

Research Article

Minimization of Functional Majorant in a Posteriori Error Analysis Based on $H(\text{div})$ Multigrid-Preconditioned CG Method

Jan Valdman

Department of Mathematics, University of Bergen, Johannes Bruns Gate 12, 5008 Bergen, Norway

Correspondence should be addressed to Jan Valdman, jan.valdman@gmail.com

Received 15 October 2008; Revised 5 May 2009; Accepted 8 June 2009

Recommended by Serge Nicaise

We consider a Poisson boundary value problem and its functional a posteriori error estimate derived by S. Repin in 1999. The estimate majorizes the H^1 seminorm of the error of the discrete solution computed by FEM method and contains a free u_x variable from the $H(\text{div})$ space. In order to keep the estimate sharp, a procedure for the minimization of the majorant term with respect to the u_x variable is introduced, computing the free u_x variable from a global linear system of equations. Since the linear system is symmetric and positive definite, few iterations of a conjugate gradient method with a geometrical multigrid preconditioner are applied. Numerical techniques are demonstrated on one benchmark example with a smooth solution on a unit square domain including the computation of the approximate value of the constant in Friedrichs' inequality.

Copyright © 2009 Jan Valdman. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

1. Introduction

A priori rate convergence estimates for finite element approximations of elliptic boundary value problems have been investigated in the 70s–80s (see, e.g., [1]). However, adaptive multilevel algorithms require a posteriori estimates able to (a) provide a reliable and directly computable estimate of the approximation error, and (b) efficient error indicator that detects the regions with excessively high errors.

In the recent decades, a posteriori error estimates for linear elliptic and parabolic problems were intensively investigated. A reader will find a systematic exposition of the main approaches to a posteriori error estimation of finite element approximations (such as residual or gradient averaging methods) in the monographs [2–5] and papers [6–8] and in literature cited therein.

In this paper, a posteriori estimates that majorate the difference between exact solution of a linear elliptic equation and any function in the admissible (energy) class are studied.

For the class of uniformly convex variational problems computable error majorants (for any conforming approximation) were derived by the variational techniques in the mid 90s using duality theory of the calculus of variations. Key publications related to this subject are in [9–11]. Another “nonvariational” method was introduced in [12]. In this paper it was stated that for linear elliptic problems both methods lead to the same error majorants. Later it was applied to many problems, including parabolic equations and nonlinear problems [13–17].

As an example of demonstration, let us consider a scalar boundary value (Poisson’s) problem

$$\Delta u + f = 0 \quad \text{in } \Omega, \quad u = 0 \quad \text{on } \partial\Omega \quad (1.1)$$

for a searched function u from a Sobolev space $\mathring{H}^1(\Omega)$. The right hand side $f \in L^2(\Omega)$ and an open bounded domain $\Omega \in \mathbb{R}^d$, where d denotes a domain dimension, that is, $d = 1, 2, 3$ are given. Assume that $v \in \mathring{H}^1(\Omega)$ is an approximation of u . Then, a functional error estimate from [9]

$$\|\nabla(u - v)\|_{\Omega} \leq \|\nabla v - y\|_{\Omega} + C_{\Omega} \|\operatorname{div} y + f\|_{\Omega} \quad (1.2)$$

holds for all functions $y \in H(\Omega, \operatorname{div})$, $\|\cdot\|_{\Omega}$ denotes the $L^2(\Omega)$ norm. Note that no mesh-dependent constants or any assumptions on regularity of an exact solution are contained in this estimate. The only global constant C_{Ω} included represents a constant from the Friedrichs’ inequality

$$\|w\|_{\Omega} \leq C_{\Omega} \|\nabla w\|_{\Omega} \quad (1.3)$$

which holds for all $w \in \mathring{H}^1(\Omega)$. Thus the constant C_{Ω} depends on the domain Ω only and can be precomputed (it is demonstrated in Subsection 3.1). For a given $y \in H(\Omega, \operatorname{div})$, the quality of the estimate (1.2) is measured by a ratio of its right hand and left hand side

$$I_{\text{eff}} := \frac{\|\nabla v - y\|_{\Omega} + C_{\Omega} \|\operatorname{div} y + f\|_{\Omega}}{\|\nabla(u - v)\|_{\Omega}} \quad (1.4)$$

known as an index of efficiency. It obviously holds $I_{\text{eff}} \geq 1$ and the equality $I_{\text{eff}} = 1$ is valid for the choice $y = \nabla u$, that is, if y is chosen as the flux of an exact solution u . Having this interpretation of y in mind, there are known ways [18] how to compute a reasonable flux approximation y from the discrete solution v .

- (1) Averaging on the mesh of the discrete solution v . In this case, the flux approximation is computed as $y = \mathcal{G}v$, where \mathcal{G} represents an averaging gradient operator, see, for example, [8] for more details. This is a cheap method providing some preliminary knowledge on the upper bound of the error.
- (2) Averaging on a refined mesh. It is similar to (1), only with the difference that the averaging is done for the solution v calculated on once more (or more times) refined

mesh. This method can be regarded as a quantitative form of the Runge's rule. It is more expensive, but provides (generally not always) sharper results.

- (3) Using partially equilibrated fluxes. By postprocessing of v , a function $y_{\bar{f}}$ is constructed such that $\operatorname{div} y_{\bar{f}} + \bar{f} = 0$ and \bar{f} are sufficiently close to f in $L^2(\Omega)$ norm. Then, the substitution of $y = y_{\bar{f}} + \tau$ into (1.2) provides an estimate

$$\|\nabla(u - v)\|_{\Omega} \leq \|\nabla v + \operatorname{curl} \tau - y_{\bar{f}}\|_{\Omega} + C_{\Omega} \|f - \bar{f}\|_{\Omega}, \quad (1.5)$$

where $\tau \in H(\Omega, \operatorname{curl})$ is arbitrary.

- (4) Minimization of the right hand side of (1.2) with respect to free variable y on the mesh of the discrete solution v . This is the most expensive method for a detailed knowledge of the error.

A comparison of methods (1), (2), and (4) for a class of problems with nonlinear boundary conditions can be found, for example, in [17]. We should mention that there are many advanced forms of error bounds for the Poisson's equation (1.1). They are based, for example, on decomposition of the domain Ω or on partial equilibrated fluxes which compute y as a solution of small local problems. For more information see [18, section 3.5] and papers [19–22].

This paper focuses merely on the method (4), that is, the minimization of the right hand side of (1.2). We are interested in fast computation of the reliable estimate rather than the indication of regions with high error (adaptivity). The main motivation is to provide people working with "functional a posteriori estimates" concepts and software to speed up their computation and go for larger size problems. In particular, the paper

- (i) formulates a majorant minimization problem in $H(\operatorname{div})$ space and an algorithm for its computation on continuous level;
- (ii) demonstrates a numerical computation of the Friedrich's constant;
- (iii) discretizes the minimization algorithm and applies RT0 elements to obtain a linear system of equations. So far, only vector nodal linear elements were applied [23, 24];
- (iv) introduces a multigrid-preconditioned conjugate gradient method as an iterative solver for resulting linear system and demonstrates its optimality on one benchmark example.

We would also like to attract attention of different groups and evoke cooperations. Since the majorant minimization problem (discretized by RT0 elements) is about 3 times larger than the Poisson problem (discretized by linear nodal elements), the other error estimates specially developed for linear problems might perform faster for the benchmark problem with a smooth solution (the exact comparison is not done here). However, functional a posteriori error estimates are the only tool to provide the guaranteed error of approximation of nonlinear problems [17, 25, 26].

2. Majorant Minimization Problem

Problem 2.1 (Minimization problem). Given a discrete solution $v \in \overset{\circ}{H}^1(\Omega)$, a right-hand side of the Poisson problem $f \in L^2(\Omega)$, the Friedrich's constant belonging to the domain Ω . Find a function $y \in H(\Omega, \text{div})$ satisfying the condition

$$\|\nabla v - y\|_{\Omega} + C_{\Omega} \|\text{div } y + f\|_{\Omega} \longrightarrow \min. \quad (2.1)$$

In order to avoid complications with the nondifferentiability of norm terms in (2.1), we apply the inequality $(a + b)^2 \leq (1 + \beta)a^2 + (1 + (1/\beta))b^2$ valid for all $\beta > 0$ to obtain

$$(\|\nabla v - y\|_{\Omega} + C_{\Omega} \|\text{div } y + f\|_{\Omega})^2 \leq \mathcal{M}(\nabla v, f, \beta, C_{\Omega}, y), \quad (2.2)$$

where the upper bound in (2.2) denotes a functional majorant

$$\mathcal{M}(\nabla v, f, \beta, C_{\Omega}, y) := (1 + \beta) \|\nabla v - y\|_{\Omega}^2 + \left(1 + \frac{1}{\beta}\right) C_{\Omega}^2 \|\text{div } y + f\|_{\Omega}^2. \quad (2.3)$$

The majorant arguments f, v, C_{Ω} are known and $\beta > 0$ and $y \in H(\Omega, \text{div})$ are free parameters. For a fixed choice of parameters β , the majorant represents a quadratic functional in y . On the other hand, for a fixed y , the parameter

$$\beta = \frac{C_{\Omega} \|\text{div } y + f\|_{\Omega}}{\|\nabla v - y\|_{\Omega}} \quad (2.4)$$

minimizes \mathcal{M} amongst all positive β . It suggests the following solution algorithm to Problem 2.1.

Algorithm 2.2 (Majorant minimization algorithm). Given $\beta > 0$.

- (a) Compute y from the minimization of the quadratic problem

$$\mathcal{M}(\nabla v, f, \beta, C_{\Omega}, y) \longrightarrow \min. \quad (2.5)$$

- (b) Update β from y using (2.4). If the convergence in y is not achieved then go to step (a).

For a detailed analysis of step (a) it is convenient to decompose \mathcal{M} in its y -independent and y -dependent parts as $\mathcal{M} = \mathcal{M}_1 + \mathcal{M}_2$, where

$$\begin{aligned} \mathcal{M}_1 &:= (1 + \beta) \|\nabla v\|_{\Omega}^2 + \left(1 + \frac{1}{\beta}\right) C_{\Omega}^2 \|f\|_{\Omega}^2, \\ \mathcal{M}_2 &:= (1 + \beta) \left(\|y\|_{\Omega}^2 - 2(\nabla v, y) \right) + \left(1 + \frac{1}{\beta}\right) C_{\Omega}^2 \left(\|\text{div } y\|_{\Omega}^2 + 2(f, \text{div } y) \right). \end{aligned} \quad (2.6)$$

Here, (\cdot, \cdot) denotes the $L^2(\Omega)$ scalar product. Instead of considering (2.5) in Algorithm 2.2, y is solved from the minimization problem

$$\mathcal{M}_2(\nabla v, f, \beta, C_\Omega, y) \longrightarrow \min \quad (2.7)$$

for given $\nabla v, f, \beta, C_\Omega$. Since $\mathcal{M}_2 = 0$ for $y = 0$, the minimal value of \mathcal{M}_2 must be nonpositive. Besides, it holds $\mathcal{M}_1 \geq 0$ and $\mathcal{M}_1 = 0$ if and only if $f = 0$ and $v = 0$.

The finite element method (FEM) is used for the discretization of the minimization problem (2.7). The domain Ω is divided by a regular triangulation \mathcal{T} in triangles in the sense of Ciarlet [1], that is, \mathcal{T} is a finite partition of Ω into closed triangles; two distinct elements T_1 and T_2 are either disjoint, or $T_1 \cap T_2$ is a complete edge or a common node of both T_1 and T_2 . Let us assume a finite element basis in the space $H(\Omega, \text{div})$ and a vector \hat{y} representing $y \in H(\Omega, \text{div})$ in this basis. Norm terms in the definition of \mathcal{M}_2 are read then after the discretization

$$\|y\|_\Omega^2 = \hat{y}^T M \hat{y}, \quad \|\text{div } y\|_\Omega^2 = \hat{y}^T \text{DIV DIV} \hat{y}, \quad (2.8)$$

where M and DIV DIV represent the "mass" and "div div" matrices. Similarly, linear functionals are discretized as

$$(\nabla v, y) = l_1^T \hat{y}, \quad (f, \text{div } y) = l_2^T \hat{y}, \quad (2.9)$$

with some vectors l_1 and l_2 . It allows to express a discrete form of \mathcal{M}_2 in the provided finite element basis,

$$\mathcal{M}_2 = \hat{y}^T \left[(1 + \beta) M + \left(1 + \frac{1}{\beta}\right) C_\Omega^2 \text{DIV DIV} \right] \hat{y} - 2\hat{y}^T \left[(1 + \beta) l_1 - \left(1 + \frac{1}{\beta}\right) C_\Omega^2 l_2 \right]. \quad (2.10)$$

The minimization of \mathcal{M}_2 with respect to the vector \hat{y} leads for a given value β to a linear system of equations

$$\left[(1 + \beta) M + \left(1 + \frac{1}{\beta}\right) C_\Omega^2 \text{DIV DIV} \right] \hat{y} = (1 + \beta) l_1 - \left(1 + \frac{1}{\beta}\right) C_\Omega^2 l_2 \quad (2.11)$$

for a minimizing vector \hat{y} . The linear system is represented by a matrix which is symmetric and positive definite. Symmetry is the result of symmetries of matrices M and DIV DIV . The positive definiteness is proved by a contradiction: if there was a nonzero vector \hat{y} for which the quadratic form in (2.10) was not positive, we could construct a sequence of arguments $\alpha_i \hat{y}$, $i = 1, 2, \dots$, in which the values of \mathcal{M}_2 would converge (for $i \rightarrow \infty$) to $-\infty$. Consequently, since \mathcal{M}_1 is independent of \hat{y} , sequence of values of the majorant \mathcal{M} would also converge to $-\infty$, which is not possible due to the reliability of the estimate (2.2).

The next benchmark example explains application issues of the majorant estimate (1.2) and of Algorithm 2.2.

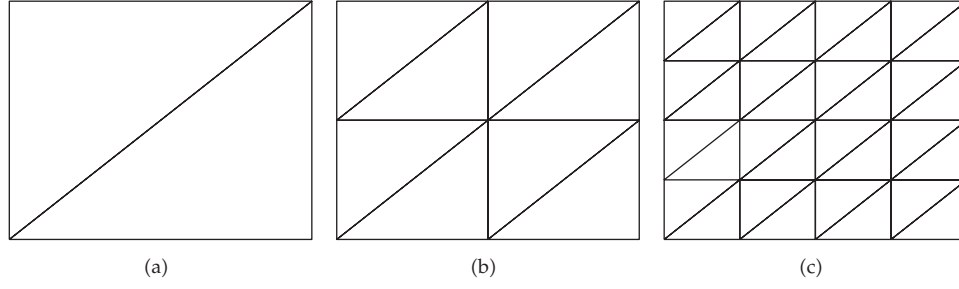


Figure 1: Uniform triangulations (a) \mathcal{T}_0 , (b) \mathcal{T}_1 , and (c) \mathcal{T}_2 .

3. Benchmark Example

Let us assume the right-hand side

$$f(x, y) = 2x(1 - x) + 2y(1 - y) \quad (3.1)$$

in the unit square domain $\Omega = (0, 1) \times (0, 1)$. The square geometry is discretized using a sequence of nested uniform triangular meshes $\mathcal{T}_0, \mathcal{T}_1, \dots$ as displayed in Figure 1. A discrete solution v is computed from by using FEM with the nodal linear (Courant) ansatz functions on each triangular mesh. The discrete solutions v are displayed as the left column pictures of Figure 2. For this particular right-hand side, the exact solution of (1.1) reads

$$u = x(1 - x)y(1 - y) \quad (3.2)$$

for all $(x, y) \in \Omega$ and its flux is

$$\frac{\partial u}{\partial x} = (1 - 2x)y(1 - y), \quad \frac{\partial u}{\partial y} = x(1 - x)(1 - 2y) \quad (3.3)$$

for all $(x, y) \in \Omega$. Therefore, the exact solution of the Benchmark example is a polynomial and, thus, integration error can be avoided.

3.1. Friedrich's Constant and Its Computation

The computation of the functional majorant (2.3) requires the knowledge of the constant C_Ω^2 from the Friedrich's inequality (1.3). Under the assumption of the finite element basis ϕ_1, \dots, ϕ_N we can represent the function $w \in \overset{\circ}{H}^1(\Omega)$ as the vector $\hat{w} \in \mathbb{R}^N$, where $w = \sum_{i=1}^N \hat{w}_i \phi_i$. In the same basis, the bilinear forms read

$$(\nabla w, \nabla w) = \hat{w}^T K_\Delta \hat{w}, \quad (w, w) = \hat{w}^T M_\Delta \hat{w}, \quad (3.4)$$

where K_Δ and M_Δ are stiffness and mass matrices of the discretized Poisson problem (1.1). If the same triangular meshes and nodal linear (Courant) ansatz functions are used, the matrix

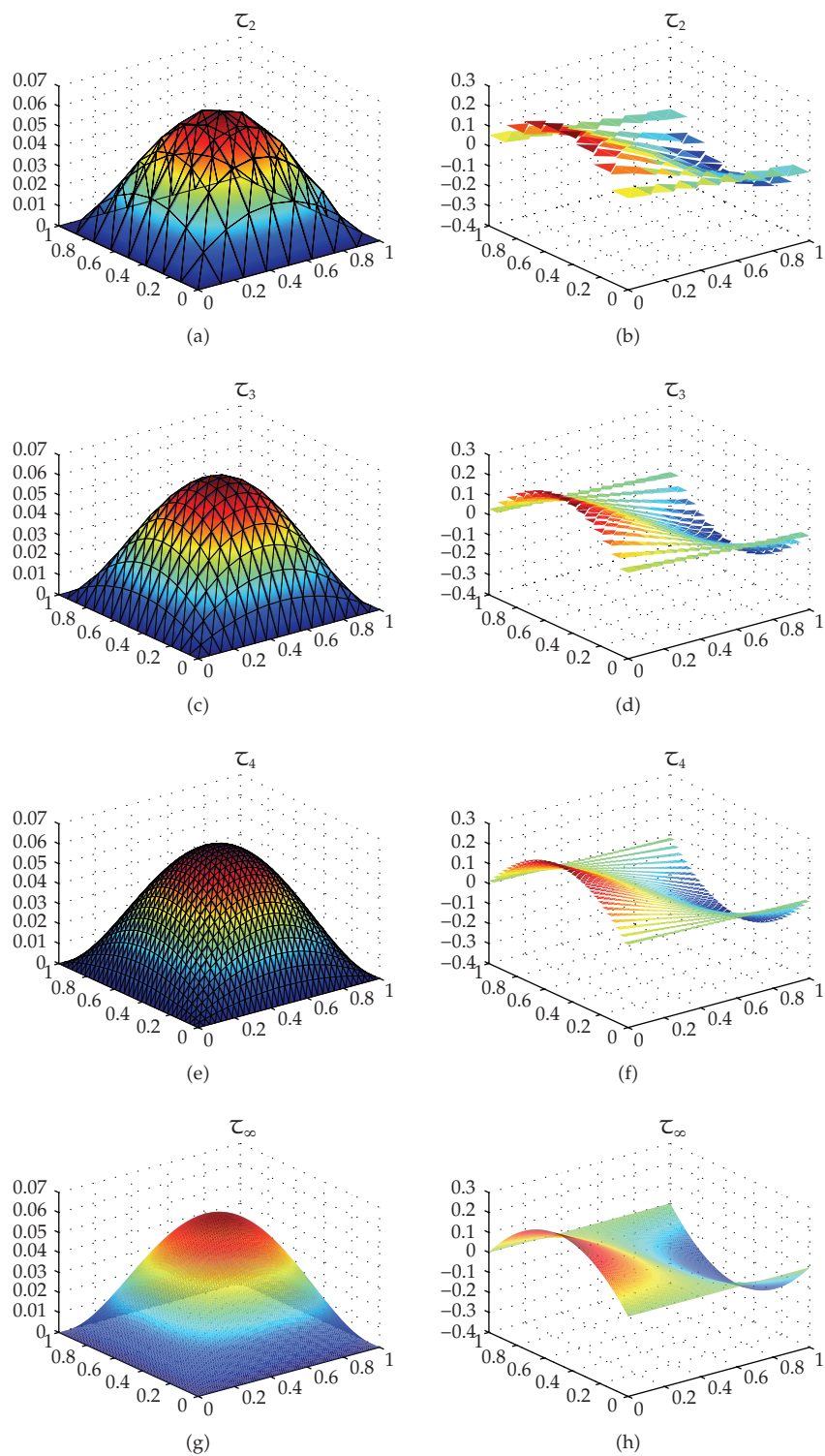


Figure 2: Discrete solutions v and y -component of the flux y computed from Algorithm 2.2 on mesh triangulations τ_2, τ_3, τ_4 , and the exact solution u and its flux y -component $\partial u / \partial y$ (in the bottom row denoted as τ_∞ triangulation).

Table 1: Friedrich's constant for the unit square domain computed from the generalized eigenvalue problem (3.7) on nested uniform meshes.

Level	Mesh size h	Error	C_Ω
1	2^{-1}	0.04830238374264	0.17677669529663
2	2^{-2}	0.01595356174205	0.20912551729722
3	2^{-3}	0.00424589066153	0.22083318837774
4	2^{-4}	0.00107875817236	0.22400032086690
5	2^{-5}	0.00027079918172	0.22480827985755
6	2^{-6}	0.00006776967260	0.22501130936667
7	2^{-7}	0.00001694679400	0.22506213224527
8	2^{-8}	0.00000423697210	0.22507484206717
9	2^{-9}	0.00000105926003	0.22507801977924
Theory	0	0	0.22507907903928

K_Δ is already at our disposal from the computation of the discrete solution v and the mass matrix is generated additively. Then the discretization of the Friedrichs' inequality reads

$$\hat{w}^T M_\Delta \hat{w} \leq C_\Omega^2 \hat{w}^T K_\Delta \hat{w} \quad (3.5)$$

for all vectors $\hat{w} \in \mathbb{R}^N$ respecting zero Dirichlet boundary conditions. The minimal value C_Ω^2 to satisfy the last inequality represents the maximum Rayleigh quotient, that is,

$$C_\Omega^2 = \max_{\hat{w}} \frac{\hat{w}^T M_\Delta \hat{w}}{\hat{w}^T K_\Delta \hat{w}}, \quad (3.6)$$

and it is also equal to the maximal eigenvalue λ of a generalized eigenvalue problem

$$M_\Delta \hat{w} = \lambda K_\Delta \hat{w}. \quad (3.7)$$

Matrices M_Δ and K_Δ were assembled in MATLAB and the default function "eigs" was applied for the computation of the approximate values of the Friedrichs' constant C_Ω . The MATLAB code can be downloaded at <http://www.mathworks.com/matlabcentral/fileexchange/authors/37756> and it is easily extensible to any 2D geometry. Table 1 reports on the values of C_Ω computed on the meshes $\mathcal{T}_1, \dots, \mathcal{T}_9$ and compares them with a theoretical value

$$C_\Omega = \frac{1}{\sqrt{2\pi}} \approx 0.22507907903928 \quad (3.8)$$

known for the unit square domain Ω and zero Dirichlet boundary conditions. Note that the discrete approximations provide a lower bound only, but converge fast to the exact value of the Friedrichs' constant. Here, we observe a quadratic convergence with respect to the mesh size h . In general, the convergence typically depends on the shape of the domain boundary [27].

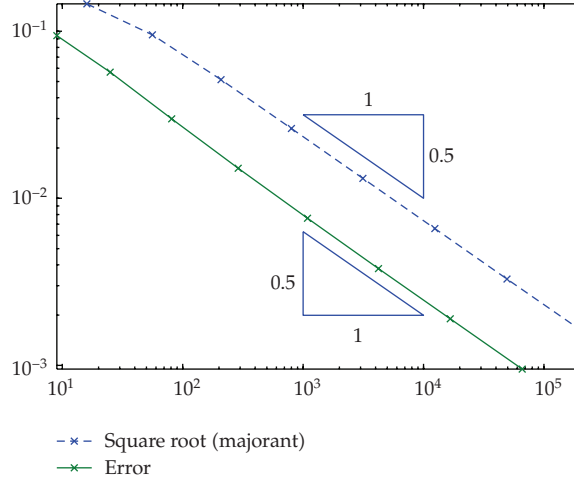


Figure 3: Values of exact error $\|\nabla(u - v)\|_{\Omega}$ and the square root of the majorant $\sqrt{\mathcal{M}}$ displayed versus degrees of freedom (displayed as DOF v and DOF y in Table 2) of their calculation on various uniform mesh levels. Both lines indicate linear convergence with respect to the mesh size.

3.2. Majorant Computation

One step (a) and one step (b) of Algorithm 2.2 assuming an initial value $\beta = 1$ were applied for the computation of the flux $y \in H(\Omega, \text{div})$. For the discretization of $H(\Omega, \text{div})$, Raviart-Thomas spaces of the zero degree (known as RT_0 elements [28]) defined on the uniform meshes $\mathcal{T}_0, \dots, \mathcal{T}_8$ were considered.

Remark 3.1 (Majorant problem size). Note that the number of elements $|\mathcal{T}_n|$ and number of edges $|\mathcal{E}_n|$ of the triangulation \mathcal{T}_n satisfy recurrences

$$|\mathcal{T}_{n+1}| = 4|\mathcal{T}_n|, \quad |\mathcal{E}_{n+1}| = 2|\mathcal{E}_n| + 3|\mathcal{T}_n| \quad (3.9)$$

subject to the conditions $|\mathcal{T}_0| = 2$, $|\mathcal{E}_0| = 5$ related to the coarse triangulation T_0 . These recurrences are solved to provide exact formulas $|\mathcal{T}_n| = 2 \cdot 4^n$, $|\mathcal{E}_n| = 2 \cdot 2^n + 3 \cdot 4^n$. By the known Euler formula for planar graphs, the number of vertices reads $|\mathcal{N}_n| = 1 + 2 \cdot 2^n + 4^n$ and it holds

$$|\mathcal{E}_n| \approx 3|\mathcal{N}_n| \approx \frac{3}{2}|\mathcal{T}_n| \quad (3.10)$$

for large n . It implies that the matrix in Algorithm 2.2 using RT_0 basis is asymptotically 3 times larger than the stiffness matrix from the discretization of Poisson problem by using linear nodal elements.

A MATLAB implementation is based on [29] with some modification with respect to the performance and extension towards a multigrid solver. Quadrature rules exact for polynomials up to the order two were taken for the computation of integrals on triangulations. The right column of Figure 2 displays computed fluxes (only one component due to symmetry reasons) and the exact flux (3.3). It can be observed that both discrete

Table 2: Comparison of the square of exact error $\|\nabla(u - v)\|_{\Omega}^2$ and majorant values \mathcal{M} and on various uniform meshes levels.

Level	DOF v	$\ \nabla(u - v)\ _{\Omega}^2$	DOF y	Majorant \mathcal{M}	I_{eff}
1	9	8.873457e-03	16	2.135435e-02	1.55
2	25	3.240138e-03	56	9.050160e-03	1.67
3	81	8.950033e-04	208	2.635564e-03	1.72
4	289	2.295153e-04	800	6.858397e-04	1.73
5	1089	5.774695e-05	3136	1.732119e-04	1.73
6	4225	1.445987e-05	12416	4.341359e-05	1.73
7	16641	3.616415e-06	49408	1.086033e-05	1.73
8	66049	9.041944e-07	197120	2.715516e-06	1.73

solutions and fluxes (at least visually) converge to exact solution and its flux. By comparing the values of Table 2 or Figure 3, both exact errors and majorant values converge linearly with respect to corresponding degrees of freedom used for their computation. The index of efficiency which remains bounded by the value $I_{\text{eff}} \approx 1.73$ for all mesh levels.

4. Improving Linear Solver

The highest computation costs of Algorithm 2.2 are caused due to the solution of the linear system of (2.11) in step (a) of Algorithm 2.2. Let us consider an iterative method for its solving. The advantage of iterative over direct methods is apparent in this context, since each iteration vector \hat{y}_k for $k = 0, 1, \dots$ as an approximation the solution \hat{y} can be inserted into the majorant (2.10), without the need for solving the linear system very accurately. Since the linear system is represented by a symmetric and positive definite matrix, a preconditioned conjugate gradient method (PCG) is considered.

Algorithm 4.1 (PCG method for a system of equations $Ax = b$ including energy computation). Let an initial iteration vector x_0 be given. Compute an initial residual $r_0 = b - Ax_0$, an initial energy $E_0 = (1/2)(Ax_0, x_0) - (b, x_0)$ and $s_0 = C^{-1}r_0$, $p_0 = s_0$, where C is a given preconditioning matrix. For the iterations $j = 0, 1, \dots$ do the loop

- (1) $\gamma_j = (r_j, s_j) / (p_j, Ap_j)$
- (2) $E_{j+1} = E_j - (\gamma_j (r_j, s_j)) / 2$
- (3) $x_{j+1} = x_j + \gamma_j p_j$
- (4) $r_{j+1} = r_j - \gamma_j Ap_j$
- (5) $s_{j+1} = C^{-1}r_{j+1}$
- (6) $\delta_{j+1} = (r_{j+1}, s_{j+1}) / (r_j, s_j)$
- (7) $p_{j+1} = s_{j+1} + \delta_{j+1} p_j$
- (8) If a given stopping criterion is fulfilled, leave the loop and output the solution x_{j+1} and the energy E_{j+1} .

This algorithm recalls Algorithm 4.1 from [30] with a modification for the computation of an energy in the step (2). The energy is defined as

$$E_j := \frac{1}{2}(Ax_j, x_j) - (b, x_j), \quad (4.1)$$

and the formula in step (2) provides a cheap update of E_{j+1} from E_j without an extra matrix-vector multiplication Ax_{j+1} in (4.1). It can be directly derived from the combination of two formulae. The first one is the known relation between the energy error of the PCG-iterations and their energy

$$\begin{aligned} E_j - E &= \frac{1}{2}(Ax_j, x_j) - \frac{1}{2}(Ax, x) - (b, x_j) + (b, x) \\ &= \frac{1}{2}((Ax_j, x_j) + (Ax, x) - 2(Ax, x_j)) \\ &= \frac{1}{2}(A(x - x_j), x - x_j) := \frac{1}{2}\|x_j - x\|_A^2 \end{aligned} \quad (4.2)$$

where x denotes the exact solution of the linear system $Ax = b$. The second one is a special version of the formula (3.6) for $d = 1$ from [30]

$$\|x - x_{j+1}\|_A^2 = \|x - x_j\|_A^2 - \gamma_j(r_j, s_j). \quad (4.3)$$

The knowledge of the energy E_j is required for the computation of the flux-dependent functional majorant part \mathcal{M}_2 , since it holds (cf. (2.10))

$$\mathcal{M}_2(\nabla v, f, \beta, C_\Omega, \hat{y}_j) = 2E_j, \quad (4.4)$$

where A and b are the matrix and the right-hand side of the linear system (2.11) and the vector x substitutes \hat{y} . For the initial iteration $\hat{y}_0 = 0$, it holds $\mathcal{M}_2(\nabla v, f, \beta, C_\Omega, \hat{y}_0) = 0$ and we obtain an estimate

$$\|\nabla v - y\|_\Omega^2 \leq \mathcal{M}_1(\nabla v, f, \beta, C_\Omega), \quad (4.5)$$

where \mathcal{M}_1 is defined in (2.6). Since $\gamma_j(r_j, s_j) \geq 0$ in (4.3), the PCG method reduces (or at least does not increase) the energy E_j and consequently the value $\mathcal{M}(\nabla v, f, \beta, C_\Omega, \hat{y}_j)$ in each iteration $j = 1, 2, \dots$ to sharpen the estimate (4.5).

4.1. Performance of PCG for Benchmark Example

Let us apply PCG method to one step (a) of Algorithm 2.2 to linear system (2.11), in which $\beta = 1$ is considered. PCG method is terminated in step (8) of Algorithm 4.1 if the stopping

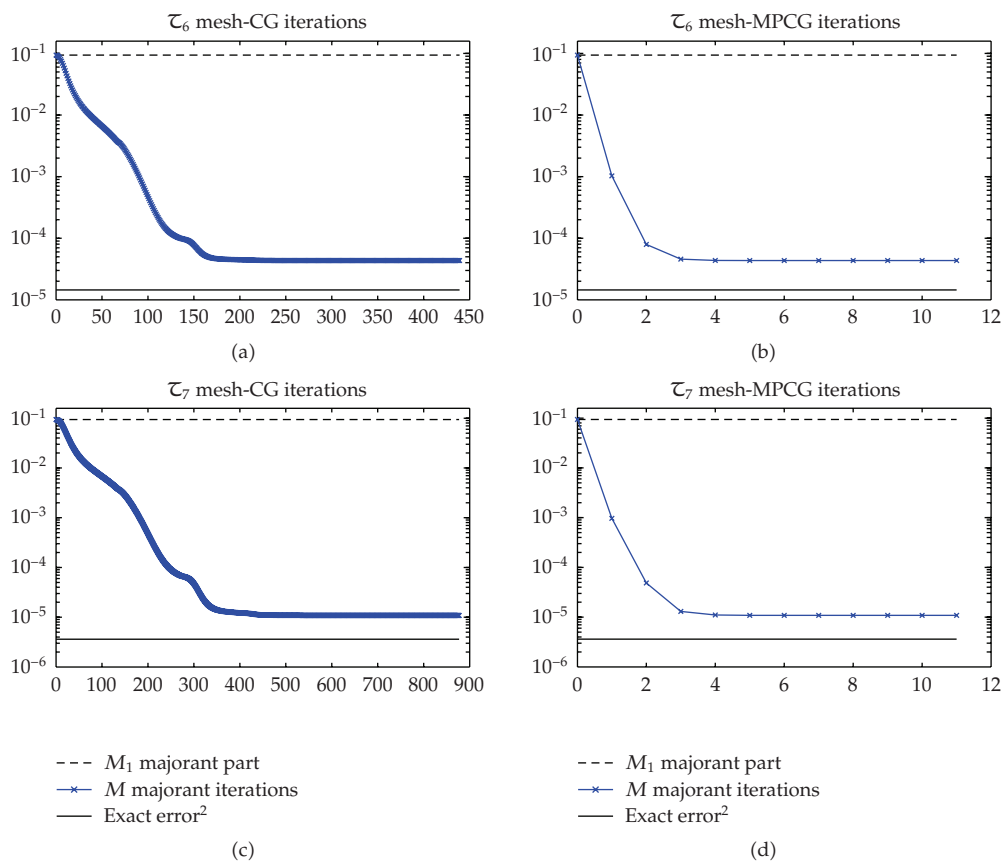


Figure 4: Majorant values displayed versus iterations of CG and MPCG method for \mathcal{T}_6 and \mathcal{T}_7 meshes. For better illustration, the majorant part \mathcal{M}_1 (the upper bound) from (2.6) and the square of the exact error, that is, $\|\nabla(u - v)\|_{\Omega}^2$ (the lower bound) are displayed.

criterion

$$\sqrt{\frac{(r_j, s_j)}{(r_0, s_0)}} \leq \varepsilon \quad (4.6)$$

is fulfilled for some given tolerance ε , here we choose $\varepsilon = 10^{-6}$.

Remark 4.2. If the preconditioner C well approximates A , that is, $C \approx A$, it holds $(r_j, s_j) = (A(x - x_j), C^{-1}A(x - x_j)) \approx \|x - x_j\|_A^2$ and (4.6) is equivalent to a stopping criterion based on the relative A -norm of the error

$$\frac{\|x - x_j\|_A}{\|x - x_0\|_A} \leq \varepsilon. \quad (4.7)$$

Table 3: Number of CG/MPCG iterations (using stopping criterion (4.6) for $\varepsilon = 10^{-6}$) for various nested uniform mesh levels.

Level	Mesh size h	Problem size	CG iterations	MPCG iterations
0	1	5	2	2
1	2^{-1}	16	5	5
2	2^{-2}	56	15	8
3	2^{-3}	208	45	10
4	2^{-4}	800	108	11
5	2^{-5}	3136	219	11
6	2^{-6}	12416	349	11
7	2^{-7}	49408	877	11
8	2^{-8}	197120	1773	11
9	2^{-9}	787456	3506	11

In the case of no preconditioning (then we speak of CG method instead of PCG), C is an identity matrix and (4.6) is equivalent to a stopping criterion

$$\frac{\|b - Ax_j\|_2}{\|b\|_2} \leq \varepsilon, \quad (4.8)$$

which is a default stopping criterion implemented in MATLAB.

As an operation of the preconditioner C in Algorithm 4.1, we apply a simple V-cycle of a geometrical multigrid method [31] based on provided hierarchy of nested triangulations $\mathcal{T}_0, \mathcal{T}_1, \mathcal{T}_2, \dots$. As the linear system (2.11) arises from a discretization in $H(\Omega, \text{div})$ space, a special smoother as a part of the multigrid method is required. Our choice is the additive version of the smoother of Arnold, Falk and Winther [32] using one presmoothing and one postsmoothing steps.

Table 3 compares numbers of iterations of non-preconditioned (CG) and multigrid-preconditioned (MPCG) method for various levels of triangulation. Single CG or MPCG iterations and the corresponding majorant values are displayed on Figure 4 for mesh levels 6 and 7. The number of iterations reflects typical properties of conjugate gradients and system matrices arising in elliptic partial boundary value problems. For shape regular triangulations, the condition number of A is known to be proportional to h^{-2} , where h is the mesh-size parameter, that is, $\text{cond}(A) \approx h^{-2}$. Furthermore, the number of CG-iterations (with respect to the same stopping tolerance ε) satisfies $\text{ITER} \approx \sqrt{\text{cond}(A)}$. Together, it holds

$$\text{ITER} \approx h^{-1}. \quad (4.9)$$

The mesh size h is halved after each uniform refinement and therefore the number of CG-iterations is according to (4.9) expected to be doubled in the non-preconditioned case. For the multigrid-preconditioned CG method (MPCG) we observe that the number of iterations remains bounded, in our case

$$\text{ITER} \leq 11 \quad (4.10)$$

is valid for all mesh levels. This property is so-called mesh independence and it demonstrates the optimality of the chosen multigrid preconditioner. A detailed observation of the right column pictures of Figure 4 indicates that 4 iterations ($ITER = 4$) already provide a very sharp majorant value without the need for additional iterations.

5. Conclusions

The minimization of the majorant term in the functional a posteriori estimate is done by solving a sequence of systems of linear equations for an unknown approximation of the flux of an exact solution. The solution of the first linear system is efficiently obtained by the multigrid-preconditioned conjugate gradient method. The considered Benchmark example shows that few iterations already provide very accurate flux approximation for the computation of the functional majorant. Therefore, an optimal strategy for the termination process of the preconditioned conjugate gradient method in connection to the majorant computation remains an interesting open question.

Acknowledgments

The author acknowledges support from the Austrian Science Fund “Fonds zur Förderung der wissenschaftlichen Forschung (FWF)” for support under Grant SFB F013/F1306 in Linz, Austria, as well as the support of YFF Project “Modelling Transport in Porous Media over Multiple Scales” at the University Bergen, Norway. The comments of S. Repin (St. Petersburg) improved an overview on a posteriori estimates, “the Linz multigrid community” (U. Langer, R. Simon, and S. Zanglmayr) provided practical hints concerning multigrid application and A. Janka (Fribourg) assisted at the MATLAB vectorization of the smoother of Arnold, Falk, and Winther.

References

- [1] P. G. Ciarlet, *The Finite Element Method for Elliptic Problems*, Studies in Mathematics and Its Applications, North-Holland, Amsterdam, The Netherlands, 1978.
- [2] M. Ainsworth and J. T. Oden, *A Posteriori Error Estimation in Finite Element Analysis*, Pure and Applied Mathematics, John Wiley & Sons, New York, NY, USA, 2000.
- [3] I. Babuška and T. Strouboulis, *The Finite Element Method and Its Reliability*, Numerical Mathematics and Scientific Computation, Oxford University Press, New York, NY, USA, 2001.
- [4] W. Bangerth and R. Rannacher, *Adaptive Finite Element Methods for Differential Equations*, Lectures in Mathematics ETH Zürich, Birkhäuser, Basel, Switzerland, 2003.
- [5] R. Verfürth, *A Review of a Posteriori Error Estimation and Adaptive Mesh-Refinement Techniques*, John Wiley & Sons, Teubner, New-York, NY, USA, 1996.
- [6] I. Babuška and W. C. Rheinboldt, “A posteriori error estimates for the finite element method,” *International Journal for Numerical Methods in Engineering*, vol. 12, pp. 1597–1615, 1978.
- [7] I. Babuška and W. C. Rheinboldt, “Error estimates for adaptive finite element computations,” *SIAM Journal on Numerical Analysis*, vol. 15, no. 4, pp. 736–754, 1978.
- [8] C. Carstensen and S. Bartels, “Each averaging technique yields reliable a posteriori error control in FEM on unstructured grids. I. Low order conforming, nonconforming, and mixed FEM,” *Mathematics of Computation*, vol. 71, no. 239, pp. 945–969, 2002.
- [9] S. I. Repin, “A posteriori error estimates for approximate solutions to variational problems with strongly convex functionals,” *Journal of Mathematical Sciences*, vol. 97, no. 4, pp. 4311–4328, 1999, translated from *Problemy Matematicheskogo Analiza*, no. 17, pp. 227–237, 1997.

- [10] S. I. Repin and L. S. Xanthis, "A posteriori error estimation for nonlinear variational problems," *Comptes Rendus de l'Académie des Sciences*, vol. 324, no. 10, pp. 1169–1174, 1997.
- [11] S. I. Repin, "A posteriori error estimation for variational problems with uniformly convex functionals," *Mathematics of Computation*, vol. 69, no. 230, pp. 481–500, 2000.
- [12] S. I. Repin, "Two-sided estimates of deviation from exact solutions of uniformly elliptic equations," in *Proceedings of the St. Petersburg Mathematical Society, Vol. IX*, vol. 209 of *American Mathematical Society Translations: Series 2*, pp. 143–171, American Mathematical Society, Providence, RI, USA.
- [13] M. Bildhauer, M. Fuchs, and S. Repin, "A posteriori error estimates for stationary slow flows of power-law fluids," *Journal of Non-Newtonian Fluid Mechanics*, vol. 142, no. 1–3, pp. 112–122, 2007.
- [14] M. Bildhauer, M. Fuchs, and S. Repin, "A functional type a posteriori error analysis for the Ramberg-Osgood model," *Zeitschrift für Angewandte Mathematik und Mechanik*, vol. 87, no. 11-12, pp. 860–876, 2007.
- [15] M. Fuchs and S. Repin, "A posteriori error estimates of functional type for variational problems related to generalized Newtonian fluids," *Mathematical Methods in the Applied Sciences*, vol. 29, no. 18, pp. 2225–2244, 2006.
- [16] M. E. Frolov, P. Neittaanmäki, and S. I. Repin, "Guaranteed functional error estimates for the Reissner-Mindlin plate problem," *Journal of Mathematical Sciences*, vol. 132, no. 4, pp. 553–561, 2006.
- [17] S. Repin and J. Valdman, "Functional a posteriori error estimates for problems with nonlinear boundary conditions," *Journal of Numerical Mathematics*, vol. 16, no. 1, pp. 51–81, 2008.
- [18] S. Repin, "A posteriori error estimation methods for partial differential equations," in *Lectures on Advanced Computational Methods in Mechanics*, J. Kraus and U. Langer, Eds., vol. 1 of *Radon Series on Computational and Applied Mathematics*, pp. 161–226, Walter de Gruyter, Berlin, Germany, 2007.
- [19] D. Braess and J. Schöberl, "Equilibrated residual error estimator for edge elements," *Mathematics of Computation*, vol. 77, no. 262, pp. 651–672, 2008.
- [20] T. Vejchodský, "Guaranteed and locally computable a posteriori error estimate," *IMA Journal of Numerical Analysis*, vol. 26, no. 3, pp. 525–540, 2006.
- [21] M. Vohralík, "A posteriori error estimation in the conforming finite element method based on its local conservativity and using local minimization," *Comptes Rendus Mathématique*, vol. 346, no. 11-12, pp. 687–690, 2008.
- [22] R. Luce and B. I. Wohlmuth, "A local a posteriori error estimator based on equilibrated fluxes," *SIAM Journal on Numerical Analysis*, vol. 42, no. 4, pp. 1394–1414, 2004.
- [23] S. Repin, S. Sauter, and A. Smolianski, "Two-sided a posteriori error estimates for mixed formulations of elliptic problems," *SIAM Journal on Numerical Analysis*, vol. 45, no. 3, pp. 928–945, 2007.
- [24] R. Lazarov, S. Repin, and S. Tomar, "Functional a posteriori error estimates for discontinuous Galerkin approximations of elliptic problems," *Numerical Methods for Partial Differential Equations*, vol. 25, no. 4, pp. 952–971, 2008.
- [25] S. Repin, S. Sauter, and A. Smolianski, "A posteriori error estimation for the Dirichlet problem with account of the error in the approximation of boundary conditions," *Computing*, vol. 70, no. 3, pp. 205–233, 2003.
- [26] S. Repin and J. Valdman, "Functional a posteriori error estimates for incremental models in elastoplasticity," *Central European Journal of Mathematics*, vol. 7, no. 3, pp. 506–519, 2009.
- [27] J. R. Kuttler and V. G. Sigillito, "Eigenvalues of the Laplacian in two dimensions," *SIAM Review*, vol. 26, no. 2, pp. 163–193, 1984.
- [28] P.-A. Raviart and J. M. Thomas, "A mixed finite element method for 2nd order elliptic problems," in *Mathematical Aspects of Finite Element Methods (Proc. Conf., Consiglio Naz. delle Ricerche (C.N.R.), Rome, 1975)*, vol. 606 of *Lecture Notes in Mathematics*, pp. 292–315, Springer, Berlin, Germany, 1977.
- [29] C. Bahriawati and C. Carstensen, "Three MATLAB implementations of the lowest-order Raviart-Thomas MFEM with a posteriori error control," *Computational Methods in Applied Mathematics*, vol. 5, no. 4, pp. 333–361, 2005.
- [30] Z. Strakoš and P. Tichý, "Error estimation in preconditioned conjugate gradients," *BIT Numerical Mathematics*, vol. 45, no. 4, pp. 789–817, 2005.
- [31] W. Hackbusch, *Multi-Grid Methods and Applications*, Springer, Berlin, Germany, 1985.
- [32] D. N. Arnold, R. S. Falk, and R. Winther, "Preconditioning in $H(\operatorname{div})$ and applications," *Mathematics of Computation*, vol. 66, no. 219, pp. 957–984, 1997.