Distribution of the discretization and algebraic error in numerical solution of partial differential equations

J. Papež[a,b,1], J. Liesen[c,2], Z. Strakoš[a,3,*]

[a] *Faculty of Mathematics and Physics, Charles University in Prague, Sokolovská 83, 186 75 Prague, Czech Republic*
[b] *Institute of Computer Science, Academy of Sciences of the Czech Republic, Pod Vodárenskou věží 2, 182 07 Prague, Czech Republic*
[c] *Institute of Mathematics, Technical University of Berlin, Straße des 17. Juni 136, 10623 Berlin, Germany*

Abstract

In the adaptive numerical solution of partial differential equations, local mesh refinement is used together with a posteriori error analysis in order to equilibrate the discretization error distribution over the domain. Since the discretized algebraic problems are *not solved exactly*, a natural question is whether the spatial distribution of the algebraic error is analogous to the spatial distribution of the discretization error. The main goal of this paper is to illustrate using standard boundary value model problems that this may not hold. On the contrary, the algebraic error can have large local components which can significantly dominate the total error in some parts of the domain. The illustrated phenomenon is of general significance and it is not restricted to some particular problems or dimensions. To our knowledge, the discrepancy between the spatial distribution of the discretization and algebraic errors has not been studied in detail elsewhere.

Keywords: Numerical solution of partial differential equations, finite element method, adaptivity, a posteriori error analysis, discretization error, algebraic error, spatial distribution of the error
2010 MSC: 65F10, 65N15, 65N30, 65N22, 65Y20

*Corresponding author
Email addresses: papez@cs.cas.cz (J. Papež), liesen@math.tu-berlin.de (J. Liesen), strakos@karlin.mff.cuni.cz (Z. Strakoš)

[1] The research of this author was supported by the GAUK grant 695612, by the ERC-CZ project LL1202 and by the project IAA100300802 of the Grant Agency of the ASCR.
[2] The research of this author was supported by the Heisenberg Program of Deutsche Forschungsgemeinschaft (DFG).
[3] The research of this author was supported by the GACR grant 201/09/0917 and by the ERC-CZ project LL1202.

1. Introduction

In numerical solution of partial differential equations, a sufficiently accurate solution (the meaning depends on the particular problem) of the linear algebraic system arising from discretization has to be considered. When the finite element method (FEM) is used for discretization, the system matrix is sparse. The sparsity of the algebraic system matrix is presented as a fundamental advantage of the FEM. It allows to obtain a numerical solution when the problem is hard and the discretized linear system is very large. It is worth, however, to examine some *mathematical* consequences which do not seem to be addressed in the FEM literature.

The FEM generates an approximate solution in form of a linear combination of basis functions with *local* supports. Each basis function multiplied by the proper coefficient thus approximates the desired solution only locally. The *global* approximation property of the FEM discrete solution is then ensured by solving the linear algebraic system for the unknown coefficients; the linear algebraic system links the local approximation of the unknown function in different parts of the domain. If the linear algebraic system is solved *exactly*, then all is fine. But in practice we do not solve exactly. In hard problems we even *do not want* to achieve a small algebraic error. That might be too costly or even impossible to get; see, e.g., [7, Sections 1–3], [24, Sections 1 and 6], [33, Section 2.6], the discussion in [34, pp. 36 and 72], and [38, Section 1]. Then, however, one should naturally ask whether the spatial distribution of the algebraic error in the domain can significantly differ from the distribution of the discretization error. There is no a priori evidence that these distributions are to be analogous. On the contrary, from the nature of algebraic solvers, either direct or iterative, there seems to be no reason for equilibrating the algebraic error over the domain. Numerical results presented in this paper demonstrate that the algebraic error can indeed significantly dominate the total error in some part of the domain. To our knowledge, apart from a brief discussion in [26, Sections 5.1 and 5.9.4], the presented phenomenon has not been studied elsewhere.

In order to avoid misunderstandings, it is worth to point out that the phenomenon described in this paper is not related to the so-called "smoothing properties" of the conjugate gradient (CG) method [23] or to the investigation of smoothing in the multilevel setting (for such analyses see, e.g., [36] or [41, Chapter 9]). Moreover, it is *not* due to the particular iterative solver or due to the specifics of the model problems used in this paper for illustration. Following the standard methodology used in the numerical PDE literature for decades (see, e.g., [8, 15, 19]), we start by illustrating the phenomenon using the simplest 1D boundary value problem. Furthermore, in order to plot illustrative figures, we use a small number of discretization nodes. In order to avoid the impression that the simplicity or specifics of the 1D model problem diminish the message, we also present numerical examples with more complicated 2D model problems that illustrate the same phenomenon.

Several other phenomena, in particular the pollution error (see, e.g., [9, 31]) and superconvergence of the discretization error in the internal nodes (see,

e.g., [42]) are also of interest in the investigation of the spatial error distributions. Investigations of such phenomena are, however, beyond the scope of this paper.

The paper is organized as follows. The 1D model problem and experimental observations for this problem are described in Section 2. In Section 3 the total error is interpreted via a modification of the discretization mesh. Section 4 explains the local behavior of the algebraic error using the spectral analysis and the approximation properties of the algebraic solver (here the CG method). Section 5 presents some numerical results that illustrate the presence of the described phenomenon on 2D model problems and adaptive PDE computations. The paper ends with concluding remarks.

2. 1D model problem

We consider the 1D Poisson boundary value problem

$$-u''(x) \quad = \quad f(x)\,, \quad 0 < x < 1\,, \qquad u(0) = u(1) = 0\,, \tag{1}$$

where $f(x)$ is a given (continuous) function, $0 \le x \le 1$. This model problem is frequently used in mathematical literature for illustrations of various analytical as well as numerical phenomena; see, e.g., [15, Section 6.2.2], [19, Section 5.5], [30], [32, Section 3.2.1].

Denoting by $H_0^1(\Omega)$ the standard Sobolev space of functions having square integrable (weak) derivatives in $\Omega \equiv (0,1)$ and vanishing on the end points (in the sense of traces), the weak formulation of (1) looks for $u \in H_0^1(\Omega)$ such that

$$a(u,v) = \ell(v) \quad \text{for all } v \in H_0^1(\Omega)\,, \tag{2}$$

where

$$a(u,v) \equiv \int_0^1 u'\,v'\,, \qquad \ell(v) \equiv \int_0^1 v\,f\,.$$

The bilinear form $a(\cdot,\cdot)$ introduces on $H_0^1(\Omega)$ the *energy norm*

$$\|v'\| = a(v,v)^{1/2}\,, \quad v \in H_0^1(\Omega)\,. \tag{3}$$

We point out that the energy norm is relevant in many applications; see, e.g., [20, Section 2.2.1].

We discretize the problem (2) by the FEM on the uniform mesh with n inner nodes, i.e. with the mesh size $h = 1/(n+1)$, using the continuous piecewise linear basis functions $\phi_j,\ j = 1,\dots,n$, satisfying

$$\begin{aligned} \phi_j(jh) &= 1\,, \\ \phi_j(x) &= 0\,, \quad 0 \le x \le (j-1)\,h \quad \text{and} \quad (j+1)\,h \le x \le 1\,. \end{aligned}$$

The discretized problem then looks for $u_h \in V_h \equiv \mathrm{span}\{\phi_1,\dots,\phi_n\}$ such that

$$a(u_h,v_h) = \ell(v_h) \quad \text{for all } v_h \in V_h\,. \tag{4}$$

The finite-dimensional problem (4) can be equivalently formulated as the system of the linear algebraic equations

$$\mathbf{A}\mathbf{x} = \mathbf{b}\,, \tag{5}$$

where the *stiffness matrix* $\mathbf{A} \in \mathbb{R}^{n\times n}$ and the *load vector* $\mathbf{b} \in \mathbb{R}^n$ are given by

$$\mathbf{A} = [A_{ij}]\,, \qquad A_{ij} = a(\phi_j, \phi_i), \tag{6}$$

$$\mathbf{b} = [b_1, \ldots, b_n]^T\,, \quad b_i = \ell(\phi_i)\,, \qquad i, j = 1, \ldots, n\,. \tag{7}$$

The solution $\mathbf{x} = [\xi_1, \ldots, \xi_n]^T$ of (5) contains the coefficients of the Galerkin FEM solution u_h of (4) with the respect to the FEM basis $\phi_1, \ldots, \phi_n$, i.e.

$$u_h = \sum_{j=1}^{n} \xi_j \phi_j\,. \tag{8}$$

In the 1D problem (1), the Galerkin FEM solution u_h is known to coincide with the solution u at the nodes of the mesh; see, e.g., [8, Corollary 4.1.1]. Therefore the coefficients ξ_j are equal to the values of u in the nodes,

$$\xi_j = u(jh)\,, \quad j = 1, \ldots, n\,. \tag{9}$$

The stiffness matrix $\mathbf{A}$ has the tridiagonal form

$$\mathbf{A} = h^{-1} \begin{pmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{pmatrix}. \tag{10}$$

The eigenvalues λ_i and eigenvectors $\mathbf{y}_i = [\eta_{1i}, \ldots, \eta_{ni}]^T$ of $\mathbf{A}$, $i = 1, \ldots, n$, are known analytically (for details and their relationship to the eigenvalues and eigenfunctions of the continuous Laplace operator see, e.g., [10]),

$$\lambda_i \;=\; 4\,h^{-1} \sin^2\left(\frac{i\,\pi}{2(n+1)}\right), \tag{11}$$

$$\eta_{ji} \;=\; \sqrt{\frac{2}{n+1}} \sin\left(\frac{j\,i\,\pi}{n+1}\right), \quad j = 1, \ldots, n\,. \tag{12}$$

The approximations w_i to the eigenfunctions of the continuous operator are then given by

$$w_i = \sum_{j=1}^{n} \eta_{ji}\,\phi_j\,, \qquad w_i(\ell h) = \eta_{\ell i}\,. \tag{13}$$

Remark 1. Unlike in the 2D Poisson problem, the stiffness matrix $\mathbf{A}$ in (10) and hence its eigenvalues in (11) depend on the mesh size through the multiplicative factor h^{-1}. This is often avoided by multiplying the system $\mathbf{Ax} = \mathbf{b}$ by h, which does not affect the conditioning of the matrix. However, since the algebraic energy norms $\|\mathbf{z}\|_{\mathbf{A}}$ and $\|\mathbf{z}\|_{(h\mathbf{A})}$ are different, such scaling would later be inconvenient, which is why we prefer to keep the matrix $\mathbf{A}$ as in (10).

We now consider solving the system $\mathbf{Ax} = \mathbf{b}$ using the (unpreconditioned) conjugate gradient (CG) method [23]. As mentioned in the Introduction, our point is to demonstrate on the simplest model problem the possible differences in the distribution of the discretization error and the algebraic error.

Given an initial approximation $\mathbf{x}_0$ and the corresponding initial residual $\mathbf{r}_0 \equiv \mathbf{b} - \mathbf{Ax}_0$, the CG method generates approximations $\mathbf{x}_k \in \mathbf{x}_0 + \mathcal{K}_k(\mathbf{A}, \mathbf{r}_0)$, where $\mathcal{K}_k(\mathbf{A}, \mathbf{r}_0) \equiv \mathrm{span}\{\mathbf{r}_0, \mathbf{Ar}_0, \ldots, \mathbf{A}^{k-1}\mathbf{r}_0\}$ is the kth Krylov subspace generated by $\mathbf{A}$ and $\mathbf{r}_0$. It is well known that these approximations minimize the $\mathbf{A}$-norm of the error, i.e.,

$$\|\mathbf{x} - \mathbf{x}_k\|_{\mathbf{A}} = \min_{\mathbf{z} \in \mathbf{x}_0 + \mathcal{K}_k(\mathbf{A}, \mathbf{r}_0)} \|\mathbf{x} - \mathbf{z}\|_{\mathbf{A}} \,;$$

see, e.g., [23, Theorem 4.3].

Writing $\mathbf{x}_k = [\xi_1^{(k)}, \ldots, \xi_n^{(k)}]^T$, the resulting approximation of the Galerkin solution u_h in (8) is given by

$$u_h^{(k)} = \sum_{j=1}^{n} \xi_j^{(k)} \phi_j. \tag{14}$$

If u is the exact solution of the model problem (2), then $u - u_h$ is the *discretization error*, $u_h - u_h^{(k)}$ is the *algebraic error*, and $u - u_h^{(k)}$ is the *total error*. As a simple consequence of the Galerkin orthogonality property, the energy norms of these errors satisfy

$$\begin{aligned} \|(u - u_h^{(k)})'\|^2 &= \|(u - u_h)'\|^2 + \|(u_h - u_h^{(k)})'\|^2 \\ &= \|(u - u_h)'\|^2 + \|\mathbf{x} - \mathbf{x}_k\|_{\mathbf{A}}^2 \,; \end{aligned} \tag{15}$$

see, e.g., [14, Theorem 1.3, p. 38]. This means that the CG method leads to an approximation $u_h^{(k)}$ that minimizes the energy norm of the total error over all approximations determined by coefficient vectors from the affine subspace $\mathbf{x}_0 + \mathcal{K}_k(\mathbf{A}, \mathbf{r}_0)$.

Remark 2. The equality (15) holds for any vector $\mathbf{x}_k \in \mathbb{R}^n$ and the corresponding approximation of the form (14). In particular, it holds also for the results of finite precision CG computations.

As in [15, p. 120], we consider the exact solution

$$u = \exp(-5\,(x - 0.5)^2) - \exp(-5/4) \tag{16}$$

of (1). To obtain the right-hand side $\mathbf{b}$ of the linear algebraic system one may use (9) and hence compute $\mathbf{b} = \mathbf{A}\mathbf{x}$. In order to use an approach analogous to higher dimensions we have chosen to evaluate $\mathbf{b}$ as in (7) using the MATLAB function `quad` (i.e. the adaptive Simpson rule). In comparison with the computation of $\mathbf{b} = \mathbf{A}\mathbf{x}$, the differences are, however, negligible. Furthermore, we have evaluated the error norms by applying the MATLAB function `quad` to the analytic expressions for $(u - u_h)'$ and $u - u_h$ in each subinterval.

Let us now describe our numerical results. We consider the FEM discretization using $n = 19$ inner nodes. This rather small number of nodes allows us to plot illustrative figures, but similar results can be obtained for any choice of n. The resulting solution u and the discretization error $u - u_h$ are shown in Figure 1. The (squared) energy and L_2 norms of the discretization error are equal to (up to the negligible rounding errors in evaluation of the norms)

$$\|(u - u_h)'\|^2 = 6.8078\text{e-}3 \qquad \text{and} \qquad \|u - u_h\|^2 = 1.7006\text{e-}6. \tag{17}$$

The condition number of the matrix $\mathbf{A}$ is $\kappa(\mathbf{A}) = \lambda_n/\lambda_1 = 161.4$.

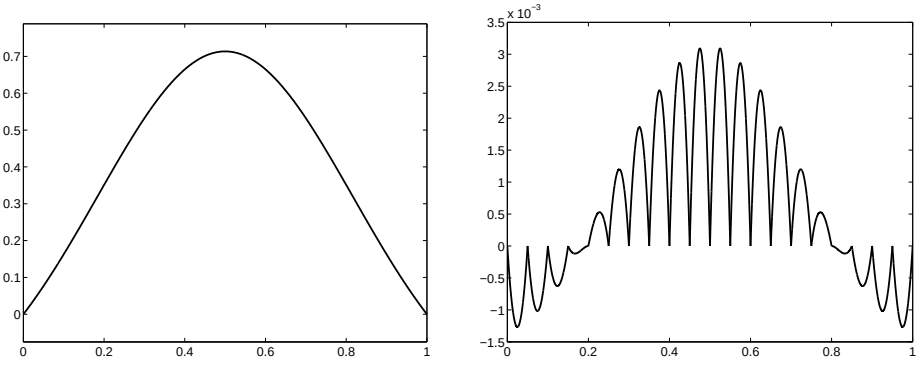

Figure 1: Left: the exact solution u (see (16)). Right: the discretization error $u - u_h$; the vertical axis is scaled by 10^{-3}.

In our experiments we apply the standard implementation of the CG method [23] with $\mathbf{x}_0 = 0$ to $\mathbf{A}\mathbf{x} = \mathbf{b}$. Figure 2 shows the relative $\mathbf{A}$-norm of the algebraic errors $\|\mathbf{x} - \mathbf{x}_k\|_{\mathbf{A}}/\|\mathbf{x} - \mathbf{x}_0\|_{\mathbf{A}}$. In order to show that rounding errors play (almost) no role in our reported results, we also plot the loss of orthogonality among the normalized residual vectors measured in the Frobenius norm for both the standard CG implementation and the CG implementation with double reorthogonalized residuals, which simulates exact arithmetic; see, e.g., [22]. We observe that the loss of orthogonality in the standard CG implementation remains close to the machine precision level, so that the effect of rounding errors indeed is negligible. Taking into account the distribution of the eigenvalues of $\mathbf{A}$ and the choice $\mathbf{x}_0 = \mathbf{0}$, this is to be expected; see [28].

The squared $\mathbf{A}$-norm of the algebraic error $\|\mathbf{x} - \mathbf{x}_k\|_{\mathbf{A}}^2$ at the iteration steps $k = 7, 8, 9, 10$ is given in the first column of Table 1. The second column

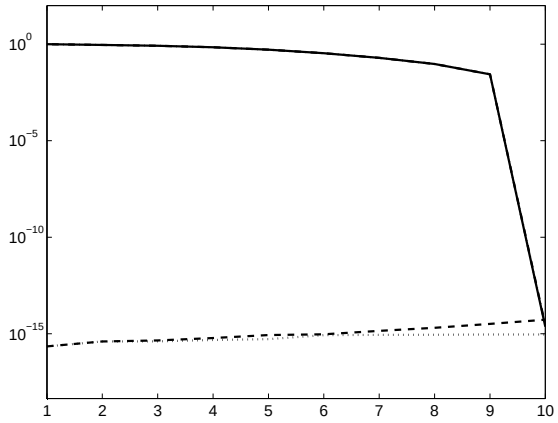

Figure 2: The relative $\mathbf{A}$-norm of the error $\|\mathbf{x}-\mathbf{x}_k\|_{\mathbf{A}}/\|\mathbf{x}-\mathbf{x}_0\|_{\mathbf{A}}$ (solid line), the loss of orthogonality in the standard CG implementation (dashed line) and the loss of orthogonality in the CG implementation with double reorthogonalized residuals (dotted line). In our computations, rounding errors do not play a significant role.

Table 1: Size of the algebraic and total error at several iteration steps for the exact solution (16).

k	$\|\mathbf{x}-\mathbf{x}_k\|_{\mathbf{A}}^2$	$\|\mathbf{x}-\mathbf{x}_k\|^2$	$\|(u-u_h^{(k)})'\|^2$	$\|u-u_h^{(k)}\|^2$
7	6.3002e-2	9.9299e-3	6.9810e-2	4.9817e-4
8	1.4505e-2	9.5751e-4	2.1313e-2	4.9570e-5
9	1.2382e-3	2.7011e-5	8.0459e-3	3.0507e-6
10	6.3248e-30	2.2880e-31	6.8078e-3	1.7006e-6

contains, for comparison, the squared Euclidean norm $\|\mathbf{x}-\mathbf{x}_k\|^2$. For the energy and the L_2 norm of the total error $u-u_h^{(k)}$ see the third and the fourth column, respectively.

The algebraic and total errors are visualized for the steps $k=8,9$ in Figure 3. To describe our main point we look at the step $k=9$. First note that at this step we have

$$\|\mathbf{x}-\mathbf{x}_9\|_{\mathbf{A}}^2 \;=\; 1.2382\text{e-}3 \;<\; 6.8078\text{e-}3 \;=\; \|(u-u_h)'\|^2\,;$$

cf. (17). In words, the globally measured energy norm of the algebraic error is smaller than the globally measured energy norm of the discretization error. On the other hand, as shown in the right part of Figure 3, the algebraic error is strongly localized in the middle of the domain; here in particular at the component $\xi_{10}^{(9)}$ of $\mathbf{x}_9$, which is much less accurate than the other components. This localization of the algebraic error substantially affects the shape of the total error and leads to the following essential observations:

(1) The spatial distributions of the discretization error and the algebraic error can be very different from each other.

(2) The value of the (globally measured) energy norm may not be descriptive.

Similar observations of the error distribution can be made for $k = 8$, which is shown for illustration in the left part of Figure 3. In this step, however, we have $\|\mathbf{x} - \mathbf{x}_8\|_{\mathbf{A}}^2 > \|(u - u_h)'\|^2$.

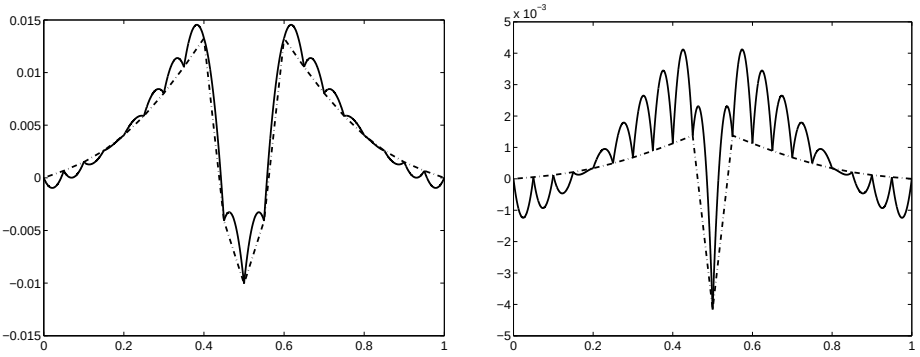

Figure 3: The algebraic error $u_h - u_h^{(k)}$ (dashed-dotted line) and the total error $u - u_h^{(k)}$ (solid line) at the 8th iteration (left) and at the 9th iteration (right). The vertical axis in the right part of the figure is scaled by 10^{-3}.

The presented example considers the simplest model problem. It does not *prove* that in practical problems the observed phenomenon appears on a catastrophic scale. On the other hand, the presented result is disturbing and poses a question about many commonly used ways of a posteriori error evaluation using global error measures, not distinguishing the sources of error or considering only the discretization error.

One may object that if the error is measured in the L_2 norm instead of the energy norm, one does not see much discrepancy – both $\|\mathbf{x}-\mathbf{x}_9\|_{\mathbf{A}}$ and $\|\mathbf{x}-\mathbf{x}_9\|$ are still relatively large in comparison to $\|u - u_h\|$. This, however, is not an objection against our two points made above. For the given model problem (as well as for a large class of problems with self-adjoint bounded and coercive operators; see, e.g., [19, 20]) the energy norm is very natural to consider. With the Galerkin discretization it allows the fundamental Pythagorean identity to be expressed in the form (15), or, more generally, as

$$\|\nabla(u - u_h^{(k)})\|^2 = \|\nabla(u - u_h)\|^2 + \|\mathbf{x} - \mathbf{x}_k\|_{\mathbf{A}}^2 \,. \tag{18}$$

This relates in a straightforward way the size of the discretization and algebraic errors. There is no equality analogous to (18) for the L_2 norm of the total, discretization and algebraic errors. Moreover, the main point is that evaluation of the algebraic error globally using *any norm* is not sufficient. It should be complemented by investigation of the spatial distribution of the error over the domain or at the local areas of interest.

In order to demonstrate that the above observations are not an artefact of the special solution u in (16), we show also the results for the polynomial exact

solution

$$u = (x-2)\,(x-1)\,x\,(x+1)\,. \tag{19}$$

We choose again $n = 19$. The exact solution u and the discretization error $u - u_h$ are given in Figure 4; the discretization error $u - u_h$ is nonnegative. The squared energy and L_2 norms of the discretization error are equal to

$$\|(u-u_h)'\|^2 = 3.5000\text{e-}3 \quad \text{and} \quad \|u-u_h\|^2 = 8.7495\text{e-}7.$$

Table 2 and Figures 5–6 give results analogous to those presented above in Table 1 and Figures 2–3, respectively.

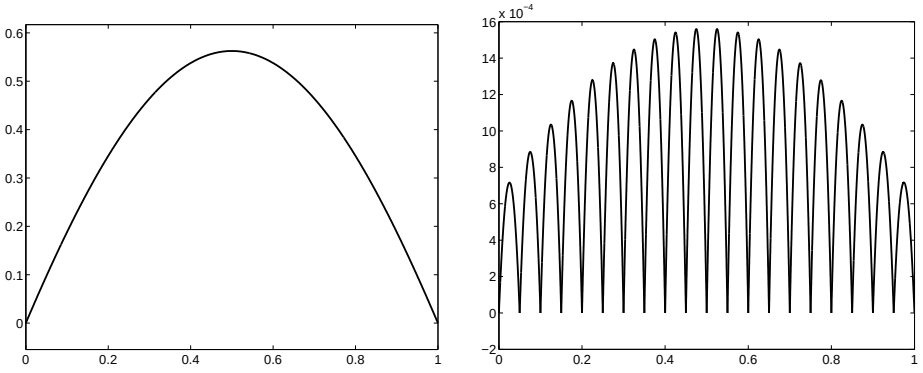

Figure 4: Left: the exact solution u (see (19)). Right: the discretization error $u - u_h$; the vertical axis is scaled by 10^{-4}.

3. Interpretation of the total error as a modification of the discretization mesh

As argued in [26, p. 9], it is desirable to interpret the inaccuracies in the solution process (including the algebraic errors) in terms of a meaningful modification of the mathematical model; see also [35, pp. 33–35]. This idea can be related to the so-called functional backward error by Arioli and others (see, e.g., [6]) where the errors are interpreted as (backward) perturbations of the weak formulation of the problem. This can be appealing in more complicated settings where such perturbation represents a modification of the mathematical model that has some physical interpretation. Within the simple problem setting considered above, an introduction of the functional backward error term counting for inaccurate solving of the discretized algebraic problem into the left-hand side of problem (2) would not satisfy this natural requirement. As pointed out in [6], in the simple case of the Poisson problem (or in similar cases where perturbation of the operator would be difficult to interpret), the operator structure can be preserved by restricting the perturbation to the right-hand side only. This can be relevant, e.g., when the right-hand side is dominated by experimental data and the perturbation is small enough in comparison with experimental

Table 2: Size of the algebraic and total error at several iteration steps for the exact solution (19).

k	$\|\mathbf{x}-\mathbf{x}_k\|_{\mathbf{A}}^2$	$\|\mathbf{x}-\mathbf{x}_k\|^2$	$\|(u-u_h^{(k)})'\|^2$	$\|u-u_h^{(k)}\|^2$
7	1.0112e-2	1.1899e-3	1.3612e-2	6.0367e-5
8	2.6905e-3	1.6856e-4	6.1905e-3	9.3021e-6
9	2.5563e-4	5.7123e-6	3.7556e-3	1.1605e-6
10	5.6776e-30	3.8081e-30	3.5000e-3	8.7495e-7

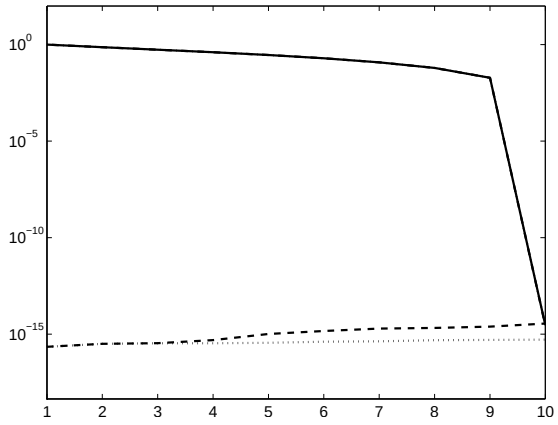

Figure 5: The relative $\mathbf{A}$-norm of the error $\|\mathbf{x}-\mathbf{x}_k\|_{\mathbf{A}}/\|\mathbf{x}-\mathbf{x}_0\|_{\mathbf{A}}$ (solid line), the loss of orthogonality in the standard CG implementation (dashed line) and the loss of orthogonality in the CG implementation with double reorthogonalized residuals (dotted line).

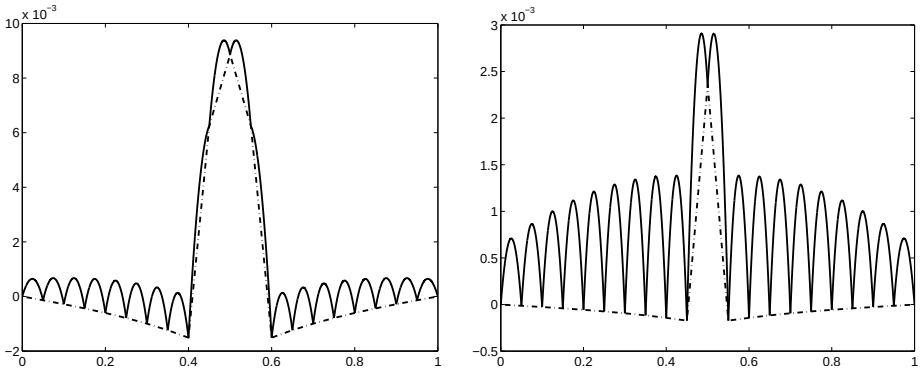

Figure 6: The algebraic error $u_h - u_h^{(k)}$ (dashed-dotted line) and the total error $u - u_h^{(k)}$ (solid line) at the 8th iteration (left) and at the 9th iteration (right); the vertical axes are scaled by 10^{-3}.

errors. In this paper we consider the change of the *discretization*, i.e. the basis functions or the mesh, as an alternative.

Interpreting the algebraic error as a transformation of the FEM basis has been considered in [21, Section 3]. We will use the idea from [21] but present the result in a slightly different way. Let the transformation of the basis $\Phi = [\phi_1, \ldots, \phi_n]$ (in our problem the basis of continuous piecewise linear hat functions) to the basis $\widehat{\Phi} = [\widehat{\phi}_1, \ldots, \widehat{\phi}_n]$ be represented by a square matrix $\mathbf{D} = [D_{\ell j}] \in \mathbb{R}^{n\times n}$,

$$\widehat{\phi}_j = \phi_j + \sum_{\ell=1}^{n} D_{\ell j}\, \phi_\ell\,, \quad j = 1, \ldots, n\,. \tag{20}$$

Please note that unlike the original FEM basis functions ϕ_j, the transformed basis functions $\widehat{\phi}_j\,, j = 1, \ldots, n$, need not be of a local support. The relation (20) can be written in the compact form as

$$\widehat{\Phi} = \Phi\,(\mathbf{I} + \mathbf{D})\,,$$

where $\mathbf{I} \in \mathbb{R}^{n\times n}$ denotes the identity matrix.

The transformation matrix $\mathbf{D}$ can be constructed in the following way. An easy calculation shows that an approximate solution $\widehat{\mathbf{x}} = [\widehat{\xi}_1, \ldots, \widehat{\xi}_n]^T$ of the algebraic system $\mathbf{A}\mathbf{x} = \mathbf{b}$ represents the *exact* solution of the perturbed system

$$(\mathbf{A} + \mathbf{E})\widehat{\mathbf{x}} = \mathbf{b}\,, \tag{21}$$

where

$$\mathbf{E} = \frac{(\mathbf{b} - \mathbf{A}\widehat{\mathbf{x}})\widehat{\mathbf{x}}^T}{\|\widehat{\mathbf{x}}\|^2}\,. \tag{22}$$

Let the Galerkin FEM solution u_h (see (4)–(8)) satisfy

$$u_h = \Phi\mathbf{x} = \sum_{j=1}^{n} \xi_j\, \phi_j = \sum_{j=1}^{n} \widehat{\xi}_j\, \widehat{\phi}_j = \widehat{\Phi}\widehat{\mathbf{x}} = \Phi\,(\mathbf{I} + \mathbf{D})\widehat{\mathbf{x}} \tag{23}$$

for some (unknown) matrix $\mathbf{D}$. Then, considering the Galerkin discretization of (2) with $u_h = \widehat{\Phi}\widetilde{\mathbf{x}}$, i.e. the discretization basis $\widehat{\phi}_1, \ldots, \widehat{\phi}_n$, and the test functions $\phi_1, \ldots, \phi_n$ gives

$$a(u_h, \phi_i) = \ell(\phi_i)\,, \quad i = 1, \ldots, n\,, \tag{24}$$

which can be formulated as the system of the linear algebraic equations

$$\widehat{\mathbf{A}}\widetilde{\mathbf{x}} = \mathbf{b},$$

where

$$\begin{aligned} \widehat{A}_{ij} &= a(\widehat{\phi}_j, \phi_i) = a(\phi_j + \sum_{\ell=1}^{n} D_{\ell j}\, \phi_\ell\,, \phi_i) \\ &= A_{ij} + \sum_{\ell=1}^{n} A_{i\ell} D_{\ell j}\,, \end{aligned} \tag{25}$$

i.e.

$$\widehat{\mathbf{A}} = \mathbf{A} + \mathbf{AD}. \tag{26}$$

Consequently, knowing the algebraic perturbation matrix $\mathbf{E}$ from (21), we can set

$$\mathbf{AD} = \mathbf{E}\,, \quad \text{giving} \quad \mathbf{D} = \mathbf{A}^{-1}\mathbf{E}\,, \tag{27}$$

with $\widehat{\mathbf{x}} = \widetilde{\mathbf{x}}$ the exact algebraic solution of (21) representing the Galerkin solution u_h of (2) in the sense of (24).

Remark 3. Since $\mathbf{E}$ is determined by the algebraic errors in solving $\mathbf{Ax} = \mathbf{b}$, we have no control of the sparsity of the transformation matrix $\mathbf{D} = \mathbf{A}^{-1}\mathbf{E}$, which is, in general, *dense*. Therefore the transformed basis functions $\widehat{\phi}_j$, $j = 1, \ldots, n$, have, in general, *global supports*. This holds also when $\mathbf{E}$ is determined using componentwise backward error with its structure of nonzeros entries determined, e.g., by the structure of nonzeros in $\mathbf{A}$. Since $\mathbf{A}^{-1}$ is, in general, dense, $\mathbf{D} = \mathbf{A}^{-1}\mathbf{E}$ is also dense.

When we set $\widehat{\mathbf{x}} = \mathbf{x}_8$ for our experimental illustration with the exact solution (16), the norms of the perturbation and transformation matrices are

$$\|\mathbf{E}\| = 3.2976\text{e-}1\,, \quad \|\mathbf{D}\| = 1.4674\text{e-}2\,.$$

Figure 7 gives the matrices $\mathbf{E}$ (see (22)) and $\mathbf{D}$ (see (27)) visualized using the MATLAB `surf` command. We can see the effect of the multiplication by $\mathbf{A}^{-1}$: the transformation matrix $\mathbf{D}$ has significantly more entries with the size far from zero than the perturbation matrix $\mathbf{E}$. It should be pointed out that our example is on purpose very simple and the mapping from $\mathbf{E}$ to $\mathbf{D} = \mathbf{A}^{-1}\mathbf{E}$ is for the given $\mathbf{A}$ rather benign (the norm $\|\mathbf{D}\|$ is even smaller than $\|\mathbf{E}\|$). In practical problems this may not be the case and $\mathbf{D}$ can have large nonzero elements. The left part of Figure 8 shows (for the same approximation $\widehat{\mathbf{x}} = \mathbf{x}_8$) the example of the transformed basis function $\widehat{\phi}_j$ (here $\widehat{\phi}_5$; see (20)). Since the entries of the matrix $\mathbf{D}$ are of the order 10^{-3}, $\widehat{\phi}_5$ looks visually the same as ϕ_5. The difference $\widehat{\phi}_5 - \phi_5$ is plotted in the right part of Figure 8. For other basis functions the situation is analogous. The size of the differences $\widehat{\phi}_j - \phi_j$, $j = 1, \ldots, n$, corresponds to the size of the algebraic error (as well as the discretization error when the algebraic and discretization errors are in balance).

When we consider the approximation $\widehat{\mathbf{x}} = \mathbf{x}_9$ given at the 9th CG iteration step, the norms of the corresponding perturbation and transformation matrices are

$$\|\mathbf{E}\| = 1.2976\text{e-}1\,, \quad \|\mathbf{D}\| = 2.4469\text{e-}3\,,$$

and the visualization of $\mathbf{E}, \mathbf{D}$ and the difference $\widehat{\phi}_j - \phi_j$, $j = 1, \ldots, n$, is analogous.

For the second example with the exact solution (19) and the approximation $\widehat{\mathbf{x}} = \mathbf{x}_9$ given at the 9th CG iteration step, the norms of the perturbation and transformation matrices are

$$\|\mathbf{E}\| = 6.8757\text{e-}2\,, \quad \|\mathbf{D}\| = 1.3220\text{e-}3\,.$$

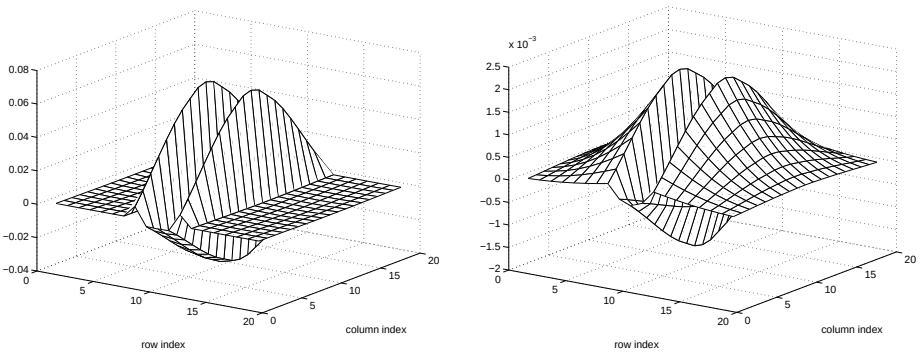

Figure 7: The perturbation matrix $\mathbf{E}$ (left) and the transformation matrix $\mathbf{D}$ (right) (with the entries visualized using the MATLAB `surf` command) for the approximation $\widehat{\mathbf{x}} = \mathbf{x}_8$ in the example with the exact solution (16). The right vertical axis is scaled by 10^{-3}.

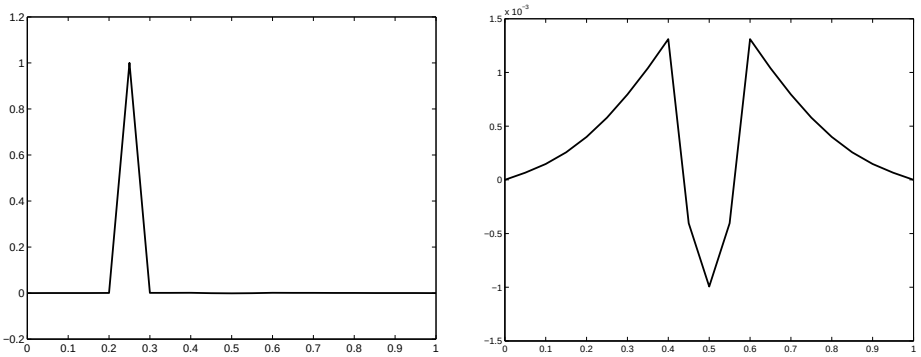

Figure 8: The transformed basis function $\widehat{\phi}_5$ (left) and the difference $\widehat{\phi}_5 - \phi_5$ (right) for the approximation $\widehat{\mathbf{x}} = \mathbf{x}_8$ in the example with the exact solution (16). For the other basis functions the situation is analogous. The right vertical axis is scaled by 10^{-3}; see the scale in the right part of Figure 1.

Figure 9 gives the matrix $\mathbf{E}$ and the matrix $\mathbf{D}$. For the transformed basis function $\widehat{\phi}_{11}$ and the difference $\widehat{\phi}_{11} - \phi_{11}$ see Figure 10.

In the rest of this section we interpret (with some unimportant inaccuracy) the total error $u - u_h^{(9)}$ for the last example (the exact solution u is given by (19) and $u_h^{(9)}$ is determined using the approximation $\mathbf{x}_9$ computed at the 9th CG step) as the discretization error $u - u_H$, where the Galerkin FEM solution u_H corresponds to a *new mesh* and new basis functions which *preserve the locality of their support*. We are aware that this interpretation is here specific for the one-dimensional problem as it is certainly not easily applicable in general, especially for higher-dimensional problems. However, the distortion of the mesh illustrated below shows the possible disturbing effects of the localization of the algebraic error.

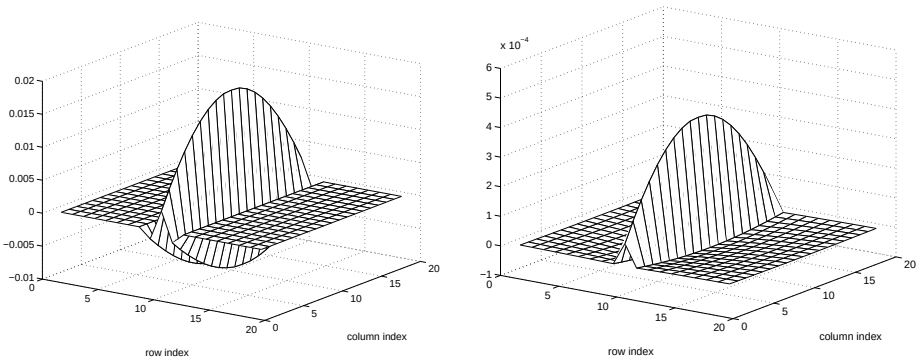

Figure 9: The perturbation matrix $\mathbf{E}$ (left) and the transformation matrix $\mathbf{D}$ (right) (with the entries visualized using the MATLAB `surf` command) for the approximation $\widehat{\mathbf{x}} = \mathbf{x}_9$ in the example with the exact solution (19). The right vertical axis is scaled by 10^{-4}.

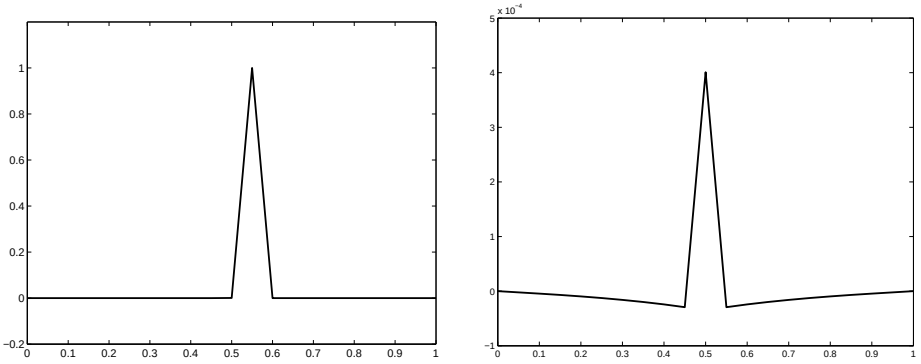

Figure 10: The transformed basis function $\widehat{\phi}_{11}$ (left) and the difference $\widehat{\phi}_{11} - \phi_{11}$ (right) for the approximation $\widehat{\mathbf{x}} = \mathbf{x}_9$ in the example with the exact solution (19). For the other basis functions the situation is analogous. The right vertical axis is scaled by 10^{-4}; see the scale in the right part of Figure 4.

The Galerkin FEM solution u_H coincides with the solution u at the nodes of the mesh; see [8, Corollary 4.1.1]. Therefore we construct the new mesh in such a way that the new nodes τ_i are given as the roots of the total error $u - u_h^{(9)}$ (i.e. the discretization error $u - u_H$) and therefore

$$u_H(\tau_i) = u(\tau_i) = u_h^{(9)}(\tau_i)\,.$$

In order to interpret the large total error in the middle of the interval as the discretization error, we replace (with no claim for optimality) the central node

0.5 of the original mesh by two nodes defined as $0.5 \pm 0.7h$, i.e.

$$\begin{aligned} \tau_i\,,\ i=1,\dots,18 &= \text{roots of } u-u_h^{(9)} \text{ for } 0<x<0.5\,,\\ \tau_{19} &= 0.5-0.7h\,,\\ \tau_{20} &= 0.5+0.7h\,,\\ \tau_i\,,\ i=21,\dots,38 &= \text{roots of } u-u_h^{(9)} \text{ for } 0.5<x<1\,. \end{aligned} \tag{28}$$

The new mesh now consists of $n = 38$ inner nodes, with 36 of them forming 18 close pairs. Please note that the new central element is 1.4 times longer than the elements in the original (uniform) mesh[4] , i.e. $\tau_{20} - \tau_{19} = 1.4\,h$.

Let $\psi_j,\ j = 1,\dots,n$, be the continuous piecewise linear FEM basis functions satisfying

$$\begin{aligned} \psi_j(\tau_j) &= 1\,,\\ \psi_j(x) &= 0\,,\quad 0\le x\le \tau_{j-1} \quad\text{and}\quad \tau_{j+1}\le x\le 1\,. \end{aligned}$$

As mentioned above, the Galerkin solution u_H coincides with the solution u at the nodes of the mesh. We can therefore write

$$u_H=\sum_{j=1}^{n}\xi_j\,\psi_j\,,\quad \xi_j=u(\tau_j)\,,\quad j=1,\dots,n\,.$$

The discretization error $u - u_H$ is nonnegative and the squared energy and L_2 norms of the discretization error $u - u_H$ are close to the analogous quantities for $u - u_h^{(9)}$,

$$\|(u-u_H)'\|^2 = 3.4224\text{e-}3 \quad \text{respectively} \quad \|u-u_H\|^2 = 9.8141\text{e-}7\,,$$

while

$$\|(u-u_h^{(9)})'\|^2 = 3.7556\text{e-}3 \quad \text{respectively} \quad \|u-u_h^{(9)}\|^2 = 1.1605\text{e-}6\,.$$

The comparison of the discretization error $u - u_H$ with the total error $u - u_h^{(9)}$ is given in the left part of Figure 11. With our choice of the nodes (28), the positive values of $u - u_h^{(9)}$ coincide, except for $\tau_{18} < x < \tau_{21}$, with the error $u - u_H$; see the detail of the comparison in the right part of Figure 11. There is a slight discrepancy between $u - u_H$ and $u - u_h^{(9)}$ for $\tau_{18} < x < \tau_{21}$.

Interpretation of the total error as the error of the *exact* discretized solution using a modified discretization mesh can rise, as illustrated above, interesting points. First, the algebraic error can be interpreted, in the sense described above, as the loss of locality of the support of the modified Galerkin basis functions. Second, the computed approximate solution $u_h^{(k)}$ which includes the

[4]This is the reason for denoting the Galerkin FEM solution corresponding to the new mesh with the subscript H commonly used for denoting the quantities corresponding to a coarser mesh.

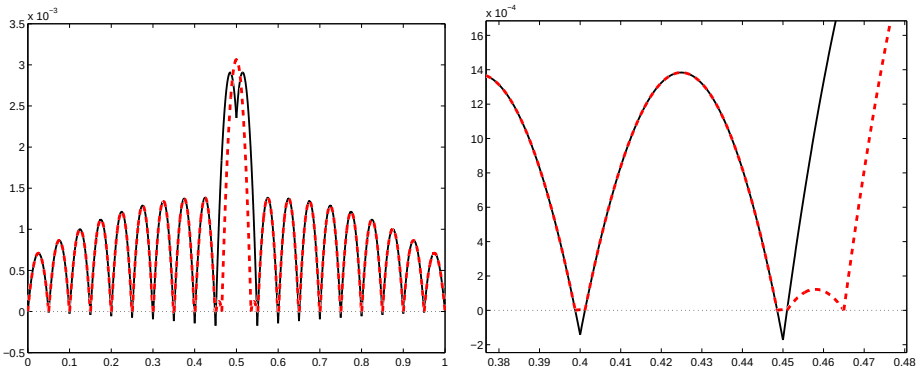

Figure 11: Left: the total error $u - u_h^{(9)}$ for the original mesh (solid line) and the discretization error $u - u_H$ on the modified mesh (dashed line); the vertical axis is scaled by 10^{-3}. Right: the detail showing the coincidence of the positive values of $u - u_h^{(9)}$ with $u - u_H$ for most of the interval and their slight discrepancy in the middle; the vertical axis is scaled by 10^{-4}.

error in the solution of the algebraic system can be interpreted (here with a small inaccuracy) as the discrete solution (with the vanishing algebraic error) for a mesh which can possibly have "holes" in the areas where the algebraic error is large (in our construction specific for the 1D problem the mesh has a "hole" in the center of the interval).

4. Spatial distribution of the error in CG computations

In this section we explain the behavior of the algebraic error observed above; see also [26, Section 5.9.4]. In the following we present the experimental illustration with the exact solution (16); see also Figures 7 and 8. The exposition uses the close relationship between CG and the Lanczos algorithm; for details see the original papers [23, 25] and also the survey [28].

Consider the spectral decomposition of the CG error at the kth step,

$$\mathbf{x} - \mathbf{x}_k = \sum_{i=1}^{n} (\mathbf{x} - \mathbf{x}_k, \mathbf{y}_i)\, \mathbf{y}_i \,, \tag{29}$$

where, as above, $\mathbf{y}_i$ denotes the ith normalized eigenvector of $\mathbf{A}$ corresponding to the eigenvalue λ_i ; see (11)-(12). We denote by $\theta_j^{(k)}$, $j = 1, \ldots, k$, the approximations of the eigenvalues of the matrix $\mathbf{A}$ (*Ritz values*) given at the kth iteration of the Lanczos algorithm applied to the matrix $\mathbf{A}$ and the starting vector $\mathbf{r}_0/\|\mathbf{r}_0\|$. Assuming exact arithmetic, a close approximation of the eigenvalue λ_i by a Ritz value $\theta_j^{(k)}$ means that the size of the ith component $|(\mathbf{x} - \mathbf{x}_k, \mathbf{y}_i)|$ of the error $\mathbf{x} - \mathbf{x}_k$ of the kth CG approximation in the direction $\mathbf{y}_i$ becomes small; see, e.g., [28, Theorem 3.3]. As mentioned above, the effect of rounding errors is in our example negligible. Consequently, the previous statement holds also for the presented results of finite precision computations.

Since some eigenvalues of $\mathbf{A}$ are approximated by Ritz values much faster than the others, this fact is reflected in the different behavior of the size of the spectral components $|(\mathbf{x}-\mathbf{x}_k, \mathbf{y}_i)|$, $i = 1, \ldots, n$, as k increases, $k = 0, 1, \ldots$. The individual eigenvectors $\mathbf{y}_i$ have different oscillating patterns; and therefore the individual spectral components of $\mathbf{x}-\mathbf{x}_k$ can develop in a rather nonuniform way as k increases. Using

$$u_h - u_h^{(k)} \quad = \quad \Phi(\mathbf{x}-\mathbf{x}_k) = \sum_{i=1}^{n} (\mathbf{x}-\mathbf{x}_k, \mathbf{y}_i)\, \Phi \mathbf{y}_i = \sum_{i=1}^{n} (\mathbf{x}-\mathbf{x}_k, \mathbf{y}_i)\, w_i\,,$$

this can result in a rather nonuniform spatial distribution of the algebraic (and the total) error in Ω. We will illustrate this situation in the following figures.

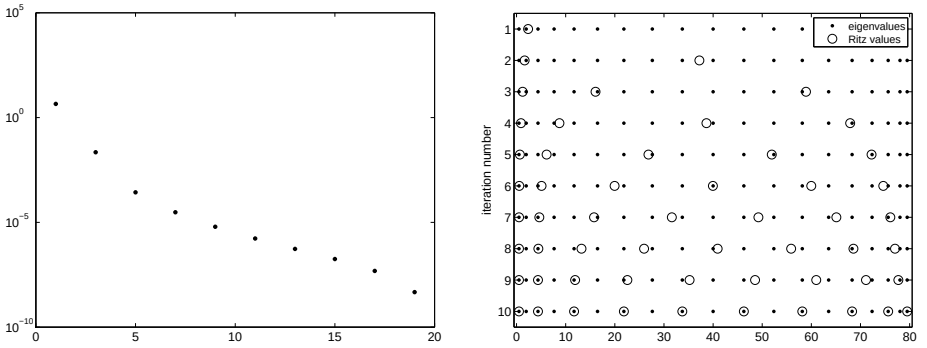

Figure 12: Left: the squared size of the spectral components $|(\mathbf{x}-\mathbf{x}_0, \mathbf{y}_i)|^2$, $i = 1, \ldots, n$, of the initial error $\mathbf{x}-\mathbf{x}_0$. Right: convergence of the Ritz values (circles) to the eigenvalues of $\mathbf{A}$ (dots) in iterations 1 through 10.

The squared size of the spectral components $|(\mathbf{x}-\mathbf{x}_0, \mathbf{y}_i)|^2$, $i = 1, \ldots, n$, of the initial error $\mathbf{x}-\mathbf{x}_0$ is given in the left part of Figure 12. Recall that $\mathbf{x}_0 = \mathbf{0}$ and therefore the initial error is equal to the solution $\mathbf{x}$. Since the solution is symmetric with respect to the center 0.5 of the given interval, the spectral components with even indices vanish (the corresponding projections computed in finite precision arithmetic are on the machine precision level). Since the initial error $\mathbf{x}-\mathbf{x}_0$ is smooth (i.e. nonoscillating), the components of the error with higher indices, which correspond to more oscillating eigenvectors (see (12)), significantly decrease with increasing index i. The Ritz values $\theta_j^{(k)}$, $j = 1, \ldots, k$, are for $k = 1, \ldots, 10$ given in the right part of Figure 12. The dots represent the eigenvalues of matrix $\mathbf{A}$. As expected, the Ritz values approximate the eigenvalues with odd indices. At the 10th iteration, all such eigenvalues are approximated, all components of the error $\mathbf{x}-\mathbf{x}_{10}$ become very small and the norm of the algebraic error drops to the machine precision level; see Figure 2 and Table 1. We can observe that the eigenvalues λ_1, λ_3 and partially also λ_5 are approximated much faster (for smaller iteration number) than the others.

In Figure 13 the development of the squared size of the spectral components of the algebraic error $\mathbf{x}-\mathbf{x}_k$ is shown for $k = 0, 7, 8, 9$ (only the values with odd

indices are plotted; the rest remain at the level 10^{-30}). We can see that the CG method reduces quickly the dominating spectral components of the error which corresponds to the fast approximation of the eigenvalues λ_1 and λ_3 by the Ritz values illustrated above. With increasing k the spectral components of $\mathbf{x} - \mathbf{x}_k$ almost equilibrate. As a consequence, the spatial distribution of the error $\mathbf{x} - \mathbf{x}_k$ changes as k increases and it eventually becomes highly nonuniform in the way substantially different than the spatial distribution of the initial error $\mathbf{x} - \mathbf{x}_0$.

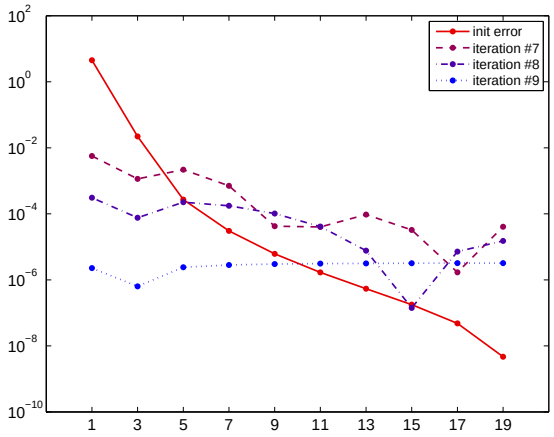

Figure 13: The development of the squared size of the spectral components of the algebraic error $|(\mathbf{x} - \mathbf{x}_k, \mathbf{y}_i)|^2$, $i = 1, 3, \ldots, 19$, for the iteration steps $k = 0, 7, 8, 9$ (solid, dashed, dashed-dotted and dotted lines respectively). We can observe equilibrating of the size of the spectral components as k increases.

This situation is illustrated in Figures 14 and 15, where we plot the most dominating approximations w_i to the eigenfunctions of the continuous operator (see (13) and (29)), corresponding to the initial error $\mathbf{x} - \mathbf{x}_0$ and to the error $\mathbf{x} - \mathbf{x}_9$ respectively. The right bottom part of Figure 14 shows the algebraic part of the initial error in the function space, which is given as the linear combination of the eigenfunction approximations with odd indices

$$u_h - u_h^{(0)} = \Phi(\mathbf{x} - \mathbf{x}_0) = \sum_{i=1}^{10} (\mathbf{x} - \mathbf{x}_0, \mathbf{y}_{2i-1})\, w_{2i-1}\,. \tag{30}$$

(As mentioned above, we use $\mathbf{x}_0 = \mathbf{0}$ and therefore $u_h - u_h^{(0)} = u_h$.) The right bottom part of Figure 15 shows the algebraic part of the error

$$u_h - u_h^{(9)} = \Phi(\mathbf{x} - \mathbf{x}_9) \approx \sum_{i=1}^{10} (\mathbf{x} - \mathbf{x}_9, \mathbf{y}_{2i-1})\, w_{2i-1}\,; \tag{31}$$

please compare with the algebraic error given in the right part of Figure 3. Here we neglect the spectral components of $\mathbf{x} - \mathbf{x}_9$ in the direction of even

eigenvectors of $\mathbf{A}$ which remain at the machine precision level (and therefore we use the approximation instead of the equality).

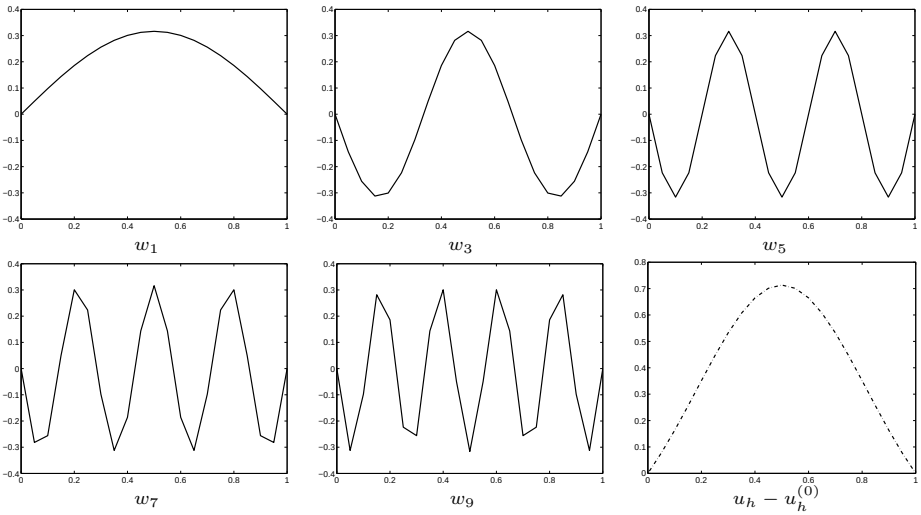

Figure 14: The approximate eigenfunctions w_i corresponding to the largest components of the initial algebraic error $\mathbf{x} - \mathbf{x}_0$ in the eigenvector basis of the matrix $\mathbf{A}$ and the algebraic part $u_h - u_h^{(0)}$ of the initial error $u - u_h^{(0)}$ (see (30)) (the dashed-dotted line in the right bottom part).

In the following remark we do not consider the effects of rounding errors (it can easily be shown that for the given point their effects are not important). Since the CG approximate solution $\mathbf{x}_k$ satisfies $\mathbf{x}_k \in \mathbf{x}_0 + \mathcal{K}_k(\mathbf{A}, \mathbf{r}_0)$, we have

$$\mathbf{x} - \mathbf{x}_k \in \mathbf{x} - \mathbf{x}_0 + \mathcal{K}_k(\mathbf{A}, \mathbf{r}_0)\,.$$

The highly irregular spatial distribution of $u_h - u_h^{(9)}$ observed above is caused by *eliminating (to some extent) the spectral components with slowly changing eigenvectors*, which dominate the initial error $u_h - u_h^{(0)}$. As we have seen, all spectral components eventually become almost equal in size and the effect of rapidly changing eigenvectors becomes pronounced. This cannot be explained as one may seemingly suggest and as we have several times experienced during the preparation of this paper, by adding an "oscillatory" vector from $\mathcal{K}_k(\mathbf{A}, \mathbf{r}_0)$ to $\mathbf{x} - \mathbf{x}_0$.

5. 2D illustrations

Using a simple 1D model problem, we illustrated above that the spatial distribution of the algebraic error can significantly differ from the spatial distribution of the discretization error. Because of its possibly large components in some parts of the domain, the algebraic error can determine the spatial distribution of the total error $u - u_h^{(k)}$ even when its globally measured size (here

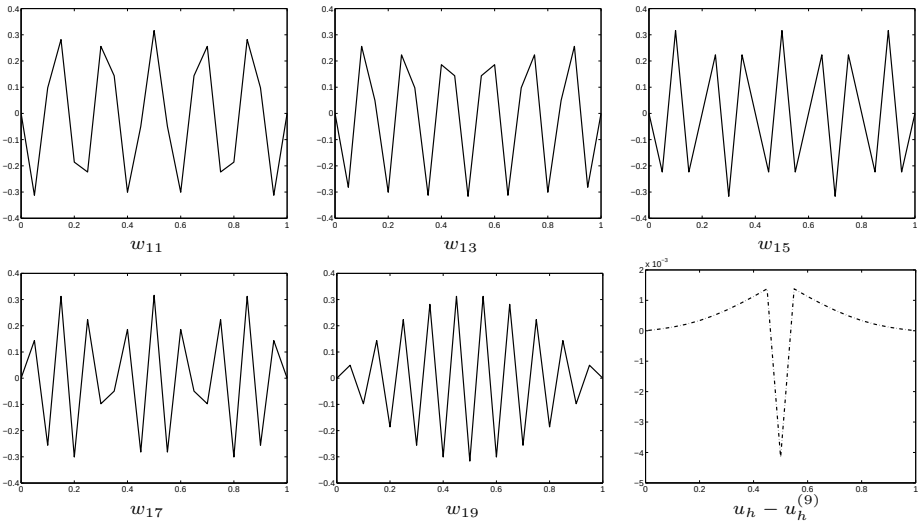

Figure 15: The approximate eigenfunctions w_i corresponding to the largest components of the algebraic error $\mathbf{x} - \mathbf{x}_9$ in the eigenvector basis of the matrix $\mathbf{A}$ and the algebraic part $u_h - u_h^{(9)}$ of the error $u - u_h^{(9)}$ (see (31)) (the dashed-dotted line in the right bottom part). The vertical axis in the right bottom part of the figure is scaled by 10^{-3}.

its energy norm) is smaller than the size of the discretization error. We emphasize that the described phenomenon is of general importance. It cannot be attributed to the specifics of the 1D model problem or the CG method used here for illustration. Of course, its appearance will be different for other problems or algebraic solvers.

In order to illustrate that the same phenomenon can appear also in more complicated settings, we present experiments using two well-known 2D model problems; see, e.g., [1, 27].

Peak problem: We consider the 2D Poisson boundary value problem

$$-\Delta u \quad = \quad f \quad \text{in } \Omega \equiv (0,1) \times (0,1)\,, \qquad u = 0 \quad \text{on } \partial\Omega\,. \tag{32}$$

The right-hand side f is chosen so that the solution u is given by

$$u(x,y) = x\,(x-1)\,y\,(y-1)\,\exp\left(-100\left(x-\frac{1}{2}\right)^2 - 100\left(y-\frac{117}{1000}\right)^2\right)\;; \tag{33}$$

see the upper left part of Figure 16.

L-shape problem: We consider the 2D Poisson boundary value problem

$$-\Delta u \quad = \quad 0 \quad \text{in } \Omega\,, \qquad u = u_D \quad \text{on } \partial\Omega\,, \tag{34}$$

where $\Omega \equiv (-1,1) \times (-1,1) \setminus (0,1) \times (-1,0)$. The Dirichlet boundary condition u_D is chosen so that the solution u is given in polar coordinates (r, θ) by

$$u(r,\theta) = r^{2/3} \sin\left(\frac{2}{3}\theta\right); \tag{35}$$

see the upper left part of Figure 17.

For each model problem we consider a sequence of partitions (meshes) of the domain Ω into the union of non-overlapping, triangular elements such that the non-empty intersection of a distinct pair of elements is a single common node or a single common edge. On a given mesh we discretize the problem, analogously to Section 2, using the piecewise affine finite elements with the basis given by the *hat-functions*, i.e. the piecewise affine functions such that each one corresponds to a node of the partition taking there value 1 and vanishing in all other nodes. The boundary condition u_D is approximated by a piecewise affine function given by the values of u_D in the boundary nodes. The stiffness matrix and right-hand side are assembled using the MATLAB code listed in [2].

Starting from the regular initial coarse mesh $\mathcal{T}_0$ consisting of 128 congruent triangles for the peak problem and of 192 congruent triangles for the L-shape problem, the sequence of adaptively refined meshes $\mathcal{T}_1, \mathcal{T}_2, \ldots$ is generated using the Adaptive Finite Element Method (AFEM). One iteration of AFEM can schematically be written as follows:

$$\text{SOLVE} \rightarrow \text{ESTIMATE} \rightarrow \text{MARK} \rightarrow \text{REFINE}$$

Here "SOLVE" means assembling and solving the system of the linear algebraic equations. We solve the systems using the MATLAB backslash operator that gives, for our experiments, sufficiently accurate approximations (i.e. approximations with a normwise relative backward error on the machine precision level). The corresponding piecewise affine approximations are denoted by u_h^*. "ESTIMATE" means the local a posteriori estimation of the error between the exact solution u and its numerical approximation u_h^*. We consider the residual-based local error estimator (indicator), for an element T of partition $\mathcal{T}_\ell$ and a piecewise affine approximation u_h^*

$$\eta_{R,T}^2(u_h^*) \equiv h_T^2 \|f\|_{L^2(T)}^2 + \sum_{E \subset \partial T} h_E \|[\nabla u_h^* \cdot n_E]\|_{L^2(E)}^2 , \tag{36}$$

where $h_T \equiv \operatorname{diam}(T)$ denotes the diameter of the element T, $h_E \equiv \operatorname{diam}(E)$ denotes the length of an edge $E \subset \partial T$, and $[\nabla u_h^* \cdot n_E]$ denotes the jump of piecewise constant function ∇u_h^* over edge E. In a comparison of 13 a posteriori error estimators on five benchmark problems, the estimator $\eta_{R,T}(u_h^*)$ was found appropriate for practical use in adaptive algorithms in [11, Section 8]. For marking the elements ("MARK") we consider the so-called *greedy algorithm*; see [11, Section 6]. Let the elements of $\mathcal{T}_\ell$ be enumerated such that $\eta_{R,T_1}(u_h^*) \geq \eta_{R,T_2}(u_h^*) \geq \cdots$ (this enumeration is used here for the sake of a full rigor; practical algorithms use techniques described in literature given below).

For a given $\Theta \in (0,1]$ we find the smallest index m such that

$$\Theta \sum_{T \in \mathcal{T}_\ell} \eta^2_{R,T}(u^*_h) \leq \sum_{j=1}^{m} \eta^2_{R,T_j}(u^*_h)\,;$$

see [12, Section 4.2] and for further development [37]. In the experiments we set $\Theta \equiv 0.25$. Finally, "REFINE" stands for the refinement of the elements $T_1, \ldots, T_m$ and the neighboring ones such that the conformity of the mesh is preserved. In the experiments we consider the refinement by Newest-Vertex-Bisection [29] implemented as in [17, Section 5.2].

For the first illustration we consider the peak model problem (32) and we use the mesh $\mathcal{T}_{13}$ given at the 13th AFEM iteration consisting of 1486 nodes. The tightly approximated squared energy norm of the discretization error (computed using the elementwise 16-node Gauss quadrature that is exact for polynomials up to degree 8; see, e.g., [13]) is equal to

$$\|\nabla(u - u^*_h)\|^2 = 9.5258\text{e-}6\,. \tag{37}$$

The discretization error $u - u^*_h$ visualized as a piecewise affine function (using the MATLAB `trisurf` command) is shown in the upper right part of Figure 16.

The linear algebraic system $\mathbf{A}\mathbf{x} = \mathbf{b}$ is of order 1436, which is equal to the number of the inner nodes in $\mathcal{T}_{13}$; the condition number is $\kappa(\mathbf{A}) = 1936.8$ (evaluated using the MATLAB `cond` function). Analogously to the 1D case, we apply the CG method with $\mathbf{x}_0 = 0$ to $\mathbf{A}\mathbf{x} = \mathbf{b}$. We stop at the iteration step $k = 67$ when the squared energy norm of the algebraic error $\|\mathbf{x}^* - \mathbf{x}_k\|^2_{\mathbf{A}}$ drops below one percent of the squared energy norm of the discretization error, i.e.

$$\|\mathbf{x}^* - \mathbf{x}_k\|^2_{\mathbf{A}} < 0.01\,\|\nabla(u - u^*_h)\|^2\,, \tag{38}$$

where $\mathbf{x}^*$ denotes the approximation to the solution $\mathbf{x}$ given by the MATLAB backslash operator. The criterion (38) is used here for a maximal rigor of our experimental illustrations. In practice a suitable approximation of $\mathbf{x}$ is not available, $\|\mathbf{x} - \mathbf{x}_k\|_{\mathbf{A}}$ is estimated in various ways and incorporating algebraic error estimates into a posteriori error analysis with using it for construction of algebraic stopping criteria requires substantial further investigation; see, e.g., [4, Section 4.1], [28, Section 5.3], [39, 40, 24, 3, 6, 5]. We denote by $u^{(k)}_h$ the piecewise affine approximation corresponding to the CG approximation $\mathbf{x}_k$. The squared energy norms of the algebraic error and the total error are equal to

$$\begin{aligned} \|\mathbf{x}^* - \mathbf{x}_k\|^2_{\mathbf{A}} &= 7.7295\text{e-}8\,, \\ \|\nabla(u - u^{(k)}_h)\|^2 &= 9.6018\text{e-}6\,. \end{aligned} \tag{39}$$

Please recall the corresponding energy norm of the discretization error (37) and see the equality (18). The norm of the total error $\|\nabla(u - u^{(k)}_h)\|^2$ is (tightly) approximated using elementwise the 16-node Gauss quadrature rule. As we can see in the bottom parts of Figure 16, the algebraic error $u^*_h - u^{(k)}_h$ substantially

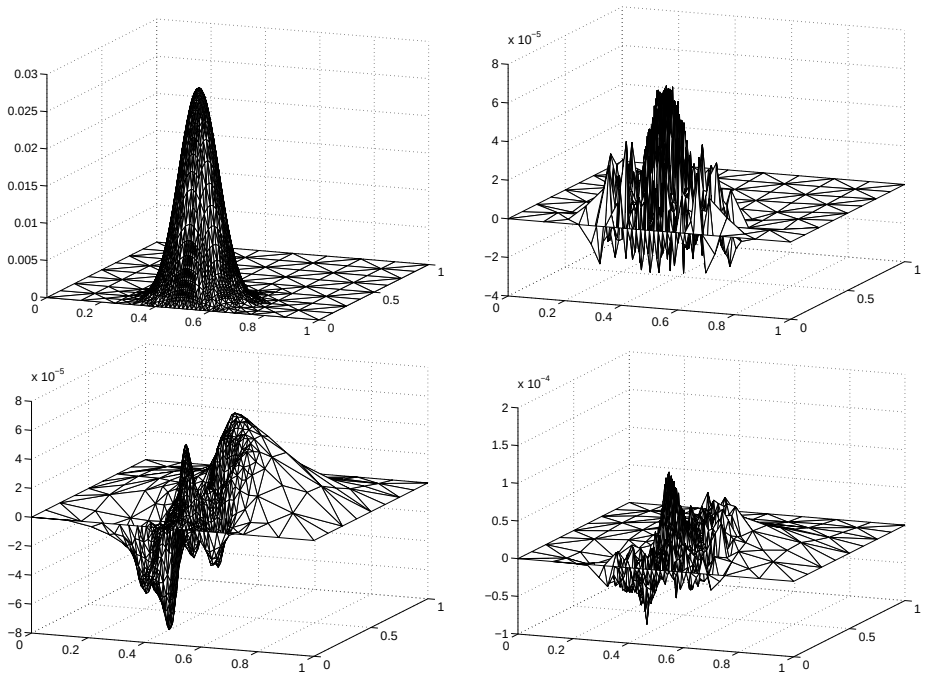

Figure 16: Peak model problem (32) solved using an adaptively refined mesh with 1486 nodes. Upper left: the solution u (33). Upper right: the discretization error $u - u_h^*$; vertical axis is scaled by 10^{-5}. Bottom left: the algebraic error $u_h^* - u_h^{(k)}$; vertical axis is scaled by 10^{-5}. Bottom right: the total error $u - u_h^{(k)}$; vertical axis is scaled by 10^{-4}. The functions are visualized as piecewise affine functions using the MATLAB `trisurf` command.

affects the shape of the total error $u - u_h^{(k)}$ in the part of the domain Ω where the solution u is (nearly) constant (with small gradients) as well as in the part where u has large gradients.

For the second illustration we consider the L-shape model problem (34) and we use the mesh $\mathcal{T}_{13}$ given at the 13th AFEM iteration consisting of 3376 nodes. The quantities analogous to those presented above in (37) and (39) are

$$\begin{aligned} \|\nabla(u - u_h^*)\|^2 &= 2.4512\text{e-}4\,, \\ \|\mathbf{x}^* - \mathbf{x}_k\|_{\mathbf{A}}^2 &= 2.3873\text{e-}6\,, \\ \|\nabla(u - u_h^{(k)})\|^2 &= 2.4751\text{e-}4\,. \end{aligned} \tag{40}$$

Here the system $\mathbf{Ax} = \mathbf{b}$ is of order 3210, and the condition number is $\kappa(\mathbf{A}) = 1230.3$ (evaluated using the MATLAB `cond` function). The stopping criterion (38) is satisfied at the iteration step $k = 107$. The piecewise affine visualization of the discretization error $u - u_h^*$ is given in the upper right part of Figure 17. As we can see in the bottom parts of Figure 17, the algebraic error $u_h^* - u_h^{(k)}$ substantially affects the shape of the total error $u - u_h^{(k)}$ in most of the domain Ω.

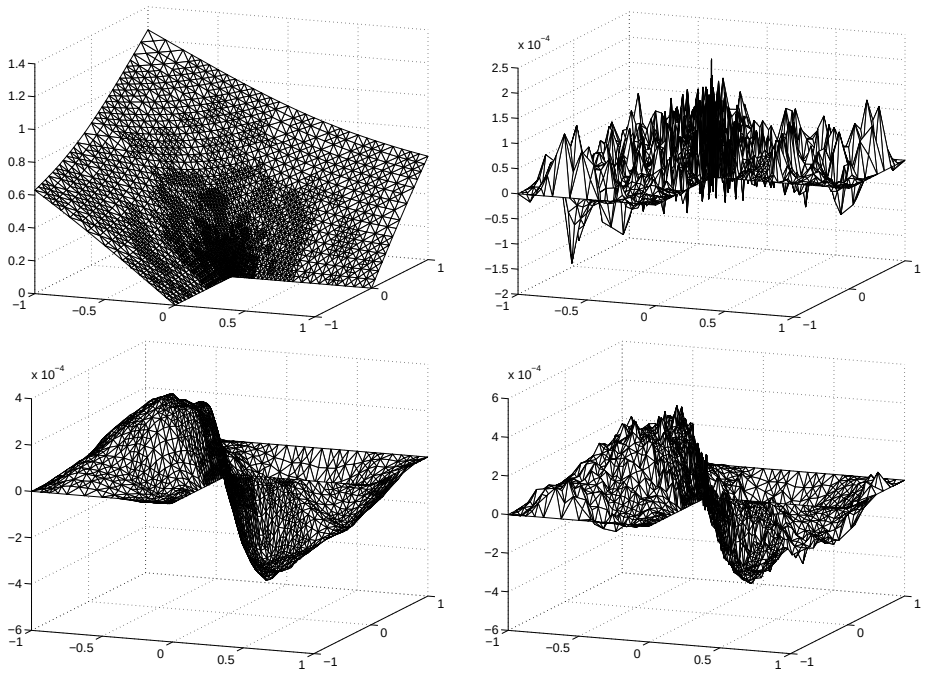

Figure 17: L-shape model problem (34) solved using an adaptively refined mesh with 3376 nodes. Upper left: the solution u (35). Upper right: the discretization error $u - u_h^*$; vertical axis is scaled by 10^{-4}. Bottom left: the algebraic error $u_h^* - u_h^{(k)}$; vertical axis is scaled by 10^{-4}. Bottom right: the total error $u - u_h^{(k)}$; vertical axis is scaled by 10^{-4}. The functions are visualized as piecewise affine functions using the MATLAB `trisurf` command.

6. Concluding remarks

The demonstrated difference between the spatial distributions of the algebraic and the discretization error across the domain (here obtained for the CG method) underlines the importance of constructing reliable stopping criteria for iterative algebraic solvers. In particular, in addition to evaluating parts of the error of different origin (discretization, inaccurate algebraic computations) in appropriate norms, such criteria should take into account spatial distribution of the total error in the function space. References to the work in this direction can be found in the recent survey [4]; see also, e.g., [24, Section 6] and [16]. One should also recall the goal-oriented adaptivity approach of Rannacher, Becker and their collaborators in the context of duality-based error control, which allows balancing discretization and iteration error in the problem-related areas of interest; see, e.g., the survey papers [33, 18] and the references given there. We believe that further developments focusing on the spatial distribution of the algebraic and total errors will be reported in the near future.

Acknowledgment: The authors are grateful to Mario Arioli for his thor-

ough and stimulating report, and to Vít Dolejší, Valeria Simoncini, Gerhard Starke, Endre Süli and Martin Vohralík for useful comments.

References

[1] M. Ainsworth, Robust a posteriori error estimation for nonconforming finite element approximation, SIAM J. Numer. Anal. 42 (2005) 2320–2341.

[2] J. Alberty, C. Carstensen, S.A. Funken, Remarks around 50 lines of Matlab: short finite element implementation, Numer. Algorithms 20 (1999) 117–137.

[3] M. Arioli, E.H. Georgoulis, D. Loghin, Stopping criteria for adaptive finite element solvers, SIAM J. Sci. Comput. 35 (2013) A1537–A1559.

[4] M. Arioli, J. Liesen, A. Międlar, Z. Strakoš, Interplay between discretization and algebraic computation in adaptive numerical solution of elliptic PDE problems, GAMM-Mitt. 36 (2013) 102–129.

[5] M. Arioli, D. Loghin, A.J. Wathen, Stopping criteria for iterations in finite element methods, Numer. Math. 99 (2005) 381–410.

[6] M. Arioli, E. Noulard, A. Russo, Stopping criteria for iterative methods: applications to PDE's, Calcolo 38 (2001) 97–112.

[7] I. Babuška, Numerical stability in problems of linear algebra, SIAM J. Numer. Anal. 9 (1972) 53–77.

[8] I. Babuška, T. Strouboulis, The finite element method and its reliability, Numerical Mathematics and Scientific Computation, Oxford University Press, New York, 2001.

[9] I. Babuška, T. Strouboulis, A. Mathur, C.S. Upadhyay, Pollution-error in the h-version of the finite-element method and the local quality of a posteriori error estimators, Finite Elem. Anal. Des. 17 (1994) 273–321.

[10] D. Boffi, Finite element approximation of eigenvalue problems, Acta Numer. 19 (2010) 1–120.

[11] C. Carstensen, C. Merdon, Estimator competition for Poisson problems, J. Comput. Math. 28 (2010) 309–330.

[12] W. Dörfler, A convergent adaptive algorithm for Poisson's equation, SIAM J. Numer. Anal. 33 (1996) 1106–1124.

[13] D.A. Dunavant, High degree efficient symmetrical Gaussian quadrature rules for the triangle, Internat. J. Numer. Methods Engrg. 21 (1985) 1129–1148.

[14] H.C. Elman, D.J. Silvester, A.J. Wathen, Finite elements and fast iterative solvers: with applications in incompressible fluid dynamics, Numerical Mathematics and Scientific Computation, Oxford University Press, New York, 2005.

[15] K. Eriksson, D. Estep, P. Hansbo, C. Johnson, Computational differential equations, Cambridge University Press, Cambridge, 1996.

[16] A. Ern, M. Vohralík, Adaptive inexact Newton methods with a posteriori stopping criteria for nonlinear diffusion PDEs, SIAM J. Sci. Comput. 35 (2013) A1761–A1791.

[17] S. Funken, D. Praetorius, P. Wissgott, Efficient implementation of adaptive P1-FEM in Matlab, Comput. Methods Appl. Math. 11 (2011) 460–490.

[18] M.B. Giles, E. Süli, Adjoint methods for PDEs: a posteriori error analysis and postprocessing by duality, Acta Numer. 11 (2002) 145–236.

[19] M.S. Gockenbach, Partial differential equations: analytical and numerical methods, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2002.

[20] M.S. Gockenbach, Understanding and implementing the finite element method, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2006.

[21] S. Gratton, P. Jiránek, X. Vasseur, Energy backward error: interpretation in numerical solution of elliptic partial differential equations and behaviour in the conjugate gradient method, Electron. Trans. Numer. Anal. 40 (2013) 338–355.

[22] A. Greenbaum, Z. Strakoš, Predicting the behavior of finite precision Lanczos and conjugate gradient computations, SIAM J. Matrix Anal. Appl. 13 (1992) 121–137.

[23] M.R. Hestenes, E. Stiefel, Methods of conjugate gradients for solving linear systems, J. Research Nat. Bur. Standards 49 (1952) 409–436.

[24] P. Jiránek, Z. Strakoš, M. Vohralík, A posteriori error estimates including algebraic error and stopping criteria for iterative solvers, SIAM J. Sci. Comput. 32 (2010) 1567–1590.

[25] C. Lanczos, Solution of systems of linear equations by minimized iterations, J. Research Nat. Bur. Standards 49 (1952) 33–53.

[26] J. Liesen, Z. Strakoš, Krylov subspace methods: Principles and analysis, Numerical Mathematics and Scientific Computation, Oxford University Press, Oxford, 2013.

[27] R. Luce, B.I. Wohlmuth, A local a posteriori error estimator based on equilibrated fluxes, SIAM J. Numer. Anal. 42 (2004) 1394–1414.

[28] G. Meurant, Z. Strakoš, The Lanczos and conjugate gradient algorithms in finite precision arithmetic, Acta Numer. 15 (2006) 471–542.

[29] P. Morin, R.H. Nochetto, K.G. Siebert, Convergence of adaptive finite element methods, SIAM Rev. 44 (2002) 631–658. Revised reprint of "Data oscillation and convergence of adaptive FEM" [SIAM J. Numer. Anal. 38 (2000) 466–488].

[30] A.E. Naiman, I. Babuška, H.C. Elman, A note on conjugate gradient convergence, Numer. Math. 76 (1997) 209–230.

[31] J.T. Oden, Y. Feng, Local and pollution error estimation for finite element approximations of elliptic boundary value problems, J. Comput. Appl. Math. 74 (1996) 245–293. TICAM Symposium (Austin, TX, 1995).

[32] A. Quarteroni, Numerical models for differential problems, volume 2 of *MS&A. Modeling, Simulation and Applications*, Springer-Verlag Italia, Milan, 2009. Translated from the 4th (2008) Italian edition by Silvia Quarteroni.

[33] R. Rannacher, A short course on numerical simulation of viscous flow: discretization, optimization and stability analysis, Discrete Contin. Dyn. Syst. Ser. S 5 (2012) 1147–1194.

[34] P.J. Roache, Verification and validation in computational science and engineering, Hermosa Publishers, Albuquerque, NM, 1998.

[35] P.J. Roache, Building PDE codes to be verifiable and validatable, Comput. Sci. Eng. 6 (2004) 30–38.

[36] V.V. Shaidurov, Some estimates of the rate of convergence for the cascadic conjugate-gradient method, Comput. Math. Appl. 31 (1996) 161–171. Selected topics in numerical methods (Miskolc, 1994).

[37] R. Stevenson, Optimality of a standard adaptive finite element method, Found. Comput. Math. 7 (2007) 245–269.

[38] Z. Strakoš, J. Liesen, On numerical stability in large scale linear algebraic computations, ZAMM Z. Angew. Math. Mech. 85 (2005) 307–325.

[39] Z. Strakoš, P. Tichý, On error estimation in the conjugate gradient method and why it works in finite precision computations, Electron. Trans. Numer. Anal. 13 (2002) 56–80.

[40] Z. Strakoš, P. Tichý, Error estimation in preconditioned conjugate gradients, BIT 45 (2005) 789–817.

[41] P.S. Vassilevski, Lecture Notes on Multigrid Methods, Technical Report LLNL-TR-439511, Lawrence Livermore National Laboratory, Livermore, CA, 2010.

[42] L.B. Wahlbin, Superconvergence in Galerkin finite element methods, volume 1605 of *Lecture Notes in Mathematics*, Springer-Verlag, Berlin, 1995.