On adaptivity, convergence and inexact computations in numerical solution of partial differential equations

Zdeněk Strakoš

Nečas Center for Mathematical Modeling
Charles University in Prague and Czech Academy of Sciences

http://www.karlin.mff.cuni.cz/~strakos

EQUADIFF 2013, Prague, August 26

Hackbusch, Computing (1995)

"Adaptivity is a further development. Instead of solving problems with a discretization of very high dimension, it is more reasonable to obtain the same solution quality by a lower dimensional but adapted discretization.

Adaptivity has created a new paradigm in mathematical computation. In traditional numerical mathematics, the fields "discretization" (e.g., FEM), its "numerical analysis" (e.g., error estimates), and "solution algorithms" (e.g., solvers for linear systems) are well separated. Adaptive techniques, however, require a combination of all three. For example, the error estimation has become a part of the algorithm. The concrete discretization is now an outgrowth of the algorithm."

Becker, Johnson, Rannacher (1995)

“ Usually, ad hoc stopping criteria are used, e.g. requiring an initial (algebraic) residual to be reduced by a certain ad hoc factor, but these criteria have no clear connection to the actual error in the corresponding approximate solution, which is the quantity of interest. This leaves the user of iterative solutions methods in a serious dilemma: [...] one has either to continue the iterations until the discrete solution error is practically “zero”, which increases the computational cost with possibly no gain in the overall precision, or take the risk of stopping the iterations prematurely. [...]

A solution to this problem can only be obtained by combining aspects of the underlying partial differential equations and the corresponding finite element discretization with aspects of the iterative discrete solution algorithm. A “pure” numerical linear algebra point of view, for instance based on the condition number of the stiffness matrix, does not appear to be able to lead to a balance of discretization and solution errors.”

Adaptive Finite Element Method (AFEM)

After setting the model and its initial discrete approximation, it proceeds with iterating the AFEM step which consists of

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

Here we will deal with SOLVE and ESTIMATE and discuss several points associated with coupling mathematical modeling, discretization and algebraic computation.

We are going to challenge some common views.

Outline

1. Operator preconditioning
2. CG in Hilbert spaces
3. Galerkin discretization and matrix CG
4. Algebraic convergence and condition numbers
5. Preconditioning as transformation of the basis
6. Spatial distribution of errors
7. Reaching an arbitrary accuracy?
8. Conclusions

1 Functional analysis setting

Let V be a real infinite dimensional Hilbert space with the inner product

$$(\cdot,\cdot)_V : V \times V \to R, \quad \text{the associated norm } \|\cdot\|_V ,$$

$V^\#$ be the dual space of bounded (continuous) linear functionals on V with the duality pairing

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

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

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

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

$$\tau : V^\# \to V .$$

1 Functional formulation in PDEs

Let $a(\cdot,\cdot) : V \times V \to R$ be a bounded and V-elliptic bilinear form. For a fixed $u \in V$ we can see $\mathcal{A}u \equiv a(u,\cdot) \in V^{\#}$, i.e. ,

$$\langle \mathcal{A}u, v\rangle = a(u,v) \quad \text{for all } v \in V\,.$$

This defines the bounded and $(\alpha-)$coercive operator

$$\mathcal{A} : V \to V^{\#}, \qquad \inf_{u\in V,\, \|u\|_V=1} \langle \mathcal{A}u, u\rangle = \alpha > 0,\ \|\mathcal{A}\| = C\,.$$

Using the Lax-Milgram theorem, the PDE problem is well-posed: For any $b \in V^{\#}$ there exist a unique solution $x \in V$ of

$$a(x,v) = \langle b, v\rangle \quad \text{for all } v \in V\,.$$

and x depends continuously on the data b ,

$$\|x\|_V \ \le\ \frac{1}{\alpha}\,\|b\|_{V^{\#}}\,.$$

1 Operator preconditioning

$$\langle \mathcal{A}x - b, v \rangle = 0 \quad \text{for all } v \in V$$

gives the (functional) equation in (the data space) $V^\#$,

$$\mathcal{A}x = b, \quad , \quad \mathcal{A} : V \to V^\#, \quad x \in V, \quad b \in V^\# .$$

Using the Riesz map,

$$(\tau\mathcal{A}x - \tau b, v)_V = 0 \quad \text{for all } v \in V .$$

Clearly, application of the Riesz map τ can be interpreted as transformation of the original problem $\mathcal{A}x = b$ in the data space $V^\#$ into the equation in the solution space V ,

$$\tau\mathcal{A}x = \tau b, \qquad \tau\mathcal{A} : V \to V, \quad x \in V, \quad \tau b \in V ,$$

which is commonly (and inaccurately) called preconditioning.

1 Operator preconditioning references

Arnold, Falk, and Winther (1997, 1997); Steinbach and Wendland (1998); Mc Lean and Tran (1997); Christiansen and Nédélec (2000, 2000); Powell and Silvester (2003); Elman, Silvester, and Wathen (2005); Axelsson and Karátson (2009); Mardal and Winther (2011); Kirby (2011); Zulehner (2011); Preconditioning Conference 2013, Oxford; ...

Inner product $\longrightarrow$ Riesz map $\longrightarrow$ Preconditioning $\longrightarrow$ Spectral bounds

There is a point to consider. What is the appropriate inner product?
A standard way is to focus on the mesh (and possibly model) parameters independence of the condition number-based convergence bounds.

1 Hiptmair, CMA (2006)

"There is a continuous operator equation posed in infinite-dimensional spaces that underlies the linear system of equations [...] awareness of this connection is key to devising efficient solution strategies for the linear systems.

Operator preconditioning is a very general recipe [...]. It is simple to apply, but may not be particularly efficient, because in case of the [*condition number*] bound of Theorem [...] is too large, the operator preconditioning offers no hint how to improve the preconditioner. Hence, operator preconditioner may often achieve [...] the much-vaunted mesh independence of the preconditioner, but it may not perform satisfactorily on a given mesh."

Mesh independence.

1 Faber, Manteuffel and Parter (1990)

"For a fixed h, using a preconditioning strategy based on an equivalent operator may not be superior to classical methods [...] Equivalence alone is not sufficient for a good preconditioning strategy. One must also choose an equivalent operator for which the bound is small.

There is no flaw in the analysis, only a flaw in the conclusions drawn from the analysis [...] asymptotic estimates ignore the constant multiplier. Methods with similar asymptotic work estimates may behave quite differently in practice."

Operator equivalence.

2 Self-adjoint $\mathcal{A}$ wrt the duality pairing, CG

CG in Hilbert spaces : $r_0 = b - \mathcal{A}x_0 \in V^{\#}, \quad p_0 = \tau r_0 \in V$

For $n = 1, 2, \ldots, n_{\max}$

$$\alpha_{n-1} = \frac{\langle r_{n-1}, \tau r_{n-1} \rangle}{\langle \mathcal{A}p_{n-1}, p_{n-1} \rangle} = \frac{(\tau r_{n-1}, \tau r_{n-1})_V}{(\tau \mathcal{A}p_{n-1}, p_{n-1})_V}$$

$$x_n = x_{n-1} + \alpha_{n-1} p_{n-1} \,,$$

stop when the stopping criterion is satisfied

$$r_n = r_{n-1} - \alpha_{n-1} \mathcal{A}p_{n-1}$$

$$\beta_n = \frac{\langle r_n, \tau r_n \rangle}{\langle r_{n-1}, \tau r_{n-1} \rangle} = \frac{(\tau r_n, \tau r_n)_V}{(\tau r_{n-1}, \tau r_{n-1})_V}$$

$$p_n = \tau r_n + \beta_n p_{n-1}$$

End

3 Galerkin discretization

Consider an N-dimensional discrete solution subspace $V_h \subset V$ with the duality pairing and the inner product as above. Then the restriction to V_h gives an approximation $x_h \in V_h$ to $x \in V$,

$$a(x_h, v) = \langle b, v \rangle \qquad \text{for all} \quad v \in V_h \,.$$

As above, the bilinear form $a(\cdot,\cdot) : V_h \times V_h \to R$ defines the operator $\mathcal{A}_h : V_h \to V_h^{\#}$ such that

$$\langle \mathcal{A}_h x_h - b, v \rangle = 0 \qquad \text{for all} \quad v \in V_h \,.$$

With restricting b to V_h, i.e. $\langle b_h, v \rangle \equiv \langle b, v \rangle$ for all $v \in V_h$, we get the equation in the discrete data space $V_h^{\#}$,

$$\mathcal{A}_h x_h = b_h, \qquad x_h \in V_h, \quad b_h \in V_h^{\#}, \quad \mathcal{A}_h : V_h \to V_h^{\#} \,.$$

3 Galerkin discretization and the matrix CG

Let $\Phi_h = (\phi_1^{(h)}, \ldots, \phi_N^{(h)})$ be the basis of V_h, $\Phi_h^{\#} = (\phi_1^{(h)\#}, \ldots, \phi_N^{(h)\#})$ the canonical basis of its dual $V_h^{\#}$. Using the coordinates in Φ_h and in $\Phi_h^{\#}$,

$$
\begin{aligned}
\langle f, v\rangle &\to \mathbf{v}^*\mathbf{f}\,,\\
(u,v)_V &\to \mathbf{v}^*\mathbf{M}\mathbf{u}, \quad (\mathbf{M}_{ij}) = ((\phi_j,\phi_i)_V)_{i,j=1,\ldots,N}\,,\\
\tau &\to \mathbf{M}^{-1}\,,\\
\mathcal{A}_h &\to \mathbf{A}, \quad (\mathbf{A}_{ij}) = (a(\phi_j,\phi_i))_{i,j=1,\ldots,N} = (\langle \mathcal{A}\phi_j,\phi_i\rangle)_{i,j=1,\ldots,N}\,,\\
b &\to \mathbf{b}\,,
\end{aligned}
$$

we get with $x_n = \Phi_h\,\mathbf{x}_n\,,\ p_n = \Phi_h\,\mathbf{p}_n\,,\ r_n = \Phi_h^{\#}\,\mathbf{r}_n$

3 Preconditioned algebraic CG $(\mathcal{A}_h x_h = b_h)$

$\mathbf{r}_0 = \mathbf{b} - \mathbf{A}\mathbf{x}_0, \quad$ solve $\quad \mathbf{M}\mathbf{z}_0 = \mathbf{r}_0, \ \mathbf{p}_0 = \mathbf{z}_0$

For $n = 1, \ldots, n_{\max}$

$$
\begin{aligned}
\alpha_{n-1} &= \frac{\mathbf{z}_{n-1}^* \mathbf{r}_{n-1}}{\mathbf{p}_{n-1}^* \mathbf{A}\mathbf{p}_{n-1}} \\
\mathbf{x}_n &= \mathbf{x}_{n-1} + \alpha_{n-1}\mathbf{p}_{n-1}\,, \qquad \text{stop when the stopping criterion is satisfied} \\
\mathbf{r}_n &= \mathbf{r}_{n-1} - \alpha_{n-1}\mathbf{A}\mathbf{p}_{n-1} \\
\mathbf{z}_n &= \mathbf{M}^{-1}\mathbf{r}_n\,, \qquad \text{solve for } \mathbf{z}_n \\
\beta_n &= \frac{\mathbf{z}_n^* \mathbf{r}_n}{\mathbf{z}_{n-1}^* \mathbf{r}_{n-1}} \\
\mathbf{p}_n &= \mathbf{z}_n + \beta_n \mathbf{p}_{n-1}
\end{aligned}
$$

End

4 CG, outliers and condition numbers

Theorem (1977, ... , 2009, 2010, 2011, 2012, 2013, ...)

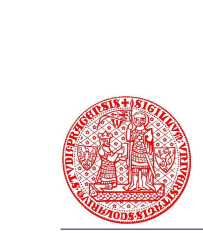
Consider the desired accuracy ϵ, $\kappa_s(\mathbf{A}) \equiv \lambda_{N-s}/\lambda_1$. Then

$$\mathbf{k} = \mathbf{s} + \left\lceil \frac{\ln(2/\epsilon)}{2} \sqrt{\kappa_s(\mathbf{A})} \right\rceil$$

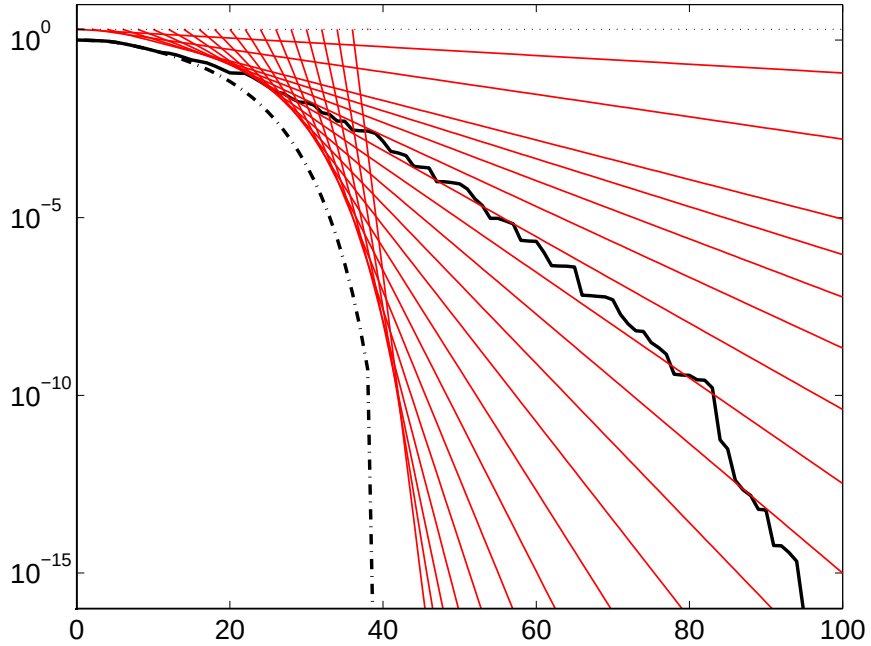
CG steps will produce the approximate solution $\mathbf{x}_n$ satisfying

$$\|\mathbf{x} - \mathbf{x}_n\|_{\mathbf{A}} \leq \epsilon \, \|\mathbf{x} - \mathbf{x}_0\|_{\mathbf{A}} \, .$$

Assuming exact arithmetic, this statement is correct. In the context of CG computations it makes, however, no sense.



4 Liesen, S (2012); Gergelits, S (2013)



Short recurrences always mean in practical computations loss of (bi-)orthogonality due to rounding errors! Principal consequences are not resolved by *the common assumption* that this phenomenon does not take place.

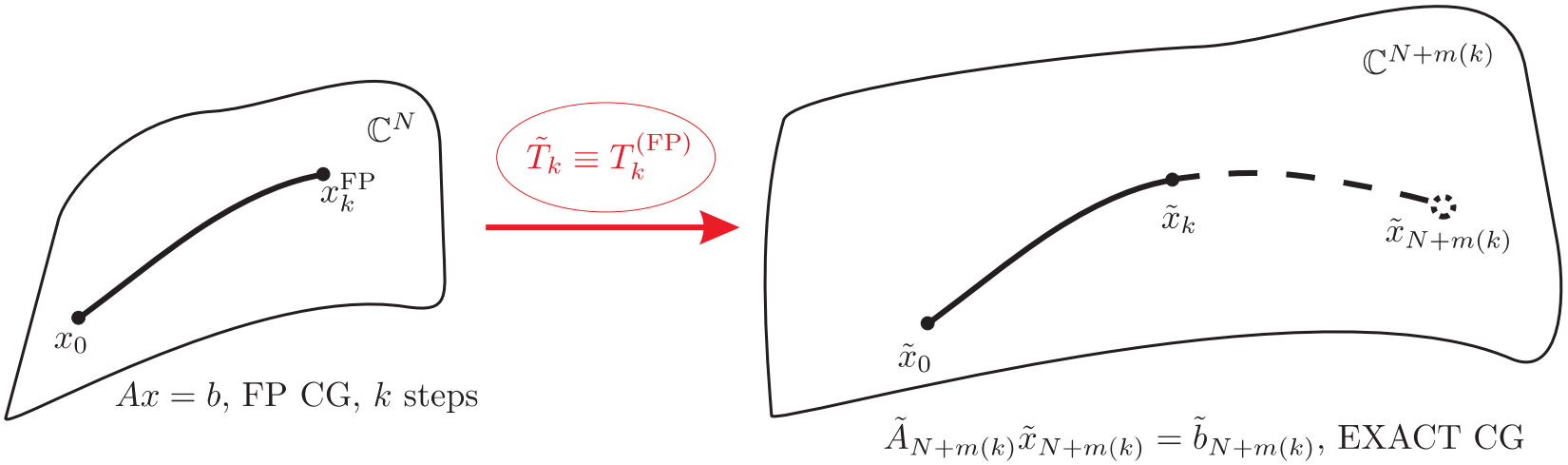
4 Clusters = fast convergence ?

- in exact arithmetic, CG applied to a matrix with the spectrum consisting of t tight clusters of eigenvalues does not find, in general, a reasonably close approximation to the solution within t steps.

- Finite precision arithmetic CG computation can be viewed as exact CG applied to a larger matrix with the individual original eigenvalues replaced by tight clusters.

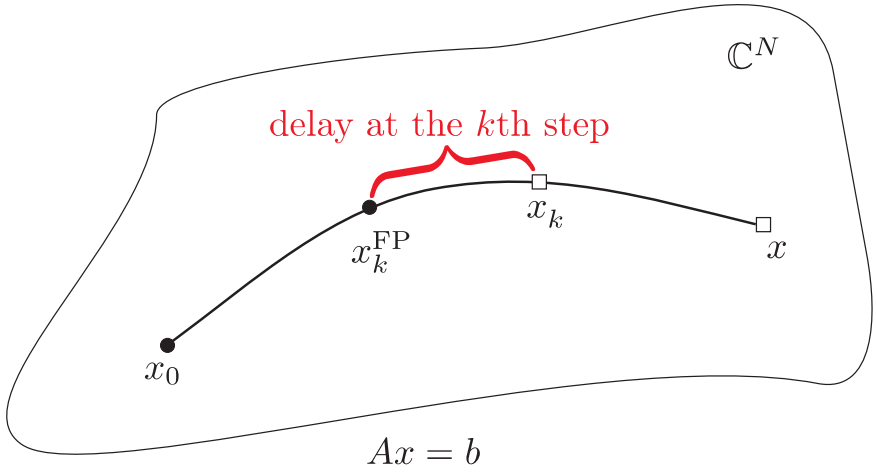
- Finite precision arithmetic CG computation with a matrix having t isolated well separated eigenvalues may require for reaching a reasonable approximate solution a significantly larger number of steps than t .

4 Analysis of FP CG behaviour



First k steps of finite precision CG (Lanczos) is analyzed as exact CG (Lanczos) for a different, possibly much larger problem. The central point is the computed Jacobi matrix.

4 Delay of convergence due to rounding errors



- □ exact computation
- ● finite precision computation

Rigorous description of the CG behaviour, including FP arithmetic, is based on the relationship with the problem of moments, orthogonal polynomials and the Gauss-Christoffel quadrature; see, e.g., Greenbaum (1989); Meurant and S (2006); S and Tichý (2002); Liesen and S (2013).

4 Widespread common knowledge

- “[...] useful insight is gained as to the relationship between Hilbert space and matrix condition numbers and translating Hilbert space fixed point iterations into matrix computations provides new ways of motivating and explaining some classic iteration schemes.” Kirby, SIREV, 2010

- “[...] in the early sweeps the convergence is very rapid but slows down, this is the sublinear behavior. The convergence then settles down to a roughly constant linear rate [...] Towards the end new speed may be picked up again, corresponding to the superlinear behavior. [...] In practice all phases need not be identifiable, nor need they appear only once and in this order.” Nevanlinna, 1993, Section 1.8

- “However, if the operator has a few eigenvalues far away from the rest of the spectrum, then the estimate is not sharp. In fact, a few ‘bad eigenvalues’ will have almost no effect on the asymptotic convergence of the method [...]” Mardal and Winther, NLAA, 2011

4 An example

Consider a fixed point iteration in the Banach space with the bounded operator $\mathcal{B}$,

$$u = \mathcal{B}\,u + f\,, \quad u^{(\ell+1)} = \mathcal{B}\,u^{(\ell)} + f\,.$$

Using polynomial acceleration we can do better,

$$u - u^{(\ell)} = p_\ell(\mathcal{B})\,(u - u^{(0)})\,.$$

Separating the operator polynomial from the initial error, it seems natural to minimize the appropriate norm of the operator polynomial

$$\|p_\ell(\mathcal{B})\| \qquad \text{subject to } \; p_\ell(0) = 1\,.$$

4 A drawback of oversimplification

Consider now a numerical (finite dimensional) approximation $\mathcal{B}_h$ of the bounded operator $\mathcal{B}$. Then

$$p(\mathcal{B}) - p(\mathcal{B}_h) = \frac{1}{2\pi\iota} \int_\Gamma p(\lambda) \left[(\lambda\mathcal{I} - \mathcal{B})^{-1} - (\lambda\mathcal{I} - \mathcal{B}_h)^{-1} \right] d\lambda \,.$$

This is considered a sufficient argument why to study algebraic iterations directly in abstract (infinite dimensional) Banach spaces.

At this level of abstraction, many challenges which one must deal with in studying finite computational processes at finite dimensional spaces are simply not visible. Abstract Banach space settings make things seemingly easier and elegant. The troubles are not seen and questions about the cost of algebraic computations (and the cost of the whole solution process) are oversimplified.

5 Preconditioning transforms the basis!

Algebraic preconditioning can be viewed as the finite dimensional CG with setting $\mathbf{M} = \mathbf{I}$ (this corresponds in Galerkin discretization of the finite dimensional CG to taking discretization basis Φ orthonormal wrt $(.,.)_V$) applied to

$$\mathbf{B}\mathbf{w} = \mathbf{c}$$

with

$$\mathbf{B} = \mathbf{L}_h^{-1}\mathbf{A}\mathbf{L}_h^{-*}, \quad \mathbf{c} = \mathbf{L}_h^{-1}\mathbf{b}, \quad \mathbf{x} = \mathbf{L}_h^{-*}\mathbf{w}\,, \quad \mathbf{M}_h = \mathbf{L}_h\mathbf{L}_h^{*}\,.$$

Observation:

The associated Hilbert space formulation of CG in V_h corresponds to the transformation of the bases

$$\Phi_t \;=\; \Phi_h\,\mathbf{L}_h^{-*}\,, \quad \Phi_t^{\#} \;=\; \Phi_h^{\#}\,\mathbf{L}_h^{*}\,.$$

5 Preconditioning transforms the basis!

$$\mathbf{B} \equiv (\mathbf{B}_{ij}) = \left(\langle \mathcal{A}\phi_j^{(t)}, \phi_i^{(t)} \rangle \right)_{i,j=1,\ldots,N} = (a(\phi_j^{(t)}, \phi_i^{(t)}))_{i,j=1,\ldots,N},$$

where

$$\phi_\ell^{(t)} = \Phi_h \left(\mathbf{L}_h^{-*} \mathbf{e}_\ell \right), \quad \ell = 1, \ldots, N$$

and the right hand side

$$\mathbf{c} = \Phi_h^{\#} \mathbf{L}_h^{*} \, \mathbf{b} \, .$$

Please recall, e.g., the hierarchical bases preconditioning
Yserentant (1985, 1986), Axelsson, Vassilevski, ... , Gockenbach (2006).

Remark. Equivalently, with the orthonormalized discretization basis Φ_t wrt $(.,.)_V$ we get $\mathbf{M} = \mathbf{I}$ and $\mathcal{A}_h \to \mathbf{B}$. With the choice $(.,.)_V = (.,.)_a$ we get $\mathbf{B} = \mathbf{I}$.

5 Standard argument used in FEM

Sparsity of the resulted matrices is always presented as the main advantage of FEM discretizations.

Sparsity means locality of information. In order to solve the problem, we need a global transfer of information. Therefore preconditioning! It is needed on the computational level in order to take care for the trouble caused by the (*computationally*) inconvenient approximation of the mathematical model when the *appropriate globally supported* basis functions are missing.

Preconditioning can be interpreted as an intentional loss of sparsity (loss of locality of the supports of the basis functions).

Sparsity is important for efficiency, but perhaps in a different meaning; see, e.g., Schaeffer, Caflisch, Hauck and Osher (2013), .

6 Spatial distribution of algebraic errors

Knupp and Salari, 2003:

"There may be incomplete iterative convergence (IICE) or round-off-error that is polluting the results. If the code uses an iterative solver, then one must be sure that the iterative stopping criteria is sufficiently tight so that the numerical and discrete solutions are close to one another. Usually in order-verification tests, one sets the iterative stopping criterion to just above the level of machine precision to circumvent this possibility."

In solving tough problems this can not be afforded.

How to measure the algebraic error ?

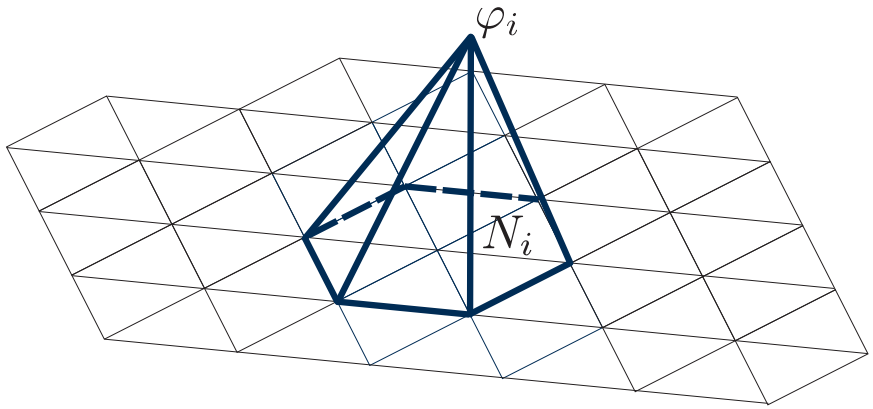
6 Local discretisation and global computation

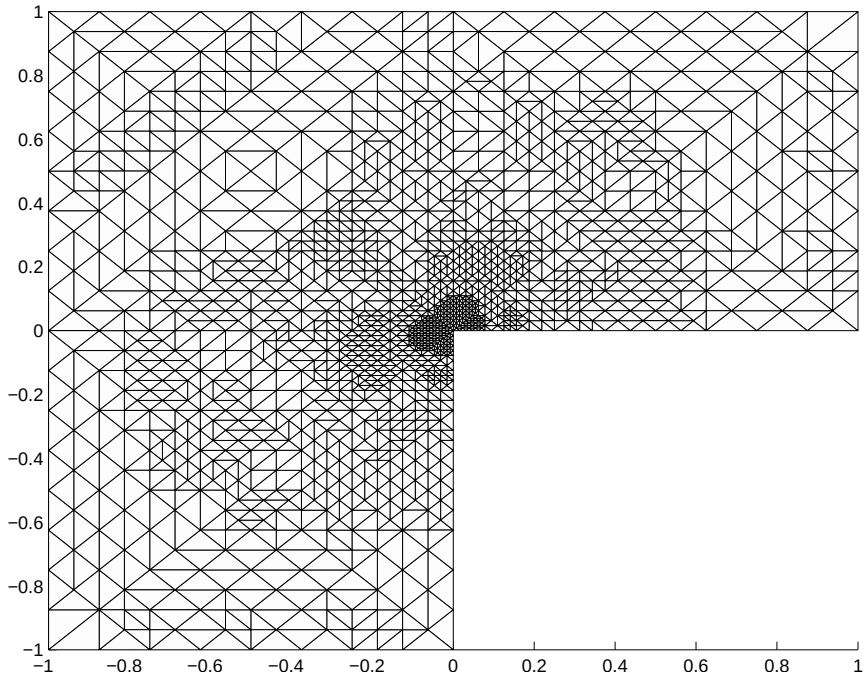
Discrete (piecewise polynomial) FEM approximation $x_h = \Phi_h \mathbf{x}_n$.

- If $\mathbf{x}_n$ is known exactly, then x_h is approximated over the given domain as the (exact) linear combination of the local basis functions.
- However, apart from trivial cases, $\mathbf{x}_n$ that supply the global information is not known exactly. Then

$$\underbrace{x - x_h^{(n)}}_{\text{total error}} = \underbrace{x - x_h}_{\text{discretisation error}} + \underbrace{x_h - x_h^{(n)}}_{\text{algebraic error}} .$$

6 Local discretisation





6 Global norm of the error (Poisson problem)

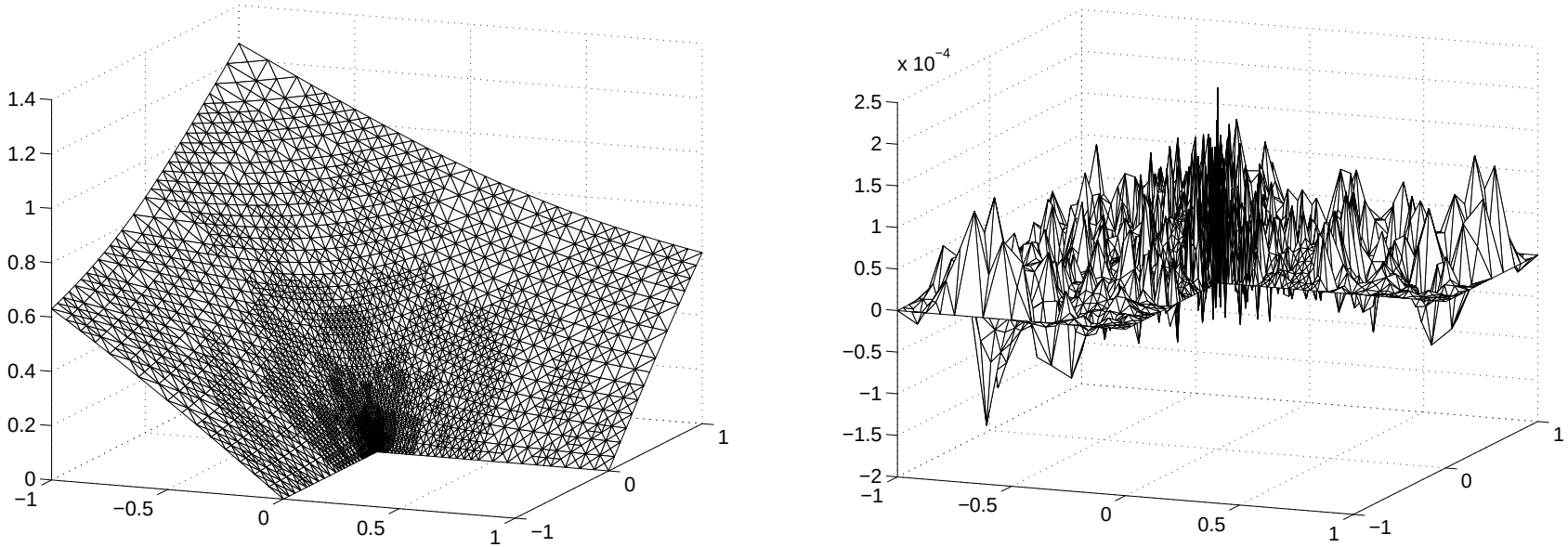
Theorem (x_h denotes the discrete Galerkin solution)

$$\begin{aligned} \|\nabla(x - x_h^{(n)})\|^2 &= \|\nabla(x - x_h)\|^2 + \|\nabla(x_h - x_h^{(n)})\|^2 \\ &= \|\nabla(x - x_h)\|^2 + \|\mathbf{x} - \mathbf{x}_n\|_{\mathbf{A}}^2 \end{aligned}$$

holds up to a small inaccuracy proportional to machine precision.

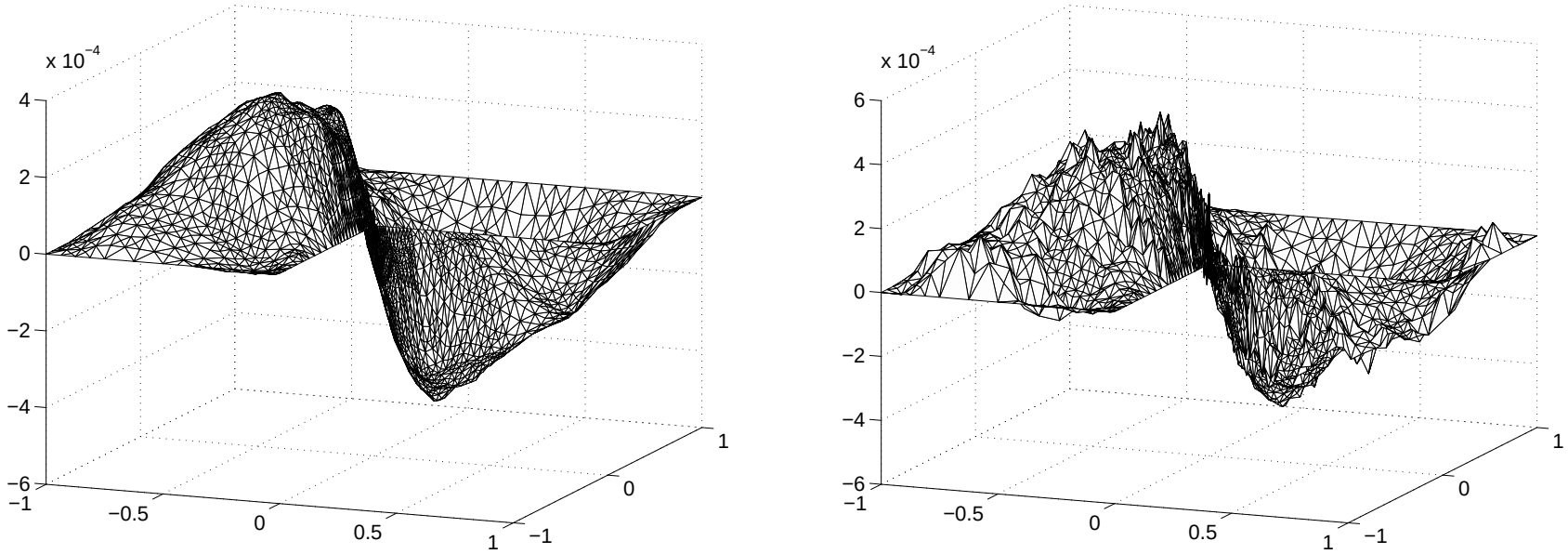
What is the distribution of the algebraic error
in the functional space ?

6 L-shape domain, Papež, Liesen, S (2013)



Exact solution x (left) and the discretisation error $x - x_h$ (right) in the Poisson model problem, linear FEM.

6 L-shape domain, Papež, Liesen, S (2013)



Algebraic error $x_h - x_h^{(n)}$ (left) and the total error $x - x_h^{(n)}$ (right). Here

$$\|\nabla(x - x_h)\| \; > \; 0.1 \, \|\mathbf{x} - \mathbf{x}_n\|_{\mathbf{A}} \, .$$

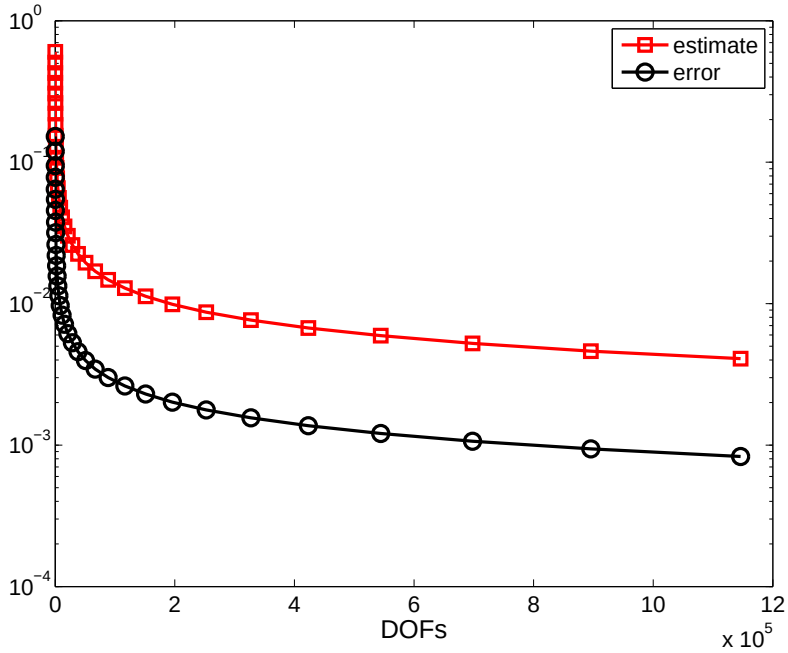
6 Adaptive stopping criteria

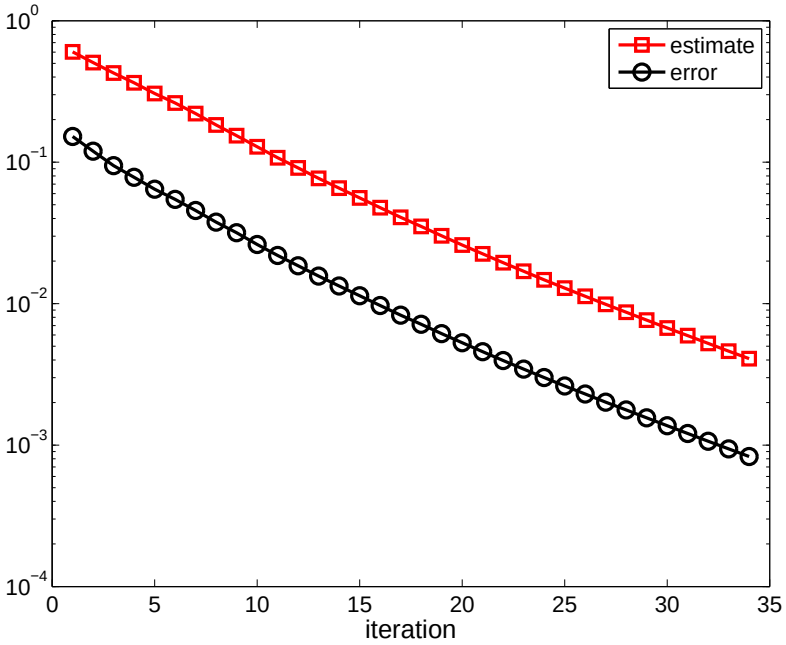
They should be based on a-posteriori error estimators which are fully computable and provide information on local distribution of the error (including the algebraic error) within the domain. Ideally, a-posteriori error estimators should satisfy the following additional properties:

- reliability (guaranteed upper bound);
- local efficiency;
- asymptotic exactness.

Verfürth (1996); Ainsworth and Oden (2000); Babuška and Strouboulis (2001); Bangerth and Rannacher (20003); ... ; Bernd, Manteuffel, and McCormick (1996); ... ; Arioli, Noulard, and Russo (2001); Arioli, Loghin, and Wathen (2005); Silvester and Simoncini (2011); ... ; Jiranek, S, and Vohralik (2011); Vohralik and Ern (2013); Arioli, Liesen, Miedlar, and S (2013); ...

7 Reaching an arbitrary accuracy?





It seems and it has been proved that an arbitrary prescribed accuracy can be reached using AFEM in a finite number of steps. Here linear FEM; see Morin, Nocheto, and Siebert (2002); Stevenson (2007). Something does not fit $\longrightarrow$ maximal attainable accuracy in matrix computations.

8 Conclusions

Patrick J. Roache's book *Validation and Verification in Computational Science*, 1998, p. 387:

"With the often noted tremendous increases in computer speed and memory, and with the less often acknowledged but equally powerful increases in algorithmic accuracy and efficiency, a natural question suggest itself. What are we doing with the new computer power? with the new GUI and other set-up advances? with the new algorithms? What *should* we do? ... Get the right answer."

This requires to consider modelling, discretisation, analysis, and computation tightly coupled parts of a single solution process.
and to avoid unjust simplifications.

8 Our contributions and work to be done

Steps in this direction:

- Operator and algebraic preconditioning in relation to the choice of the discrerization basis.
- Krylov subspace methods viewed as the matching moments model reduction (infinite or finite dimenstional setting).
- *A-posteriori* evaluation of the total error which is based on quantities of interest and includes the algebraic part.
- Adaptivity and stopping criteria for iterative solvers.
- Numerical stability analysis of adaptive numerical schemes.

References

- J. Liesen and Z.S., Krylov Subspace Methods, Principles and Analysis. Oxford University Press (2013)
- T. Gergelits and Z.S., Composite convergence bounds based on Chebyshev polynomials and finite precision conjugate gradient computations, Numerical Algorithms (2013) (DOI 10.1007/s11075-013-9713-z)
- J. Papez, J. Liesen and Z.S., On distribution of the discretization and algebraic error in numerical solution of partial differential equations, Preprint MORE/2012/03, (2013)
- M. Arioli, J. Liesen, A. Miedlar, and Z.S., Interplay between discretization and algebraic computation in adaptive numerical solution of elliptic PDE problems, GAMM Mitteilungen 36, 102-129 (2013)
- J. Málek and Z.S., From PDEs through functional analysis to iterative methods, or there and back again. In preparation.

Thank you very much for kind patience!