Conjugate-Gradient Like Methods for Non-Symmetric and Indefinite Problems

Kent-Andre Mardal†

† Simula Research Laboratory and University of Oslo, Norway

Nonlinear Iteration

We remember the general nonlinear iteration,

$$x_n = x_{n-1} + \alpha_n p_n \tag{1}$$

We need to choose

- p_n, a vector, the search direction
- α_n, a scalar to be determined

Observations with the Conjugate-Gradient method

- We saw that we could use the residual r_m thanks to Cayley-Hamilton theorem
- We also saw that it was advantagous to compute a search vector based on the residual
- The search vectors were orthogonal in a certain sense

Linear Iteration

We also remember the general linear iteration,

$$x_n = x_{n-1} + \tau B r_m \tag{2}$$

Hence,

- $p_n = B r_m$
- $\alpha_n = \tau$

Notice that the idea of preconditioning applies equally well to the Conjugate-Gradient like methods!

Conjugate-Gradient Method

In the Conjugate Gradient method α_m and p_m were constructed such that

- $(Ap_m, p_k) = 0$, for $m \neq k$
- $x_m = \sum_{k=0}^{m} \alpha_m p_m$
- α_m were constructed by optimal searches in the p_m directions
- Equivalently, $\sum_i \alpha_i (Ap_i, p_j) = (b, p_j) \quad \forall j$

Conjugate-Gradient Method

The Conjugate Gradient (CG) method works for general symmetric positive matrices (SPD).

Moreover, CG is the *de facto* method!

Indefinite Problems

The same apply to symmetric indefinite problems!

- The minimum residual method (MinRes) is good!
- It requires 2 search vectors (in contrast CG only requires 1)
- Another alternative is Symlq
- Preconditioning requires a little more care

Nonsymmetric problems

- Nonsymmetric problems are much harder
- No best method!

Categories:

- Symmetric and positive matrices (Good)
- Symmetric and indefinite matrices (Good)
- Nonsymmetric and positive matrices (Bad)
- Nonsymmetric and indefinite matrices (Ugly)

Four Families

Essentially four families of methods:

- Methods for the normal equation: CGN, CGNE and LSQR
- Orthogonalization methods: Orthomin, Orthodir, GMRES, GCR, ..
- Biorthogonalization method: Biomin, BiCGStab, QMR, CGS, ..
- Other methods: Usymlq and Usymqr

CGN

Conjugate-Gradient method on the normal equation is derived as follows. We want to solve the equation,

$$Ax = b \tag{3}$$

where A is a general matrix.

We multiply the equation with A^T and obtain

$$A^T Ax = A^T b \tag{4}$$

The matrix $A^T A$ is SPD!

CGN II

The matrix $A^T A$ is SPD!

- $(A^T Ax, y) = (Ax, Ay) = (x, A^T Ay)$
- $(A^T Ax, x) = (Ax, Ax) > 0$, if $x \neq 0$ and A has full rank

CGN III

We can apply the CG method on

$$A^T Ax = A^T b \tag{5}$$

- It works!
- $A^T A$ is never formed in practice!
- But it has poor convergence properties...

CGN IV

We remember the convergence estimate for the CG method

$$\|e_m\|_A \le (\frac{\sqrt{K(A)}-1}{\sqrt{K(A)}+1})^m \|e_0\|_A \tag{6}$$

It is better than the convergence obtained by the Richardson method

$$\|e_m\| \le (\frac{K(A)-1}{K(A)+1})^m \|e_0\| \tag{7}$$

The result for CG on the normal equation reads,

$$\|e_m\|_{A^TA} \le (\frac{\sqrt{K(A^TA)}-1}{\sqrt{K(A^TA)}+1})^m \|e_0\|_{A^TA} \tag{8}$$

This is similar to the Richardson iteration. Notice that this estimate is not sharp. CGN tends to accelerate in the course of the iteration.

CG

- Krylov subspaces, $K_n =< b, Ab, \ldots, (A)^{n-1}b >$
- We seek to minimize the residual

$$\|e_n\|_A = minimum \tag{9}$$

- Or equivalent,

$$r_n \perp < b, Ab, \ldots, A^{n-1}b > \tag{10}$$

- conjugate search vectors,

$$(Ap_m, p_k) = 0, fo^r\ m \neq k \tag{11}$$

CGN

- Krylov subspaces, $K_n =< A^T b, (A^T A) A^T b, \ldots, (A^T A)^{n-1} A^T b >$
- We seek to minimize the residual

$$\|r_n\| = minimum \tag{12}$$

- Or equivalent,

$$r_n \perp < A^T b, (A^T A) A^T b, \ldots, (A^T A)^{n-1} A^T b > \tag{13}$$

- "conjugate" search vectors,

$$(A^T A p_m, p_k) = 0, fo^r\ m \neq k \tag{14}$$

Orthogonalization Methods

Many variants, Orthomin, Orthores, GMRES etc.
GMRES:

- Krylov subspaces, $K_n =< b, Ab, \ldots, A^{n-1}b >$
- We seek to minimize the residual

$$||r_n|| = minimum \tag{15}$$

- Or equivalent,

$$r_n \perp < b, Ab, \ldots, A^{n-1}b > \tag{16}$$

- *Not* conjugate search vectors,

$$(Ap_m, p_k) \neq 0 \tag{17}$$

Notice that the Krylov subspace differ from CGN and that the search vectors are not orthogonal in the A inner product!

Orthogonalization Methods II

The problem is that there is no simple way to construct orthogonal search vectors!

:-(

- We are forced to do a Gram-Schmidt process

Recap From the CG Lecture

- The residuals r_j are linearly independent
- $(Ar_{k+1}, p_i) = 0$ for $i \leq k-1$
- $$p_{k+1} = r_{k+1} - \sum_{m=0}^{k} \frac{(Ar_{k+1}, p_m)}{(Ap_m, p_m)} p_m$$
- $p_{k+1} = r_{k+1} + \beta p_k$
- Thus the search vectors can be computed iteratively

Orthogonalization Methods III

- We must store all the previous search vectors, p_{k-1}
- The new search vector, p_k, should be orthogonal to the previous ones
- A standard trick is to *truncate* or *restart* the algorithm and keep the number of search vectors small.

Biorthogonalization Methods

Many variants, CGS, BCG, BiCGStab etc.
CGS and BCG:

- Krylov subspaces, $K_n =< b, Ab, \ldots, A^{n-1}b >$
- The orthogonality condition is modified,

$$r_n \perp < b, A^T b, \ldots, (A^T)^{n-1} b > \tag{18}$$

- Only three search vectors are needed
- These algorithms are not as stable as the orthogonalization methods
- These algorithms require 2 matrix vector products per iteration
- The residual may jump up and down!

Other Methods

- We will not go into the other methods: Unsymlq and Unsymqr.
- These are harder to understand!

Comparison

The methods may break down! (Blow up or stagnate)

- CGN is robust
- The orthogonalization methods are more robust than the biorthogonalization methods
- The biorthogonalization methods are often faster when not breaking down

In general, it is hard to say which method will perform better. Experiments are required.

A good preconditioner is necessary for all of the above methods!

Convergence Monitors

All the iterative methods need some kind of stopping criteria. There are many possible choices!
Absolute criteria:

- $||r_k||$
- $||Br_k||$, where B is the preconditioner
- $||x_k - x_k||$

Relative criteria:

- $\frac{||r_k||}{||r_0||}$
- $\frac{||Br_k||}{||Br_0||}$
- $\frac{||x_k - x_k||}{||x_0||}$

Choosing wrong criteria may give you a headache!

Convergence Monitors II

- The true residual,$r_k = b - Ax_k$, involves a matrix vector product and is therefore usually avoided
- The residual is usually computed iteratively (CG: $r_k = r_{k-1} - \alpha_k t_k$)
- A problem with the recursively computed residual is round-off errors
- This may cause problems (in particular with a to strict convergence criteria)

Implementation

All the Conjugate-Gradient like methods are easy to implement.

- Short (20 code lines) algorithms
- Diffpack has lots of them
- Find them on the web:
  - www.netlib.org
  - www.mgnet.org

Further Reading

- A. M. Bruaset, *A Survey of Preconditioned Iterative Methods*, Addison-Wesley Pitman, 1995.
- H. P. Langtangen, *Computational Partial Differential Equations – Numerical Methods and Diffpack Programming*, Springer, 2003
- N. M. Nachtical, S. C. Reddy and L. N. Trefethen, *How Fast are Nonsymmetric Matrix Iterations*, SIAM J. Matrix Anal. Appl., 1992, vol 13, pp. 778-795.