Preconditioning for linear least-squares problems

Miroslav Tůma

Institute of Computer Science

Academy of Sciences of the Czech Republic

tuma@cs.cas.cz

joint work with Rafael Bru, José Marín and José Mas

Sparse Days, Toulouse,

France, June 25-26, 2012

Outline

Introduction: the Problem

$$\min_x \|b - Ax\|_2,\ A \in R^{m,n},\ m \geq n$$

Introduction: the Problem

$$\min_x \|b - Ax\|_2,\ A \in R^{m,n},\ m \geq n$$

- Large and sparse (overdetermined) linear least squares
- Method of choice: Preconditioned CGLS

Iterative solution techniques often weak and far from being general

- Just basic algorithms, no block or hierarchical framework

Outline

Solving the Least Squares by Preconditioned Iterative Method

Three main classes of general purpose preconditioning

- Preconditioning the normal equations by SPD decompositions
  - Based on the system $A^T Ax = A^T b$
- LU-based strategies
  - A approximately decomposed as LU where U is upper triangular, L is lower trapezoidal
  - LU assumed approximate since there are better ways to solve the problem with direct methods
- Incomplete QR preconditioning of A
  - A approximately decomposed as QR where R is upper triangular, Q is approximately orthogonal

Outline

Preconditioning the normal equations by SPD decompositions: I.

$$A^T A x = A^T b$$

- Some information can be lost by forming $A^T A$ unless sufficient precision is used
- See, e.g., Benoit (1924); nice overview in Björck (1996), a lot of beautiful application papers. See also Bellavia, Gondzio, Morini, 2012.
- $A^T A$ most often decomposed by the incomplete Cholesky algorithm
- Often hoped to solve problems by breakdown-free Cholesky decompositions (Jennings-Ajiz, 1984; Tismenetsky, 1991; Kaporin, 1998; etc.)
- Standard approach with well-known pros and cons

Preconditioning the normal equations by SPD decompositions: II.

Some specific features

- Little theory characterizing problems where one of these methods can be better than the others
- Iterative part implemented similarly in all CGLS schemes: without explicit multiplication $B = A^T A$.
- System matrix $A^T A$ can be formed explicitly or implicitly just during decomposition of $A^T A$: mentioned by Björck, 1996; used in RIF (Benzi, T., 2003).
- Our way to tune this approach : using the balanced incomplete decomposition BIF (Bru et al., 2008)

Preconditioning the normal equations by SPD decompositions: III.

Balanced incomplete decomposition: I.

- Standard SPD B biconjugate decomposition (e.g., Chu, Funderlic, Golub, 1995): For an SPD matrix $B = A^T A \in R^{n,n}$ find $Z, D \in R^{n,n}$ such that D is diagonal and

$$Z^T B Z = D$$

- Z can be computed as upper triangular - biconjugation can be considered as the decomposition of the matrix inverse.
- AINV, SAINV are algorithms to get the biconjugate factors Z, D taking into account sparsity and incompleteness (Benzi, Meyer, T., 1996; Benzi, Cullum, T., 2000), dense decompositions dating much more back.
- Numerical properties of the involved orthogonalization schemes: Kopal, Rozložník, Smoktunowicz, T., 2012.

Preconditioning the normal equations by SPD decompositions: IV.

Balanced incomplete decomposition: II.

- $$Z^T BZ = D \leftrightarrow Z^T LDL^T Z = D \leftrightarrow I = ZL^T$$

  Use of L^T and D computed with the use of Z: RIF (Benzi, T, 2003)
- See also Cui, 2009; Cui, Hayami, Yin, 2011 for experiments with RIF derived via the Greville's method (Greville, 1960).
- $$Z^T(I - B^{-1})^{-1}Z = D \leftrightarrow I = Z(Z^T + V^T),\ V^T = Z^T + L^T$$

  Use of L^T and D computed at the same time with Z: BIF (Bru et al., 2008)
- Both direct and inverse factors are computed.
- Sparse computation of both factors is reasonably cheap.

Preconditioning the normal equations by SPD decompositions: VI.

BIF versus Tismenetsky for an SPD matrix PWTK, n=217,918, nnz=5,926,171

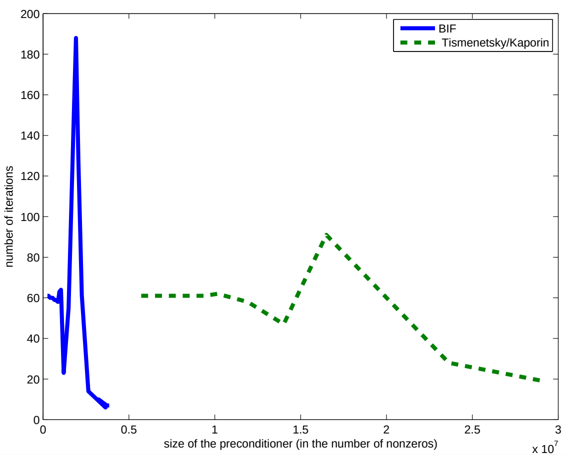


Figure: Iteration counts for CG preconditioned by BIF and Tismenetsky/Kaporin IC versus preconditioner size for the matrix PWTK.

Preconditioning the normal equations by SPD decompositions: VII.

BIF versus Tismenetsky for an SPD matrix PWTK, n=217,918, nnz=5,926,171: II.

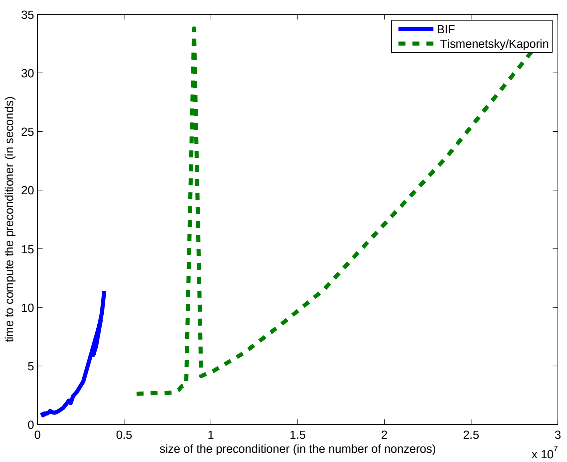


Figure: Preconditioner construction time for CG preconditioned by BIF and Tismenetsky/Kaporin IC versus preconditioner size for the matrix PWTK.

Preconditioning the normal equations by SPD decompositions: VIII.

BIF versus Tismenetsky for an SPD matrix PWTK, n=217,918, nnz=5,926,171: III.

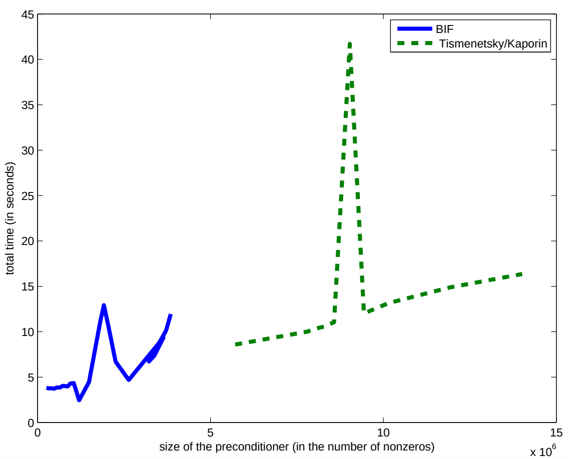


Figure: Preconditioner construction time for CG preconditioned by BIF and Tismenetsky/Kaporin IC versus preconditioner size for the matrix PWTK.

Preconditioning the normal equations by SPD decompositions: IX.

- Does the power of BIF transfer into the preconditioning of normal equations?
- At least sometimes?

Preconditioning the normal equations by SPD decompositions: X.

BIF versus IC, NE, S from the animal breeding package, n=1959, m=3140

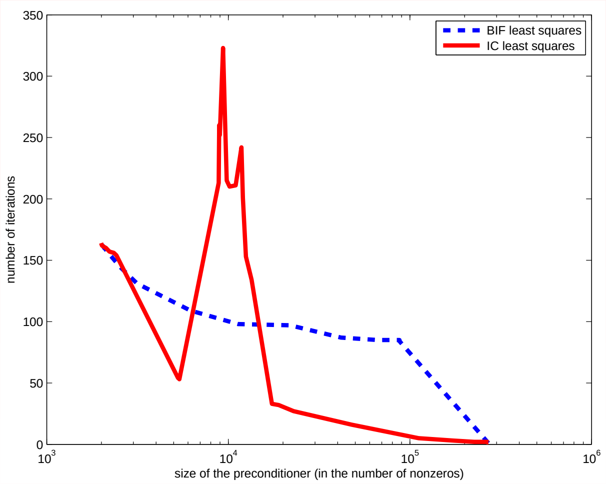


Figure: BIF versus IC, normal equations, S matrix

Preconditioning the normal equations by SPD decompositions: XI.

BIF versus IC, NE, M from the animal breeding package, n=6019, m=9397

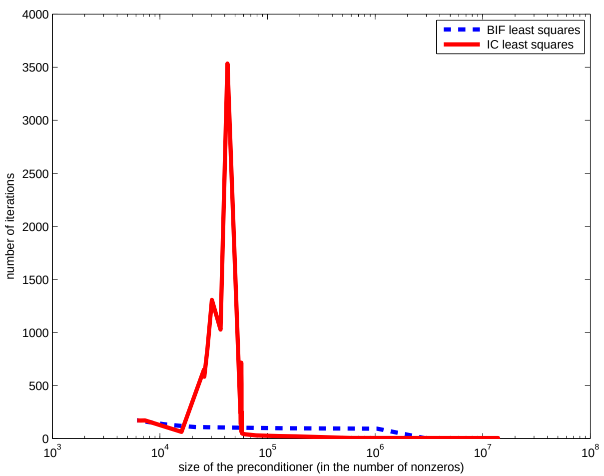


Figure: BIF versus IC, normal equations, M matrix

Preconditioning the normal equations by SPD decompositions: XII.

BIF versus IC, NE, L from the animal breeding package, n=28254, m=17150

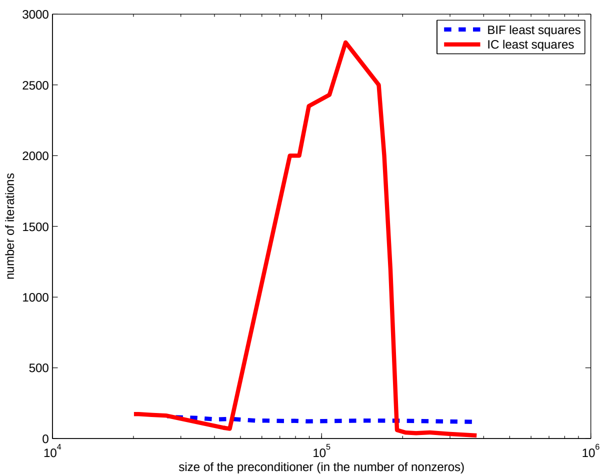


Figure: BIF versus IC, normal equations, L matrix

Preconditioning the normal equations by SPD decompositions: XIII.

BIF versus IC, NE, WELL1033, n=28254, m=17150

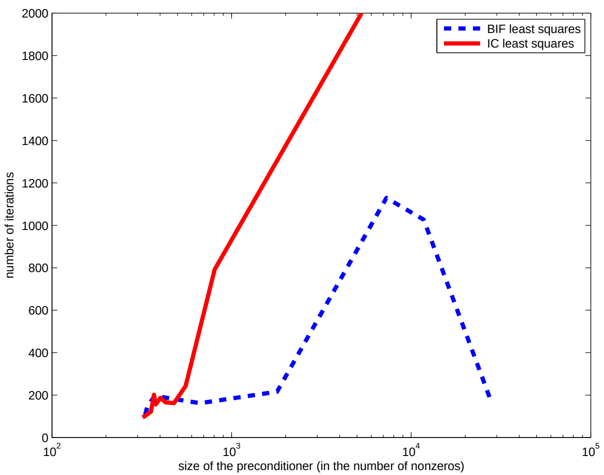


Figure: BIF versus IC, normal equations, WELL1033

Preconditioning the normal equations by SPD decompositions: XIV.

BIF versus IC, NE, ILLC1033, n=28254, m=17150

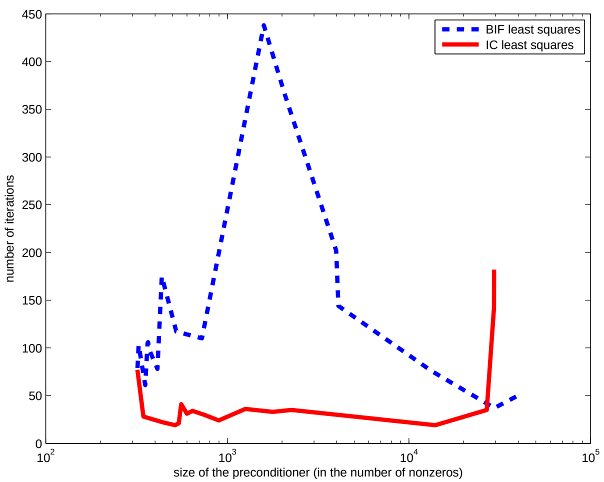


Figure: BIF versus IC, normal equations, ILLC1033

Outline

Preconditioning by LU factors of A: I.

$$PA = \begin{pmatrix} A_1 \\ A_2 \end{pmatrix} = \begin{pmatrix} L_1 \\ L_2 \end{pmatrix} U,$$

- $L_1 U$ is used as a preconditioner for the normal equations and CGLS.
- That is, we apply A_1^{-1} and we do not solve least-squares in for L (Peters, Wilkinson) if we would have well-conditioned L
- Column permutation we get implicitly by numerical pivoting
- Less information exploited for preconditioning, hopefully less information excluded.
- Introduced by Läuchli, 1961, see also Freund, 1987, Björck (1996), Björck, Yuan (1999)
- In practice, rather underestimated approach.

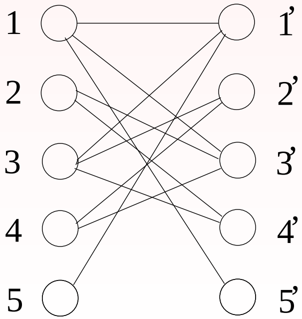
Preconditioning by LU factors of A: Ia.

- Finding P is the critical and hard task, see also the talk of Iain Duff in Valencia.
- Here just one approach from those that we tried:
  - modifications of P4 (Hellermann, Rarick, 1972)
  - various crash LP procedures
  - matchings
- Matchings, but note that in matrices with ones, minus ones and a few of additional values there is often not much to be optimized. Numerical regularity (at least) is very important.

Preconditioning by LU factors of A: II.

- A_1 should be regular, $A_2 A_1^{-1}$ should be reasonably well-conditioned.
- Here just one approach: graph-based preprocessing
- Transversal: set of nonzero matrix entries no two of which are in the same row or column
- An example to remind this concept for square matrices:

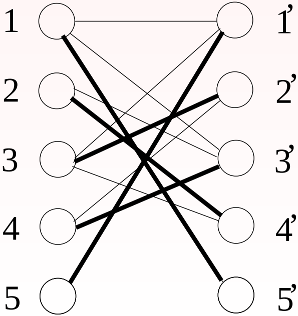
$$\begin{pmatrix} * & & * & & * \\ & & * & * & \\ * & * & & * & \\ & * & * & & \\ * & & & & \end{pmatrix}$$



Preconditioning by LU factors of A: II.

- A_1 should be regular, $A_2 A_1^{-1}$ should be reasonably well-conditioned.
- Our approach: graph-based preprocessing
- Transversal: set of nonzero matrix entries no two of which are in the same row or column
- An example to remind this concept for square matrices:

$$\begin{pmatrix} * & & * & & \spadesuit \\ & & * & \spadesuit & \\ * & \spadesuit & & * & \\ & * & \spadesuit & & \\ \spadesuit & & & & \end{pmatrix}$$



Preconditioning by LU factors of A: III.

Weighted transversals

- Weighted transversals: maximizing product of absolute value diagonal entries

$$\prod_{j=1}^{n} |a_{p(j),j}| \tag{1}$$

- Equivalent to minimizing the sum

$$\sum_{j=1}^{n} |c_{p(j),j}|, \tag{2}$$

for

$$c_{ij} = \begin{cases} \log \bar{a}_j - \log |a_{ij}|, & a_{ij} \neq 0, \\ 0, & a_{ij} = 0, \end{cases}, \tag{3}$$

Preconditioning by LU factors of A: IV.

Transversal in the rectangular case

- But we do not consider this transversal as optimal one:

$$\begin{pmatrix} 1 & & \\ & 2 & \\ & & 3 \\ \color{red}{2} & 3 & 4 \\ 3 & \color{red}{5} & 6 \\ 5 & 6 & \color{red}{8} \end{pmatrix}. \tag{4}$$

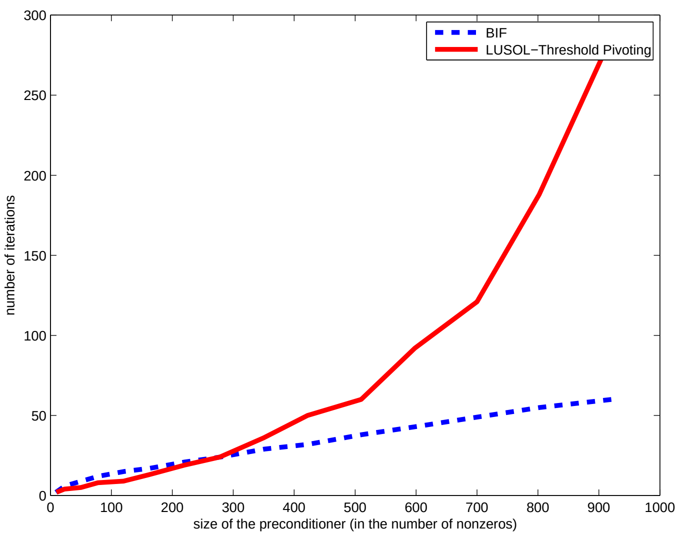
- Enough to find large entries in rows and avoid dense submatrices.
- weights by rows (loosing the minimization property) and additional degrees of freedom (sparsity).

$$c_{ij} = \begin{cases} \log \bar{a}_i - \log |a_{ij}|, & a_{ij} \neq 0, \\ 0, & a_{ij} = 0, \end{cases}, \tag{5}$$

- Incomplete LU done by nonsymmetric balanced incomplete factorization BIF (Bru et al, 2010)

Preconditioning by LU factors of A: V.

LU preconditioning by BIF versus LUSOL with threshold pivoting (version 7.0, 2008), artificial matrix derived from 2D Laplacian, $n = m/2$



Preconditioning by LU factors of A: VI.

LU preconditioning by BIF versus LUSOL, threshold pivoting, M from breeding package

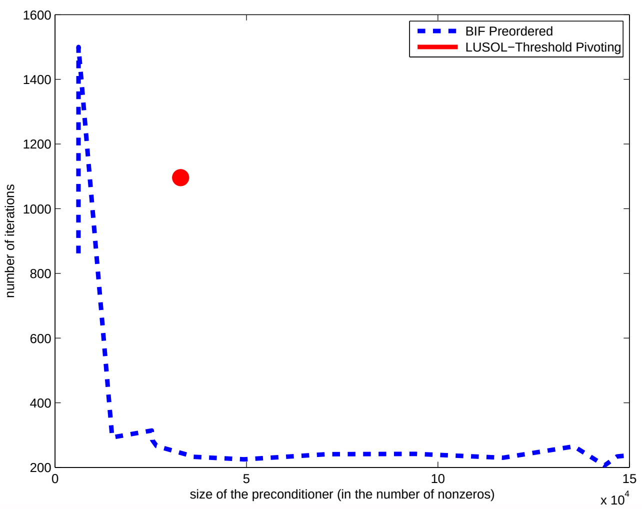


Figure: LU BIF versus LUSOL for the M matrix, animal breeding package.

Preconditioning by LU factors of A: VII.

LU preconditioning by BIF versus LUSOL, threshold pivoting, L from breeding package

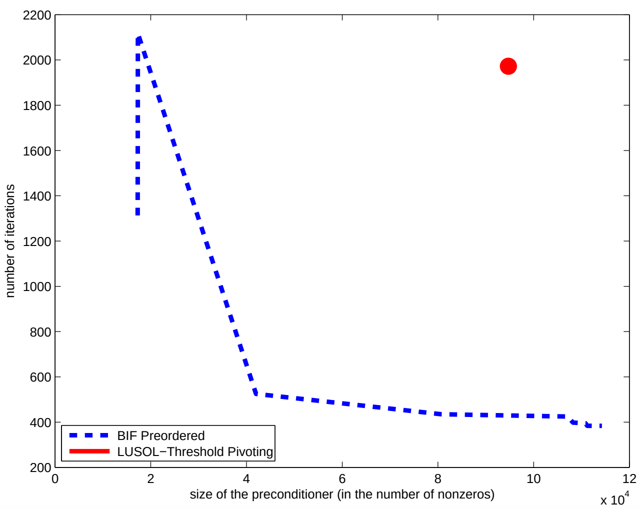


Figure: LU BIF versus LUSOL for the L matrix, animal breeding package.

Outline

Incomplete QR preconditioning A: I.

- Again, can be applied both for CGLS or a nonsymmetric iterative method. Here we stick to CGLS.
- Long history of their development: Ajiz, Jennings, 1984, Saad, 1988, Bai, Duff, Wathen, 2001, Benzi, Kouhia, T., 2001, Papadopoulos, Duff, Wathen, 2005, Bai, Duff, Yin, 2009 etc.
- Despite a lot of attention devoted to them, incomplete QR decompositions much less understood and typically very fragile. Far from having as strong procedures as in the case of incomplete Cholesky or even incomplete LU.

Incomplete QR preconditioning A: II.

- Here adding just an experiment with no dropping in Q (all dropping in R) introduced by Ajiz, Jennings, 1984, popularized as CIMGS by Wang, 1993 and Wang, Gallivan, Bramley, 1997 which is equivalent to the incomplete LU/Cholesky described by Tismenetsky, 1991.
- Tismenetsky decomposition seems to be very robust, but it is not cheap. New results on fast implementation of this approach for HSL under development: Scott, T., 2012.

Incomplete QR preconditioning A: II.

Incomplete QR as Tismenetsky decomposition versus SPD BIF using CGLS, Hirlam matrix from meteorological observations, m=1385270, n=452200.

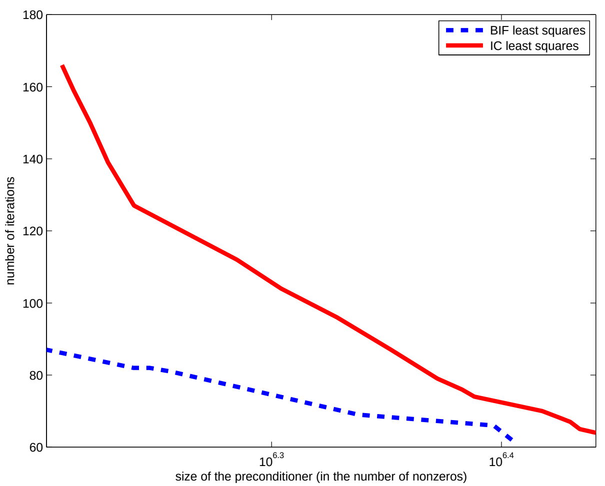


Figure: Comparison of the BIF preconditioning of the normal equations with the Tismenetsky preconditioner for the matrix Hirlam.

Outline

Conclusions

- Three approaches to precondition CGLS considered

Conclusions

- Three approaches to precondition CGLS considered
- Reasonable robustness of BIF applied to normal equations demonstrated.

Conclusions

- Three approaches to precondition CGLS considered
- Reasonable robustness of BIF applied to normal equations demonstrated.
- LU-based decomposition of rectangular systems completed by new combinatorial preprocessing

Conclusions

- Three approaches to precondition CGLS considered
- Reasonable robustness of BIF applied to normal equations demonstrated.
- LU-based decomposition of rectangular systems completed by new combinatorial preprocessing
- A lot of things to be done. Preconditioned CGLS far from being really powerful.

Last but not least

Thank you for your attention!

Last but not least

Thank you for your attention!

Last but not least

Thank you for your attention!

Last but not least

Thank you for your attention!