An optimal Q-OR Krylov subspace method for solving linear systems

Gérard MEURANT

Toulouse, SD 2017
1. Introduction
2. Q-OR methods
3. Properties of Q-OR methods
4. Construction of “good” bases for Q-OR
5. A non-orthogonal optimal basis
6. Properties of the optimal basis
7. The algorithm
8. Numerical experiments
Many Krylov methods have been proposed over the years for solving linear systems $Ax = b$

Many of them can be classified as quasi-orthogonal (Q-OR) or quasi-minimum residual (Q-MR)

Q-OR: FOM, BiCG, Hessenberg, . . .

Q-MR: GMRES, QMR, CMRH, . . .
Whatever their definition, these methods share many fundamental properties


They differ by the basis of the Krylov space that is constructed:
- orthogonal for FOM/GMRES,
- bi-orthogonal for BiCG/QMR,
- based on an LU factorization for Hessenberg/CMRH

We use $x_0 = 0$ and assume that $\|b\| = 1$, the Krylov space is

$$\mathcal{K} = \{ b, Ab, \ldots, A^{n-1}b \}$$
Q-OR methods

We assume that the vectors spanning \( \mathcal{K} \) are linearly independent and that we have a basis \( V \) of the Krylov space (with columns of unit norm) such that \( K = VU \) with

\[
K = \begin{pmatrix} b & Ab & A^2b & \cdots & A^{n-1}b \end{pmatrix}
\]

\( V \) nonsingular with \( v_1 = b \) and \( U \) upper triangular.

We define \( H = UCU^{-1} \), upper Hessenberg, where \( C \) is the companion matrix for the eigenvalues of \( A \). We have \( AK = KC \).

As a consequence \( AV = VH \). It yields an Arnoldi-like relation

\[
AV_k = V_k H_k + h_{k+1,k} v_{k+1} e_k^T
\]

where \( V_k \) is the matrix of the \( k \) first columns of \( V \) and \( H_k \) is the \( k \times k \) principal matrix of \( H \).
The iterates are defined as

\[ x_k = V_k y^{(k)} \]

The residual \( r_k = b - Ax_k \) is

\[ r_k = V_k e_1 - AV_k y^{(k)} \]
\[ = V_k (e_1 - H_k y^{(k)}) - h_{k+1,k} y^{(k)}_k v_{k+1} \]

The Q-OR method is defined (provided that \( H_k \) is nonsingular) by

\[ H_k y^{(k)} = e_1 \]

This annihilates the first term in the residual and the residual norm is \( h_{k+1,k} |y^{(k)}_k| \)
Properties of Q-OR methods

Let $r_k^O$ be the residual vectors of the Q-OR method. Whatever the basis is, we can show by induction that

$$
|(U^{-1})_{1,k}| = |v_{1,k}| = \frac{1}{\|r_{k-1}^O\|}
$$

The inverses of the Q-OR residual norms can be read from the first row of the inverse of $U$ (remember that $K = VU$).

For this property and more see:

Construction of “good” bases

We would like to find bases which lead to a “good” (or even optimal) convergence of the Q-OR method

- The matrix $V$ of the basis is related to the Krylov matrix $K$ by $K = VU$ with $U$ upper triangular

- The entries of the first row of $U^{-1}$ are the inverses of the Q-OR residual norms (up to the sign)

Constructing a “good” basis may seem easy since one can think that we can just construct any upper triangular matrix $U^{-1}$ with entries of large modulus on the first row

But, it is not so since the columns of $V$ have to be of unit norm

Moreover, it is not recommended to use the matrix $U$ numerically
A non-orthogonal optimal basis

Can we construct a basis such that Q-OR minimizes the residual norms?

We would like to construct $H$ column by column without using $U$. We have

$$H_j = U_j E_j U_j^{-1} + \begin{pmatrix} 0 & \cdots & 0 & \frac{1}{u_{i,j}} U_{1:j,j+1} \end{pmatrix}$$

$E_j$ down-shift matrix

It yields

$$\sum_{j=1}^{k+1} \nu_{1,j} h_{j,k} = 0 \Rightarrow \nu_{1,k+1} = -\frac{1}{h_{k+1,k}} \sum_{j=1}^{k} \nu_{1,j} h_{j,k}$$
At step $k$ we have already computed $\nu_{1,j}, j = 1, \ldots, k$ and we would like to choose $h_{j,k}, j = 1, \ldots, k + 1$ to maximize the absolute value of $\nu_{1,k+1}$

But $h_{k+1,k}$ has to be chosen to obtain a vector $\nu_{k+1}$ of unit norm

Let

$$\tilde{\nu} = A\nu_k - \sum_{j=1}^{k} h_{j,k} \nu_j$$

the next basis vector is $\nu_{k+1} = \tilde{\nu} / h_{k+1,k}$ with $h_{k+1,k} = \|\tilde{\nu}\|$.

$$|\nu_{1,k+1}| = \frac{|\nu^T y|}{\|d - By\|}$$

with

$$d = A\nu_k, \quad B = V_k = (\nu_1 \cdots \nu_k), \quad y = (h_{1,k} \cdots h_{k,k})^T$$

$$\nu = (\nu_{1,1} \cdots \nu_{1,k})$$

We need to minimize $1/|\nu_{1,k+1}|^2$
We would like to solve

$$\gamma_{opt} = \min_{y \in \mathbb{R}^k, \nu^Ty \neq 0} \frac{\|d - By\|^2}{(\nu^Ty)^2}$$

The minimum is given by

$$\gamma_{opt} = \frac{\alpha}{\alpha \nu^T (B^T B)^{-1} \nu + \omega^2}$$

with $\alpha = d^T d - d^T B (B^T B)^{-1} B^T d$ and $\omega = d^T B (B^T B)^{-1} \nu$

Moreover, if $\omega \neq 0$, a solution $y_{opt}$ of the minimization problem is given by

$$y_{opt} = (B^T B)^{-1} B^T d + \frac{\alpha}{\omega} (B^T B)^{-1} \nu$$

$$= s + \frac{\alpha}{\omega} t$$

In our case for computing the solution we have to solve

$$(V_k^T V_k)s = V_k^T A v_k, \quad (V_k^T V_k)t = \nu$$
Properties of the optimal basis

\[ V_{k+1}^T v_{k+1} = \frac{1}{\nu_{1,k+1}} \begin{pmatrix} \nu_{1,1} \\ \vdots \\ \nu_{1,k} \\ \nu_{1,k+1} \end{pmatrix} \]

\[ V_k^T V_k = \begin{pmatrix} \frac{1}{\nu_{1,2}} & \frac{1}{\nu_{1,3}} & \frac{1}{\nu_{1,2}} & \cdots & \frac{1}{\nu_{1,k}} \\ \frac{1}{\nu_{1,3}} & \frac{1}{\nu_{1,2}} & \frac{1}{\nu_{1,3}} & \cdots & \frac{1}{\nu_{1,k}} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{1}{\nu_{1,k}} & \frac{1}{\nu_{1,k}} & \frac{1}{\nu_{1,k}} & \cdots & \frac{1}{\nu_{1,k}} \end{pmatrix} \]

When the method converges, the basis is more and more orthogonal.
The inverse of $V_k^T V_k$ is tridiagonal and the matrix $V_k^T A V_k$ is upper triangular.

It means that we have constructed a right-conjugate direction method.

Moreover

$$t = (V_k^T V_k)^{-1} \nu = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ \nu_{1,k} \end{pmatrix}$$

This relation is used to simplify the construction of the basis vectors.
We can simplify the formulas for the new vector

$$\omega = t^T V_k^T A v_k = \nu_{1,k} v_k^T A v_k$$

Let $$y_{opt} = s + \frac{\alpha}{\omega} t$$

$$\tilde{v} = A v_k - V_k y_{opt}$$

$$= A v_k - V_k s - \frac{\alpha}{\omega} V_k t$$

$$= A v_k - V_k s - \frac{\alpha}{\omega} \nu_{1,k} v_k$$

$$= A v_k - V_k s - \frac{\alpha}{v_k^T A v_k} v_k$$

and

$$h_{1:k,k} = s + \beta e_k, \quad \beta = \frac{\alpha}{v_k^T A v_k}$$
The Q-OR optimal algorithm

We compute incrementally the inverses of the Cholesky factors of $V_k^T V_k$

Let $v_k^A = A v_k$

1- $v_k^V = V_{k-1}^T v_k$, $v_k^{tA} = V_k^T v_k^A$

2- $\ell_k = \tilde{L}_{k-1} v_k^V$, $y_k^T = \ell_k^T \tilde{L}_{k-1}$

3- if $\ell_k^T \ell_k < 1$, $\ell_{k,k} = \sqrt{1 - \ell_k^T \ell_k}$, else $(p_k^v)^T = y_k^T V_{k-1}^T$

$\ell_{k,k} = \| v_k - p_k^v \|$ end
4- 

\[ \tilde{L}_k = \begin{pmatrix} \tilde{L}_{k-1} & 0 \\ -\frac{1}{\ell_{k,k}} y_k^T & \frac{1}{\ell_{k,k}} \end{pmatrix} \]

5- \( \ell_A = \tilde{L}_k v_k^T, \ s = \tilde{L}_k^T \ell_A \)

6- \( \alpha = (v_k^A)^T v_k^A - \ell_A^T \ell_A, \ \beta = \frac{\alpha}{(v_k^A)_k} \)

7- 

\[ h_{1:k,k} = \begin{pmatrix} h_{1,k} \\ \vdots \\ h_{k,k} \end{pmatrix} = s + \beta e_k \]
\[ \tilde{v} = v_k^A - V_k h_{1:k,k}, \quad h_{k+1,k} = \|\tilde{v}\|, \quad \nu_{1,k+1} = -\frac{1}{h_{k+1,k}} \nu^T h_{1:k,k} \]

\[ \nu = \begin{pmatrix} \nu_{1,1} & \cdots & \nu_{1,k+1} \end{pmatrix}^T \]

9- \[ v_{k+1} = \frac{1}{h_{k+1,k}} \tilde{v} \quad \text{and} \quad v_{k+1}^A = Av_{k+1} \]

10- if needed, solve \( H_k y^{(k)} = \|b\|e_1 \) using Givens rotations, \( x_k = V_k y^{(k)} \)

In this algorithm almost everything is expressed in terms of matrix-vector products
Numerical experiments

**SUPG scheme (Streamwise upwind Galerkin)**
Convection-diffusion equation in a square with a mesh size of $1/41$
The diffusion coefficient is $\nu = 0.01$
This matrix is of order 1600 and has 13924 non zero entries. Its norm is $4.8716 \times 10^{-2}$ and the condition number is 40.518

Difference of the true residual norms of **GMRES-MGS and Q-OR**
optimal, supg 1600
True residual norms of **GMRES-MGS** (blue) and **Q-OR-opt** (red), 
\( \text{supg } 1600, \ n = 1600 \)
True residual norms for $k = 200$

- GMRES-CGS $1.54043 \times 10^{-13}$
- GMRES-CGS with reorthogonalization $7.05585 \times 10^{-15}$
- GMRES-CGS with double reorthogonalization $7.23790 \times 10^{-15}$
- GMRES-MGS $1.33776 \times 10^{-14}$
- GMRES-MGS with reorthogonalization $6.70649 \times 10^{-15}$
- GMRES-MGS with double reorthogonalization $6.70339 \times 10^{-15}$
- GMRES-Householder $2.03961 \times 10^{-14}$
- QOR opt $5.50626 \times 10^{-15}$
Could we use the fact that \((V_k^T V_k)^{-1}\) is tridiagonal?

We can compute the non-zero entries of \((V_k^T V_k)^{-1}\).

Previously we used the Cholesky factors of the matrix \((V_k^T V_k)^{-1}\) to solve \(V_k^T V_k s = v_k^{tA}\).

In theory these factors are bidiagonal matrices. However, we computed all their entries for the sake of numerical stability.

Now we would like to investigate to what extent we can use the fact that \((V_k^T V_k)^{-1}\) is tridiagonal to compute the vector \(s\).

This would save many dot products.
fs 680 1c, true residual norms, **GMRES -MGS**(plain) and **Q-OR-opt-trid** (dashed)
The problem is that the values of $\nu_{1,j}$ which are used to compute the inverse of the matrix $V_k^T V_k$ are not directly linked to the computed vectors $v_j$

After a while there is a discrepancy between $(V_k^T V_k)^{-1}$ and what is computed with the $\nu_{1,j}$'s

We can compute the relative residual norms in two ways: The first one is $\frac{1}{|\nu_{1,k+1}|}$

Let $\tilde{r}_0$ be the residual at the beginning of the cycle, the second way of computing the relative residual norm is obtained from solving $H_k y^{(k)} = \|b\| e_1$ which gives $h_{k+1,k} |y_k^{(k)}| / \|\tilde{r}_0\|$.

A simple remedy to our problems is to restart the algorithm when there is a too large difference between the two ways of estimating the relative residual norms.
supg 1600, true residual norms, \textbf{GMRES -MGS}(plain) and \textbf{Q-OR-opt-trid} with restart $\varepsilon_\nu = 10^{-2}$ (dashed)
However, things are not always so nice...

fs 680 1c, true residual norms, \textbf{GMRES -MGS}(plain) and \textbf{Q-OR-opt-trid} with restart $\varepsilon_\nu = 10^{-2}$ (dashed)
Conclusion

Using the properties of the Q-OR methods we were able to construct a non-orthogonal basis for which Q-OR gives the same residual norms as GMRES.

The algorithm is slightly more expensive than GMRES but it can be simplified using automatic restarts.

It is more parallel than GMRES-MGS and most of the operations are matrix-vector products.

In many cases the maximum attainable accuracy is better than with GMRES-MGS.

It remains to study its stability in finite precision arithmetic and to see how to use it on parallel computers.