On the Algebraic Error in Numerical Solution of Partial Differential Equations – Part I

Jan Papež[*]

Seminar on Numerical Analysis, January 25–29, 2021

[*] Institute of Mathematics of the CAS

Structure of the lectures

Part I Motivation, illustrations, and several topics related to the algebraic error and (inexact) numerical solution of PDEs

Part II "Estimating algebraic error using flux reconstructions"
→ construction of estimators that provide guaranteed upper bounds on the error, allow for local estimation, and involve no unknown constants

Acknowledgement

The presented results are joint work with

Zdeněk Strakoš
Martin Vohralík

Ani Anciaux-Sedrakian
Laura Grigori
Zakariae Jorti
Jörg Liesen
Ani Miraçi
Uli Rüde
Barbara Wohlmuth
Soleiman Yousef

Outline

Introduction, notation, and motivation

How algebraic error can look like

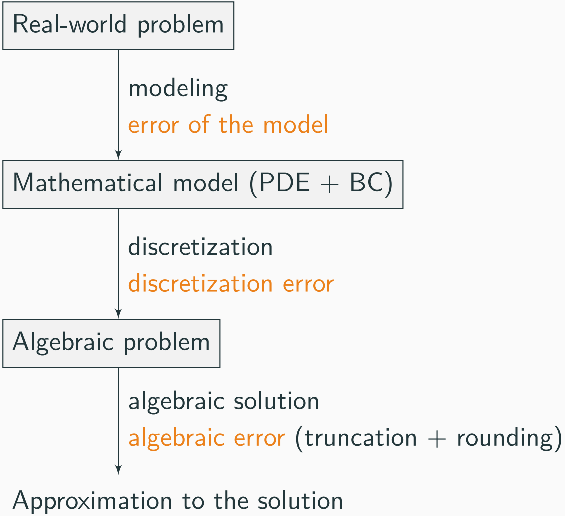
Algebraic error and residual-based error estimator

Preconditioning as transformation of the discretization basis

Backward interpretation of the algebraic error

Re-use of error estimators

Phases of the solution process in numerical PDEs



Real-world problem

modeling

error of the model

Mathematical model (PDE + BC)

discretization

discretization error

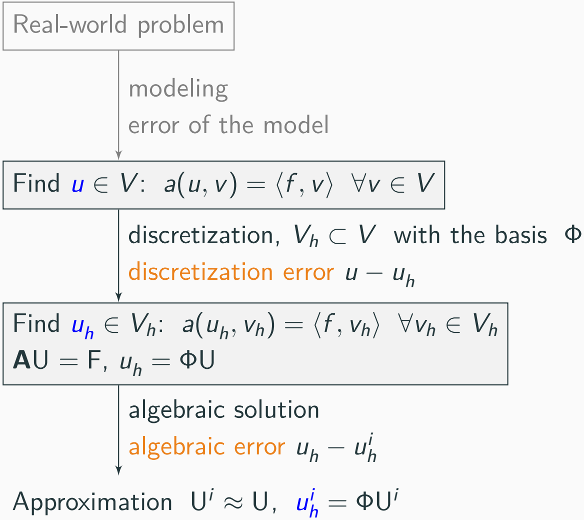
Algebraic problem

algebraic solution

algebraic error (truncation + rounding)

Approximation to the solution

Phases of the solution process in numerical PDEs



Real-world problem

modeling

error of the model

Find $u \in V$: $a(u,v) = \langle f, v \rangle \ \forall v \in V$

discretization, $V_h \subset V$ with the basis Φ

discretization error $u - u_h$

Find $u_h \in V_h$: $a(u_h, v_h) = \langle f, v_h \rangle \ \forall v_h \in V_h$

$\mathbf{A}\mathrm{U} = \mathrm{F}$, $u_h = \Phi \mathrm{U}$

algebraic solution

algebraic error $u_h - u_h^i$

Approximation $\mathrm{U}^i \approx \mathrm{U}$, $u_h^i = \Phi \mathrm{U}^i$

Solution is a two-way process

problem to solve (approximate)

properties

a priori information

validation of the result

Solution is a two-way process

problem to solve (approximate)

properties

a priori information

validation of the result

For example, for the algebraic solution phase:

$\mathbf{A}\mathrm{U} = \mathrm{F}$

$\mathbf{A}$ is SPD

we want to minimize the $\mathbf{A}$-norm of the error

Solution is a two-way process

problem to solve (approximate)
properties
a priori information

validation of the result

For example, for the algebraic solution phase:

$\mathbf{A}\mathrm{U} = \mathrm{F}$
$\mathbf{A}$ is SPD
we want to minimize the **A**-norm of the error

$\longrightarrow$ compute an approximation
(using a proper method, implementation) $\longrightarrow$

Solution is a two-way process

problem to solve (approximate)
properties
a priori information

validation of the result

For example, for the algebraic solution phase:

$\mathbf{A}\mathrm{U} = \mathrm{F}$
$\mathbf{A}$ is SPD
we want to minimize the **A**-norm of the error

computing error estimators
checking stop. criterion

$\longrightarrow$ compute an approximation (using a proper method, implementation) $\longrightarrow$

Solution is a two-way process

problem to solve (approximate)
properties
a priori information

validation of the result

For example, for the algebraic solution phase:

$\mathbf{A}\mathrm{U} = \mathrm{F}$
$\mathbf{A}$ is SPD
we want to minimize the $\mathbf{A}$-norm of the error
preconditioner

computing error estimators
checking stop. criterion

$\longrightarrow$ compute an approximation (using a proper method, implementation) $\longrightarrow$

Message of the lectures

- The algebraic error can substantially differ from the errors of other origin. In particular, its spatial distribution can be significantly different from the discretization error.
- For systems with a sparse matrix arising from FEM discretizations, the algebraic solution accounts for global interactions in the discretization domain.
- Theoretical results based on the assumption of exact algebraic solution should not be used for computed approximations. A derivation (or revision) of results that take into account inexact algebraic computations can be more difficult and/or the results might be weaker.
- An efficient solution procedure requires thorough understanding and interaction between all phases of the solution, such as discretization, preconditioning, algebraic solution, and error estimation.

For the sake of simplicity, we will for illustration (mostly) consider Poisson problem with homogeneous Dirichlet boundary condition

$$a(u, v) \equiv (\nabla u, \nabla v), \qquad V \equiv H_0^1(\Omega),$$

and conforming FEM discretization $V_h \subset V$ by continuous piecewise polynomial functions.

The errors then satisfy

$$\underbrace{u - u_h^i}_{\text{total error}} = \underbrace{u - u_h}_{\text{discretization error}} + \underbrace{u_h - u_h^i}_{\text{algebraic error}}$$

and

$$\|\nabla(u - u_h^i)\|^2 = \|\nabla(u - u_h)\|^2 + \|\nabla(u_h - u_h^i)\|^2.$$

How algebraic error can look like

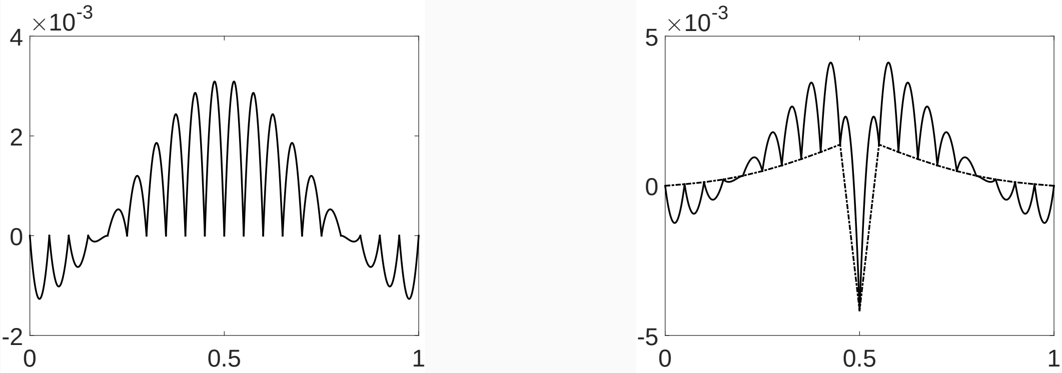
J. Papež, J. Liesen, Z. Strakoš:

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

Linear Algebra Appl. 449 (2014), pp. 89–114.

Spatial distribution of the errors of different origin (1D)

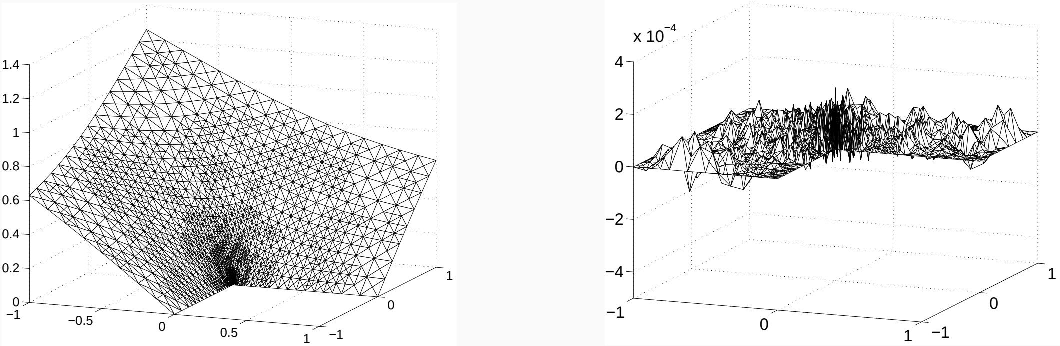
1D Poisson problem, uniform partition with 19 nodes, P1 FEM.



Left: discretization error $u - u_h$. Right: algebraic error $u_h - u_h^9$ (dashed-dotted line) and total error $u - u_h^9$ (solid line).

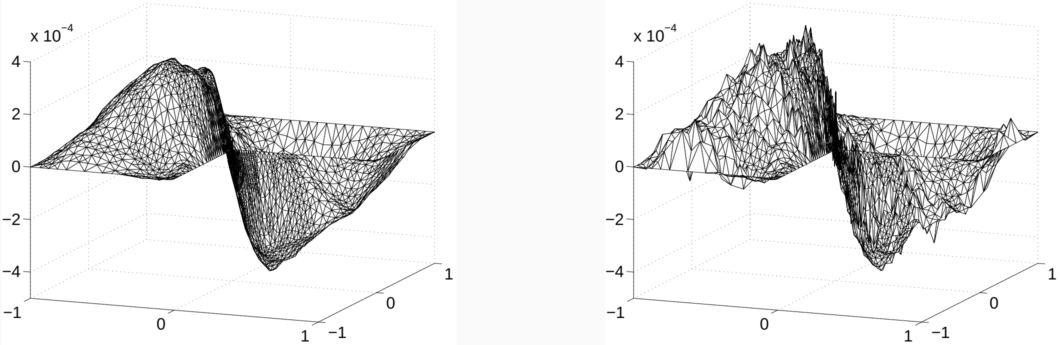
$$\|u_h - u_h^9\|_a = 1.23 \times 10^{-3} < 6.81 \times 10^{-3} = \|u - u_h\|_a$$

Spatial distribution of the errors of different origin (2D)



Exact solution u (left) and the discretization error $u - u_h$ (right) in the Poisson model problem on the L-shaped domain.

Spatial distribution of the errors of different origin (2D)



Algebraic error $u_h - u_h^i$ (left) and the total error $u - u_h^i$ (right). Here

$$\|\nabla(u_h - u_h^i)\| < 0.1\,\|\nabla(u - u_h)\|.$$

Algebraic error and residual-based error estimator

J. Papež and Z. Strakoš.
On a residual-based a posteriori error estimator for the total error.
IMA Journal of Numerical Analysis, 38(3):1164–1184, Sep 2017.

J. Papež.
***Algebraic Error in Matrix Computations in the Context of Numerical Solution of Partial Differential Equations*.**
PhD thesis, Charles University, Prague, November 2016.

Residual-based error estimator – notation

In this part, we consider a discretization using the piecewise affine conforming finite elements.

We denote by

- $\mathcal{T}_h$ the triangulation of Ω with the nodes $\mathcal{N}$ and edges $\mathcal{E}$,
- φ_z, $z \in \mathcal{N}$, the hat-function with the support ω_z (the patch).

Define the oscillations of the source term $f \in L^2(\Omega)$

$$\mathrm{osc} \equiv \Big(\sum_{z \in \mathcal{N}} |\omega_z| \, \|f - \mathrm{mean}(f, \omega_z)\|^2_{\omega_z} \Big)^{1/2},$$

and for $w_h \in V_h$ the edge residual measuring the jumps of a piecewise constant function ∇w_h over the inner edges

$$J(w_h) \equiv \Big(\sum_{E \in \mathcal{E} \setminus \partial\Omega} |E| \; \|[\nabla w_h \cdot n_E]\|^2_E \Big)^{1/2}.$$

Residual-based error estimator

For the Galerkin solution u_h there exists a factor $C > 0$ depending on the minimal angle of the triangulation such that

$$\|\nabla(u - u_h)\|^2 \leq C \left(J_h^2(u_h) + \mathrm{osc}^2 \right) \ ;$$

see, e.g., [Carstensen (1999)].

The proof uses the so-called Clément quasi-interpolation operator

$$\mathcal{I} : L^1(\Omega) \to V_h \, .$$

Bounding the total error

[Becker, Mao (2009), Lemma 3.1]:

$$\|\nabla(u - w_h)\|^2 \leq C\left(J_h^2(w_h) + \mathrm{osc}^2\right) + 2\,\|\nabla(u_h - w_h)\|^2\,.$$

Proof: “The upper bound with $w_h = u_h$ has been proven by [Carstensen (1999)] introducing a weighted Clément-type quasi-interpolation operator. The generalization to $w_h \neq u_h$ follows from the triangle inequality.”

Bounding the total error

[Becker, Mao (2009), Lemma 3.1]:

$$\|\nabla(u - w_h)\|^2 \leq C\left(J_h^2(w_h) + \mathrm{osc}^2\right) + 2\,\|\nabla(u_h - w_h)\|^2\,.$$

Proof: “The upper bound with $w_h = u_h$ has been proven by [Carstensen (1999)] introducing a weighted Clément-type quasi-interpolation operator. The generalization to $w_h \neq u_h$ follows from the triangle inequality.”

[Arioli, Georgoulis, Loghin (2013), proof of Theorem 3.3]:

$$\|\nabla(u - w_h)\|^2 \leq 2C_{2.2}\big(J_h^2(w_h) + \widetilde{\mathrm{osc}}^2\big) + (1 + 2C_{2.2}C_{3.1})\|\nabla(u_h - w_h)\|^2\,.$$

In the numerical experiments they empirically set $C_{2.2} := 40$, $C_{3.1} := 10$.

Revised bound

Elaborating on [Carstensen (1999)], we can show that

$$\|\nabla(u - w_h)\|^2 \leq C(J_h^2(w_h) + \mathrm{osc}^2) + 2\,\widetilde{C}^2_{\mathrm{intp}}(w_h)\,\|\nabla(u_h - w_h)\|^2 \,.$$

with

$$\widetilde{C}_{\mathrm{intp}}(w_h) \equiv \frac{\|\nabla(\mathcal{I}u - \mathcal{I}w_h)\|}{\|\nabla(u - w_h)\|} \,.$$

Revised bound

Elaborating on [Carstensen (1999)], we can show that

$$\|\nabla(u - w_h)\|^2 \leq C(J_h^2(w_h) + \mathrm{osc}^2) + 2\,\widetilde{C}_{\mathrm{intp}}^2(w_h)\,\|\nabla(u_h - w_h)\|^2\,.$$

with

$$\widetilde{C}_{\mathrm{intp}}(w_h) \equiv \frac{\|\nabla(\mathcal{I}u - \mathcal{I}w_h)\|}{\|\nabla(u - w_h)\|}\,.$$

A priori bound [Carstensen (1999), Theorem 3.1]:
There exists a factor $C_{\mathrm{intp}} > 0$ depending only on the triangulation $\mathcal{T}$ such that, for all $w \in H_0^1(\Omega)$,

$$\|\nabla\mathcal{I}w\| \leq C_{\mathrm{intp}}\|\nabla w\|\,.$$

This gives $C_{\mathrm{intp}} \geq \widetilde{C}_{\mathrm{intp}}(w_h)$, for any $w_h \in V_h$.

Solution-independent factor and overestimation

The factor C_{intp} represents the worst-case scenario and one may expect that most likely $C_{\mathrm{intp}} \gg \widetilde{C}_{\mathrm{intp}}(w_h)$.

Using the discussion in [Carstensen (2006), Section 2], for a square domain Ω, homogeneous Dirichlet BC and a shape-regular mesh, there holds

$$C_{\mathrm{intp}} \approx 6\,.$$

In general, “it may be very large for small angles in the triangulation”.

Numerical illustration

Poisson problem on the square $\Omega \equiv (-1,1) \times (-1,1)$, Delaunay triangulation with 1368 elements and with the minimal angle of the mesh equal to 35.9° (the average of the minimal angles of the elements is 50.3°). We recall, that in this setting $C_{\mathrm{intp}} \approx 6$.

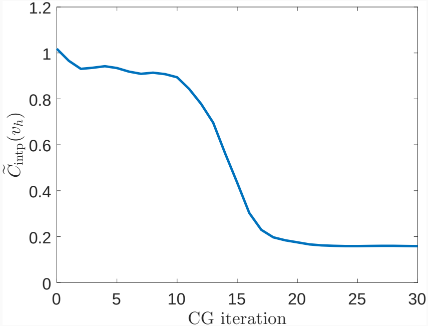
The exact solution is set as

$$u(x,y) = (x-1)(x+1)(y-1)(y+1)\,,$$

and we plot $\widetilde{C}_{\mathrm{intp}}(u_h^i)$ for the approximations u_h^i generated by the conjugate gradient method with zero initial vector for solving the discretized problem.

Numerical illustration

Poisson problem on the square $\Omega \equiv (-1,1) \times (-1,1)$, Delaunay triangulation with 1368 elements and with the minimal angle of the mesh equal to 35.9° (the average of the minimal angles of the elements is 50.3°). We recall, that in this setting $C_{\mathrm{intp}} \approx 6$.



The factor $\widetilde{C}_{\mathrm{intp}}(u_h^i)$ for the approximations u_h^i generated in the iterations of the conjugate gradient method.

Comments

The difference in the revised bound may seem for the given simple model problem only technical with preserving the structure of the estimate. Even here, the difference in the size of the multiplicative factors can be substantial.

Comments

The difference in the revised bound may seem for the given simple model problem only technical with preserving the structure of the estimate. Even here, the difference in the size of the multiplicative factors can be substantial.

There is, however, no guarantee, in general, that the structure of the estimates taking into account rigorously algebraic errors remains the same as the structure of the estimates based on the Galerkin orthogonality.

Moreover, providing a guaranteed and meaningful upper bound for the energy norm of the algebraic error is a highly nontrivial challenge.

Adaptive mesh refinement based on $\mathrm{EST}(u_h^i)$

For simplicity, we denote $\mathrm{EST}(w_h) \equiv (J_h^2(w_h) + \mathrm{osc}^2)^{1/2}$.

- $\mathrm{EST}(u_h)$ bounds the *discretization* error and allows its local estimation. The adaptive mesh refinement based on the associated error indicators has been studied and mathematically justified, e.g. in [Morin *et al.* (2002)].
- The *efficiency* of adaptive procedures based on $\mathrm{EST}(u_h^i)$ remains an open question. Does $\mathrm{EST}(u_h^i)$ indicate the parts of the computational domain where the discretization error is large?
- $\mathrm{EST}(w_h)$ can be evaluated locally. Algebraic error?

$$\|\nabla(u - u_h^i)\|^2 \leq C \cdot \mathrm{EST}^2(u_h^i) + C_{\mathrm{intp}} \|\nabla(u_h - u_h^i)\|^2 .$$

Adaptive mesh refinement based on EST(u_h^i)

Adaptive mesh refinement (1 step)

SOLVE → ESTIMATE → MARK → REFINE

Numerical experiment

We compare two sequences of meshes generated by AFEM:

1. In SOLVE, we compute the Galerkin solution u_h and refine the mesh using the estimator EST(u_h).

Adaptive mesh refinement based on EST(u_h^i)

Adaptive mesh refinement (1 step)

SOLVE → ESTIMATE → MARK → REFINE

Numerical experiment

We compare two sequences of meshes generated by AFEM:

1. In SOLVE, we compute the Galerkin solution u_h and refine the mesh using the estimator EST(u_h).

2. In SOLVE we compute (using CG) an approximation u_h^i with

$$\|\nabla(u_h - u_h^i)\|^2 \leq 0.01\|\nabla(u - u_h)\|^2$$

(here u_h in general differs from case 1. because the mesh can be different). Then we evaluate EST(u_h^i) and use it in marking and mesh refinement.

Adaptive mesh refinement based on EST(u_h^i)

Adaptive mesh refinement (1 step)

SOLVE → ESTIMATE → MARK → REFINE

Numerical experiment

We compare two sequences of meshes generated by AFEM:

1. In SOLVE, we compute the Galerkin solution u_h and refine the mesh using the estimator EST(u_h).

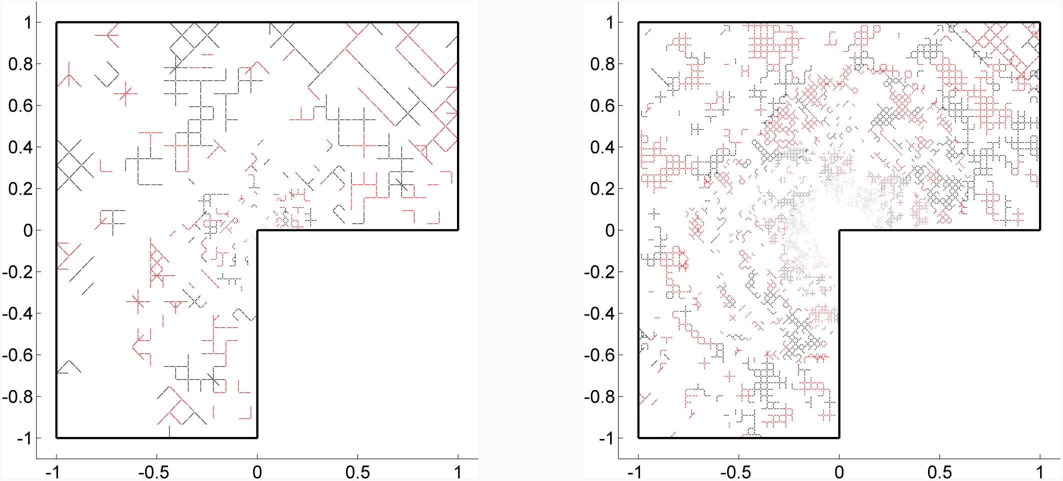
2. In SOLVE we compute (using CG) an approximation u_h^i with

$$\|\nabla(u_h - u_h^i)\|^2 \leq 0.01\|\nabla(u - u_h)\|^2$$

(here u_h in general differs from case 1. because the mesh can be different). Then we evaluate EST(u_h^i) and use it in marking and mesh refinement.

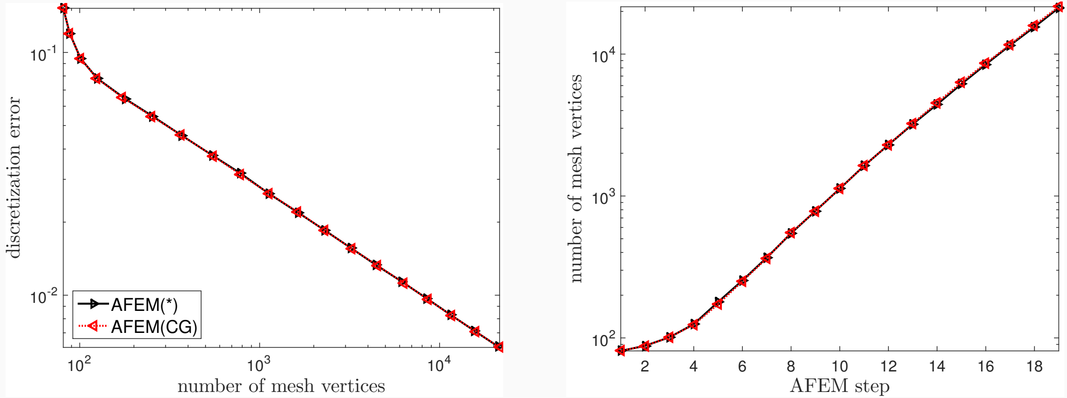
Finally, we compare the meshes after some number of AFEM steps and we plot the decrease of the discretization error for both sequences.

Adaptive mesh refinement based on EST(u_h^i)



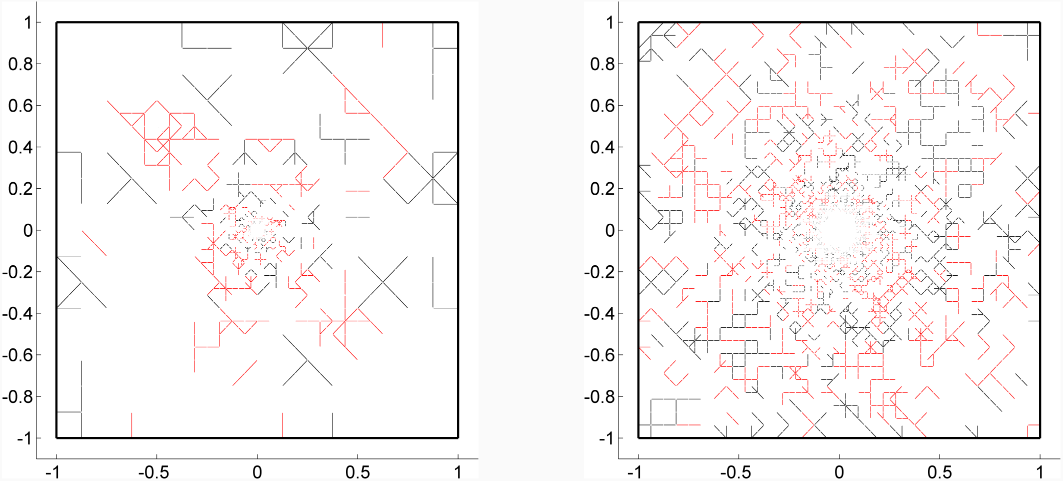
The difference in the adaptively refined meshes after 13 (left) and 19 (right) adaptive refinements.

Adaptive mesh refinement based on EST(u_h^i)



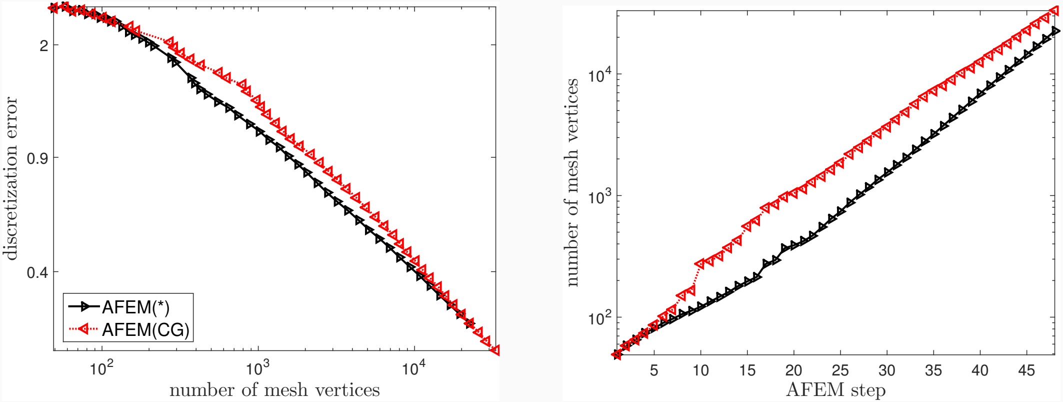
Left: the decrease of the discretization error norm in adaptive FEM that is based on EST(u_h) (black) and EST(u_h^i) (red), respectively. Right: the corresponding number of degrees of freedom in refinement steps.

Adaptive mesh refinement based on EST(u_h^i)



The difference in the adaptively refined meshes after 35 (left) and 47 (right) adaptive refinements

Adaptive mesh refinement based on EST(u_h^i)



Left: the decrease of the discretization error norm in adaptive FEM that is based on EST(u_h) (black) and EST(u_h^i) (red), respectively. Right: the corresponding number of degrees of freedom in refinement steps.

Preconditioning as transformation of the discretization basis

- J. Málek and Z. Strakoš.

  Preconditioning and the conjugate gradient method in the context of solving PDEs

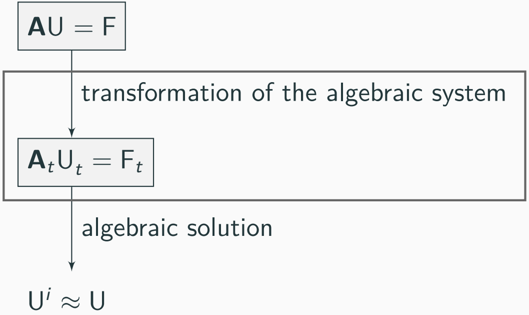
  volume 1 of *SIAM Spotlights*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2015.

- J. Papež.

  Algebraic Error in Matrix Computations in the Context of Numerical Solution of Partial Differential Equations.

  PhD thesis, Charles University, Prague, November 2016.

Algebraic preconditioning



$\mathbf{A}\mathsf{U} = \mathsf{F}$

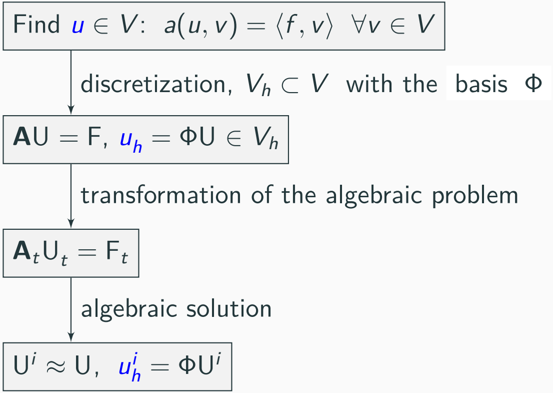
transformation of the algebraic system

$\mathbf{A}_t\mathsf{U}_t = \mathsf{F}_t$

algebraic solution

$\mathsf{U}^i \approx \mathsf{U}$

Preconditioning as transformation of the discretization basis



Find $u \in V$: $a(u,v) = \langle f, v \rangle \quad \forall v \in V$

discretization, $V_h \subset V$ with the basis Φ

$\mathbf{A}\mathrm{U} = \mathrm{F}$, $u_h = \Phi \mathrm{U} \in V_h$

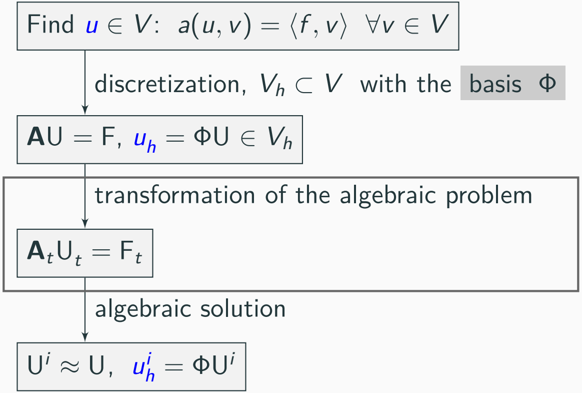
transformation of the algebraic problem

$\mathbf{A}_t \mathrm{U}_t = \mathrm{F}_t$

algebraic solution

$\mathrm{U}^i \approx \mathrm{U}$, $u_h^i = \Phi \mathrm{U}^i$

Preconditioning as transformation of the discretization basis



Find $u \in V$: $a(u, v) = \langle f, v \rangle \quad \forall v \in V$

discretization, $V_h \subset V$ with the basis Φ

$\mathbf{A}\mathrm{U} = \mathrm{F}$, $u_h = \Phi \mathrm{U} \in V_h$

transformation of the algebraic problem

$\mathbf{A}_t \mathrm{U}_t = \mathrm{F}_t$

algebraic solution

$\mathrm{U}^i \approx \mathrm{U}$, $u_h^i = \Phi \mathrm{U}^i$

Relationship between the preconditioning and the choice of the discretization basis Φ?

PCG in Hilbert space V

We will now briefly recall results of [Málek, Strakoš (2015)]. We will proceed as follows

1. given PDE in an operator equation, define PCG in (infinite-dimensional) Hilbert space V,
2. discretize V using $V_h \subset V$ with a basis Φ,
3. write the algebraic (finite-dimensional) formulation of PCG.

PCG in Hilbert space V

We will now briefly recall results of [Málek, Strakoš (2015)]. We will proceed as follows

1. given PDE in an operator equation, define PCG in (infinite-dimensional) Hilbert space V,
2. discretize V using $V_h \subset V$ with a basis Φ,
3. write the algebraic (finite-dimensional) formulation of PCG.

Then, we will see that

- this procedure gives us naturally a preconditioner $\mathbf{M}$,

PCG in Hilbert space V

We will now briefly recall results of [Málek, Strakoš (2015)]. We will proceed as follows

1. given PDE in an operator equation, define PCG in (infinite-dimensional) Hilbert space V,
2. discretize V using $V_h \subset V$ with a basis Φ,
3. write the algebraic (finite-dimensional) formulation of PCG.

Then, we will see that

- this procedure gives us naturally a preconditioner $\mathbf{M}$,
- the preconditioner $\mathbf{M}$, the inner product in V (or in V_h), and the choice of the discretization basis Φ are closely related.

Basic notation

V is a real (infinite dimensional) Hilbert space with the inner product

$$(\cdot,\cdot)_V \,:\, V \times V \to \mathbb{R}\,,$$

$V^{\#}$ is the dual space of bounded linear functionals on V with the duality pairing

$$\langle\cdot,\cdot\rangle \,:\, V^{\#} \times V \to \mathbb{R}\,.$$

PDE problem is described in the form of the functional equation

$$\mathcal{A}u = f\,, \quad \mathcal{A} : V \to V^{\#}, \quad u \in V, \quad f \in V^{\#} \qquad (*)$$

where $\mathcal{A}$ is linear, bounded, coercive, and self-adjoint w.r.t. the duality pairing $\langle\cdot,\cdot\rangle$. In our setting $\mathcal{A}u = a(u,\cdot)$.

Riesz map and operator preconditioning

For each $f \in V^{\#}$ there exists a unique $\tau f \in V$ such that

$$\langle f, v \rangle = (\tau f, v)_V \quad \text{for all} \quad v \in V \,.$$

In this way the inner product $(\cdot, \cdot)_V$ determines the Riesz map

$$\tau \,:\, V^{\#} \to V \,.$$

The transformation of (*) using the Riesz map gives

$$\tau \mathcal{A} u = \tau f \,, \qquad \tau \mathcal{A} \,:\, V \to V \,, \quad u \in V \,, \quad \tau f \in V \,,$$

which is called operator preconditioning. Key property: we can compute powers of $\tau\mathcal{A}$, which is needed to build Krylov subspaces.

Preconditioned CG in Hilbert spaces

$$r_0 = f - \mathcal{A}u_0 \in V^{\#}, \quad p_0 = \tau r_0 \in V$$

$$\begin{aligned}
u_n &= u_{n-1} + \alpha_{n-1} p_{n-1}\,, \\
&\qquad \alpha_{n-1} = \frac{\langle r_{n-1}, \tau r_{n-1} \rangle}{\langle \mathcal{A}p_{n-1}, p_{n-1} \rangle} = \frac{(\tau r_{n-1}, \tau r_{n-1})_V}{(\tau \mathcal{A}p_{n-1}, p_{n-1})_V}\,, \\
r_n &= r_{n-1} - \alpha_{n-1} \mathcal{A}p_{n-1}\,, \\
p_n &= \tau r_n + \beta_n p_{n-1}\,, \\
&\qquad \beta_n = \frac{\langle r_n, \tau r_n \rangle}{\langle r_{n-1}, \tau r_{n-1} \rangle} = \frac{(\tau r_n, \tau r_n)_V}{(\tau r_{n-1}, \tau r_{n-1})_V}\,.
\end{aligned}$$

The same formulas can be used for PCG in V_h.

Discretization and finite dimensional CG

$\Phi = \{\phi_1, \ldots, \phi_N\}$

basis of the finite-dimensional subspace $V_h \subset V$,

$\Phi^\# = \{\phi_1^\#, \ldots, \phi_N^\#\}$

canonical basis of the dual $V_h^\#$, $\Phi^\#\Phi = \mathbf{I}$.

Using the coordinates in Φ and in $\Phi^\#$,

$$\begin{aligned}
\langle f, v\rangle &= \langle \Phi^\# \mathsf{F}, \Phi \mathsf{V}\rangle &&= \mathsf{V}^* \mathsf{F}\,,\\
(u, v)_{V_h} &= (\Phi \mathsf{U}, \Phi \mathsf{V})_{V_h} &&= \mathsf{V}^* \mathbf{M} \mathsf{U}\,,\\
\mathcal{A}u &= \mathcal{A}\Phi\, \mathsf{U} &&= \Phi^\# \mathbf{A} \mathsf{U}\,,\\
\tau f &= \tau \Phi^\# \mathsf{F} &&= \Phi\, \mathbf{M}^{-1}\, \mathsf{F}\,;
\end{aligned}$$

where

$$\begin{aligned}
\mathbf{M} = [M_{ij}] &= [\,(\phi_j, \phi_i)_{V_h}\,]\,,\\
\mathbf{A} = [A_{ij}] &= [\,\langle \mathcal{A}\phi_j, \phi_i\rangle\,]\,, \qquad i, j = 1, \ldots, N\,.
\end{aligned}$$

Preconditioned algebraic CG

Preconditioned CG

With $f = \Phi^{\#}\,\mathsf{F}\,,\quad u_n = \Phi\,\mathsf{U}_n\,,\ p_n = \Phi\,\mathsf{P}_n\,,\ r_n = \Phi^{\#}\,\mathsf{R}_n$ we get the standard preconditioned algebraic CG with the preconditioner $\mathbf{M}$.

Unpreconditioned CG is in this setting an oxymoron!

Unpreconditioned CG, i.e. $\mathbf{M} = \mathbf{I}$, corresponds to the basis Φ orthonormal w.r.t. the inner product $(\cdot,\cdot)_{V_h}$.

Orthogonalization of the discretization basis

Consider the decomposition $\mathbf{M} = \mathbf{L}\mathbf{L}^*$, then the transformed discretization basis $\Phi_t = \Phi\,(\mathbf{L}^*)^{-1}$ is orthonormal w.r.t. $(\cdot,\cdot)_{V_h}$. Indeed,

$$(\Phi_t,\Phi_t)_{V_h} = \mathbf{L}^{-1}(\Phi,\Phi)_{V_h}(\mathbf{L}^*)^{-1} = \mathbf{L}^{-1}\mathbf{M}(\mathbf{L}^*)^{-1} = \mathbf{I}.$$

Interpretation of an algebraic preconditioning

Natural question: can we proceed the opposite way, starting from *algebraic* PCG and *arbitrary* SPD preconditioner $\widehat{\mathbf{M}}$?

Interpretation of an algebraic preconditioning

Natural question: can we proceed the opposite way, starting from *algebraic* PCG and *arbitrary* SPD preconditioner $\widehat{\mathbf{M}}$?

For the algebraic preconditioning with $\widehat{\mathbf{L}}\widehat{\mathbf{L}}^* = \widehat{\mathbf{M}} \neq \mathbf{M}$, the (transformed) discretization basis $\widehat{\Phi} = \Phi\,(\widehat{\mathbf{L}}^*)^{-1}$ is not orthonormal w.r.t. $(\cdot,\cdot)_{V_h}$.

In order to obtain the interpretation of the algebraic preconditioning $\widehat{\mathbf{M}}$ as the transformation of the basis $\Phi \to \widehat{\Phi}$, we have to change also the inner product in V_h:

$$(u,v)_{V_h} = (\Phi\,\mathrm{U}, \Phi\,\mathrm{V})_{V_h} = \mathrm{V}^*\mathbf{M}\mathrm{U}\,,$$

has to be replaced by

$$(u,v)_{\mathrm{new},V_h} = (\widehat{\Phi}\,\widehat{\mathrm{U}}, \widehat{\Phi}\,\widehat{\mathrm{V}})_{\mathrm{new},V_h} \equiv \widehat{\mathrm{V}}^*\widehat{\mathrm{U}} = \mathrm{V}^*\widehat{\mathbf{M}}\mathrm{U}\,.$$

Observations

Algebraic preconditioning associated with the operator preconditioning is equivalent to the orthogonalization of the discretization basis in the given finite-dimensional Hilbert space V_h.

Algebraic preconditioning can be interpreted as transformation of the discretization basis and, at the same time, transformation of the inner product in V_h such that the transformed basis $\widehat{\Phi}$ is orthonormal with respect to the transformed inner product.

Observations

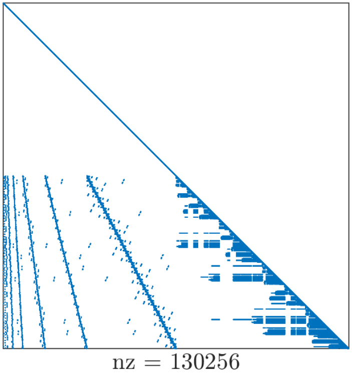
Algebraic preconditioning associated with the operator preconditioning is equivalent to the orthogonalization of the discretization basis in the given finite-dimensional Hilbert space V_h.

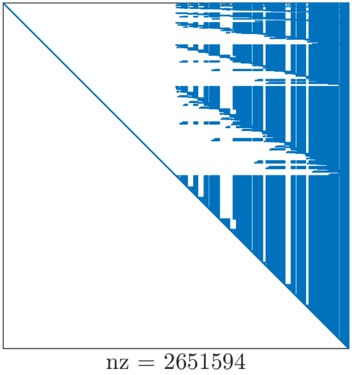
Algebraic preconditioning can be interpreted as transformation of the discretization basis and, at the same time, transformation of the inner product in V_h such that the transformed basis $\widehat{\Phi}$ is orthonormal with respect to the transformed inner product.

Even for a sparse preconditioner $\mathbf{M} = [\,(\phi_j, \phi_i)_{V_h}\,]$, the inverse $\mathbf{L}^{-1}$ of its Cholesky factor is typically dense. Therefore, the transformed (orthogonalized) basis is of global support.

Sparsity of Cholesky factors

An example of Cholesky factor **L** of the preconditioner and its transposed inverse $(\mathbf{L}^*)^{-1}$, taken from [P. 2016] - problem with inhomogeneous diffusion tensor, uniform mesh, Laplace preconditioner.





Backward interpretation of the algebraic error

J. Papež.

***Algebraic Error in Matrix Computations in the Context of Numerical Solution of Partial Differential Equations*.**

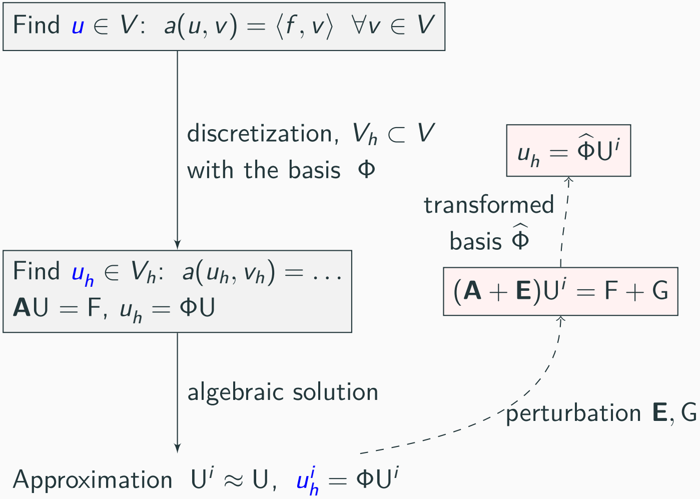
PhD thesis, Charles University, Prague, November 2016.

Backward interpretation of the algebraic error

$$\begin{tikzcd}
\boxed{\mathbf{A}\mathrm{U} = \mathrm{F}} \arrow[d, "\text{algebraic solution}"] & \boxed{(\mathbf{A}+\mathbf{E})\mathrm{U}^i = \mathrm{F}+\mathrm{G}} \\
\mathrm{U}^i \approx \mathrm{U} \arrow[ur, dashed, bend right, "\text{perturbation } \mathbf{E}, \mathrm{G} \text{ of the original data}"'] &
\end{tikzcd}$$

We interpret the algebraic backward errors $\mathbf{E}, \mathrm{G}$ within the other phases of the solution process.

Backward interpretation of the algebraic error



Find $u \in V$: $a(u, v) = \langle f, v \rangle \ \forall v \in V$

discretization, $V_h \subset V$

with the basis Φ

Find $u_h \in V_h$: $a(u_h, v_h) = \dots$

$\mathbf{A}\mathrm{U} = \mathrm{F}$, $u_h = \Phi \mathrm{U}$

algebraic solution

Approximation $\mathrm{U}^i \approx \mathrm{U}$, $u_h^i = \Phi \mathrm{U}^i$

perturbation $\mathbf{E}, \mathrm{G}$

$(\mathbf{A} + \mathbf{E})\mathrm{U}^i = \mathrm{F} + \mathrm{G}$

transformed basis $\widehat{\Phi}$

$u_h = \widehat{\Phi} \mathrm{U}^i$

Algebraic backward error

Let the computed algebraic vector $\widehat{\mathrm{U}}$ that approximates U solve the perturbed system

$$(\mathbf{A} + \mathbf{E})\,\widehat{\mathrm{U}} = \mathrm{F} + \mathrm{G}\,.$$

Our aim is to interpret the perturbations $\mathbf{E}, \mathrm{G}$ as transformations of the discretization bases.

One can consider possibly different transformations of the discretization and the test bases

$$\begin{aligned} \Psi &= \Phi\,(\mathbf{I} + \mathbf{D})\,, \\ \mathcal{X} &= \Phi\,(\mathbf{I} + \mathbf{H})\,. \end{aligned}$$

Transformation of the discretization bases

Let the Galerkin solution $u_h = \Phi\,\mathrm{U}$

$$a(u_h, \phi_i) \;=\; (f, \phi_i), \qquad i = 1, \ldots, N$$

can be expressed as the Galerkin solution $u_h = \Psi\,\widehat{\mathrm{U}} = \Phi\,(\mathbf{I} + \mathbf{D})\,\widehat{\mathrm{U}}$ of the discrete system with the transformed bases

$$a(u_h, \chi_i) \;=\; (f, \chi_i), \qquad i = 1, \ldots, N\,,$$

that results in the linear algebraic system

$$\widetilde{\mathbf{A}}\widehat{\mathrm{U}} = \widetilde{\mathrm{F}}\,, \qquad \widetilde{A}_{ij} \;=\; a(\psi_j, \chi_i)\,, \quad \widetilde{F}_i = (f, \chi_i)\,,$$

where $\widetilde{\mathbf{A}} = (\mathbf{I} + \mathbf{H})^T\mathbf{A}\,(\mathbf{I} + \mathbf{D})$ and $\widetilde{\mathrm{F}} = (\mathbf{I} + \mathbf{H})^T\mathrm{F}\,.$

Interpretation of the algebraic error

The computed approximation $\widehat{\mathrm{U}}$ solves the system $\widetilde{\mathbf{A}}\widehat{\mathrm{U}} = \widetilde{\mathrm{F}}$ exactly. The algebraic error is absorbed in the transformed bases $\Psi, \mathcal{X}$.

Identifying the perturbations in

$$(\mathbf{A} + \mathbf{E})\,\widehat{\mathrm{U}} = \mathrm{F} + \mathrm{G}$$

with transformation of the discretization bases gives

$$\begin{aligned} \mathbf{A} + \mathbf{E} &= (\mathbf{I} + \mathbf{H})^T \mathbf{A}\,(\mathbf{I} + \mathbf{D})\,, \\ \mathrm{F} + \mathrm{G} &= (\mathbf{I} + \mathbf{H})^T \mathrm{F}\,. \end{aligned}$$

Three classes of backward errors

- general case, $\mathbf{E} \neq 0$, $\mathrm{G} \neq 0$
- symmetric case, $\mathbf{A} + \mathbf{E}$ is SPD and $\mathbf{D} = \mathbf{H}$
- no perturbation of rhs, $\mathrm{G} = 0$

Interpretation of the algebraic error

The computed approximation $\widehat{\mathrm{U}}$ solves the system $\widetilde{\mathbf{A}}\widehat{\mathrm{U}} = \widetilde{\mathrm{F}}$ exactly. The algebraic error is absorbed in the transformed bases $\Psi, \mathcal{X}$.

Identifying the perturbations in

$$(\mathbf{A} + \mathbf{E})\,\widehat{\mathrm{U}} = \mathrm{F} + \mathrm{G}$$

with transformation of the discretization bases gives

$$\begin{aligned} \mathbf{A} + \mathbf{E} &= (\mathbf{I} + \mathbf{H})^T \mathbf{A}\,(\mathbf{I} + \mathbf{D})\,, \\ \mathrm{F} + \mathrm{G} &= (\mathbf{I} + \mathbf{H})^T \mathrm{F}\,. \end{aligned}$$

Three classes of backward errors

- general case, $\mathbf{E} \neq 0$, $\mathrm{G} \neq 0$ $\quad\rightarrow$ not unique transformation
- symmetric case, $\mathbf{A} + \mathbf{E}$ is SPD and $\mathbf{D} = \mathbf{H}$
- no perturbation of rhs, $\mathrm{G} = 0$

Interpretation of the algebraic error

The computed approximation $\widehat{\mathrm{U}}$ solves the system $\widetilde{\mathbf{A}}\widehat{\mathrm{U}} = \widetilde{\mathrm{F}}$ exactly. The algebraic error is absorbed in the transformed bases $\Psi, \mathcal{X}$.

Identifying the perturbations in

$$(\mathbf{A} + \mathbf{E})\,\widehat{\mathrm{U}} = \mathrm{F} + \mathrm{G}$$

with transformation of the discretization bases gives

$$\begin{aligned} \mathbf{A} + \mathbf{E} &= (\mathbf{I} + \mathbf{H})^T \mathbf{A}\,(\mathbf{I} + \mathbf{D})\,, \\ \mathrm{F} + \mathrm{G} &= (\mathbf{I} + \mathbf{H})^T \mathrm{F}\,. \end{aligned}$$

Three classes of backward errors

- general case, $\mathbf{E} \neq 0$, $\mathrm{G} \neq 0$ → not unique transformation
- symmetric case, $\mathbf{A} + \mathbf{E}$ is SPD and $\mathbf{D} = \mathbf{H}$ → does not exist in general
- no perturbation of rhs, $\mathrm{G} = 0$

Interpretation of the algebraic error

The computed approximation $\widehat{\mathrm{U}}$ solves the system $\widetilde{\mathbf{A}}\widehat{\mathrm{U}} = \widetilde{\mathrm{F}}$ exactly. The algebraic error is absorbed in the transformed bases $\Psi, \mathcal{X}$.

Identifying the perturbations in

$$(\mathbf{A} + \mathbf{E})\,\widehat{\mathrm{U}} = \mathrm{F} + \mathrm{G}$$

with transformation of the discretization bases gives

$$\begin{aligned} \mathbf{A} + \mathbf{E} &= (\mathbf{I} + \mathbf{H})^T \mathbf{A}\,(\mathbf{I} + \mathbf{D})\,, \\ \mathrm{F} + \mathrm{G} &= (\mathbf{I} + \mathbf{H})^T \mathrm{F}\,. \end{aligned}$$

Three classes of backward errors

- general case, $\mathbf{E} \neq 0$, $\mathrm{G} \neq 0$ → not unique transformation
- symmetric case, $\mathbf{A} + \mathbf{E}$ is SPD and $\mathbf{D} = \mathbf{H}$ → does not exist in general
- no perturbation of rhs, $\mathrm{G} = 0$ → will be illustrated now

Backward error with $\mathrm{G} = 0$

For $\mathrm{G} = 0$ it is natural to set $\mathbf{H} = 0$, i.e. to consider the original test functions. This case was considered in [Gratton, Jiránek, Vasseur (2013)]; [P., Liesen, Strakoš (2014)].

From $\mathbf{A} + \mathbf{E} = \mathbf{A}\,(\mathbf{I} + \mathbf{D})$ we have $\mathbf{AD} = \mathbf{E}$ and $\mathbf{D} = \mathbf{A}^{-1}\mathbf{E}$.

The transformed basis $\Psi = \Phi\,(\mathbf{I} + \mathbf{D})$ has global support (**D** is dense)!

Global support of the transformed basis

1D Poisson model problem,

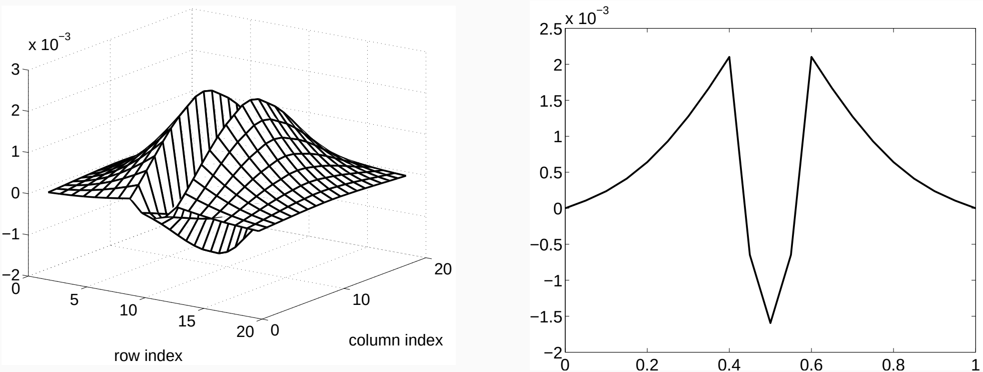
$$(\mathbf{A} + \mathbf{E})\,\widehat{\mathrm{U}} = \mathrm{F} \qquad (\text{i.e.}\ \ \mathrm{G} = 0)\,.$$

- $$\mathbf{E} = (\mathrm{F} - \mathbf{A}\widehat{\mathrm{U}})\,\frac{\widehat{\mathrm{U}}^T}{\|\widehat{\mathrm{U}}\|_2^2}\,, \qquad \text{then}\ \ \mathbf{D} = \mathbf{A}^{-1}\mathbf{E} = (\mathrm{U} - \widehat{\mathrm{U}})\,\frac{\widehat{\mathrm{U}}^T}{\|\widehat{\mathrm{U}}\|_2^2}\,,$$
- symmetric perturbation $\mathbf{E}_{\text{sym}}$ with the minimal Frobenius norm

  $$\mathbf{E}_{\text{sym}} = \arg\min\left\{\|\mathbf{E}\|_F \quad \middle| \quad \mathbf{E} = \mathbf{E}^T,\ (\mathbf{A} + \mathbf{E})\,\widehat{\mathrm{U}} = \mathbf{b}\right\}\,;$$

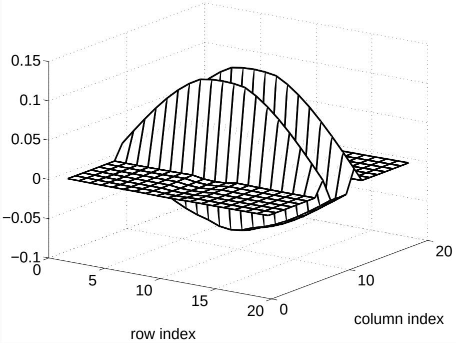
  see [Bunch, Demmel, van Loan (1989)].

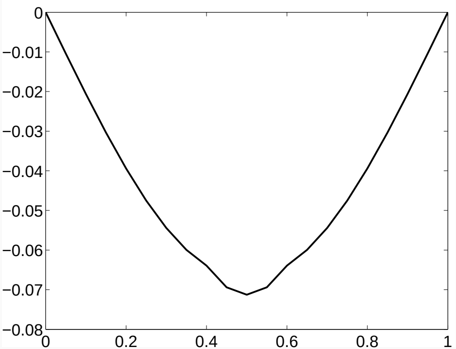
Transformed basis



MATLAB `surf` plot of the transformation matrix $\mathbf{D} = \mathbf{A}^{-1}\mathbf{E}$ (left) and the difference $\psi_j - \phi_j$ (right).

Transformed basis for symmetric perturbation matrix





MATLAB `surf` plot of the transformation matrix $\mathbf{D} = \mathbf{A}^{-1}\mathbf{E}_{\text{sym}}$ (left) and the difference $\psi_j - \phi_j$ (right).

Please note that $\|\mathbf{A}^{-1}\mathbf{E}_{\text{sym}}\| \gg \|\mathbf{A}^{-1}\mathbf{E}\|$!

Re-use of error estimators

A. Miraçi, J. Papež, and M. Vohralík.
A Multilevel Algebraic Error Estimator and the Corresponding Iterative Solver with p-Robust Behavior.
SINUM, 58(5):2856–2884, 2020.

A. Anciaux-Sedrakian, L. Grigori, Z. Jorti, J. Papež, and S. Yousef.
Adaptive solution of linear systems of equations based on a posteriori error estimators.
Numerical Algorithms, 84(1):331–364, 2020.

Lower bound on the algebraic error

The algebraic error in Poisson problem satisfy

$$\|\nabla(u_h - u_h^i)\| = \sup_{v_h \in V_h, \|\nabla v_h\| = 1} (\nabla(u_h - u_h^i), \nabla v_h)$$

$$= \sup_{v_h \in V_h, \|\nabla v_h\| = 1} \{(f, v_h) - (\nabla u_h^i, \nabla v_h)\}$$

and the supremum is attained for $v_h = (u_h - u_h^i) \,/\, \|\nabla(u_h - u_h^i)\|$.

Lower bound on the algebraic error

The algebraic error in Poisson problem satisfy

$$\|\nabla(u_h - u_h^i)\| = \sup_{v_h \in V_h, \|\nabla v_h\|=1} (\nabla(u_h - u_h^i), \nabla v_h)$$

$$= \sup_{v_h \in V_h, \|\nabla v_h\|=1} \{(f, v_h) - (\nabla u_h^i, \nabla v_h)\}$$

and the supremum is attained for $v_h = (u_h - u_h^i) \,/\, \|\nabla(u_h - u_h^i)\|$.

A simple lower bound can be computed as

$$\mu(w_h) := \frac{|(f, w_h) - (\nabla u_h^i, \nabla w_h)|}{\|\nabla w_h\|}.$$

Clearly, $\mu(w_h) \approx \|\nabla(u_h - u_h^i)\|$ iff $w_h \approx C(u_h - u_h^i)$.

Lower bound on the algebraic error, cont.

Given the algebraic residual R^i (which is the only quantity we have), we compute its lifting, $\rho_h^i \in V_h$, $\rho_h^i = \rho_h^i(\mathrm{R}^i)$, and estimate the error using $\mu(\rho_h^i)$.

Lower bound on the algebraic error, cont.

Given the algebraic residual R^i (which is the only quantity we have), we compute its lifting, $\rho_h^i \in V_h$, $\rho_h^i = \rho_h^i(\mathrm{R}^i)$, and estimate the error using $\mu(\rho_h^i)$.

However, when ρ_h^i is computed, we can *also* use it to define a new approximation (or consider ρ_h^i as a preconditioned residual).

$\Rightarrow$ the lifting ρ_h^i can be used to relate

- error estimator (lower bound) $\mu(\rho_h^i)$,
- algebraic solver, $u_h^{i+1} := u_h^i + \alpha\rho_h^i$
- preconditioner " $\mathrm{R}^i \mapsto \rho_h^i$ ",

and their properties can be studied together.

Lower bound on the algebraic error, cont.

Given the algebraic residual R^i (which is the only quantity we have), we compute its lifting, $\rho_h^i \in V_h$, $\rho_h^i = \rho_h^i(\mathrm{R}^i)$, and estimate the error using $\mu(\rho_h^i)$.

However, when ρ_h^i is computed, we can *also* use it to define a new approximation (or consider ρ_h^i as a preconditioned residual).

$\Rightarrow$ the lifting ρ_h^i can be used to relate

- error estimator (lower bound) $\mu(\rho_h^i)$,
- algebraic solver, $u_h^{i+1} := u_h^i + \alpha\rho_h^i$
- preconditioner " $\mathrm{R}^i \mapsto \rho_h^i$ ",

and their properties can be studied together.

For example, we studied the robustness of the error estimator and the algebraic solver with respect to the polynomial degree of the FEM approximation.

Adaptive preconditioner based on local error indicators

Motivation

When the algebraic error (and its local distribution) is estimated, can we do anything to speed-up the following algebraic computation?

Analogy in the discretization phase: adaptive mesh refinement

Adaptive preconditioner based on local error indicators

Motivation

When the algebraic error (and its local distribution) is estimated, can we do anything to speed-up the following algebraic computation?

Analogy in the discretization phase: adaptive mesh refinement

Crucial difference

(In our setting) the discretization error can be bounded by the interpolation error, while the algebraic error is typically of a global origin.

I am not aware of any procedure to reduce the algebraic error *locally* $\Rightarrow$ only heuristic approaches.

Adaptive preconditioner based on local error indicators

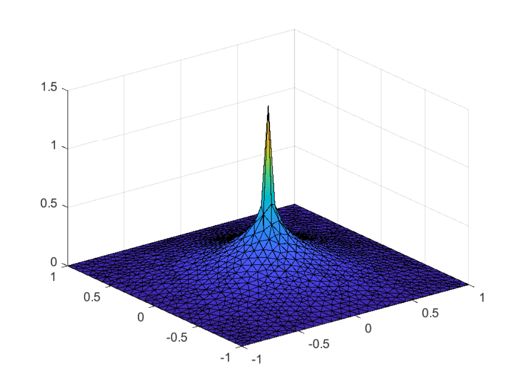
Our procedure

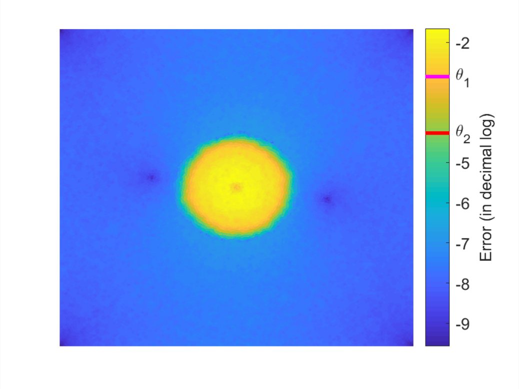
1. start PCG with a given preconditioner
2. at some iteration, evaluate (local) error indicators and identify the domain Ω_1 with the set L of the degrees of freedom, where the algebraic error is (expected to be) large
3. "treat" the part of the matrix associated with L:
   - Schur complement approach (leads to solving a smaller but possibly dense system)
   - build a new preconditioner and a new initial guess
4. continue PCG iterations (either for Schur system or for the original system with the new preconditioner and the initial guess)

Here "treat" means that we assure that the residual associated to DOFs in L vanishes in the subsequent iterations.

The procedure requires the inversion (or Cholesky factors) of the block $\mathbf{A}_L$ - the part of the original matrix associated with L.

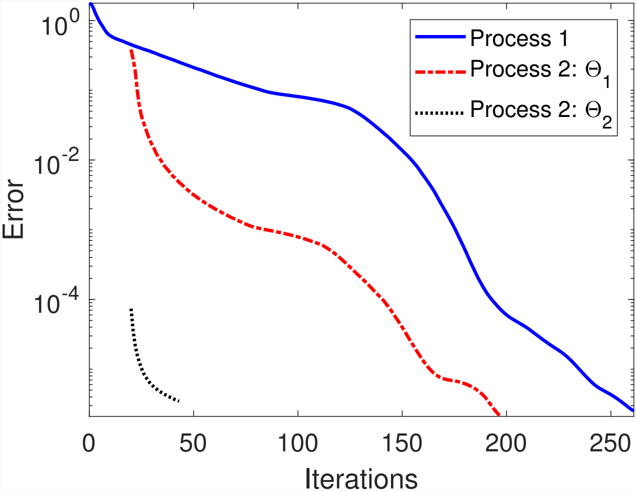
Adaptive preconditioner - num. experiment I





Galerkin solution, algebraic error (energy norm on the elements) after 20 initial PCG iterations

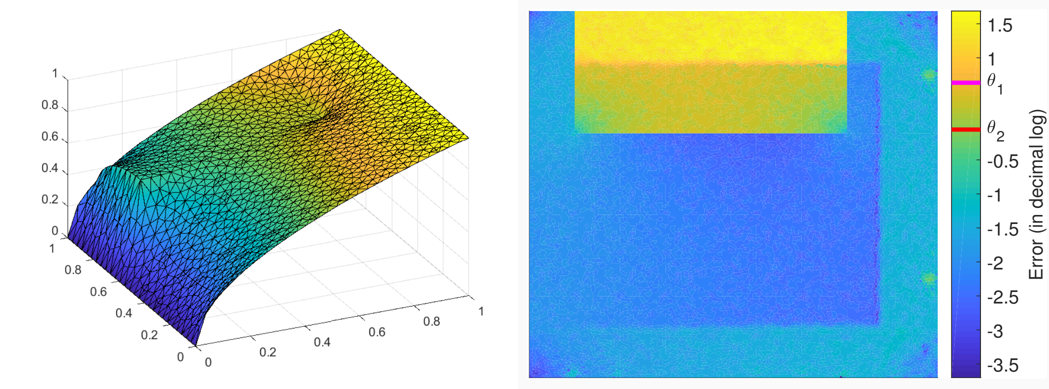
Adaptive preconditioner - num. experiment I



Convergence of the energy norm of the algebraic error

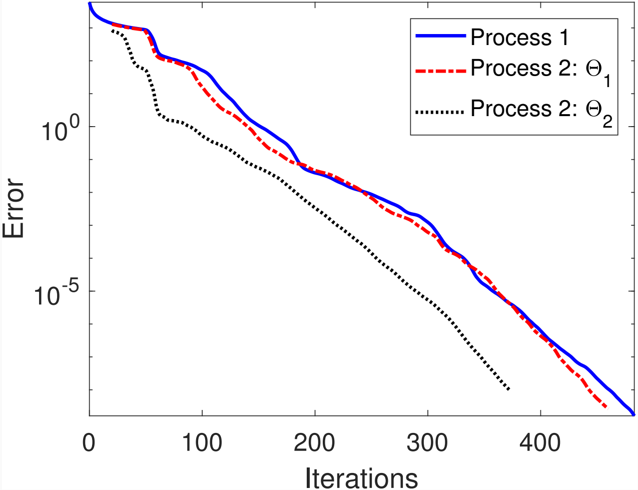
Adaptive preconditioner - num. experiment II

A test case with inhomogeneous diffusion tensor (contrast = 9e5)



Galerkin solution, algebraic error (energy norm on the elements) after 20 initial PCG iterations

Adaptive preconditioner - num. experiment II



Convergence of the energy norm of the algebraic error

Conclusion (of Part I)

I would like to recall the "message of the lecture" set at the beginning:

- The algebraic error can substantially differ from the errors of other origin. In particular, its spatial distribution can be significantly different from the discretization error.
- For systems with a sparse matrix arising from FEM discretizations, the algebraic solution accounts for global interactions in the discretization domain.
- Results based on the assumption of exact algebraic solution should not be used for computed approximations. A derivation (or revision) of theoretical results that take into account inexact algebraic solution can be more difficult and/or the results might be weaker.
- An efficient solution procedure requires thorough understanding and interaction between all phases of the solution, such as discretization, preconditioning, algebraic solution, and error estimation.

Thank you for your attention!

papez@math.cas.cz