A new algorithm for computing quadrature-based bounds in CG

Gérard Meurant and Petr Tichý

Abstract

Today the (preconditioned) Conjugate Gradient (CG) algorithm by Hestenes and Stiefel is the iterative method of choice for solving linear systems $Ax = b$ with a real positive definite symmetric matrix A. An important question is when to stop the iterations. Ideally, one would like to stop the iterations when some norm of the error $x - x_k$, where x_k are the CG iterates, is small enough. However, the error is unknown and most CG implementations rely on stopping criteria that use the residual norm $\|r_k\| = \|b - Ax_k\|$ as a measure of convergence. These types of stopping criteria can provide misleading information about the actual error. It can stop the iterations too early when the norm of the error is still too large, or too late in which case too many floating point operations have been done for obtaining the required accuracy. This motivated researchers to look for ways to compute estimates of some norms of the error during CG iterations. The norm of the error which is particularly interesting for CG is the A-norm which is minimized at each iteration,

$$\|x - x_k\|_A = ((x - x_k)^T A(x - x_k))^{1/2}.$$

Inspired by the connection of CG with the Gauss quadrature rule for a Riemann-Stieltjes integral, a way of research on this topic was started by Gene Golub in the 1970s and continued throughout the years with several collaborators (e.g., Dahlquist, Eisenstat, Fischer, Meurant, Strakoš). The main idea of Golub and his collaborators was to obtain bounds for the integral using different quadrature rules. It turns out that these bounds can be computed without the knowledge of the stepwise constant measure and at almost no cost during the CG iterations.

These techniques were used by Golub and Meurant in 1994 for providing lower and upper bounds on quadratic forms $u^T f(A)u$ where f is a smooth function, A is a symmetric matrix and u is a given vector. Their algorithm GQL (Gauss Quadrature and Lanczos) was based on the Lanczos algorithm and on computing functions of Jacobi matrices. Later in [1], these techniques were adapted to CG to compute lower and upper bounds on the A-norm of the error for which the function is $f(\lambda) = \lambda^{-1}$. The idea was to use CG instead of the Lanczos algorithm, to compute explicitly the entries of the corresponding Jacobi matrices and their modifications from the CG coefficients, and then to use the same formulas as in GQL. The formulas were summarized in the CGQL algorithm (QL standing again for Quadrature and Lanczos), whose most recent version is described in the book [2].

The CGQL algorithm may seem complicated, particularly for computing bounds with the Gauss-Radau or Gauss-Lobatto quadrature rules. In this presentation based on our paper [4], we intend to show that the CGQL formulas can be considerably simplified. We use the fact that CG computes the Cholesky decomposition of the Jacobi matrix which is given implicitly, and derive new algebraic formulas by working with the LDL^T factorizations of the Jacobi matrices and their modifications instead of computing the Lanczos coefficients explicitly. In other words, one can obtain the bounds from the CG coefficients without computing the Lanczos coefficients. The algebraic derivation of the new formulas is more difficult than it was when using Jacobi matrices but, in the end, the formulas are simpler. Obtaining simple formulas is a prerequisite for analyzing the behaviour of the bounds in finite precision arithmetic and also for a better understanding of their dependence on the auxiliary parameters μ and η which are lower and upper bounds (or estimates) of the smallest and largest eigenvalues. We hope that these improvements will be useful for the implementation of quadrature-based error bounds into existing and forthcoming CG codes.

We first focus on explanation of the main idea of quadrature-based estimates of the A-norm of the error in CG. For simplicity, we will just concentrate on the Gauss and Gauss-Radau quadrature rules. Using the ideas of [3] and [5], we end up with the formula

$$\|x - x_k\|_A^2 \;=\; \widehat{Q}_{k,d} \;+\; \widehat{\mathcal{R}}_{k+d}\,,$$

where $\widehat{\mathcal{R}}_{k+d}$ stands for the remainder of the considered quadrature rule, $\widehat{Q}_{k,d}$ is computable, and $d > 0$ is a chosen integer. The remainder is positive when using the Gauss quadrature rule, and it is negative when using the Gauss-Radau quadrature rule with a prescribed node $\mu > 0$ that is strictly smaller than the smallest eigenvalue of A. Hence, $\widehat{Q}_{k,d}$ can provide a lower bound or an upper bound on $\|x - x_k\|_A^2$.

The question is how to compute $\widehat{Q}_{k,d}$ efficiently. The algorithm CGQL computes $\widehat{Q}_{k,d}$ using the entries of the corresponding Jacobi matrices and their rank-one or rank-two modifications. Our new algorithm CGQ (Conjugate Gradients and Quadrature) and its preconditioned version compute $\widehat{Q}_{k,d}$ directly from the CG coefficients. In particular, the lower bound based on the Gauss quadrature rule can be computed using the sum

$$\widehat{Q}_{k,d} = \sum_{j=k}^{k+d-1} \Delta_j, \qquad \Delta_j \equiv \gamma_j \|r_j\|^2,$$

where γ_j are the CG coefficients, see also [5], and the upper bound based on the Gauss-Radau quadrature rule can be computed using

$$\widehat{Q}_{k,d} = \sum_{j=k}^{k+d-2} \Delta_j + \Delta_k^{(\mu)}$$

where $\Delta_k^{(\mu)}$ is updated using the new formula

$$\Delta_k^{(\mu)} = \frac{\|r_k\|^2 \left(\Delta_{k-1}^{(\mu)} - \Delta_{k-1}\right)}{\mu \left(\Delta_{k-1}^{(\mu)} - \Delta_{k-1}\right) + \|r_k\|^2}\,, \qquad \Delta_0^{(\mu)} = \frac{\|r_0\|^2}{\mu}.$$

In the final numerical experiment we will illustrate some of the difficulties that may arise with modified quadrature rules when computing in finite precision arithmetic.

References

[1] G. H. Golub and G. Meurant, *Matrices, moments and quadrature. II. How to compute the norm of the error in iterative methods*, BIT, 37 (1997), pp. 687–705.

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

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

[4] 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.

[5] 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.