The SCBiCG class of Methods for Solving Complex Symmetric Linear Systems

Xian-Ming Gu[1,2]

Joint work: T.-Z. Huang (UESTC, China), M. Clemens (Bergische Universität Wuppertal, Germany), T. Sogeba (Aichi Prefectural University, Japan), L. Li (UESTC, China) and B. Carpentieri (RUG, Netherlands)

1. School of Math. Sci., University of Electronic Science and Technology of China
2. Institute of Mathematics and Computing Science, Rijksuniversiteit Groningen

February 2, 2015

Outline

1. Introduction
2. The SCBiCG class of iterative methods
3. Polynomial analysis:
    - COCG
    - BiCGCR/COCR
    - Mathematical equivalence of COCR and BiCGCR
4. Numerical examples and concluding remarks

Introduction

Consider the iterative solutions of following linear systems

$$Ax = b, \quad A \neq A^H,\ A = A^T \in \mathbb{C}^{m\times m}, \quad b \in \mathbb{C}^m. \tag{1}$$

Quite often modern Krylov subspace methods can be used:

- Short recurrence-type methods: BiCG, QMR, CGS, BiCGSTAB(ℓ), TFQMR, IDR(s), CORS/BiCORSTAB. . . .
- Arnoldi-based methods: FOM/GMRES(m), FGMRES, GCR, GMRESR. . . .

✠ Comments:

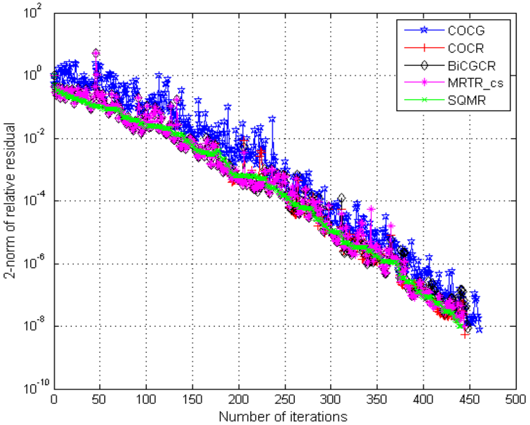
- Ignore the given symmetry of A.
- Need more than **ONE** matrix-vector product (referred to as MVP)

Introduction

Some particular iterative solvers are proposed which exploit the symmetry of A:

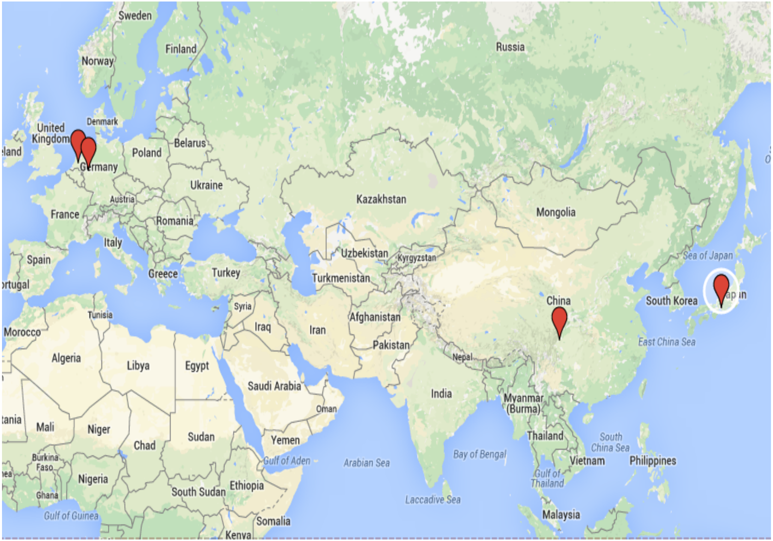
- COCG (H.A. van der Vorst and J. Mellissen, IEEE Trans. Magn., 1990)
- SQMR (R.W. Freund, SISC, 1992)
- SCBiCG (M. Clemens et al., 7th Copper Mountain Conference, 1996)
- BiCGCR (M. Clemens et al., IEEE Trans. Magn., 1998)
- CSYM (A. Bunse-Gerstner and R. Stöver, LAA, 1999)
- COCR (T. Sogabe and S.-L. Zhang, JCAM, 2006)
- MRTR_cs (A. Shiode, et al., Asia Simulation Conference 2006, 2007)
- CS-MINRES-QLP (S.-C. Choi, May 23, 2013, arXiv preprint)
- SCBiCG(Γ, n) (revisited by X.-M. Gu et al., CPC, accepted, 26 Jan. 2015)

A 3D Helmholtz problem ($n = 31, k = 20$)



PS: COCG(iter = 461, TRR = -8.1085); COCR(iter = 445, TRR = -8.2725); BiCGCR(iter = 448, TRR = -8.0709); MRTR_cs(iter = 444, TRR = -8.0084); SQMR(iter = 441, TRR = -8.0032); CSYM(iter = 3939, TRR = -8.0003); CS-MINRES-QLP(iter > 3900, TRR = -7.9413).

Introduction



Purpose: Revisit and summarize the independent work in three countries as the framework of China.

The SCBiCG class of methods

CBiCG (1 + 1 mvps) $\xrightarrow[A = A^T]{\text{Complex symm.}}$ SCBiCG(Γ, n) (1 mvp)

1) CBiCG is proposed by D.A.H. Jacobs, in 1981 and 1986. (Need: $A^H v$, Av)
2) SCBiCG class of methods is proposed by M. Clemens in 1996 . (Only need: Av)

Two bi-orthogonality conditions of CBiCG hold (Y. Saad, 1996):

$$\begin{aligned} \langle r_i, \tilde{r}_j \rangle &= \tilde{r}_j^H r_i = 0 \ \ \forall\ i \neq j, & (2) \\ \langle A p_i, \tilde{p}_j \rangle &= \tilde{p}_j^H A p_i = 0 \ \ \forall\ i \neq j. & (3) \end{aligned}$$

Simplify CBiCG for symmetric complex matrices, we use the specific choice of the start pseudo-residual vector (SPRV)

$$\tilde{r}_0 = \Pi_n(\bar{A})\bar{r}_0, \tag{4}$$

in the CBiCG method, where $\Pi_n(z) = \sum_{i=0}^{n} c_i \cdot z^i$ is a polynomial of degree n with a fixed set of coefficients $\Gamma = \{c_0, c_1, \ldots, c_n\}, \quad c_i \in \mathbb{R}$.

The SCBiCG class of methods

The deliberate complex conjugation operation of the pseudo vectors reduces the inner product to a bilinear form $\langle x, \bar{y} \rangle = y^T x$, i.e.,

$$\alpha_k = \frac{\langle r_k, \Pi_n(\bar{A})\bar{r}_k \rangle}{\langle Ap_k, \Pi_n(\bar{A})\bar{p}_k \rangle} = \frac{r_k^T \Pi_n(A) r_k}{p_k^T A \Pi_n(A) p_k}, \tag{5}$$

$$\beta_k = \frac{\langle r_{k+1}, \Pi_n(\bar{A})\bar{r}_{k+1} \rangle}{\langle r_k, \Pi_n(\bar{A})\bar{r}_k \rangle} = \frac{r_{k+1}^T \Pi_n(A) r_{k+1}}{r_k^T \Pi_n(A) r_k}. \tag{6}$$

To avoid the additional MVPs for evaluating both $\Pi_n(A) r_{k+1} = \sum_{i=0}^{n} c_i A^i r_{k+1}$ and $A \cdot \Pi_n(A) p_k = \sum_{i=0}^{n} c_i A^{i+1} p_k$ per inner iteration step k, additional auxiliary vectors are introduced

$$a_k^i = A^i r_k, \quad i = 0, \ldots, n; \tag{7}$$

$$h_k^i = A^i p_k, \quad i = 0, \ldots, n+1, \tag{8}$$

for $k \in \mathcal{N}$.

The SCBiCG class of methods

The auxiliary vectors are updated in additional SAXPY per inner iteration, which yields

SCBiCG Algorithm

Choose x_0; $r_0 = b - Ax_0$; $p_0 = r_0$; $a_0^0 = r_0$; $h_0^0 = p_0$;

For $i = 0, 1, \ldots, n-1$ **do:**

$$a_0^{i+1} = Aa_0^i; \qquad h_0^{i+1} = a_0^{i+1};$$

EndDo

$h_0^{n+1} = Ah_0^n$;

For $k = 0, 1, \ldots$ **do until convergence or breakdown:**

$$\alpha_k = \frac{\sum_{i=0}^{n} c_i a_k^{i\ T} r_k}{\sum_{i=0}^{n} c_i p_k^T h_k^{i+1}};$$

$$x_{k+1} = x_k + \alpha_k p_k; \qquad r_{k+1} = r_k - \alpha_k h_k^1; \qquad a_{k+1}^0 = r_{k+1};$$

For $j = 1, \ldots, n$ **do:**

$$a_{k+1}^j = a_k^j - \alpha_k h_k^{j+1};$$

EndDo

$$\beta_k = \frac{\sum_{i=0}^{n} c_i a_{k+1}^{i\ T} r_{k+1}}{\sum_{i=0}^{n} c_i a_k^{i\ T} r_k};$$

$$p_{k+1} = r_{k+1} + \beta_k p_k; \qquad h_{k+1}^0 = p_{k+1};$$

For $j = 1, \ldots, n$ **do:**

$$h_{k+1}^j = a_{k+1}^j + \beta_k h_k^j;$$

EndDo

$$h_{k+1}^{n+1} = Ah_{k+1}^n;$$

EndDo

Polynomial analysis – COCG, COCR and BiCGCR

- The first specific choice of SPRV:

$$\tilde{r}_0 = \bar{r}_0, \qquad n = 0, \quad \Gamma = \{c_0 = 1\} \tag{9}$$

corresponds to SCBiCG and yields the so-called COCG method.

- Another specific choice of SPRV:

$$\tilde{r}_0 = \bar{A}\bar{r}_0, \quad n = 1, \quad \Gamma = \{0, 1\} \tag{10}$$

corresponds to SCBiCG and results in two methods:

  - Firstly, the BiCGCR method proposed by Clemens et al.
  - Later, the COCR method proposed by Sogabe and Zhang.

Both BiCGCR and COCR use different, but mathematically equivalent formulae for the coefficients α_k and β_k.

Polynomial analysis – COCG, COCR and BiCGCR

With the Krylov subspace $\mathcal{K}_j(A, r_0) = \mathrm{span}\{r_0, Ar_0, \ldots, A^{j-1}r_0\}$ and constraints subspace $\mathcal{L} = \mathcal{L}_j(A^H, \tilde{r}_0)$, these scalar parameters involved can be determined by the following orthogonality conditions:

$$x_j \in x_0 + \mathcal{K}_j(A, r_0), \quad r_{j+1} = b - Ax_{j+1} \perp \mathcal{L} \text{ and } Ap_{j+1} \perp \mathcal{L}. \tag{11}$$

- $\mathcal{L} = \mathcal{K}_n(\bar{A}, \bar{r}_0) \Rightarrow$ COCG;
- $\mathcal{L} = \mathcal{K}_n(\bar{A}, \bar{A}\bar{r}_0) \Rightarrow$ BiCGCR;
- $\mathcal{L} = \bar{A}\mathcal{K}_n(\bar{A}, \bar{r}_0) \Rightarrow$ COCR. (Hint: easily show that $\mathcal{L}^{\mathrm{BiCGCR}} = \mathcal{L}^{\mathrm{COCR}}$.)

BiCGCR is shown to coincide with CR for Hermitian matrices, which has a residual minimal property. This can be considered an explanation why BiCG-typical oscillations in the residual norm occurring with BiCGCR/COCR are smaller than those occurring in the intermediate residual norms during the iteration process.

The earlier published PBiCGCR only differs to this variant of PCOCR in the math. equiv. calculation of the scalar factors

$$\alpha_k^{PBiCGCR} = \frac{p_k^T AM^{-1} r_k}{p_k^T AM^{-1} A p_k}, \qquad \alpha_k^{PCOCR} = \frac{r_k^T M^{-1} AM^{-1} r_k}{p_k^T AM^{-1} A p_k}; \tag{12}$$

and

$$\beta_k^{PBiCGCR} = -\frac{p_k^T AM^{-1} AM^{-1} r_{k+1}}{p_k^T AM^{-1} A p_k}, \qquad \beta_k^{PCOCR} = \frac{r_{k+1}^T M^{-1} AM^{-1} r_{k+1}}{r_k^T M^{-1} AM^{-1} r_k} \tag{13}$$

within the inner iteration loop, but is identical otherwise.

Theorem

For all $k \in \mathcal{N}$

$$\alpha_k^{PBiCGCR} = \alpha_k^{PCOCR}, \qquad \beta_k^{PBiCGCR} = \beta_k^{PCOCR}. \tag{14}$$

♠ Due to time constraints, the detailed proof can be found in our accepted paper (*X.-M. Gu, M. Clemens, et al. 2015*).

Numerical examples

In our paper, we present the comparative study of recent iterative solutions to complex symmetric linear systems arising in many CEM problems. For simplicity, we only choose two representative test problems for investigating the conclusions.

- Example ♯ 1:
  - Test matrix comes from the HDG-$\mathbb{P}_1$ discretization of 3D time-harmonic Maxwell's equation, refer to *Li et al., J. Comput. Phys., 2014.*
  - Size of A: 26460×26460;
  - Preconditioner: SSOR(ω), $\omega = 0.6$ is selected experimentally;
  - Initial guess: $x_0 = 0$; Stop criterion: $\|r_k\|/\|r_0\| < tol = 10^{-8}$.
  - "*Iters*": number of iterations, $TRR := \log_{10}(\|b - Ax_m\|/\|b - Ax_0\|)$, "CPU": CPU time elapsed.

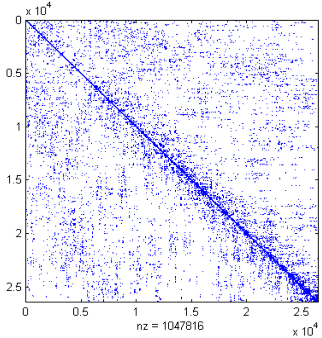


Figure: Sparsity of HDG matrix.

Numerical examples

Table: Numerical results of different iterative solvers for test problem.

Solver	*Iters*	TRR	CPU
BiCGCR	2978	-8.0385	19.14
COCR	2957	-8.0172	18.95
COCG	3183	-8.0122	19.61
SQMR	3006	-8.0156	20.01
PBiCGCR	**983**	-8.0216	**14.57**
PCOCR	1022	-8.1488	15.30
PCOCG	1027	-8.0687	14.83
PSQMR	1025	-8.0214	15.48

- COCG is not efficient unless the suitable preconditioners are used.
- PBiCGCR > PCOCR (particular case)
- SQMR is expensive in terms of CPU time.

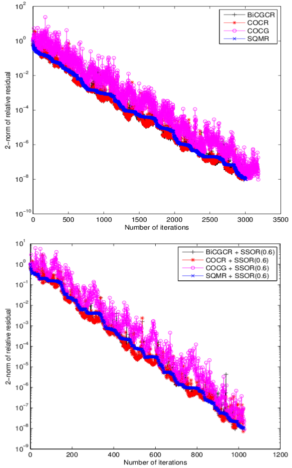


Figure: Convergence behaviors.

Numerical examples

Test matrices from the simulation of 2D electro-quasistatic complex frequency domain model, see *Schmidthäußler and Clemens, IEEE Trans. Magn., 2013*.

- Example $\sharp$ 2:
  - Size of test matrix A: 9738×9738;
  - It has the form: $A = A_1 + \iota\omega A_2$;
  - Preconditioner: Two-step precond. "SP-IC(ϵ)" is defined in our report.
    - Step 1: $P = A_1 + \omega A_2$, s.p.d.;
    - Step 2: IC(ϵ) of P, $\epsilon = 10^{-3}$.

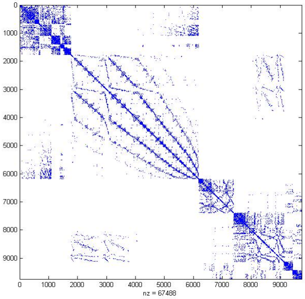


Figure: Sparsity of test matrix

Numerical examples

Table: Numerical results of different iterative solvers for test problem.

Solver	*Iters*	TRR	CPU
BiCGCR	7994	-8.0447	5.32
COCR	7836	-8.0001	4.94
COCG	8945	-8.0687	4.96
SQMR	8409	-8.0008	6.19
PBiCGCR	49	-8.4147	0.15
PCOCR	49	-8.4147	0.14
PCOCG	49	-8.3088	0.13
PSQMR	49	-8.2364	0.16

- COCG is not efficient unless the suitable preconditioners are used.
- PBiCGCR = PCOCR (Math. equiv.)
- SQMR is expensive in terms of CPU time.

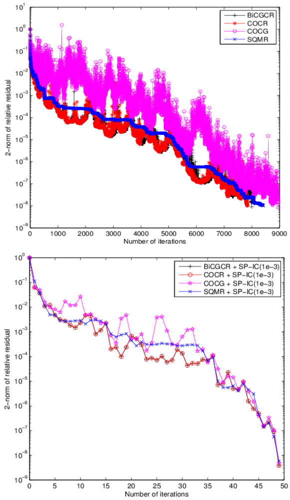


Figure: Convergence behaviors.

Conclusions

To sum up, BiCGCR and COCR indeed enjoy the same constraints subspace $\mathcal{L}$ and conjugate orthogonality conditions, also the iterative procedures are similar.

- BiCGCR is ofen more expensive than COCR due to having 3 dot products per iteration step. But they both need less iterations than COCG/SQMR;
- Let $M = I$, the α_k and β_k of BiCGCR and COCR are math. equiv.;
- COCR is a math. equiv. but numer. improved generalized version of BiCGCR;
- Without look-ahead techniques, SQMR is not cost-effective;
- For nearly dense (but still sparse) matrices, COCR/BiCGCR are more promising than COCG/SQMR;

Conclusions

Finally, we exploit the framework to obtain a new solver named BiCGCR2, which is proved that it is mathematically equivalent to BiCR and BiCOR.

Table: Different iterative solvers for `epb1` problem (Matrix: `epb1.mat`, 14734×14734, `b` choose randomly).

Solver	*Iters*	TRR	CPU
BiCGCR2	722	-8.2079	0.3713
BiCR	923	-8.3901	0.4622
BiCG	-	-7.7974	3.6515
BiCOR	921	-8.3643	0.8089
QMR	1066	-8.0120	1.0103

- BiCG is not efficient unless the suitable preconditioners are used.
- BiCGCR2 > BiCR & BiCOR
- QMR is expensive in terms of CPU time.

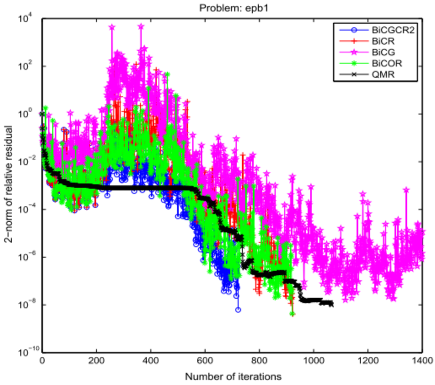


Figure: Convergence behaviors.

Conclusions

Table: Different iterative solvers with ILU(0) preconditioning for epb3 (Matrix: epb3.mat, 84617×84617, b also choose randomly).

Solver	*Iters*	TRR	CPU
PBiCGCR2	146	-8.6211	3.0328
PBiCR	187	-8.1432	3.8847
PBiCG	178	-8.0306	3.8680
PBiCOR	174	-8.2180	4.2685
PQMR	180	-8.1139	4.2240

- Convergence curve of PBiCG is still jagged.
- PBiCGCR2 > PBiCR & PBiCOR (more irregular)
- PQMR is expensive in terms of CPU time.
- The lase phase of convergence curve of P-BiCGCR2 is very attractive.

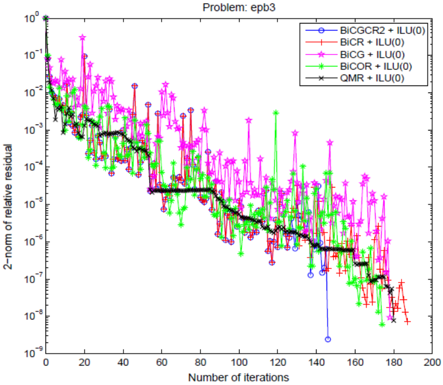


Figure: Convergence behaviors.

References

1. **X.-M. Gu**, M. Clemens, T.-Z. Huang, L. Li, The SCBiCG class of algorithms for complex symmetric systems with applications in several electromagneticmodel problems, Comput. Phys. Commun., *accepted*, 26 Jan. 2015, 25 pages.
2. **X.-M. Gu**, T.-Z. Huang, B. Carpentieri, BiCGCR2: A new extension of conjugate residual method for solving non-Hermitian linear systems, *submitted to journal*, January 2015, 21 pages.
3. **X.-M. Gu**, T.-Z. Huang, J. Meng, T. Sogabe, H.-B. Li, L. Li, BiCR-type methods for families of shifted linear systems, Comput. Math. Appl., Vol. 68, No. 7, 2014, pp. 746-758.

Thank you for your attention !