ISM Preconditioners for Nonsymmetric Matrices

R. Bru* J. Marín* J. Mas* M. Tůma**

*Institut de Matemàtica Multidisciplinar
Universitat Politècnica de València
**Institute of Computer Science,
Academy of Sciences of the Czech Republic

Applied Linear Algebra
in honor of Hans Schneider
May 25, 2010

Outline

Outline

Outline

1. Motivation
   - Motivation
2. Improving the BIF Algorithm
   - Background
   - Improvements on BIF
3. The Nonsymmetric Balanced Incomplete Factorization (NBIF) Algorithm

Outline

Outline

Outline

Motivation

Understand relations between ILU and ISM and construct a preconditioner:

- Based on a new approximate factorization which computes direct and inverse factors
- This factorization is derived from Sherman-Morrison formula
- Factors obtained are those of ILU
- With droping rules that mutually balance the factors. Entries droped if

$$|l_{ij}|\|e_j^T L^{-1}\| \leq \tau \quad \text{and} \quad |\ell_{ij}|\|e_j^T L\| \leq \tau.$$

See Bollhöfer [LAA 2001, SISC 2003] and Bollhöfer and Saad [SIMAX 2002].

Motivation

Understand relations between ILU and ISM and construct a preconditioner:

- Based on a new approximate factorization which computes direct and inverse factors
- This factorization is derived from Sherman-Morrison formula
- Factors obtained are those of ILU
- With droping rules that mutually balance the factors. Entries droped if

$$|l_{ij}|\|e_j^T L^{-1}\| \leq \tau \quad \text{and} \quad |\ell_{ij}|\|e_j^T L\| \leq \tau.$$

See Bollhöfer [LAA 2001, SISC 2003] and Bollhöfer and Saad [SIMAX 2002].

Motivation

Understand relations between ILU and ISM and construct a preconditioner:

- Based on a new approximate factorization which computes direct and inverse factors
- This factorization is derived from Sherman-Morrison formula
- Factors obtained are those of ILU
- With droping rules that mutually balance the factors. Entries droped if

$$|l_{ij}|\|e_j^T L^{-1}\| \leq \tau \quad \text{and} \quad |\ell_{ij}|\|e_j^T L\| \leq \tau.$$

See Bollhöfer [LAA 2001, SISC 2003] and Bollhöfer and Saad [SIMAX 2002].

Motivation

Understand relations between ILU and ISM and construct a preconditioner:

- Based on a new approximate factorization which computes direct and inverse factors
- This factorization is derived from Sherman-Morrison formula
- Factors obtained are those of ILU
- With droping rules that mutually balance the factors. Entries droped if

$$|l_{ij}|\|e_j^T L^{-1}\| \leq \tau \quad \text{and} \quad |\ell_{ij}|\|e_j^T L\| \leq \tau.$$

See Bollhöfer [LAA 2001, SISC 2003] and Bollhöfer and Saad [SIMAX 2002].

Motivation

Understand relations between ILU and ISM and construct a preconditioner:

- Based on a new approximate factorization which computes direct and inverse factors
- This factorization is derived from Sherman-Morrison formula
- Factors obtained are those of ILU
- With droping rules that mutually balance the factors. Entries droped if

$$|l_{ij}| \| e_j^T L^{-1} \| \leq \tau \quad \text{and} \quad |\ell_{ij}| \| e_j^T L \| \leq \tau.$$

See Bollhöfer [LAA 2001, SISC 2003] and Bollhöfer and Saad [SIMAX 2002].

Outline

ISM factorization

Given A nonsingular and $s > 0$ (parameter)

$$z_k = e_k - \sum_{i=1}^{k-1} \frac{v_i^T e_k}{s r_i} z_i \quad \text{and} \quad v_k = y_k - \sum_{i=1}^{k-1} \frac{y_k^T z_i}{s r_i} v_i, \quad k = 1, \ldots, n \quad (1)$$

$r_k = 1 + y_k^T z_k / s = 1 + v_k^T e_k / s \ (\neq 0)$,
e_k and y_k columns of I and $Y = A^T - sI$, respectively.

Inverse Sherman-Morrison factorization

$$s^{-1} I - A^{-1} = s^{-2} Z_s D_s^{-1} V_s^T$$

$Z_s = [z_1, \ldots, z_n], \quad D_s = \mathrm{diag}(r_1, \ldots, r_n), \quad V_s = [v_1, \ldots, v_n]$

ISM factorization

Given A nonsingular and $s > 0$ (parameter)

$$z_k = e_k - \sum_{i=1}^{k-1} \frac{v_i^T e_k}{s r_i} z_i \quad \text{and} \quad v_k = y_k - \sum_{i=1}^{k-1} \frac{y_k^T z_i}{s r_i} v_i, \quad k = 1, \dots, n \quad (1)$$

$r_k = 1 + y_k^T z_k / s = 1 + v_k^T e_k / s \ (\neq 0)$,
e_k and y_k columns of I and $Y = A^T - sI$, respectively.

Inverse Sherman-Morrison factorization

$$s^{-1} I - A^{-1} = s^{-2} Z_s D_s^{-1} V_s^T$$

$Z_s = [z_1, \dots, z_n], \quad D_s = \mathrm{diag}(r_1, \dots, r_n), \quad V_s = [v_1, \dots, v_n]$

ISM factorization

Given A nonsingular and $s > 0$ (parameter)

$$z_k = e_k - \sum_{i=1}^{k-1} \frac{v_i^T e_k}{s r_i} z_i \quad \text{and} \quad v_k = y_k - \sum_{i=1}^{k-1} \frac{y_k^T z_i}{s r_i} v_i, \quad k = 1, \dots, n \quad (1)$$

$r_k = 1 + y_k^T z_k / s = 1 + v_k^T e_k / s \ (\neq 0)$,
e_k and y_k columns of I and $Y = A^T - sI$, respectively.

Inverse Sherman-Morrison factorization

$$s^{-1} I - A^{-1} = s^{-2} Z_s D_s^{-1} V_s^T$$

$Z_s = [z_1, \dots, z_n]$, $\quad D_s = \text{diag}(r_1, \dots, r_n)$, $\quad V_s = [v_1, \dots, v_n]$

Relation between ISM and LDU factorizations

ISM relations [BCMM, SISC 2003]

- Z_s is independent of s (denoted Z)
- $sD_s = tD_t$

Relation between ISM and LDU factorizations [BMMT, SISC 2008]

$$D_s = sD, \qquad Z = U^{-1}, \qquad V_s = U^T D - sL^{-T}, \tag{2}$$

Relation between ISM and LDU factorizations

ISM relations [BCMM, SISC 2003]

- Z_s is independent of s (denoted Z)
- $sD_s = tD_t$

Relation between ISM and LDU factorizations [BMMT, SISC 2008]

$$D_s = sD, \qquad Z = U^{-1}, \qquad V_s = U^T D - sL^{-T}, \tag{2}$$

Relation between ISM and LDU factorizations

ISM relations [BCMM, SISC 2003]

- Z_s is independent of s (denoted Z)
- $sD_s = tD_t$

Relation between ISM and LDU factorizations [BMMT, SISC 2008]

$$D_s = sD, \qquad Z = U^{-1}, \qquad V_s = U^T D - sL^{-T}, \tag{2}$$

Relation between ISM and LDU factorizations

ISM relations [BCMM, SISC 2003]

- Z_s is independent of s (denoted Z)
- $sD_s = tD_t$

Relation between ISM and LDU factorizations [BMMT, SISC 2008]

$$D_s = sD, \qquad Z = U^{-1}, \qquad V_s = U^T D - sL^{-T}, \tag{2}$$

V_s structure (picture)

$$V_s = \begin{bmatrix} \ddots & & -sL^{-T} \\ & \ddots & \\ U^T D & & \ddots \end{bmatrix}, \qquad \mathrm{diag}(V_s) = D - sI.$$

- V_s stores inverse and direct factors of A: U and L^{-1}.
- $U^{-1} = Z$
- One needs to compute L.

V_s structure (picture)

$$V_s = \begin{bmatrix} \ddots & & -sL^{-T} \\ & \ddots & \\ U^T D & & \ddots \end{bmatrix}, \quad \text{diag}(V_s) = D - sI.$$

- V_s stores inverse and direct factors of A: U and L^{-1}.
- $U^{-1} = Z$
- One needs to compute L.

V_s structure (picture)

$$V_s = \begin{bmatrix} \ddots & & -sL^{-T} \\ & \ddots & \\ U^T D & & \ddots \end{bmatrix}, \quad \mathrm{diag}(V_s) = D - sI.$$

- V_s stores inverse and direct factors of A: U and L^{-1}.
- $U^{-1} = Z$
- One needs to compute L.

V_s structure (picture)

$$V_s = \begin{bmatrix} \ddots & & -sL^{-T} \\ & \ddots & \\ U^T D & & \ddots \end{bmatrix}, \quad \mathrm{diag}(V_s) = D - sI.$$

- V_s stores inverse and direct factors of A: U and L^{-1}.
- $U^{-1} = Z$
- One needs to compute L.

Improvements on BIF

A new proof of

Theorem (BMMT, SIAM J. Sci. Comput., 30(5):2302–2318, 2008)

Let $A = LDU$ be the LDU decomposition of A. Then

$$U = Z^{-1}, \qquad V_s = U^T D - sL^{-T}.$$

gives a new (simpler) expression for computing some entries of V_S

Theorem

If $p < k$

$$v_{pk} = sl_{kp} - \sum_{i=p+1}^{k-1} l_{ki} v_{pi}. \tag{3}$$

Improvements on BIF

A new proof of

Theorem (BMMT, SIAM J. Sci. Comput., 30(5):2302–2318, 2008)

Let $A = LDU$ be the LDU decomposition of A. Then

$$U = Z^{-1}, \qquad V_s = U^T D - sL^{-T}.$$

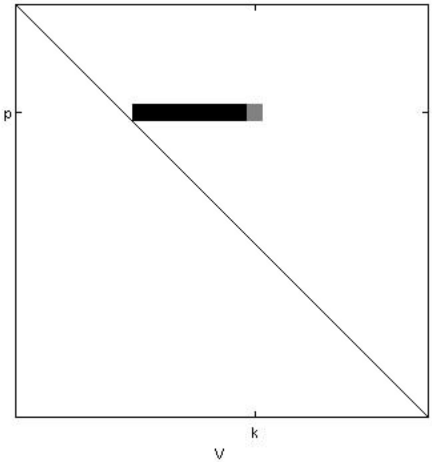
gives a new (simpler) expression for computing some entries of V_S

Theorem

If $p < k$

$$v_{pk} = sl_{kp} - \sum_{i=p+1}^{k-1} l_{ki} v_{pi}. \tag{3}$$

Entries of V_s used to compute v_{pk}.



Outline

Notation

Notation

$\tilde{Z}$, $\tilde{V}_s$ and $\tilde{D}_s = \text{diag}(\tilde{r}_1, \ldots, \tilde{r}_n)$ factors of the ISM decomposition of A^T,

$$s^{-1}I - A^{-T} = s^{-2}\tilde{Z}\tilde{D}_s^{-1}\tilde{V}_s^{T}.$$

Observation

Z (factor of A) is embedded in $\tilde{V}_s^{T}$ and $\tilde{Z}$ is embedded in V_s.

Notation

Notation

$\tilde{Z}$, $\tilde{V}_s$ and $\tilde{D}_s = \text{diag}(\tilde{r}_1, \ldots, \tilde{r}_n)$ factors of the ISM decomposition of A^T,

$$s^{-1} I - A^{-T} = s^{-2} \tilde{Z} {\tilde{D}_s}^{-1} {\tilde{V}_s}^T.$$

Observation

Z (factor of A) is embedded in ${\tilde{V}_s}^T$ and $\tilde{Z}$ is embedded in V_s.

Computing v_{pk}

Lemma

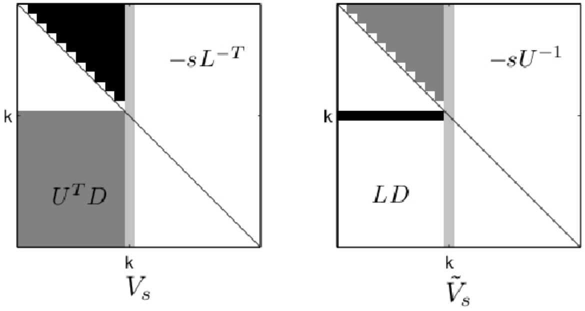
Instead as in (3) ($v_{pk} = sl_{kp} - \sum_{i=p+1}^{k-1} l_{ki} v_{pi}$), for $p < k$ compute

$$v_{pk} = s\frac{\tilde{v}_{kp}}{d_p} - \sum_{i=p+1}^{k-1} \frac{\tilde{v}_{ki}}{d_i} v_{pi}.$$

and also as

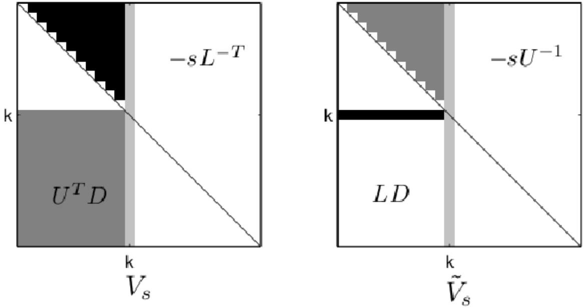
$$v_{pk} = s\frac{a^k z_p}{d_p} - \sum_{j=p+1}^{k-1} \frac{\tilde{v}_{kj}}{d_j} v_{pj}, \quad \text{for} \quad p < k, \tag{4}$$

Construction of the k-th row of V



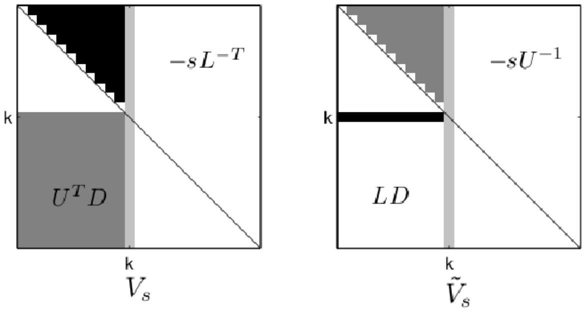
- Black: Used to compute the first $p-1$ components.
- Grey: Used to compute the last $n-p$ components.

Construction of the k-th row of V



- Black: Used to compute the first $p-1$ components.
- Grey: Used to compute the last $n-p$ components.

Construction of the k-th row of V



- Black: Used to compute the first $p - 1$ components.
- Grey: Used to compute the last $n - p$ components.

Comments

- Two interleaved processes to compute V_s and $\tilde{V}_s$ (due to dropping rules and to the changes in Lemma 3).
- Computation of Z and $\tilde{Z}$ can be avoided since

$$u_p = -[v_{1:p-1,p}^T/s, -1.0, zeros_{p+1:n}^T]^T.$$

but . . .

Comments

- Two interleaved processes to compute V_s and $\tilde{V}_s$ (due to dropping rules and to the changes in Lemma 3).
- Computation of Z and $\tilde{Z}$ can be avoided since

$$u_p = -[v^T_{1:p-1,p}/s, -1.0, zeros^T_{p+1:n}]^T.$$

but . . .

Comments

- Two interleaved processes to compute V_s and $\tilde{V}_s$ (due to dropping rules and to the changes in Lemma 3).
- Computation of Z and $\tilde{Z}$ can be avoided since

$$u_p = -[v_{1:p-1,p}^T/s, -1.0, zeros_{p+1:n}^T]^T.$$

but . . .

Comments

- Two interleaved processes to compute V_s and $\tilde{V}_s$ (due to dropping rules and to the changes in Lemma 3).
- Computation of Z and $\tilde{Z}$ can be avoided since

$$u_p = -[v_{1:p-1,p}^T / s, -1.0, zeros_{p+1:n}^T]^T.$$

but . . .

Comments

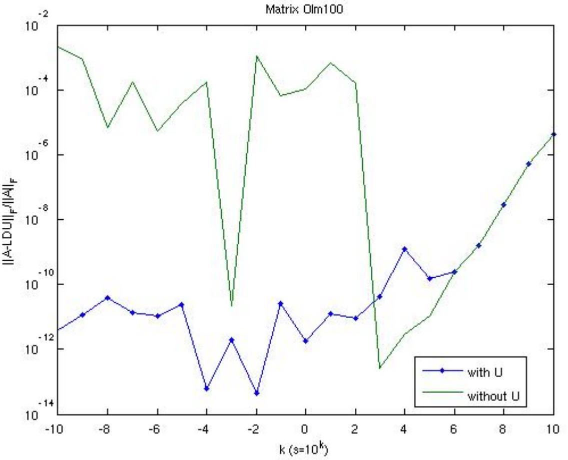
- Two interleaved processes to compute V_s and $\tilde{V}_s$ (due to dropping rules and to the changes in Lemma 3).
- Computation of Z and $\tilde{Z}$ can be avoided since

$$u_p = -[v_{1:p-1,p}^T/s, -1.0, zeros_{p+1:n}^T]^T.$$

but . . .

Behavior with different values of s

Error of the LDU factorization of the matrix OLM100 in function of s.



Outline

Setup, parameters, . . .

- $s = 1$
- Preprocess: Reordering and scaling: MC64 of HSL library (maximum product transversal with rows and column scalings).
- Method: BiCGStab
- Stopping criteria: a relative decrease 10^{-8} of the system backward error, (except for: STOMACH, 10^{-3}, and TORSO3, $2 \cdot 10^{-3}$)
- Maximum number of iterations: 2,000
- Right-hand side: Artificial, computed as $b = Ae$.
- Initial guess: the vector of all zeros.
- Storage sparse by columns $+10n$ by rows.
- Z and $\tilde{Z}$ are computed and stored.

Setup, parameters, . . .

- $s = 1$
- Preprocess: Reordering and scaling: MC64 of HSL library (maximum product transversal with rows and column scalings).
- Method: BiCGStab
- Stopping criteria: a relative decrease 10^{-8} of the system backward error, (except for: STOMACH, 10^{-3}, and TORSO3, $2 \cdot 10^{-3}$)
- Maximum number of iterations: 2,000
- Right-hand side: Artificial, computed as $b = Ae$.
- Initial guess: the vector of all zeros.
- Storage sparse by columns $+10n$ by rows.
- Z and $\tilde{Z}$ are computed and stored.

Setup, parameters, . . .

- $s = 1$
- Preprocess: Reordering and scaling: MC64 of HSL library (maximum product transversal with rows and column scalings).
- Method: BiCGStab
- Stopping criteria: a relative decrease 10^{-8} of the system backward error, (except for: STOMACH, 10^{-3}, and TORSO3, $2 \cdot 10^{-3}$)
- Maximum number of iterations: 2,000
- Right-hand side: Artificial, computed as $b = Ae$.
- Initial guess: the vector of all zeros.
- Storage sparse by columns $+10n$ by rows.
- Z and $\tilde{Z}$ are computed and stored.

Setup, parameters, . . .

- $s = 1$
- Preprocess: Reordering and scaling: MC64 of HSL library (maximum product transversal with rows and column scalings).
- Method: BiCGStab
- Stopping criteria: a relative decrease 10^{-8} of the system backward error, (except for: STOMACH, 10^{-3}, and TORSO3, $2 \cdot 10^{-3}$)
- Maximum number of iterations: 2,000
- Right-hand side: Artificial, computed as $b = Ae$.
- Initial guess: the vector of all zeros.
- Storage sparse by columns $+10n$ by rows.
- Z and $\tilde{Z}$ are computed and stored.

Setup, parameters, . . .

- $s = 1$
- Preprocess: Reordering and scaling: MC64 of HSL library (maximum product transversal with rows and column scalings).
- Method: BiCGStab
- Stopping criteria: a relative decrease 10^{-8} of the system backward error, (except for: STOMACH, 10^{-3}, and TORSO3, $2 \cdot 10^{-3}$)
- Maximum number of iterations: 2,000
- Right-hand side: Artificial, computed as $b = Ae$.
- Initial guess: the vector of all zeros.
- Storage sparse by columns $+10n$ by rows.
- Z and $\tilde{Z}$ are computed and stored.

Setup, parameters, ...

- $s = 1$
- Preprocess: Reordering and scaling: MC64 of HSL library (maximum product transversal with rows and column scalings).
- Method: BiCGStab
- Stopping criteria: a relative decrease 10^{-8} of the system backward error, (except for: STOMACH, 10^{-3}, and TORSO3, $2 \cdot 10^{-3}$)
- Maximum number of iterations: 2,000
- Right-hand side: Artificial, computed as $b = Ae$.
- Initial guess: the vector of all zeros.
- Storage sparse by columns $+10n$ by rows.
- Z and $\tilde{Z}$ are computed and stored.

Setup, parameters, . . .

- $s = 1$
- Preprocess: Reordering and scaling: MC64 of HSL library (maximum product transversal with rows and column scalings).
- Method: BiCGStab
- Stopping criteria: a relative decrease 10^{-8} of the system backward error, (except for: STOMACH, 10^{-3}, and TORSO3, $2 \cdot 10^{-3}$)
- Maximum number of iterations: 2,000
- Right-hand side: Artificial, computed as $b = Ae$.
- Initial guess: the vector of all zeros.
- Storage sparse by columns $+10n$ by rows.
- Z and $\tilde{Z}$ are computed and stored.

Setup, parameters, . . .

- $s = 1$
- Preprocess: Reordering and scaling: MC64 of HSL library (maximum product transversal with rows and column scalings).
- Method: BiCGStab
- Stopping criteria: a relative decrease 10^{-8} of the system backward error, (except for: STOMACH, 10^{-3}, and TORSO3, $2 \cdot 10^{-3}$)
- Maximum number of iterations: 2,000
- Right-hand side: Artificial, computed as $b = Ae$.
- Initial guess: the vector of all zeros.
- Storage sparse by columns $+10n$ by rows.
- Z and $\tilde{Z}$ are computed and stored.

Setup, parameters, ...

- $s = 1$
- Preprocess: Reordering and scaling: MC64 of HSL library (maximum product transversal with rows and column scalings).
- Method: BiCGStab
- Stopping criteria: a relative decrease 10^{-8} of the system backward error, (except for: STOMACH, 10^{-3}, and TORSO3, $2 \cdot 10^{-3}$)
- Maximum number of iterations: 2,000
- Right-hand side: Artificial, computed as $b = Ae$.
- Initial guess: the vector of all zeros.
- Storage sparse by columns $+10n$ by rows.
- Z and $\tilde{Z}$ are computed and stored.

Setup, parameters, . . .

- $s = 1$
- Preprocess: Reordering and scaling: MC64 of HSL library (maximum product transversal with rows and column scalings).
- Method: BiCGStab
- Stopping criteria: a relative decrease 10^{-8} of the system backward error, (except for: STOMACH, 10^{-3}, and TORSO3, $2 \cdot 10^{-3}$)
- Maximum number of iterations: 2,000
- Right-hand side: Artificial, computed as $b = Ae$.
- Initial guess: the vector of all zeros.
- Storage sparse by columns $+10n$ by rows.
- Z and $\tilde{Z}$ are computed and stored.

Test problems (Univ. Florida Sparse Matrix Collection)

Matrix	n	nz	Application
CHEM_MASTER1	40,401	201,201	chemical engineering 2D/3D problem
EPB3	84,617	463,625	thermal problem
POISSON3DB	85,623	2,374,949	computational fluid dynamics
RAJAT20	86,916	604,299	circuit simulation problem
HCIRCUIT	105,676	513,072	circuit simulation problem
TRANS4	116,835	749,800	circuit simulation
CAGE12	130,228	2,032,536	directed weighted graph
FEM_3D_THERMAL2	147,900	3,489,300	thermal problem
XENON2	157,464	3,866,668	materials problem
CRASHBASIS	160,000	1,750,416	optimization problem
MAJORBASIS	160,000	1,750,416	optimization problem
STOMACH	213,360	3,021,648	2D/3D problem
TORSO3	256,156	4,429,042	2D/3D problem
ASIC_320KS	321,671	1,316,085	circuit simulation problem
LANGUAGE	399,130	1,216,334	directed weighted graph
CAGE13	445,315	7,479,343	directed weighted graph
RAJAT30	643,994	6,175,244	circuit simulation problem
ASIC_680K	682,862	2,638,997	circuit simulation problem
CAGE14	1,505,785	27,130,439	directed weighted graph

Data reported

- *relsize* $= \frac{\mathrm{nnz}(L)+\mathrm{nnz}(U)}{\mathrm{nnz}(A)}$
- t_p time for constructing the preconditioner.
- t_it time for the iterative solution phase.

Preconditioner parameters

Parameters selected to . . .

- . . . obtain similar sizes.
- . . . sparse preconditioners.

Data reported

- *relsize* $= \frac{\mathrm{nnz}(L)+\mathrm{nnz}(U)}{\mathrm{nnz}(A)}$
- t_p time for constructing the preconditioner.
- t_it time for the iterative solution phase.

Preconditioner parameters

Parameters selected to ...

- ... obtain similar sizes.
- ... sparse preconditioners.

Data reported

- *relsize* $= \frac{\mathrm{nnz}(L)+\mathrm{nnz}(U)}{\mathrm{nnz}(A)}$
- *t_p* time for constructing the preconditioner.
- *t_it* time for the iterative solution phase.

Preconditioner parameters

Parameters selected to ...

- ... obtain similar sizes.
- ... sparse preconditioners.

Data reported

- $relsize = \frac{\mathrm{nnz}(L)+\mathrm{nnz}(U)}{\mathrm{nnz}(A)}$
- t_p time for constructing the preconditioner.
- t_it time for the iterative solution phase.

Preconditioner parameters

Parameters selected to . . .

- . . . obtain similar sizes.
- . . . sparse preconditioners.

Data reported

- *relsize* $= \frac{\mathrm{nnz}(L)+\mathrm{nnz}(U)}{\mathrm{nnz}(A)}$
- *t_p* time for constructing the preconditioner.
- *t_it* time for the iterative solution phase.

Preconditioner parameters

Parameters selected to . . .

- . . . obtain similar sizes.
- . . . sparse preconditioners.

Data reported

- *relsize* $= \frac{\mathrm{nnz}(L)+\mathrm{nnz}(U)}{\mathrm{nnz}(A)}$
- *t _ p* time for constructing the preconditioner.
- *t _ it* time for the iterative solution phase.

Preconditioner parameters

Parameters selected to ...

- ... obtain similar sizes.
- ... sparse preconditioners.

Data reported

- $relsize = \frac{\mathrm{nnz}(L)+\mathrm{nnz}(U)}{\mathrm{nnz}(A)}$
- t_p time for constructing the preconditioner.
- t_it time for the iterative solution phase.

Preconditioner parameters

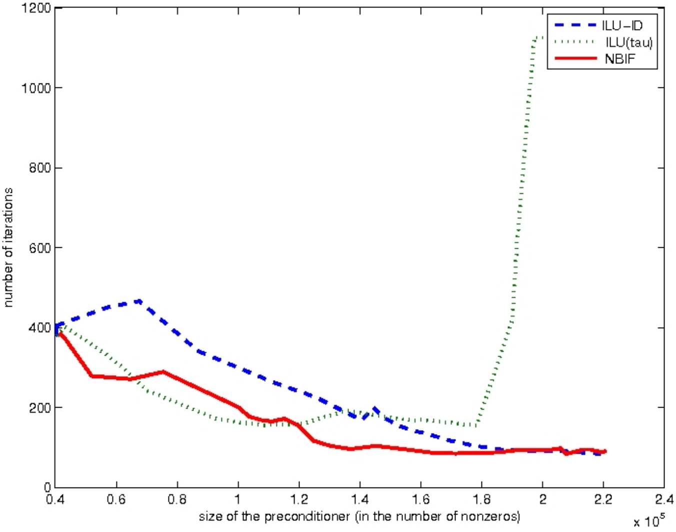
Parameters selected to . . .

- . . . obtain similar sizes.
- . . . sparse preconditioners.

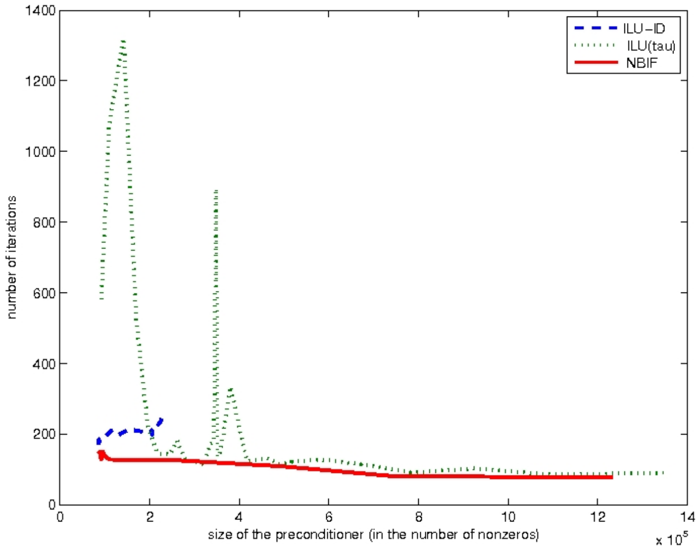
Comparing NBIF ILU-ID

Matrix	NBIF				ILU-ID			
	rlsize	*t_p*	*its*	*t_it*	*rlsize*	*t_p*	*its*	*t_it*
CHEM_MASTER1	0.53	0.22	169	0.73	0.70	0.05	168	0.83
EPB3	0.99	0.72	83	1.20	1.06	0.08	120	1.50
POISSON3DB	0.11	0.69	126	3.48	0.09	0.39	199	5.20
RAJAT20	0.17	0.14	8	0.09	0.15	0.13	9	0.09
HCIRCUIT	0.40	0.14	182	2.56	0.21	0.11	203	2.31
TRANS4	0.45	0.23	3	0.06	0.48	1.13	5	0.09
CAGE12	0.31	0.55	5	0.14	0.35	0.38	9	0.22
FEM_3D_THERMAL2	0.06	0.52	20	0.61	0.06	0.43	26	0.80
XENON2	0.05	0.60	539	19.5	0.05	0.44	690	27.1
CRASHBASIS	0.18	0.39	29	0.71	0.20	0.28	14	0.34
MAJORBASIS	0.36	0.73	15	0.42	0.33	0.27	15	0.41
STOMACH	0.07	0.53	20	0.66	0.12	0.42	21	0.67
TORSO3	0.06	0.78	6	0.28	0.07	0.55	6	0.30
ASIC_320KS	0.26	0.47	20	0.94	0.18	0.28	20	0.88
LANGUAGE	0.53	0.70	9	0.55	0.33	0.36	20	1.06
CAGE13	0.06	1.41	6	0.55	0.06	1.05	7	0.61
RAJAT30	0.11	1.56	3	0.34	0.11	1.17	3	0.34
ASIC_680KS	0.36	0.97	5	0.22	0.43	0.59	5	0.55
CAGE14	0.07	12.2	6	2.02	0.07	4.30	8	2.81

CHEM_MASTER1

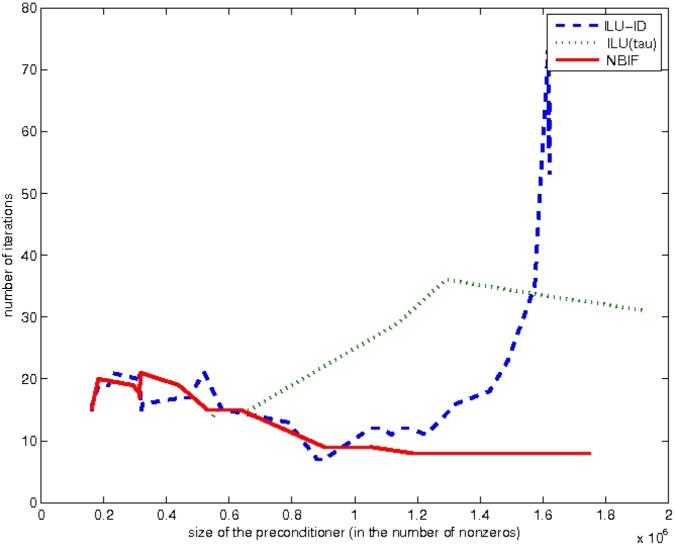


POISSON3DB



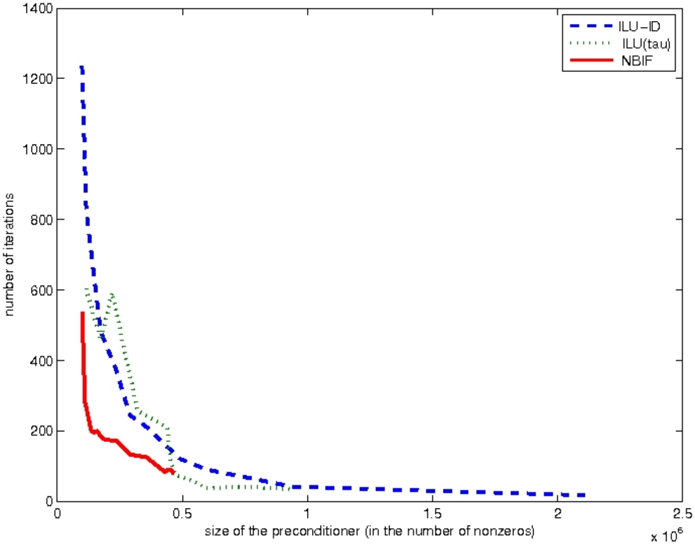
1400
1200
1000
800
600
400
200
0
number of iterations
0
2
4
6
8
10
12
14
size of the preconditioner (in the number of nonzeros)
x 10^5
ILU−ID
ILU(tau)
NBIF

MAJORBASIS



80
70
60
50
40
30
20
10
0
number of iterations
0
0.2
0.4
0.6
0.8
1
1.2
1.4
1.6
1.8
2
size of the preconditioner (in the number of nonzeros)
x 10^6
ILU−ID
ILU(tau)
NBIF

EPB3


1400
1200
1000
800
600
400
200
0
0
0.5
1
1.5
2
2.5
x 10^6
number of iterations
size of the preconditioner (in the number of nonzeros)
ILU−ID
ILU(tau)
NBIF

Outline

Conclusions and future work

Conclusions

- New insight into ISM factorization (dependence of direct and inverse factors).
- Improvements on BIF algorithm.
- A version for the nonsymmetric case: NBIF.
- A robust preconditioner

Work in progress

Extensions to

- Least squares problems.
- Block version.
- Implementation details.
- Floating-point analysis.

Conclusions and future work

Conclusions

- New insight into ISM factorization (dependence of direct and inverse factors).
- Improvements on BIF algorithm.
- A version for the nonsymmetric case: NBIF.
- A robust preconditioner

Work in progress

Extensions to

- Least squares problems.
- Block version.
- Implementation details.
- Floating-point analysis.

Conclusions and future work

Conclusions

- New insight into ISM factorization (dependence of direct and inverse factors).
- Improvements on BIF algorithm.
- A version for the nonsymmetric case: NBIF.
- A robust preconditioner

Work in progress

Extensions to

- Least squares problems.
- Block version.
- Implementation details.
- Floating-point analysis.

Conclusions and future work

Conclusions

- New insight into ISM factorization (dependence of direct and inverse factors).
- Improvements on BIF algorithm.
- A version for the nonsymmetric case: NBIF.
- A robust preconditioner

Work in progress

Extensions to

- Least squares problems.
- Block version.
- Implementation details.
- Floating-point analysis.

Conclusions and future work

Conclusions

- New insight into ISM factorization (dependence of direct and inverse factors).
- Improvements on BIF algorithm.
- A version for the nonsymmetric case: NBIF.
- A robust preconditioner

Work in progress

Extensions to

- Least squares problems.
- Block version.
- Implementation details.
- Floating-point analysis.

Conclusions and future work

Conclusions

- New insight into ISM factorization (dependence of direct and inverse factors).
- Improvements on BIF algorithm.
- A version for the nonsymmetric case: NBIF.
- A robust preconditioner

Work in progress

Extensions to

- Least squares problems.
- Block version.
- Implementation details.
- Floating-point analysis.

Conclusions and future work

Conclusions

- New insight into ISM factorization (dependence of direct and inverse factors).
- Improvements on BIF algorithm.
- A version for the nonsymmetric case: NBIF.
- A robust preconditioner

Work in progress

Extensions to

- Least squares problems.
- Block version.
- Implementation details.
- Floating-point analysis.

Conclusions and future work

Conclusions

- New insight into ISM factorization (dependence of direct and inverse factors).
- Improvements on BIF algorithm.
- A version for the nonsymmetric case: NBIF.
- A robust preconditioner

Work in progress

Extensions to

- Least squares problems.
- Block version.
- Implementation details.
- Floating-point analysis.

Conclusions and future work

Conclusions

- New insight into ISM factorization (dependence of direct and inverse factors).
- Improvements on BIF algorithm.
- A version for the nonsymmetric case: NBIF.
- A robust preconditioner

Work in progress

Extensions to

- Least squares problems.
- Block version.
- Implementation details.
- Floating-point analysis.

Conclusions and future work

Conclusions

- New insight into ISM factorization (dependence of direct and inverse factors).
- Improvements on BIF algorithm.
- A version for the nonsymmetric case: NBIF.
- A robust preconditioner

Work in progress

Extensions to

- Least squares problems.
- Block version.
- Implementation details.
- Floating-point analysis.

Thanks a lot