convergence analysis of perturbed two-grid and multigrid methods”. In: *SIAM Journal on Numerical Analysis* 45.3 (2007), pp. 1035–1044. DOI: 10.1137/060652312.

[20] A. Reisner, L. N. Olson, and J. D. Moulton. “Scaling structured multigrid to 500k+ cores through coarse-grid redistribution”. In: *SIAM Journal on Scientific Computing* 40.4 (2018), pp. C581–C604.

[21] U. Trottenberg, C. W. Oosterlee, and A. Schuller. *Multigrid*. London: Academic Press, 2001.

[22] S. Williams et al. “s-Step Krylov Subspace Methods as Bottom Solvers for Geometric Multigrid”. In: *2014 IEEE 28th International Parallel and Distributed Processing Symposium*. 2014, pp. 1149–1158. DOI: 10.1109/IPDPS.2014.119.

[23] J. Xu. “Iterative methods by space decomposition and subspace correction”. In: *SIAM Review* 34.4 (1992), pp. 581–613. DOI: 10.1137/1034116.

[24] X. Xu and C.-S. Zhang. “Convergence Analysis of Inexact Two-Grid Methods: A Theoretical Framework”. In: *SIAM Journal on Numerical Analysis* 60.1 (2022), pp. 133–156. DOI: 10.1137/20M1356075.

[25] H. Yserentant. “Old and new convergence proofs for multigrid methods”. In: *Acta Numerica* 2 (1993), pp. 285–326.

2 A posteriori error estimates based on multilevel decompositions with large problems on the coarsest level

In the previous chapter we studied the effects of approximate coarsest-level solves on the convergence of a V-cycle method. In this chapter, we focus on a related question formulated in the introduction regarding multilevel a posteriori error estimates:

c) Consider the residual-based multilevel a posteriori error estimates such as in [33, Section 2.6]. Is it possible to compute the term associated with the coarsest-level approximately while preserving the efficiency and accuracy of the estimate?

We show that the way in which the term associated with the coarsest level is approximated is substantial. It can affect both the efficiency and accuracy of the overall error estimates and their robustness with respect to the size of the coarsest-level problem. We propose a new approximation of the coarsest-level term, based on using the conjugate gradient method with an appropriate stopping criterion. We prove that the resulting estimates still have the desired properties, even though we use approximate computation on the coarsest-level.

This chapter contains a version of the paper: P. Vacek, J. Papež and Z. Strakoš, “A posteriori error estimates based on multilevel decompositions with large problems on the coarsest level”, `https://arxiv.org/abs/2405.06532`, which was submitted to a peer-reviewed journal in May 2024.

2.1 Introduction

Multilevel methods [6, 19, 9, 39] are frequently used for solving systems of linear equations obtained from the discretization of partial differential equations (PDEs). They are applied either as standalone iterative solvers or as preconditioners. In *geometric* multigrid methods the hierarchy of systems is obtained by discretizations of an infinite dimensional problem on a sequence of nested meshes. In *algebraic* multigrid methods the coarse systems are constructed using algebraic properties of the matrix. Each multigrid cycle contains smoothing on fine levels, prolongation, restriction, and solving a system of linear equations on the coarsest level. Smoothing is typically done by a few iterations of a stationary iterative method. If the size permits, it is typical to solve the coarsest-level problem using a direct method based on LU or Cholesky decomposition. Although this does not provide a computed result with a zero error, many theoretical results on multigrid methods are proved under the assumption that the coarsest-level problem is solved exactly; see, e.g., [43, 45].

Multilevel methods can in practice also use hierarchies where the problem on the coarsest level is large and can only be solved approximately to a properly chosen accuracy, e.g., by Krylov subspace methods, or direct methods based on low-rank matrix approximations. This arises for problems on complicated domains or for large-scale problems solved on modern parallel computers; see, e.g., [10]. Effects of approximate coarsest-level solves on convergence of multigrid method were analysed, e.g., in [28, 44, 41].

The multilevel structure can also be used to construct estimates of total and algebraic errors; see, e.g., [4, 33, 20, 23, 31, 27]. The estimates of [4, 33, 20, 23, 31, 27] are, however, not suited for multilevel hierarchies with large coarsest-level problems, which are being used for complicated domains and/or in parallel implementations. They either assume that the coarsest-level problem is solved exactly [4, 31, 27], or they require computation of the term $\mathbf{r}_0^* \mathbf{A}_0^{-1} \mathbf{r}_0$ associated with the coarsest level, where $\mathbf{A}_0$ is the coarsest-level system matrix and $\mathbf{r}_0$ a projection of a finest-level residual to the coarsest level, [33, 20, 23]. The term $\mathbf{r}_0^* \mathbf{A}_0^{-1} \mathbf{r}_0$ can be approximated, e.g., using the conjugate gradient method (CG) as in [23], or by replacing the system matrix with a diagonal matrix as in [20]. Then proving efficiency and robustness of estimates becomes an important challenge.

In this text, we discuss properties of the error estimates in multilevel settings where the system matrix on the coarsest level is large and the associated terms are only approximated. We consider several a posteriori estimates on total and algebraic errors based on decomposing the error into a sequence of finite element subspaces and using either approximation properties of quasi-interpolation operators [4], stable splittings [33, 23], or so-called frames [20]. The main contribution of this paper is a new procedure for approximating the term associated with the coarsest level that is based on using the conjugate gradient method with an appropriate stopping criterion. We prove that the resulting estimates are *efficient and robust* with respect to the size of the coarsest-level problem.

The text is organized as follows. First, we present a model problem, its discretization, and the notation used in the text. Derivations of error estimates for total and algebraic errors are presented in Section 2.3. In Section 2.4, we comment on the efficiency of the bounds. Main results are presented in Section 2.5 where we describe how to replace the (uncomputable) terms in the estimates by a computable approximation and present an adaptive procedure for approximating the coarsest-level term $\mathbf{r}_0^* \mathbf{A}_0^{-1} \mathbf{r}_0$. Numerical illustrations are given in Section 2.6 and conclusions in Section 2.7. Not to interrupt the presentation, we present detailed theoretical results, which are used in the derivation of the estimates, in Appendices. Appendix 2.8.1 recalls some standard results from PDE and finite element method (FEM) analyses. Appendix 2.8.2 presents properties of the quasi-interpolation operator, and Appendix 2.8.3 recalls results on stable-splittings and frames. This enables an easy comparison of different results that are presented separately in literature.

2.2 Model problem, setting, and notation

The estimates will be studied for a standard model problem, a prototype for elliptic equations, the Poisson's problem with homogeneous Dirichlet boundary conditions. Let $\Omega \subset \mathbb{R}^d$, $d = 2, 3$, be an open bounded polytope with a Lipschitz-continuous boundary. Given $f \in L^2(\Omega)$, the weak form reads: find $u \in H_0^1(\Omega)$

such that

$$\int_\Omega \nabla u \cdot \nabla v = \int_\Omega f v \qquad \forall v \in H_0^1(\Omega). \tag{2.1}$$

In this section, we introduce notation for meshes and finite element spaces, and the multilevel framework. Further, we present the Galerkin finite element discretization of the model problem on a particular level, define its approximate solution, the error, and (scaled) residuals associated with individual levels of the multilevel hierarchy.

Similarly to a standard literature, we introduce some simplifying assumptions, e.g., on the model problem or mesh hierarchies. This is done in order to reduce the complexity of proofs (that are already quite technical) and to allow us to refer to particular results in the literature. We use a standard notation for Lebesgue and Sobolev (Hilbert) spaces, norms, and seminorms; see, e.g., [8].

2.2.1 Notation for a single level

Throughout the paper, we consider simplicial meshes of Ω, matching in the sense that for two distinct elements of a mesh $\mathcal{T}$ (triangles in 2D, or tetrahedra in 3D), their intersection is either an empty set or a common node (vertex), edge, or face. By $\mathcal{E}_\mathcal{T}$ and $\mathcal{N}_\mathcal{T}$ we denote the set of $(d-1)$-dimensional faces and set of nodes in the mesh $\mathcal{T}$, respectively. By $\mathcal{E}_{\mathcal{T},\mathrm{int}}$ we denote the set of all faces that are not on the boundary $\partial\Omega$. By $\mathcal{K}_\mathcal{T} \subset \mathcal{N}_\mathcal{T}$ we denote the set of all nodes in the mesh $\mathcal{T}$, which are not on the boundary, i.e., *free nodes*. For any element (simplex) $K \in \mathcal{T}$, $\mathcal{E}_K \subset \mathcal{E}_\mathcal{T}$ denotes the set of faces of the element K, $\mathcal{N}_K \subset \mathcal{N}_\mathcal{T}$ denotes the set of nodes of the element K, $\mathcal{E}_{K,\mathrm{int}} = \mathcal{E}_K \cap \mathcal{E}_{\mathcal{T},\mathrm{int}}$, and $\mathcal{K}_K = \mathcal{N}_K \cap \mathcal{K}_\mathcal{T}$. We use hash to denote the cardinality of a set, for example $\#\mathcal{K}_\mathcal{T}$ denotes the number of free nodes in the mesh $\mathcal{T}$. For the ease of presentation, we will assume that the nodes in $\mathcal{N}_\mathcal{T}$ are ordered such that nodes $1, \dots, \#\mathcal{K}_\mathcal{T}$ belong to $\mathcal{K}_\mathcal{T}$, i.e., we first have the free nodes and then the nodes on the boundary.

By h_K we denote the diameter of $K \in \mathcal{T}$ and define a mesh-size $h_\mathcal{T} \in L^\infty(\Omega)$ as

$$h_\mathcal{T}(x) = h_K, \quad x \in K, \quad \forall K \in \mathcal{T}.$$

Similarly h_ω denotes the diameter of a domain ω. We in particular use h_Ω, the diameter of the domain Ω. By $|\omega|$ we denote the Lebesgue measure of a domain ω.

For any element $K \in \mathcal{T}$, ω_K denotes the patch of elements that share at least one common vertex with K, i.e.,

$$\omega_K = \bigcup_{K' \in \mathcal{T}; K' \cap K \neq \emptyset} K'.$$

By ρ_K we denote the diameter of the largest ball inscribed in the element K.

For every node $z \in \mathcal{N}_\mathcal{T}$, let ϕ_z be the continuous piecewise linear function (*hat function*) that has a value one at node z and vanishes at all the other nodes in $\mathcal{N}_\mathcal{T}$. Let $S_\mathcal{T}$ denote the space of continuous, piecewise linear functions,

$$S_\mathcal{T} = \{v \in H^1(\Omega), v|_K \in \mathbb{P}^1(K),\ \forall K \in \mathcal{T}\} = \mathrm{span}\{\phi_z,\ z \in \mathcal{N}_\mathcal{T}\}$$

and $V_\mathcal{T} \subset S_\mathcal{T}$ the subspace of functions vanishing on the boundary $\partial\Omega$,

$$V_\mathcal{T} = \{v \in H_0^1(\Omega), v|_K \in \mathbb{P}^1(K),\ \forall K \in \mathcal{T}\} = \mathrm{span}\{\phi_z,\ z \in \mathcal{K}_\mathcal{T}\}.$$

We write the basis of $V_{\mathcal{T}}$ as $\Phi_{\mathcal{T}} = (\phi_1, \ldots, \phi_{\#\mathcal{K}_{\mathcal{T}}})$.

One of the key properties of a mesh that affects the size of the constants in the estimates derived below in this text is the so-called *shape regularity* of the mesh. This can be quantified by the shape-regularity constant, i.e., the smallest $\gamma_{\mathcal{T}} > 0$ satisfying

$$\frac{h_K}{\rho_K} \leq \gamma_{\mathcal{T}}, \quad \forall K \in \mathcal{T}; \tag{2.2}$$

see, e.g., [34, p. 484].

2.2.2 Multilevel framework

As the title of the paper suggests, we will work with a sequence of levels $j = 0, 1, \ldots, J$. For some parts of the theory, we will consider also infinite sequences of levels $j = 0, 1, \ldots, J, \ldots$. To simplify the previously introduced notation, we will replace in the subscripts $\mathcal{T}_j$ by j to denote objects associated with the mesh $\mathcal{T}_j$ on the jth level.

Let $\mathcal{T}_0$ be an initial mesh of Ω. We consider a sequence of meshes $\mathcal{T}_1, \mathcal{T}_2, \ldots$ obtained by successive uniform dyadic refinements of $\mathcal{T}_0$, i.e., each element is refined into 2^d elements (congruent triangles in 2D, for a proper nondegenerating 3D mesh refinement; see, e.g., [46]). We recall that S_j and V_j, $j = 0, 1, \ldots$, are the finite element spaces of continuous piecewise linear functions on $\mathcal{T}_j$, respectively spaces of continuous piecewise linear functions on $\mathcal{T}_j$ that vanish on the boundary $\partial\Omega$. These spaces are nested, i.e.,

$$S_0 \subset S_1 \subset \cdots \subset H^1(\Omega), \qquad V_0 \subset V_1 \subset \cdots \subset H_0^1(\Omega).$$

On each level j, we consider a quasi-interpolation operator

$$I_{V_j} : L^1(\Omega) \to V_j$$

with the definition and properties described in detail in Appendix 2.8.2.

Due to the uniform refinement, the mesh sizes h_j of $\mathcal{T}_j$, $j \geq 0$, satisfy $h_j = 2^{-j} h_0$. Moreover, the uniform refinement assures that the shape-regularity constants γ_j of the meshes are the same on all levels in 2D, i.e., $\gamma_0 = \gamma_j$, $j \in \mathbb{N}$, and that in 3D there exists a constant $C_{3\mathrm{D}} > 0$ such that $\gamma_j \leq C_{3\mathrm{D}} \gamma_0$, $j \in \mathbb{N}$; see [46].

2.2.3 Discretization, approximate solution, and residuals

Discretizing the model problem (2.1) on the subspace V_J, for some $J \geq 0$, using the Galerkin method reads as: find $u_J \in V_J$ such that

$$\int_\Omega \nabla u_J \cdot \nabla w_J = \int_\Omega f w_J, \quad \forall w_J \in V_J. \tag{2.3}$$

Let $v_J \in V_J$ be a (computed) approximation of the discrete solution u_J. Our goal is to bound the energy norm of the total error $e = u - v_J$ using computable quantities involving v_J and f. The squared energy norm of the error $\|\nabla e\|^2$ can be expressed as

$$\|\nabla e\|^2 = \|\nabla(u - v_J)\|^2 = \int_\Omega \nabla(u - v_J) \cdot \nabla(u - v_J) = \int_\Omega f(u - v_J) - \nabla v_J \cdot \nabla(u - v_J).$$

Denote by $(H_0^1(\Omega))^{\#}$ the dual space to $H_0^1(\Omega)$ and define the residual $r \in (H_0^1(\Omega))^{\#}$ as

$$\langle r, w\rangle = \int_\Omega fw - \nabla v_J \cdot \nabla w, \quad \forall w \in H_0^1(\Omega). \tag{2.4}$$

Then (2.4) yields the so-called residual equation

$$\|\nabla e\|^2 = \langle r, e\rangle, \tag{2.5}$$

which is the key formula for the development of error bounds presented below. Moreover, it can be shown (see, e.g., [42, Section 1.4.1]) that

$$\|\nabla e\| = \|r\|_{\left(H_0^1(\Omega)\right)^{\#}}.$$

In order to derive computable estimates we consider Riesz representations of the infinite-dimensional residual r in the finite-dimensional spaces V_j, $j = 0, 1, \ldots$. In particular, let $r_j \in V_j$, $j = 1, \ldots$, be the Riesz representation of r in the space V_j with the scaled L^2-inner product, i.e.,

$$\langle r, w_j\rangle = \int_\Omega h_j^{-2} r_j w_j, \quad \forall w_j \in V_j, \tag{2.6}$$

and let $r_0 \in V_0$ be the Riesz representation of the residual r in the space V_0 with the H_0^1-inner product, i.e.,

$$\langle r, w_0\rangle = \int_\Omega \nabla r_0 \cdot \nabla w_0, \quad \forall w_0 \in V_0. \tag{2.7}$$

These definitions are used in [33, Section 2.6] where r_j are called *scaled residuals*. In [4, Section 5] the authors use Riesz representations of r in the spaces V_j, $j = 1, \ldots, J$, with the classical L^2-inner products and call them discrete residuals. The different definition we use results in a slightly different form of the estimates below in comparison to [4, Section 5].

2.3 Residual-based error estimates

In this section we recall several published error estimates with their derivation. We first recall the standard residual-based error estimator for the discretization error in a single-level setting assuming exact algebraic computations or to steer an adaptive mesh refinement.

Consider the model problem (2.1) discretized on a level $J \geq 0$ of a multilevel hierarchy as in Section 2.2.2. The classical residual-based estimator (see, e.g., [1, Section 3], [42, Section 1.4]) is for a (computed) approximation $v_J \in V_J$ defined as

$$\begin{aligned}
\eta_J^2 &= \left(\eta_J^{\mathrm{RHS}}\right)^2 + \left(\eta_J^{\mathrm{JUMP}}\right)^2 + (\mathrm{osc}_J)^2, \\
\left(\eta_J^{\mathrm{RHS}}\right)^2 &= \sum_{K \in \mathcal{T}_J} h_K^2 \|f_K\|_K^2, \\
\left(\eta_J^{\mathrm{JUMP}}\right)^2 &= \frac{1}{2} \sum_{K \in \mathcal{T}_J} h_K \sum_{E \in \mathcal{E}_{K,\mathrm{int}}} \| [\nabla v_J] \|_E^2, \\
(\mathrm{osc}_J)^2 &= \sum_{K \in \mathcal{T}_J} h_K^2 \|f - f_K\|_K^2,
\end{aligned}$$

where $[\cdot]$ denotes the jump of a piecewise constant function over the $(d-1)$-dimensional faces (faces in 3D and edges in 2D) and f_K is the mean value of f on K. Other choices of f_K are also possible; see, e.g., [20].

The following result (see, e.g., [4, Lemma 3], [35, Section 4], or [42, Section 1.4]) will be useful below. There exists a constant $C_{\mathrm{cls}} > 0$ depending only on the dimension d and the shape-regularity parameter γ_0 such that

$$\langle r, w - I_{V_J} w\rangle \le C_{\mathrm{cls}} \eta_J \|\nabla w\|, \quad \forall w \in H_0^1(\Omega). \tag{2.8}$$

Note that if v_J is equal to the Galerkin solution u_J, the associated residual $r = r(u_J)$ satisfies the Galerkin orthogonality on the finest level, i.e.,

$$\langle r, w_J\rangle = 0, \quad \forall w_J \in V_J. \tag{2.9}$$

Then

$$\|\nabla(u-u_J)\|^2 = \langle r, (u-u_J) - I_{V_J}(u-u_J)\rangle,$$

and using (2.8) for $w = u - u_J$ yields the standard bound on the discretization error

$$\|\nabla(u-u_J)\| \le C_{\mathrm{cls}} \eta_J(u_J).$$

2.3.1 Estimates of Becker, Johnson & Rannacher

The following derivation is motivated by [4] and uses decomposition of the error via quasi-interpolation operators. Considering the residual equation (2.5) and writing the error $e = u - v_J$ as

$$e = e - I_{V_J} e + \sum_{j=1}^{J} \left(I_{V_j} e - I_{V_{j-1}} e \right) + I_{V_0} e, \tag{2.10}$$

yields

$$\|\nabla e\|^2 = \langle r, e\rangle = \langle r, e - I_{V_J} e\rangle + \sum_{j=1}^{J} \langle r, I_{V_j} e - I_{V_{j-1}} e\rangle + \langle r, I_{V_0} e\rangle. \tag{2.11}$$

The first term on the right-hand side of (2.11) can be bounded using (2.8) as

$$\langle r, e - I_{V_J} e\rangle \le C_{\mathrm{cls}} \eta_J \|\nabla e\|. \tag{2.12}$$

The second and the third term on the right-hand side of (2.11) can be rewritten using the scaled residuals (2.6), (2.7) and subsequently bounded as

$$\begin{aligned}\sum_{j=1}^{J} \langle r, I_{V_j} e - I_{V_{j-1}} e\rangle + \langle r, I_{V_0} e\rangle &= \sum_{j=1}^{J} \int_\Omega h_j^{-2} r_j (I_{V_j} e - I_{V_{j-1}} e) + \int_\Omega \nabla r_0 \cdot \nabla I_{V_0} e \\ &\le \sum_{j=1}^{J} \|h_j^{-1} r_j\| \cdot \|h_j^{-1}(I_{V_j} e - I_{V_{j-1}} e)\| + \|\nabla r_0\| \cdot \|\nabla I_{V_0} e\|.\end{aligned} \tag{2.13}$$

Further, using the bound on the difference of the quasi-interpolants on two consecutive levels (Appendix 2.8.2, Theorem 2.6) and the stability of the quasi-interpolation operator on the coarsest level in the $H_0^1(\Omega)$-norm (Appendix 2.8.2, Theorem 2.5, inequality (2.88)), we get

$$\begin{aligned}\sum_{j=1}^{J} \|h_j^{-1} r_j\| \cdot \|h_j^{-1}(I_{V_j} e - I_{V_{j-1}} e)\| + \|\nabla r_0\| \cdot \|\nabla I_{V_0} e\| \\ \leq C_{I,2\mathrm{lvl}} \left(\sum_{j=1}^{J} \|h_j^{-1} r_j\| \right) \|\nabla e\| + \|\nabla r_0\| \cdot C_{I_{V_0},4} \cdot \|\nabla e\|.\end{aligned} \tag{2.14}$$

Combining (2.11)–(2.14) yields

Estimate on total error 1.

$$\|\nabla e\| \leq C_{\mathrm{cls}} \eta_J + C_{I,2\mathrm{lvl}} \sum_{j=1}^{J} \|h_j^{-1} r_j\| + C_{I_{V_0},4} \|\nabla r_0\|. \tag{2.15}$$

In [4] the authors assume that the approximation v_J is computed by a multigrid scheme without post-smoothing and with the exact solution of the problem on the coarsest level. This yields the Galerkin orthogonality on the coarsest level, i.e.,

$$\langle r, w_0 \rangle = 0, \quad \forall w_0 \in V_0. \tag{2.16}$$

As a consequence, their estimate on the energy norm of the error (see [4, Theorem 1]) does not contain the term corresponding to the coarsest level. Another difference between (2.15) and the estimate in [4, Theorem 1] is due to the difference in the definitions of the scaled/discrete residuals described in Section 2.2.3.

Instead of using the bound on the difference of the quasi-interpolants on two consecutive levels (Appendix 2.8.2, Theorem 2.6), and the stability of the quasi-interpolation operator on the coarsest level (Appendix 2.8.2, Theorem 2.5, inequality (2.88)), we can use the stability of the decomposition of the space $H_0^1(\Omega)$ via the quasi-interpolation operators I_{V_j} (Appendix 2.8.2, Theorem 2.9). In particular,

$$\begin{aligned}&\sum_{j=1}^{J} \|h_j^{-1} r_j\| \cdot \|h_j^{-1}(I_{V_j} e - I_{V_{j-1}} e)\| + \|\nabla r_0\| \cdot \|\nabla I_{V_0} e\| \\ &\quad \leq \left(\sum_{j=1}^{J} \|h_j^{-1} r_j\|^2 + \|\nabla r_0\|^2 \right)^{\frac{1}{2}} \left(\sum_{j=1}^{J} \|h_j^{-1}(I_{V_j} e - I_{V_{j-1}} e)\|^2 + \|\nabla I_{V_0} e\|^2 \right)^{\frac{1}{2}} \\ &\qquad \leq \left(\sum_{j=1}^{J} \|h_j^{-1} r_j\|^2 + \|\nabla r_0\|^2 \right)^{\frac{1}{2}} C_{S,I_V}{}^{\frac{1}{2}} \|\nabla e\|.\end{aligned}$$

Combining this inequality with (2.11)–(2.13) and using $\sqrt{a} + \sqrt{b} \leq \sqrt{2}\sqrt{a+b}$ leads to

Estimate on total error 2.

$$\|\nabla e\| \leq \sqrt{2} \left(C_{\mathrm{cls}}^2 \eta_J^2 + C_{S,I_V} \left(\sum_{j=1}^{J} \|h_j^{-1} r_j\|^2 + \|\nabla r_0\|^2 \right) \right)^{\frac{1}{2}}. \tag{2.17}$$

As we will see in Section 2.4, this estimate is efficient with an efficiency constant independent of the number of levels in the hierarchy, i.e., independent of J.

Observing that

$$\begin{aligned} \|\nabla(u_J - v_J)\|^2 &= \int_\Omega f(u_J - v_J) - \int_\Omega \nabla v_J \cdot \nabla(u_J - v_J) = \langle r, u_J - v_J \rangle \\ &= \sum_{j=1}^{J} \langle r, I_{V_j}(u_J - v_J) - I_{V_{j-1}}(u_J - v_J) \rangle + \langle r, I_{V_0}(u_J - v_J) \rangle, \end{aligned}$$

analogous steps can be applied to show that the following "algebraic parts" of the presented estimates provide upper bounds on the algebraic error,

Estimate on algebraic error 1.

$$\|\nabla(u_J - v_J)\| \leq C_{I,\mathrm{2lvl}} \sum_{j=1}^{J} \|h_j^{-1} r_j\| + C_{I_0,3} \|\nabla r_0\|, \tag{2.18}$$

Estimate on algebraic error 2.

$$\|\nabla(u_J - v_J)\| \leq C_{S,I_V}{}^{\frac{1}{2}} \left(\sum_{j=1}^{J} \|h_j^{-1} r_j\|^2 + \|\nabla r_0\|^2 \right)^{\frac{1}{2}}. \tag{2.19}$$

2.3.2 Estimates of Rüde & Huber

The following derivation is motivated by [33, Section 2.6] and [23, Sections 4.1–4.3]. Considering the residual equation (2.5) and decomposing the error using the quasi-interpolation operator on the finest level I_{V_J} yields

$$\|\nabla e\|^2 = \langle r, e - I_{V_J} e \rangle + \langle r, I_{V_J} e \rangle. \tag{2.20}$$

The first term can be bounded as in (2.12). Rewriting the second term using the exact solution of the discrete problem u_J gives

$$\begin{aligned} \langle r, I_{V_J} e \rangle &= \int_\Omega \nabla(u - v_J) \nabla I_{V_J} e \\ &= \int_\Omega \nabla(u - u_J) \nabla I_{V_J} e + \int_\Omega \nabla(u_J - v_J) \nabla I_{V_J} e. \end{aligned}$$

The Galerkin orthogonality on the finest level yields that $\int_\Omega \nabla(u - u_J) \nabla I_{V_J} e$ vanishes and thus

$$\langle r, I_{V_J} e \rangle = \int_\Omega \nabla(u_J - v_J) \nabla I_{V_J} e \leq \|\nabla(u_J - v_J)\| \, \|\nabla I_{V_J} e\|. \tag{2.21}$$

After bounding the term $\|\nabla I_{V_J} e\|$ using the stability property of the quasi-interpolation operator (Appendix 2.8.2, Theorem 2.5, inequality (2.88)) as

$$\|\nabla I_{V_J} e\| \leq C_{I_{V_J},4} \|\nabla e\|, \tag{2.22}$$

it remains to bound the energy norm of the algebraic error $\|\nabla(u_J - v_J)\|$. This can be done using stable splitting of piecewise linear function space, see Appendix 2.8.3, Theorem 2.12 or [33, Theorem 2.6.2]. Consider an arbitrary decomposition of the algebraic error $u_J - v_J$ into the subspaces V_j, i.e.,

$$u_J - v_J = \sum_{j=0}^{J} e_j, \quad e_j \in V_j, \quad j = 0, 1, \ldots, J. \tag{2.23}$$

Then

$$\begin{aligned}
\|\nabla(u_J - v_J)\|^2 &= \langle r, u_J - v_J \rangle = \sum_{j=0}^{J} \langle r, e_j \rangle \\
&\leq \|\nabla r_0\| \cdot \|\nabla e_0\| + \sum_{j=1}^{J} \|h_j^{-1} r_j\| \cdot \|h_j^{-1} e_j\| \\
&\leq \left(\|\nabla r_0\|^2 + \sum_{j=1}^{J} \|h_j^{-1} r_j\|^2 \right)^{\frac{1}{2}} \cdot \left(\|\nabla e_0\|^2 + \sum_{j=1}^{J} \|h_j^{-1} e_j\|^2 \right)^{\frac{1}{2}}.
\end{aligned}$$

Taking the infimum over all possible decompositions (2.23) and using Appendix 2.8.3, Theorem 2.12 yields

Estimate on algebraic error 3.

$$\|\nabla(u_J - v_J)\| \leq C_S^{\frac{1}{2}} \left(\sum_{j=1}^{J} \|h_j^{-1} r_j\|^2 + \|\nabla r_0\|^2 \right)^{\frac{1}{2}}. \tag{2.24}$$

Combining (2.20)–(2.22), the estimate (2.24) on the algebraic error, and using the inequality $\sqrt{a} + \sqrt{b} \leq \sqrt{2}\sqrt{a+b}$, we have

Estimate on total error 3.

$$\|\nabla e\| \leq \sqrt{2} \left(C_{\text{cls}}^2 \eta_J^2 + C_{I_{V_J},4}^2 C_S \left(\sum_{j=0}^{J} \|h_j^{-1} r_j\|^2 + \|\nabla r_0\|^2 \right) \right)^{\frac{1}{2}}. \tag{2.25}$$

2.3.3 Estimates of Harbrecht & Schneider

In this section we present a derivation motivated by [20], which is based on the fact that the basis functions provide a frame in $(H_0^1(\Omega))^{\#}$; see Appendix 2.8.3, Theorem 2.14. Recall that $(H_0^1(\Omega))^{\#}$ is the dual space to $H_0^1(\Omega)$. Using the upper bound for the residual yields

$$\|\nabla e\| = \|r\|_{\left(H_0^1(\Omega)\right)^\#} \leq C_S^{\frac{1}{2}} \overline{C}_B^{\frac{1}{2}} \left(\|\nabla r_0\|^2 + \sum_{j=1}^{+\infty} \sum_{i=1}^{\#\mathcal{K}_j} \frac{\langle r, \phi_i^{(j)} \rangle^2}{\|\nabla \phi_i^{(j)}\|^2} \right)^{\frac{1}{2}}. \tag{2.26}$$

Following the derivation in [20, Proof of Theorem 5.1], it can be shown that the sum of the terms corresponding to levels $j > J$, i.e.,

$$\sum_{j=J+1}^{+\infty} \sum_{i=1}^{\#\mathcal{K}_j} \frac{\langle r, \phi_i^{(j)} \rangle^2}{\|\nabla \phi_i^{(j)}\|^2},$$

can be bounded by the classic residual based estimator on the Jth level up to a constant $C_{\mathrm{HS}} > 0$ depending only on d and γ_0, i.e.,

$$\sum_{j=J+1}^{+\infty} \sum_{i=1}^{\#\mathcal{K}_j} \frac{\langle r, \phi_i^{(j)} \rangle^2}{\|\nabla \phi_i^{(j)}\|^2} \leq C_{\mathrm{HS}} \eta_J^2. \tag{2.27}$$

Combining (2.26) and (2.27) yields

Estimate on total error 4.

$$\|\nabla e\| \leq C_S^{\frac{1}{2}} \overline{C}_B^{\frac{1}{2}} \left(C_{\mathrm{HS}} \eta_J^2 + \sum_{j=1}^{J} \sum_{i=1}^{\#\mathcal{K}_j} \frac{\langle r, \phi_i^{(j)} \rangle^2}{\|\nabla \phi_i^{(j)}\|^2} + \|\nabla r_0\|^2 \right)^{\frac{1}{2}}. \tag{2.28}$$

Considering the residual r as a functional on V_J, which is possible since $(H_0^1(\Omega))^\# \subset V_J^\#$, one can show that

$$\|\nabla (u_J - v_J)\| = \|r\|_{V_J^\#}.$$

From Appendix 2.8.3, Theorem 2.15, it yields that a part of the total error estimator (2.28) is an upper bound on the algebraic error,

Estimate on algebraic error 4.

$$\|\nabla (u_J - v_J)\| \leq C_S^{\frac{1}{2}} \overline{C}_B^{\frac{1}{2}} \left(\sum_{j=1}^{J} \sum_{i=1}^{\#\mathcal{K}_j} \frac{\langle r, \phi_i^{(j)} \rangle^2}{\|\nabla \phi_i^{(j)}\|^2} + \|\nabla r_0\|^2 \right)^{\frac{1}{2}}. \tag{2.29}$$

2.3.4 New estimate derived using stable splitting

The approach from [20] can also be modified in the following way. Consider the residual equation (2.5) and an arbitrary decomposition of the error $e = \sum_{j=0}^{+\infty} e_j$, $e_j \in V_j$, $j \in \mathbb{N}_0$. Using the definition of scaled residuals (2.6) and (2.7), and the Cauchy–Schwarz inequality, we have

$$\|\nabla e\|^2 = \langle r, e \rangle = \sum_{j=0}^{+\infty} \langle r, e_j \rangle \leq \|\nabla r_0\| \cdot \|\nabla e_0\| + \sum_{j=1}^{J} \|h_j^{-1} r_j\| \cdot \|h_j^{-1} e_j\| + \sum_{j=J+1}^{+\infty} \langle r, e_j \rangle. \tag{2.30}$$

Consider first the terms $\langle r, e_j \rangle$, for $j > J$. Using the definition of r, Green's theorem on elements $K \in \mathcal{T}_J$, and the definition of jump leads to

$$\begin{aligned}\langle r, e_j \rangle &= \int_\Omega f e_j - \int_\Omega \nabla v_J \cdot \nabla e_j \\ &= \sum_{K \in \mathcal{T}_J} \left(\int_K (f + \Delta v_J) e_j - \frac{1}{2} \sum_{E \in \mathcal{E}_{K,\mathrm{int}}} \int_E \left[\nabla v_J \right] e_j \right).\end{aligned}$$

Since v_J is a piecewise affine function, the term Δv_J vanishes. Adding and subtracting f_J yields

$$\langle r, e_j \rangle = \sum_{K \in \mathcal{T}_J} \left(\int_K f_J e_j + \int_K (f - f_J) e_j - \frac{1}{2} \sum_{E \in \mathcal{E}_{K,\mathrm{int}}} \int_E \left[\nabla v_J \right] e_j \right).$$

Inserting $h_K h_K^{-1}$ and $h_K^{1/2} h_K^{-1/2}$, respectively, and using the Cauchy–Schwarz inequality for integrals and subsequently for sums leads to

$$\begin{aligned}\langle r, e_j \rangle \leq &\left(\sum_{K_j \in \mathcal{T}_j} h_{K_j}^2 \| f_J \|_{K_j}^2 + \sum_{K_j \in \mathcal{T}_j} h_{K_j}^2 \| f - f_J \|_{K_j}^2 + \frac{1}{2} \sum_{K_j \in \mathcal{T}_j} h_{K_j} \sum_{E \in \mathcal{E}_{K_j,\mathrm{int}}} \| \left[\nabla v_J \right] \|_E^2 \right)^{\frac{1}{2}} \\ &\cdot \left(2 \sum_{K_j \in \mathcal{T}_j} h_{K_j}^{-2} \| e_j \|_{K_j}^2 + \sum_{K_j \in \mathcal{T}_j} h_{K_j}^{-1} \| e_j \|_{\partial K_j}^2 \right)^{\frac{1}{2}}.\end{aligned}$$

Since $h_j = 2^{J-j} h_J$, we have

$$\begin{aligned}\sum_{K_j \in \mathcal{T}_j} h_{K_j}^2 \| f_J \|_{K_j}^2 &= 2^{2(J-j)} \sum_{K_J \in \mathcal{T}_J} h_{K_J}^2 \| f_J \|_{K_J}^2 = 2^{2(J-j)} (\eta_J^{\mathrm{RHS}})^2, \\ \sum_{K_j \in \mathcal{T}_j} h_{K_j}^2 \| f - f_J \|_{K_j}^2 &= 2^{2(J-j)} \sum_{K_J \in \mathcal{T}_J} h_{K_J}^2 \| f - f_J \|_{K_J}^2 = 2^{2(J-j)} (\mathrm{osc}_J)^2.\end{aligned}$$

Using that ∇v_J is constant on elements in $\mathcal{T}_J$ gives

$$\sum_{K_j \in \mathcal{T}_j} \sum_{E \in \mathcal{E}_{K_j,\mathrm{int}}} \frac{h_{K_j}}{2} \| \left[\nabla v \right] \|_E^2 = 2^{J-j} \sum_{K_J \in \mathcal{T}_J} \frac{h_{K_J}}{2} \sum_{E \in \mathcal{E}_{K_J,\mathrm{int}}} \| \left[\nabla v_J \right] \|_E^2 = 2^{J-j} (\eta_J^{\mathrm{JUMP}})^2.$$

The sum $\sum_{K_j \in \mathcal{T}_j} h_{K_j}^{-1} \| e_j \|_{\partial K_j}^2$ can be bounded using Appendix 2.8.1, Lemma 2.5 as

$$\sum_{K_j \in \mathcal{T}_j} h_{K_j}^{-1} \| e_j \|_{\partial K_j}^2 \leq C_{\mathrm{TI}} \sum_{K_j \in \mathcal{T}_j} h_{K_j}^{-2} \| e_j \|_{K_j}^2.$$

Thus we get

$$\langle r, e_j \rangle \leq \underbrace{(2 + C_{\mathrm{TI}})^{\frac{1}{2}} \left(2^{2(J-j)} \left(\left(\eta_J^{\mathrm{RHS}} \right)^2 + (\mathrm{osc}_J)^2 \right) + 2^{J-j} \left(\eta_J^{\mathrm{JUMP}} \right)^2 \right)^{\frac{1}{2}}}_{\theta_j} \| h_j^{-1} e_j \|. \tag{2.31}$$

Combining (2.30) and (2.31) yields

$$\begin{aligned}\|\nabla e\|^2 &\leq \|\nabla r_0\| \cdot \|\nabla e_0\| + \sum_{j=1}^{J} \|h_j^{-1} r_j\| \cdot \|h_j^{-1} e_j\| + \sum_{j=J+1}^{+\infty} \theta_j \|h_j^{-1} e_j\| \\ &\leq \left(\|\nabla r_0\|^2 + \sum_{j=1}^{J} \|h_j^{-1} r_j\|^2 + \sum_{j=J+1}^{+\infty} \theta_j^2 \right)^{\frac{1}{2}} \\ &\qquad \cdot \left(\|\nabla e_0\|^2 + \sum_{j=1}^{J} \|h_j^{-1} e_j\|^2 + \sum_{j=J+1}^{+\infty} \|h_j^{-1} e_j\|^2 \right)^{\frac{1}{2}}.\end{aligned}$$

Since the decomposition of $e = \sum_{j=0}^{+\infty} e_j$, $e_j \in V_j$, $j \in \mathbb{N}_0$ is arbitrary, the stability of the splitting (Appendix 2.8.3, Theorem 2.10) gives

$$\|\nabla e\| \leq \left(\|\nabla r_0\|^2 + \sum_{j=1}^{J} \|h_j^{-1} r_j\|^2 + \sum_{j=J+1}^{+\infty} \theta_j^2 \right)^{\frac{1}{2}}.$$

Using

$$\sum_{j=J+1}^{+\infty} 2^{2(J-j)} = \frac{1}{3} \quad \text{and} \quad \sum_{j=J+1}^{+\infty} 2^{J-j} = 1,$$

the infinite sum can be bounded as

$$\sum_{j=J+1}^{+\infty} \theta_j^2 \leq C_\theta \eta_J^2,$$

where $C_\theta > 0$ is a constant depending only on the dimension d and the shape-regularity parameter γ_0. This results in the following estimate.

Estimate on total error 5.

$$\|\nabla e\| \leq C_S^{\frac{1}{2}} \left(C_\theta \eta_J^2 + \sum_{j=1}^{J} \|h_j^{-1} r_j\|^2 + \|\nabla r_0\|^2 \right)^{\frac{1}{2}}. \tag{2.32}$$

The fact that $C_S^{\frac{1}{2}} \left(\sum_{j=1}^{J} \|h_j^{-1} r_j\|^2 + \|\nabla r_0\|^2 \right)^{\frac{1}{2}}$ provides an upper bound on the algebraic error has already been shown in Section 2.3.2; see (2.24).

2.4 Efficiency of the estimates

Efficiency of the estimates is described by the constant C_{eff}, such that

$$\text{estimate} \leq C_{\text{eff}} \cdot \|\text{error}\|.$$

Here we in particular focus on whether C_{eff} depends on the number of levels J, quasi-uniformity of the coarsest mesh, and/or on the ratio $h_\Omega / \min_{K \in \mathcal{T}_0} h_K$, which is related to the size of the coarsest-level problem.

2.4.1 Efficiency of the estimates on the algebraic error

We will first discuss the estimates in the form

$$\|\nabla(u_J - v_J)\| \leq C \left(\sum_{j=1}^{J} \|h_j^{-1} r_j\|^2 + \|\nabla r_0\|^2 \right)^{1/2}, \tag{2.33}$$

where either $C = {C_{S,I_V}}^{\frac{1}{2}}$, or $C = {C_S}^{\frac{1}{2}}$ is a constant depending only on the dimension d and the shape-regularity parameter γ_0; see (2.19) and (2.24). Using the definition of scaled residuals (2.6)–(2.7), the Cauchy–Schwarz inequality and the lower bound from Appendix 2.8.3, Theorem 2.12 we have (see also the proof of Theorem 2.6.2 in [33])

$$\begin{aligned}
\sum_{j=1}^{J} \|h_j^{-1} r_j\|^2 + \|\nabla r_0\|^2 &= \sum_{j=0}^{J} \langle r, r_j \rangle \\
&= \int_\Omega \nabla(u_J - v_J) \cdot \nabla \left(\sum_{j=0}^{J} r_j \right) \\
&\leq \|\nabla(u_J - v_J)\| \cdot \left\| \nabla \left(\sum_{j=0}^{J} r_j \right) \right\| \\
&\leq \|\nabla(u_J - v_J)\| \cdot {c_S}^{-\frac{1}{2}} \left(\sum_{j=1}^{J} \|h_j^{-2} r_j\|^2 + \|\nabla r_0\|^2 \right)^{1/2}.
\end{aligned}$$

Consequently,

$$\left(\sum_{j=1}^{J} \|h_j^{-1} r_j\|^2 + \|\nabla r_0\|^2 \right)^{1/2} \leq {c_S}^{-\frac{1}{2}} \|\nabla(u_J - v_J)\|, \tag{2.34}$$

i.e., the efficiency constant depends only on the dimension d and the shape-regularity parameter γ_0.

The efficiency of the estimate (2.29)

$$\|\nabla(u_J - v_J)\| \leq C_S^{\frac{1}{2}} \overline{C}_B^{\frac{1}{2}} \left(\sum_{j=1}^{J} \sum_{i=1}^{\#\mathcal{K}_j} \frac{\langle r, \phi_i^{(j)} \rangle^2}{\|\nabla \phi_i^{(j)}\|^2} + \|\nabla r_0\|^2 \right)^{1/2},$$

can be shown by using $\|\nabla(u_J - v_J)\| = \|r\|_{(V_J)^\#}$ and the lower bound from Appendix 2.8.3, Theorem 2.15, giving

$$\left(\sum_{j=1}^{J} \sum_{i=1}^{\#\mathcal{K}_j} \frac{\langle r, \phi_i^{(j)} \rangle^2}{\|\nabla \phi_i^{(j)}\|^2} + \|\nabla r_0\|^2 \right)^{1/2} \leq c_S^{-\frac{1}{2}} \underline{c}_B^{-\frac{1}{2}} \|\nabla(u_J - v_J)\|.$$

The efficiency constant depends only on the dimension d and the shape-regularity parameter γ_0.

Finally, for the estimate (2.18)

$$\|\nabla(u_J - v_J)\| \leq C_{I,2\mathrm{lvl}} \sum_{j=1}^{J} \|h_j^{-1} r_j\| + C_{I_0,3} \|\nabla r_0\|,$$

the equivalence of the Euclidean and ℓ^1-norm,

$$\sum_{j=1}^{J} \|h_j^{-1} r_j\| + \|\nabla r_0\| \leq \sqrt{J} \left(\sum_{j=1}^{J} \|h_j^{-1} r_j\|^2 + \|\nabla r_0\|^2 \right)^{1/2},$$

and (2.34) yield

$$C_{I,2\text{lvl}} \sum_{j=1}^{J} \|h_j^{-1} r_j\| + C_{I_0,3} \|\nabla r_0\| \leq \sqrt{J} \max\{C_{I,2\text{lvl}}, C_{I_0,3}\}\, c_S^{-\frac{1}{2}} \|\nabla(u_J - v_J)\|.$$

This shows the efficiency of (2.18) with $C_{\text{eff}} = \sqrt{J}\widetilde{C}_{\text{eff}}$, where $\widetilde{C}_{\text{eff}}$ depends only on d and γ_0. This result does not necessarily imply a dependence of (2.18) on the number of levels J. However, we present below in Section 2.6.1 a numerical experiment indicating such behaviour.

2.4.2 Efficiency of estimates on total error

The efficiency of the total error estimates follows from the standard result on the efficiency of the classical (one-level) residual-based error estimator. There exists a positive constant $\overline{C}_{\text{eff}}$ depending on the shape regularity of $\mathcal{T}_J$ such that

$$\left(\left(\eta_J^{\text{RHS}}\right)^2 + \left(\eta_J^{\text{JUMP}}\right)^2 \right)^{\frac{1}{2}} \leq \overline{C}_{\text{eff}} \left(\|\nabla e\| + \text{osc}_J \right); \tag{2.35}$$

see, e.g., [42, Section 1.4]. Since $\|\nabla(u_J - v_J)\| \leq \|\nabla e\|$, we can use the efficiency of the algebraic error estimates together with (2.35) to show the efficiency of the estimates on the total error (up to the oscillation term). The resulting efficiency constants depend on the same quantities as the efficiency constants for the algebraic error estimates.

For example, for the estimate (2.25) associated with the algebraic error estimate (2.24),

$$\sqrt{2} \left(C_{\text{cls}}^2 \eta_J^2 + C_{I_{V_J},4}^2 C_S \left(\sum_{j=0}^{J} \|h_j^{-1} r_j\|^2 + \|\nabla r_0\|^2 \right) \right)^{\frac{1}{2}} \leq C \left(\|\nabla e\| + \text{osc}_J \right),$$

with $C^2 = 2(C_{\text{cls}}^2(\overline{C}_{\text{eff}}^2 + 1) + C_{I_{V_J},4}^2 C_S c_S^{-1})$.

2.5 Computability of the error estimates

In this section we address several ways in which the scaled residual norms from the estimates presented in Section 2.3 can be evaluated or bounded. When the scaled residual norms are replaced by their bounds, proving the efficiency of the estimates from Section 2.3 becomes a nontrivial task.

We first state an algebraic formulation of the problem (2.3). Then we present the algebraic representation of the scaled residual norms and some of their bounds from the literature. As the main contribution of this paper, we present in Section 2.5.4 a new approach for approximating the scaled residual norm on the coarsest level using adaptive number of conjugate gradient iterations. This yields total and algebraic error estimates which are provably efficient and robust with respect to the size of the coarsest-level problem.

2.5.1 Algebraic formulation of the problem, residual vectors

Given a basis Φ_J of V_J, the problem (2.3) can be algebraically formulated as finding the vector of coefficients $\mathbf{u}_J \in \mathbb{R}^{\#\mathcal{K}_J}$ of the function u_J in the basis Φ_J such that

$$\mathbf{A}_J \mathbf{u}_J = \mathbf{f}_J, \tag{2.36}$$

where $\mathbf{A}_J$ is the *stiffness matrix* on the finest level J,

$$[\mathbf{A}_J]_{mn} = \int_\Omega \nabla \phi_n^{(J)} \cdot \nabla \phi_m^{(J)},$$

and

$$[\mathbf{f}_J]_m = \int_\Omega f \phi_m^{(J)}, \qquad m, n = 1, \ldots, \#\mathcal{K}_J.$$

Recall that $\#\mathcal{K}_J$ is the cardinality of the basis Φ_J. We use the standard assumption that the right-hand side vector $\mathbf{f}_J$ can be computed exactly using a numerical quadrature. If $\mathbf{f}_J$ is only known approximately, an additional term must be added to the error bounds presented above; see e.g., the discussion in [35, Section 6].

Let v_J be an approximation of the solution u_J of (2.3) and $\mathbf{v}_J$ be the vector of coefficients of v_J in the basis Φ_J. Let r be the residual (2.4) associated with v_J. Consider the residual vectors $\mathbf{r}_j \in \mathbb{R}^{\#\mathcal{K}_j}$, $j = 0, \ldots, J$,

$$[\mathbf{r}_j]_m = \langle r, \phi_m^{(j)} \rangle, \quad m = 1, \ldots, \#\mathcal{K}_j. \tag{2.37}$$

The vector $\mathbf{r}_J$ corresponding to the finest level can be computed as

$$\mathbf{r}_J = \mathbf{f}_J - \mathbf{A}_J \mathbf{v}_J. \tag{2.38}$$

The residual vectors corresponding to coarser levels can be computed from $\mathbf{r}_J$ by restriction. For the prolongation matrices $\mathbf{P}_j^J \in \mathbb{R}^{\#\mathcal{K}_J \times \#\mathcal{K}_j}$ associated with the (nested) finite element spaces V_j, V_J and the bases Φ_j, Φ_J,

$$\mathbf{r}_j = (\mathbf{P}_j^J)^\top \mathbf{r}_J; \tag{2.39}$$

see, e.g., [33, Section 3.2], or [40, Section 2.4] for a more detailed explanation.

2.5.2 The terms associated with fine levels

In this section we present an algebraic form of the term $\|h_j^{-1} r_j\|^2$, $j = 1, \ldots, J$, and several ways of bounding it by computable quantities presented in literature.

Let $\mathbf{c}_j$ be the vector of coefficients of r_j in the basis Φ_j. The definitions (2.37) of $\mathbf{r}_j$ and (2.6) of r_j give

$$[\mathbf{r}_j]_m = \langle r, \phi_m^{(j)} \rangle = \int_\Omega h_j^{-2} r_j \phi_m^{(j)} = \sum_n \int_\Omega h_j^{-2} \, [\mathbf{c}_j]_n \, \phi_n^{(j)} \phi_m^{(j)}, \qquad \forall m = 1, \ldots, \#\mathcal{K}_j. \tag{2.40}$$

Let $\mathbf{M}_j^{\mathrm{S}}$ be a *scaled mass matrix* defined as

$$\left[\mathbf{M}_j^{\mathrm{S}}\right]_{m,n} = \int_\Omega h_j^{-2} \phi_n^{(j)} \phi_m^{(j)}, \qquad \forall m, n = 1, \ldots, \#\mathcal{K}_j.$$

The equation (2.40) can then be expressed as $\mathbf{r}_j = \mathbf{M}_j^{\mathrm{S}}\mathbf{c}_j$ and therefore

$$\|h_j^{-1}r_j\|^2 = \int_\Omega h_j^{-2}\Phi_j\mathbf{c}_j \cdot \Phi_j\mathbf{c}_j = \mathbf{c}_j^*\mathbf{M}_j^{\mathrm{S}}\mathbf{c}_j = \mathbf{r}_j^*(\mathbf{M}_j^{\mathrm{S}})^{-1}\mathbf{r}_j. \tag{2.41}$$

The evaluation of the term (2.41) thus involves the solution of a system with a possibly large matrix $\mathbf{M}_j^{\mathrm{S}}$. Instead of computing this quantity, one can seek a computable upper bound.

Let $\mathbf{D}_j$ be a diagonal matrix $[\mathbf{D}_j]_{m,m} = \int_\Omega \nabla\phi_m^{(j)} \cdot \nabla\phi_m^{(j)}$, $m = 1, \dots, \#\mathcal{K}_j$. The stability of basis functions (Appendix 2.8.3, Lemma 2.12) and (2.123) give

$$c_B\mathbf{r}_j^*\mathbf{D}_j^{-1}\mathbf{r}_j \le \|h_j^{-1}r_j\|^2 = \mathbf{r}_j^*(\mathbf{M}_j^{\mathrm{S}})^{-1}\mathbf{r}_j \le C_B\mathbf{r}_j^*\mathbf{D}_j^{-1}\mathbf{r}_j. \tag{2.42}$$

The upper bound in (2.42) is used in [33, 23] to bound the algebraic error as

$$\begin{aligned}\|\nabla(u_J - v_J)\| &\le C_S^{\frac{1}{2}}\left(C_B\sum_{j=1}^{J}\mathbf{r}_j^*\mathbf{D}_j^{-1}\mathbf{r}_j + \|\nabla r_0\|^2\right)^{\frac{1}{2}} \\ &\le C_S^{\frac{1}{2}}\overline{C}_B^{\frac{1}{2}}\left(\sum_{j=1}^{J}\mathbf{r}_j^*\mathbf{D}_j^{-1}\mathbf{r}_j + \|\nabla r_0\|^2\right)^{\frac{1}{2}},\end{aligned} \tag{2.43}$$

where $\overline{C}_B = \max\{1, C_B\}$. For $\bar{c}_B = \min\{1, c_B\}$, using the lower bound in (2.42) and (2.34)

$$\left(\sum_{j=1}^{J}\mathbf{r}_j^*\mathbf{D}_j^{-1}\mathbf{r}_j + \|\nabla r_0\|^2\right)^{\frac{1}{2}} \le \left(c_B^{-1}\sum_{j=1}^{J}\|h_j^{-1}r_j\|^2 + \|\nabla r_0\|^2\right)^{\frac{1}{2}} \le \bar{c}_B^{-\frac{1}{2}}c_S^{-\frac{1}{2}}\|\nabla(u_J - v_J)\|, \tag{2.44}$$

which proves the efficiency of the bound (2.43). Recall that c_B, C_B, c_S, and C_S only depend on d and γ_0.

Noting that

$$\mathbf{r}_j^*\mathbf{D}_j^{-1}\mathbf{r}_j = \sum_{i=1}^{\#\mathcal{K}_j}\frac{\langle r, \phi_i^{(j)}\rangle^2}{\|\nabla\phi_i^{(j)}\|^2}, \tag{2.45}$$

we see that the algebraic error bounds (2.29) and (2.43) are identical.

The term (2.41) can also be bounded using other techniques, e.g., using the so-called mass lumping (suggested in [4, Section 4]) or the multigrid smoothing routines (see the discussion in [23, Section 4.5.2]). By using these techniques, however, we introduce another unknown constant into the overall estimate and possibly weaken its efficiency.

In order to get a fully computable bound on (2.41) (i.e., a bound without any unknown constant) and to avoid solving an algebraic problem with a large matrix, we can proceed similarly to [30]. Define $\bar{r}_j \in L^2(\Omega)$ to be (discontinuous) piecewise affine functions on $\mathcal{T}_j$ such that for all $K \in \mathcal{T}_j$,

$$\int_K h_j^{-2}\bar{r}_j\phi_m^{(j)} = [\mathbf{r}_j]_m \cdot \frac{1}{\#\left\{\bar{K} \in \mathcal{T}_j; m \text{ is vertex of } \bar{K}\right\}} =: [\mathbf{r}_{j,K}]_m \quad \forall\phi_m^{(j)}. \tag{2.46}$$

This ensures that

$$\int_\Omega h_j^{-2}\bar{r}_j\phi_m^{(j)} = [\mathbf{r}_j]_m = \langle r, \phi_m^{(j)}\rangle. \tag{2.47}$$

Since $\bar{r}_j$ is piecewise affine on elements, the norms $\|h_j^{-1}\bar{r}_j\|_K^2$ can be computed using the solutions of systems with local scaled mass matrices, i.e., $\|h_j^{-1}\bar{r}_j\|_K^2 = \mathbf{r}_{j,K}^*(\mathbf{M}_{j,K}^{\mathrm{S}})^{-1}\mathbf{r}_{j,K}$, where

$$\left[\mathbf{M}_{j,K}^S\right]_{m,n} = \int_K h_j^{-2}\phi_n^{(j)}\phi_m^{(j)}, \qquad \forall m,n \in \mathcal{N}_K.$$

For the whole term $\|h_j^{-1}r_j\|$ we have

$$\|h_j^{-1}r_j\|^2 \leq \|h_j^{-1}\bar{r}_j\|^2 = \sum_{K\in\mathcal{T}_j} \mathbf{r}_{j,K}^*(\mathbf{M}_{j,K}^{\mathrm{S}})^{-1}\mathbf{r}_{j,K}; \tag{2.48}$$

cf. [30, Eq. (5.9)].

2.5.3 The term associated with the coarsest level

In this section we present the algebraic form of the term $\|\nabla r_0\|$ and several ways of bounding it adapted from literature.

Let $\mathbf{c}_0$ be the vector of coefficients of r_0 in the basis Φ_0. Analogously to (2.41), using the definitions (2.37) of $\mathbf{r}_0$ and (2.7) of r_0, we have

$$[\mathbf{r}_0]_m = \langle r, \phi_m^{(0)}\rangle = \int_\Omega \nabla r_0 \cdot \nabla\phi_m^{(0)} = \sum_n \int_\Omega [\mathbf{c}_0]_n \nabla\phi_n^{(0)}\cdot\nabla\phi_m^{(0)}, \quad \forall m = 1,\ldots,\#\mathcal{K}_0.$$

Let $\mathbf{A}_0$ be the stiffness matrix associated with the coarsest level

$$[\mathbf{A}_0]_{mn} = \int_\Omega \nabla\phi_n^{(0)}\cdot\nabla\phi_m^{(0)}, \qquad m,n = 1,\ldots,\#\mathcal{K}_0.$$

The vector of coefficients $\mathbf{c}_0$ then satisfies $\mathbf{A}_0\mathbf{c}_0 = \mathbf{r}_0$. This leads to

$$\|\nabla r_0\|^2 = \mathbf{c}_0^*\mathbf{A}_0\mathbf{c}_0 = \mathbf{r}_0^*\mathbf{A}_0^{-1}\mathbf{r}_0. \tag{2.49}$$

The evaluation of the term $\|\nabla r_0\|^2$ thus requires solution of the system with the stiffness matrix associated with the coarsest level. For problems where the stiffness matrix is large, this can be too costly and in some settings even unfeasible.

An approximate solution $\widetilde{\mathbf{c}}_0$ of $\mathbf{A}_0\mathbf{c}_0 = \mathbf{r}_0$ computed by the (preconditioned) conjugate gradient method with a fixed number of iterations was used in [23, Section 4.5.2]. The resulting term $\widetilde{\mathbf{c}}_0^*\mathbf{r}_0$ might not be, however, an upper bound on $\|\nabla r_0\|^2$. Therefore, the resulting value may not led to an upper bound on the algebraic nor the total error.

The term (2.49) can also be bounded using a quantity involving only the inverse of a diagonal matrix. Friedrich's inequality (Appendix 2.8.1, Lemma 2.2) imply that

$$\|w_0\|^2 \leq C_F^2 h_\Omega^2 \|\nabla w_0\|^2, \quad \forall w_0 \in V_0. \tag{2.50}$$

Let $\mathbf{M}_0$ be the mass matrix associated with the coarsest level, i.e., $[\mathbf{M}_0]_{mn} = \int_\Omega \phi_n^{(0)}\phi_m^{(0)}$, $m,n = 1,\ldots,\#\mathcal{K}_0$. The inequality (2.50) can be equivalently expressed algebraically as

$$\mathbf{w}^*\mathbf{M}_0\mathbf{w} \leq C_F^2 h_\Omega^2 \mathbf{w}^*\mathbf{A}_0\mathbf{w}, \quad \forall \mathbf{w} \in \mathbb{R}^{\#\mathcal{K}_0}.$$

Since $\mathbf{A}_0$ and $\mathbf{M}_0$ are symmetric positive definite matrices we have

$$\mathbf{w}^*\mathbf{A}_0^{-1}\mathbf{w} \leq C_F^2 h_\Omega^2 \mathbf{w}^*\mathbf{M}_0^{-1}\mathbf{w}, \quad \forall \mathbf{w} \in \mathbb{R}^{\#\mathcal{K}_0}.$$

This bound can be a possibly large overestimation; see the discussion in [30, Sects. 3.1 and 5.2]. Define the diagonal matrix $\mathbf{D}_0$ as $[\mathbf{D}_0]_{m,m} = \int_\Omega \nabla\phi_m^{(0)} \cdot \nabla\phi_m^{(0)}$, $m = 1, \ldots, \#\mathcal{K}_0$. The term on the right-hand side can be further simplified using the spectral equivalence of the mass matrix $\mathbf{M}_0$ with $\mathbf{D}_0$; see inequality (2.124) in Appendix 2.8.3. Altogether we have

$$\|\nabla r_0\|^2 = \mathbf{r}_0^* \mathbf{A}_0^{-1} \mathbf{r}_0 \leq C_F^2 h_\Omega^2 \mathbf{r}_0^* \mathbf{M}_0^{-1} \mathbf{r}_0 \leq C_M C_F^2 \frac{h_\Omega^2}{\min_{K \in \mathcal{T}_0} h_K^2} \mathbf{r}_0^* \mathbf{D}_0^{-1} \mathbf{r}_0. \tag{2.51}$$

As for the efficiency, this allows to prove, using the Inverse inequality (Appendix 2.8.1, Lemma 2.4) and (2.124),

$$\mathbf{r}_0^* \mathbf{D}_0^{-1} \mathbf{r}_0 \leq \frac{C_{\mathrm{INV}}^2}{c_M} \frac{\max_{K \in \mathcal{T}_0} h_K^2}{\min_{K \in \mathcal{T}_0} h_K^2} \|\nabla r_0\|^2,$$

which indicates that bound (2.51) may not be robust with respect to

$$\frac{h_\Omega^2}{\min_{K \in \mathcal{T}_0} h_K^2}.$$

Numerical experiments in Section 2.6 illustrate this deficiency.

2.5.4 Adaptive approximation of the coarsest-level term

In order to overcome the deficiencies described above, we now present a new approach for approximating the term (2.49). It consists of applying the preconditioned conjugate gradient method (PCG) to $\mathbf{A}_0\mathbf{c}_0 = \mathbf{r}_0$ and using lower and upper bounds on the error in PCG. A number of PCG iterations is determined adaptively in order to ensure the efficiency of the resulting bounds on total and algebraic errors.

Let $\mathbf{c}_0^{(i)}$ be the approximation of $\mathbf{c}_0 = \mathbf{A}_0^{-1}\mathbf{r}_0$ computed at the i-th iteration of PCG a with zero initial guess. Let $\|\cdot\|_{\mathbf{A}_0}$ be the norm generated by the matrix $\mathbf{A}_0$, i.e., $\|\mathbf{v}\|_{\mathbf{A}_0}^2 = \mathbf{v}^*\mathbf{A}_0\mathbf{v}$, for all $\mathbf{v} \in \mathbb{R}^{\#\mathcal{K}_0}$. The term (2.49) can be expressed using the following decomposition

$$\mathbf{c}_0^* \mathbf{A}_0 \mathbf{c}_0 = \underbrace{\sum_{m=0}^{i-1} \|\mathbf{c}_0^{(m+1)} - \mathbf{c}_0^{(m)}\|_{\mathbf{A}_0}^2}_{=:\mu_i^2} + \|\mathbf{c}_0 - \mathbf{c}_0^{(i)}\|_{\mathbf{A}_0}^2, \tag{2.52}$$

which is a consequence of the *local orthogonality* in PCG. This formula was already shown for CG in the seminal paper [22, Theorem 6:1, Eq. (6:2)]. The terms $\|\mathbf{c}_0^{(m)} - \mathbf{c}_0^{(m+1)}\|_{\mathbf{A}_0}^2$ can be computed at a minimal cost from the scalars available during the computations.

It is crucial to note that the local orthogonality is in CG computations preserved proportionally to the machine precision. Therefore, (2.52) is valid, up to a negligible error, also in finite-precision computations; see the derivation and proofs in [37] (resp. in [36] for the preconditioned variant).

Let ζ_i^2 be an upper bound on the squared $\mathbf{A}_0$-norm of the error in the PCG computation, i.e., on $\|\mathbf{c}_0 - \mathbf{c}_0^{(i)}\|_{\mathbf{A}_0}^2$; see, e.g., [18], [26] and the references therein[1].

[1] Strictly speaking, numerical stability of the upper bounds to the A-norm of the error in CG computations has not been rigorously proved. Well-justified heuristics supported by numerical experiments however suggest their validity also in finite-precision computations; see [18], [26].

The derivation of such a bound is based on the interpretation of CG as a procedure for computing the Gauss quadrature approximation to a Riemann–Stieltjes integral and typically requires a lower bound on the smallest eigenvalue of $\mathbf{A}_0$. A simple lower bound can be derived using [17, Theorem 3], as

$$\lambda_{\min}(\mathbf{A}_0) \geq C_F^{-2} h_\Omega^{-2} \min_{K \in \mathcal{T}_0} \lambda_{\min}(\mathbf{M}_{0,K}),$$

where C_F is the constant from Friedrich's inequality (Appendix 2.8.1, Lemma 2.2) and $\mathbf{M}_{0,K}$ is the local mass matrix corresponding to $K \in \mathcal{T}_0$. If an upper bound ζ_i is not available, the $\mathbf{A}_0$-norm of the error $\|\mathbf{c}_0 - \mathbf{c}_0^{(i)}\|_{\mathbf{A}_0}^2$ can be bounded using the ideas presented in Section 2.5.3; see also [30, Section 3.2].

The approach then consists of running PCG for the coarsest problem until

$$\zeta_i^2 \leq \theta \left(\sum_{j=1}^{J} \mathbf{r}_j^* \mathbf{D}_j^{-1} \mathbf{r}_j + \mu_i^2 \right), \tag{2.53}$$

where $\theta > 0$ is a chosen parameter. Then we consider the bound

$$\mathbf{r}_0^* \mathbf{A}_0^{-1} \mathbf{r}_0 \leq \mu_i^2 + \zeta_i^2, \tag{2.54}$$

which can be combined, e.g., with (2.43) to get an upper bound on the algebraic error

$$\|\nabla(u_J - v_J)\| \leq C_S^{\frac{1}{2}} \overline{C}_B^{\frac{1}{2}} \left(\sum_{j=1}^{J} \mathbf{r}_j^* \mathbf{D}_j^{-1} \mathbf{r}_j + \mu_i^2 + \zeta_i^2 \right)^{\frac{1}{2}}. \tag{2.55}$$

The criterion (2.53) guarantees that

$$\mathbf{r}_0^* \mathbf{A}_0^{-1} \mathbf{r}_0 \leq \mu_i^2 + \zeta_i^2 \leq \theta \sum_{j=1}^{J} \mathbf{r}_j^* \mathbf{D}_j^{-1} \mathbf{r}_j + (1+\theta)\mu_i^2, \tag{2.56}$$

which allows us to prove the efficiency of (2.55). Indeed, using (2.56), $\mu_i^2 \leq \|\nabla r_0\|^2$ (see (2.52)), and (2.44)

$$\begin{aligned} \left(\sum_{j=1}^{J} \mathbf{r}_j^* \mathbf{D}_j^{-1} \mathbf{r}_j + \mu_i^2 + \zeta_i^2 \right)^{\frac{1}{2}} &\leq (1+\theta)^{\frac{1}{2}} \left(\sum_{j=1}^{J} \mathbf{r}_j^* \mathbf{D}_j^{-1} \mathbf{r}_j + \|\nabla r_0\|^2 \right)^{\frac{1}{2}} \\ &\leq (1+\theta)^{\frac{1}{2}} c_S^{-\frac{1}{2}} c_B^{-\frac{1}{2}} \|\nabla(u_J - v_J)\|. \end{aligned}$$

The proposed strategy follows the ideas of [30, Section 3.2]. In principle, the possible overestimation in $\|\mathbf{c}_0 - \mathbf{c}_0^{(i)}\|_{\mathbf{A}_0}^2 \leq \zeta_i^2$ is controlled by (2.53) and it is compensated for within the procedure by performing extra iterations. This allows us to prove the efficiency even if the estimate ζ_i is not very tight. However, in such case the number of extra iterations might be quite large; see [30, Section 7.1].

Finally, we note that $\mathbf{r}_j^* \mathbf{D}_j^{-1} \mathbf{r}_j$ in (2.53) can be replaced by any (efficient) bound on $\|h_j^{-1} r_j\|^2$. Then the algebraic error bound (2.55) should be changed accordingly, replacing $\mathbf{r}_j^* \mathbf{D}_j^{-1} \mathbf{r}_j$ and $\overline{C}_B$.

2.6 Numerical experiments

The experiments focus on the efficiency of the error estimates on the algebraic error. In particular, we consider the estimate

$$C\Big(\sum_{j=1}^{J} \mathbf{r}_j^* \mathbf{D}_j^{-1} \mathbf{r}_j + \mathbf{r}_0^* \mathbf{A}_0^{-1} \mathbf{r}_0 \Big)^{\frac{1}{2}} \tag{2.57}$$

and variants where $\mathbf{r}_0^* \mathbf{A}_0^{-1} \mathbf{r}_0 = \|\nabla r_0\|^2$ is replaced by computable approximations. This prototype covers most of the algebraic error estimates from Section 2.3, where the scaled residual norms $\|h_j^{-1} r_j\|^2$ on the fine levels are efficiently approximated by $\mathbf{r}_j^* \mathbf{D}_j^{-1} \mathbf{r}_j$ using (2.42). As shown in the previous sections, approximating the coarsest-level term $\|\nabla r_0\|^2$ while preserving the efficiency is more subtle.

For the experiments, we consider a 3D Poisson problem on a unit cube, $\Omega = (0,1)^3$, with the exact solution

$$u(x,y,z) = x(x-1)y(y-1)z(z-1)e^{-100((x-\frac{1}{2})^2+(y-\frac{1}{2})^2+(z-\frac{1}{2})^2)}.$$

The problem is discretized by the standard Galerkin finite element method with piecewise affine polynomials on a sequence of six uniformly refined meshes with the same shape regularity (2.2). The associated matrices are generated in the FE software FEniCS [2, 24], and the computations are done in MATLAB 2023a. The codes for the experiments are available from `https://github.com/vacek-petr/inMLEstimate`.

Given the mesh $\mathcal{T}_J$ (the finest mesh varies in the experiments), the associated Galerkin solution u_J of (2.3) is for the purpose of the evaluation of the efficiency of the estimates considered (with a negligible inaccuracy) as a result of using the MATLAB backslash, or, for very large problems, using the multigrid V-cycle with an excessive number (30) of V-cycle repetitions. The approximation v_J to u_J is given by a multigrid solver starting with a zero approximation and repeating V-cycles until the relative energy norm of the (algebraic) error $u_J - v_J$ drops below 10^{-11}. Each multigrid V-cycle uses 3 pre and 3 post Gauss–Seidel smoothing iterations. The problem on the coarsest level is solved using CG where the stopping criterion is based on the relative residual with the tolerance 10^{-1}. In order to monitor the efficiency for varying algebraic error, we will also plot below intermediate results after completing each multigrid V-cycle.

We observed very similar results also for a set of two-dimensional problems and a 3D problem with a more complicated geometry. These experiments can be found in the repository `https://github.com/vacek-petr/inMLEstimate` where also the data and codes are available.

2.6.1 Robustness with respect to the number of levels

The first experiment studies the efficiency of the estimates while varying the number of levels $J = 1, 2, \ldots, 5$ in the hierarchy. We fix the size of the problem on the coarsest-level and, consequently, the size of the finest problem grows; see Table 2.1.

coarsest-level DoFs	finest-level DoFs
125	1 331
125	12 167
125	103 823
125	857 375
125	6 967 871

Table 2.1 Size of the problems for the experiment in Section 2.6.1.

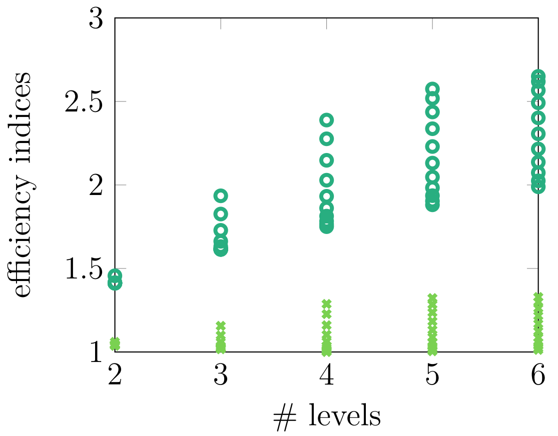


Figure 2.1 Efficiency indices I_1 (✖) and I_2 (⊙), (2.58) and (2.59), for varying number of levels J. We plot the efficiency for approximations v_J and for the associated intermediate results after each V-cycle; each corresponds to a single mark.

For the prototype estimate (2.57), the efficiency index

$$I_1 = \frac{C_{\mathrm{numexp}} \left(\sum_{j=1}^{J} \mathbf{r}_j^* \mathbf{D}_j^{-1} \mathbf{r}_j + \mathbf{r}_0^* \mathbf{A}_0^{-1} \mathbf{r}_0 \right)^{\frac{1}{2}}}{\|\nabla(u_J - v_J)\|}, \tag{2.58}$$

is evaluated for every J, v_J, and also for intermediate results after each V-cycle. The factor C_{numexp} accounts for $C_S^{\frac{1}{2}} \overline{C}_B^{\frac{1}{2}}$; see (2.43). For the purpose of the experiment, it is chosen as the minimal value such that the efficiency indices I_1 are for all J and in all V-cycle repetitions above or equal to one; $C_{\mathrm{numexp}} = 1.28$. In order to examine the difference, we also evaluate the index

$$I_2 = \frac{C_{\mathrm{numexp}} \left(\sum_{j=1}^{J} \left(\mathbf{r}_j^* \mathbf{D}_j^{-1} \mathbf{r}_j \right)^{\frac{1}{2}} + \left(\mathbf{r}_0^* \mathbf{A}_0^{-1} \mathbf{r}_0 \right)^{\frac{1}{2}} \right)}{\|\nabla(u_J - v_J)\|}, \tag{2.59}$$

which corresponds to the algebraic error bound (2.18).

The index I_1 (2.58) corresponds to the estimate (2.43) that is proved to be robust with respect to the number of levels J and consequently also to the size of the finest problem; see Section 2.4.1 or the original papers [33, 20]. This is what the experiment confirms; see Figure 2.1. Contrary to that, I_2 (2.59) deteriorates with increasing J. This is with alignment with the discussion at the end of of Section 2.4.1, where we proved the efficiency of the estimate with a factor depending on $\sqrt{J}$.

2.6.2 Robustness with respect to the size of the coarsest-level problem

The second experiment describes the effect of the size of the coarsest-level problem on the efficiency of the estimates. We fix the number of levels to two ($J = 1$) and vary the coarse and fine level problems; see Table 2.2. For the approximation v_1 and intermediate results computed after each V-cycle, we plot the efficiency index

$$I_3 = \frac{C_{\mathrm{numexp}}\left(\mathbf{r}_1^*\mathbf{D}_1^{-1}\mathbf{r}_1 + \eta\right)^{\frac{1}{2}}}{\|\nabla(u_1 - v_1)\|}, \tag{2.60}$$

where η denotes the following approximations to $\mathbf{r}_0^*\mathbf{A}_0^{-1}\mathbf{r}_0 = \|\nabla r_0\|^2$:

(i) $\eta = \mathbf{r}_0^*\overline{\mathbf{c}}_0$, where $\overline{\mathbf{c}}_0$ is computed using a direct solver for $\mathbf{A}_0\mathbf{c}_0 = \mathbf{r}_0$;

(ii) $\eta = \mathbf{r}_0^*\tilde{\mathbf{c}}_0$, where $\tilde{\mathbf{c}}_0$ is computed by 4 iterations of CG on $\mathbf{A}_0\mathbf{c}_0 = \mathbf{r}_0$ with a zero initial approximation;

(iii) $\eta = \dfrac{h_\Omega^2}{\min_{K\in\mathcal{T}_0} h_K^2}\mathbf{r}_0^*\mathbf{D}_0^{-1}\mathbf{r}_0$;

(iv) $\eta = \mu_i^2 + \zeta_i^2$; see (2.54) and the adaptive approach from Section 2.5.4 using PCG. Here ζ_i^2 is the upper bound on the $\mathbf{A}_0$-norm in PCG from [26, second inequality in (3.5) with updating formula for a coefficient (3.3)]. For evaluating ζ_i^2, an estimate of the smallest eigenvalue of $\mathbf{A}_0$ is computed by the MATLAB `eigs` function for the first four problems and extrapolated for the largest problem. In (2.53) we set $\theta = 0.1$.

coarsest-level DoFs	finest-level DoFs	$h_\Omega^2/\min_{K\in\mathcal{T}_0} h_K^2$
125	1 331	36
1 331	12 167	144
12 167	103 823	576
103 823	857 375	2 304
857 375	6 967 871	9 216

Table 2.2 Size of the problems for the experiment in Section 2.6.2. The table also gives the squared ratios of the diameter of the computational domain and the coarsest-level meshsize.

The factor $C_{\mathrm{numexp}} = 1.28$ accounts for $C_S^{\frac{1}{2}}\overline{C}_B^{\frac{1}{2}}$; and was set as a minimal value such that the efficiency index (2.60) for the variant (i) with the direct solver is above or equal to one. The results are plotted in Figure 2.2.

The variant (i), where the coarsest-level term is computed using a direct solver, exhibits only a very mild increase of the efficiency index I_3 (2.60). Recall, however, that using a direct solver is for large problems in practice unfeasible.

The variant (ii), which uses four iterations of CG to approximate the term on the coarsest level, provides no longer an upper bound on the algebraic error. It is not surprising that a fixed number of CG iterations is not sufficient for problems

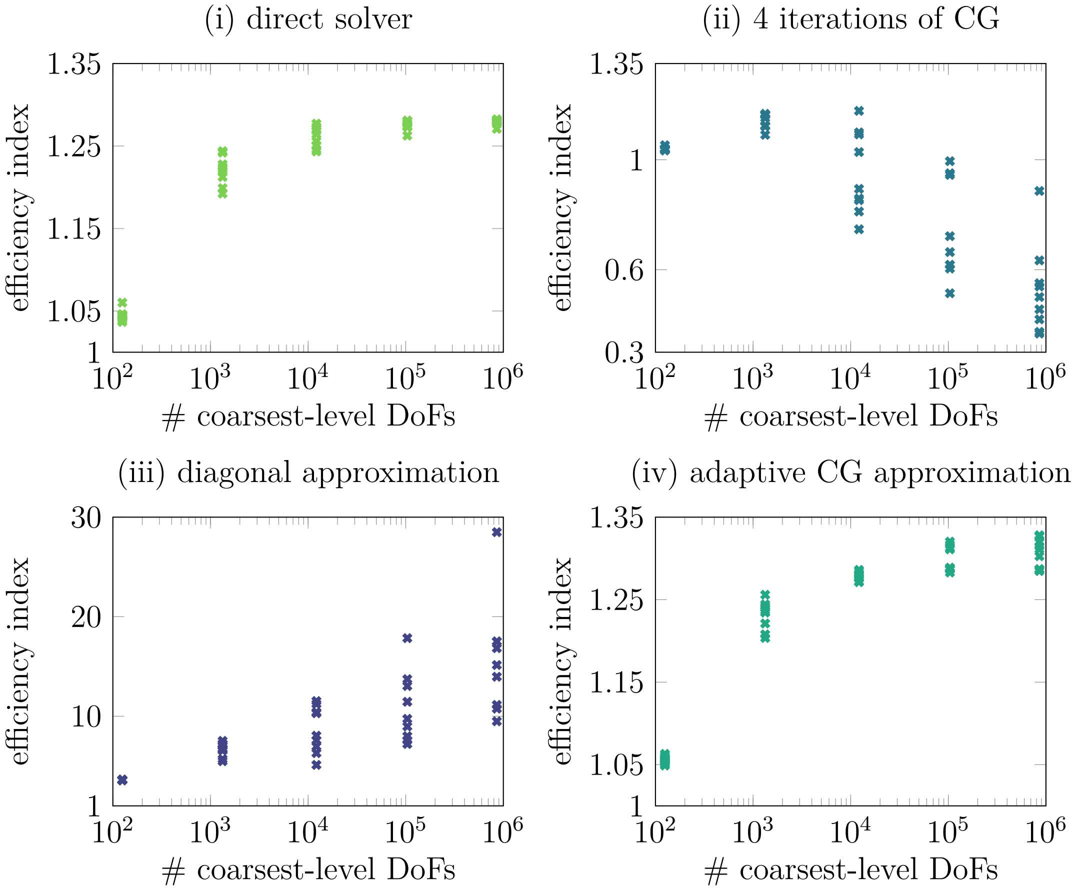


Figure 2.2 Efficiency indices I_3 (2.60) for the experiment in Section 2.6.2. The estimates differ in the way of approximating the coarsest-level term $\|\nabla r_0\|^2 = \mathbf{r}_0^* \mathbf{A}_0^{-1} \mathbf{r}_0$. This term is: computed by a direct solver for the coarsest problem (i), approximated using four iterations of the CG solver (ii), approximated by replacing the stiffness matrix by its scaled diagonal approximation (iii), determined using the adaptive CG approximation (iv).

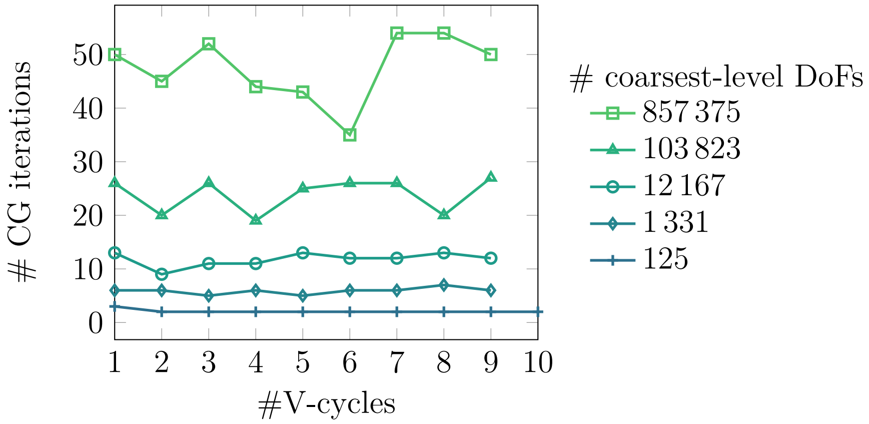


Figure 2.3 Number of CG iterations determined by the adaptive approach described in Section 2.5.4, which is used to estimate the residual norm $\|\nabla r_0\|$ associated with the coarsest level. The horizontal axis indicates the number of V-cycles used in computing the approximation v_J.

with increasing size. In the newly proposed adaptive approach, the number of CG iteration varies and it is determined automatically.

For the variant (iii), where the stiffness matrix on the coarsest level is replaced by its scaled diagonal (see (2.51)), the efficiency indices deteriorate with the increasing ratio $h_\Omega^2/\min_{K\in\mathcal{T}_0} h_K^2$; see Table 2.2. The experiment illustrates that the estimate is not robust with respect to this ratio; see the discussion at the end of Section 2.5.3.

When the term $\|\nabla r_0\|$ is approximated using the adaptive computation (iv) proposed in Section 2.5.4, the efficiency behaves as in the case (i). Unlike in (i), the approximation in (iv) is computable even for very large problems on the coarsest-level. The adaptively chosen number of CG iterations performed within the new procedure is plotted in Figure 2.3.

2.7 Conclusions

This paper presents residual-based a posteriori error estimates on total and algebraic errors in multilevel frameworks inspired by several derivations from the literature. It starts with algebraic error estimates containing sum of the (scaled) residual norms over the levels, including the coarsest one. Total error estimates incorporate additionally the standard residual-based estimator evaluated on the finest level. Efficiency and robustness with respect to the number of levels and the size of the algebraic problem on the coarsest level were for several estimates of this type proved in literature. However, the estimates containing residual norms are not easily computable and applicable in practice.

Approximation of the scaled residual norms, i.e., the terms $\mathbf{r}_j^*\mathbf{X}_j^{-1}\mathbf{r}_j$, where $\mathbf{r}_j$ is the algebraic residual associated with the level j on all but the coarsest level does not represent a significant difficulty. Except for the coarsest level, $\mathbf{X}_j$ is the scaled mass matrix denoted in the paper as $\mathbf{M}_j^{\mathrm{S}}$ and the term $\mathbf{r}_j^*(\mathbf{M}_j^{\mathrm{S}})^{-1}\mathbf{r}_j$ can be bounded from above by the simpler term $\mathbf{r}_j^*(\mathbf{D}_j)^{-1}\mathbf{r}_j$, where $\mathbf{D}_j$ is an appropriate diagonal matrix, without affecting the efficiency and robustness.

Evaluating the residual norm $\|\nabla r_0\|^2 = \mathbf{r}_0^*\mathbf{A}_0^{-1}\mathbf{r}_0$ associated with the coarsest level, where $\mathbf{A}_0$ is the stiffness matrix, is more subtle. When using bounds or techniques to approximate $\mathbf{r}_0^*\mathbf{A}_0^{-1}\mathbf{r}_0$ presented in the literature, the resulting (multilevel) estimates on the total and algebraic errors are no longer guaranteed to be independent of the size of the coarsest-level problem. This behaviour is illustrated by numerical experiments.

The approach proposed in this paper approximates the coarsest-level term $\|\nabla r_0\|^2$ using the preconditioned conjugate gradient iterates. A number of PCG iterations is determined adaptively such that the efficiency of the bound does not deteriorate with increasing size of the coarsest-level problem, and the efficiency and robustness of the multilevel error estimates is preserved. Numerical results support the theoretical findings.

The estimates for total and algebraic errors involve some constants that must be approximately determined, which involves heuristics. For residual-based error estimates, the constants can be determined for smaller problems with the same or analogous geometry where an approximation with very small algebraic error can be computed; see, e.g., the discussion in [4, Section 7]. Since the new result in Section 2.5.4 proves the robustness of the adaptive estimate with respect to the size of the coarsest-level problem, it provides a justification for extrapolating the estimated values of the constants from smaller to larger problems.

In view of a recent trend on using multiple precision in multigrid algorithms (see, e.g., [25, 38]), it is worth considering extension of the presented results to include effects of inexact (limited-precision) operations. This will require substantial further analysis. We plan to address this topic in the future.

2.8 Appendix

2.8.1 Auxiliary results from the theory of PDEs and FEM

The following results are standard in PDE and FEM analysis. They are presented in various forms and sometimes with different names. We provide them in forms suitable for our development, with some standard references where the proofs can be found.

Lemma 2.1 (Bramble–Hilbert lemma). *There exists a constant $C_{\mathrm{BH}}(\mathcal{T}) > 0$ depending only on d and $\gamma_{\mathcal{T}}$ such that for all $K \in \mathcal{T}$*

$$\inf_{c\in\mathbb{R}} \|w - c\|_{\omega_K} \leq C_{\mathrm{BH}}(\mathcal{T}) h_K \|\nabla w\|_{\omega_K} \qquad \forall w \in H^1(\omega_K), \tag{2.61}$$

$$\inf_{p\in\mathbb{P}^1(\omega_K)} \|w - p\|_{\omega_K} \leq C_{\mathrm{BH}}(\mathcal{T}) h_K^2 |w|_{H^2(\omega_K)} \qquad \forall w \in H^2(\omega_K). \tag{2.62}$$

For the proof, see, e.g., [34, p. 490] and references therein.

Lemma 2.2 (Friedrich's inequality). *Let $\omega \subset \mathbb{R}^d$ be a bounded domain. There exists a constant $C_F(\omega) > 0$ such that for all $w \in H^1(\omega)$ which have a zero trace on a part of the boundary $\partial\omega$ of nonzero measure*

$$\|w\|_\omega \leq C_F(\omega) h_\omega \|\nabla w\|_\omega. \tag{2.63}$$

When using Friedrich's inequality on patches associated with the elements of the triangulation $\mathcal{T}$, there exists a constant $C_F(\mathcal{T})$ depending only on d and $\gamma_{\mathcal{T}}$ such that for all $K \in \mathcal{T}$

$$C_F(\omega_K) \leq C_F(\mathcal{T});$$

see, e.g., [32, Chapter 18].

Lemma 2.3 (Trace inequality). *There exists a constant $C_{\mathrm{TR}}(\mathcal{T}) > 0$ depending only on d and $\gamma_{\mathcal{T}}$ such that for all $K \in \mathcal{T}$ and all $w \in H^1(K)$*

$$\|w\|^2_{\partial K} \leq C_{\mathrm{TR}}(\mathcal{T}) \left(h_K^{-1} \|w\|_K^2 + h_K \|\nabla w\|_K^2 \right). \tag{2.64}$$

For the proof, see, e.g., [11, Proposition 4.1].

Lemma 2.4 (Inverse inequality). *There exists a constant $C_{\mathrm{INV}}(\mathcal{T}) > 0$ depending only on d and $\gamma_{\mathcal{T}}$ such that for all $K \in \mathcal{T}$ and all $w_{\mathcal{T}} \in S_{\mathcal{T}}$*

$$\|\nabla w_{\mathcal{T}}\|_K \leq C_{\mathrm{INV}}(\mathcal{T}) h_K^{-1} \|w_{\mathcal{T}}\|_K. \tag{2.65}$$

For the proof, see, e.g., [16, Lemma 1.27].

The following lemma is a consequence of Lemma 2.3 and Lemma 2.4.

Lemma 2.5. *There exists a constant $C_{\mathrm{TI}}(\mathcal{T}) > 0$ depending only on d and $\gamma_{\mathcal{T}}$ such that for all $K \in \mathcal{T}$ and all $w_{\mathcal{T}} \in S_{\mathcal{T}}$*

$$\|w_{\mathcal{T}}\|_{\partial K}^2 \leq C_{\mathrm{TI}}(\mathcal{T}) h_K^{-1} \|w_{\mathcal{T}}\|_K^2. \tag{2.66}$$

Proof. Bounding $\|w_{\mathcal{T}}\|_{\partial K}^2$ using the trace inequality yields

$$\|w_{\mathcal{T}}\|_{\partial K}^2 \leq C_{\mathrm{TR}}(\mathcal{T}) \left(h_K^{-1} \|w_{\mathcal{T}}\|_K^2 + h_K \|\nabla w_{\mathcal{T}}\|_K^2 \right).$$

Applying the inverse inequality gives

$$\begin{aligned} \|w_{\mathcal{T}}\|_{\partial K}^2 &\leq C_{\mathrm{TR}}(\mathcal{T}) \left(h_K^{-1} \|w_{\mathcal{T}}\|_K^2 + h_K^{-1} C_{\mathrm{INV}}^2(\mathcal{T}) \|w_{\mathcal{T}}\|_K \right) \\ &= C_{\mathrm{TR}}(\mathcal{T})(1 + C_{\mathrm{INV}}^2(\mathcal{T})) h_K^{-1} \|w_{\mathcal{T}}\|_K^2. \end{aligned}$$

□

2.8.2 Quasi-interpolation operators

A quasi-interpolation operator is not explicitly used in the construction of the estimators but it is a crucial tool for proving the bounds. In this section we present a quasi-interpolation operator as a generalization of nodal interpolation to integrable functions. We consider the quasi-interpolation operator used in [29], which is closely related to the operator from [34]. Other, slightly different quasi-interpolation operators can be found, e.g., in [13, 42, 11]. We list and prove some of the properties of the operator to be used later. The proofs of the properties are based on standard techniques. To keep the text self-contained and formally accurate we provide most of the proofs below.

The results in this section are mostly derived for a single mesh $\mathcal{T}$. We show that the constants only depend on the dimension d and the shape-regularity $\gamma_{\mathcal{T}}$ and therefore we can again use them in the mesh hierarchy with the dependence on d and γ_0.

Nodal interpolation and its generalization

For a node $z \in \mathcal{N}_{\mathcal{T}}$, let $\Psi_z : C(\overline{\Omega}) \to \mathbb{R}$ denote the linear functional evaluation at point z, i.e.,

$$\Psi_z(w) = w(z) \quad \forall w \in C(\overline{\Omega}).$$

The standard nodal interpolation operator $\mathcal{I} : C(\overline{\Omega}) \to S_{\mathcal{T}}$ for continuous functions is defined as (see, e.g., [12, 7])

$$\mathcal{I} w = \sum_{z \in \mathcal{N}_{\mathcal{T}}} \Psi_z(w) \phi_z \quad \forall w \in C(\overline{\Omega}).$$

In order to construct an analogy of the operator $\mathcal{I}$ for functions from $L^1(\Omega)$, the point evaluation is replaced by an appropriate average of the approximated function. We will consider the quasi-interpolation operator defined in [29] and [35].

For a node $z \in \mathcal{N}_{\mathcal{T}}$, let K_z be a fixed element having z as its vertex, i.e., $z \in K_z$. Let $\mathbb{P}^1(K_z)$ denote the space of linear polynomials on K_z and denote by $\widetilde{\Psi}_z$ the restriction of the linear functional Ψ_z to functions from $\mathbb{P}^1(K_z)$. Since

$\mathbb{P}^1(K_z)$ is a finite-dimensional space, the linear functional $\widetilde{\Psi}_z$ is bounded and it therefore belongs to the dual space $(\mathbb{P}^1(K_z))^{\#}$. Considering the space $\mathbb{P}^1(K_z)$ equipped with the L^2-inner product, the Riesz representation theorem (see, e.g., [7, Sect. 2.4]) yields the existence of a function $\psi_z \in \mathbb{P}^1(K_z)$ such that

$$\widetilde{\Psi}_z(w) = w(z) = \int_{K_z} w\psi_z, \quad \forall w \in \mathbb{P}^1(K_z).$$

Since ψ_z is the Riesz representation of the point evaluation at z, it holds for all $z_1, z_2 \in \mathcal{N}_{\mathcal{T}}$ (recall that ϕ_{z_2} is the hat function associated with z_2) that

$$\int_{K_{z_1}} \phi_{z_2}\psi_{z_1} = \phi_{z_2}(z_1) = \begin{cases} 1 & z_1 = z_2, \\ 0 & z_1 \neq z_2. \end{cases} \tag{2.67}$$

We will consider the quasi-interpolation operators defined as follows

$$I_{S_{\mathcal{T}}} : L^1(\Omega) \to S_{\mathcal{T}}, \quad I_{S_{\mathcal{T}}}w = \sum_{z\in\mathcal{N}_{\mathcal{T}}} \left(\int_{K_z} w\psi_z\right)\phi_z, \tag{2.68}$$

$$I_{V_{\mathcal{T}}} : L^1(\Omega) \to V_{\mathcal{T}}, \quad I_{V_{\mathcal{T}}}w = \sum_{z\in\mathcal{K}_{\mathcal{T}}} \left(\int_{K_z} w\psi_z\right)\phi_z. \tag{2.69}$$

These definitions and relation (2.67) imply that $I_{S_{\mathcal{T}}}$ and $I_{V_{\mathcal{T}}}$ are projections onto $S_{\mathcal{T}}$ and $V_{\mathcal{T}}$, respectively. Further, $I_{S_{\mathcal{T}}}$ preserves linear polynomials on Ω and $I_{V_{\mathcal{T}}}$ preserves linear polynomials on ω_K for any element $K \in \mathcal{T}$ whose patch ω_K does not intersect with the boundary of Ω, i.e., $\overline{\omega_K} \cap \partial\Omega = \emptyset$.

Local estimates

We now present local (elementwise) bounds on an interpolant $I_{S_{\mathcal{T}}}w$ and the interpolation error $w - I_{S_{\mathcal{T}}}w$.

Theorem 2.1. *There exist positive constants $\widehat{C}_{I_{S_{\mathcal{T}}},\ell}$, $\ell = 1, 2, 3, 4$, depending only on d and $\gamma_{\mathcal{T}}$ such that for all elements $K \in \mathcal{T}$,*

$$\|I_{S_{\mathcal{T}}}w\|_K \le \widehat{C}_{I_{S_{\mathcal{T}}},1}\|w\|_{\omega_K} \qquad \forall w \in L^2(\omega_K), \tag{2.70}$$

$$\|w - I_{S_{\mathcal{T}}}w\|_K \le \widehat{C}_{I_{S_{\mathcal{T}}},2}h_K\|\nabla w\|_{\omega_K} \qquad \forall w \in H^1(\omega_K), \tag{2.71}$$

$$\|w - I_{S_{\mathcal{T}}}w\|_K \le \widehat{C}_{I_{S_{\mathcal{T}}},3}h_K^2|w|_{H^2(\omega_K)} \qquad \forall w \in H^2(\omega_K), \tag{2.72}$$

$$\|\nabla I_{S_{\mathcal{T}}}w\|_K \le \widehat{C}_{I_{S_{\mathcal{T}}},4}\|\nabla w\|_{\omega_K} \qquad \forall w \in H^1(\omega_K). \tag{2.73}$$

Proof. The steps in the proof are inspired by [29, pp. 17–18] and [34, Sections 3–4].

Using standard affine transformation to a reference element it can be shown that there exists a constant $C_\psi > 0$ depending only on d and $\gamma_{\mathcal{T}}$ such that for all $z \in \mathcal{N}_{\mathcal{T}}$,

$$\|\psi_z\|_{L^\infty(K_z)} \le C_\psi|K_z|^{-1}, \tag{2.74}$$

and that there exists a constant $C_\phi > 0$ depending only on d and $\gamma_{\mathcal{T}}$ such that for all $K \in \mathcal{T}$ and all $z \in \mathcal{K}_K$,

$$\|\nabla\phi_z\|_{L^\infty(K)} \le C_\phi\rho_K^{-1}; \tag{2.75}$$

see, e.g., [34, pp. 487–488].

Using Hölder's inequality and (2.74) we can show that for all $z \in \mathcal{N}_\mathcal{T}$ and all $w \in L^2(K_z)$

$$\left|\int_{K_z} w\psi_z\right|^2 \le \|\psi_z\|^2_{L^\infty(K_z)} \left(\int_{K_z} |w|\right)^2 \le C_\psi^2 |K_z|^{-2}|K_z|\|w\|^2_{K_z} = C_\psi^2|K_z|^{-1}\|w\|^2_{K_z}. \tag{2.76}$$

We now proceed to prove the inequality (2.70). Using that $0 \le \phi_z \le 1$ gives

$$\begin{aligned}\|I_{S_\mathcal{T}}w\|_K^2 = \left\|\sum_{z\in\mathcal{N}_K}\left(\int_{K_z} w\psi_z\right)\phi_z\right\|_K^2 &\le \left|\sum_{z\in\mathcal{N}_K}\int_{K_z} w\psi_z\right|^2 |K| \\ &\le (\#\mathcal{N}_K)|K|\sum_{z\in\mathcal{N}_K}\left|\int_{K_z} w\psi_z\right|^2.\end{aligned}$$

The inequality (2.76) and the fact that $\#\mathcal{N}_K \le d+1$ yields

$$\begin{aligned}\|I_{S_\mathcal{T}}w\|_K^2 &\le (d+1)|K|\sum_{z\in\mathcal{K}_K} C_\psi^2|K_z|^{-1}\|w\|^2_{K_z} \\ &\le (d+1)|K|C_\psi^2 \max_{z\in\mathcal{N}_K}|K_z|^{-1}\|w\|^2_{\omega_K} \qquad (2.77)\\ &\le (d+1)C_\psi^2\frac{|K|}{\min_{z\in\mathcal{N}_K}|K_z|}\|w\|^2_{\omega_K}. \qquad (2.78)\end{aligned}$$

Since $|K|$ and $|K_z|$, $z \in \mathcal{N}_K$, are comparable up to a constant depending on d and $\gamma_\mathcal{T}$ (in a shape-regular mesh, we can compare the size of any neighboring elements), inequality (2.70) follows.

To prove the inequalities (2.71) and (2.72), let p be a constant or linear polynomial on ω_K. Using the fact that $I_{S_\mathcal{T}}$ reproduces linear polynomials and (2.70) we get

$$\begin{aligned}\|w - I_{S_\mathcal{T}}w\|_K &= \|w - p - I_{S_\mathcal{T}}(w-p)\|_K \\ &\le \|w-p\|_K + \widehat{C}_{I_{S_\mathcal{T}},1}\|w-p\|_{\omega_K} \\ &\le (\widehat{C}_{I_{S_\mathcal{T}},1}+1)\|w-p\|_{\omega_K}.\end{aligned}$$

Using the Bramble–Hilbert lemma (Lemma 2.1) gives

$$\|w - I_{S_\mathcal{T}}w\|_K \le (\widehat{C}_{I_{S_\mathcal{T}},1}+1)C_{\mathrm{BH}}(\mathcal{T})h_K\|\nabla w\|_{\omega_K}$$

or

$$\|w - I_{S_\mathcal{T}}w\|_K \le (\widehat{C}_{I_{S_\mathcal{T}},1}+1)C_{\mathrm{BH}}(\mathcal{T})h_K^2|w|_{H^2(\omega_K)}.$$

It remains to verify the inequality (2.73). Using the fact that $I_{S_\mathcal{T}}$ reproduces constants, we have, for arbitrary $c \in \mathbb{R}$,

$$\begin{aligned}\|\nabla I_{S_\mathcal{T}}w\|_K^2 = \|\nabla I_{S_\mathcal{T}}(w-c)\|_K^2 = \int_K\left|\sum_{z\in\mathcal{N}_K}\left(\int_{K_z}(w-c)\psi_z\right)\nabla\phi_z\right|^2 \\ \le (\#\mathcal{N}_K)\sum_{z\in\mathcal{N}_K}\|\nabla\phi_z\|^2_{L^\infty(K)}\int_K\left|\int_{K_z}(w-c)\psi_z\right|^2 \\ \le (d+1)\sum_{z\in\mathcal{N}_K}\|\nabla\phi_z\|^2_{L^\infty(K)}\left|\int_{K_z}(w-c)\psi_z\right|^2|K| \\ \le (d+1)C_\phi^2\rho_K^{-2}|K|\sum_{z\in\mathcal{N}_K}\left|\int_{K_z}(w-c)\psi_z\right|^2,\end{aligned}$$

where we also used (2.75). Then, from (2.76), we get

$$\|\nabla I_{S_{\mathcal{T}}} w\|_K^2 \le (d+1) C_\phi^2 \rho_K^{-2} |K| C_\psi^2 \max_{z \in \mathcal{N}_K} |K_z|^{-1} \|w - c\|_{\omega_K}^2 .$$

Using the Bramble–Hilbert lemma (Lemma 2.1) and rearranging yields

$$\|\nabla I_{S_{\mathcal{T}}} w\|_K^2 \le (d+1) C_\phi^2 C_\psi^2 \Big(C_{\mathrm{BH}}(\mathcal{T}) \Big)^2 \frac{|K|}{\min_{z \in \mathcal{K}_K} |K_z|} \cdot \frac{h_K^2}{\rho_K^2} \|\nabla w\|_{\omega_K}^2 .$$

□

For the interpolation operator $I_{V_{\mathcal{T}}}$, we can derive bounds analogous to those of Theorem 2.1. For the "inner" elements, i.e., the elements $K \in \mathcal{T}$ such that patch ω_K does not intersect with the boundary of Ω, i.e., $\overline{\omega_K} \cap \partial\Omega = \emptyset$, the forms of the bounds and their proofs are analogous to Theorem 2.1, because $I_{V_{\mathcal{T}}}$ also reproduces constants on ω_K. For the elements whose patch intersects with the boundary of Ω, one cannot use this property and the Bramble–Hilbert lemma (Lemma 2.1) must be replaced by Friedrich's inequality (Lemma 2.2) in the proofs.

Theorem 2.2. *There exist positive constants $\widehat{C}_{I_{V_{\mathcal{T}}},\ell}$, $\ell = 1, 2, 4$, depending only on d and $\gamma_{\mathcal{T}}$ such that for all elements $K \in \mathcal{T}$,*

$$\|I_{V_{\mathcal{T}}} w\|_K \le \widehat{C}_{I_{V_{\mathcal{T}}},1} \|w\|_{\omega_K}, \quad \forall w \in L^2(\omega_K), \tag{2.79}$$

and for all $w \in H^1(\omega_K)$ if $\overline{\omega_K} \cap \partial\Omega = \emptyset$, or for all $w \in H^1(\omega_K) \cap H_0^1(\Omega)$ otherwise,

$$\|w - I_{V_{\mathcal{T}}} w\|_K \le \widehat{C}_{I_{V_{\mathcal{T}}},2} h_K \|\nabla w\|_{\omega_K}, \tag{2.80}$$

$$\|\nabla I_{V_{\mathcal{T}}} w\|_K \le \widehat{C}_{I_{V_{\mathcal{T}}},4} \|\nabla w\|_{\omega_K}. \tag{2.81}$$

For the local interpolation error over the faces, we have the following bound.

Theorem 2.3. *There exists a positive constant $\widehat{C}_{I_{V_{\mathcal{T}}},5}$ depending only on d and $\gamma_{\mathcal{T}}$ such that for all elements $K \in \mathcal{T}$,*

$$\|w - I_{V_{\mathcal{T}}} w\|_{\partial K}^2 \le \widehat{C}_{I_{V_{\mathcal{T}}},5} h_K \|\nabla w\|_{\omega_K}^2 . \tag{2.82}$$

Proof. Using the trace inequality (Lemma 2.3) and the properties of $I_{V_{\mathcal{T}}}$ from Theorem 2.2 yields

$$\begin{aligned}
\|w - I_{V_{\mathcal{T}}} w\|_{\partial K} &\le C_{\mathrm{TR}}(\mathcal{T}) [h_K^{-1} \|w - I_{V_{\mathcal{T}}} w\|_K^2 + h_K \|\nabla (w - I_{V_{\mathcal{T}}} w)\|_K^2] \\
&\le C_{\mathrm{TR}}(\mathcal{T}) \left[h_K^{-1} \|w - I_{V_{\mathcal{T}}} w\|_K^2 + h_K \cdot 2 \cdot \left(\|\nabla w\|_K^2 + \|\nabla I_{V_{\mathcal{T}}} w\|_K^2 \right) \right] \\
&\le C_{\mathrm{TR}}(\mathcal{T}) \Bigg[h_K^{-1} \left(\widehat{C}_{I_{V_{\mathcal{T}}},2} \right)^2 h_K^2 \|\nabla w\|_{\omega_K}^2 \\
&\qquad + h_K \cdot 2 \left(1 + \left(\widehat{C}_{I_{V_{\mathcal{T}}},4} \right)^2 \right) \|\nabla w\|_{\omega_K}^2 \Bigg] .
\end{aligned}$$

□

Global estimates

We now state global variants of estimates for quasi-interpolants and interpolation errors.

For any $K \in \mathcal{T}$, let $C_{\text{ovrlp}}(K)$ denote the number of patches this element is contained in, i.e.,

$$C_{\text{ovrlp}}(K) = \#\left\{K' \in \mathcal{T}; K \subset \omega_{K'}\right\}.$$

The constant $C_{\text{ovrlp}}(K)$ depends only on the geometry of the mesh $\mathcal{T}$, i.e., d and the shape regularity $\gamma_{\mathcal{T}}$.

Theorem 2.4. *There exist positive constants $C_{I_{S_{\mathcal{T}}},\ell}$, $\ell = 1, 2, 4$, depending only on d and $\gamma_{\mathcal{T}}$ such that*

$$\|I_{S_{\mathcal{T}}}w\| \le C_{I_{S_{\mathcal{T}}},1}\|w\| \qquad \forall w \in L^2(\Omega), \tag{2.83}$$

$$\left(\sum_{K\in\mathcal{T}} h_K^{-2}\|w - I_{S_{\mathcal{T}}}w\|_K^2\right)^{\frac{1}{2}} = \|h_{\mathcal{T}}^{-1}(w - I_{S_{\mathcal{T}}}w)\| \le C_{I_{S_{\mathcal{T}}},2}\|\nabla w\| \quad \forall w \in H^1(\Omega), \tag{2.84}$$

$$\|\nabla(I_{S_{\mathcal{T}}}w)\| \le C_{I_{S_{\mathcal{T}}},4}\|\nabla w\| \quad \forall w \in H^1(\Omega). \tag{2.85}$$

Proof. Using Theorem 2.1,

$$\begin{aligned}\|I_{S_{\mathcal{T}}}w\|^2 = \sum_{K\in\mathcal{T}} \|I_{S_{\mathcal{T}}}w\|_K^2 &\le \sum_{K\in\mathcal{T}} \left(\widehat{C}_{I_{S_{\mathcal{T}}},1}\right)^2 \|w\|_{\omega_K}^2 \le \left(\widehat{C}_{I_{S_{\mathcal{T}}},1}\right)^2 \sum_{K\in\mathcal{T}} C_{\text{ovrlp}}(K)\|w\|_K^2 \\ &\le \left(\widehat{C}_{I_{S_{\mathcal{T}}},1}\right)^2 \max_{K\in\mathcal{T}} C_{\text{ovrlp}}(K) \sum_{K\in\mathcal{T}} \|w\|_K^2.\end{aligned}$$

The proofs of the other three inequalities are analogous. □

Theorem 2.5. *There exist positive constants $C_{I_{V_{\mathcal{T}}},\ell}$, $\ell = 1, 2, 4, 5$, depending only on d and the shape-regularity constant $\gamma_{\mathcal{T}}$ such that*

$$\|I_{V_{\mathcal{T}}}w\| \le C_{I_{V_{\mathcal{T}}},1}\|w\| \qquad \forall w \in L^2(\Omega), \tag{2.86}$$

$$\left(\sum_{K\in\mathcal{T}} h_K^{-2}\|w - I_{V_{\mathcal{T}}}w\|_K^2\right)^{\frac{1}{2}} = \|h_{\mathcal{T}}^{-1}(w - I_{V_{\mathcal{T}}}w)\| \le C_{I_{V_{\mathcal{T}}},2}\|\nabla w\| \quad \forall w \in H_0^1(\Omega), \tag{2.87}$$

$$\|\nabla(I_{V_{\mathcal{T}}}w)\| \le C_{I_{V_{\mathcal{T}}},4}\|\nabla w\| \quad \forall w \in H_0^1(\Omega), \tag{2.88}$$

$$\left(\sum_{K\in\mathcal{T}} h_K^{-1}\|w - I_{V_{\mathcal{T}}}w\|_{\partial K}^2\right)^{\frac{1}{2}} \le C_{I_{V_{\mathcal{T}}},5}\|\nabla w\| \quad \forall w \in H_0^1(\Omega). \tag{2.89}$$

Let us now consider the mesh hierarchy as in Section 2.2.2. Since the constants $C_{I_{S_j},\ell}$ and $C_{I_{V_j},\ell}$ depend only on d and γ_j, they can be bounded by constants $C_{I_S,\ell}$ and $C_{I_V,\ell}$ depending only on d and the shape regularity γ_0 of the initial mesh $\mathcal{T}_0$.

Finally, we bound the difference of quasi-interpolates on two consecutive levels.

Theorem 2.6. *There exists a constant $C_{I,2\mathrm{lvl}} > 0$ depending only on d and γ_0 such that for all $j \geq 1$ and all $w \in H_0^1(\Omega)$,*

$$\|h_j^{-1}(I_{V_j}w - I_{V_{j-1}}w)\| \leq C_{I,2\mathrm{lvl}}\|\nabla w\|. \tag{2.90}$$

Proof. Using the fact that $h_j^{-1} = 2h_{j-1}^{-1}$ and the estimate (2.87) from Theorem 2.5,

$$\begin{aligned}\|h_j^{-1}(I_{V_j}w - I_{V_{j-1}}w)\| &\leq \|h_j^{-1}(w - I_{V_j}w)\| + \|h_j^{-1}(w - I_{V_{j-1}}w)\| \\ &= \|h_j^{-1}(w - I_{V_j}w)\| + 2\|h_{j-1}^{-1}(w - I_{V_{j-1}}w)\| \\ &\leq (C_{I_V,2} + 2C_{I_V,2})\|\nabla w\|.\end{aligned}$$

Taking $C_{I,2\mathrm{lvl}}$ as $C_{I,2\mathrm{lvl}} = C_{I_V,2} + 2C_{I_V,2}$ finishes the proof. □

2.8.3 Stable splitting

This section presents several results on splitting (decomposing) a $H_0^1(\Omega)$-function or a piecewise polynomial function into a sum of piecewise polynomial functions. Let a sequence of uniformly refined meshes $\mathcal{T}_j$, $j = 0, 1, \dots$ as in Section 2.2.2 be given.

Splitting of $H_0^1(\Omega)$ into subspaces of piecewise linear functions

To make the text easier to follow we first state the main result of this section and subsequently provide auxiliary results and proofs. We will show that any function $w \in H_0^1(\Omega)$ can be uniquely decomposed using the quasi-interpolation operators I_{V_j}, $j \in \mathbb{N}_0$, as

$$w = I_{V_0}w + \sum_{j=1}^{+\infty}(I_{V_j} - I_{V_{j-1}})w;$$

the convergence of the sum is understood in the space $H_0^1(\Omega)$ with the norm $\|\nabla \cdot \|$. This decomposition is stable, meaning that there exist positive constants c_{S,I_V}, C_{S,I_V} such that for all $w \in H_0^1(\Omega)$,

$$c_{S,I_V}\|\nabla w\|^2 \leq \|\nabla I_{V_0}w\|^2 + \sum_{j=1}^{+\infty}\|h_j^{-1}(I_{V_j}w - I_{V_{j-1}}w)\|^2 \leq C_{S,I_V}\|\nabla w\|^2. \tag{2.91}$$

We will also show that the splitting of the space $H_0^1(\Omega)$ into subspaces V_j, $j \in \mathbb{N}_0$, is stable in the sense that there exist positive constants c_S, C_S such that for all $w \in H_0^1(\Omega)$,

$$c_S\|\nabla w\|^2 \leq \inf_{w_j \in V_j;\ w = \sum_{j=0}^{+\infty} w_j} \|\nabla w_0\|^2 + \sum_{j=1}^{+\infty}\|h_j^{-1}w_j\|^2 \leq C_S\|\nabla w\|^2; \tag{2.92}$$

the infimum is taken over all $(H_0^1(\Omega), \|\nabla \cdot \|)$-convergent decompositions.

We will show that the stability constants c_{S,I_V}, C_{S,I_V} and c_S, C_S depend *only* on d and the shape regularity γ_0 of the initial mesh. In particular, the constants do not depend on the quasi-uniformity of the initial mesh or the ratio $h_\Omega / \min_{K \in \mathcal{T}_0} h_K$. This result is important when considering settings where the problem associated with the coarsest level is difficult to solve and in practice can only be solved approximately.

Variants of these results can be found, e.g., in [29, 33, 15, 14, 5] and references therein. Our form is, however, to the best of our knowledge, not presented in the literature. The results in [29, 33, 15, 14] are derived under the assumption that the initial mesh is quasi-uniform, and the authors do not track the dependence of the constants on $h_\Omega / \min_{K\in\mathcal{T}_0} h_K$. The results of [5] are derived without the assumption on the quasi-uniformity of the initial mesh. The authors however consider only the splitting of piecewise linear functions.

We combine the approaches from [29] and [5]. We first focus on showing the upper bound from (2.91), then continue with the lower bound, and later generalize it to show (2.92).

First, consider the K-functional in analogy to [5, Section 7, eq. (7.4)]. For $\omega \subset \mathbb{R}^d$, $w \in L^2(\omega)$, it is defined as

$$K(t, w, \omega) = \inf_{g\in H^2(\omega)} \left\{ \|w - g\|^2_{L^2(\omega)} + t^2 |g|^2_{H^2(\omega)} \right\}^{\frac{1}{2}}, \quad t > 0. \tag{2.93}$$

Lemma 2.6. *There exists a constant $C > 0$ such that for all $w \in H^1(\mathbb{R}^d)$ that have compact support in $\mathbb{R}^d$, it holds that*

$$\sum_{j=0}^{+\infty} 2^{2j} K(2^{-2j}, w, \mathbb{R}^d)^2 \le C \|\nabla w\|^2_{L^2(\mathbb{R}^d)}. \tag{2.94}$$

Proof. A brief proof for $d = 2$ is given in [5, Lemma 7.3]. We present its key part in more detail and for $d = 2, 3$. We will show that the K-functional can be expressed in terms of the Fourier transform (here denoted by $F[\cdot]$) as

$$K(t, w, \mathbb{R}^d)^2 = \frac{1}{(2\pi)^{\frac{d}{2}}} \int_{\mathbb{R}^d} \frac{t^2|\xi|^4}{1 + t^2|\xi|^4} \Big| F[w](\xi) \Big|^2 d\xi. \tag{2.95}$$

For $w \in H^1(\mathbb{R}^d)$, $g \in H^2(\mathbb{R}^d)$, using the properties of the Fourier transform,

$$\begin{aligned} &\|w - g\|^2_{L^2(\mathbb{R}^d)} + t^2 |g|^2_{H^2(\mathbb{R}^d)} \\ &\quad = \frac{1}{(2\pi)^{\frac{d}{2}}} \int_{\mathbb{R}^d} \frac{t^2|\xi|^4}{1 + t^2|\xi|^4} \Big| F[w](\xi) \Big|^2 + (1 + t^2|\xi|^4) \left(F[g](\xi) - \frac{F[w](\xi)}{1 + t^2|\xi|^4} \right)^2 d\xi. \end{aligned} \tag{2.96}$$

By simple manipulations, one can show that the minimum is attained for

$$\widetilde{g}(x) = w(x) * F^{-1} \left[\frac{1}{1 + t^2|\xi|^4} \right](x), \tag{2.97}$$

and it remains to show that $\widetilde{g} \in H^2(\mathbb{R}^d)$. First, note that

$$\int_{\mathbb{R}^d} (1 + |\xi|)^2 \Big(\frac{1}{1 + t^2|\xi|^4} \Big)^2 < \infty,$$

and therefore, due to the characterization of Sobolev spaces using Fourier transformations (see, e.g., [29, Section 3.1.1]), $F^{-1}[(1 + t^2|\xi|^4)^{-1}](x) \in H^2(\mathbb{R}^d)$. Then use Young's inequality for convolution (recall that by assumption, w is compactly supported and therefore $w \in L^1(\mathbb{R}^d)$) and the fact that $\partial/\partial\xi_i(f * h) = (\partial f/\partial\xi_i * h)$ to show that the H^2-norm of $\widetilde{g}$ is bounded.

The equality (2.95) then follows by plugging in the expression for $\widetilde{g}$ into (2.96) and performing algebraic manipulations. The rest of the proof of the lemma follows as in [5, Lemma 7.3]. □

Lemma 2.7. *Let $\omega \subset \mathbb{R}^d$ be a domain with a Lipschitz-continuous boundary. There exists a constant $C_\alpha(\omega) > 0$ depending on the shape of ω such that for all $w \in H^1(\omega)$,*

$$\sum_{j=0}^{+\infty} 2^{2j} K(2^{-2j}, w, \omega)^2 \leq C_\alpha(\omega) \|\nabla w\|^2_{L^2(\omega)}. \tag{2.98}$$

Proof. The proof for $d = 2$ is given in [5, Lemma 7.4]. It is based on the use of an extension operator and Lemma 2.6. For the three-dimensional case, the proof is analogous as Lemma 2.6 is valid also for $d = 3$. □

Lemma 2.8. *There exists a constant $C_\beta > 0$ depending only on d and γ_0 such that for all $K \in \mathcal{T}_0$ and all $w \in H^1_0(\Omega)$,*

$$h_K^{-2} \sum_{j=0}^{+\infty} 2^{2j} \|w - I_{S_j} w\|^2_K \leq C_\beta \|\nabla w\|^2_{\omega_K}. \tag{2.99}$$

Proof. The steps in the proof are inspired by the development in [29, Section 2.3] and [5, Section 7].

We will use a scaling argument to consider an element $\widetilde{K}$ with $h_{\widetilde{K}} = 1$. This is done by a transformation $x = h_K \widetilde{x}$, where $x \in K$, $\widetilde{x} \in \widetilde{K}$. We denote $\widetilde{f}(\widetilde{x}) := f(x)$ for any function f defined on ω_K. Then

$$\|w - I_{S_j} w\|^2_K = h_K^d \|\widetilde{w} - \widetilde{I_{S_j} w}\|^2_{\widetilde{K}}. \tag{2.100}$$

From the definition of the interpolation operator one can write $\widetilde{I_{S_j} w} = \widetilde{I_{S_j}} \widetilde{w}$. In words, one can either consider the transformation of the interpolant $I_{S_j} w$ or transform the function w to the element $\widetilde{K}$ first and then consider the quasi-interpolation $\widetilde{I_{S_j}}$ associated with the transformed mesh.

We will show that there exists a constant $C_\delta > 0$ depending only on d and γ_0 such that

$$\|\widetilde{w} - \widetilde{I_{S_j}} \widetilde{w}\|^2_{\widetilde{K}} \leq C_\delta \cdot \left(K(2^{-2j}, \widetilde{w}, \omega_{\widetilde{K}}) \right)^2. \tag{2.101}$$

Let $\widetilde{g} \in H^2(\omega_{\widetilde{K}})$. Then

$$\|\widetilde{w} - \widetilde{I_{S_j}} \widetilde{w}\|_{\widetilde{K}} \leq \|\widetilde{w} - \widetilde{g}\|_{\widetilde{K}} + \|\widetilde{g} - \widetilde{I_{S_j}} \widetilde{g}\|_{\widetilde{K}} + \|\widetilde{I_{S_j}}(\widetilde{g} - \widetilde{w})\|_{\widetilde{K}}. \tag{2.102}$$

Let $\widetilde{K}_j \in \widetilde{\mathcal{T}}_j$ such that $\widetilde{K}_j \subset \widetilde{K}$. Then (thanks to the uniform refinement and $h_{\widetilde{K}} = 1$, $h_{\widetilde{K}_j} = 2^{-j}$) from Theorem 2.1 (inequalities (2.70) and (2.72)),

$$\|\widetilde{I_{S_j}}(\widetilde{g} - \widetilde{w})\|_{\widetilde{K}_j} \leq \widehat{C}_{\widetilde{I_{S_j}},1} \|\widetilde{g} - \widetilde{w}\|_{\omega_{\widetilde{K}_j}}, \tag{2.103}$$

$$\|\widetilde{g} - \widetilde{I_{S_j}} \widetilde{g}\|_{\widetilde{K}_j} \leq \widehat{C}_{\widetilde{I_{S_j}},3} 2^{-2j} |\widetilde{g}|_{H^2(\omega_{\widetilde{K}_j})}. \tag{2.104}$$

Define

$$U(\widetilde{K},j) = \left\{ \cup\omega_{\widetilde{K}_j}; \widetilde{K}_j \in \widetilde{\mathcal{T}}_j, \widetilde{K}_j \subset \widetilde{K} \right\}.$$

The term on the right hand side of (2.103) can be bounded as

$$\begin{aligned}
\|\widetilde{I_{S_j}}(\widetilde{g}-\widetilde{w})\|_{\widetilde{K}} &= \sum_{\widetilde{K}_j\in\widetilde{\mathcal{T}}_j,\widetilde{K}_j\subset\widetilde{K}} \|\widetilde{I_{S_j}}(\widetilde{g}-\widetilde{w})\|_{\widetilde{K}_j} \\
&\leq \sum_{\widetilde{K}_j\in\widetilde{\mathcal{T}}_j,\widetilde{K}_j\subset\widetilde{K}} \widehat{C}_{\widetilde{I_{S_j}},1} \|\widetilde{g}-\widetilde{w}\|_{\omega_{\widetilde{K}_j}} \\
&\leq \widehat{C}_{\widetilde{I_{S_j}},1} \max_{\widetilde{K}_j\in\widetilde{\mathcal{T}}_j;\widetilde{K}_j\in U(\widetilde{K},j)} C_{\mathrm{ovrlp}}(\widetilde{K}_j)\|\widetilde{g}-\widetilde{w}\|_{U(\widetilde{K},j)} \\
&\leq \widehat{C}_{\widetilde{I_{S_j}},1} \max_{\widetilde{K}_j\in\widetilde{\mathcal{T}}_j;\widetilde{K}_j\in U(\widetilde{K},j)} C_{\mathrm{ovrlp}}(\widetilde{K}_j)\|\widetilde{g}-\widetilde{w}\|_{\omega_{\widetilde{K}}} \\
&\leq C_{I_S,1}\|\widetilde{g}-\widetilde{w}\|_{\omega_{\widetilde{K}}}, \qquad (2.105)
\end{aligned}$$

where the last inequality follows from the fact that $\widehat{C}_{\widetilde{I_{S_j}},1} = \widehat{C}_{I_{S_j},1}$ (scaling does not change the geometry and shape regularity) and from the definition of $C_{I_S,1}$. The term on the right hand side of (2.104) can be bounded as

$$\begin{aligned}
\|\widetilde{g}-\widetilde{I_{S_j}}\widetilde{g}\|^2_{\widetilde{K}} &= \sum_{\widetilde{K}_j\in\widetilde{\mathcal{T}}_j,\widetilde{K}_j\subset\widetilde{K}} \|\widetilde{g}-\widetilde{I_{S_j}}\widetilde{g}\|^2_{\widetilde{K}_j} \\
&\leq \sum_{\widetilde{K}_j\in\widetilde{\mathcal{T}}_j,\widetilde{K}_j\subset\widetilde{K}} \left(\widehat{C}_{\widetilde{I_{S_j}},3} 2^{-2j}|\widetilde{g}|_{\omega_{\widetilde{K}_j}}\right)^2 \\
&\leq \left(\widehat{C}_{\widetilde{I_{S_j}},3}\right)^2 \max_{\widetilde{K}_j\in\widetilde{\mathcal{T}}_j,\widetilde{K}_j\in U(\widetilde{K},j)} C_{\mathrm{ovrlp}}(\widetilde{K}_j)2^{-4j}|\widetilde{g}|^2_{H^2(U(\widetilde{K},j))} \\
&\leq \left(\widehat{C}_{\widetilde{I_{S_j}},3}\right)^2 \max_{\widetilde{K}_j\in\widetilde{\mathcal{T}}_j,\widetilde{K}_j\subset\widetilde{K}} C_{\mathrm{ovrlp}}(\widetilde{K}_j)2^{-4j}|\widetilde{g}|^2_{H^2(\omega_{\widetilde{K}})} \\
&\leq \left(C_{I_S,3}\cdot 2^{-2j}|\widetilde{g}|_{H^2(\omega_{\widetilde{K}})}\right)^2. \qquad (2.106)
\end{aligned}$$

Combining (2.102) - (2.106) yields

$$\|\widetilde{w}-\widetilde{I_{S_j}}\widetilde{w}\|_{\widetilde{K}} \leq \max_{\ell=1,3} C_{I_S,\ell}\left(\|\widetilde{g}-\widetilde{w}\|_{\omega_{\widetilde{K}}} + 2^{-2j}|\widetilde{g}|_{H^2(\omega_{\widetilde{K}})}\right).$$

From the definition of the K-functional,

$$\begin{aligned}
\|\widetilde{w}-\widetilde{I_{S_j}}\widetilde{w}\|^2_{\widetilde{K}} &\leq \max_{\ell=1,3} C^2_{I_S,\ell}\left(\|\widetilde{g}-\widetilde{w}\|_{\omega_{\widetilde{K}}} + 2^{-2j}|\widetilde{g}|_{H^2(\omega_{\widetilde{K}})}\right)^2 \\
&\leq 2\cdot\max_{\ell=1,3} C^2_{I_S,\ell}\left(\|\widetilde{g}-\widetilde{w}\|^2_{\omega_{\widetilde{K}}} + \left(2^{-2j}\right)^2|\widetilde{g}|^2_{H^2(\omega_{\widetilde{K}})}\right) \\
&= 2\cdot\max_{\ell=1,3} C^2_{I_S,\ell}\cdot\left(K(2^{-2j},\widetilde{w},\omega_{\widetilde{K}})\right)^2.
\end{aligned}$$

In the notation introduced above, $C_\delta = 2\cdot\max_{\ell=1,3} C^2_{I_S,\ell}$.

Using (2.100), (2.101) and Lemma 2.7 yields

$$\begin{aligned}
\sum_{j=0}^{+\infty} 2^{2j}\|w-I_{S_j}w\|^2_K &\leq h^d_K C_\delta \sum_{j=0}^{+\infty} 2^{2j}\left(K(2^{-2j},\widetilde{w},\omega_{\widetilde{K}})\right)^2 \\
&\leq h^d_K C_\delta C_\alpha(\omega_{\widetilde{K}})\|\nabla\widetilde{w}\|^2_{\omega_{\widetilde{K}}}.
\end{aligned}$$

Re-scaling back to K,

$$\sum_{j=0}^{+\infty} 2^{2j}\|w - I_{S_j}w\|_K^2 \le h_K^d C_\delta C_\alpha(\omega_{\widetilde{K}}) h_K^2 h_K^{-d} \|\nabla w\|_{\omega_K}^2.$$

Finally, note that the shape of $\omega_{\widetilde{K}}$ depends on the shape regularity of the initial mesh and therefore $C_\alpha(\omega_{\widetilde{K}})$ can be bounded, for all $K \in \mathcal{T}_0$, by a constant C_α depending only on d and γ_0. □

Theorem 2.7. *There exists a constant $C_{S,I_S} > 0$ depending only on d and γ_0, such that for all $w \in H_0^1(\Omega)$,*

$$\|\nabla I_{S_0}w\|^2 + \sum_{j=1}^{+\infty} \|h_j^{-1}(I_{S_j}w - I_{S_{j-1}}w)\|^2 \le C_{S,I_S}\|\nabla w\|^2. \tag{2.107}$$

Proof. From Theorem 2.4,

$$\|\nabla I_{S_0}w\|^2 \le C_{I_{S_0},4}^2 \|\nabla w\|^2.$$

For the rest of the sum,

$$\begin{aligned}
\sum_{j=1}^{+\infty} \|h_j^{-1}(I_{S_j} - I_{S_{j-1}})w\|^2 &= \sum_{j=1}^{+\infty} \|h_j^{-1}(I_{S_j}w - w + w - I_{S_{j-1}}w)\|^2 \\
&\le 2\sum_{j=1}^{+\infty} \left(\|h_j^{-1}(w - I_{S_j}w)\|^2 + \|h_j^{-1}(w - I_{S_{j-1}}w)\|^2\right) \\
&\le 2\sum_{j=1}^{+\infty} \left(\|h_j^{-1}(w - I_{S_j}w)\|^2 + 4\|h_{j-1}^{-1}(w - I_{S_{j-1}}w)\|^2\right) \\
&\le 2\left(\sum_{j=1}^{+\infty} \|h_j^{-1}(w - I_{S_j}w)\|^2 + 4\sum_{j=0}^{+\infty} \|h_j^{-1}(w - I_{S_j}w)\|^2\right) \\
&\le 2 \cdot 5 \sum_{j=0}^{+\infty} \|h_j^{-1}(w - I_{S_j}w)\|^2 \\
&= 10 \sum_{K\in\mathcal{T}_0} \sum_{j=0}^{+\infty} 2^{2j} h_K^{-2} \|w - I_{S_j}w\|_K^2 \\
&\le 10 \sum_{K\in\mathcal{T}_0} C_\beta \|\nabla w\|_{\omega_K}^2 \le 10 \cdot C_\beta \cdot \max_{K\in\mathcal{T}_0} C_{\text{ovrlp}}(K)\|\nabla w\|^2,
\end{aligned}$$

where we have used Lemma 2.8 in the second to last inequality. □

Theorem 2.8. *There exists a constant $C_{S,I_V} > 0$ depending only on d and γ_0 such that for all $w \in H_0^1(\Omega)$,*

$$\|\nabla I_{V_0}w\|^2 + \sum_{j=1}^{+\infty} \|h_j^{-1}(I_{V_j}w - I_{V_{j-1}}w)\|^2 \le C_{S,I_V}\|\nabla w\|^2. \tag{2.108}$$

Proof. The key steps of the following proof of the upper bound were provided to us by professor P. Oswald in personal communications. From Theorem 2.5,

$$\|\nabla I_{V_0}w\|^2 \le C_{I_{V_0},4}^2 \|\nabla w\|^2.$$

To bound $\sum_{j=1}^{+\infty}\left\|h_j^{-1}(I_{V_j}w - I_{V_{j-1}}w)\right\|^2$, consider the sequence $w_j = (I_{S_j} - I_{S_{j-1}})w$, $j \in \mathbb{N}$. Let $K \in \mathcal{T}_0$. We will first show that there exists a constant $C_\epsilon > 0$ depending only on d and γ_0 such that

$$\sum_{j=1}^{+\infty} 2^{2j}\|(I_{V_j} - I_{V_{j-1}})w\|_K^2 \le C_\epsilon \sum_{i=1}^{+\infty} 2^{2i}\|w_i\|_{\omega_K}^2. \tag{2.109}$$

Since I_{V_j} are projections onto V_j, it holds that

$$(I_{V_j} - I_{V_{j-1}})w_i = 0, \quad j > i. \tag{2.110}$$

Then for all $j \ge 1$, using the Cauchy–Schwarz inequality for sums and Theorem 2.2,

$$\begin{aligned}
\|(I_{V_j} - I_{V_{j-1}})w\|_K^2 &= \int_K \left(\sum_{i=j}^{+\infty}(I_{V_j} - I_{V_{j-1}})w_i\right)^2 \\
&\le \int_K \left(\sum_{i=j}^{+\infty} 2^{-i}\right)\left(\sum_{i=j}^{+\infty} 2^i\left((I_{V_j} - I_{V_{j-1}})w_i\right)^2\right) \\
&\le 2\cdot 2^{-j}\sum_{i=j}^{+\infty} 2^i\|(I_{V_j} - I_{V_{j-1}})w_i\|_K^2 \\
&\le 2\cdot 2^{-j}\sum_{i=j}^{+\infty} 2^i\cdot 2\left(\|I_{V_j}w_i\|_K^2 + \|I_{V_{j-1}}w_i\|_K^2\right) \\
&\le 2\cdot 2^{-j}\sum_{i=j}^{+\infty} 2^i\cdot 2\cdot 2\cdot\left(\widehat{C}_{I_V,1}\right)^2\|w_i\|_{\omega_K}^2.
\end{aligned}$$

Consequently,

$$\sum_{j=1}^{+\infty} 2^{2j}\|(I_{V_j} - I_{V_{j-1}})w\|_K^2 \le \underbrace{8\cdot\left(\widehat{C}_{I_{V_\mathcal{T}},1}\right)^2}_{C_\epsilon}\sum_{j=1}^{+\infty} 2^j\sum_{i=j}^{+\infty} 2^i\|w_i\|_{\omega_K}^2.$$

Changing the order of summation,

$$\sum_{j=1}^{+\infty} 2^j\sum_{i=j}^{+\infty} 2^i\|w_i\|_{\omega_K}^2 = \sum_{i=1}^{+\infty} 2^i\|w_i\|_{\omega_K}^2\sum_{j=1}^{i-1} 2^j \le \sum_{i=1}^{+\infty} 2^{2i}\|w_i\|_{\omega_K}^2.$$

For the sum $\sum_{j=1}^{+\infty}\|h_j^{-1}(I_{V_j} - I_{V_{j-1}})w\|^2$, using (2.109),

$$\begin{aligned}
\sum_{j=1}^{+\infty}\|h_j^{-1}(I_{V_j} - I_{V_{j-1}})w\|^2 &= \sum_{K\in\mathcal{T}_0} h_K^{-2}\sum_{j=1}^{+\infty} 2^{2j}\|(I_{V_j} - I_{V_{j-1}})w\|_K^2 \\
&\le C_\epsilon\sum_{K\in\mathcal{T}_0} h_K^{-2}\sum_{i=1}^{+\infty} 2^{2i}\|w_i\|_{\omega_K}^2 \\
&\le C_\epsilon\sum_{i=1}^{+\infty} 2^{2i}\sum_{K\in\mathcal{T}_0} h_K^{-2}\|w_i\|_{\omega_K}^2 \\
&\le C_\epsilon\sum_{i=1}^{+\infty} 2^{2i}\sum_{K\in\mathcal{T}_0} C_{\mathrm{ovrlp}}(K)\max_{\bar{K}\subset\omega_K} h_{\bar{K}}^{-2}\|w_i\|_K^2 \\
&\le \underbrace{C_\epsilon\max_{K\in\mathcal{T}_0}\left[C_{\mathrm{ovrlp}}(K)\frac{\max_{\bar{K}\subset\omega_K} h_{\bar{K}}^{-2}}{h_K^{-2}}\right]}_{C_\zeta}\sum_{i=1}^{+\infty} 2^{2i}\sum_{K\in\mathcal{T}_0} h_K^{-2}\|w_i\|_K^2.
\end{aligned}$$

Finally, Theorem 2.7 gives

$$\sum_{j=1}^{+\infty} \|h_j^{-1}(I_{V_j} - I_{V_{j-1}})w\|^2 \leq C_\zeta C_{S,I_S} \|\nabla w\|^2.$$

□

Now proceed with bounding the norm of the splittings from below by a H^1-seminorm. We start with some auxiliary lemmas.

Lemma 2.9. *There exists a constant $c_S > 0$ depending only on d and γ_0 such that for any $N \in \mathbb{N}_0$ and any sequence $(w_j)_{j=0}^N$, $w_j \in S_j$, $j = 0, \ldots, N$, it holds that*

$$\left\| \nabla \left(\sum_{j=0}^N w_j \right) \right\|^2 \leq \frac{1}{c_S} \left(\|\nabla w_0\|^2 + \sum_{j=1}^N \|h_j^{-1} w_j\|^2 \right). \tag{2.111}$$

Proof. The proof for $d = 2$ is given in [5, Lemma 3.4]. It is based on the so-called Strengthened Cauchy–Schwarz inequality. As the Strengthened Cauchy–Schwarz inequality is valid also for $d = 3$ (see, e.g., [43, Lemma 6.1]), the proof of the theorem in the three-dimensional case is analogous to the two-dimensional one. □

Lemma 2.10. *Let $(w_j)_{j=0}^{+\infty}$, $w_j \in V_j$, $j \in \mathbb{N}_0$, be a sequence which satisfies*

$$\|\nabla w_0\|^2 + \sum_{j=1}^{+\infty} \|h_j^{-1} w_j\|^2 < +\infty.$$

Then $\sum_{j=0}^{+\infty} w_j$ converges in $(H_0^1(\Omega), \|\nabla \cdot \|)$.

Proof. We will use Lemma 2.9 to show that $(\sum_{j=0}^N w_j)_{N=0}^{+\infty}$ is a Cauchy sequence in $(H_0^1(\Omega), \|\nabla \cdot \|)$. Let $\epsilon > 0$. Since $\|\nabla w_0\|^2 + \sum_{j=1}^{+\infty} \|h_j^{-1} w_j\|^2$ converges in $\mathbb{R}$, there exists $M \in \mathbb{N}$ such that for all $m > n > M$, it holds that

$$\sum_{j=n}^{m} \|h_j^{-1} w_j\|^2 < c_S \epsilon^2.$$

Using Lemma 2.9 for w_j, $j = n, \ldots, m$, and the previous inequality,

$$\left\| \nabla \left(\sum_{j=n}^m w_j \right) \right\|^2 \leq \frac{1}{c_S} \sum_{j=n}^m \|h_j^{-1} w_j\|^2 < \epsilon^2,$$

i.e., $(\sum_{j=0}^N w_j)_{N=0}^{+\infty}$ is a Cauchy sequence in $(H_0^1(\Omega), \|\nabla \cdot \|)$ and thus $\sum_{j=0}^{+\infty} w_j$ converges in $(H_0^1(\Omega), \|\nabla \cdot \|)$. □

Lemma 2.11. *Let c_S be the constant from Lemma 2.9. Let $(w_j)_{j=0}^{+\infty}$, $w_j \in V_j$, $j \in \mathbb{N}_0$, be a sequence such that $\sum_{j=0}^{+\infty} w_j$ converges in $(H_0^1(\Omega), \|\nabla \cdot \|)$. Then*

$$\left\| \nabla \left(\sum_{j=0}^{+\infty} w_j \right) \right\|^2 \leq \frac{1}{c_S} \left(\|\nabla w_0\|^2 + \sum_{j=1}^{+\infty} \|h_j^{-1} w_j\|^2 \right).$$

Proof. For any $N \in \mathbb{N}_0$, Lemma 2.9 gives

$$\left\| \nabla \left(\sum_{j=0}^{N} w_j \right) \right\|^2 \leq \frac{1}{c_S} \left(\|\nabla w_0\|^2 + \sum_{j=1}^{N} \|h_j^{-1} w_j\|^2 \right).$$

Since $\sum_{j=0}^{+\infty} w_j$ converges in $(H_0^1(\Omega), \|\nabla \cdot\|)$, we may switch the following limit and norm, giving

$$\begin{aligned}\left\| \nabla \left(\lim_{N \to +\infty} \sum_{j=0}^{N} w_j \right) \right\|^2 &= \lim_{N \to +\infty} \left\| \nabla \left(\sum_{j=0}^{N} w_j \right) \right\|^2 \\ &\leq \lim_{N \to +\infty} \frac{1}{c_S} \left(\|\nabla w_0\|^2 + \sum_{j=1}^{N} \|h_j^{-1} w_j\|^2 \right) = \frac{1}{c_S} \left(\|\nabla w_0\|^2 + \sum_{j=1}^{+\infty} \|h_j^{-1} w_j\|^2 \right).\end{aligned}$$

□

Theorem 2.9. *Any function $w \in H_0^1(\Omega)$ can be uniquely decomposed as*

$$w = I_{V_0} w + \sum_{j=1}^{+\infty} (I_{V_j} - I_{V_{j-1}}) w;$$

(the convergence of the sum is understood in the space $(H_0^1(\Omega), \|\nabla \cdot \|)$). Let c_S be the constant from Lemma 2.9 and C_{S,I_V} the constant from Theorem 2.8. Then for all $w \in H_0^1(\Omega)$,

$$c_S \|\nabla w\|^2 \leq \|\nabla I_{V_0} w\|^2 + \sum_{j=1}^{+\infty} \|h_j^{-1} (I_{V_j} w - I_{V_{j-1}} w)\|^2 \leq C_{S,I_V} \|\nabla w\|^2. \tag{2.112}$$

Proof. The upper bound is proven in Theorem 2.8. Now we will prove the lower bound. Having the upper bound, we can use Lemma 2.10 to show that the sum $I_{V_0} w + \sum_{j=1}^{+\infty} (I_{V_j} - I_{V_{j-1}}) w$ converges in $(H_0^1(\Omega), \|\nabla \cdot \|)$ and consequently, from Lemma 2.11 with $w_0 := I_{V_0} w$ and $w_j := (I_{V_j} - I_{V_{j-1}}) w$,

$$c_S \left\| \nabla \left(I_{V_0} w + \sum_{j=1}^{+\infty} (I_{V_j} - I_{V_{j-1}}) w \right) \right\|^2 \leq \|\nabla I_{V_0} w\|^2 + \sum_{j=1}^{+\infty} \|h_j^{-1} (I_{V_j} w - I_{V_{j-1}} w)\|^2.$$

It remains to show that $I_{V_0} w + \sum_{j=0}^{+\infty} (I_{V_j} - I_{V_{j-1}}) w = w$ in $(H_0^1(\Omega), \|\nabla \cdot \|)$. Since, for arbitrary $N \in \mathbb{N}$ (see Theorem 2.5),

$$\left\| w - \left(I_{V_0} w + \sum_{j=1}^{N} (I_{V_j} - I_{V_{j-1}}) w \right) \right\| = \|w - I_{V_N} w\| \leq C_{I_V,2} \max_{K \in \mathcal{T}_N} h_K \|\nabla w\|,$$

and $\max_{K \in \mathcal{T}_N} h_K \to 0$, $I_{V_0} w + \sum_{j=1}^{+\infty} (I_{V_j} - I_{V_{j-1}}) w = w$ in $L^2(\Omega)$. We will show by contradiction that $I_{V_0} w + \sum_{j=1}^{+\infty} (I_{V_j} - I_{V_{j-1}}) w = w$ also in $(H_0^1(\Omega), \|\nabla \cdot \|)$. Let the sequence $I_{V_0} w + \sum_{j=1}^{N} (I_{V_j} - I_{V_{j-1}}) w$ converge in $(H_0^1(\Omega), \|\nabla \cdot \|)$ to $\bar{w} \neq w$. Then, thanks to Friedrich's inequality (Lemma 2.2) the sequence converges to $\bar{w}$ in $L^2(\Omega)$, which is a contradiction with the uniqueness of the limit. □

Theorem 2.10. *Let c_S be the constant from Lemma 2.11. There exists a constant $C_S > 0$ depending only on d and γ_0 such that for all $w \in H_0^1(\Omega)$,*

$$c_S \|\nabla w\|^2 \leq \inf_{w_j \in V_j;\ w=\sum_{j=0}^{+\infty} w_j} \|\nabla w_0\|^2 + \sum_{j=1}^{+\infty} \|h_j^{-1} w_j\|^2 \leq C_S \|\nabla w\|^2. \tag{2.113}$$

Proof. From Theorem 2.9 we know that for any $w \in H_0^1(\Omega)$ there exists a decomposition $w = \sum_{j=0}^{+\infty} w_j$, $w_j \in V_j$, $j \in \mathbb{N}_0$, for which the upper bound holds with the constant C_{S,I_V}, so that we can take an infimum over all possible decompositions giving $C_S \leq C_{S,I_V}$. The lower bound in (2.113) follows from Lemma 2.11. □

Splitting of $H_0^1(\Omega)$ into basis function spaces

This section presents a result on splitting a $H_0^1(\Omega)$-function into basis function spaces. Denote by $V_{j,i}$, $j = 1, 2 \ldots$, $i = 1, \ldots, \#\mathcal{K}_j$, the space spanned by the basis function $\phi_i^{(j)}$, $V_{j,i} \subset V_j$.

First we will show that splitting a function $w_j \in V_j$ into the basis function spaces $V_{j,i}$, $i = 1, \ldots, \#\mathcal{K}_j$ is stable. This property is called stability of basis functions in the literature; see, e.g., [33, Definition 2.5.5] and [29, Assumption (A1), p. 17]. We present this property in a form which suits our further development.

Lemma 2.12 (Stability of basis functions)**.** *There exist positive constants c_B and C_B depending only on d and γ_0 such that for all $j \in \mathbb{N}_0$ and all*

$$w_j = \sum_{i=1}^{\#\mathcal{K}_j} w_{j,i} \in V_j, \quad w_{j,i} \in V_{j,i}, \quad i = 1, \ldots, \#\mathcal{K}_j,$$

it holds that

$$c_B \|h_j^{-1} w_j\|^2 \leq \sum_{i=1}^{\#\mathcal{K}_j} \|\nabla w_{j,i}\|^2 \leq C_B \|h_j^{-1} w_j\|^2. \tag{2.114}$$

Let $\mathbf{M}_j^{\mathrm{S}}$ be the so-called scaled mass matrix and $\mathbf{D}_j$ the diagonal matrix defined as

$$\left[\mathbf{M}_j^{\mathrm{S}}\right]_{m,n} = \int_\Omega h_j^{-2} \phi_n^{(j)} \phi_m^{(j)}, \qquad [\mathbf{D}_j]_{m,m} = \int_\Omega \nabla\phi_m^{(j)} \cdot \nabla\phi_m^{(j)}, \quad \forall m, n = 1, \ldots, \#\mathcal{K}_j.$$

Let $\mathbf{w}_j$ be the vector of coefficients of a function $w_j \in V_j$ in the basis Φ_j. Then $w_j = \sum_{i=1}^{\#\mathcal{K}_j} w_{j,i}$, $w_{j,i} = [\mathbf{w}_j]_i \phi_i^{(j)}$ and (2.114) is equivalent to

$$c_B \mathbf{w}_j^* \mathbf{M}_j^S \mathbf{w}_j \leq \mathbf{w}_j^* \mathbf{D}_j \mathbf{w}_j \leq C_B \mathbf{w}_j^* \mathbf{M}_j^S \mathbf{w}_j. \tag{2.115}$$

That is, the matrices $\mathbf{M}_j^{\mathrm{S}}$ and $\mathbf{D}_j$ are spectrally equivalent with constants c_B and C_B.

Proof. The proof is inspired by [16, Proposition 1.30, Problem 1.35]; see also [17]. We prove the spectral equivalence of local matrices associated with a mesh element. The assertion of the theorem for global matrices then follows by summing the local inequalities over the elements and taking the overlap into account.

Let $\mathbf{M}_{j,K}^{\mathrm{S}}$ be a local scaled mass matrix corresponding to an element $K \in \mathcal{T}_j$ defined as

$$\left[\mathbf{M}_{j,K}^{\mathrm{S}}\right]_{m,n} = \int_K h_K^{-2} \phi_n^{(j)} \phi_m^{(j)}, \qquad \forall m, n \in \mathcal{N}_K, \tag{2.116}$$

and let $\mathbf{M}^{\mathrm{S}}_{\hat{K}}$ be the local scaled mass matrix on a reference element $\hat{K}$, which does not depend on j, K, or $\mathcal{T}_j$. Using standard arguments of affine transformation to a reference element, it holds that

$$\mathbf{M}^{\mathrm{S}}_{j,K} = \frac{|K|}{h_K^2}\mathbf{M}^{\mathrm{S}}_{\hat{K}}. \tag{2.117}$$

If we denote by $c_{\hat{K}}$ and $C_{\hat{K}}$ the smallest and the largest eigenvalues of $\mathbf{M}^{\mathrm{S}}_{\hat{K}}$, respectively, the eigenvalues of $\mathbf{M}^{\mathrm{S}}_{j,K}$ can be bounded by $c_{\hat{K}}|K|/h_K^2$ and $C_{\hat{K}}|K|/h_K^2$. Consequently,

$$\frac{c_{\hat{K}}|K|}{h_K^2}\mathbf{x}^*\mathbf{x} \leq \mathbf{x}^*\mathbf{M}^{\mathrm{S}}_{j,K}\mathbf{x} \leq \frac{C_{\hat{K}}|K|}{h_K^2}\mathbf{x}^*\mathbf{x}, \quad \forall \mathbf{x} \in \mathbb{R}^{(d+1)}. \tag{2.118}$$

By choosing $\mathbf{x}$ as the mth column of the identity matrix of size $d+1$,

$$\frac{c_{\hat{K}}|K|}{h_K^2} \leq \left[\mathbf{M}^{\mathrm{S}}_{j,K}\right]_{m,m} = \frac{\|\phi_m^{(j)}\|_K^2}{h_K^2} \leq \frac{C_{\hat{K}}|K|}{h_K^2}. \tag{2.119}$$

Let $\mathbf{D}_{j,K}$ be the local variant of $\mathbf{D}_j$, i.e.,

$$[\mathbf{D}_{j,K}]_{m,m} = \int_K \nabla\phi_m^{(j)}\nabla\phi_m^{(j)} = \|\nabla\phi_m^{(j)}\|_K^2. \tag{2.120}$$

Using the inverse inequality (Lemma 2.4) and (2.119),

$$\|\nabla\phi_m^{(j)}\|_K^2 \leq C_{\mathrm{INV}}^2 h_K^{-2}\|\phi_m^{(j)}\|_K^2 \leq C_{\mathrm{INV}}^2 C_{\hat{K}}\frac{|K|}{h_K^2}. \tag{2.121}$$

Similarly, using Friedrich's inequality (Lemma 2.2),

$$\frac{c_{\hat{K}}|K|}{C_F^2 h_K^2} \leq \frac{1}{C_F^2 h_K^2}\|\phi_m^{(j)}\|_K^2 \leq \|\nabla\phi_m^{(j)}\|_K^2. \tag{2.122}$$

Thus the matrix $\mathbf{D}_{j,K}$ is spectrally equivalent to the identity matrix times $\frac{|K|}{h_K^2}$. From (2.118), we conclude that $\mathbf{D}_{j,K}$ is also spectrally equivalent to $\mathbf{M}^{\mathrm{S}}_{j,K}$ with the equivalency constants involving C_{INV}, C_F, $c_{\hat{K}}$, and $C_{\hat{K}}$, i.e., depending only on d and the shape regularity γ_j. □

Since $\mathbf{M}^{\mathrm{S}}_j$ and $\mathbf{D}_j$ are spectrally equivalent matrices and they are symmetric positive definite, we can use the generalized Hermitian eigenvalue decomposition (see, e.g., [3, Eq. (5.3)]) and algebraic manipulations to show that $\left(\mathbf{M}^{\mathrm{S}}_j\right)^{-1}$ and $\mathbf{D}_j^{-1}$ are also spectrally equivalent, i.e.,

$$\frac{1}{C_B}\mathbf{w}^*\left(\mathbf{M}^{\mathrm{S}}_j\right)^{-1}\mathbf{w} \leq \mathbf{w}^*\mathbf{D}_j^{-1}\mathbf{w} \leq \frac{1}{c_B}\mathbf{w}^*\left(\mathbf{M}^{\mathrm{S}}_j\right)^{-1}\mathbf{w}, \qquad \forall \mathbf{w} \in \mathbb{R}^{\#\mathcal{K}_j}. \tag{2.123}$$

Let $\mathbf{M}_j$ denote the mass matrix associated with the jth level, i.e., $[\mathbf{M}_j]_{mn} = \int_\Omega \phi_n^{(j)}\phi_m^{(j)}$, $m,n = 1,\ldots,\#\mathcal{K}_j$. Analogously to (2.115) we can show the spectral equivalence of the mass matrix $\mathbf{M}_j$ with the diagonal matrix $\mathbf{D}_j$ in the following form. There exist positive constants c_M, C_M depending only on d and γ_0 such that

$$c_M \min_{K\in\mathcal{T}_j} h_K^{-2}\mathbf{w}^*\mathbf{M}_j\mathbf{w} \leq \mathbf{w}^*\mathbf{D}_j\mathbf{w} \leq C_M \max_{K\in\mathcal{T}_j} h_K^{-2}\mathbf{w}^*\mathbf{M}_j\mathbf{w}, \quad \forall \mathbf{w} \in \mathbb{R}^{\#\mathcal{K}_j}. \tag{2.124}$$

Combining Theorem 2.10 and Lemma 2.12 yields the following theorem on splitting a $H_0^1(\Omega)$-function into basis function spaces. It can be proven by the same technique as in [33, Theorem 2.3.1].

Theorem 2.11. *Let c_S, C_S be the constants from Theorem 2.10, c_B, C_B the constants from Lemma 2.12, and let $\overline{c}_B = \min\{1, c_B\}$ and $\overline{C}_B = \max\{1, C_B\}$. Then for all $w \in H_0^1(\Omega)$,*

$$c_S\overline{c}_B\|\nabla w\|^2 \leq \inf_{\substack{w_0\in V_0, w_{j,i}\in V_{j,i}\\ w=w_0+\sum_{j=1}^{+\infty}\sum_{i=1}^{\#\mathcal{K}_j} w_{j,i}}} \|\nabla w_0\|^2 + \sum_{j=1}^{+\infty}\sum_{i=1}^{\#\mathcal{K}_j} \|\nabla w_{j,i}\|^2 \leq C_S\overline{C}_B\|\nabla w\|^2. \tag{2.125}$$

Splitting of spaces of piecewise linear functions

We now present consequences of the previous theorems for finite-dimensional piecewise linear functions from V_J, $J \geq 0$. The following theorems can be proven by the same techniques as the results in [33, Section 2.4].

Theorem 2.12. *Let c_S and C_S be the constants from Theorem 2.10. Let $J \geq 0$. For all $w_J \in V_J$,*

$$c_S\|\nabla w_J\|^2 \leq \inf_{w_j\in V_j;\ w_J=\sum_{j=0}^{J} w_j} \|\nabla w_0\|^2 + \sum_{j=1}^{J} \|h_j^{-1}w_j\|^2 \leq C_S\|\nabla w_J\|^2. \tag{2.126}$$

Theorem 2.13. *Let c_S and C_S be the constants from Theorem 2.10 and $\overline{c}_B, \overline{C}_B$ the constants from Theorem 2.11. Let $J \geq 0$. For all $w_J \in V_J$,*

$$c_S\overline{c}_B\|\nabla w_J\|^2 \leq \inf_{\substack{w_0\in V_0, w_{j,i}\in V_{j,i}\\ w_J=w_0+\sum_{j=1}^{J}\sum_{i=1}^{\#\mathcal{K}_j} w_{j,i}}} \|\nabla w_0\|^2 + \sum_{j=1}^{J}\sum_{i=1}^{\#\mathcal{K}_j} \|\nabla w_{j,i}\|^2 \leq C_S\overline{C}_B\|\nabla w_J\|^2. \tag{2.127}$$

Frame

Finally, we present a consequence of the stability of the splittings presented in Theorem 2.11, which is closely related to the fact that the normalized basis functions form a so-called *frame in* $(H_0^1(\Omega))^{\#}$; see, e.g., [20, Section 3], [21].

Theorem 2.14. *Let c_S and C_S be the constants from Theorem 2.10 and $\overline{c}_B, \overline{C}_B$ the constants from Theorem 2.11. For all $g \in (H_0^1(\Omega))^{\#}$,*

$$\begin{aligned} c_S\overline{c}_B\left(\|\nabla g_0\|^2 + \sum_{j=1}^{+\infty}\sum_{i=1}^{\#\mathcal{K}_j} \frac{\langle g, \phi_i^{(j)}\rangle^2}{\|\nabla\phi_i^{(j)}\|^2}\right) &\leq \|g\|^2_{(H_0^1(\Omega))^{\#}} \\ &\leq C_S\overline{C}_B\left(\|\nabla g_0\|^2 + \sum_{j=1}^{+\infty}\sum_{i=1}^{\#\mathcal{K}_j} \frac{\langle g, \phi_i^{(j)}\rangle^2}{\|\nabla\phi_i^{(j)}\|^2}\right), \end{aligned}$$

where $g_0 \in V_0$ is the Riesz representation of the functional g in the space V_0 with respect to the inner product $(u_0, v_0)_0 = \int_\Omega \nabla v_0 \cdot \nabla u_0$, $\forall u_0, v_0 \in V_0$.

Proof. The proof is inspired by the proof of [33, Theorem 2.6.2].

We will start with the upper bound. Let $w \in H_0^1(\Omega)$ and consider an arbitrary decomposition $w = w_0 + \sum_{j=1}^{+\infty}\sum_{i=1}^{\#\mathcal{K}_j} w_{j,i}$, $w_0 \in V_0, w_{j,i} \in V_{j,i}$. Using the fact that

$$w_{i,j} = \frac{\operatorname{sign}(w_{i,j})\,\|\nabla w_{i,j}\|}{\|\nabla\phi_i^{(j)}\|}\,\phi_i^{(j)},$$

we have

$$\begin{aligned}
|\langle g, w\rangle| &\leq |\langle g, w_0\rangle| + \sum_{j=1}^{+\infty}\sum_{i=1}^{\#\mathcal{K}_j} |\langle g, w_{i,j}\rangle| \\
&\leq \|g_0\| \cdot \|\nabla w_0\| + \sum_{j=1}^{+\infty}\sum_{i=1}^{\#\mathcal{K}_j} \left|\left\langle g, \frac{\phi_i^{(j)}}{\|\nabla \phi_i^{(j)}\|}\right\rangle\right| \cdot \|\nabla w_{i,j}\| \\
&\leq \left(\|\nabla g_0\|^2 + \sum_{j=1}^{+\infty}\sum_{i=1}^{\#\mathcal{K}_j} \frac{\langle g, \phi_i^{(j)}\rangle^2}{\|\nabla \phi_i^{(j)}\|^2}\right)^{\frac{1}{2}} \cdot \left(\|\nabla w_0\|^2 + \sum_{j=1}^{+\infty}\sum_{i=1}^{\#\mathcal{K}_j} \|\nabla w_{j,i}\|^2\right)^{\frac{1}{2}}.
\end{aligned}$$

Taking the infimum over all decompositions $w = w_0 + \sum_{j=1}^{+\infty}\sum_{i=1}^{\#\mathcal{K}_j} w_{j,i}$, $w_0 \in V_0, w_{j,i} \in V_{j,i}$ and using the stability of the decomposition into spaces defined by basis functions (Theorem 2.11) yields

$$|\langle g, w\rangle| \leq \left(\|\nabla g_0\|^2 + \sum_{j=1}^{+\infty}\sum_{i=1}^{\#\mathcal{K}_j} \frac{\langle g, \phi_i^{(j)}\rangle^2}{\|\nabla \phi_i^{(j)}\|^2}\right)^{\frac{1}{2}} \cdot C_S^{\frac{1}{2}} \overline{C}_B^{\frac{1}{2}} \|\nabla w\|.$$

Taking the supremum over all $w \in H_0^1(\Omega)$ such that $\|\nabla w\| = 1$ gives the upper bound.

Proving the lower bound is more subtle. We will first show that for any $N \in \mathbb{N}$,

$$\|\nabla g_0\|^2 + \sum_{j=1}^{N}\sum_{i=1}^{\#\mathcal{K}_j} \frac{\langle g, \phi_i^{(j)}\rangle^2}{\|\nabla \phi_i^{(j)}\|^2} \leq \frac{1}{c_S \overline{c}_B} \|g\|^2_{\left(H_0^1(\Omega)\right)^\#}. \tag{2.128}$$

First, it holds that

$$\begin{aligned}
\|\nabla g_0\|^2 + \sum_{j=1}^{N}\sum_{i=1}^{\#\mathcal{K}_j} \frac{\langle g, \phi_i^{(j)}\rangle^2}{\|\nabla \phi_i^{(j)}\|^2} &= \langle g, g_0\rangle + \sum_{j=1}^{N}\sum_{i=1}^{\#\mathcal{K}_j} \left\langle g, \frac{\langle g, \phi_i^{(j)}\rangle}{\|\nabla \phi_i^{(j)}\|^2} \phi_i^{(j)}\right\rangle \\
&= \left\langle g, g_0 + \sum_{j=1}^{N}\sum_{i=1}^{\#\mathcal{K}_j} \frac{\langle g, \phi_i^{(j)}\rangle}{\|\nabla \phi_i^{(j)}\|^2} \phi_i^{(j)}\right\rangle.
\end{aligned}$$

Let $g_{j,i} = \frac{\langle g, \phi_i^{(j)}\rangle}{\|\nabla \phi_i^{(j)}\|^2} \phi_i^{(j)} \in V_{j,i}$. Then using Theorem 2.13,

$$\begin{aligned}
\left\langle g, g_0 + \sum_{j=1}^{N}\sum_{i=1}^{\#\mathcal{K}_j} g_{j,i}\right\rangle &= \|g\|_{\left(H_0^1(\Omega)\right)^\#} \left\|\nabla\left(g_0 + \sum_{j=1}^{N}\sum_{i=1}^{\#\mathcal{K}_j} g_{j,i}\right)\right\| \\
&\leq \|g\|_{\left(H_0^1(\Omega)\right)^\#} \frac{1}{c_S^{\frac{1}{2}} \overline{c}_B^{\frac{1}{2}}} \left(\|\nabla g_0\|^2 + \sum_{j=1}^{N}\sum_{i=1}^{\#\mathcal{K}_j} \|\nabla g_{j,i}\|^2\right)^{\frac{1}{2}} \\
&= \|g\|_{\left(H_0^1(\Omega)\right)^\#} \frac{1}{c_S^{\frac{1}{2}} \overline{c}_B^{\frac{1}{2}}} \left(\|\nabla g_0\|^2 + \sum_{j=1}^{N}\sum_{i=1}^{\#\mathcal{K}_j} \frac{\langle g, \phi_i^{(j)}\rangle^2}{\|\nabla \phi_i^{(j)}\|^2}\right)^{\frac{1}{2}}.
\end{aligned}$$

This yields (2.128). Taking N to infinity in (2.128) finishes the proof. □

Theorem 2.15. *Let c_S and C_S be the constants from Theorem 2.10 and $\overline{c}_B, \overline{C}_B$ the constants from Theorem 2.11. Let $J \geq 0$ and consider the space V_J with the norm $\|\nabla \cdot \|$. For all $g_J \in V_J^{\#}$,*

$$c_S \overline{c}_B \left(\|\nabla g_0\|^2 + \sum_{j=1}^{J} \sum_{i=1}^{\#\mathcal{K}_j} \frac{\langle g, \phi_i^{(j)} \rangle^2}{\|\nabla \phi_i^{(j)}\|^2} \right) \leq \|g_J\|_{V_J^{\#}}^2$$
$$\leq C_S \overline{C}_B \left(\|\nabla g_0\|^2 + \sum_{j=1}^{J} \sum_{i=1}^{\#\mathcal{K}_j} \frac{\langle g, \phi_i^{(j)} \rangle^2}{\|\nabla \phi_i^{(j)}\|^2} \right),$$

where $g_0 \in V_0$ is the Riesz representation function of the functional g in the space V_0 with respect to the inner product $(u_0, v_0) = \int_\Omega \nabla v_0 \cdot \nabla u_0$, $\forall u_0, v_0 \in V_0$.

Proof. The proof is analogous to the proof of Theorem 2.14. □

Acknowledgments

The authors wish to thank to Peter Oswald for his help on proving Theorem 2.8 and to Erin C. Carson for valuable comments, which improved the text.

Bibliography

[1] M. Ainsworth and J. T. Oden. “A posteriori error estimation in finite element analysis”. In: *Computer methods in applied mechanics and engineering* 142.1-2 (1997), pp. 1–88. DOI: https://doi.org/10.1016/S0045-7825(96)01107-3.

[2] M. S. Alnaes, J. Blechta, J. Hake, et al. “The FEniCS Project Version 1.5”. In: *Archive of Numerical Software* 3 (2015). DOI: 10.11588/ans.2015.100.20553.

[3] Z. Bai et al., eds. *Templates for the solution of algebraic eigenvalue problems.* Vol. 11. Software, Environments, and Tools. Philadelphia, PA: SIAM, 2000, pp. xxx+410. DOI: 10.1137/1.9780898719581.

[4] R. Becker, C. Johnson, and R. Rannacher. “Adaptive error control for multigrid finite element methods”. In: *Computing* 55.4 (1995), pp. 271–288. DOI: 10.1007/BF02238483.

[5] F. Bornemann and H. Yserentant. “A basic norm equivalence for the theory of multilevel methods”. In: *Numerische Mathematik* 64.1 (1993), pp. 455–476.

[6] A. Brandt. *Multigrid Techniques 1984 Guide with Applications to Fluid Dynamics Revised Edition.* Philadelphia, PA: SIAM, 2011. DOI: 10.1137/1.9781611970753.

[7] S. C. Brenner and L. R. Scott. *The Mathematical Theory of Finite Element Methods.* third. Vol. 15. Texts in Applied Mathematics. New York: Springer-Verlag, 2007.

[8] H. Brezis. *Functional analysis, Sobolev spaces and partial differential equations.* Universitext. New York: Springer, 2011.

[9] W. L. Briggs, V. E. Henson, and S. F. McCormick. *A Multigrid Tutorial*. Second. Philadelphia, PA: SIAM, 2000, pp. xii+193. DOI: 10.1137/1.9780898719505.

[10] A. Buttari et al. "Block low-rank single precision coarse grid solvers for extreme scale multigrid methods". In: *Numerical Linear Algebra with Applications* 29.1 (2022), e2407. DOI: 10.1002/nla.2407.

[11] C. Carstensen. "Quasi-interpolation and a posteriori error analysis in finite element methods". In: *M2AN Math. Model. Numer. Anal.* 33.6 (1999), pp. 1187–1202. DOI: 10.1051/m2an:1999140.

[12] P. G. Ciarlet. *The Finite Element Method for Elliptic Problems*. Amsterdam: North-Holland, 1978.

[13] P. Clément. "Approximation by finite element functions using local regularization". In: *RAIRO Analyse Numérique* 9.R2 (1975), pp. 77–84. DOI: 10.1051/m2an/197509R200771.

[14] W. Dahmen. "Wavelet and multiscale methods for operator equations". In: *Acta Numerica* 6 (1997), pp. 55–228. DOI: 10.1017/S0962492900002713.

[15] W. Dahmen and A. Kunoth. "Multilevel preconditioning". In: *Numerische Mathematik* 63.3 (1992), pp. 315–344. ISSN: 0029-599X. DOI: 10.1007/BF01385864. URL: https://doi.org/10.1007/BF01385864.

[16] H. C. Elman, D. J. Silvester, and A. J. Wathen. *Finite elements and fast iterative solvers: with applications in incompressible fluid dynamics*. Numerical mathematics and scientific computation. Oxford: Oxford University Press, 2005.

[17] I. Fried. "Bounds on the extremal eigenvalues of the finite element stiffness and mass matrices and their spectral condition number". In: *Journal of Sound and Vibration* 22.4 (1972), pp. 407–418. DOI: 10.1016/0022-460X(72)90452-X.

[18] G. H. Golub and Z. Strakoš. "Estimates in quadratic formulas". In: *Numer. Algorithms* 8.2-4 (1994), pp. 241–268. DOI: 10.1007/BF02142693.

[19] W. Hackbusch. *Iterative solution of large sparse systems of equations*. Second. Vol. 95. Applied Mathematical Sciences. Cham: Springer, 2016, pp. xxiii+509. DOI: 10.1007/978-3-319-28483-5.

[20] H. Harbrecht and R. Schneider. "A note on multilevel based error estimation". In: *Comput. Methods Appl. Math.* 16.3 (2016), pp. 447–458. DOI: 10.1515/cmam-2016-0013.

[21] H. Harbrecht, R. Schneider, and C. Schwab. "Multilevel frames for sparse tensor product spaces". In: *Numerische Mathematik* 110.2 (2008), p. 199.

[22] M. R. Hestenes and E. Stiefel. "Methods of conjugate gradients for solving linear systems". In: *Journal of Research of the National Bureau of Standards* 49.6 (1952), pp. 409–436. DOI: 10.6028/jres.049.044.

[23] M. Huber. "Massively parallel and fault-tolerant multigrid solvers on petascale systems". PhD thesis. Technical University of Munich, Germany, 2019. URL: http://www.dr.hut-verlag.de/978-3-8439-3917-1.html.

[24] A. Logg et al. *Automated Solution of Differential Equations by the Finite Element Method*. Springer, 2012. DOI: 10.1007/978-3-642-23099-8.

[25] S. F. McCormick, J. Benzaken, and R. Tamstorf. "Algebraic Error Analysis for Mixed-Precision Multigrid Solvers". In: *SIAM Journal on Scientific Computing* 43.5 (2021), S392–S419. DOI: 10.1137/20M1348571.

[26] G. Meurant and P. Tichý. "The behaviour of the Gauss-Radau upper bound of the error norm in CG". In: *Numerical Algorithms* 94 (2023), pp. 847–876. DOI: 10.1007/s11075-023-01522-z.

[27] A. Miraçi, J. Papež, and M. Vohralík. "A-posteriori-steered *p*-robust multigrid with optimal step-sizes and adaptive number of smoothing steps". In: *SIAM Journal on Scientific Computing* 43.5 (2021), S117–S145. DOI: 10.1137/20M1349503.

[28] Y. Notay. "Convergence analysis of perturbed two-grid and multigrid methods". In: *SIAM Journal on Numerical Analysis* 45.3 (2007), pp. 1035–1044. DOI: 10.1137/060652312.

[29] P. Oswald. *Multilevel finite element approximation*. Teubner Skripten zur Numerik. Theory and applications. Stuttgart: B. G. Teubner, 1994, p. 160. DOI: 10.1007/978-3-322-91215-2.

[30] J. Papež, Z. Strakoš, and M. Vohralík. "Estimating and localizing the algebraic and total numerical errors using flux reconstructions". In: *Numerische Mathematik* 138.3 (2018), pp. 681–721. DOI: 10.1007/s00211-017-0915-5.

[31] J. Papež et al. "Sharp algebraic and total a posteriori error bounds for *h* and *p* finite elements via a multilevel approach: recovering mass balance in any situation". In: *Comput. Methods Appl. Mech. Engrg.* 371 (2020), pp. 113243, 39. DOI: 10.1016/j.cma.2020.113243.

[32] K. Rektorys. *Variational methods in mathematics, science and engineering*. Second. Translated from the Czech by Michael Basch. Dordrecht-Holland, Boston, USA: D. Reidel Publishing Co., 1980, p. 571.

[33] U. Rüde. *Mathematical and computational techniques for multilevel adaptive methods*. Philadelphia, PA: SIAM, 1993.

[34] L. R. Scott and S. Zhang. "Finite element interpolation of nonsmooth functions satisfying boundary conditions". In: *Mathematics of Computation* 54.190 (1990), pp. 483–493.

[35] R. Stevenson. "Optimality of a standard adaptive finite element method". In: *Foundations of Computational Mathematics* 7.2 (2007), pp. 245–269.

[36] Z. Strakoš and P. Tichý. "Error estimation in preconditioned conjugate gradients". In: *BIT* 45.4 (2005), pp. 789–817.

[37] Z. Strakoš and P. Tichý. "On error estimation in the conjugate gradient method and why it works in finite precision computations." eng. In: *Electronic Transactions on Numerical Analysis* 13 (2002), pp. 56–80. URL: http://eudml.org/doc/123075.

[38] R. Tamstorf, J. Benzaken, and S. F. McCormick. "Discretization -Error-Accurate Mixed-Precision Multigrid Solvers". In: *SIAM Journal on Scientific Computing* 43.5 (2021), S420–S447. DOI: 10.1137/20M1349230.

[39] U. Trottenberg, C. W. Oosterlee, and A. Schuller. *Multigrid*. London: Academic Press, 2001.

[40] P. Vacek. “Multilevel methods”. Master’s thesis. Charles University, 2020. URL: `http://hdl.handle.net/20.500.11956/116819`.

[41] P. Vacek, E. Carson, and K. M. Soodhalter. “The Effect of Approximate Coarsest-Level Solves on the Convergence of Multigrid V-Cycle Methods”. In: *SIAM Journal on Scientific Computing* 46.4 (2024), A2634–A2659. DOI: `10.1137/23M1578255`.

[42] R. Verfürth. *A Posteriori Error Estimation Techniques for Finite Element Methods*. Numerical Mathematics and Scientific Computation. Oxford: Oxford University Press, 2013. DOI: `10.1093/acprof:oso/9780199679423.001.0001`.

[43] J. Xu. “Iterative methods by space decomposition and subspace correction”. In: *SIAM Review* 34.4 (1992), pp. 581–613. DOI: `10.1137/1034116`.

[44] X. Xu and C.-S. Zhang. “Convergence Analysis of Inexact Two-Grid Methods: A Theoretical Framework”. In: *SIAM Journal on Numerical Analysis* 60.1 (2022), pp. 133–156. DOI: `10.1137/20M1356075`.

[45] H. Yserentant. “Old and new convergence proofs for multigrid methods”. In: *Acta Numerica* 2 (1993), pp. 285–326.

[46] S. Zhang. “Successive subdivisions of tetrahedra and multigrid methods on tetrahedral meshes”. In: *Houston Journal of Mathematics* 21.3 (1995), pp. 541–556.

3 Mixed precision multigrid with smoothing based on incomplete Cholesky factorization

In the first two chapters we studied the effects of approximate computation on the coarsest-level on the convergence of the V-cycle method or on properties of multilevel a posteriori error estimates. For simplicity of the analysis, we assumed that the computation was done in infinite precision arithmetic.

In this chapter we focus on the effects of finite precision errors. In particular, we study a mixed precision V-cycle method with smoothing based on incomplete Cholesky (IC) factorizarion. Our effort is motivated by the fourth question stated in the introduction:

d) Can the execution time of the mixed precision V-cycle method with IC smoothers be reduced by introducing additional precisions for the applications of the smoothers? For example, using different precisions for storing the IC factors or solving the triangular systems. Can we analytically describe the requirements on these individual precisions?

We present a mixed precision formulation of the V-cycle scheme. Instead of assuming that a smoother or the coarsest-level solver is applied in a precision with certain unit roundoff, we impose an assumption on the finite precision error resulting from its application. This allow us to consider also mixed precision smoothers or coarsest-level solvers. We derive a bound on the finite precision error of the V-cycle scheme which gives insight into how the finite precision errors from the individual parts of the V-cycle scheme may affect the overall finite precision error. Further, we present a mixed precision formulation of the IC smoother and derive a bound on the finite precision error of its application. The theoretical results indicate that in some settings (depending on the properties of the IC factors) the IC smoother on a concrete level may be applied in a precision lower than the precision used for computing the residual, restriction, prolongation, and addition on the level.

We test the theoretical results on a series of numerical experiments, where we solve elliptic PDEs discretized using the finite element method. We run experiments with simulated floating point arithmetics in MATLAB and experiments on GPUs using the Ginkgo library. The results show that applying the IC smoother in low precision can yield a significant speed up in the computational time of the V-cycle method in comparison to its uniform double precision variant.

The text presented in this chapter is a result of a collaboration with H. Anzt, E. Carson, N. Kohl, U. Rüde and Y.-H. Tsai.

3.1 Introduction

Running large scale simulations on modern supercomputers requires significant energy and time resources. Research focused on reducing the energy footprint and optimizing the computation process ranges from manufacturing new hardware components to designing novel numerical methods. The two mentioned directions are heavily interconnected. Even though hardware is usually manufactured for a specific task, the introduction of a powerful hardware component can also influence the design of new mathematical methods capable of fully exploiting its potential. The introduction and availability of GPUs and subsequent effort on designing parallel numerical algorithms utilizing multiple floating point arithmetics is a good example. Many classic methods in numerical linear algebra can be redesigned to employ mixed precision approach by running parts of the computation in higher precisions and parts in a low precisions. In some cases it is possible to achieve the same overall accuracy in a smaller amount of time, requiring less memory and consuming less energy; see, e.g., the surveys [12, 1]. The improvements can be accomplished by both using low-precision computational units, which can perform more floating point operations per second, and by storing data in low precision formats, which enables faster memory movements.

In this text we study mixed precision variants of multigrid methods [27, 6] which are frequently used when solving systems of linear equations. Multigrid methods are applied both as a standalone solver and as a preconditioner for an iterative method. Even though they were historically introduced as a method for solving systems coming from discretization of elliptic PDEs they are nowadays being used in various settings. The computation relies on having a hierarchy of problems, which can be obtained either by discretizing a continuous problem on a multiple nested meshes (geometric multigrid) or constructed based on the properties of the system matrix (algebraic multigrid). The approximate solution is computed using so called smoothing on fine levels and by solving a system of linear equations on the coarsest-level. There are different multigrid schemes (V-cycle, W-cycle, full multigrid) varying in the pattern in which the individual levels are visited during the computation.

Implementations of multigrid methods employing different precision formats for different parts of the method are being developed and tested on various problems, see e.g., [29, 30, 28, 35]. A first finite precision error analysis of mixed precision multigrid methods was presented in [17] and further extended in [18]. The results were used by the authors in a paper focusing on achieving discretization error accuracy when solving elliptic PDEs [25] and adapted also when studying multigrid methods with block floating point arithmetic in [16].

The finite precision error analysis of the multigrid V-cycle method presented in [17, 18] is based on viewing the method as an iterative refinement method on the finest level with restarted V-cycle method as the inner solver. This point of view enables using three different finite precisions on the finest level. The authors further introduce “progressive” finite precisions associated with individual levels of the V-cycle method. They derive bound on the finite precision error of the V-cycle scheme and discuss requirements on the individual precisions based on the properties of the system and prolongation matrices and smoothing routines on the associated levels. The approach assumes that the smoothing routine on a concrete fine level or the coarsest-level solver is applied in one finite precision associated with the level. The authors discuss analysis of smoothing routines based on the Richardson and Jacobi methods within this framework.

Multigrid methods are in practice also applied with more computationally intensive smoothers. Smoothing routines based on incomplete Cholesky (IC) factorization are, for example, used when solving elliptic PDEs with large anisotropy; see e.g. early papers [15, 32, 31, 14] or [9, 26]. To use the IC smoothing, the IC factorization has to be precomputed. Its application then requires solving triangular systems with the IC factor and its transpose. Using the IC smoother may thus require more computational resources than other simpler smoothers such as smoothers based on the Richardson or Jacobi methods.

It is therefore a valid question to ask whether the mixed precision approach could be used to speed up the application of the IC smoothers, by computing the IC factorization in low precision, and/or storing the IC factor in low-precision, and/or by solving the triangular systems in low-precision. This opens a series of questions. What precisions should be used in the mentioned stages of the IC smoother? How should these precisions be chosen with respect to the application of the smoother inside the V-cycle method?

Motivated by these questions, we present a formulation of the V-cycle scheme with general assumptions on the smoothers and the coarsest-level solver. Rather than assuming that the smoothers and the coarsest-level solver are applied in a certain precision, we impose assumptions on the finite precision errors resulting from their applications. Inspired by the papers [17, 18], we present a finite precision analysis of the formulated V-cycle correction schemes. Our approach enables the analysis of multigrid methods with general (mixed precision) smoothers and coarsest-level solvers.

We further formulate a mixed precision IC smoothing routine and present a bound on the finite precision error on its application. We assume that the triangular problems are solved using substitution. We do not take into account the finite precision error coming from computing the IC factorization in finite precision.

We test the theoretical results and performance of the presented methods through a series of numerical experiments. We solve systems coming from finite element (FE) discretization of elliptic PDEs. We run experiments with simulated floating point arithmetics in MATLAB using the Advanpix toolbox [2] as well as experiments performed on GPUs using the Ginkgo library [5, 8].

The paper is organized as follows. In Section 3.2, we establish the notation, present the standard rounding model, and state bounds on the finite precision errors in basic vector and matrix operations. Section 3.3 contains the description of the mixed precision iterative refinement method and a summary of results on its convergence in the energy norm. The mixed precision two-grid correction scheme is presented in Section 3.4 together with its finite precision error analysis. These results are generalized to a multigrid V-cycle correction scheme in Section 3.5. In Section 3.6, we present a mixed precision smoothing routine based on incomplete Cholesky factorization and derive a bound on finite precision errors occurring in its application. Section 3.7 contains simple strategies for scaling the matrices and right-hand side vectors to help avoid out of range results when using low precision formats. We illustrate the theoretical results on a series of numerical experiments in Section 3.8. The text closes with conclusions and a summary of related open problems in Section 3.9.

3.2 Model problem, notation, finite precision arithmetic and standard rounding model

We consider all vectors and matrices in this paper to be real. We denote the Euclidean inner product as $\langle \cdot, \cdot \rangle$, and the Euclidean vector or matrix norm as $\| \cdot \|$. For a symmetric positive definite (SPD) matrix $\mathbf{A}$, we denote the $\mathbf{A}$ vector norm of a vector $\mathbf{v}$ as $\|\mathbf{v}\|_{\mathbf{A}} = \sqrt{\langle \mathbf{A}\mathbf{v}, \mathbf{v} \rangle}$; we use the same notation for the associated matrix norm. Throughout the text we use the following relations between the Euclidean vector norm and the $\mathbf{A}$ vector norm without explicitly commenting on their use. For any vector $\mathbf{v}$ it holds that (see Appendix 3.10.1)

$$\begin{aligned} \|\mathbf{v}\|_{\mathbf{A}} &\leq \|\mathbf{A}\|^{\frac{1}{2}} \|\mathbf{v}\|, \\ \|\mathbf{v}\| &\leq \|\mathbf{A}^{-1}\|^{\frac{1}{2}} \|\mathbf{v}\|_{\mathbf{A}}, \\ \|\mathbf{A}\mathbf{v}\| &\leq \|\mathbf{A}\|^{\frac{1}{2}} \|\mathbf{v}\|_{\mathbf{A}}, \\ \|\mathbf{A}^{-1}\mathbf{v}\|_{\mathbf{A}} &\leq \|\mathbf{A}^{-1}\|^{\frac{1}{2}} \|\mathbf{v}\|. \end{aligned}$$

Let $\mathbf{K}$ be an invertible matrix and let $|\mathbf{K}|$ denote the matrix with the component-wise absolute values of the entries of matrix $\mathbf{K}$. By κ_K we denote the condition number of $\mathbf{K}$ and by $\underline{\kappa}_K$ a variant of the condition number containing $\||\mathbf{K}|\|$ instead of $\|\mathbf{K}\|$, i.e.,

$$\kappa_K = \|\mathbf{K}^{-1}\| \|\mathbf{K}\|, \quad \underline{\kappa}_K = \|\mathbf{K}^{-1}\| \||\mathbf{K}|\|.$$

We consider the standard model for accounting for finite precision errors, which is also used in the existing finite precision analysis of multigrid methods in [17, 18]. For an introduction to the analysis of finite precision errors in numerical methods, see, e.g., [11].

Consider a floating point arithmetic with unit roundoff ε. Instead of saying that a computation was done in a floating point arithmetic with unit roundoff ε or that a vector or a matrix was rounded to a floating point arithmetic with unit roundoff ε, for simplicity we write computed in, or rounded to ε-precision. Let x be a number within the range of the ε-precision. We assume that rounding x to ε-precision results in

$$x + \delta, \quad |\delta| \leq \varepsilon |x|.$$

Let $\circ$ denote one of the basic scalar operations, i.e., addition, subtraction, multiplication, or division. Let x and y be two numbers in the ε-precision arithmetic. Assuming that $x \circ y$ is in the range of the ε-precision arithmetic, we assume that computing $x \circ y$ in ε-precision results in

$$x \circ y + \delta, \quad |\delta| \leq \varepsilon |x \circ y|.$$

Further, when using this model, we always assume that the inputs and outputs are within the range of the considered finite precision arithmetic, i.e., the computation does not break down due to overflow or underflow, which is a standard assumption in the literature. We comment on practical issues regarding underflow and overflow later in Section 3.7. In the text we use the hat symbol to highlight that a term is computed in finite precision arithmetic; for example, for $s = x \circ y$ computed in ε-precision, we write $\hat{s} = s + \delta$, $|\delta| \leq \varepsilon |s|$.

Based on the model the following results, which will be used below, can be shown; see, e.g., [18, Section 2] and [17, Section 2]. Let $\mathbf{v}$ and $\mathbf{w}$ be two vectors and let $\mathbf{K}$ be a matrix. Let m_K denote the maximum number of nonzero entries in a row of $\mathbf{K}$ and let $\underline{m}_K$ denote the maximum number of nonzero entries in a row or a column of $\mathbf{K}$. The constants $m_{K,\varepsilon}$ and $\underline{m}_{K,\varepsilon}$ are defined as

$$m_{K,\varepsilon} = \frac{m_K}{1 - m_K\varepsilon}, \qquad \underline{m}_{K,\varepsilon} = \frac{\underline{m}_K}{1 - \underline{m}_K\varepsilon}.$$

Rounding the vector $\mathbf{v}$ to ε-precision arithmetic results in

$$\mathbf{v} + \delta, \quad \|\delta\| \leq \varepsilon\|\mathbf{v}\|. \tag{3.1}$$

Rounding the matrix $\mathbf{K}$ to ε-precision results in

$$\mathbf{K} + \Delta\mathbf{K}, \quad |\Delta\mathbf{K}| \leq \varepsilon|\mathbf{K}|, \tag{3.2}$$

where the inequality is understood entry by entry.

Assuming the entries of $\mathbf{v}$ and $\mathbf{w}$ and $\mathbf{K}$ belong to the ε-precision arithmetic, computing $\mathbf{v} + \mathbf{w}$ and $\mathbf{Kw}$ both in ε-precision results in, respectively,

$$\mathbf{v} + \mathbf{w} + \delta, \quad \|\delta\| \leq \varepsilon\|\mathbf{v} + \mathbf{w}\|, \tag{3.3}$$

$$\mathbf{Kw} + \delta, \quad \|\delta\| \leq \varepsilon \underline{m}_{K,\varepsilon}\||\mathbf{K}|\|\|\mathbf{w}\|. \tag{3.4}$$

Assume that $\mathbf{v}$ and $\mathbf{w}$ belong to the ε-precision arithmetic. Let $\mathbf{K} + \Delta\mathbf{K}$ be the matrix obtained by rounding matrix $\mathbf{K}$ to ε-precision, i.e., $|\Delta\mathbf{K}| \leq \varepsilon|\mathbf{K}|$. Computing $(\mathbf{K} + \Delta\mathbf{K})\mathbf{w}$ and $\mathbf{v} - (\mathbf{K} + \Delta\mathbf{K})\mathbf{w}$ both in ε-precision results in (see Appendix 3.10.2), respectively,

$$\mathbf{Kw} + \delta, \quad \|\delta\| \leq (\varepsilon(\underline{m}_{K,\varepsilon} + 1) + \varepsilon^2 \underline{m}_{K,\varepsilon})\||\mathbf{K}|\|\|\mathbf{w}\|, \tag{3.5}$$

$$\mathbf{v} - \mathbf{Kw} + \delta, \quad \|\delta\| \leq (\varepsilon(\underline{m}_{K,\varepsilon} + 2) + \varepsilon^2(2\underline{m}_{K,\varepsilon} + 1 + \varepsilon \underline{m}_{K,\varepsilon})))(\|\mathbf{v}\| + \||\mathbf{K}|\|\|\mathbf{w}\|). \tag{3.6}$$

3.3 Iterative refinement

In this section, we present the mixed precision iterative refinement (IR) method (see, e.g., [7, 17]) for computing an approximate solution to the problem

$$\mathbf{Ax} = \mathbf{b},$$

where $\mathbf{A} \in \mathbb{R}^{n\times n}$ is a sparse SPD matrix with a maximum of m_A nonzero elements per row, and $\mathbf{b} \in \mathbb{R}^n$. We have in mind that the matrix $\mathbf{A}$ may be coming from a discretization of an elliptic PDE. In such a case, the matrix is rounded to a floating point arithmetic. We assume that both $\mathbf{A}$ and $\mathbf{b}$ are rounded to a floating point arithmetic with unit roundoff $\check{\varepsilon}$, resulting in $\check{\mathbf{A}} = \mathbf{A} + \check{\Delta}\mathbf{A}$, where $|\check{\Delta}\mathbf{A}| \leq \check{\varepsilon}|\mathbf{A}|$ and $\check{\mathbf{b}} = \mathbf{b} + \check{\Delta}\mathbf{b}$, where $|\check{\Delta}\mathbf{b}| \leq \check{\varepsilon}|\mathbf{b}|$.

We consider the IR method described in Algorithm 3.1, cf. [17, 25]. We assume that the residual is computed in a precision with a unit roundoff $\bar{\varepsilon}$, where $\bar{\varepsilon} \leq \check{\varepsilon}$. We assume that the application of the inner solver in finite precision reduces the $\mathbf{A}$-norm of the error by a least a factor $\rho < 1$, i.e., for any right-hand side $\mathbf{r}$ the approximate solution $\hat{\mathbf{y}}_{\mathrm{in}}$ of $\mathbf{Ay} = \mathbf{r}$ computed using the inner solver in finite precision satisfies

$$\|\mathbf{y} - \hat{\mathbf{y}}_{\mathrm{in}}\|_{\mathbf{A}} \leq \rho \|\mathbf{y}\|_{\mathbf{A}}.$$

We assume that the computed approximation $\hat{\mathbf{y}}_{\mathrm{in}}$ belongs a precision with a unit roundoff ε, where $\varepsilon \geq \bar{\varepsilon}$. Note that there are no explicit assumptions on the finite precision arithmetic used inside the inner solver. The requirements on the finite precision are implicitly included in the assumption on the reduction of the **A**-norm of the error. In order to use Algorithm 3.1 we also need to specify a number of iterations N and a stopping criterion.

Algorithm 3.1 Iterative refinement, $\mathbf{IR}(\check{\mathbf{b}}, \hat{\mathbf{x}}^{(0)}, N)$.

```
for $i = 0, 1, \ldots, N-1$ do
    $\hat{\mathbf{r}} = \check{\mathbf{b}} - \check{\mathbf{A}}\hat{\mathbf{x}}^{(i)}$ {Compute residual in $\bar{\varepsilon}$-precision.}
    if stopping criterion is satisfied then
        return $\hat{\mathbf{x}}^{(i)}$
    end if
    $\hat{\mathbf{y}}_{\mathrm{in}}$ = InnerSolver($\hat{\mathbf{r}}$) {Approximately solve $\mathbf{Ay} = \mathbf{r}$.}
    $\hat{\mathbf{x}}^{(i+1)} = \hat{\mathbf{x}}^{(i)} + \hat{\mathbf{y}}_{\mathrm{in}}$ {Correct the approximation in $\varepsilon$-precision.}
end for
% the stopping criterion was not satisfied after $N$ iterations
```

The convergence of the IR method in the **A**-norm including finite precision errors was studied in [17, 25]. The convergence result derived in [25, Theorem 4.5] reads as follows; we refer also to the discussion in [17, Remark 4.2]. Let $\hat{\mathbf{x}}^{\mathrm{new}}$ be the approximate solution of $\mathbf{Ax} = \mathbf{b}$ computed after one iteration of the IR method Algorithm 3.1 using finite precision computation starting with an approximation $\mathbf{x}^{\mathrm{prev}}$ belonging to the $\bar{\varepsilon}$-precision arithmetic. One iteration of the IR method reduces the relative **A**-norm of the error by a least a factor, $\rho + \delta_{\mathrm{IR}}$, up to an absolute additional limiting factor χ, i.e.,

$$\frac{\|\mathbf{x} - \hat{\mathbf{x}}^{\mathrm{new}}\|_{\mathbf{A}}}{\|\mathbf{x}\|_{\mathbf{A}}} \leq (\rho + \delta_{\mathrm{IR}}) \frac{\|\mathbf{x} - \mathbf{x}^{\mathrm{prev}}\|_{\mathbf{A}}}{\|\mathbf{x}\|_{\mathbf{A}}} + \chi.$$

The term δ_{IR} can be seen as a bound on a potential delay in the rate of convergence occurring due to finite precision errors and the term χ as the bound on the limiting accuracy. Assuming that κ_A and $\underline{\kappa}_A$ are approximately the same and that $\check{\varepsilon}$ and $\bar{\varepsilon}$ are small enough such that $\check{\varepsilon}\kappa_A \ll 1$ and $m_A \bar{\varepsilon} \kappa_A \ll 1$, the bound on the delay of convergence δ_{IR} and the bound on the limiting accuracy χ are on the order of $\varepsilon \kappa_A^{\frac{1}{2}}$.

3.4 Two-grid correction scheme

In this section, we present a mixed precision formulation of the two-grid (TG) correction scheme and its rounding error analysis. For an introduction to multigrid methods we refer to, e.g., [6, 27].

We consider the TG correction scheme for computing an approximate solution of $\mathbf{Ay} = \mathbf{f}$, where $\mathbf{f} \in \mathbb{R}^n$. The TG correction scheme is based on having two formulations of the problem, the original problem with matrix $\mathbf{A}$ which is called the fine-grid problem and a projection of the original problem to a smaller dimensional subspace called the coarse-grid problem. The approximate solution is computed using so-called smoothing on the fine grid and a so-called coarse-grid correction. We describe the mentioned routines, state assumptions on their parts, and subsequently formulate the TG correction scheme.

Smoothing usually consists of applying few iterations of a stationary iterative method. One iteration of such a method consists of computing the residual, solving an error equation with a simple approximation of the matrix $\mathbf{A}$, and correcting the previous approximation. For simplicity of the forthcoming rounding error analysis, we consider only one smoothing iteration starting with a zero initial approximation. The smoothing thus reduces to solving an error equation where the matrix $\mathbf{A}$ is replaced by its simple approximation.

We assume that the application of smoothing in infinite precision can be for any vector $\mathbf{f}$ written as $\mathbf{Mf}$, where $\mathbf{M} \in \mathbb{R}^{n\times n}$ is a non-singular matrix approximating the inverse of $\mathbf{A}$ in the sense

$$\|\mathbf{I} - \mathbf{MA}\|_{\mathbf{A}} < 1, \tag{3.7}$$

where $\mathbf{I}$ denotes the identity matrix. This is a standard assumption in the multigrid literature; see, e.g., [17, Section 5], [34, p. 293] or [33].

We assume that before the application of the smoother, the right-hand side is rounded to a finite precision with unit roundoff $\dot{\varepsilon}$. Rather than assuming that the smoothing is applied in finite precision with a certain unit roundoff, we assume that there exists a positive constant Λ_M such that the application of the smoother in finite precision for any vector $\mathbf{f}$ results in

$$\mathbf{Mf} + \delta_M, \quad \|\delta_M\| \leq \Lambda_M \|\mathbf{f}\|. \tag{3.8}$$

We assume that the resulting vector belongs to the $\dot{\varepsilon}$-precision arithmetic. This modular approach allows us to also analyze mixed precision smoothers.

The coarse-grid correction consists of computing the residual, restricting it to the coarse-grid, applying the coarse-grid solver to an error equation on the coarse-grid, prolongation of the computed correction to the fine-grid, and correcting the previous approximation. We consider that the residual is computed in $\dot{\varepsilon}$-precision and that the matrix $\mathbf{A}$ is rounded to $\dot{\varepsilon}$-precision for the residual computation, resulting in $\dot{\mathbf{A}} = \mathbf{A} + \dot{\Delta}\mathbf{A}$, where $|\dot{\Delta}\mathbf{A}| \leq \dot{\varepsilon}|\mathbf{A}|$.

We assume that there exists an SPD coarse-grid matrix $\mathbf{A}_{\mathrm{C}} \in \mathbb{R}^{n_{\mathrm{C}}\times n_{\mathrm{C}}}$, and a full rank prolongation matrix $\mathbf{P} \in \mathbb{R}^{n\times n_{\mathrm{C}}}$, such that the Galerkin condition is satisfied, i.e.,

$$\mathbf{A}_{\mathrm{C}} = \mathbf{P}^\top \mathbf{A} \mathbf{P}. \tag{3.9}$$

We assume that the restriction matrix is transpose of the prolongation matrix, and that both the restriction and prolongation are performed in $\dot{\varepsilon}$-precision. We further assume that when computing with the prolongation matrix, it is rounded to $\dot{\varepsilon}$-precision, resulting in $\dot{\mathbf{P}} = \mathbf{P} + \dot{\Delta}\mathbf{P}$, where $|\dot{\Delta}\mathbf{P}| \leq \dot{\varepsilon}|\mathbf{P}|$. Note that in geometric multigrid, the Galerkin condition may hold only in infinite precision.

We assume that for any vector $\mathbf{f}_{\mathrm{C}}$, the application of the coarse-grid solver in infinite precision can be written as $\mathbf{M}_{\mathrm{C}}\mathbf{f}_{\mathrm{C}}$, where $\mathbf{M}_{\mathrm{C}}$ is a non-singular matrix approximating the inverse of $\mathbf{A}_{\mathrm{C}}$ in the sense that

$$\|\mathbf{I}_{\mathrm{C}} - \mathbf{M}_{\mathrm{C}}\mathbf{A}_{\mathrm{C}}\|_{\mathbf{A}_{\mathrm{C}}} < 1, \tag{3.10}$$

where $\mathbf{I}_{\mathrm{C}}$ denotes the identity matrix on the coarse level. We note that the standard TG correction scheme assumes an exact coarse-grid solve, i.e., $\mathbf{M}_{\mathrm{C}} = \mathbf{A}_{\mathrm{C}}^{-1}$, for which the assumption (3.10) is automatically satisfied. The formulation here allows for using approximate linear solvers. The TG scheme is generalized to the multilevel V-cycle scheme in the next section.

To take into account the finite precision errors which may occur during the coarse-level solve, we assume that there exists a positive constant Λ_{C} such that the application of the coarse-grid solver in finite precision for any vector $\mathbf{f}_{\mathrm{C}}$ results in

$$\mathbf{M}_{\mathrm{C}}\mathbf{f}_{\mathrm{C}} + \delta_{\mathrm{C}}, \quad \|\delta_{\mathrm{C}}\|_{\mathbf{A}_{\mathrm{C}}} \leq \Lambda_{\mathrm{C}}\|\mathbf{A}_{\mathrm{C}}^{-1}\mathbf{f}_{\mathrm{C}}\|_{\mathbf{A}_{\mathrm{C}}}. \tag{3.11}$$

We assume that the computed approximation belongs to the $\dot{\varepsilon}$-precision arithmetic and that the addition of the prolonged correction is done in $\dot{\varepsilon}$-precision.

The TG correction scheme is formulated in Algorithm 3.2. For simplicity, we consider a version where smoothing is done only before the coarse-level correction. This version is called a TG correction scheme with pre-smoothing in the literature. There are other versions where smoothing is done both before and after the coarse-level correction (TG scheme with pre- and post-smoothing) or just after the coarse-level correction (TG scheme with post-smoothing); see, e.g., [6, 27].

Algorithm 3.2 Two-grid correction scheme, $\mathbf{TG}(\mathbf{f})$.

```
$\hat{\mathbf{f}} = \mathrm{Round}(\mathbf{f}, \dot{\varepsilon})$ {Round the right-hand side $\mathbf{f}$ to $\dot{\varepsilon}$-precision.}
$\hat{\mathbf{v}}^{[1]} = \mathbf{M}\hat{\mathbf{f}}$ {Apply smoothing.}
$\hat{\mathbf{r}}^{[1]} = \hat{\mathbf{f}} - \dot{\mathbf{A}}\hat{\mathbf{v}}^{[1]}$ {Compute residual in $\dot{\varepsilon}$-precision.}
$\hat{\mathbf{r}}_{\mathrm{C}}^{[1]} = \dot{\mathbf{P}}^{\top}\hat{\mathbf{r}}^{[1]}$ {Restrict the residual to the coarse grid in $\dot{\varepsilon}$-precision.}
$\hat{\mathbf{v}}_{\mathrm{C}}^{[2]} = \mathbf{M}_{\mathrm{C}}\hat{\mathbf{r}}_{\mathrm{C}}^{[1]}$ {Approximately solve $\mathbf{A}_{\mathrm{C}}\mathbf{v}_{\mathrm{C}} = \mathbf{r}_{\mathrm{C}}^{[1]}$.}
$\hat{\mathbf{v}}^{[2]} = \dot{\mathbf{P}}\hat{\mathbf{v}}_{\mathrm{C}}^{[2]}$ {Prolongate the correction to the fine level in $\dot{\varepsilon}$-precision.}
$\hat{\mathbf{v}}^{[3]} = \hat{\mathbf{v}}^{[1]} + \hat{\mathbf{v}}^{[2]}$ {Correct the previous approximation in $\dot{\varepsilon}$-precision.}
return $\hat{\mathbf{y}}_{\mathrm{TG}} = \hat{\mathbf{v}}^{[3]}$
```

Let $\mathbf{y}_{\mathrm{TG}}$ be the approximation computed by Algorithm 3.2 in infinite precision, by which we mean that all computations are done exactly without rounding errors and the matrices $\mathbf{A}$ and $\mathbf{P}$ and vector $\mathbf{f}$ are not rounded. We assume that the two-grid correction scheme applied in infinite precision reduces the $\mathbf{A}$-norm of the error, i.e., there exists a constant $\rho_{\mathrm{TG}} < 1$ such that

$$\|\mathbf{y} - \mathbf{y}_{\mathrm{TG}}\|_{\mathbf{A}} \leq \rho_{\mathrm{TG}}\|\mathbf{y}\|_{\mathbf{A}}. \tag{3.12}$$

For convergence analysis of multigrid methods in infinite precision see, e.g., [34, 27, 20, 19].

We present the following result on the effects of finite precision errors on the convergence of the TG correction scheme. Its proof can be found below.

Theorem 3.1. *Let $\mathbf{y}_{\mathrm{TG}}$ and $\hat{\mathbf{y}}_{\mathrm{TG}}$ be the approximate solution of $\mathbf{A}\mathbf{y} = \mathbf{f}$ computed using the TG correction scheme applied in infinite and finite precision, respectively. Then*

$$\|\mathbf{y}_{\mathrm{TG}} - \hat{\mathbf{y}}_{\mathrm{TG}}\|_{\mathbf{A}} \leq \delta_{\mathrm{TG}}\|\mathbf{y}\|_{\mathbf{A}}, \tag{3.13}$$

$$\|\mathbf{y} - \hat{\mathbf{y}}_{\mathrm{TG}}\|_{\mathbf{A}} \leq (\rho_{\mathrm{TG}} + \delta_{\mathrm{TG}})\|\mathbf{y}\|_{\mathbf{A}}, \tag{3.14}$$

and δ_{TG} can be expressed as

$$\delta_{\mathrm{TG}} = \Lambda_{\mathrm{C}} + 3\|\mathbf{A}\|\Lambda_M + \dot{\varepsilon}\kappa_A^{\frac{1}{2}}(C_1|||\mathbf{A}|||\|\mathbf{M}\| + C_2) + 3\dot{\varepsilon}\|\mathbf{A}\|\|\mathbf{M}\| + R,$$

where C_1 and C_2 are positive constants depending only on $\|\mathbf{P}\|$, $|||\mathbf{P}|||$, $m_{A,\dot{\varepsilon}}$, $\underline{m}_{P,\dot{\varepsilon}}$ and the ratio $\|\mathbf{A}_{\mathrm{C}}^{-1}\|^{\frac{1}{2}}/\|\mathbf{A}^{-1}\|^{\frac{1}{2}}$. The quantity R contains higher order terms, i.e., terms which involve at least second powers of at least one of $\dot{\varepsilon}$, Λ_{C}, Λ_M, or a product of at least two of them. The TG correction scheme reduces the $\mathbf{A}$-norm of the error if $\rho_{\mathrm{TG}} + \delta_{\mathrm{TG}} < 1$.

We intentionally present this theorem without any additional assumptions on the individual terms in the estimates so that it can be used in various different settings. An even more detailed expression with explicit formulas for C_1 and C_2 can be found inside the proof if needed.

An important feature of the estimate is that it provides insight into how the finite precision errors coming from applying the smoother or the coarse-level solver may affect the overall finite precision error. In particular, we see that the bound on the relative finite precision error of the coarsest-level solver Λ_{C} is present as a standalone term. The bound on the relative finite precision error of the smoother is present multiplied by $3\|\mathbf{A}\|$, but not for example by $\|\mathbf{A}^{-1}\|^{\frac{1}{2}}$ or $\kappa_A^{\frac{1}{2}}$. Another useful observation is that the choice of the smoother may affect the requirements on the $\dot{\varepsilon}$-precision through the term $\|\mathbf{M}\|$. The larger the term $\|\mathbf{M}\|$ is, the smaller $\dot{\varepsilon}$ may have to be in order to have a sufficiently small finite precision error.

We generalize the result of this theorem to a multigrid V-cycle correction scheme in the next section.

Proof of Theorem 3.1. The proof is inspired by the proofs of [18, Theorem 1], [17, Theorem 7.2], and [25, Theorem 4.5].

We let $\mathbf{v}^{[1]}$, $\mathbf{r}^{[1]}$, $\mathbf{r}_{\mathrm{C}}^{[1]}$, $\mathbf{v}_{\mathrm{C}}^{[2]}$, $\mathbf{v}^{[2]}$, and $\mathbf{v}^{[3]}$ denote the infinite-precision counterparts of the terms in Algorithm 3.2, i.e., the terms that result when all computations are done exactly without finite precision errors and the matrices $\mathbf{A}$ and $\mathbf{P}$ and the vector $\mathbf{f}$ are not rounded.

We first present a series of bounds which are used below. For any fine-level vector $\mathbf{v}$, it holds that

$$\|\mathbf{A}_{\mathrm{C}}^{-1}\mathbf{P}^{\top}\mathbf{v}\|_{\mathbf{A}_{\mathrm{C}}} \leq \|\mathbf{A}^{-1}\mathbf{v}\|_{\mathbf{A}}. \tag{3.15}$$

This is a key inequality for the derivation below. As we will see, it is a consequence of the orthogonality of a certain projection. If we consider $\mathbf{v}$ to be a right-hand side vector on the fine level, the above inequality can be interpreted as the $\mathbf{A}_{\mathrm{C}}$-norm of the approximation to the solution computed on the coarse level being less than the $\mathbf{A}$-norm of the solution.

We will also make use of the following bound involving the matrix representing the coarse-level solver:

$$\|\mathbf{M}_{\mathrm{C}}\mathbf{A}_{\mathrm{C}}\|_{\mathbf{A}_{\mathrm{C}}} < 2, \tag{3.16}$$

and the following bounds of the norms of intermediate results in Algorithm 3.2:

$$\|\mathbf{v}^{[3]}\|_{\mathbf{A}} \leq 2\|\mathbf{y}\|_{\mathbf{A}}, \tag{3.17}$$

$$\|\mathbf{A}^{-1}\mathbf{r}^{[1]}\|_{\mathbf{A}} \leq \|\mathbf{y}\|_{\mathbf{A}}, \tag{3.18}$$

$$\|\mathbf{r}^{[1]}\| \leq \|\mathbf{A}\|^{\frac{1}{2}}\|\mathbf{y}\|_{\mathbf{A}}, \tag{3.19}$$

$$\|\mathbf{A}_{\mathrm{C}}^{-1}\mathbf{r}_{\mathrm{C}}^{[1]}\|_{\mathbf{A}_{\mathrm{C}}} \leq \|\mathbf{y}\|_{\mathbf{A}}, \tag{3.20}$$

$$\|\mathbf{v}_{\mathrm{C}}^{[2]}\|_{\mathbf{A}_{\mathrm{C}}} \leq 2\|\mathbf{y}\|_{\mathbf{A}}. \tag{3.21}$$

Variants of the bounds (3.15)-(3.21) can be found, e.g., in [18]. We include their derivation in Appendix 3.10.3 for self consistency of the text.

We focus on deriving a bound on the **A**-norm of the error caused by computation in finite precision arithmetic in the TG correction scheme (3.13), i.e.,

$$\|\mathbf{y}_{\mathrm{TG}} - \hat{\mathbf{y}}_{\mathrm{TG}}\|_{\mathbf{A}} \leq \delta_{\mathrm{TG}}\|\mathbf{y}\|_{\mathbf{A}}.$$

Analogously as in the proof of [18, Theorem 1], we go line by line in Algorithm 3.2 and bound the finite precision errors. Our goal is to derive a bound on the relative error in the **A**-norm. Since some of the assumptions or bounds we use contain the Euclidean norm and some the **A**-norm, we switch between these norms in the derivation.

Line 1: Rounding $\mathbf{f}$ to $\dot{\varepsilon}$-precision results in $\hat{\mathbf{f}} = \mathbf{f} + \delta_f$, where, using (3.1),

$$\|\delta_f\| \leq \dot{\varepsilon}\|\mathbf{f}\| = \dot{\varepsilon}\|\mathbf{A}\mathbf{y}\| \leq \underbrace{\dot{\varepsilon}\|\mathbf{A}\|^{\frac{1}{2}}}_{K_0}\|\mathbf{y}\|_{\mathbf{A}}. \tag{3.22}$$

Line 2: Applying the smoothing to $\hat{\mathbf{f}} = \mathbf{f} + \delta_f$ in finite precision results in $\mathbf{M}(\mathbf{f} + \delta_f) + \delta_{v^{[1]}}$, where, using the assumption (3.8) and (3.22),

$$\|\delta_{v^{[1]}}\| \leq \Lambda_M(\|\mathbf{f}\| + \|\delta_f\|) \leq \underbrace{(\Lambda_M\|\mathbf{A}\|^{\frac{1}{2}} + \Lambda_M K_0)}_{K_1}\|\mathbf{y}\|_{\mathbf{A}}. \tag{3.23}$$

The computed term $\hat{\mathbf{v}}^{[1]}$ can be written as $\hat{\mathbf{v}}^{[1]} = \mathbf{v}^{[1]} + \Delta_{v^{[1]}}$, where $\Delta_{v^{[1]}} = \mathbf{M}\delta_f + \delta_{v^{[1]}}$ is the accumulated error and, using (3.22) and (3.23),

$$\|\Delta_{v^{[1]}}\| \leq \|\mathbf{M}\delta_f\| + \|\delta_{v^{[1]}}\| \leq \underbrace{(\|\mathbf{M}\|K_0 + K_1)}_{K_2}\|\mathbf{y}\|_{\mathbf{A}}. \tag{3.24}$$

Line 3: Computing $(\mathbf{f} + \delta_f) - (\mathbf{A} + \dot{\Delta}\mathbf{A})(\mathbf{v}^{[1]} + \Delta_{v^{[1]}})$ in $\dot{\varepsilon}$-precision results in $\mathbf{f} + \delta_f - \mathbf{A}(\mathbf{v}^{[1]} + \Delta_{v^{[1]}}) + \delta_{r^{[1]}}$, where, using (3.6), $\mathbf{v}^{[1]} = \mathbf{M}\mathbf{f}$, (3.22), and (3.24),

$$\begin{aligned}\|\delta_{r^{[1]}}\| &\leq (\dot{\varepsilon}(m_{A,\dot{\varepsilon}} + 2) + \dot{\varepsilon}^2\underbrace{(2m_{A,\dot{\varepsilon}} + 1 + \dot{\varepsilon}m_{A,\dot{\varepsilon}})}_{K_\alpha})(\|\mathbf{f} + \delta_f\| + \||\mathbf{A}|\| \cdot \|\mathbf{v}^{[1]} + \Delta_{v^{[1]}}\|)\\ &\leq (\dot{\varepsilon}(m_{A,\dot{\varepsilon}} + 2) + \dot{\varepsilon}^2 K_\alpha)(\|\mathbf{f}\| + \|\delta_f\| + \||\mathbf{A}|\|(\|\mathbf{v}^{[1]}\| + \|\Delta_{v^{[1]}}\|))\\ &\leq \underbrace{(\dot{\varepsilon}(m_{A,\dot{\varepsilon}} + 2) + \dot{\varepsilon}^2 K_\alpha)(\|\mathbf{A}\|^{\frac{1}{2}} + K_0 + \||\mathbf{A}|\|(\|\mathbf{M}\|\|\mathbf{A}\|^{\frac{1}{2}} + K_2))}_{K_3}\|\mathbf{y}\|_{\mathbf{A}}.\end{aligned} \tag{3.25}$$

The computed term $\hat{\mathbf{r}}^{[1]}$ can be written as $\hat{\mathbf{r}}^{[1]} = \mathbf{r}^{[1]} + \Delta_{r^{[1]}}$, where

$$\Delta_{r^{[1]}} = \delta_f - \mathbf{A}\Delta_{v^{[1]}} + \delta_{r^{[1]}}$$

is the accumulated error, and using (3.22), (3.24), and (3.25),

$$\begin{aligned}\|\Delta_{r^{[1]}}\| &= \|\delta_f - \mathbf{A}\Delta_{v^{[1]}} + \delta_{r^{[1]}}\| \le \|\delta_f\| + \|\mathbf{A}\|\|\Delta_{v^{[1]}}\| + \|\delta_{r^{[1]}}\| \\ &\le \underbrace{(K_0 + \|\mathbf{A}\|K_2 + K_3)}_{K_4}\|\mathbf{y}\|_{\mathbf{A}}. \qquad (3.26)\end{aligned}$$

Line 4: Computing $(\mathbf{P} + \dot{\Delta}\mathbf{P})^\top(\mathbf{r}^{[1]} + \Delta_{r^{[1]}})$ in $\dot{\varepsilon}$-precision results in $\mathbf{P}^\top(\mathbf{r}^{[1]} + \Delta_{r^{[1]}}) + \delta_{r_{\mathrm{C}}^{[1]}}$, where, using (3.5), (3.19), and (3.26),

$$\begin{aligned}\|\delta_{r_{\mathrm{C}}^{[1]}}\| &\le (\dot{\varepsilon}(\underline{m}_{P,\dot{\varepsilon}} + 1) + \dot{\varepsilon}^2\underline{m}_{P,\dot{\varepsilon}})\|\,|\mathbf{P}|\,\|(\|\mathbf{r}^{[1]}\| + \|\Delta_{r^{[1]}}\|) \\ &\le \underbrace{(\dot{\varepsilon}(\underline{m}_{P,\dot{\varepsilon}} + 1) + \dot{\varepsilon}^2\underline{m}_{P,\dot{\varepsilon}})\|\,|\mathbf{P}|\,\|(\|\mathbf{A}\|^{\frac{1}{2}} + K_4)}_{K_5}\|\mathbf{y}\|_{\mathbf{A}}. \qquad (3.27)\end{aligned}$$

The computed term $\hat{\mathbf{r}}_{\mathrm{C}}^{[1]}$ can be written as $\hat{\mathbf{r}}_{\mathrm{C}}^{[1]} = \mathbf{r}_{\mathrm{C}}^{[1]} + \Delta_{r_{\mathrm{C}}^{[1]}}$, where $\Delta_{r_{\mathrm{C}}^{[1]}} = \mathbf{P}^\top\Delta_{r^{[1]}} + \delta_{r_{\mathrm{C}}^{[1]}}$ is the accumulated error and, using (3.15), (3.24), (3.22), (3.25), and (3.27),

$$\begin{aligned}\|\mathbf{A}_{\mathrm{C}}^{-1}\Delta_{r_{\mathrm{C}}^{[1]}}\|_{\mathbf{A}_{\mathrm{C}}} &= \|\mathbf{A}_{\mathrm{C}}^{-1}(\mathbf{P}^\top\Delta_{r^{[1]}} + \delta_{r_{\mathrm{C}}^{[1]}})\|_{\mathbf{A}_{\mathrm{C}}} \\ &= \|\mathbf{A}_{\mathrm{C}}^{-1}(\mathbf{P}^\top(\delta_f - \mathbf{A}\Delta_{v^{[1]}} + \delta_{r^{[1]}}) + \delta_{r_{\mathrm{C}}^{[1]}})\|_{\mathbf{A}_{\mathrm{C}}} \\ &\le \|\mathbf{A}_{\mathrm{C}}^{-1}\mathbf{P}^\top\mathbf{A}\Delta_{v^{[1]}}\|_{\mathbf{A}_{\mathrm{C}}} + \|\mathbf{A}_{\mathrm{C}}^{-1}(\mathbf{P}^\top(\delta_f + \delta_{r^{[1]}}) + \delta_{r_{\mathrm{C}}^{[1]}})\|_{\mathbf{A}_{\mathrm{C}}} \\ &\le \|\Delta_{v^{[1]}}\|_{\mathbf{A}} + \|\mathbf{A}_{\mathrm{C}}^{-1}\|^{\frac{1}{2}}\|\mathbf{P}^\top(\delta_f + \delta_{r^{[1]}}) + \delta_{r_{\mathrm{C}}^{[1]}}\| \\ &\le \|\Delta_{v^{[1]}}\|_{\mathbf{A}} + \|\mathbf{A}_{\mathrm{C}}^{-1}\|^{\frac{1}{2}}(\|\mathbf{P}\|(\|\delta_f\| + \|\delta_{r^{[1]}}\|) + \|\delta_{r_{\mathrm{C}}^{[1]}}\|) \\ &\le \underbrace{(\|\mathbf{A}\|^{\frac{1}{2}}K_2 + \|\mathbf{A}_{\mathrm{C}}^{-1}\|^{\frac{1}{2}}(\|\mathbf{P}\|(K_0 + K_3) + K_5))}_{K_6}\|\mathbf{y}\|_{\mathbf{A}}. \qquad (3.28)\end{aligned}$$

Line 5: Applying the coarse-level solver to $\mathbf{r}_{\mathrm{C}}^{[1]} + \Delta_{r_{\mathrm{C}}^{[1]}}$ in finite precision results in

$$\mathbf{M}_{\mathrm{C}}(\mathbf{r}_{\mathrm{C}}^{[1]} + \Delta_{r_{\mathrm{C}}^{[1]}}) + \delta_{v_{\mathrm{C}}^{[2]}},$$

where, using the assumption (3.11) and the estimates (3.20) and (3.28),

$$\begin{aligned}\|\delta_{v_{\mathrm{C}}^{[2]}}\|_{\mathbf{A}_{\mathrm{C}}} &\le \Lambda_{\mathrm{C}}\|\mathbf{A}_{\mathrm{C}}^{-1}(\mathbf{r}_{\mathrm{C}}^{[1]} + \Delta_{r_{\mathrm{C}}^{[1]}})\|_{\mathbf{A}_{\mathrm{C}}} \\ &\le \Lambda_{\mathrm{C}}(\|\mathbf{A}_{\mathrm{C}}^{-1}\mathbf{r}_{\mathrm{C}}^{[1]}\|_{\mathbf{A}_{\mathrm{C}}} + \|\mathbf{A}_{\mathrm{C}}^{-1}\Delta_{r_{\mathrm{C}}^{[1]}}\|_{\mathbf{A}_{\mathrm{C}}}) \\ &\le \underbrace{\Lambda_{\mathrm{C}}(1 + K_6)}_{K_7}\|\mathbf{y}\|_{\mathbf{A}}. \qquad (3.29)\end{aligned}$$

The computed term $\hat{\mathbf{v}}_{\mathrm{C}}^{[2]}$ can be written as $\hat{\mathbf{v}}_{\mathrm{C}}^{[2]} = \mathbf{v}_{\mathrm{C}}^{[2]} + \Delta_{v_{\mathrm{C}}^{[2]}}$, where $\Delta_{v_{\mathrm{C}}^{[2]}} = \mathbf{M}_{\mathrm{C}}\Delta_{r_{\mathrm{C}}^{[1]}} + \delta_{v_{\mathrm{C}}^{[2]}}$ is the accumulated error and, using (3.16), (3.28), and (3.29),

$$\begin{aligned}\|\Delta_{v_{\mathrm{C}}^{[2]}}\|_{\mathbf{A}_{\mathrm{C}}} &\le \|\mathbf{M}_{\mathrm{C}}\mathbf{A}_{\mathrm{C}}\mathbf{A}_{\mathrm{C}}^{-1}\Delta_{r_{\mathrm{C}}^{[1]}}\|_{\mathbf{A}_{\mathrm{C}}} + \|\delta_{v_{\mathrm{C}}^{[2]}}\|_{\mathbf{A}_{\mathrm{C}}} \\ &\le \|\mathbf{M}_{\mathrm{C}}\mathbf{A}_{\mathrm{C}}\|_{\mathbf{A}_{\mathrm{C}}} \cdot \|\mathbf{A}_{\mathrm{C}}^{-1}\Delta_{r_{\mathrm{C}}^{[1]}}\|_{\mathbf{A}_{\mathrm{C}}} + \|\delta_{v_{\mathrm{C}}^{[2]}}\|_{\mathbf{A}_{\mathrm{C}}} \\ &\le \underbrace{(2K_6 + K_7)}_{K_8}\|\mathbf{y}\|_{\mathbf{A}}. \qquad (3.30)\end{aligned}$$

Line 6: Computing $(\mathbf{P}+\dot{\Delta}\mathbf{P})(\mathbf{v}_{\mathrm{C}}^{[2]}+\Delta_{v_{\mathrm{C}}^{[2]}})$ in $\dot{\varepsilon}$-precision results in $\mathbf{P}(\mathbf{v}_{\mathrm{C}}^{[2]}+\Delta_{v_{\mathrm{C}}^{[2]}})+\delta_{v^{[2]}}$, where, using (3.5), (3.21), and (3.30),

$$\begin{aligned}
\|\delta_{v^{[2]}}\|_{\mathbf{A}} &\leq \|\mathbf{A}\|^{\frac{1}{2}}\|\delta_{v^{[2]}}\| \\
&\leq \|\mathbf{A}\|^{\frac{1}{2}}(\dot{\varepsilon}(\underline{m}_{P,\dot{\varepsilon}}+1)+\dot{\varepsilon}^2\underline{m}_{P,\dot{\varepsilon}})|||\mathbf{P}|||\cdot\|\mathbf{v}_{\mathrm{C}}^{[2]}+\Delta_{v_{\mathrm{C}}^{[2]}}\| \\
&\leq \|\mathbf{A}\|^{\frac{1}{2}}(\dot{\varepsilon}(\underline{m}_{P,\dot{\varepsilon}}+1)+\dot{\varepsilon}^2\underline{m}_{P,\dot{\varepsilon}})|||\mathbf{P}|||\cdot\|\mathbf{A}_{\mathrm{C}}^{-1}\|^{\frac{1}{2}}\|\mathbf{v}_{\mathrm{C}}^{[2]}+\Delta_{v_{\mathrm{C}}^{[2]}}\|_{\mathbf{A}_{\mathrm{C}}} \\
&\leq \|\mathbf{A}\|^{\frac{1}{2}}(\dot{\varepsilon}(\underline{m}_{P,\dot{\varepsilon}}+1)+\dot{\varepsilon}^2\underline{m}_{P,\dot{\varepsilon}})|||\mathbf{P}|||\cdot\|\mathbf{A}_{\mathrm{C}}^{-1}\|^{\frac{1}{2}}(\|\mathbf{v}_{\mathrm{C}}^{[2]}\|_{\mathbf{A}_{\mathrm{C}}}+\|\Delta_{v_{\mathrm{C}}^{[2]}}\|_{\mathbf{A}_{\mathrm{C}}}) \\
&\leq \|\mathbf{A}\|^{\frac{1}{2}}(\dot{\varepsilon}(\underline{m}_{P,\dot{\varepsilon}}+1)+\dot{\varepsilon}^2\underline{m}_{P,\dot{\varepsilon}})|||\mathbf{P}|||\cdot\|\mathbf{A}_{\mathrm{C}}^{-1}\|^{\frac{1}{2}}(2\|\mathbf{y}\|_{\mathbf{A}}+\|\Delta_{v_{\mathrm{C}}^{[2]}}\|_{\mathbf{A}_{\mathrm{C}}}) \\
&\leq \underbrace{\|\mathbf{A}\|^{\frac{1}{2}}(\dot{\varepsilon}(\underline{m}_{P,\dot{\varepsilon}}+1)+\dot{\varepsilon}^2\underline{m}_{P,\dot{\varepsilon}})|||\mathbf{P}|||\cdot\|\mathbf{A}_{\mathrm{C}}^{-1}\|^{\frac{1}{2}}(2+K_8)}_{K_9}\|\mathbf{y}\|_{\mathbf{A}}. \qquad (3.31)
\end{aligned}$$

The computed term $\hat{\mathbf{v}}^{[2]}$ can be written as $\hat{\mathbf{v}}^{[2]}=\mathbf{v}^{[2]}+\Delta_{v^{[2]}}$, where

$$\Delta_{v^{[2]}}=\mathbf{P}\Delta_{v_{\mathrm{C}}^{[2]}}+\delta_{v^{[2]}}$$

is the accumulated error, and using (3.9), (3.30), and (3.31),

$$\begin{aligned}
\|\Delta_{v^{[2]}}\|_{\mathbf{A}} &\leq \|\mathbf{P}\Delta_{v_{\mathrm{C}}^{[2]}}+\delta_{v^{[2]}}\|_{\mathbf{A}} \leq \|\mathbf{P}\Delta_{v_{\mathrm{C}}^{[2]}}\|_{\mathbf{A}}+\|\delta_{v^{[2]}}\|_{\mathbf{A}}=\|\Delta_{v_{\mathrm{C}}^{[2]}}\|_{\mathbf{A}_{\mathrm{C}}}+\|\delta_{v^{[2]}}\|_{\mathbf{A}} \\
&\leq \underbrace{(K_8+K_9)}_{K_{10}}\|\mathbf{y}\|_{\mathbf{A}}. \qquad (3.32)
\end{aligned}$$

Line 7: Computing $\mathbf{v}^{[1]}+\Delta_{v^{[1]}}+\mathbf{v}^{[2]}+\Delta_{v^{[2]}}$ in $\dot{\varepsilon}$-precision result in

$$\mathbf{v}^{[1]}+\Delta_{v^{[1]}}+\mathbf{v}^{[2]}+\Delta_{v^{[2]}}+\delta_{v^{[3]}},$$

where, using (3.3), $\mathbf{v}^{[3]}=\mathbf{v}^{[1]}+\mathbf{v}^{[2]}$, (3.17), (3.24), and (3.32),

$$\begin{aligned}
\|\delta_{v^{[3]}}\| &\leq \dot{\varepsilon}\|\mathbf{v}^{[1]}+\Delta_{v^{[1]}}+\mathbf{v}^{[2]}+\Delta_{v^{[2]}}\| \\
&\leq \dot{\varepsilon}(\|\mathbf{v}^{[3]}\|+\|\Delta_{v^{[1]}}\|+\|\Delta_{v^{[2]}}\|) \\
&\leq \dot{\varepsilon}(\|\mathbf{A}^{-1}\|^{\frac{1}{2}}\|\mathbf{v}^{[3]}\|_{\mathbf{A}}+\|\Delta_{v^{[1]}}\|+\|\mathbf{A}^{-1}\|^{\frac{1}{2}}\|\Delta_{v^{[2]}}\|_{\mathbf{A}}) \\
&\leq \underbrace{\dot{\varepsilon}(2\|\mathbf{A}^{-1}\|^{\frac{1}{2}}+K_2+\|\mathbf{A}^{-1}\|^{\frac{1}{2}}K_{10})}_{K_{11}}\|\mathbf{y}\|_{\mathbf{A}}. \qquad (3.33)
\end{aligned}$$

Finally the computed approximation $\hat{\mathbf{v}}^{[3]}$ can be written as $\hat{\mathbf{v}}^{[3]}=\mathbf{v}^{[3]}+\Delta_{v^{[3]}}$, where

$$\Delta_{v^{[3]}}=\Delta_{v^{[1]}}+\Delta_{v^{[2]}}+\delta_{v^{[3]}}$$

is the accumulated error and using (3.24), (3.32), and (3.33),

$$\begin{aligned}
\|\Delta_{v^{[3]}}\|_{\mathbf{A}} &\leq \|\Delta_{v^{[1]}}\|_{\mathbf{A}}+\|\Delta_{v^{[2]}}\|_{\mathbf{A}}+\|\delta_{v^{[3]}}\|_{\mathbf{A}} \\
&\leq \underbrace{(\|\mathbf{A}\|^{\frac{1}{2}}K_2+K_{10}+\|\mathbf{A}\|^{\frac{1}{2}}K_{11})}_{\delta_{\mathrm{TG}}}\|\mathbf{y}\|_{\mathbf{A}}.
\end{aligned}$$

Since $\hat{\mathbf{v}}^{[3]}=\hat{\mathbf{y}}_{\mathrm{TG}}$ and $\mathbf{v}^{[3]}=\mathbf{y}_{\mathrm{TG}}$, we have $\Delta_{v^{[3]}}=\hat{\mathbf{y}}_{\mathrm{TG}}-\mathbf{y}_{\mathrm{TG}}$, and

$$\|\mathbf{y}_{\mathrm{TG}}-\hat{\mathbf{y}}_{\mathrm{TG}}\|_{\mathbf{A}}\leq\delta_{\mathrm{TG}}\|\mathbf{y}\|_{\mathbf{A}}.$$

Consequently,

$$\|\mathbf{y} - \hat{\mathbf{y}}_{\mathrm{TG}}\|_{\mathbf{A}} \leq \|\mathbf{y} - \mathbf{y}_{\mathrm{TG}}\|_{\mathbf{A}} + \|\mathbf{y}_{\mathrm{TG}} - \hat{\mathbf{y}}_{\mathrm{TG}}\|_{\mathbf{A}} \leq (\rho_{\mathrm{TG}} + \delta_{\mathrm{TG}})\|\mathbf{y}\|_{\mathbf{A}}.$$

We simplify the expression for δ_{TG} by grouping higher order terms in a remainder R. We say that a term is of higher order when it involves at least second powers of at least one of $\dot{\varepsilon}$, Λ_{C}, Λ_M, or a product of at least two of them. All remainder terms R_k, $k = 1, \ldots, 6$ defined below contain only high order terms.

Listing and rewriting the constants K_0, K_1, K_2, K_α, and K_3 leads to

$$\begin{aligned}
K_0 &= \dot{\varepsilon}\|\mathbf{A}\|^{\frac{1}{2}}, \\
K_1 &= (\Lambda_M\|\mathbf{A}\|^{\frac{1}{2}} + \Lambda_M K_0) = (\Lambda_M + \Lambda_M\dot{\varepsilon})\|\mathbf{A}\|^{\frac{1}{2}}, \\
K_2 &= \|\mathbf{M}\|K_0 + K_1 = (\|\mathbf{M}\|\dot{\varepsilon} + \Lambda_M + \Lambda_M\dot{\varepsilon})\|\mathbf{A}\|^{\frac{1}{2}}, \\
K_\alpha &= (2m_{A,\dot{\varepsilon}} + 1 + \dot{\varepsilon}m_{A,\dot{\varepsilon}}), \\
K_3 &= (\dot{\varepsilon}(m_{A,\dot{\varepsilon}} + 2) + \dot{\varepsilon}^2 K_\alpha)(\|\mathbf{A}\|^{\frac{1}{2}} + K_0 + |||\mathbf{A}|||(\|\mathbf{M}\|\|\mathbf{A}\|^{\frac{1}{2}} + K_2)) \\
&= \dot{\varepsilon}(m_{A,\dot{\varepsilon}} + 2)(\|\mathbf{A}\|^{\frac{1}{2}} + K_0 + |||\mathbf{A}|||(\|\mathbf{M}\|\|\mathbf{A}\|^{\frac{1}{2}} + K_2)) \\
&\quad + \dot{\varepsilon}^2 K_\alpha(\|\mathbf{A}\|^{\frac{1}{2}} + K_0 + |||\mathbf{A}|||(\|\mathbf{M}\|\|\mathbf{A}\|^{\frac{1}{2}} + K_2)) \\
&= \dot{\varepsilon}(m_{A,\dot{\varepsilon}} + 2)\|\mathbf{A}\|^{\frac{1}{2}}(1 + |||\mathbf{A}|||\|\mathbf{M}\|) \\
&\quad + \underbrace{\dot{\varepsilon}(m_{A,\dot{\varepsilon}} + 2)(K_0 + |||\mathbf{A}|||K_2) + \dot{\varepsilon}^2 K_\alpha(\|\mathbf{A}\|^{\frac{1}{2}} + K_0 + |||\mathbf{A}|||(\|\mathbf{M}\|\|\mathbf{A}\|^{\frac{1}{2}} + K_2))}_{R_1}.
\end{aligned}$$

The constants K_4 and K_6 can be rewritten as

$$\begin{aligned}
K_4 &= K_0 + \|\mathbf{A}\|K_2 + K_3 \\
&= (\dot{\varepsilon} + \|\mathbf{A}\|(\|\mathbf{M}\|\dot{\varepsilon} + \Lambda_M + \Lambda_M\dot{\varepsilon}))\|\mathbf{A}\|^{\frac{1}{2}} \\
&\quad + \dot{\varepsilon}(m_{A,\dot{\varepsilon}} + 2)\|\mathbf{A}\|^{\frac{1}{2}}(1 + |||\mathbf{A}|||\|\mathbf{M}\|) + R_1, \\
K_5 &= (\dot{\varepsilon}(\underline{m}_{P,\dot{\varepsilon}} + 1) + \dot{\varepsilon}^2\underline{m}_{P,\dot{\varepsilon}})|||\mathbf{P}|||(\|\mathbf{A}\|^{\frac{1}{2}} + K_4) \\
&= \dot{\varepsilon}(\underline{m}_{P,\dot{\varepsilon}} + 1)|||\mathbf{P}|||(\|\mathbf{A}\|^{\frac{1}{2}} + K_4) + \dot{\varepsilon}^2\underline{m}_{P,\dot{\varepsilon}}|||\mathbf{P}|||(\|\mathbf{A}\|^{\frac{1}{2}} + K_4) \\
&= \dot{\varepsilon}(\underline{m}_{P,\dot{\varepsilon}} + 1)|||\mathbf{P}|||\|\mathbf{A}\|^{\frac{1}{2}} + \underbrace{\dot{\varepsilon}(\underline{m}_{P,\dot{\varepsilon}} + 1)|||\mathbf{P}|||K_4 + \dot{\varepsilon}^2\underline{m}_{P,\dot{\varepsilon}}|||\mathbf{P}|||(\|\mathbf{A}\|^{\frac{1}{2}} + K_4)}_{R_2}.
\end{aligned}$$

Let ξ denote the ratio $\|\mathbf{A}_{\mathrm{C}}^{-1}\|^{\frac{1}{2}}/\|\mathbf{A}^{-1}\|^{\frac{1}{2}}$. The constant K_6 can be rewritten as

$$\begin{aligned}
K_6 &= \|\mathbf{A}\|^{\frac{1}{2}}K_2 + \|\mathbf{A}_{\mathrm{C}}^{-1}\|^{\frac{1}{2}}(\|\mathbf{P}\|(K_0 + K_3) + K_5) \\
&= \|\mathbf{A}\|(\|\mathbf{M}\|\dot{\varepsilon} + \Lambda_M) \\
&\qquad + \dot{\varepsilon}\kappa_A^{\frac{1}{2}}\xi(\|\mathbf{P}\|(1 + (m_{A,\dot{\varepsilon}} + 2)(1 + |||\mathbf{A}|||\|\mathbf{M}\|)) + (\underline{m}_{P,\dot{\varepsilon}} + 1)|||\mathbf{P}|||) \\
&\qquad + \underbrace{\|\mathbf{A}\|\Lambda_M\dot{\varepsilon} + \|\mathbf{A}_{\mathrm{C}}^{-1}\|^{\frac{1}{2}}\|\mathbf{P}\|R_1 + \|\mathbf{A}_{\mathrm{C}}^{-1}\|^{\frac{1}{2}}R_2}_{R_3}.
\end{aligned}$$

The constant K_7 can be expressed as

$$K_7 = \Lambda_{\mathrm{C}}(1 + K_6) = \Lambda_{\mathrm{C}} + \underbrace{\Lambda_{\mathrm{C}}K_6}_{R_4}.$$

Rewriting the constants K_8 and K_9 yields

$$\begin{aligned} K_8 &= 2K_6 + K_7 = 2K_6 + \Lambda_{\mathrm{C}} + R_4, \\ K_9 &= \|\mathbf{A}\|^{\frac{1}{2}}(\dot{\varepsilon}(\underline{m}_{P,\dot{\varepsilon}} + 1) + \dot{\varepsilon}^2\underline{m}_{P,\dot{\varepsilon}})\||\mathbf{P}|\| \cdot \|\mathbf{A}_{\mathrm{C}}^{-1}\|^{\frac{1}{2}}(2 + K_8) \\ &= 2\dot{\varepsilon}(\underline{m}_{P,\dot{\varepsilon}} + 1)\kappa_A^{\frac{1}{2}}\xi\||\mathbf{P}|\| \\ &\quad + \underbrace{\dot{\varepsilon}(\underline{m}_{P,\dot{\varepsilon}} + 1)\kappa_A^{\frac{1}{2}}\xi\||\mathbf{P}|\|K_8 + \dot{\varepsilon}^2\underline{m}_{P,\dot{\varepsilon}}\kappa_A^{\frac{1}{2}}\xi\||\mathbf{P}|\|(2 + K_8)}_{R_5}. \end{aligned}$$

The constants K_{10} and K_{11} can be expressed as

$$\begin{aligned} K_{10} &= K_8 + K_9 \\ K_{11} &= \dot{\varepsilon}(2\|\mathbf{A}^{-1}\|^{\frac{1}{2}} + K_2 + \|\mathbf{A}^{-1}\|^{\frac{1}{2}}K_{10}) \\ &= 2\dot{\varepsilon}\|\mathbf{A}^{-1}\|^{\frac{1}{2}} + \underbrace{\dot{\varepsilon}K_2 + \dot{\varepsilon}\|\mathbf{A}^{-1}\|^{\frac{1}{2}}K_{10}}_{R_6}. \end{aligned}$$

Finally, δ_{TG} can be simplified as

$$\begin{aligned} \delta_{\mathrm{TG}} &= \|\mathbf{A}\|^{\frac{1}{2}}K_2 + K_{10} + \|\mathbf{A}\|^{\frac{1}{2}}K_{11} \\ &= \|\mathbf{A}\|(\|\mathbf{M}\|\dot{\varepsilon} + \Lambda_M) + 2K_6 + \Lambda_{\mathrm{C}} + R_4 + 2(\underline{m}_{P,\dot{\varepsilon}} + 1)\dot{\varepsilon}\kappa_A^{\frac{1}{2}}\xi\||\mathbf{P}|\| \\ &\qquad + R_5 + 2\dot{\varepsilon}\kappa_A^{\frac{1}{2}} + \|\mathbf{A}\|^{\frac{1}{2}}R_6 \\ &= 3\|\mathbf{A}\|(\|\mathbf{M}\|\dot{\varepsilon} + \Lambda_M) + \dot{\varepsilon}\kappa_A^{\frac{1}{2}}\xi(2\|\mathbf{P}\|(1 + (m_{A,\dot{\varepsilon}} + 2)(1 + \||\mathbf{A}|\|\|\mathbf{M}\|)) \\ &\qquad + 4(\underline{m}_{P,\dot{\varepsilon}} + 1)\||\mathbf{P}|\|) + \Lambda_{\mathrm{C}} + 2\dot{\varepsilon}\kappa_A^{\frac{1}{2}} + \underbrace{2R_3 + R_4 + R_5 + \|\mathbf{A}\|^{\frac{1}{2}}R_6}_{R}, \end{aligned}$$

where R is a remainder containing higher order terms. □

3.5 V-cycle correction scheme

In this section, we present a mixed precision formulation of the V-cycle correction scheme and its finite precision error analysis. The V-cycle correction scheme can be seen as a generalization of the two-grid correction scheme to multiple levels.

We consider using the V-cycle correction scheme for computing an approximate solution of $\mathbf{Ay} = \mathbf{f}$, where $\mathbf{f} \in \mathbb{R}^n$. The approximate solution is computed using a hierarchy of $J + 1$ levels numbered from 0 to J. Each level contains a stiffness matrix $\mathbf{A}_j \in \mathbb{R}^{n_j \times n_j}$, $j = 0, \ldots, J$, with $\mathbf{A}_J = \mathbf{A}$. The information is transferred between the $(j-1)$th and jth levels using the full-rank prolongation matrix $\mathbf{P}_j \in \mathbb{R}^{n_j \times n_{j-1}}$, $j = 1, \ldots, J$, and its transpose. We assume that the stiffness matrices satisfy the Galerkin condition i.e., $\mathbf{A}_{j-1} = \mathbf{P}_j^\top \mathbf{A}_j \mathbf{P}_j$, $j = 1, \ldots, J$. We denote by $\mathbf{I}_j$ the identity matrix on level j.

The computation consists of smoothing on fine levels and solving a system of linear equations on the coarsest-level. We assume that all operations on a fine level j, $j = 1, \ldots, J$, besides the smoothing, are done in finite precision arithmetic with unit roundoff $\dot{\varepsilon}_j$. We consider that the precision used on level j, $j = 2, \ldots, J$, is higher or equal to the precision used on the coarser level $j - 1$, i.e., $\dot{\varepsilon}_j \leq \dot{\varepsilon}_{j-1}$.

We assume that the matrices $\mathbf{A}_j$ and $\mathbf{P}_j$ on level j are rounded to the $\dot{\varepsilon}$-precision for the residual computation and for computing the restriction and prolongation, resulting in $\dot{\mathbf{A}}_j = \mathbf{A}_j + \dot{\Delta}\mathbf{A}_j$, where $|\dot{\Delta}\mathbf{A}_j| \leq \dot{\varepsilon}_j|\mathbf{A}_j|$, and $\dot{\mathbf{P}}_j = \mathbf{P}_j + \dot{\Delta}\mathbf{P}_j$, where $|\dot{\Delta}\mathbf{P}_j| \leq \dot{\varepsilon}_j|\mathbf{P}_j|$.

Analogously, as in the TG correction scheme, we assume that the application of smoothing in infinite precision on level j, $j = 1, \ldots, J$, can, for any vector $\mathbf{f}_j$, be expressed as $\mathbf{M}_j\mathbf{f}_j$, where $\mathbf{M}_j \in \mathbb{R}^{n_j \times n_j}$ is a non-singular matrix approximating the inverse of $\mathbf{A}_j$ in the sense

$$\|\mathbf{I}_j - \mathbf{M}_j\mathbf{A}_j\|_{\mathbf{A}_j} < 1. \tag{3.34}$$

We assume that there exists a positive constant Λ_{M_j} such that for any vector $\mathbf{f}_j$, the application of smoothing in finite precision on the jth level results in

$$\mathbf{M}_j\mathbf{f}_j + \delta_{M_j}, \quad \|\delta_{M_j}\| \leq \Lambda_{M_j}\|\mathbf{f}_j\|, \tag{3.35}$$

and the result belongs to the $\dot{\varepsilon}_j$-precision arithmetic.

For any vector $\mathbf{f}_0$, we assume that we can write the application of the coarsest-level solver in infinite precision as $\mathbf{M}_0\mathbf{f}_0$, where $\mathbf{M}_0$ is a non-singular matrix satisfying

$$\|\mathbf{I}_0 - \mathbf{M}_0\mathbf{A}_0\|_{\mathbf{A}_0} < 1. \tag{3.36}$$

We consider that there exists a positive constant Λ_0 such that the application of the coarsest-level solver in finite precision for any vector $\mathbf{f}_0$ results in

$$\mathbf{M}_0\mathbf{f}_0 + \delta_0, \quad \|\delta_0\|_{\mathbf{A}_0} \leq \Lambda_0\|\mathbf{A}_0^{-1}\mathbf{f}_0\|_{\mathbf{A}_0}, \tag{3.37}$$

and the result belongs to the $\dot{\varepsilon}_1$-precision arithmetic.

The V-cycle correction scheme is described in Algorithm 3.3; variants in which the smoothing is applied after the recursive call or both before and after the recursive call can be found in the literature; see, e.g., [6, 27].

Algorithm 3.3 V-cycle correction scheme, $\mathbf{V}(\mathbf{f}_j, j)$.

```
if $j \neq 0$ then
    $\hat{\mathbf{f}}_j = \mathrm{Round}(\mathbf{f}_j, \dot{\varepsilon}_j)$ {Round the right-hand side $\mathbf{f}_j$ to $\dot{\varepsilon}_j$-precision.}
    $\hat{\mathbf{v}}_j^{[1]} = \mathbf{M}_j\hat{\mathbf{f}}_j$ {Apply smoothing.}
    $\hat{\mathbf{r}}_j^{[1]} = \hat{\mathbf{f}}_j - \dot{\mathbf{A}}_j\hat{\mathbf{v}}_j^{[1]}$ {Compute residual in $\dot{\varepsilon}_j$-precision.}
    $\hat{\mathbf{r}}_{j-1}^{[1]} = \dot{\mathbf{P}}_j^\top\hat{\mathbf{r}}_j^{[1]}$ {Restrict the residual to level $j$ in $\dot{\varepsilon}_j$-precision.}
    $\hat{\mathbf{v}}_{j-1}^{[2]} = \mathbf{V}(\hat{\mathbf{r}}_{j-1}^{[1]}, j-1)$ {Recursive call.}
    $\hat{\mathbf{v}}_j^{[2]} = \dot{\mathbf{P}}_j\hat{\mathbf{v}}_{j-1}^{[2]}$ {Prolongate the correction to level $j$ in $\dot{\varepsilon}_j$-precision.}
    $\hat{\mathbf{v}}_j^{[3]} = \hat{\mathbf{v}}_j^{[1]} + \hat{\mathbf{v}}_j^{[2]}$ {Correct the previous approximation in $\dot{\varepsilon}_j$-precision.}
    return $\hat{\mathbf{y}}_{\mathrm{V},j} = \hat{\mathbf{v}}_j^{[3]}$.
else
    return $\mathbf{M}_0\mathbf{f}_0$ {Approximately solve $\mathbf{A}_0\mathbf{v}_0 = \mathbf{f}_0$.}
end if
```

We assume that the V-cycle correction scheme converges uniformly in the following sense. Let $\mathbf{f}_j$, $j = 1, \ldots, J$, be the right-hand side vector on the jth level and let $\mathbf{y}_j$, $j = 1, \ldots, J$, be the solution of $\mathbf{A}_j \mathbf{y}_j = \mathbf{f}_j$. We assume that the V-cycle correction scheme with $j+1$ levels, $0, \ldots, j$, in infinite precision reduces the $\mathbf{A}_j$-norm of the error by at least a factor $\rho_{\mathrm{V}} < 1$, which is independent of j, i.e.,

$$\|\mathbf{y}_j - \mathbf{y}_{\mathrm{V},j}\|_{\mathbf{A}_j} \leq \rho_{\mathrm{V}} \|\mathbf{y}_j\|_{\mathbf{A}_j}, \tag{3.38}$$

where $\mathbf{y}_{\mathrm{V},j}$ is the approximation computed using the V-cycle correction scheme with $j+1$ levels, $0, \ldots, j$.

We present the following result on the effects of finite precision errors on the convergence of the V-cycle correction scheme. Its proof, based on consecutive usage of Theorem 3.1, is presented below.

Theorem 3.2. *Let $\mathbf{y}_{\mathrm{V}}$ and $\hat{\mathbf{y}}_{\mathrm{V}}$ be the approximate solution of $\mathbf{A}\mathbf{y} = \mathbf{f}$ computed using the V-cycle correction scheme, Algorithm 3.3, with $J+1$ levels, applied in infinite and in finite precision, respectively. Then*

$$\|\mathbf{y}_{\mathrm{V}} - \hat{\mathbf{y}}_{\mathrm{V}}\|_{\mathbf{A}} \leq \sum_{j=0}^{J} \delta_{\mathrm{V},j} \|\mathbf{y}\|_{\mathbf{A}},$$

$$\|\mathbf{y} - \hat{\mathbf{y}}_{\mathrm{V}}\|_{\mathbf{A}} \leq \left(\rho_{\mathrm{V}} + \sum_{j=0}^{J} \delta_{\mathrm{V},j} \right) \|\mathbf{y}\|_{\mathbf{A}},$$

where $\delta_{\mathrm{V},0} = \Lambda_0$ and for $j = 1, \ldots, J$, $\delta_{\mathrm{V},j}$ is expressed as

$$\delta_{\mathrm{V},j} = 3\|\mathbf{A}_j\|\Lambda_{M_j} + \dot{\varepsilon}_j (C_{1,j} \kappa_{A_j}^{\frac{1}{2}} |||\mathbf{A}_j||| \|\mathbf{M}_j\| + C_{2,j}) + 3\dot{\varepsilon}_j \|\mathbf{A}_j\| \|\mathbf{M}_j\| + R_j,$$

where $C_{1,j}$ and $C_{2,j}$ are constants depending only on $\|\mathbf{P}_j\|$, $|||\mathbf{P}_j|||$, $m_{A_j,\dot{\varepsilon}_j}$, $\underline{m}_{P_j,\dot{\varepsilon}_j}$ and the ratio $\|\mathbf{A}_{j-1}^{-1}\|^{\frac{1}{2}} / \|\mathbf{A}_j^{-1}\|^{\frac{1}{2}}$. The remainder R_j contains higher order terms, i.e., terms which involve at least second powers of at least one of $\dot{\varepsilon}_j$, $\Lambda_j = \sum_{i=0}^{j-1} \delta_{\mathrm{V},i}$, Λ_{M_j}, or a product of at least two of them. The V-cycle correction scheme reduces the $\mathbf{A}$-norm of the error if $\rho_{\mathrm{V}} + \sum_{j=0}^{J} \delta_{\mathrm{V},j} < 1$.

This theorem provides insight into how the finite precision errors coming from the coarsest-level solver, the smoothers and the error term resulting from computing the residual, restriction, prolongation and correction in $\dot{\varepsilon}_j$-precision on the individual levels may affect the overall finite precision error. We see that the requirement on the $\dot{\varepsilon}_j$-precision as well as the finite precision error of the smoother may differ on each fine level based on the properties of the corresponding system and prolongation matrices and the chosen smoother.

In Section 3.6 we present a mixed precision IC smoothing routine and its finite precision error analysis with bounds on the corresponding Λ_M and $\|\mathbf{M}\|$. We utilize the result of Theorem 3.2 in Section 3.8, when discussing requirements on the finite precisions used in the V-cycle scheme with IC smoothers for solving systems obtained by FE discretization of elliptic PDEs.

Proof of Theorem 3.2. We prove the theorem using induction on the number of levels. The V-cycle correction scheme with two levels can be seen as the TG correction scheme with $\mathbf{M} = \mathbf{M}_1$ and $\mathbf{M}_{\mathrm{C}} = \mathbf{M}_0$. Since the assumptions of Theorem 3.1 are satisfied, the statement holds for $j = 1$.

Let $\mathbf{V}_j$, $j = 1, \ldots, J$, be the matrix corresponding to applying the V-cycle correction scheme in infinite precision with $j + 1$ levels $0, \ldots, j$. Such a matrix exists; see e.g., [27, Theorem 2.4.1]. The assumption (3.38) yields

$$\|\mathbf{I}_j - \mathbf{V}_j \mathbf{A}_j\|_{\mathbf{A}_j} = \max_{\mathbf{y}_j} \frac{\|(\mathbf{I}_j - \mathbf{V}_j \mathbf{A}_j)\mathbf{y}_j\|_{\mathbf{A}_j}}{\|\mathbf{y}_j\|_{\mathbf{A}_j}} \leq \rho_V < 1. \tag{3.39}$$

We assume that the statement of the theorem holds for the V-cycle correction scheme with j levels. We can view the V-cycle correction scheme with $j+1$ levels as a two-grid correction scheme where the coarse-grid solver is the V-cycle correction scheme with j levels, i.e., $\mathbf{M} = \mathbf{M}_j$ and $\mathbf{M}_\mathrm{C} = \mathbf{V}_{j-1}$. Since the smoothing routine on level j and the coarse-grid solver satisfy the assumptions of Theorem 3.1, in particular,

$$\|\mathbf{I}_\mathrm{C} - \mathbf{M}_\mathrm{C}\mathbf{A}_\mathrm{C}\|_{\mathbf{A}_\mathrm{C}} = \|\mathbf{I}_{j-1} - \mathbf{V}_{j-1}\mathbf{A}_{j-1}\|_{\mathbf{A}_{j-1}} < 1,$$
$$\Lambda_\mathrm{C} = \Lambda_{j-1} = \sum_{i=0}^{j-1} \delta_{\mathrm{V},i},$$

the result also holds for the V-cycle correction scheme with $j + 1$ levels. □

3.6 Smoothing based on incomplete Cholesky factorization

In this section, we formulate a mixed precision smoothing routine based on incomplete Cholesky factorization (IC) and present its finite precision error analysis.

A smoothing routine computes an approximate solution of $\mathbf{Ay} = \mathbf{f}$, where $\mathbf{f} \in \mathbb{R}^n$. In this text, we for simplicity, consider only one smoothing iteration starting with a zero initial approximation. The smoothing thus reduces to solving an error equation where the matrix $\mathbf{A}$ is replaced by its approximation. We consider that the matrix $\mathbf{A}$ is approximated by an incomplete Cholesky factorization $\mathbf{L}\mathbf{L}^\top$, where $\mathbf{L}$ is a lower triangular matrix; see, e.g., [22, Chapter 10], [23, Chapter 10]. There are many variants of incomplete Cholesky factorization, e.g., variants with dropping based on a given tolerance, dropping based on the degree of fill-in and others. In this paper, we do not study which variant is the most effective. We rather assume that the selected variant works well and we focus on the finite precision error analysis of the corresponding smoothing routine.

The application of the smoother consists of solving two triangular systems with the matrix $\mathbf{L}$ and its transpose. We consider that the right-hand side $\mathbf{f}$ is first rounded to a precision with unit roundoff ε^S. We assume that the factor $\mathbf{L}$ is rounded to a precision[1] with unit roundoff ε^R, $\varepsilon^\mathrm{R} \geq \varepsilon^\mathrm{S}$, resulting in $\mathbf{L}^\mathrm{R} = \mathbf{L} + \Delta^\mathrm{R}\mathbf{L}$, where $|\Delta^\mathrm{R}\mathbf{L}| \leq \varepsilon^\mathrm{R}|\mathbf{L}|$. We consider that the triangular problems are solved using substitution (see, e.g., [11, Chapter 8]) performed in ε^S-precision. We include here the rounding to a lower ε^R-precision, since the factor $\mathbf{L}$ may be in practice stored

[1] The subscripts S and R here stand for Solve and stoRe, respectively. They indicate that the corresponding ε^S- and ε^R-precision are used for solving the triangular systems and for storing the matrix, respectively.

in a lower precision than which is used for solving the triangular systems. This may yield to faster memory movements and thus to a faster runtime.

The smoothing routine based on incomplete Cholesky factorization is described in Algorithm 3.4.

Algorithm 3.4 IC smoother, $\mathbf{ICS(f)}$.

```
$\hat{\mathbf{f}} = \mathrm{Round}(\mathbf{f}, \varepsilon^{\mathrm{S}})$ {Round the right-hand side $\mathbf{f}$ to $\varepsilon^{\mathrm{S}}$-precision.}
$\hat{\mathbf{v}} = \mathrm{Substitution}(\mathbf{L}^{\mathrm{R}}, \hat{\mathbf{f}}, \varepsilon^{\mathrm{S}})$ {Apply substitution in $\varepsilon^{\mathrm{S}}$-precision.}
$\hat{\mathbf{w}} = \mathrm{Substitution}((\mathbf{L}^{\mathrm{R}})^{\top}, \hat{\mathbf{v}}, \varepsilon^{\mathrm{S}})$ {Apply substitution in $\varepsilon^{\mathrm{S}}$-precision.}
return $\hat{\mathbf{w}}_{\mathrm{IC}} = \hat{\mathbf{w}}$.
```

Further we present finite precision analysis of the smoothing routine. As previously mentioned, we do not take into account the finite precision errors occurring when computing the factor $\mathbf{L}$. To our knowledge there is no finite precision error analysis of incomplete Cholesky factorization for general SPD matrices in the literature.

We first present a bound on the finite precision error of a general perturbed triangular solve via substitution.

3.6.1 Finite precision error analysis of solving sparse perturbed triangular system via substitution

Let $\mathbf{T} \in \mathbb{R}^{n \times n}$ be a sparse invertible triangular matrix with maximum m_T nonzero entries in any of its rows and $\mathbf{b} \in \mathbb{R}^n$ a right-hand side vector. We consider computing an approximate solution of the problem

$$\mathbf{Tx} = \mathbf{b} \tag{3.40}$$

using substitution in finite precision. By modifying the proof of [11, Theorem 8.5], using that there are maximum m_T nonzero entries in a row of $\mathbf{T}$, we can get the following result; we present its proof in Appendix 3.10.4.

Lemma 3.1. *Assume that the entries of matrix $\mathbf{T}$ belong to the ε^{S}-precision arithmetic. Let $\hat{\mathbf{x}}$ be the approximate solution of $\mathbf{Tx} = \mathbf{b}$ computed using substitution in finite precision with unit roundoff ε^{S}. There exists a matrix $\mathbf{E}$ such that the computed solution $\hat{\mathbf{x}}$ satisfies*

$$(\mathbf{T} + \mathbf{E})\hat{\mathbf{x}} = \mathbf{b}, \quad |\mathbf{E}| \leq \varepsilon^{\mathrm{S}} m_{T,\varepsilon^{\mathrm{S}}} |\mathbf{T}|. \tag{3.41}$$

We use this result to prove the following lemma containing a bound on the finite precision error of an approximate solution of $\mathbf{Tx} = \mathbf{b}$ computed via substitution applied to a problem with matrix $\mathbf{T}$ rounded to a lower precision and perturbed right-hand side.

Lemma 3.2. *Let $\mathbf{T}^{\mathrm{R}}$ be the matrix obtained from rounding matrix $\mathbf{T}$ to precision with unit roundoff ε^{R}. Let $\Delta\mathbf{b}$ be a perturbation of the right-hand side $\mathbf{b}$ satisfying $\|\Delta\mathbf{b}\| \leq \delta_b \|\mathbf{b}\|$ for a positive constant δ_b. Let $\hat{\mathbf{x}}$ be the approximate solution of $\mathbf{Tx} = \mathbf{b}$ computed using substitution in finite precision with an unit roundoff ε^{S}, $\varepsilon^{\mathrm{S}} \leq \varepsilon^{\mathrm{R}}$, applied to*

$$\mathbf{T}^{\mathrm{R}}\tilde{\mathbf{x}} = \mathbf{b} + \Delta\mathbf{b}. \tag{3.42}$$

Let $\delta_{T,\mathrm{R},\mathrm{S}} = \varepsilon^{\mathrm{R}} + \varepsilon^{\mathrm{S}} m_{T,\varepsilon^{\mathrm{S}}} + \varepsilon^{\mathrm{R}}\varepsilon^{\mathrm{S}} m_{T,\varepsilon^{\mathrm{S}}}$*. Assuming that* $\delta_{T,\mathrm{R},\mathrm{S}} \cdot \underline{\kappa}_T < 1$*, the following bounds hold:*

$$\frac{\|\mathbf{x} - \hat{\mathbf{x}}\|}{\|\mathbf{x}\|} \leq \frac{\delta_b \kappa_T + \delta_{T,\mathrm{R},\mathrm{S}} \cdot \underline{\kappa}_T}{1 - \delta_{T,\mathrm{R},\mathrm{S}} \cdot \underline{\kappa}_T}, \tag{3.43}$$

$$\|\mathbf{x} - \hat{\mathbf{x}}\| \leq \frac{\delta_b \|\mathbf{T}^{-1}\| \|\mathbf{b}\| + \delta_{T,\mathrm{R},\mathrm{S}} \cdot \underline{\kappa}_T \|\mathbf{x}\|}{1 - \delta_{T,\mathrm{R},\mathrm{S}} \cdot \underline{\kappa}_T}. \tag{3.44}$$

Proof. The proof goes as follows. We first use the bound on the error when rounding a matrix to a lower precision and Lemma 3.1 to write down a perturbed equation which the computed approximation $\hat{\mathbf{x}}$ satisfies. Further we use this equation to derive the bounds (3.43) and (3.44).

Using (3.2) we get that $\mathbf{T}^{\mathrm{R}} = \mathbf{T} + \Delta\mathbf{T}$, $|\Delta\mathbf{T}| \leq \varepsilon^{\mathrm{R}}|\mathbf{T}|$. Using Lemma 3.1 for the perturbed problem (3.42), we get that there exist a matrix $\mathbf{F}$,

$$|\mathbf{F}| \leq \varepsilon^{\mathrm{S}} m_{T+\Delta T,\varepsilon^{\mathrm{S}}} |\mathbf{T} + \Delta\mathbf{T}|, \tag{3.45}$$

such that the approximate solution $\hat{\mathbf{x}}$ computed using substitution satisfies

$$(\mathbf{T} + \Delta\mathbf{T} + \mathbf{F})\,\hat{\mathbf{x}} = \mathbf{b} + \Delta\mathbf{b}. \tag{3.46}$$

Note that rounding a matrix can only result in it having fewer non-zero elements, thus $m_{T+\Delta T} \leq m_T$ and consequently $m_{T+\Delta T,\varepsilon^{\mathrm{S}}} \leq m_{T,\varepsilon^{\mathrm{S}}}$.

From $\mathbf{Tx} = \mathbf{b}$ and (3.46) we have

$$\begin{aligned}\mathbf{T}(\mathbf{x} - \hat{\mathbf{x}}) &= \mathbf{b} - \mathbf{b} - \Delta\mathbf{b} + (\Delta\mathbf{T} + \mathbf{F})\hat{\mathbf{x}} \\ &= -\Delta\mathbf{b} + (\Delta\mathbf{T} + \mathbf{F})(\hat{\mathbf{x}} - \mathbf{x}) + (\Delta\mathbf{T} + \mathbf{F})\mathbf{x}.\end{aligned}$$

Consequently,

$$\mathbf{x} - \hat{\mathbf{x}} = -\mathbf{T}^{-1}\Delta\mathbf{b} + \mathbf{T}^{-1}(\Delta\mathbf{T} + \mathbf{F})(\hat{\mathbf{x}} - \mathbf{x}) + \mathbf{T}^{-1}(\Delta\mathbf{T} + \mathbf{F})\mathbf{x},$$

and

$$\|\mathbf{x} - \hat{\mathbf{x}}\| \leq \|\mathbf{T}^{-1}\| \|\Delta\mathbf{b}\| + \|\mathbf{T}^{-1}\| \|\Delta\mathbf{T} + \mathbf{F}\| \|\hat{\mathbf{x}} - \mathbf{x}\| + \|\mathbf{T}^{-1}\| \|\Delta\mathbf{T} + \mathbf{F}\| \|\mathbf{x}\|. \tag{3.47}$$

Using $|\Delta\mathbf{T}| \leq \varepsilon^{\mathrm{R}}|\mathbf{T}|$ and the bound (3.45), $|\Delta\mathbf{T} + \mathbf{F}|$ can be bounded as

$$\begin{aligned}|\Delta\mathbf{T} + \mathbf{F}| &\leq |\Delta\mathbf{T}| + |\mathbf{F}| \\ &\leq \varepsilon^{\mathrm{R}}|\mathbf{T}| + \varepsilon^{\mathrm{S}} m_{T,\varepsilon^{\mathrm{S}}} |\mathbf{T} + \Delta\mathbf{T}| \\ &\leq \varepsilon^{\mathrm{R}}|\mathbf{T}| + \varepsilon^{\mathrm{S}} m_{T,\varepsilon^{\mathrm{S}}} |\mathbf{T}| + \varepsilon^{\mathrm{S}} m_{T,\varepsilon^{\mathrm{S}}} |\Delta\mathbf{T}| \\ &\leq \underbrace{\left(\varepsilon^{\mathrm{R}} + \varepsilon^{\mathrm{S}} m_{T,\varepsilon^{\mathrm{S}}} + \varepsilon^{\mathrm{R}}\varepsilon^{\mathrm{S}} m_{T,\varepsilon^{\mathrm{S}}}\right)}_{\delta_{T,\mathrm{R},\mathrm{S}}} |\mathbf{T}|.\end{aligned}$$

This yields (see, e.g., [11, Lemma 6.6, case (b)]) the estimate

$$\|\Delta\mathbf{T} + \mathbf{F}\| \leq \delta_{T,\mathrm{R},\mathrm{S}} \||\mathbf{T}|\|. \tag{3.48}$$

Using $\|\Delta\mathbf{b}\| \leq \delta_b\|\mathbf{b}\|$ and (3.48) to bound the corresponding terms in (3.47) and using the definition of $\underline{\kappa}_T$ leads to

$$\begin{aligned}\|\mathbf{x} - \hat{\mathbf{x}}\| &\leq \|\mathbf{T}^{-1}\|\delta_b\|\mathbf{b}\| + \|\mathbf{T}^{-1}\|\delta_{T,\mathrm{R},\mathrm{S}}\||\mathbf{T}|\|\|\hat{\mathbf{x}} - \mathbf{x}\| + \|\mathbf{T}^{-1}\|\delta_{T,\mathrm{R},\mathrm{S}}\||\mathbf{T}|\|\|\mathbf{x}\| \\ &= \delta_b\|\mathbf{T}^{-1}\|\|\mathbf{b}\| + \delta_{T,\mathrm{R},\mathrm{S}} \cdot \underline{\kappa}_T \cdot \|\hat{\mathbf{x}} - \mathbf{x}\| + \delta_{T,\mathrm{R},\mathrm{S}} \cdot \underline{\kappa}_T \cdot \|\mathbf{x}\|,\end{aligned}$$

and subsequently,

$$(1 - \delta_{T,\mathrm{R,S}} \cdot \underline{\kappa}_T)\|\mathbf{x} - \hat{\mathbf{x}}\| \leq \delta_b \|\mathbf{T}^{-1}\|\|\mathbf{b}\| + \delta_{T,\mathrm{R,S}} \cdot \underline{\kappa}_T \|\mathbf{x}\|.$$

Utilizing the assumption $\delta_{T,\mathrm{R,S}} \cdot \underline{\kappa}_T < 1$ we have

$$\|\mathbf{x} - \hat{\mathbf{x}}\| \leq \frac{\delta_b \|\mathbf{T}^{-1}\|\|\mathbf{b}\| + \delta_{T,\mathrm{R,S}} \cdot \underline{\kappa}_T \|\mathbf{x}\|}{1 - \delta_{T,\mathrm{R,S}} \cdot \underline{\kappa}_T}.$$

Bounding $\|\mathbf{b}\| \leq \|\mathbf{T}\|\|\mathbf{x}\|$ and dividing both sides by $\|\mathbf{x}\|$ leads to (3.43). □

3.6.2 Finite precision error analysis of mixed precision IC smoother

In this section we use the results from the previous section and present a bound on the finite precision errors in the application of the IC smoother.

Theorem 3.3. *Let $\mathbf{w}_{\mathrm{IC}}$ and $\hat{\mathbf{w}}_{\mathrm{IC}}$ be the approximations computed by applying the IC smoother, Algorithm 3.4, to a vector $\mathbf{f}$ in infinite precision and in finite precision, respectively. Let $\delta_{L,\mathrm{R,S}} = \varepsilon^{\mathrm{R}} + \varepsilon^{\mathrm{S}} \underline{m}_{L,\varepsilon^{\mathrm{S}}} + \varepsilon^{\mathrm{R}}\varepsilon^{\mathrm{S}} \underline{m}_{L,\varepsilon^{\mathrm{S}}}$. Assuming $\delta_{L,\mathrm{R,S}} \cdot \underline{\kappa}_L < 1$ the difference of $\mathbf{w}_{\mathrm{IC}}$ and $\hat{\mathbf{w}}_{\mathrm{IC}}$ in the Euclidean norm can be bounded as*

$$\|\mathbf{w}_{\mathrm{IC}} - \hat{\mathbf{w}}_{\mathrm{IC}}\| \leq (\varepsilon^{\mathrm{S}} \kappa_L + 2\varepsilon^{\mathrm{S}} \underline{m}_{L,\varepsilon^{\mathrm{S}}} \underline{\kappa}_L + 2\varepsilon^{\mathrm{R}} \underline{\kappa}_L + R)\|\mathbf{L}^{-1}\|^2 \|\mathbf{f}\|, \tag{3.49}$$

where the remainder R, contains higher order terms, i.e., terms which involve $(\varepsilon^{\mathrm{S}})^2$, $(\varepsilon^{\mathrm{R}})^2$, or $\varepsilon^{\mathrm{S}}\varepsilon^{\mathrm{R}}$.

Proof. Rounding $\mathbf{f}$ to ε^{S}-precision results in $\hat{\mathbf{f}} = \mathbf{f} + \Delta\mathbf{f}$, $\|\Delta\mathbf{f}\| \leq \varepsilon^{\mathrm{S}}\|\mathbf{f}\|$; see (3.1). Let $\mathbf{v}$ be the exact solution of $\mathbf{L}\mathbf{v} = \mathbf{f}$. Let $\hat{\mathbf{v}}$ be the approximate solution computed in line 2 of Algorithm 3.4.

Let $\delta_{L,\mathrm{R,S}} = \varepsilon^{\mathrm{R}} + \varepsilon^{\mathrm{S}} \underline{m}_{L,\varepsilon^{\mathrm{S}}} + \varepsilon^{\mathrm{R}}\varepsilon^{\mathrm{S}} \underline{m}_{L,\varepsilon^{\mathrm{S}}}$. Using Lemma 3.2, bound (3.43), the relative error of the intermediate result $\hat{\mathbf{v}}$ in line 2 of Algorithm 3.4 can be bounded as

$$\frac{\|\mathbf{v} - \hat{\mathbf{v}}\|}{\|\mathbf{v}\|} \leq \frac{\varepsilon^{\mathrm{S}} \kappa_L + \varepsilon^{\mathrm{R}} \underline{\kappa}_L + \varepsilon^{\mathrm{S}} \underline{m}_{L,\varepsilon^{\mathrm{S}}} \underline{\kappa}_L + \varepsilon^{\mathrm{R}}\varepsilon^{\mathrm{S}} \underline{m}_{L,\varepsilon^{\mathrm{S}}} \underline{\kappa}_L}{1 - \delta_{L,\mathrm{R,S}} \cdot \underline{\kappa}_L}$$

Expanding $(1 - \delta_{L,\mathrm{R,S}} \cdot \underline{\kappa}_L)^{-1}$ as $(1 - \delta_{L,\mathrm{R,S}} \cdot \underline{\kappa}_L)^{-1} = 1 - \delta_{L,\mathrm{R,S}} \cdot \underline{\kappa}_L + R_1$, where the remainder R_1 containing higher order terms, i.e., terms involving $(\varepsilon^{\mathrm{S}})^2$, $(\varepsilon^{\mathrm{R}})^2$, or $\varepsilon^{\mathrm{S}}\varepsilon^{\mathrm{R}}$ we get

$$\begin{aligned}\frac{\|\mathbf{v} - \hat{\mathbf{v}}\|}{\|\mathbf{v}\|} &\leq (\varepsilon^{\mathrm{S}} \kappa_L + \varepsilon^{\mathrm{R}} \underline{\kappa}_L + \varepsilon^{\mathrm{S}} \underline{m}_{L,\varepsilon^{\mathrm{S}}} \underline{\kappa}_L + \varepsilon^{\mathrm{R}}\varepsilon^{\mathrm{S}} \underline{m}_{L,\varepsilon^{\mathrm{S}}} \underline{\kappa}_L)(1 - \delta_{L,\mathrm{R,S}} \cdot \underline{\kappa}_L + R_1) \\ &\leq \underbrace{\varepsilon^{\mathrm{S}} \kappa_L + \varepsilon^{\mathrm{R}} \underline{\kappa}_L + \varepsilon^{\mathrm{S}} \underline{m}_{L,\varepsilon^{\mathrm{S}}} \underline{\kappa}_L + R_2}_{\delta_v},\end{aligned}$$

where R_2 is a reminder containing higher order terms.

The approximation $\mathbf{w}_{\mathrm{IC}}$ can be expressed as $\mathbf{w}_{\mathrm{IC}} = \mathbf{L}^{-\top}\mathbf{v}$. Using bound (3.44) from Lemma 3.2, the result $\hat{\mathbf{w}}_{\mathrm{IC}}$ in line 3 of Algorithm 3.4 can be bounded as

$$\|\mathbf{w}_{\mathrm{IC}} - \hat{\mathbf{w}}_{\mathrm{IC}}\| \leq \frac{\delta_v \|\mathbf{L}^{-1}\|\|\mathbf{v}\| + \delta_{L,\mathrm{R,S}} \cdot \underline{\kappa}_L \|\mathbf{w}_{\mathrm{IC}}\|}{1 - \delta_{L,\mathrm{R,S}} \cdot \underline{\kappa}_L}.$$

Since $\mathbf{w}_{\mathrm{IC}} = (\mathbf{L}^{\top})^{-1}\mathbf{L}^{-1}\mathbf{f}$ and $\mathbf{v} = \mathbf{L}^{-1}\mathbf{f}$, we have $\|\mathbf{w}_{\mathrm{IC}}\| \leq \|\mathbf{L}^{-1}\|^2\|\mathbf{f}\|$, and $\|\mathbf{v}\| \leq \|\mathbf{L}^{-1}\|\|\mathbf{f}\|$. Using again the expansion $(1-\delta_{L,\mathrm{R,S}}\cdot\underline{\kappa}_L)^{-1} = 1-\delta_{L,\mathrm{R,S}}\cdot\underline{\kappa}_L+R_1$ and combining the previous yields

$$\begin{aligned}\|\mathbf{w}_{\mathrm{IC}} - \hat{\mathbf{w}}_{\mathrm{IC}}\| &\leq (\delta_v + \delta_{L,\mathrm{R,S}}\cdot\underline{\kappa}_L)(1-\delta_{L,\mathrm{R,S}}\cdot\underline{\kappa}_L + R_1)\|\mathbf{L}^{-1}\|^2\|\mathbf{f}\| \\ &\leq (\varepsilon^{\mathrm{S}}\kappa_L + 2\varepsilon^{\mathrm{S}}\underline{m}_{L,\varepsilon^{\mathrm{S}}}\underline{\kappa}_L + 2\varepsilon^{\mathrm{R}}\underline{\kappa}_L + R)\|\mathbf{L}^{-1}\|^2\|\mathbf{f}\|,\end{aligned}$$

where the remainder R contains higher order terms. □

We remark that the estimate (3.49) is the worst case scenario estimate. The actual error could be significantly smaller. An important feature of the bound is that it depends only $\underline{m}_L$, i.e., the maximum number of non-zero elements in a row or a column of $\mathbf{L}$, but not on the matrix size of $\mathbf{L}$. The number $\underline{m}_L$ depends on the sparsity pattern of matrix $\mathbf{A}$ and the fill-in occurring in the incomplete factorization.

Assuming that κ_L is approximately the same as $\underline{\kappa}_L$ and the ε^{R}- and ε^{S}-precisions are sufficiently small such that

$$2\varepsilon^{\mathrm{R}}\underline{\kappa}_L\|\mathbf{L}^{-1}\|^2 \ll 1, \quad 2\varepsilon^{\mathrm{S}}\underline{m}_{L,\varepsilon^{\mathrm{S}}}\underline{\kappa}_L\|\mathbf{L}^{-1}\|^2 \ll 1,$$

the relative finite precision error of the IC smoother $\|\mathbf{w}_{\mathrm{IC}} - \hat{\mathbf{w}}_{\mathrm{IC}}\|/\|\mathbf{f}\|$ is small. We see that the requirements on the ε^{R}- and ε^{S}-precisions differ only in the multiplicative constant $\underline{m}_{L,\varepsilon^{\mathrm{S}}}$.

3.6.3 V-cycle correction scheme with IC smoothing

In this section, we discuss application of the IC smoothers inside the V-cycle correction scheme Algorithm 3.3.

We consider that the IC smoothers are used on all fine levels. We have that $\mathbf{M}_j = \mathbf{L}_j^{-\top}\mathbf{L}_j^{-1}$, $j = 1, \ldots, J$, in infinite precision arithmetic. We consider that the $\varepsilon^{\mathrm{R}}_j$- and $\varepsilon^{\mathrm{S}}_j$-precisions are lower than or equal to the $\dot{\varepsilon}_j$- precision on the jth level of the V-cycle, i.e. $\varepsilon^{\mathrm{R}}_j \geq \varepsilon^{\mathrm{S}}_j \geq \dot{\varepsilon}_j$. Using Theorem 3.3 we see that the assumption on the finite precision errors when applying the smoother (3.35) holds with

$$\Lambda_{M_j} \approx \left(\varepsilon^{\mathrm{S}}_j\kappa_{L_j} + 2\varepsilon^{\mathrm{S}}_j\underline{m}_{L_j,\varepsilon^{\mathrm{S}}_j}\underline{\kappa}_{L_j} + 2\varepsilon^{\mathrm{R}}_j\underline{\kappa}_{L_j}\right)\|\mathbf{L}_j^{-1}\|^2.$$

We note that we are not able to theoretically verify the assumption (3.34). It can be however done numerically. In Section 3.8, we combine the result of this section and the results of Theorem 3.2 to discussing requirements on the finite precisions used inside a V-cycle scheme with IC smoothing when solving elliptic PDEs.

3.7 Scaling system matrices and right-hand sides

Rounding matrices or vectors to a low precision arithmetic can result in overflow or underflow; we refer e.g., to the discussions in [13] and [24]. Scaling the data before rounding can help to partially overcome this issue.

We present a simple scaling strategy for the system and prolongation matrices in a multigrid hierarchy, which preserves the Galerkin condition. The matrix $\mathbf{A}_j$ on the jth level, $j = 0, \ldots, J$, is scaled as $\bar{\mathbf{A}}_j = s_j\mathbf{A}_j$, where $s_j = 1/\max_{k,\ell}|[\mathbf{A}_j]_{k,\ell}|$.

The prolongation matrix on the jth level, $j = 0, \dots, J-1$, is scaled as $\bar{\mathbf{P}}_j = \frac{\sqrt{s_{j-1}}}{\sqrt{s_j}}\mathbf{P}_j$. The Galerkin condition holds (in infinite precision) also for the scaled matrices

$$\bar{\mathbf{P}}_j^{\top}\bar{\mathbf{A}}_j\bar{\mathbf{P}}_j = \frac{\sqrt{s_{j-1}}}{\sqrt{s_j}}\mathbf{P}_j^{\top} s_j \mathbf{A}_j \frac{\sqrt{s_{j-1}}}{\sqrt{s_j}}\mathbf{P}_j = s_{j-1}\mathbf{P}_j^{\top}\mathbf{A}_j\mathbf{P}_j = s_{j-1}\mathbf{A}_{j-1} = \bar{\mathbf{A}}_{j-1}.$$

An approximate solution of the original problem on the finest level $\mathbf{A}_J\mathbf{y} = \mathbf{f}_J$ can be computed using the multigrid V-cycle correction scheme with the scaled matrices applied to $\bar{\mathbf{A}}_J\mathbf{y} = s_J\mathbf{f}_J$.

Scaling can be also applied to a right-hand side vector before calling a correction scheme or a smoothing routine, where the right-hand side is rounded to a lower precision arithmetic; see [7, Section 6]. We first compute the infinity norm of the right-hand side vector $\mathbf{f}$, i.e., $s_f = \|\mathbf{f}\|_\infty$. The right-hand side is then scaled as $\bar{\mathbf{f}} = s_f^{-1}\mathbf{f}$ and the correction routine is called with the scaled vector $\bar{\mathbf{f}}$. The result of the correction routine is subsequently re-scaled back by multiplying with s_f.

We remark that the discussed scaling may partially help with staying inside a range of a low precision arithmetic; however, it does not guarantee that the subsequent computation in low precision arithmetic will not break down due to overflow or underflow.

3.8 Numerical experiments

In this section, we present a series of numerical experiments illustrating the theoretical results. We consider solving elliptic PDEs discretized using the continuous Galerkin finite element method. We use the IR method with a geometric multigrid V-cycle correction scheme (IR-V-cycle) with IC smoothing. We first describe the model problems and their discretization and subsequently present the numerical experiments.

3.8.1 Model problems, discretization, and data generation

We consider the following 1D and 3D elliptic PDEs. The 1D problem consists of finding $u : (0,1) \to \mathbb{R}$ such that

$$-u'' = f \quad \text{in } (0,1), \quad u(0) = u(1) = 0,$$

where the right-hand side function f is chosen to correspond to the manufactured solution

$$u(x) = x(x-1)\sin(2\pi x) \quad x \in (0,1).$$

The 3D problems feature different anisotropy in the x-axis. We aim to find $u : (0,1)^3 \to \mathbb{R}$ such that

$$\begin{aligned} -\nabla \cdot (\mathbb{K}\nabla u) = 1, &\quad \text{in } (0,1)^3, \quad \mathbb{K} = \operatorname{diag}(\epsilon, 1, 1), \\ u = 0, &\quad \text{on } \partial(0,1)^3, \end{aligned}$$

where $\epsilon = 1, 10^{-2}$, 10^{-4}, or 10^{-6}.

The 1D problem is discretized by the continuous Galerkin FE method with piecewise polynomials of degree five on a hierarchy of 15 uniformly refined meshes.

We choose this high order element space since it leads to systems which are more difficult to solve. When choosing lower order FE spaces, the systems were approximated to a required tolerance after only one iteration of IR-V-cycle with IC, in settings where we allowed fill-in when computing the IC factors. We note that IC smoothing may not be the most effective smoothing routine for this concrete problem. We, however, still consider the 1D problem since we are able to run the computation with V-cycle corrections schemes with up to 15 levels. This would not be possible for 2D or 3D problems with high anisotropy (where using the IC smoothers may make more sense) due to the size of the problems. The 3D problems are discretized using the continuous Galerkin FE method with piecewise linear functions on a hierarchy of 7 uniformly refined triangulations. We discretize the problems on each level obtaining a geometric multigrid hierarchy. We consider the standard prolongation matrices associated with the finite elements spaces.

The matrices are assembled in the finite element software FEniCS [3] in double precision. The FEniCS matrix assembly uses all nodes of the mesh. The homogeneous Dirichlet boundary condition is then applied by setting to zero all non-diagonal elements in rows and columns which correspond to nodes on the boundary and setting to zero the corresponding elements in the right-hand side vector. We modify the stiffness matrices, the prolongation matrices, and the right-hand side vectors so that the resulting systems contain just free-node variables. The Galerkin condition is then satisfied on all coarse levels. The numbers of degrees of freedom (DoFs) on each level can be found in Table 3.1. We scale the system and prolongation matrices and the right-hand side vectors using the strategy described in Section 3.7. We also filter values at around the level of double precision unit roundoff; these values would most likely be equal to zero in exact precision, but are present due to the use of finite precision computation. The data and codes for reproducing the results of the experiments can be found at `https://doi.org/10.5281/zenodo.13858607`.

level	1D	3D
1	24	8
2	49	125
3	99	1,331
4	199	12,167
5	399	103,823
6	799	857,375
7	1,599	6,967,871
8	3,199	
9	6,399	
10	12,799	
11	25,599	
12	51,199	
13	102,399	
14	204,799	
15	409,599	

Table 3.1 Number of DoF on individual levels of multigrid hierarchies.

3.8.2 Experiment 1: Finding the lowest precisions for the inner V-cycle solver while preserving the IR double precision convergence rate

We solve the discretized 1D problem using the IR-V-cycle method (Algorithms 3.1 and 3.3) with smoothing based on IC factorization (Algorithm 3.4). The computation is done in MATLAB version 2023a. The data are imported from FEniCS. The goals of this experiment are to

- show that the $\varepsilon_j^{\mathrm{S}}$- and $\varepsilon_j^{\mathrm{R}}$-precisions used when applying the IC smoother on the jth level of the V-cycle correction scheme can be significantly lower than the $\dot{\varepsilon}_j$-precision used for computing the residual, restriction, projection, and correction on the jth level, and
- show that the requirements on the $\varepsilon_j^{\mathrm{S}}$-, $\varepsilon_j^{\mathrm{R}}$- and $\dot{\varepsilon}_j$-precisions for a V-cycle scheme with smoothing based on IC factorization with fill-in may be higher than for the IC factorization with zero fill-in.

We run the experiment in several different settings. We assume that the coarsest level is fixed and we solve the problems $\mathbf{A}_J\mathbf{x}_J = \mathbf{b}_J$, $J = 2, \ldots, 14$, using the IR-V-cycle method with $J+1$ levels. We consider the V-cycle correction scheme formulated above; in particular, we use only one pre-smoothing step and no post-smoothing. The $\dot{\varepsilon}_j$-precision is fixed on all fine levels i.e., $\dot{\varepsilon}_j = \dot{\varepsilon}_J$, $j = 1, \ldots, J$. The same holds for the precisions used in IC smoothers, where we additionally assume that the $\varepsilon_j^{\mathrm{R}}$-precision is the same as the $\varepsilon_j^{\mathrm{S}}$-precision, i.e., $\varepsilon_j^{\mathrm{R}} = \varepsilon_j^{\mathrm{S}} = \varepsilon_J^{\mathrm{S}}$, $j = 1, \ldots, J$. We consider two variants of IC smoothers based on different IC factorizations. Both factorizations are computed using the MATLAB ichol function in double precision. We use the factorizations with zero fill-in, IC(0), and the factorizations with local dropping tolerance $5 \cdot 10^{-3}$, ICT(dpt=$5 \cdot 10^{-3}$); see the MATLAB ichol documentation. Allowing fill-in in the IC factorization typically yields a better approximation of the matrix $\mathbf{A}$ by $\mathbf{L}\mathbf{L}^\top$ and consequently leads to a faster convergence rate of the V-cycle scheme with the corresponding IC smoothing. The solver on the coarsest-level is the MATLAB backslash operator applied in double precision.

Expectations based on theory

Before describing the experiment in more detail and presenting its result, we look at the properties of the system matrices and the IC factors and discuss the expected requirements on the $\dot{\varepsilon}_J$-precision and $\varepsilon_J^{\mathrm{S}}$-precision based on the theoretical results presented in Section 3.6.3.

The approximate values of $\kappa_{A_j}^{\frac{1}{2}}$, κ_{L_j}, $\underline{\kappa}_{L_j}$ and $\|\mathbf{L}_j^{-1}\|^2$ are summarized in Table 3.2. We see that $\kappa_{A_{j+1}}^{\frac{1}{2}}$ is approximately twice as large as $\kappa_{A_j}^{\frac{1}{2}}$. The condition numbers κ_{L_j} and $\underline{\kappa}_{L_j}$ for the same variant of the IC factorization do not differ substantially. Their values also do not significantly change on different fine levels. The same holds for the terms $\|\mathbf{L}_j^{-1}\|^2$. We have also approximately computed the following properties (they are nearly the same on all levels) $\|\mathbf{A}_j\| \approx |||\mathbf{A}_j||| \approx 2.55$, $\|\mathbf{A}_{j-1}^{-1}\|^{\frac{1}{2}}/\|\mathbf{A}_j^{-1}\|^{\frac{1}{2}} \approx 2$, $\|\mathbf{P}_j\| \approx |||\mathbf{P}_j||| \approx 3.14$, $m_{A_j} = 11$, $\underline{m}_{P_j} = 11$, $\underline{m}_{L_j} = 10$ for

	$\mathbf{A}_j$	IC(0)			ICT(dpt$= 5 \cdot 10^{-3}$)		
level	$\kappa_{\mathbf{A}_j}^{\frac{1}{2}}$	$\kappa_{\mathbf{L}_j}$	$\underline{\kappa}_{\mathbf{L}_j}$	$\Vert\mathbf{L}_j^{-1}\Vert^2$	$\kappa_{\mathbf{L}_j}$	$\underline{\kappa}_{\mathbf{L}_j}$	$\Vert\mathbf{L}_j^{-1}\Vert^2$
1	3.75E+01	1.06E+01	1.06E+01	3.92E+01	2.83E+01	3.47E+01	3.16E+02
2	7.50E+01	1.08E+01	1.08E+01	3.98E+01	3.50E+01	4.34E+01	4.80E+02
3	1.50E+02	1.08E+01	1.08E+01	3.99E+01	3.74E+01	4.67E+01	5.50E+02
4	3.00E+02	1.08E+01	1.08E+01	3.99E+01	3.81E+01	4.76E+01	5.70E+02
5	6.01E+02	1.08E+01	1.08E+01	3.99E+01	3.83E+01	4.78E+01	5.76E+02
6	1.20E+03	1.08E+01	1.08E+01	3.99E+01	3.84E+01	4.79E+01	5.77E+02
7	2.40E+03	1.08E+01	1.08E+01	3.99E+01	3.84E+01	4.79E+01	5.77E+02
8	4.81E+03	1.08E+01	1.08E+01	3.99E+01	3.84E+01	4.79E+01	5.77E+02
9	9.61E+03	1.08E+01	1.08E+01	3.99E+01	3.84E+01	4.79E+01	5.77E+02
10	1.92E+04	1.08E+01	1.08E+01	3.99E+01	3.84E+01	4.79E+01	5.77E+02
11	3.84E+04	1.08E+01	1.08E+01	3.99E+01	3.84E+01	4.79E+01	5.77E+02
12	7.69E+04	1.08E+01	1.08E+01	3.99E+01	3.84E+01	4.79E+01	5.77E+02
13	1.54E+05	1.08E+01	1.08E+01	3.99E+01	3.84E+01	4.79E+01	5.77E+02
14	3.08E+05	1.08E+01	1.08E+01	3.99E+01	3.84E+01	4.79E+01	5.77E+02
15	6.15E+05	1.08E+01	1.08E+01	3.99E+01	3.84E+01	4.79E+01	5.77E+02

Table 3.2 1D problem. Properties of $\mathbf{A}_j$ and $\mathbf{L}_j$.

both IC(0) and ICT(dpt=$5 \cdot 10^{-3}$); we note that $\underline{m}_{L_j}$ is the maximum number of nonzero entries in both a row or a column of $\mathbf{L}_j$.

Knowing approximate values of these properties, we can use the results from Theorems 3.2 and 3.3 and discuss the requirements on the finite precisions used inside the V-cycle scheme. The following discussion works for both variants of the IC factorization. Since we are using the IC smoother, we have $\mathbf{M}_j = \mathbf{L}_j^{-\top}\mathbf{L}_j^{-1}$ in the notation from Section 3.5. As stated before, we consider that the precisions used when applying the smoothers are fixed on all levels, i.e., $\varepsilon_j^{\mathrm{R}} = \varepsilon_j^{\mathrm{S}} = \varepsilon_J^{\mathrm{S}}$. Let $\delta_{L,\mathrm{R,S}} = \varepsilon_J^{\mathrm{S}} + \varepsilon_J^{\mathrm{S}}\underline{m}_{L_J,\varepsilon_J^{\mathrm{S}}} + (\varepsilon_J^{\mathrm{S}})^2\underline{m}_{L_J,\varepsilon_J^{\mathrm{S}}}$. We assume that the $\varepsilon_J^{\mathrm{S}}$-precision is chosen such that $\delta_{L,\mathrm{R,S}} \cdot \underline{\kappa}_{L_J} < 1$. Since $\kappa_{L_J} \approx \kappa_{L_j} \approx \underline{\kappa}_{L_j} \approx \underline{\kappa}_{L_J}$ and $\underline{m}_{L_j}$ is constant on all levels, using Theorem 3.3 yields

$$\Lambda_{M_j} \approx \Lambda_{M_J} \approx 2\varepsilon_J^{\mathrm{S}}\underline{m}_{L_J}\kappa_{L_J}\Vert\mathbf{L}_J^{-1}\Vert^2. \tag{3.50}$$

Since the coarsest-level solver is applied in double precision to a small well-conditioned problem, we expect the associated finite precision error to be negligible. We also assume that the theoretical assumptions on the convergence of the smoothers (3.39) and the uniform convergence of the V-cycle scheme (3.38) are satisfied, although we do not verify them here. Theorem 3.2 yields the following estimate on the finite precision error of the V-cycle correction scheme

$$\begin{aligned}\frac{\Vert\mathbf{y}_{\mathrm{V},J} - \hat{\mathbf{y}}_{\mathrm{V},J}\Vert_{\mathbf{A}_J}}{\Vert\mathbf{y}_J\Vert_{\mathbf{A}_J}} \lesssim &\sum_{j=1}^{J} 3\Vert\mathbf{A}_j\Vert \cdot 2 \cdot \varepsilon_J^{\mathrm{S}}\underline{m}_{L_J}\kappa_{L_J}\Vert\mathbf{L}_J^{-1}\Vert^2 \\ &+ \sum_{j=1}^{J} \dot{\varepsilon}_j(C_{1,j}\kappa_{A_j}^{\frac{1}{2}}|||\mathbf{A}_j|||\Vert\mathbf{L}_J^{-1}\Vert^2 + C_{2,j}) \\ &+ 3\sum_{j=1}^{J} \dot{\varepsilon}_j\Vert\mathbf{A}_j\Vert\Vert\mathbf{L}_J^{-1}\Vert^2,\end{aligned} \tag{3.51}$$

where the constants $C_{1,j}$ and $C_{2,j}$ depends only on $\Vert\mathbf{P}_j\Vert$, $|||\mathbf{P}_j|||$, $m_{A_j,\dot{\varepsilon}_j}$, $\underline{m}_{P_j,\dot{\varepsilon}_j}$ and the ratio $\Vert\mathbf{A}_{j-1}^{-1}\Vert^{\frac{1}{2}}/\Vert\mathbf{A}_j^{-1}\Vert^{\frac{1}{2}}$.

Since $\dot{\varepsilon}_j = \dot{\varepsilon}_J$-precision is fixed on all levels, $2\kappa_{A_j}^{\frac{1}{2}} \approx \kappa_{A_{j+1}}^{\frac{1}{2}}$ and the values of m_{A_j}, $\underline{m}_{P_j}$, $\underline{m}_{L_j}$, $\Vert\mathbf{A}_j\Vert$, $|||\mathbf{A}_j|||$ $\Vert\mathbf{P}_j\Vert$, and $|||\mathbf{P}_j|||$ do not significantly differ on different levels, we can estimate the folowing sum as

$$\begin{aligned}\sum_{j=1}^{J} \dot{\varepsilon}_j C_{1,j} \kappa_{A_j}^{\frac{1}{2}} \| |\mathbf{A}_j| \| \| |\mathbf{L}_j^{-1}\|^2 &\lesssim \dot{\varepsilon}_J C_{1,J} \| |\mathbf{A}_J| \| \| |\mathbf{L}_J^{-1}\|^2 \sum_{j=1}^{J} \frac{\kappa_{A_J}^{\frac{1}{2}}}{2^{J-j}} \\ &\lesssim \dot{\varepsilon}_J C_{1,J} \| |\mathbf{A}_J| \| \| |\mathbf{L}_J^{-1}\|^2 \cdot 2 \cdot \kappa_{A_J}^{\frac{1}{2}}, \end{aligned} \tag{3.52}$$

where we have also used the upper bound on a sum of geometric sequence.

Using (3.52) and neglecting the terms $\sum_{j=1}^{J} \dot{\varepsilon}_j C_{2,j}$ and $3\sum_{j=1}^{J} \dot{\varepsilon}_j \|\mathbf{A}_j\| \| |\mathbf{L}_J^{-1}\|^2$, which are much smaller that the other terms in the estimate (3.51), the estimate (3.51) can be approximately simplified to

$$\frac{\|\mathbf{y}_{\mathrm{V},J} - \hat{\mathbf{y}}_{\mathrm{V},J}\|_{\mathbf{A}_J}}{\|\mathbf{y}_J\|_{\mathbf{A}_J}} \lesssim 6J\varepsilon_J^{\mathrm{S}} \underline{m}_{L_J,\varepsilon_J^{\mathrm{S}}} \|\mathbf{A}_J\| \| |\mathbf{L}_J^{-1}\|^2 \kappa_{L_J} + 2\dot{\varepsilon}_J C_{1,J} \| |\mathbf{A}_J| \| \| |\mathbf{L}_J^{-1}\|^2 \kappa_{A_J}^{\frac{1}{2}}. \tag{3.53}$$

If ε_J- and $\varepsilon_J^{\mathrm{S}}$-precisions are chosen such that the right-hand side of (3.53) is much smaller than one, the convergence rate of the V-cycle correction scheme should not be significantly influenced by the finite precision errors. Estimate (3.53) allows us to make the following predictions.

Since $\underline{\kappa}_{L_J}$ is constant on the fine levels (for larger $J = 6, \ldots, 14$), whereas $\kappa_{A_J}^{\frac{1}{2}}$ grows approximately by a factor of 2 with each finer level, we expect that on fine levels (for larger $J = 8, \ldots, 14$), $\varepsilon_J^{\mathrm{S}}$ could be significantly smaller than $\dot{\varepsilon}_J$ while preserving the same convergence rate. This is valid for both the IC(0) and ICT(dpt=$5 \cdot 10^{-3}$) variants.

Since the values of $\|\mathbf{L}_J^{-1}\|^2$ and $\underline{\kappa}_{L_J}$ are larger for the variant with ICT(dpt=$5 \cdot 10^{-3}$) than the corresponding values for the variant IC(0), we expect that the $\dot{\varepsilon}_J$- and $\varepsilon_J^{\mathrm{S}}$-precisions for the variant with ICT(dpt=$5 \cdot 10^{-3}$) might have to be chosen higher than for the IC(0) variant.

Detailed description of the experiments

We describe the details of the experiment. We consider that the residual computation and the correction in IR are both done in double precision. The solver on the coarsest-level, the MATLAB backslash operator, is applied in double precision to a problem with matrix $\mathbf{A}_0$ rounded to the $\dot{\varepsilon}_J$-precision, and the computed coarsest-level approximation is rounded to the $\dot{\varepsilon}_J$-precision. As previously stated, we assume that the coarsest level is fixed and we solve problems $\mathbf{A}_J\mathbf{x}_J = \mathbf{b}_J$, $J = 2, \ldots, 14$, using IR-V-cycle with $J + 1$ levels. The IR-V-cycle method is run starting with zero initial approximation and stopped when the absolute error in the $\mathbf{A}_J$-norm is (approximately) less than 10^{-5} (the solution for computing the error is approximated by the MATLAB backslash operator in double precision). The initial error is approximately 10^{-1}. The level of attainable accuracy in the $\mathbf{A}_J$-norm is different for each problem. It is approximately 10^{-13} for the problem with $J = 2$ and it grows up to $2 \cdot 10^{-6}$ for the problem with $J = 14$. The stopping tolerance 10^{-5} is chosen since it is attainable for all problems.

We use the Advanpix toolbox [2], version 5.1.0.15432, for simulating low precision floating point arithmetic. It allows only to specify the number of decimal digits d, simulating the floating point precision with approximate unit roundoff

10^{-d}. It has 64 bits for representing the exponent, beside the variant with $d = 34$ where it is 15 bits; see e.g., [25, Section 8]. The large number of bits for representing the exponent yields that the computation is not affected by the limited range as when using the standard single and especially half precision, which have 8 and 5 bits for storing the exponent, respectively.

We first run the computation with all the precisions set to double. Then we assume that the $\dot{\varepsilon}_J$- and $\varepsilon^{\mathrm{S}}_J$-precisions are the same and we run the computation using the Advanpix toolbox simulating $\dot{\varepsilon}_J = 10^{-\dot{d}_J}$, for $\dot{d}_J = 1, 2, \ldots$. We find the smallest $\dot{d}_J$, denoted as $\dot{d}_{J,\mathrm{min}}$, for which the method converges in the same number of IR iterations as the corresponding variant in double precision. Further, we fix $\dot{d}_J = \dot{d}_{J,\mathrm{min}}$ and run the experiments simulating $\varepsilon^{\mathrm{S}}_J = \varepsilon^{\mathrm{R}}_J = 10^{-d^{\mathrm{S}}_J}$, $d^{\mathrm{S}}_J = 1, 2, \ldots$. We again find the minimal d^{S}_J, denoted as $d^{\mathrm{S}}_{J,\mathrm{min}}$, for which the method converges in the same number of IR iterations as the corresponding variant in double precision.

Results

The results of the experiments containing the values of $\dot{d}_{J,\mathrm{min}}$ and $d^{\mathrm{S}}_{J,\mathrm{min}}$ for the variants with IC(0) or ICT(dpt=$5 \cdot 10^{-3}$) are summarized in Figure 3.1 together with the required number of IR iterations. We see that the variants with ICT(dpt=$5 \cdot 10^{-3}$) requires significantly fewer IR iterations to reach the chosen tolerance than the variants with IC(0). Regardless of the variant of the IC factorization, the values of $d^{\mathrm{S}}_{J,min}$ corresponding to the $\varepsilon^{\mathrm{S}}_J$-precision used in the smoothing are smaller than the corresponding values of $\dot{d}_{J,min}$ corresponding to the $\dot{\varepsilon}_J$-precision. Moreover $\dot{d}_{J,min}$ increases when increasing J, while $d^{\mathrm{S}}_{J,min}$ stays constant. This illustrates that the $\varepsilon^{\mathrm{S}}_J$-precision may be, in some settings, significantly lower than the $\dot{\varepsilon}_J$-precision. We observe that the values of $\dot{d}_{J,\mathrm{min}}$ for the variant with ICT(dpt=$5 \cdot 10^{-3}$) are larger than or equal to the corresponding values for the variant with IC(0). The same holds for the values of $d^{\mathrm{S}}_{J,\mathrm{min}}$. We see that the variant with ICT(dpt=$5 \cdot 10^{-3}$) requires higher or equal $\dot{\varepsilon}_J$-precision and $\varepsilon^{\mathrm{S}}_J$-precision than the variant with IC(0). On the other hand, the convergence rate of the variant with IC(0) is lower. We conclude that the results are in alignment with the predictions made based on the theory.

Even though we run the experiments with the $\dot{\varepsilon}_J$-precision fixed for all levels, this experiment also illustrates that the $\dot{\varepsilon}_j$-precision $j = 1 \ldots, J$, could be chosen to be lower on the coarse levels and progressively increased.

3.8.3 Experiment 2: solving 3D elliptic PDEs with high anisotropy on GPUs using the Ginkgo library

We solve the discretized 3D problems with different values of anisotropy in the x-axis, $\epsilon = 1, 10^{-2}, 10^{-4}, 10^{-6}$, using the IR-V-cycle (Algorithms 3.1 and 3.3) with smoothing based on IC factorization (Algorithm 3.4). The computation is done using the Ginkgo library [5, 8] on a GPU. The data are imported from FEniCS. The goal of this experiment is to show that applying the IC smoothers in low-precision when solving complicated problems on GPUs may result in a significant speedup in the runtime in comparison to using uniform double precision.

We use a V-cycle correction scheme with 7 levels, with one pre-smoothing iteration of the IC smoother and no post-smoothing. We again assume that

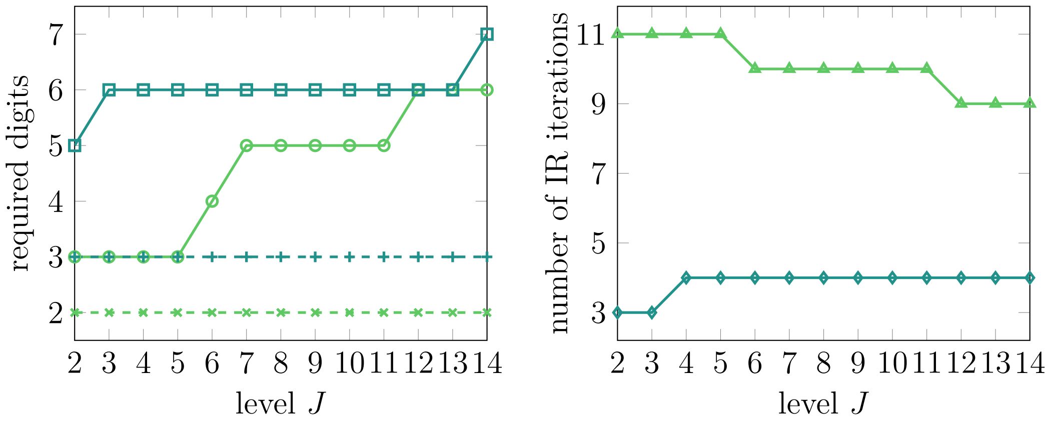


Figure 3.1 Result of Experiment 1. 1D problem solved by IR-V-cycle with IC smoother. The plot on the left contains the values of $\dot{d}_{J,\mathrm{min}}$ and $d^{\mathrm{S}}_{J,\mathrm{min}}$, i.e., the minimal values of $\dot{d}_J$ and d^{S}_J such that the variant with $\dot{\varepsilon}_J = 10^{-\dot{d}_J}$-precision and $\varepsilon^{\mathrm{S}}_J = \varepsilon^{\mathrm{R}}_J = 10^{-d^{\mathrm{S}}_J}$-precision converges in the same number of IR iterations as the corresponding variant in double precision. The lines are labeled as $\dot{d}_{J,\mathrm{min}}$ (—○—), $d^{\mathrm{S}}_{J,\mathrm{min}}$ (- × -) for the variant with IC(0) and $\dot{d}_{J,\mathrm{min}}$ (—□—), $d^{\mathrm{S}}_{J,\mathrm{min}}$ (- + -) for the variant with ICT(dpt=$5 \cdot 10^{-3}$). The plot on the right contains the number of IR iterations required for convergence for the variant with IC(0) (—▲—) and ICT(dpt=$5 \cdot 10^{-3}$) (—◆—).

the $\dot{\varepsilon}_j$-, $\varepsilon^{\mathrm{R}}_j$- and $\varepsilon^{\mathrm{S}}_j$ precisions are fixed on all fine levels, i.e., $\dot{\varepsilon}_j = \dot{\varepsilon}_J$, $\varepsilon^{\mathrm{R}}_j = \varepsilon^{\mathrm{R}}_J$, and $\varepsilon^{\mathrm{S}}_j = \varepsilon^{\mathrm{S}}_J$, $j = 1, \ldots, J$. In contrast to the previous experiment, we consider that the $\varepsilon^{\mathrm{S}}_J$- and $\varepsilon^{\mathrm{R}}_J$-precisions may differ. We consider two variants of the IC factorization: the variant with zero fill-in, IC(0), and the variant which limits the fill-in to the maximum of two times the number of nonzeros of the system matrix, ICT(fill-in=2). The factorizations are computed in double precision on CPUs and subsequently moved to the GPU. The coarsest-level solver is a direct solver applied in double precision.

Expectations based on theory

Before describing the experiment in more detail, we again look at properties of the system matrices and the IC factors and discuss the expected requirements on the finite precisions inside the V-cycle scheme. The approximate values of $\kappa^{\frac{1}{2}}_{A_j}$, κ_{L_j}, $\underline{\kappa}_{L_j}$, and $\|\mathbf{L}_j^{-1}\|^2$ approximately computed in MATLAB are summarized in Tables 3.3 to 3.6 for the problems with $\epsilon = 1, 10^{-2}, 10^{-4}$, and 10^{-6}, respectively. Due to the size of the problems on the finest level, we were not able to approximate the values of $\kappa^{\frac{1}{2}}_{A_j}$, $j = 5, 6$, we include in the tables their extrapolated values using $\kappa^{\frac{1}{2}}_{A_j} = 2\kappa^{\frac{1}{2}}_{A_{j-1}}$. We see that for all values of the anisotropy ϵ, $\kappa^{\frac{1}{2}}_{A_j}$ grows by approximately a factor of two with each finer level. For the problems with higher anisotropy, the values of $\kappa^{\frac{1}{2}}_{A_j}$ are slightly larger than the corresponding values for the problems with lower anisotropy. The condition numbers κ_{L_j} and $\underline{\kappa}_{L_j}$ for the same size of the anisotropy and the same variant of the IC factorization do not significantly differ. Their values also do not substantially change on finer levels. The same holds for the terms $\|\mathbf{L}_j^{-1}\|^2$. For the problems with higher anisotropy, the values of κ_{L_j}, $\underline{\kappa}_{L_j}$, and $\|\mathbf{L}_j^{-1}\|^2$ are slightly larger than

	$\mathbf{A}_j$	IC(0)			ICT(fill-in=2)		
level	$\kappa^{\frac{1}{2}}_{A_j}$	κ_{L_j}	$\underline{\kappa}_{L_j}$	$\|\mathbf{L}_j^{-1}\|^2$	κ_{L_j}	$\underline{\kappa}_{L_j}$	$\|\mathbf{L}_j^{-1}\|^2$
2	3.84E+00	2.66E+00	2.66E+00	4.48E+00	3.38E+00	3.77E+00	7.76E+00
3	7.80E+00	3.09E+00	3.09E+00	5.66E+00	4.48E+00	5.13E+00	1.31E+01
4	1.56E+01	3.23E+00	3.23E+00	6.07E+00	4.90E+00	5.65E+00	1.55E+01
5	3.13E+01	3.27E+00	3.27E+00	6.21E+00	5.10E+00	5.89E+00	1.67E+01
6	6.27E+01 *	3.29E+00	3.29E+00	6.26E+00	5.12E+00	5.93E+00	1.69E+01
7	1.25E+02 *	3.29E+00	3.29E+00	6.26E+00	5.15E+00	5.96E+00	1.70E+01

Table 3.3 3D problem, $\epsilon = 1$. Properties of $\mathbf{A}_j$ and $\mathbf{L}_j$. $^{*)}$Values of $\kappa^{\frac{1}{2}}_{A_j}$, $j = 5, 6$ are extrapolated using $\kappa^{\frac{1}{2}}_{A_j} = 2\kappa^{\frac{1}{2}}_{A_{j-1}}$.

	$\mathbf{A}_j$	IC(0)			ICT(fill-in=2)		
level	$\kappa^{\frac{1}{2}}_{A_j}$	κ_{L_j}	$\underline{\kappa}_{L_j}$	$\|\mathbf{L}_j^{-1}\|^2$	κ_{L_j}	$\underline{\kappa}_{L_j}$	$\|\mathbf{L}_j^{-1}\|^2$
2	3.93E+00	3.00E+00	3.00E+00	5.50E+00	3.90E+00	4.36E+00	9.89E+00
3	7.95E+00	3.58E+00	3.58E+00	7.18E+00	7.02E+00	8.57E+00	3.06E+01
4	1.60E+01	3.77E+00	3.77E+00	7.89E+00	9.39E+00	1.17E+01	5.41E+01
5	3.19E+01	3.88E+00	3.88E+00	8.28E+00	1.04E+01	1.31E+01	6.62E+01
6	6.39E+01 *	3.92E+00	3.92E+00	8.40E+00	1.07E+01	1.35E+01	6.95E+01
7	1.28E+02 *	3.97E+00	3.97E+00	8.55E+00	1.07E+01	1.36E+01	6.96E+01

Table 3.4 3D problem, $\epsilon = 10^{-2}$. Properties of $\mathbf{A}_j$ and $\mathbf{L}_j$. $^{*)}$Values of $\kappa^{\frac{1}{2}}_{A_j}$, $j = 5, 6$ are extrapolated using $\kappa^{\frac{1}{2}}_{A_j} = 2\kappa^{\frac{1}{2}}_{A_{j-1}}$.

the corresponding values for the problems with lower anisotropy. We have also approximately computed the following properties: regardless the size of the anisotropy ϵ, $\max_j \|\mathbf{A}_j\| \approx \max_j |||\mathbf{A}_j||| \approx 1.64$, $\max_j |||\mathbf{P}_j||| \approx \max_j \|\mathbf{P}_j\| \approx 2.3$, $\|\mathbf{A}_{j-1}^{-1}\|^{\frac{1}{2}}/\|\mathbf{A}_j^{-1}\|^{\frac{1}{2}} \approx 2$, $\max_j m_{A_j} = 7$, $\max_j \underline{m}_{P_j} = 99$, and $\max_j \underline{m}_{L_j} = 7$ for IC(0). The value of $\max_j \underline{m}_{L_j}$ for ICT(fill-in=2) are $31, 89, 91$ and 91 for the problems with anisotropy ϵ equals to $1, 10^{-2}, 10^{-4}$ and 10^{-6}, respectively.

As for the previous experiment, knowing approximate values of these properties we can again use the results from Theorems 3.2 and 3.3 and discuss the requirements on the finite precisions used inside the V-cycle scheme. Using analogous discussion as for the previous experiment, with the difference that $\varepsilon_J^{\mathrm{S}}$ and $\varepsilon_J^{\mathrm{R}}$ might differ, it can be shown that

$$\frac{\|\mathbf{y}_{\mathrm{V}} - \hat{\mathbf{y}}_{\mathrm{V}}\|_{\mathbf{A}}}{\|\mathbf{y}\|_{\mathbf{A}}} \lesssim 6J(\varepsilon_J^{\mathrm{S}}\underline{m}_{L_J,\varepsilon_J^{\mathrm{S}}} + \varepsilon_J^{\mathrm{R}})\|\mathbf{A}_J\|\|\mathbf{L}_J^{-1}\|^2\kappa_{L_J} + 2\dot{\varepsilon}_J C_{1,J}|||\mathbf{A}_J|||\|\mathbf{L}_J^{-1}\|^2\kappa^{\frac{1}{2}}_{A_J},$$

where the constant $C_{1,J}$ depends only on $\|\mathbf{P}_J\|$, $|||\mathbf{P}_J|||$, $m_{A_J,\dot{\varepsilon}_J}$, $\underline{m}_{P_J,\dot{\varepsilon}_J}$ and the

	$\mathbf{A}_j$	IC(0)			ICT(fill-in=2)		
level	$\kappa^{\frac{1}{2}}_{A_j}$	κ_{L_j}	$\underline{\kappa}_{L_j}$	$\|\mathbf{L}_j^{-1}\|^2$	κ_{L_j}	$\underline{\kappa}_{L_j}$	$\|\mathbf{L}_j^{-1}\|^2$
2	3.93E+00	3.02E+00	3.02E+00	5.57E+00	3.93E+00	4.41E+00	1.01E+01
3	7.96E+00	3.61E+00	3.61E+00	7.30E+00	7.43E+00	9.11E+00	3.42E+01
4	1.60E+01	3.82E+00	3.82E+00	8.04E+00	1.03E+01	1.29E+01	6.49E+01
5	3.19E+01	3.93E+00	3.93E+00	8.45E+00	1.15E+01	1.46E+01	8.11E+01
6	6.39E+01 *	3.97E+00	3.97E+00	8.57E+00	1.18E+01	1.51E+01	8.52E+01
7	1.28E+02 *	4.01E+00	4.01E+00	8.73E+00	1.20E+01	1.53E+01	8.73E+01

Table 3.5 3D problem, $\epsilon = 10^{-4}$. Properties of $\mathbf{A}_j$ and $\mathbf{L}_j$. $^{*)}$Values of $\kappa^{\frac{1}{2}}_{A_j}$, $j = 5, 6$ are extrapolated using $\kappa^{\frac{1}{2}}_{A_j} = 2\kappa^{\frac{1}{2}}_{A_{j-1}}$.

	$\mathbf{A}_j$	IC(0)			ICT(fill-in=2)		
level	$\kappa_{A_j}^{\frac{1}{2}}$	κ_{L_j}	$\underline{\kappa}_{L_j}$	$\Vert\mathbf{L}_j^{-1}\Vert^2$	κ_{L_j}	$\underline{\kappa}_{L_j}$	$\Vert\mathbf{L}_j^{-1}\Vert^2$
2	3.93E+00	3.02E+00	3.02E+00	5.57E+00	3.93E+00	4.41E+00	1.01E+01
3	7.96E+00	3.61E+00	3.61E+00	7.30E+00	7.43E+00	9.11E+00	3.42E+01
4	1.60E+01	3.82E+00	3.82E+00	8.04E+00	1.03E+01	1.29E+01	6.49E+01
5	3.19E+01	3.93E+00	3.93E+00	8.45E+00	1.15E+01	1.46E+01	8.11E+01
6	6.39E+01 *	3.97E+00	3.97E+00	8.57E+00	1.18E+01	1.51E+01	8.52E+01
7	1.28E+02 *	4.02E+00	4.02E+00	8.73E+00	1.20E+01	1.53E+01	8.73E+01

Table 3.6 3D problem, $\epsilon = 10^{-6}$. Properties of $\mathbf{A}_j$ and $\mathbf{L}_j$. $^{*)}$Values of $\kappa_{A_j}^{\frac{1}{2}}$, $j = 5, 6$ are extrapolated using $\kappa_{A_j}^{\frac{1}{2}} = 2\kappa_{A_{j-1}}^{\frac{1}{2}}$.

ratio $\Vert\mathbf{A}_{J-1}^{-1}\Vert^{\frac{1}{2}}/\Vert\mathbf{A}_J^{-1}\Vert^{\frac{1}{2}}$. This allows us to make the following predictions.

We see that the requirements on $\varepsilon_J^{\mathrm{S}}$ and $\varepsilon_J^{\mathrm{R}}$ are lower than on $\dot{\varepsilon}_J$, regardless of the values of the anisotropy ϵ. Based on the values of the terms, we may be able to use single precision (unit roundoff $\approx 10^{-8}$) for solving the triangular systems when applying the smoothers, and for storing the matrices $\mathbf{L}_j$ possibly even half precision (unit roundoff $\approx 10^{-4}$).

Detailed description of the experiment

We describe the details of the experiment. We run the IR method with zero initial approximation and stop when the Euclidean norm of the relative residual is less than 10^{-8}, i.e., $\Vert\mathbf{b} - \mathbf{A}\hat{\mathbf{x}}^{(i)}\Vert/\Vert\mathbf{b}\Vert \leq 10^{-8}$. The computation is done in double precision arithmetic except for the application of the IC smoothers. We consider the following three subvariants of the method based on the use of different finite precision(s) when applying the smoothers: a subvariant *double*, where the computation is done in double precision; a subvariant *single*, where the matrix $\mathbf{L}_j$ and the right-hand side vector are rounded to single precision and the triangular solves are done in single precision, i.e., $\varepsilon^{\mathrm{S}} = \varepsilon^{\mathrm{R}}$; and a subvariant *half*, in which the matrix $\mathbf{L}_j$ and the right-hand side vector are rounded to half precision, the triangular solve routine uses single precision in the arithmetic operations, but uses half precision to store the values in global memory unless the value has been computed by a thread of the same thread block and can thus be communicated cheaply in single precision via shared memory.

We use scaling of the right-hand side vectors described in Section 3.7 when applying the smoothing. For comparison we also use IR-V-cycle with one iteration of Jacobi smoothing in double precision. The computation is done on NVIDIA A100-SXM4-80GB GPU, with CUDA version 12.1, V12.1.105, on system Guyot at the Innovative Computing Laboratory, University of Tennessee.

Results

Results of the variants in double precision are summarized in Figure 3.2. We plot the execution time, which does not involve computing the IC factorization. We see that the variant with the Jacobi smoother is the fastest for the problem with $\epsilon = 1$. For problems with higher anisotropy, the variant with ICT(fill-in=2) requires the least amount of time.

The variants which use low precision for application of the smoother converge in the same number of IR iterations as the corresponding variants in double precision,

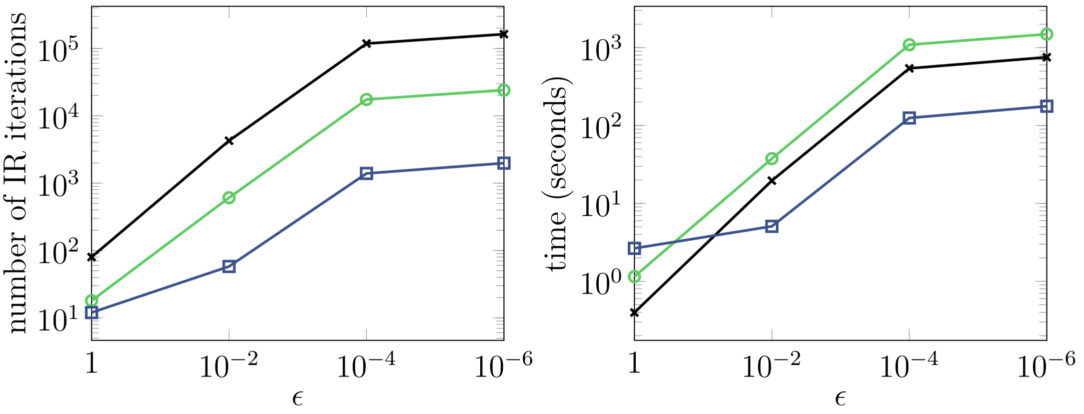


Figure 3.2 Results of Experiment 2. 3D elliptic PDEs with different values of anisotropy ϵ in the x-axis, solved by IR-V-cycle with Jacobi smoother (—×—), IC(0) smoother (—○—) and ICT(fill-in=2) smoother (—□—). The problems have $6,967,871$ DoFs. All variants are computed in double precision. The plot on the right contains the execution time of the method, which does not include the time needed for the IC factorization.

except the setting $\epsilon = 10^{-4}$ with IC(0) half, which requires one additional IR iteration. The speedups of the variants using low precision over the corresponding double precision variants are plotted in Figure 3.3. We see significant speedups when using the low precision variants for all problems. The speedups of the half precision variants are larger than the corresponding speedups of the single precision variants.

Comparing the problems with different anisotropy, the speedups are the largest for the problem with $\epsilon = 1$. For this problem the speedup of the variants with ICT(fill-in=2) is significantly larger than for the corresponding variants with IC(0). This is the opposite for the problems with higher anisotropy. We currently do not have an explanation for this behavior. We would like to investigate it further using, e.g., available profiling tools.

3.9 Conclusions

We present a mixed precision formulation of the V-cycle correction scheme with general assumptions on the finite precision errors of the coarsest-level solver and smoothers. Inspired by existing analysis, we derive a bound on the relative finite precision error of the V-cycle scheme which involves bounds on the finite precision errors of the coarsest-level solver, the smoothing routines, and error terms coming from computing the residuals, restrictions, projections, and corrections on the individual levels. Our results give insight into how the finite precision errors from the individual components of the V-cycle scheme may affect the overall finite precision error. The presented approach enables analyses of V-cycle schemes with various (mixed precision) coarsest-level solvers and smoothers. This was not possible in the previous approaches in the literature.

In this work, we focus on mixed precision smoothers based on IC factorization. We derive a bound on the finite precision error resulting from their application.

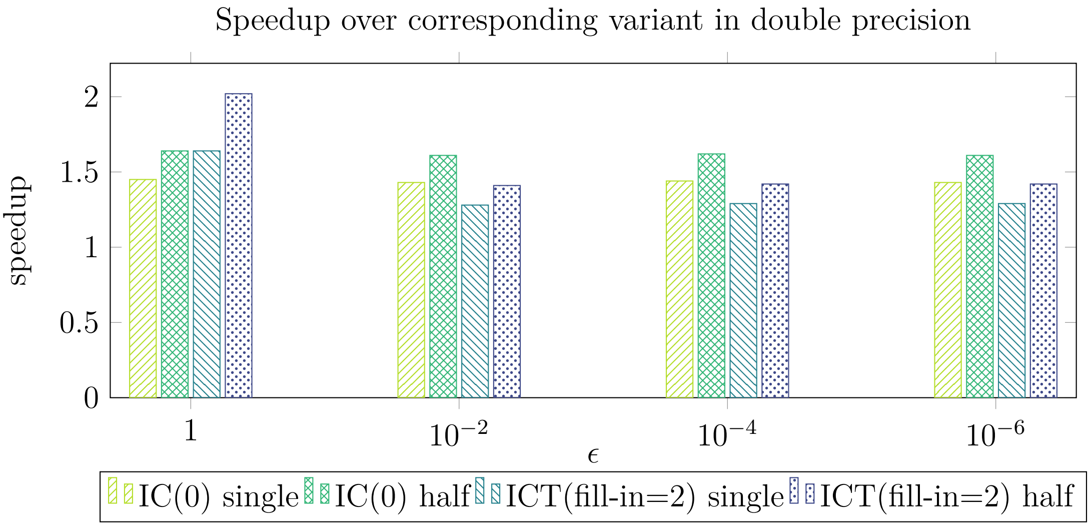


Figure 3.3 Results of Experiment 2. 3D elliptic PDEs with different values of anisotropy ϵ in the x-axis. The problems have $6,967,871$ DoFs.

We test the theoretical results and proposed methods in numerical experiments. We solve systems coming from FE discretization of elliptic PDEs. The experiments illustrate the theoretical findings and show that in the considered settings the IC smoothers can be applied in low precisions, resulting in significant speedups over their corresponding double precision variant.

Further we list several interesting open problems. For the simplicity of the analysis in this work we consider only V-cycle schemes with one iteration of pre-smoothing and no post-smoothing. Based on the related work in [17, 18] we believe that it should be possible to extend our results to cover multiple smoothing iterations and post-smoothing.

In this work, we focus on the V-cycle scheme with IC smoothing. In future work we would like to use the theoretical results to compare requirements on the finite precisions inside the V-cycle scheme when using other smoothing routines such as the Jacobi or the Gauss-Seidel method.

When deriving the bound on the finite precision error of the V-cycle scheme, we assume that the solver on the coarsest-level is linear. However, in practice, multigrid methods are also applied with iterative coarsest-level solvers, for example with CG [10] or GMRES [21] stopped with residual-based stopping criteria. It is not obvious whether the presented analysis can be generalized to cover such coarsest-level solvers and what assumptions on the error reduction or stability of the solver should be imposed.

We present numerical experiments on GPUs using the Ginkgo library which illustrate the theoretical results and show significant speedups when applying the IC smoothers in low precision. In future work, we would like to perform more numerical experiments and focus more on the performance of the methods. We would like to test the presented mixed precision methods on large-scale problems involving different anisotropy tensors or in settings with algebraic multigrid methods. It would be also interesting to study variants of the V-cycle schemes where the precisions on the individual levels vary.

In the presented experiments, we compute the IC factorizations on CPUs in double precision. Computing IC factorizations in low precision arithmetics (see, e.g., [24]), and/or using parallel versions of the IC algorithm on GPUs (e.g., [4]) are worth investigating as well.

Another series of open problems lies in the theoretical analysis of the approximation and stability properties of the IC factorization. Having more theoretical results at least for certain problem classes would be beneficial.

3.10 Appendix

3.10.1 Relations between Euclidean and A vector norms

We present derivations of the following four inequalities

$$\begin{aligned}
\|\mathbf{v}\|_{\mathbf{A}} &\leq \|\mathbf{A}\|^{\frac{1}{2}}\|\mathbf{v}\|, \\
\|\mathbf{v}\| &\leq \|\mathbf{A}^{-1}\|^{\frac{1}{2}}\|\mathbf{v}\|_{\mathbf{A}}, \\
\|\mathbf{A}\mathbf{v}\| &\leq \|\mathbf{A}\|^{\frac{1}{2}}\|\mathbf{v}\|_{\mathbf{A}}, \\
\|\mathbf{A}^{-1}\mathbf{v}\|_{\mathbf{A}} &\leq \|\mathbf{A}^{-1}\|^{\frac{1}{2}}\|\mathbf{v}\|.
\end{aligned}$$

Derivation of the first inequality

$$\|\mathbf{v}\|_{\mathbf{A}}^2 = \langle \mathbf{A}\mathbf{v}, \mathbf{v}\rangle \leq \|\mathbf{A}\|\langle \mathbf{v}, \mathbf{v}\rangle = \|\mathbf{A}\|\|\mathbf{v}\|^2.$$

Derivation of the second inequality

$$\begin{aligned}
\|\mathbf{v}\|^2 = \langle \mathbf{v}, \mathbf{v}\rangle &= \langle \mathbf{A}^{-\frac{1}{2}}\mathbf{A}^{\frac{1}{2}}\mathbf{v}, \mathbf{A}^{-\frac{1}{2}}\mathbf{A}^{\frac{1}{2}}\mathbf{v}\rangle = \langle \mathbf{A}^{-\frac{1}{2}}\mathbf{A}^{-\frac{1}{2}}\mathbf{A}^{\frac{1}{2}}\mathbf{v}, \mathbf{A}^{\frac{1}{2}}\mathbf{v}\rangle \\
&= \langle \mathbf{A}^{-1}\mathbf{A}^{\frac{1}{2}}\mathbf{v}, \mathbf{A}^{\frac{1}{2}}\mathbf{v}\rangle \leq \|\mathbf{A}^{-1}\|\langle \mathbf{A}^{\frac{1}{2}}\mathbf{v}, \mathbf{A}^{\frac{1}{2}}\mathbf{v}\rangle = \|\mathbf{A}^{-1}\|\langle \mathbf{A}\mathbf{v}, \mathbf{v}\rangle = \|\mathbf{A}^{-1}\|\|\mathbf{v}\|_{\mathbf{A}}^2.
\end{aligned}$$

Derivation of the third inequality

$$\begin{aligned}
\|\mathbf{A}\mathbf{v}\|^2 = \langle \mathbf{A}\mathbf{v}, \mathbf{A}\mathbf{v}\rangle &= \langle \mathbf{A}^{\frac{1}{2}}\mathbf{A}^{\frac{1}{2}}\mathbf{v}, \mathbf{A}^{\frac{1}{2}}\mathbf{A}^{\frac{1}{2}}\mathbf{v}\rangle = \langle \mathbf{A}^{\frac{1}{2}}\mathbf{A}^{\frac{1}{2}}\mathbf{A}^{\frac{1}{2}}\mathbf{v}, \mathbf{A}^{\frac{1}{2}}\mathbf{v}\rangle = \langle \mathbf{A}\mathbf{A}^{\frac{1}{2}}\mathbf{v}, \mathbf{A}^{\frac{1}{2}}\mathbf{v}\rangle \\
&\leq \|\mathbf{A}\|\langle \mathbf{A}^{\frac{1}{2}}\mathbf{v}, \mathbf{A}^{\frac{1}{2}}\mathbf{v}\rangle = \|\mathbf{A}\|\langle \mathbf{A}\mathbf{v}, \mathbf{v}\rangle = \|\mathbf{A}\|\|\mathbf{v}\|_{\mathbf{A}}^2.
\end{aligned}$$

Derivation of the forth inequality

$$\|\mathbf{A}^{-1}\mathbf{v}\|_{\mathbf{A}} = \langle \mathbf{A}\mathbf{A}^{-1}\mathbf{v}, \mathbf{A}^{-1}\mathbf{v}\rangle = \langle \mathbf{v}, \mathbf{A}^{-1}\mathbf{v}\rangle \leq \|\mathbf{A}^{-1}\|\langle \mathbf{v}, \mathbf{v}\rangle = \|\mathbf{A}^{-1}\|\|\mathbf{v}\|^2.$$

3.10.2 Derivation of bounds on finite precision errors of certain basic routines

Derivation of (3.5): Rounding $\mathbf{K}$ to ε-precision, results in $\mathbf{K} + \Delta\mathbf{K}$, $|\Delta\mathbf{K}| \leq \varepsilon|\mathbf{K}|$. Computing $(\mathbf{K} + \Delta\mathbf{K})\mathbf{w}$ in ε-precision results in $(\mathbf{K} + \Delta\mathbf{K})\mathbf{w} + \delta_2$, where

$$\begin{aligned}
\|\delta_2\| &\leq \varepsilon m_{K,\varepsilon}(1+\varepsilon)\||\mathbf{K}|\|\|\mathbf{w}\| \\
&\leq \varepsilon(m_{K,\varepsilon} + \varepsilon m_{K,\varepsilon})\||\mathbf{K}|\|\|\mathbf{w}\|.
\end{aligned}$$

Computing $\mathbf{K}\mathbf{w}$ in ε-precision results in $\mathbf{K}\mathbf{w} + \delta_1$, where $\delta_1 = \Delta\mathbf{K}\mathbf{w} + \delta_2$ is the accumulated error and

$$\|\delta_1\| = \|\Delta\mathbf{K}\|\|\mathbf{w}\| + \|\delta_2\| \leq \varepsilon\||\mathbf{K}|\|\|\mathbf{w}\| + \|\delta_2\| \leq \varepsilon(m_{K,\varepsilon} + 1 + \varepsilon m_{K,\varepsilon})\||\mathbf{K}|\|\|\mathbf{w}\|.$$

Derivation of (3.6): We may use the previous result and add the error occurring due to the subsequent addition. Computing $\mathbf{v} + (\mathbf{K}\mathbf{w} + \delta_1)$ in ε-precision results in $\mathbf{v} + \mathbf{K}\mathbf{w} + \delta_4$, where (using the previous)

$$\begin{aligned}\|\delta_4\| &\leq \varepsilon(\|\mathbf{v}\| + \|\mathbf{K}\mathbf{w}\| + \|\delta_1\|)\\ &\leq \varepsilon(\|\mathbf{v}\| + \||\mathbf{K}|\|\|\mathbf{w}\| + \|\delta_1\|)\\ &\leq \varepsilon(\|\mathbf{v}\| + (1 + \varepsilon(m_{K,\varepsilon} + 1 + \varepsilon m_{K,\varepsilon}))\||\mathbf{K}|\|\|\mathbf{w}\|)\\ &\leq \varepsilon(1 + \varepsilon(m_{K,\varepsilon} + 1 + \varepsilon m_{K,\varepsilon}))(\|\mathbf{v}\| + \||\mathbf{K}|\|\|\mathbf{w}\|).\end{aligned}$$

Accumulating the errors in $\delta_3 = \delta_1 + \delta_4$ we have

$$\|\delta_3\| \leq \varepsilon(m_{K,\varepsilon} + 2 + \varepsilon(2m_{K,\varepsilon} + 1 + \varepsilon m_{K,\varepsilon}))(\|\mathbf{v}\| + \||\mathbf{K}|\|\|\mathbf{w}\|).$$

3.10.3 Derivation of multigrid related bounds

Derivation of (3.15): We rewrite $\|\mathbf{A}_\mathrm{C}^{-1}\mathbf{P}^\top\mathbf{v}\|^2_{\mathbf{A}_\mathrm{C}}$ as

$$\begin{aligned}\|\mathbf{A}_\mathrm{C}^{-1}\mathbf{P}^\top\mathbf{v}\|^2_{\mathbf{A}_\mathrm{C}} &= \langle\mathbf{A}_\mathrm{C}\mathbf{A}_\mathrm{C}^{-1}\mathbf{P}^\top\mathbf{v}, \mathbf{A}_\mathrm{C}^{-1}\mathbf{P}^\top\mathbf{v}\rangle\\ &= \langle\mathbf{P}^\top\mathbf{v}, \mathbf{A}_\mathrm{C}^{-1}\mathbf{P}^\top\mathbf{v}\rangle\\ &= \langle\mathbf{v}, \mathbf{P}\mathbf{A}_\mathrm{C}^{-1}\mathbf{P}^\top\mathbf{v}\rangle\\ &= \langle\mathbf{A}^{\frac{1}{2}}\mathbf{A}^{-\frac{1}{2}}\mathbf{v}, \mathbf{P}\mathbf{A}_\mathrm{C}^{-1}\mathbf{P}^\top\mathbf{A}^{\frac{1}{2}}\mathbf{A}^{-\frac{1}{2}}\mathbf{v}\rangle\\ &= \langle\mathbf{A}^{-\frac{1}{2}}\mathbf{v}, \mathbf{A}^{\frac{1}{2}}\mathbf{P}\mathbf{A}_\mathrm{C}^{-1}\mathbf{P}^\top\mathbf{A}^{\frac{1}{2}}\mathbf{A}^{-\frac{1}{2}}\mathbf{v}\rangle.\end{aligned} \tag{3.54}$$

Since

$$\begin{aligned}\mathbf{A}^{\frac{1}{2}}\mathbf{P}\mathbf{A}_\mathrm{C}^{-1}\mathbf{P}^\top\mathbf{A}^{\frac{1}{2}} &= \mathbf{A}^{\frac{1}{2}}\mathbf{P}(\mathbf{P}^\top\mathbf{A}\mathbf{P})^{-1}\mathbf{P}^\top\mathbf{A}^{\frac{1}{2}}\\ &= \mathbf{A}^{\frac{1}{2}}\mathbf{P}((\mathbf{A}^{\frac{1}{2}}\mathbf{P})^\top\mathbf{A}^{\frac{1}{2}}\mathbf{P})^{-1}(\mathbf{A}^{\frac{1}{2}}\mathbf{P})^\top\end{aligned}$$

is the orthogonal projection onto the range of $\mathbf{A}^{\frac{1}{2}}\mathbf{P}$, there holds

$$\|\mathbf{A}^{\frac{1}{2}}\mathbf{P}\mathbf{A}_\mathrm{C}^{-1}\mathbf{P}^\top\mathbf{A}^{\frac{1}{2}}\| \leq 1.$$

Combining this and (3.54) leads to

$$\begin{aligned}\|\mathbf{A}_\mathrm{C}^{-1}\mathbf{P}^\top\mathbf{v}\|^2_{\mathbf{A}_\mathrm{C}} &\leq \langle\mathbf{A}^{-\frac{1}{2}}\mathbf{v}, \mathbf{A}^{-\frac{1}{2}}\mathbf{v}\rangle\\ &= \langle\mathbf{v}, \mathbf{A}^{-1}\mathbf{v}\rangle = \langle\mathbf{A}\mathbf{A}^{-1}\mathbf{v}, \mathbf{A}^{-1}\mathbf{v}\rangle\\ &= \|\mathbf{A}^{-1}\mathbf{v}\|^2_{\mathbf{A}}.\end{aligned}$$

Derivation of (3.16). Using assumption (3.10) yields

$$\|\mathbf{M}_\mathrm{C}\mathbf{A}_\mathrm{C}\|_{\mathbf{A}_\mathrm{C}} \leq \|\mathbf{I}_\mathrm{C}\|_{\mathbf{A}_\mathrm{C}} + \|\mathbf{I}_\mathrm{C} - \mathbf{M}_\mathrm{C}\mathbf{A}_\mathrm{C}\|_{\mathbf{A}_\mathrm{C}} < 2.$$

Derivation of (3.17): Using the assumption (3.12) yields

$$\|\mathbf{v}^{[3]}\|_{\mathbf{A}} = \|\mathbf{y}_\mathrm{TG}\|_{\mathbf{A}} \leq \|\mathbf{y}_\mathrm{TG} - \mathbf{y}\|_{\mathbf{A}} + \|\mathbf{y}\|_{\mathbf{A}} \leq 2\|\mathbf{y}\|_{\mathbf{A}}.$$

Derivation of (3.18): Using $\mathbf{A}\mathbf{y} = \mathbf{f}$, $\mathbf{r}^{[1]} = \mathbf{f} - \mathbf{A}\mathbf{M}\mathbf{f}$, and the assumption (3.7) results in

$$\begin{aligned}\|\mathbf{A}^{-1}\mathbf{r}^{[1]}\|_{\mathbf{A}} &= \|\mathbf{A}^{-1}(\mathbf{f} - \mathbf{A}\mathbf{M}\mathbf{f})\|_{\mathbf{A}} = \|\mathbf{y} - \mathbf{M}\mathbf{A}\mathbf{y}\|_{\mathbf{A}}\\ &\leq \|\mathbf{I} - \mathbf{M}\mathbf{A}\|_{\mathbf{A}}\|\mathbf{y}\|_{\mathbf{A}} \leq \|\mathbf{y}\|_{\mathbf{A}}.\end{aligned}$$

Derivation of (3.19): Using (3.18) yields

$$\|\mathbf{r}^{[1]}\|_{\mathbf{A}} \leq \|\mathbf{A}\|^{\frac{1}{2}}\|\mathbf{A}^{-1}\mathbf{r}^{[1]}\|_{\mathbf{A}} \leq \|\mathbf{A}\|^{\frac{1}{2}}\|\mathbf{y}\|_{\mathbf{A}}.$$

Derivation of (3.20): Using $\mathbf{r}_{\mathrm{C}}^{[1]} = \mathbf{P}^{\top}\mathbf{r}^{[1]}$, (3.15) and (3.18) results in

$$\|\mathbf{A}_{\mathrm{C}}^{-1}\mathbf{r}_{\mathrm{C}}^{[1]}\|_{\mathbf{A}_{\mathrm{C}}} = \|\mathbf{A}_{\mathrm{C}}^{-1}\mathbf{P}^{\top}\mathbf{r}^{[1]}\|_{\mathbf{A}_{\mathrm{C}}} \leq \|\mathbf{A}^{-1}\mathbf{r}^{[1]}\|_{\mathbf{A}} \leq \|\mathbf{y}\|_{\mathbf{A}}.$$

Derivation of (3.21): Using the expressions $\mathbf{v}_{\mathrm{C}}^{[2]} = \mathbf{M}_{\mathrm{C}}\mathbf{r}_{\mathrm{C}}^{[1]}$ and bound (3.16) and (3.20) results in

$$\|\mathbf{v}_{\mathrm{C}}^{[2]}\|_{\mathbf{A}_{\mathrm{C}}} = \|\mathbf{M}_{\mathrm{C}}\mathbf{A}_{\mathrm{C}}\mathbf{A}_{\mathrm{C}}^{-1}\mathbf{r}_{\mathrm{C}}^{[1]}\|_{\mathbf{A}_{\mathrm{C}}} \leq \|\mathbf{M}_{\mathrm{C}}\mathbf{A}_{\mathrm{C}}\|_{\mathbf{A}_{\mathrm{C}}}\|\mathbf{A}_{\mathrm{C}}^{-1}\mathbf{r}_{\mathrm{C}}^{[1]}\|_{\mathbf{A}_{\mathrm{C}}} \leq 2\|\mathbf{y}\|_{\mathbf{A}}.$$

3.10.4 Proof of Lemma 3.1

We present a proof for a lower-triangular matrix $\mathbf{T}$. Proof for an upper-triangular matrix $\mathbf{T}$ is analogous. The proof is based on using the following lemma.

Lemma 3.3. *[11, Lemma 8.4] Let k be a natural number an let δ, α_i, $i = 1,\ldots,k-1$, β_i, $i = 1,\ldots,k$ be real numbers belonging to a finite precision arithmetic with unit roundoff ε^{S}. Computing*

$$\gamma = (\delta - \sum_{i=1}^{k-1}\alpha_i\beta_i)/\beta_k$$

in ε^{S}-precision results in $\hat{\gamma}$ satisfying, no matter the order of evaluation,

$$\beta_k(1+\theta_k^{(0)})\hat{\gamma} = \delta - \sum_{i=1}^{k-1}\alpha_i\beta_i(1+\theta_k^{(i)}),$$

where $|\theta_k^{(i)}| \leq k\varepsilon^{\mathrm{S}}/(1-k\varepsilon^{\mathrm{S}})$, $i = 0,1,\ldots,k$.

We use induction on size of the leading sub-matrices. Let $T_{i,j}$ and $E_{i,j}$ denote the entries of matrices $\mathbf{T}$ and $\mathbf{E}$, respectively, in the ith row and jth column, and let b_i denote the ith entry of the right-hand side vector $\mathbf{b}$.

We start by showing that the statement holds for the leading sub-matrix of size 1×1. Using Lemma 3.3 for $k = 1$, computing $x_1 = b_1/T_{1,1}$ in ε^{S}-precision results in $\hat{x}_1$ satisfying $T_{1,1}(1+\theta_1^{(0)})\hat{x}_1 = b_1$, where $|\theta_1^{(0)}| \leq \varepsilon^{\mathrm{S}}/(1-\varepsilon^{\mathrm{S}})$. We can take $E_{1,1} = T_{1,1}\theta_1^{(0)}$.

Assume that the statement holds for the leading sub-matrix of size $n \times n$. We will show that it holds also for the leading sub-matrix of size $(n+1) \times (n+1)$. The induction assumption and the fact that $\mathbf{T}$ is a lower triangular matrix yields that it only remains to show existence of suitable entries in the $(n+1)$th row of $\mathbf{E}$. Let $\hat{x}_i$, $i = 1,\ldots,n$ denote the computed entries of $\hat{\mathbf{x}}$ after n steps of the substitution. The $n+1$ substitution step consists of computing

$$x_{n+1} = (b_{n+1} - \sum_{i=1}^{n}\hat{x}_iT_{n+1,i})/T_{n+1,n+1}.$$

Since we assume there is maximum m_T nonzero elements in a row of $\mathbf{T}$, the sum consist of maximum $m_T - 1$ nonzero terms. The equation can be rewritten as

$$x_{n+1} = (b_{n+1} - \sum_{\ell; T_{n+1,\ell} \neq 0} \hat{x}_\ell T_{n+1,\ell}) / T_{n+1,n+1}.$$

Using Lemma 3.3 in this setting yields

$$T_{n+1,n+1}(1 + \theta_{m_T}^{(0)})\hat{x}_{n+1} = b_{n+1} - \sum_{\ell; T_{n+1,\ell} \neq 0} \hat{x}_\ell T_{n+1,\ell}(1 + \theta_{m_T}^{(\ell)}),$$

where $|\theta_{m_T}^{(i)}| \leq (m_T \varepsilon^{\mathrm{S}})/(1 - m_T \varepsilon^{\mathrm{S}})$, $i = 0, \ldots, m_T - 1$. Taking $E_{n+1,\ell} = T_{n+1,\ell}\theta_{m_T}^{(\ell)}$, for ℓ such that $T_{n+1,\ell} \neq 0$ and $E_{n+1,n+1} = T_{n+1,n+1}\theta_{m_T}^{(0)}$ yields that the statement hold also for the $(n+1) \times (n+1)$ leading sub-matrix.

Acknowledgments

We acknowledge the support of the Innovative Computing Laboratory, University of Tennessee, which provided the computational resources for running the expermiments on GPUs. P. Vacek acknowledges the support of the Erasmus+ program that enabled him to spend October 2023 - March 2024 at Friedrich-Alexander-Universität Erlangen-Nürnberg. P. Vacek is also grateful for the support from the Technical University of Munich for his visit in Heilbronn in April 2024. During these visits the basis of the paper was developed.

Bibliography

[1] A. Abdelfattah et al. "A survey of numerical linear algebra methods utilizing mixed-precision arithmetic". In: *The International Journal of High Performance Computing Applications* 35.4 (2021), pp. 344–369. DOI: 10.1177/10943420211003313.

[2] *Advanpix Multiprecision Computing Toolbox for MATLAB ver. 5.1.0.15432*. Yokohama, Japan: Advanpix LLC. URL: https://www.advanpix.com/.

[3] M. S. Alnaes, J. Blechta, J. Hake, et al. "The FEniCS Project Version 1.5". In: *Archive of Numerical Software* 3 (2015). DOI: 10.11588/ans.2015.100.20553.

[4] H. Anzt, E. Chow, and J. Dongarra. "ParILUT—A New Parallel Threshold ILU Factorization". In: *SIAM Journal on Scientific Computing* 40.4 (2018), pp. C503–C519. DOI: 10.1137/16M1079506.

[5] H. Anzt et al. "Ginkgo: A Modern Linear Operator Algebra Framework for High Performance Computing". In: *ACM Transactions on Mathematical Software* 48.1 (2022), 2:1–2:33. DOI: 10.1145/3480935.

[6] W. L. Briggs, V. E. Henson, and S. F. McCormick. *A Multigrid Tutorial*. Second. Philadelphia, PA: SIAM, 2000, pp. xii+193. DOI: 10.1137/1.9780898719505.

[7] E. Carson and N. J. Higham. “Accelerating the Solution of Linear Systems by Iterative Refinement in Three Precisions”. In: *SIAM Journal on Scientific Computing* 40.2 (2018), A817–A847. DOI: 10.1137/17M1140819.

[8] T. Cojean et al. “Ginkgo - A math library designed to accelerate Exascale Computing Project science applications”. In: *The International Journal of High Performance Computing Applications* (2024). DOI: 10.1177/10943420241268323.

[9] D. Drzisga, A. Wagner, and B. Wohlmuth. “A Matrix-Free ILU Realization Based on Surrogates”. In: *SIAM Journal on Scientific Computing* 45.6 (2023), pp. C304–C329. DOI: 10.1137/22M1529415.

[10] M. R. Hestenes and E. Stiefel. “Methods of conjugate gradients for solving linear systems”. In: *Journal of Research of the National Bureau of Standards* 49.6 (1952), pp. 409–436. DOI: 10.6028/jres.049.044.

[11] N. J. Higham. *Accuracy and Stability of Numerical Algorithms*. Second. Society for Industrial and Applied Mathematics, 2002. DOI: 10.1137/1.9780898718027.

[12] N. J. Higham and T. Mary. “Mixed precision algorithms in numerical linear algebra”. In: *Acta Numerica* 31 (2022), pp. 347–414. DOI: 10.1017/S0962492922000022.

[13] N. J. Higham, S. Pranesh, and M. Zounon. “Squeezing a Matrix into Half Precision, with an Application to Solving Linear Systems”. In: *SIAM Journal on Scientific Computing* 41.4 (2019), A2536–A2551. DOI: 10.1137/18M1229511.

[14] R. Kettler and P. Wesseling. “Aspects of multigrid methods for problems in three dimensions”. In: *Applied Mathematics and Computation* 19.1 (1986), pp. 159–168. DOI: 10.1016/0096-3003(86)90102-5.

[15] R. Kettler. “Analysis and comparison of relaxation schemes in robust multigrid and preconditioned conjugate gradient methods”. In: *Multigrid Methods*. Ed. by W. Hackbusch and U. Trottenberg. Berlin, Heidelberg: Springer Berlin Heidelberg, 1982, pp. 502–534. DOI: 10.1007/BFb0069941.

[16] N. Kohl, S. F. McCormick, and R. Tamstorf. “Multigrid Methods Using Block Floating Point Arithmetic”. In: *SIAM Journal on Scientific Computing* (2024), S202–S224. DOI: 10.1137/23M1581819.

[17] S. F. McCormick, J. Benzaken, and R. Tamstorf. “Algebraic Error Analysis for Mixed-Precision Multigrid Solvers”. In: *SIAM Journal on Scientific Computing* 43.5 (2021), S392–S419. DOI: 10.1137/20M1348571.

[18] S. F. McCormick and R. Tamstorf. “Rounding-Error Analysis of Multigrid V-Cycles”. In: *SIAM Journal on Scientific Computing* (2024), S88–S95. DOI: 10.1137/23M1582898.

[19] Y. Notay. “Algebraic Theory of Two-Grid Methods”. In: *Numerical Mathematics: Theory, Methods and Applications* 8.2 (2015), pp. 168–198. DOI: 10.4208/nmtma.2015.w04si.

[20] Y. Notay. “Convergence analysis of perturbed two-grid and multigrid methods”. In: *SIAM Journal on Numerical Analysis* 45.3 (2007), pp. 1035–1044. DOI: 10.1137/060652312.

[21] C. C. Paige and M. A. Saunders. “Solution of Sparse Indefinite Systems of Linear Equations”. In: *SIAM Journal on Numerical Analysis* 12.4 (1975), pp. 617–629. DOI: 10.1137/0712047.

[22] Y. Saad. *Iterative Methods for Sparse Linear Systems*. Second. Society for Industrial and Applied Mathematics, 2003. DOI: 10.1137/1.9780898718003.

[23] J. Scott and M. Tůma. *Algorithms for sparse linear systems*. Springer Nature, 2023. DOI: 10.1007/978-3-031-25820-6.

[24] J. Scott and M. Tůma. “Avoiding Breakdown in Incomplete Factorizations in Low Precision Arithmetic”. In: *ACM Trans. Math. Softw.* 50.2 (2024). DOI: 10.1145/3651155.

[25] R. Tamstorf, J. Benzaken, and S. F. McCormick. “Discretization -Error-Accurate Mixed-Precision Multigrid Solvers”. In: *SIAM Journal on Scientific Computing* 43.5 (2021), S420–S447. DOI: 10.1137/20M1349230.

[26] S. Thomas et al. “Scaled ILU smoothers for Navier–Stokes pressure projection”. In: *International Journal for Numerical Methods in Fluids* 96.4 (2024), pp. 537–560. DOI: 10.1002/fld.5254.

[27] U. Trottenberg, C. W. Oosterlee, and A. Schuller. *Multigrid*. London: Academic Press, 2001.

[28] Y.-H. Tsai. “Portable Mixed Precision Algebraic Multigrid on High Performance GPUs”. PhD thesis. Karlsruher Institut für Technologie (KIT), 2024. 116 pp. DOI: 10.5445/IR/1000168914.

[29] Y.-H. M. Tsai, N. Beams, and H. Anzt. “Mixed Precision Algebraic Multigrid on GPUs”. In: *Parallel Processing and Applied Mathematics*. Ed. by R. Wyrzykowski et al. Cham: Springer International Publishing, 2023, pp. 113–125. DOI: 10.1007/978-3-031-30442-2_9.

[30] Y.-H. M. Tsai, N. Beams, and H. Anzt. “Three-precision algebraic multigrid on GPUs”. In: *Future Generation Computer Systems* 149 (2023), pp. 280–293. DOI: 10.1016/j.future.2023.07.024.

[31] P. Wesseling. “A robust and efficient multigrid method”. In: *Multigrid Methods*. Ed. by W. Hackbusch and U. Trottenberg. Berlin, Heidelberg: Springer Berlin Heidelberg, 1982, pp. 614–630. DOI: 10.1007/BFb0069947.

[32] P. Wesseling. “Theoretical and Practical Aspects of a Multigrid Method”. In: *SIAM Journal on Scientific and Statistical Computing* 3.4 (1982), pp. 387–407. DOI: 10.1137/0903025.

[33] J. Xu. “Iterative methods by space decomposition and subspace correction”. In: *SIAM Review* 34.4 (1992), pp. 581–613. DOI: 10.1137/1034116.

[34] H. Yserentant. “Old and new convergence proofs for multigrid methods”. In: *Acta Numerica* 2 (1993), pp. 285–326.

[35] Y. Zong et al. “FP16 Acceleration in Structured Multigrid Preconditioner for Real-World Applications”. In: *Proceedings of the 53rd International Conference on Parallel Processing*. ICPP ’24. Gotland, Sweden: Association for Computing Machinery, 2024, pp. 52–62. DOI: 10.1145/3673038.3673040.

Conclusion

In this thesis, we study multigrid methods and multilevel a posteriori error estimates. The work is motivated by computational challenges arising when solving large-scale problems. We demonstrate that certain parts of the computation can be done approximately, either using an approximate coarsest-level solver, or using low precision arithmetic, while preserving the convergence of the methods or properties of a posteriori error estimates. The approximate techniques have to be, however, used with caution and should be based on theoretical analysis.

Further, we restate the research questions formulated in the introduction and summarize the obtained results:

a) Can we analytically describe how the accuracy of the coarsest-level solver affects the convergence behavior of the multigrid method?

b) Can we design effective stopping criteria for an iterative coarsest-level solver such that the multigrid method converges in nearly the same number of iterations as its variant with an exact coarsest-level solver?

We focus on these questions in Chapter 1. We present a novel approach to analyzing the effects of approximate coarsest-level solves on the convergence of the V-cycle method for symmetric positive definite problems. The approach is used to derive new coarsest-level stopping criterion with the required properties in question b). The theoretical results can also be used to obtain insights into how the convergence of the V-cycle method may be affected by the choice of tolerance in the coarsest-level stopping criterion based on the Euclidean norm of the relative residual.

c) Consider the residual-based multilevel a posteriori error estimates such as in [1, Section 2.6]. Is it possible to compute the term associated with the coarsest-level approximately while preserving the efficiency and accuracy of the estimate?

We address this question in Chapter 2. We show that this is possible by proposing a new approximation of the coarsest-level term, which relies on using the conjugate gradient method with an appropriate stopping criterion. We provide theoretical analysis showing that the resulting estimates have the desired properties.

d) Can the execution time of the mixed precision V-cycle method with IC smoothers be reduced by introducing additional precisions for the applications of the smoothers? For example, using different precisions for storing the IC factors or solving the triangular systems. Can we analytically describe the requirements on these individual precisions?

We focus on this question in Chapter 3. We formulate a mixed precision V-cycle scheme with general smoothers (not necessarily based on IC factorization). Our approach is based on imposing assumptions on the finite precision error resulting from the application of a smoother or the coarsest-level solver, rather than assuming that it is applied in a precision with a certain unit roundoff. This enables the analysis of mixed precision smoothers and coarsest-level solvers. We derive a bound on the finite precision error of the V-cycle scheme, which gives insight into how the finite precision errors from the individual parts of the V-cycle scheme may affect the overall finite precision error. Further, we focus on the IC smoother, we present its mixed precision formulation, and derive a bound on the finite precision error of its application. The theoretical results can be used to describe the requirements on the individual finite precisions in concrete settings. Numerical experiments on GPUs using the Ginkgo library show a significant speedup when applying IC smoothers in low precisions.

We believe that the presented results can be beneficial when designing and implementing multigrid solvers for large-scale problems.

We list several open problems, which we would like to investigate in the future. In the first chapter we focus on multigrid methods as a standalone solvers. In practice, multigrid methods are also frequently used as preconditioners for Krylov subspace methods. It is therefore an interesting question of how the approximate coarsest-level solve affects the behavior of multigrid methods as preconditioners. We note that in general a multigrid method with an approximate coarsest-level solver would have to be applied as a flexible preconditioner.

In the second chapter we assume that the computation of the terms in the a posteriori error estimate is done in infinite precision arithmetic. It would be useful to understand the effects of the finite precision errors on the accuracy and efficiency of the estimate. Would it make sense to compute the terms associated with different levels in different precisions?

The derivation of the bound on the finite precision error of the V-cycle scheme in the third chapter was done assuming that the coarsest-level solver is linear. As we discuss in the first chapter, this assumption is not satisfied for all solvers which are used in practice. It is an interesting open question whether the presented analysis can be generalized to cover general coarsest-level solvers and what assumptions on the error reduction or stability of the solver should be imposed.

Bibliography

[1] U. Rüde. *Mathematical and computational techniques for multilevel adaptive methods.* Philadelphia, PA: SIAM, 1993.