A QMR method based on an A-biorthogonalization process

CREST/COE Workshop
October 25th, 2006
The University of Tokyo

Satsuki Minami, Tomohiro Sogabe,
Masaaki Sugihara, and Shao-Liang Zhang
Dept. Appl. Phys. , Univ. of Tokyo

Table of Contents

- Background / Motivation

  Numerical methods for solving linear systems （The Krylov subspace methods）
- Theory
  - ・Bi-CG and QMR
  - ・Bi-CR and Our method
- Numerical experiments
- Summary and future plan

Numerical methods for solving linear systems

$$\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$$

The Krylov subspace method

$A = A^T$	$A \neq A^T$
CG	Bi-CG
CR	Bi-CR
MINRES	QMR
	GMRES

The Krylov subspace method

$$A\boldsymbol{x} = \boldsymbol{b}$$

Krylov subspace： $K_n(A; \boldsymbol{r}_0) = span(\boldsymbol{r}_0, A\boldsymbol{r}_0, \ldots, A^{n-1}\boldsymbol{r}_0)$

⬇

1．Generate basis vectors, $V_n = [\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_n]$

$V_n \longrightarrow$ **Basis algorithm**
（Lanczos process、Arnoldi process･･･）

2．Get approximate solution, $\boldsymbol{x}_n = \boldsymbol{x}_0 + V_n \boldsymbol{y}_n, \ \boldsymbol{y}_n \in \boldsymbol{R}^n$

$\boldsymbol{y}_n \longrightarrow$ **Residual condition**
（Orthogonal condition, Norm minimal condition･･･）

Motivation

Smooth

Residual / Basis	Orthogonal condition	QMR property
Bi-Lanczos process	Bi-CG	QMR
A -biortho-gonalization process	Bi-CR	Our method

Fast

Bi-CG and QMR

Basis algorithm：Bi-Lanczos process

- generates the basis vectors $\{v_1^L, \ldots, v_n^L\}$, $\{w_1^L, \ldots, w_n^L\}$ of two Krylov subspace, $K_n(A; r_0)$, $K_n(A^T; r_0^*)$
- Bi-orthogonality: $(w_i^L, v_j^L) = \delta_{ij}$
- Matrix form:

$$\begin{aligned} AV_n^L &= V_{n+1}^L \tilde{T}_n^L, \\ A^T W_n^L &= W_{n+1}^L (\tilde{T}_n^L)^T \end{aligned}$$

$\tilde{T}_n^L$: (n+1) × n tri-diagonal matrix with coefficients calculated in Bi-lanczos process

Bi-CG and QMR

Basis algorithm: $AV_n^L = V_{n+1}^L \tilde{T}_n^L$

Residual condition:

- Bi-CG → Orthogonal condition $r_n^B \perp K_n(A^T; r_0^*)$
- QMR → Quasi-minimal residual (QMR) property $\min_{y_n \in R^n} \|g_1 - \tilde{T}_n^L y_n\|$

Approximate solution: $x_n = x_0 + V_n^L y_n$

Residual vector: $r_n = \underbrace{V_{n+1}^L}_{\times \text{Orthogonal}} \underbrace{(g_1 - \tilde{T}_n^L y_n)}_{\text{Quasi-residual}}, \; g_1 = \|r_0\| e_1$

Bi-CR and Our method

Basis algorithm : A -biorthogonalization process

- generates the basis vectors $\{v_1^A, \ldots, v_n^A\}$, $\{w_1^A, \ldots, w_n^A\}$ of two Krylov subspace, $K_n(A; r_0)$, $K_n(A^T; r_0^*)$
- A -biorthogonality: $(w_i, Av_j) = \delta_{ij}$
- Matrix form:

$$
\begin{aligned}
AV_n^A &= V_{n+1}^A \tilde{T}_n^A, \\
A^T W_n^A &= W_{n+1}^A (\tilde{T}_n^A)^T
\end{aligned}
$$

$\tilde{T}_n^A$: (n+1) × n tri-diagonal matrix with coefficients calculated in A -biorthogonalization process

Bi-CR and Our method

Basis algorithm： $AV_n^A = V_{n+1}^A \tilde{T}_n^A$

Residual condition：

- Bi-CR ➡ Orthogonal condition $r_n^R \perp A^T K_n(A^T; r_0^*)$
- Our method ➡ QMR property $\min_{y_n \in R^n} \|g_1 - \tilde{T}_n^A y_n\|$

Residual vector: $r_n = V_{n+1}^A (g_1 - \tilde{T}_n^A y_n)$, $g_1 = \|r_0\| e_1$

Quasi-residual

On Algorithm

- QMR from Bi-Lanczos process / A-biorthogonalization process

  Basis algorithm ➡ Bi-Lanczos process, A-biorthogonalization process

- QMR from Bi-CG / Bi-CR

  Basis algorithm ➡ Bi-CG, Bi-CR

QMR from Bi-CG
（later QMRBi-CG）

$\boldsymbol{x}_0^Q$ is an initial guess, $\boldsymbol{r}_0^Q = \boldsymbol{b} - A\boldsymbol{x}_0^Q$,

set $\boldsymbol{r}_0^B = \boldsymbol{r}_0^Q$, choose $\boldsymbol{r}_0^{B*}$ (e.g., $\boldsymbol{r}_0^{B*} = \boldsymbol{r}_0^B$),

set $\boldsymbol{p}_{-1}^{B*} = \boldsymbol{p}_{-1}^B = \boldsymbol{d}_{-1}^Q = \boldsymbol{0}$,

$\beta_{-1} = 0$, $\tau_{-1} = \|\boldsymbol{r}_0^B\|$, $\vartheta_{-1} = 0$,

for $n = 0, 1, \ldots,$ until convergence, do:

$$\boldsymbol{p}_n^B = \boldsymbol{r}_n^B + \beta_{n-1}\boldsymbol{p}_{n-1}^B,\ \boldsymbol{p}_n^{B*} = \boldsymbol{r}_n^{B*} + \beta_{n-1}\boldsymbol{p}_{n-1}^{B*},$$

$$\alpha_n = \frac{(\boldsymbol{r}_n^{B*}, \boldsymbol{r}_n^B)}{(\boldsymbol{p}_n^{B*}, A\boldsymbol{p}_n^B)},$$

$$\boldsymbol{x}_{n+1}^B = \boldsymbol{x}_n^B + \alpha_n \boldsymbol{p}_n^B,$$

$$\boldsymbol{r}_{n+1}^B = \boldsymbol{r}_n^B - \alpha_n A\boldsymbol{p}_n^B,\ \boldsymbol{r}_{n+1}^{B*} = \boldsymbol{r}_n^{B*} - \alpha_n A^T \boldsymbol{p}_n^{B*},$$

$$\vartheta_n = \frac{\|\boldsymbol{r}_{n+1}^B\|}{\tau_{n-1}},\qquad c_n = \frac{1}{\sqrt{1+\vartheta_n^2}},\qquad \tau_n = \tau_{n-1}\vartheta_n c_n,$$

$$\boldsymbol{d}_n^Q = c_n^2\vartheta_{n-1}^2 \boldsymbol{d}_{n-1}^Q + c_n^2\alpha_n \boldsymbol{p}_n^B,\ (A\boldsymbol{d}_n^Q = c_n^2\vartheta_{n-1}^2 A\boldsymbol{d}_{n-1}^Q + c_n^2\alpha_n A\boldsymbol{p}_n^B,)$$

$$\boldsymbol{x}_{n+1}^Q = \boldsymbol{x}_n^Q + \boldsymbol{d}_n^Q,\qquad \boldsymbol{r}_{n+1}^Q = \boldsymbol{r}_n^Q - A\boldsymbol{d}_n^Q,$$

$$\beta_n = \frac{(\boldsymbol{r}_{n+1}^{B*}, \boldsymbol{r}_{n+1}^B)}{(\boldsymbol{r}_n^{B*}, \boldsymbol{r}_n^B)},$$

end

QMR from Bi-CR
(later QMRBi-CR)

$\boldsymbol{x}_0^{QR}$ is an initial guess, $\boldsymbol{r}_0^{QR} = \boldsymbol{b} - A\boldsymbol{x}_0^{QR}$,
set $\boldsymbol{r}_0^{R} = \boldsymbol{r}_0^{QR}$, choose $\boldsymbol{r}_0^{R*}$ (e.g., $\boldsymbol{r}_0^{R*} = \boldsymbol{r}_0^{R}$),
set $\boldsymbol{p}_{-1}^{R*} = \boldsymbol{p}_{-1}^{R} = \boldsymbol{d}_{-1}^{QR} = \boldsymbol{0}$,
$\beta_{-1} = 0$, $\tau_{-1} = \|\boldsymbol{r}_0^{R}\|$, $\vartheta_{-1} = 0$,
for $n = 0, 1, \ldots,$ until convergence, do:

$$\boldsymbol{p}_n^{R} = \boldsymbol{r}_n^{R} + \beta_{n-1}\boldsymbol{p}_{n-1}^{R}, \qquad \boldsymbol{p}_n^{R*} = \boldsymbol{r}_n^{R*} + \beta_{n-1}\boldsymbol{p}_{n-1}^{R*},$$
$$(A\boldsymbol{p}_n^{R} = A\boldsymbol{r}_n^{R} + \beta_{n-1}A\boldsymbol{p}_{n-1}^{R},)$$
$$\alpha_n = \frac{(\boldsymbol{r}_n^{R*}, A\boldsymbol{r}_n^{R})}{(A^T\boldsymbol{p}_n^{R*}, A\boldsymbol{p}_n^{R})},$$
$$\boldsymbol{x}_{n+1}^{R} = \boldsymbol{x}_n^{R} + \alpha_n\boldsymbol{p}_n^{R},$$
$$\boldsymbol{r}_{n+1}^{R} = \boldsymbol{r}_n^{R} - \alpha_n A\boldsymbol{p}_n^{R}, \qquad \boldsymbol{r}_{n+1}^{R*} = \boldsymbol{r}_n^{R*} - \alpha_n A^T\boldsymbol{p}_n^{R*},$$
$$\vartheta_n = \frac{\|\boldsymbol{r}_{n+1}^{R}\|}{\tau_{n-1}}, \quad c_n = \frac{1}{\sqrt{1+\vartheta_n^2}}, \quad \tau_n = \tau_{n-1}\vartheta_n c_n,$$
$$\boldsymbol{d}_n^{QR} = c_n^2\vartheta_{n-1}^2\boldsymbol{d}_{n-1}^{QR} + c_n^2\alpha_n\boldsymbol{p}_n^{R}, \ (A\boldsymbol{d}_n^{QR} = c_n^2\vartheta_{n-1}^2 A\boldsymbol{d}_{n-1}^{QR} + c_n^2\alpha_n A\boldsymbol{p}_n^{R},)$$
$$\boldsymbol{x}_{n+1}^{QR} = \boldsymbol{x}_n^{QR} + \boldsymbol{d}_n^{QR}, \qquad \boldsymbol{r}_{n+1}^{QR} = \boldsymbol{r}_n^{QR} - A\boldsymbol{d}_n^{QR},$$
$$\beta_n = \frac{(\boldsymbol{r}_{n+1}^{R*}, A\boldsymbol{r}_{n+1}^{R})}{(\boldsymbol{r}_n^{R*}, A\boldsymbol{r}_n^{R})},$$

end

Numerical Experiments

CPU	Intel (R) Xeon (TM) 2.66GHz
Memory	512MB
Compiler	Fortran77 Double Precision
Matrix A	Matrix Market
Initial guess $\boldsymbol{x}_0$	0
Vector $\boldsymbol{r}_0^*$	The same as $\boldsymbol{r}_0$
Right-hand side $\boldsymbol{b}$	Random
Convergence criterion	$\|\boldsymbol{r}_n\|/\|\boldsymbol{r}_0\| \leq 10^{-6}$(for WATT2) $\|\boldsymbol{r}_n\|/\|\boldsymbol{r}_0\| \leq 10^{-12}$(for others)

Results

Matrix	Iteration			
	Bi-CG	QMR	Bi-CR	QMR
BFW782A	[illegible]	[illegible]	[illegible]	[illegible]
CDDE5	[illegible]	[illegible]	[illegible]	[illegible]
DW2048	2295	2280	1716	1717
FS_760_1	163	163	161	161
FIDAP022	2554	2447	1805	1803
JPWH991	82	81	81	82
ORSIRR2	1206	1204	1163	1165
PDE2961	330	330	333	329
PDE900	[illegible]	[illegible]	[illegible]	[illegible]
SHERMAN5	[illegible]	[illegible]	[illegible]	[illegible]
WATT2	[illegible]	[illegible]	[illegible]	[illegible]

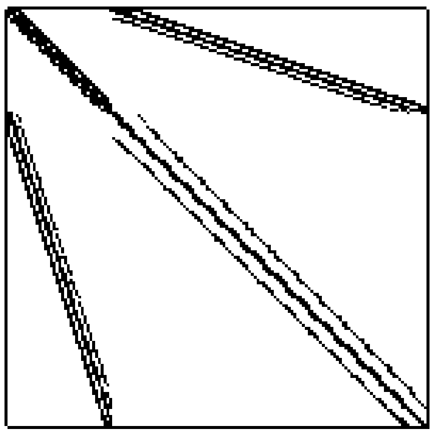
Almost the same iteration count as Bi-CR

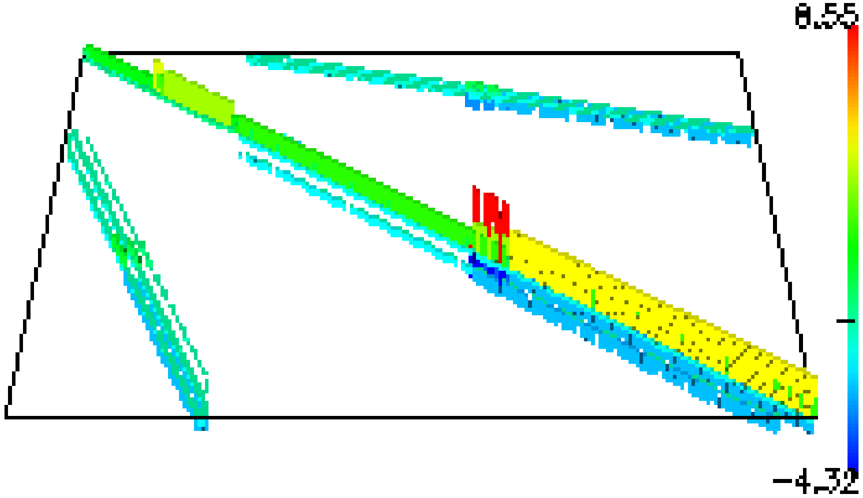
Faster convergence than QMRBi-CG

Almost the same iter. count

Matrix Market

BFW782A：Electrical engineering



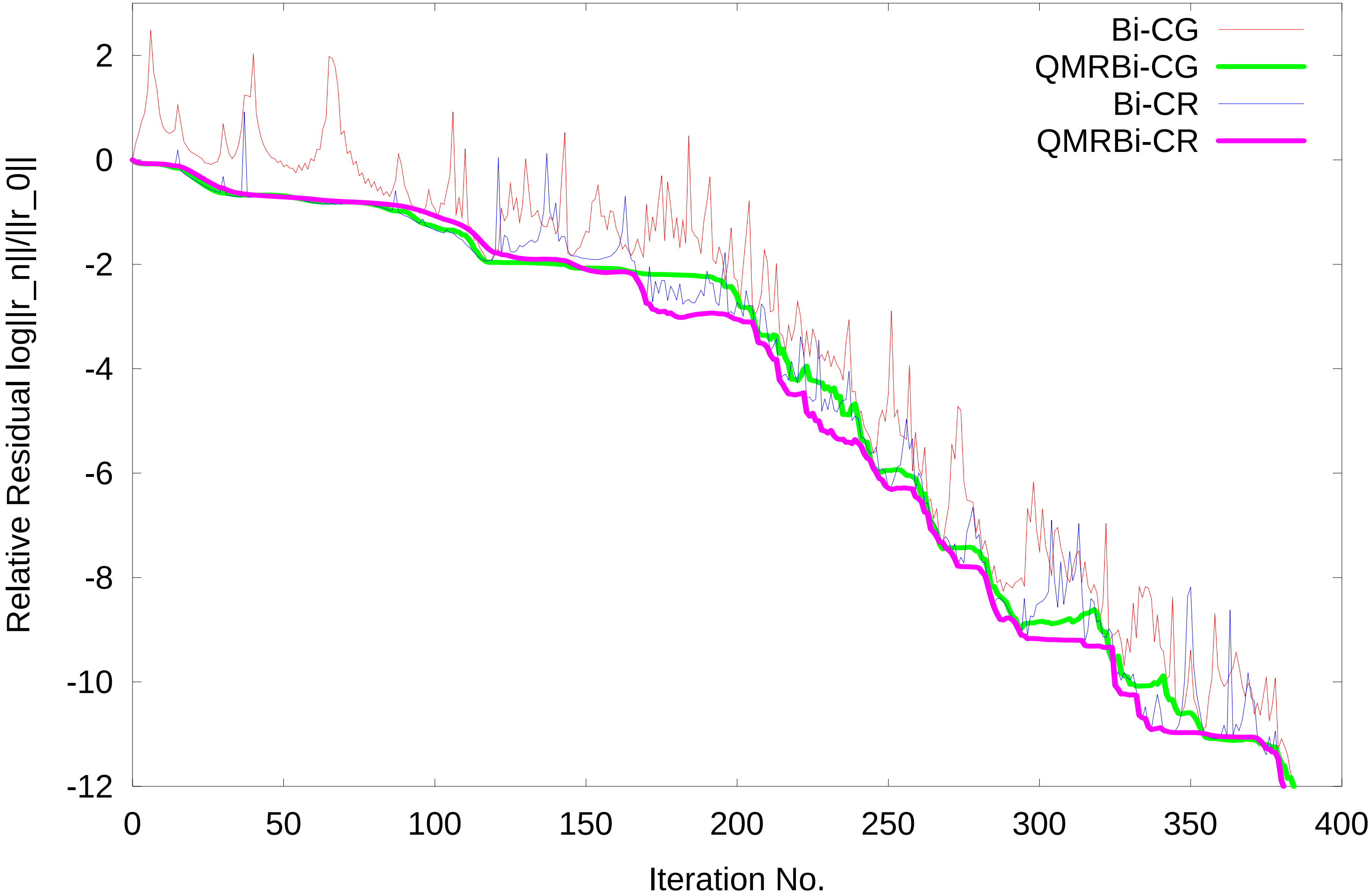


Size	Non Zeros	Type
782	7514	real unsymmetric

Almost the same iter. count

Matrix Market

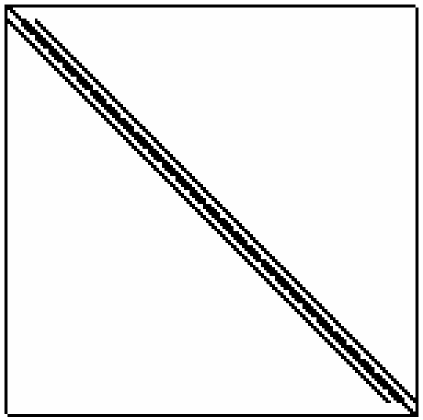
BFW782A : Electrical engineering

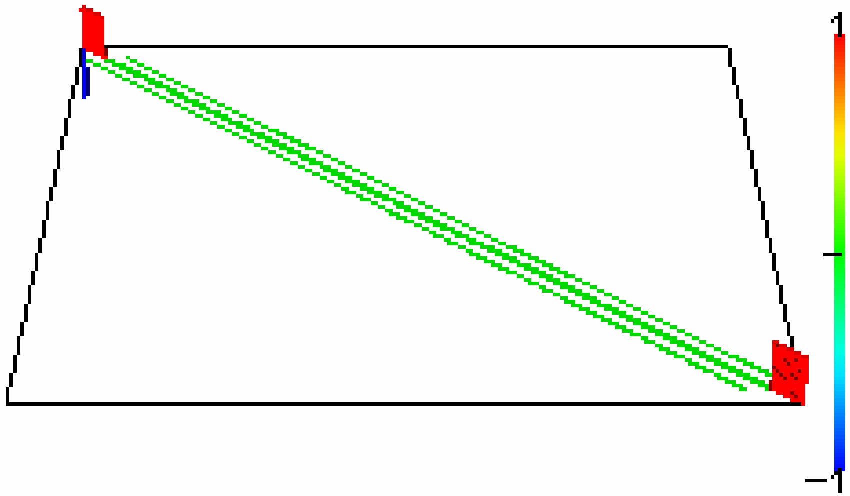


Faster convergence

Matrix Market

WATT2：Petroleum engineering



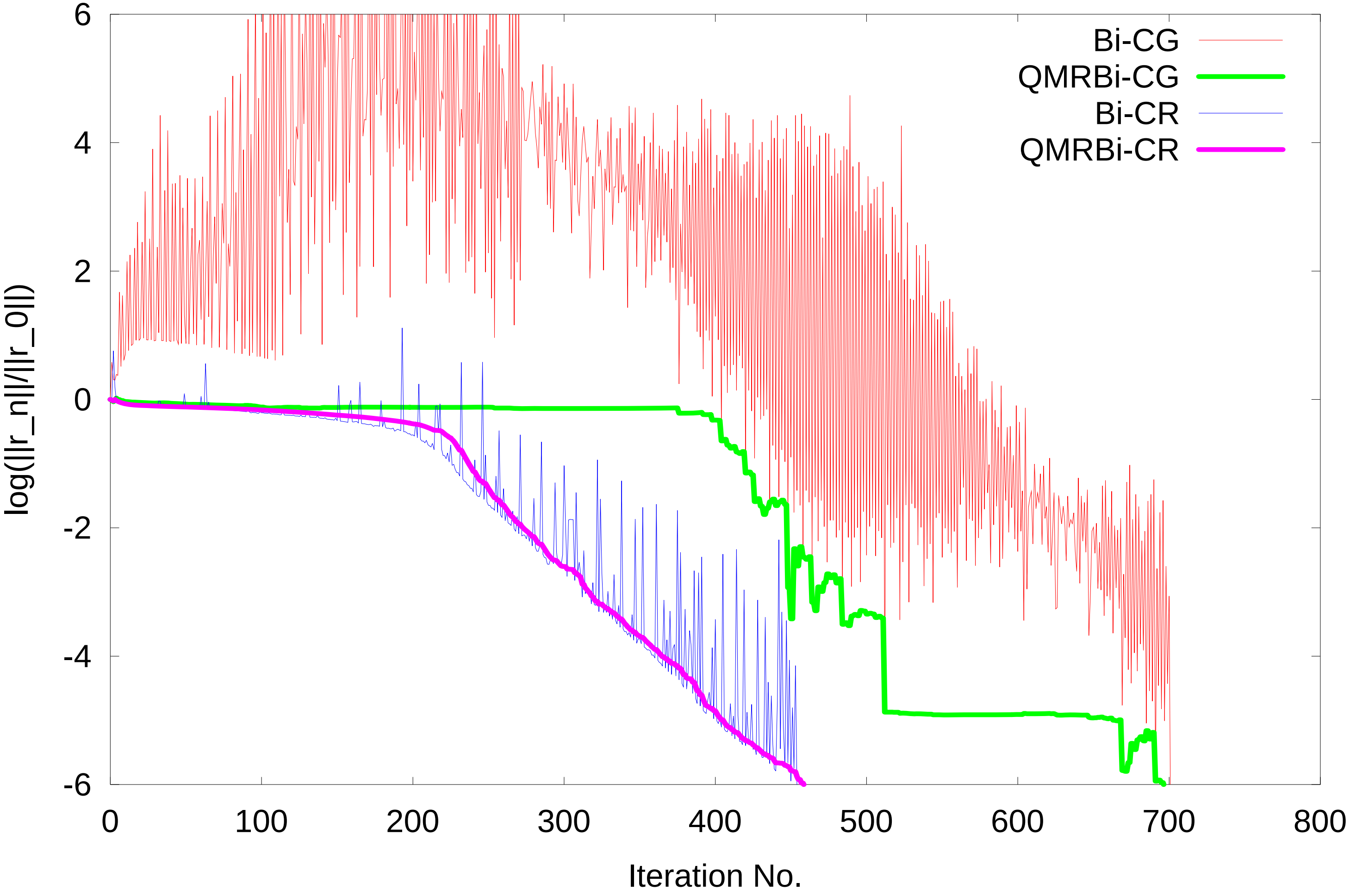


Size	Non Zeros	Type
1856	11550	real unsymmetric

Faster convergence

Matrix Market

WATT2：Petroleum engineering

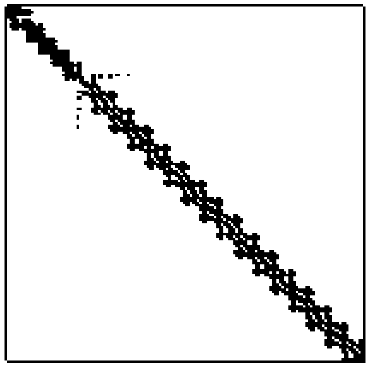


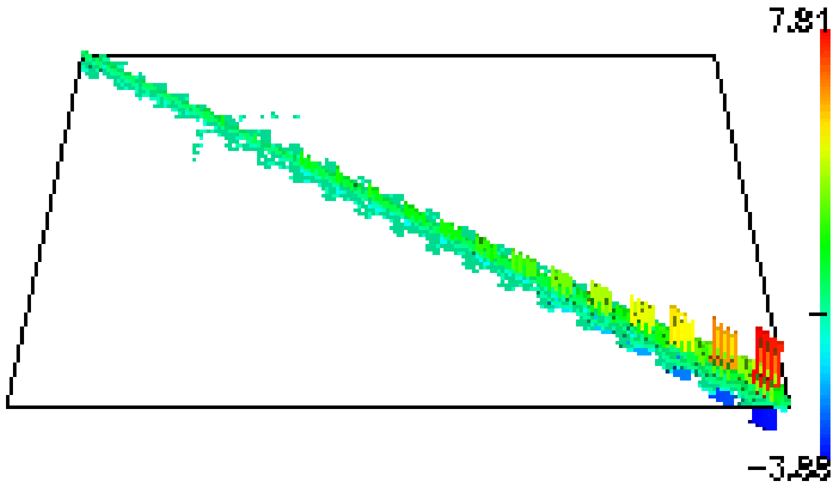
Summary and future plan

- We showed a QMR method based on an A-biorthogonalization process.
- From numerical experiments, we confirmed
  - ・the method has a smoother convergence property compared to Bi-CR.
  - ・the method has a faster convergence property compared to QMRBi-CG.
- More numerical experiments

Matrix Market

fidap022：Finite element modeling



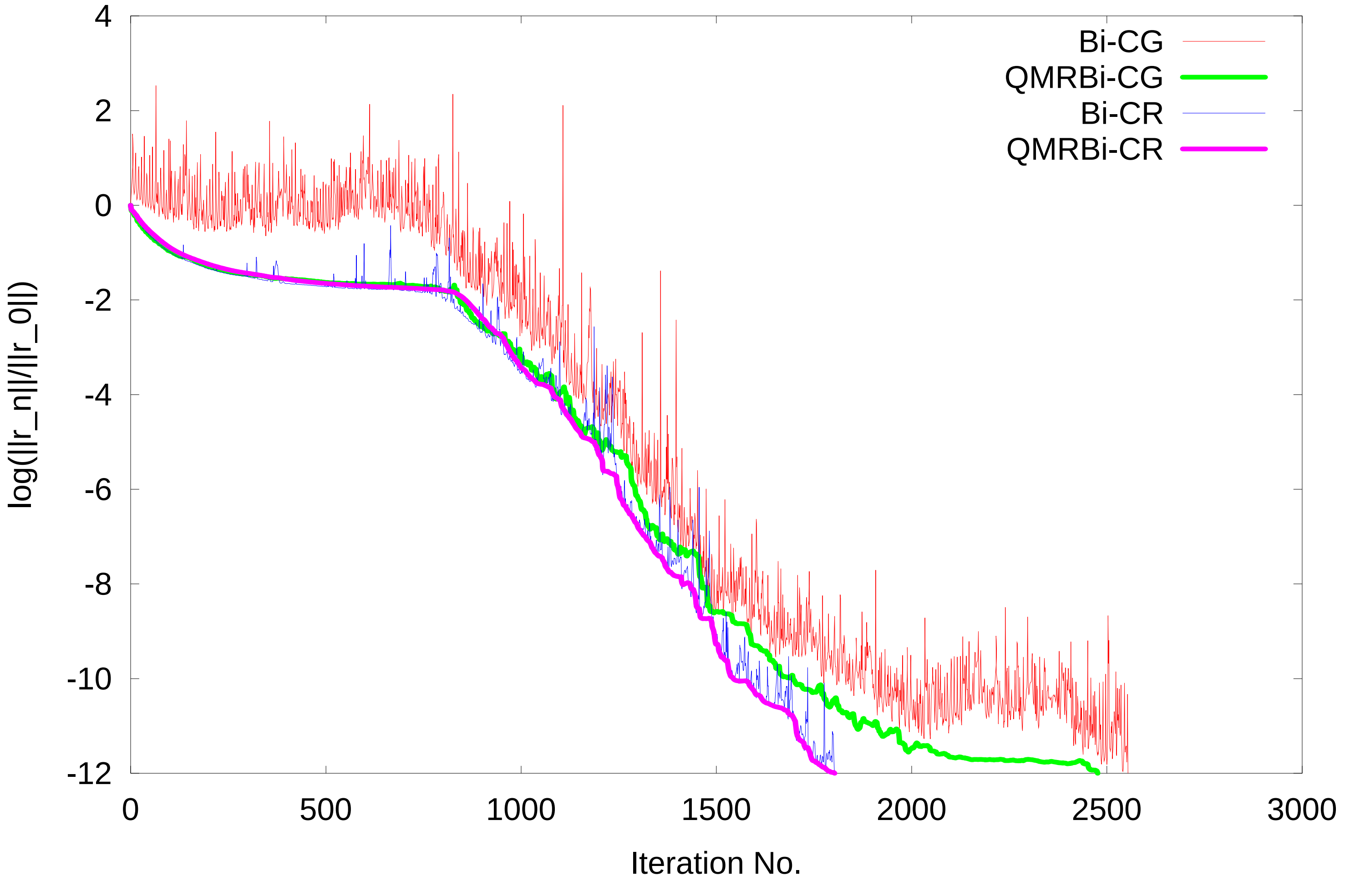


Size	Non Zeros	Type
839	22613	real unsymmetric

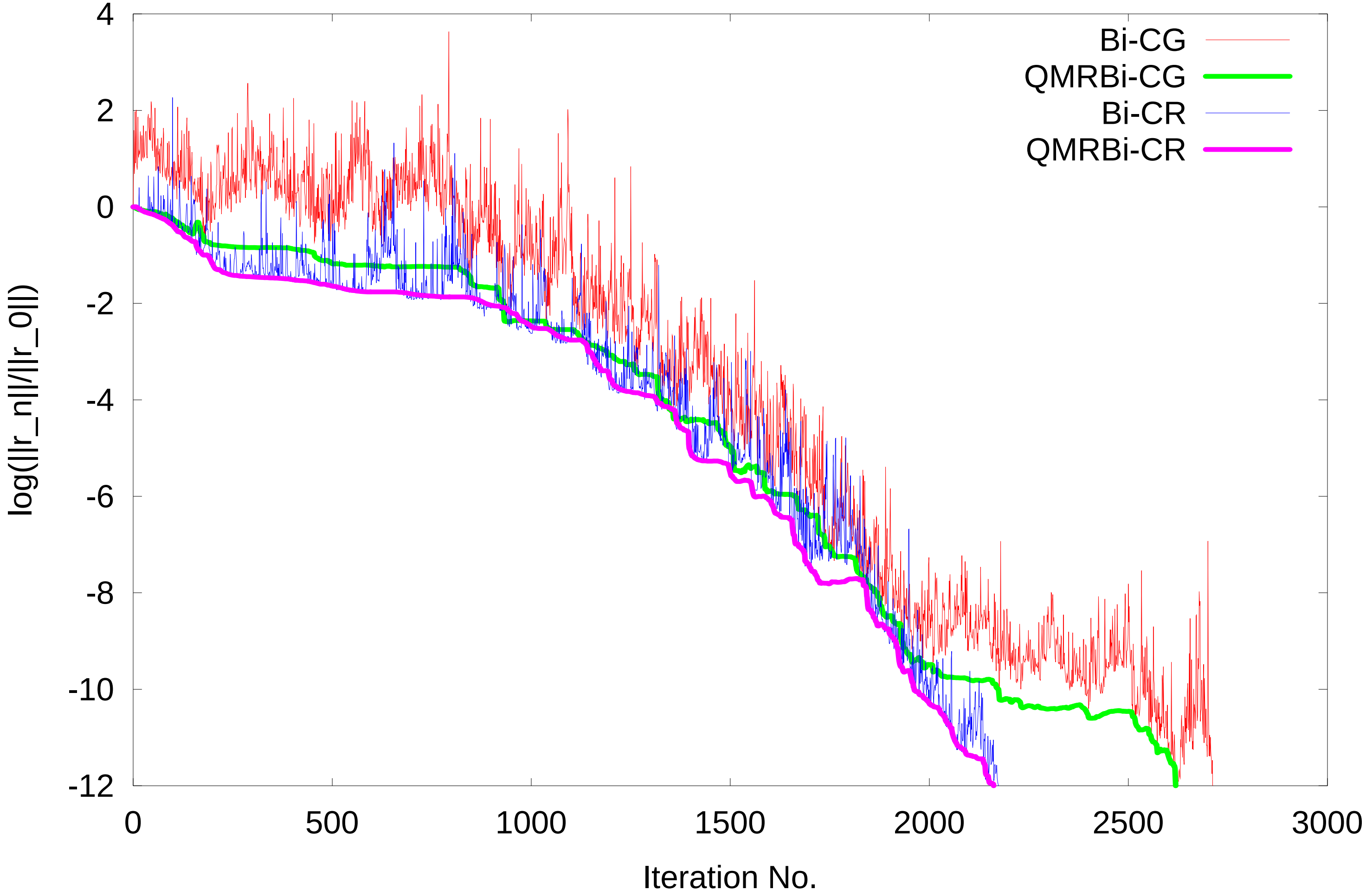
Faster convergence

Matrix Market

fidap022：Finite element modeling



sherman5



Bi-CG
QMRBi-CG
Bi-CR
QMRBi-CR
4
2
0
-2
-4
-6
-8
-10
-12
0
500
1000
1500
2000
2500
3000
log(||r_n||/||r_0||)
Iteration No.