On the numerical behavior of quadrature-based bounds for the A-norm of the error in CG

Gérard Meurant and Petr Tichý

Institute of Computer Science, Czech Academy of Sciences
Faculty of Mathematics and Physics, Charles University in Prague
gerard.meurant@gmail.com, tichy@cs.cas.cz

Abstract

The A-norm of the error in the conjugate gradient (CG) method can be estimated using quadrature-based bounds. The formulas for computing these bounds are derived assuming exact arithmetic, using properties of orthogonal polynomials. However, in **finite precision arithmetic**, orthogonality is lost, and convergence of CG is delayed. Then, it is relevant to ask whether the formulas for computing the bounds still approximate the A-norm of the actual error even though the computed approximate solution is far away from its exact counterpart. This nontrivial phenomenon was explained in [2]. In [4] we have shown that the lower bound based on **Gauss quadrature**, and computed in a convenient way, is **numerically stable** until the A-norm of the error reaches its ultimate level of accuracy.

Recently, in [3], we improved the CGQL algorithm [1] by Golub and Meurant for computing various kinds of quadrature-based bounds. The new CGQ algorithm uses simpler formulas than CGQL, and computes quadrature-based bounds directly from the CG coefficients.

Our numerical experiments predict that some numerical difficulties may arise when computing upper bounds based on modified quadrature rules like **Gauss-Radau**, no matter whether we use CGQL or CGQ. In this presentation we investigate why these difficulties arise.

The Conjugate Gradient algorithm

The Conjugate Gradient (CG) algorithm is the iterative method of choice for solving linear systems $Ax = b$ with a real positive definite symmetric matrix A. For simplicity we assume that $\|b\| = 1$, $x_0 = 0$.

```
input $A$, $b$
$r_0 = b$, $p_0 = r_0$
for $k = 1, 2, \dots$ do
    $\gamma_{k-1} = \dfrac{r_{k-1}^T r_{k-1}}{p_{k-1}^T A p_{k-1}}$
    $x_k = x_{k-1} + \gamma_{k-1} p_{k-1}$
    $r_k = r_{k-1} - \gamma_{k-1} A p_{k-1}$
    $\delta_k = \dfrac{r_k^T r_k}{r_{k-1}^T r_{k-1}}$
    $p_k = r_k + \delta_k p_{k-1}$
end for
```

In exact arithmetic, the CG iterates x_k minimize the A-norm of the error over the kth Krylov subspace,

$$\|x - x_k\|_A = \min_{y \in \mathcal{K}_k(A,b)} \|x - y\|_A .$$

CG and Gauss Quadrature

The optimality of the CG method can be seen via Gauss quadrature. At any iteration step k, CG (implicitly) determines weights and nodes of the k-point Gauss quadrature

$$\int_\zeta^\xi f(\lambda)\, d\omega(\lambda) \;=\; \sum_{i=1}^{k} \omega_i\, f(\theta_i) \;+\; \mathcal{R}_k[f] . \qquad (1)$$

In particular, for $f(\lambda) \equiv \lambda^{-1}$, the **remainder is positive**, and (1) can be written using γ_j and $\|r_j\|^2$,

$$\|x\|_A^2 \;=\; \sum_{j=0}^{k-1} \gamma_j \|r_j\|^2 \;+\; \|x - x_k\|_A^2 .$$

Following the idea of [2], we can consider quadrature rules at iterations k and $k + d$ for some integer $d > 0$. Then, we get the formula

$$\|x - x_k\|_A^2 \;=\; \sum_{j=k}^{k+d-1} \gamma_j \|r_j\|^2 \;+\; \|x - x_{k+d}\|_A^2 .$$

By neglecting the error at iteration $k + d$ we obtain a lower bound for the A-norm of the error at iteration k. In [4] we have shown that the above identity holds (up to some small inaccuracy) also for numerically computed quantities, and it can be used for estimating the A-norm of the actual error.

Gauss-Radau quadrature

To obtain an upper bound, one can modify the Gauss quadrature rule. Consider the Gauss-Radau quadrature

$$\int_\zeta^\xi f(\lambda)\, d\omega(\lambda) = \left[\sum_{i=1}^{k} \widetilde{\omega}_i f\left(\widetilde{\theta}_i\right) + \widetilde{\omega}_{k+1} f(\mu) \right] \;+\; \widetilde{\mathcal{R}}_k[f],$$

where the node μ, $0 < \mu < \lambda_1$, is prescribed, and the unknown nodes and weights are chosen to maximize the degree of exactness. For $f(\lambda) \equiv \lambda^{-1}$, the **remainder is negative** and the value of the Gauss-Radau quadrature can be determined algebraically by computing the value $\left(\widetilde{T}_{k+1}^{-1}\right)_{1,1}$,

$$\widetilde{T}_{k+1} = \begin{bmatrix} \alpha_1 & \beta_1 & & & \\ \beta_1 & \ddots & \ddots & & \\ & \ddots & \ddots & \beta_{k-1} & \\ & & \beta_{k-1} & \alpha_k & \beta_k \\ & & & \beta_k & \widetilde{\alpha}_{k+1} \end{bmatrix},$$

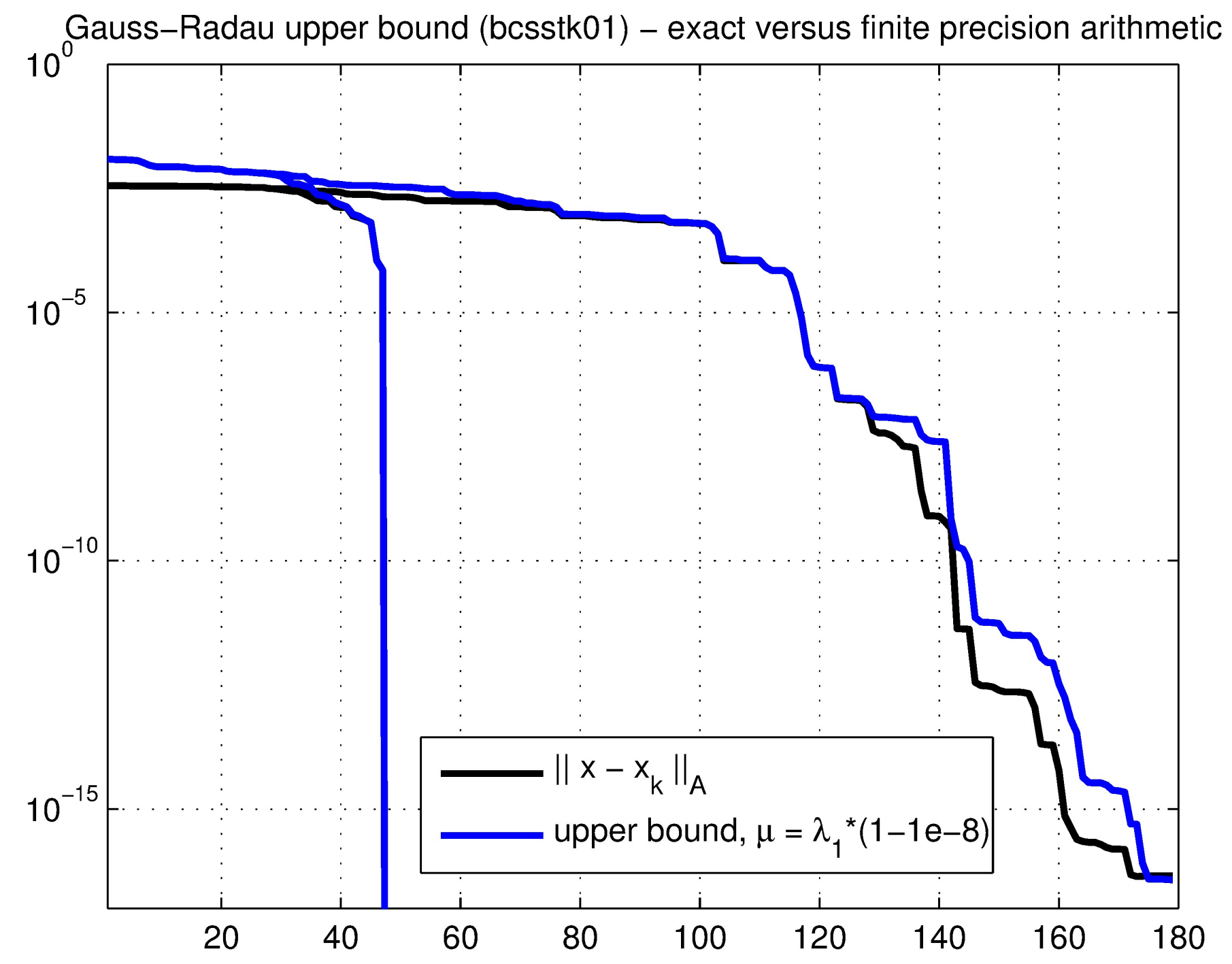
where $\widetilde{\alpha}_{k+1}$ is determined such that μ is an eigenvalue of $\widetilde{T}_{k+1}$, and α_j and β_j are the Lanczos coefficients that can easily be computed from the CG coefficients. Considering Gauss quadrature rule at iteration k and Gauss-Radau rule at iteration $k+d$, we can use the same trick as for the Gauss quadrature bound. In particular, for $d = 1$ we get

$$\|x - x_k\|_A^2 = \left[\left(\widetilde{T}_{k+1}^{-1}\right)_{1,1} - \left(T_k^{-1}\right)_{1,1} \right] \;+\; \widetilde{\mathcal{R}}_k[\lambda^{-1}] . \qquad (2)$$

Neglecting the remainder that is negative, we get an upper bound for the squared A-norm of the error. The bound can be computed using CGQL or CGQ at a negligible cost.

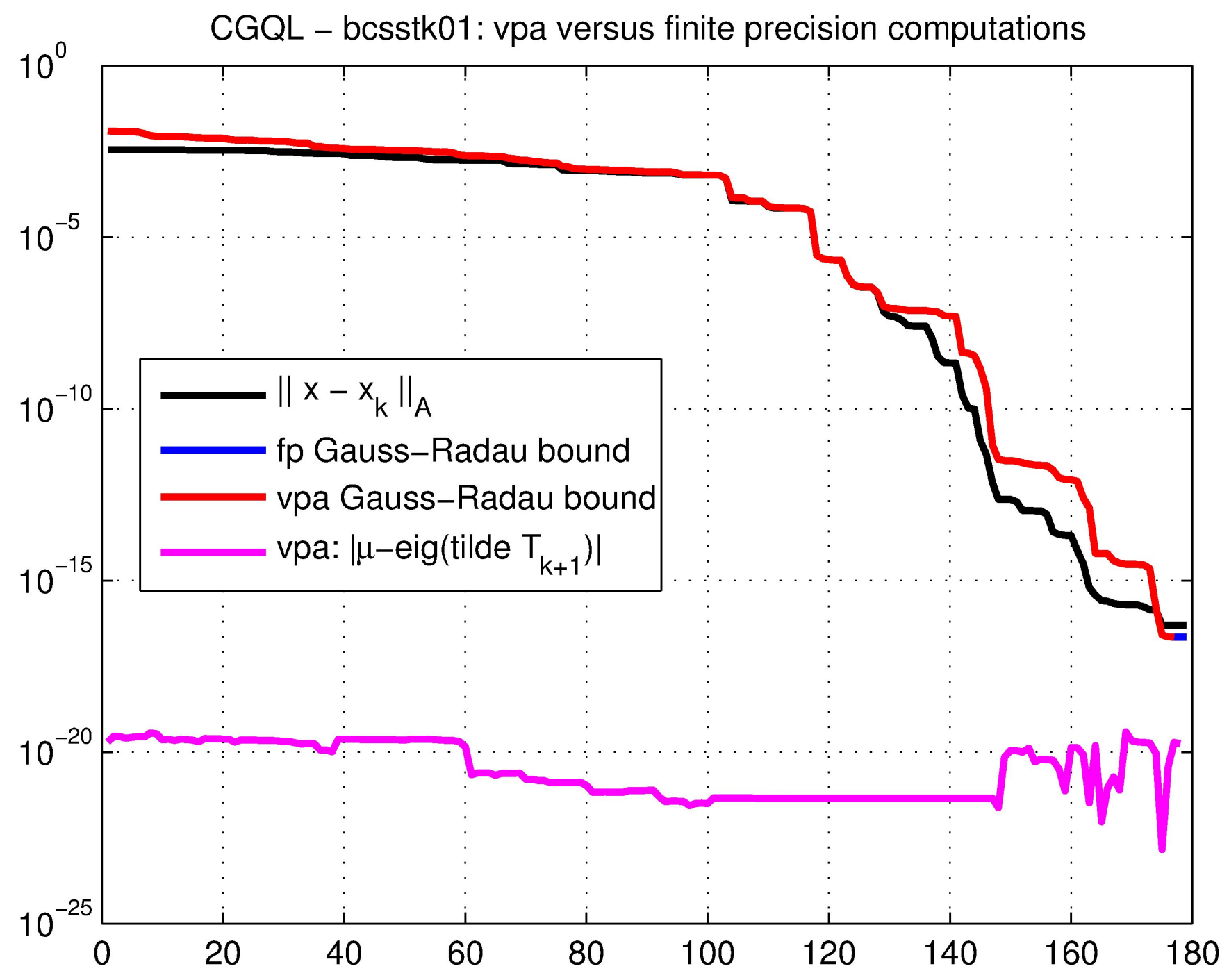
The upper bound in finite precision arithmetic

We observed that in finite precision arithmetic, the upper bound based on Gauss-Radau quadrature starts to differ from the A-norm of the error in later iterations (it is delayed).



It was not clear whether the difference was caused by rounding **errors in the computation of the bound**, or by other phenomenon related to finite precision CG computations. Here λ_1 is the smallest eigenvalue of A.

Experiment: We ran finite precision CG to generate the CG coefficients, and used variable precision arithmetic (vpa) to compute the upper bound.



The results show that the formulas used in CGQL and CGQ produce in finite precision arithmetic the same upper bound (up to some small inaccuracy) as the formulas that use vpa arithmetic. In other words, even if we compute the bound exactly, it would remain the same.

Observation

In later iterations, the eigenvalues of T_{k+1} usually closely approximate the eigenvalues of A. Because of finite precision arithmetic, several Ritz values can approximate a single eigenvalue of A. We observed that when the gradient of the characteristic polynomial of T_{k+1} in λ_1 is large, the Gauss-Radau upper bound starts to differ from the A-norm of the actual error. This situation is often related to convergence of the second Ritz value to λ_1.

The entry $\widetilde{\alpha}_{k+1}$ of the modified matrix $\widetilde{T}_{k+1}$ need not be close to α_{k+1}. When the Gauss-Radau upper bound starts to differ from the A-norm of the error, many of the eigenvalues of $\widetilde{T}_{k+1}$ are still clustered about the eigenvalues of A. However, the number of eigenvalues of $\widetilde{T}_{k+1}$ in a cluster can differ substantially from the number of eigenvalues of T_{k+1} in the same cluster.

In summary, it seems that if the smallest eigenvalue of A is closely approximated by several Ritz values, then prescribing the eigenvalue $\mu < \lambda_1$ of $\widetilde{T}_{k+1}$ changes the nature of the problem, and the values

$$\left(T_{k+1}^{-1}\right)_{1,1} \quad \text{and} \quad \left(\widetilde{T}_{k+1}^{-1}\right)_{1,1}$$

start to differ significantly. As a consequence, convergence of the Gauss-Radau upper bound is delayed.

Note that there is **another problem** related to the use of the Gauss-Radau upper bound. In the previous we assumed that a prescribed node $\mu < \lambda_1$ is given. However, in practical computations, the parameter μ has to be determined. This represents a nontrivial task. The node μ should be smaller than λ_1, and, simultaneously, very close to λ_1, otherwise the Gauss-Radau upper bound would be a poor approximation of the A-norm of the error. One can approximate λ_1 using Ritz values during the CG computations. However, Ritz values provide only upper bounds on λ_1, and some heuristics has to be used to obtain $\mu < \lambda_1$ such that $\mu \approx \lambda_1$. When using a heuristics, one can hardly obtain a **guaranteed upper bound** for the A-norm of the error.

The work of the second author was supported by the GACR grant P201/13-06684 S.

References

[1] G. H. Golub and G. Meurant, *Matrices, moments and quadrature with applications*, Princeton University Press, USA, 2010.

[2] G. H. Golub and Z. Strakoš, *Estimates in quadratic formulas*, Numerical Algorithms, 8 (1994), pp. 241–268.

[3] G. Meurant and P. Tichý, *On computing quadrature-based bounds for the A-norm of the error in conjugate gradients*, Numer. Algorithms, 62 (2013), pp. 163–191.

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

SIAM Conference on Applied Linear Algebra, Atlanta, USA, 2015