On Preconditioning Symmetric Indefinite Linear Systems

Miroslav Tůma

Institute of Computer Science

Academy of Sciences of the Czech Republic

and

Technical University in Liberec

5th WORKSHOP of the ERCIM WORKING GROUP: MATRIX COMPUTATIONS AND STATISTICS, PRAGUE,
August 27-29, 2004

Based on joint work with

Michele Benzi, Emory University, Atlanta

Outline

1. Basic problem
2. Indefinite solvers - an overview
3. Preprocessing techniques
4. Preconditioning strategies
5. Numerical experiments
6. Conclusions

1. Basic problem

$$Ax = b$$

- A is $n \times n$, symmetric and indefinite
- A is large and sparse

Arises in many applications, often in specialized forms

- saddle-point problems (CFD, mixed FEM, optimization, optimal control, ...)
- "shift-and-invert", Jacobi-Davidson algorithms

2. Indefinite solvers - an overview

Saddle-point problems: specialized solvers

$$\begin{pmatrix} A & B \\ B^T & -C \end{pmatrix}$$

- Reduction to a definite system
  - Schur complement approach
  - dual variable (null-space) approach

- Solving original indefinite system
  - direct solvers
  - preconditioned iterative solvers
    (block DIAG, block TR, constraint pre-
    conditioners,
    inner block reductions)

- Split and solve approaches (HSS iterations, HSS preconditioners)

Our focus: general indefinite systems

- Sparse direct methods (MA27, MA47, MA57; Duff et al.)
  - very powerful; inherent limits of direct methods
- Preconditioned iterative methods
  - Block SSOR and symmetric ILUT preconditioners (Freund, 1997)
  - Diagonal pivoting and inverse diagonal pivoting preconditioners; symmetric Krylov methods (Benzi, T., 2002); often useful for weakly indefinite systems
  - Approximate diagonal pivoting decompositions (right-looking, based on linked-lists) for smoothing (Qu, Fish, 2001)
  - Diagonal pivoting preconditioners with diagonal

and Bunch-Kaufmann pivoting (left-looking); nonsymmetric Krylov methods (Li, Saad, 2004).

– the problem is difficult: implementations/algorithms typically very fragile

3. Preprocessing techniques

Powerful matrix preorderings: nonsymmetric case

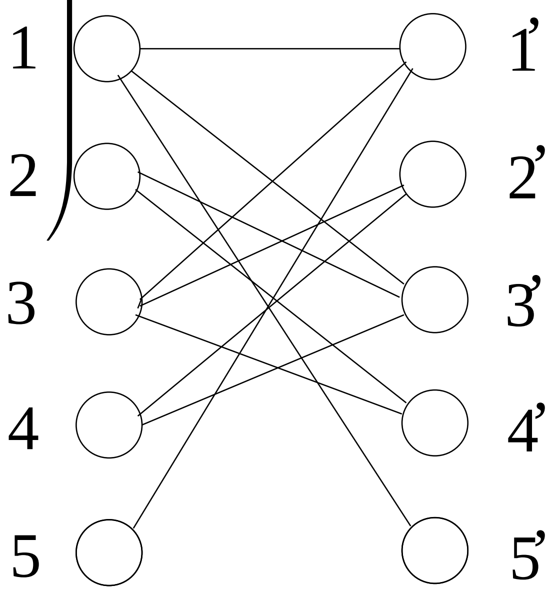
- permutation to get a nonzero diagonal – a classical technique for nonsymmetric matrices (Duff, 1977)

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

Powerful matrix preorderings: nonsymmetric case

- permutation to get a nonzero diagonal – a classical technique for nonsymmetric matrices (Duff, 1977)

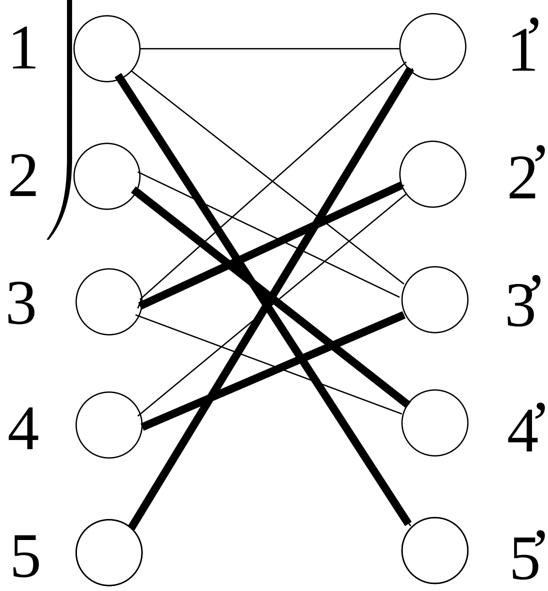
$$\begin{pmatrix} * & & * & & * \\ & & * & * & \\ * & * & & * & \\ & * & * & & \\ * & & & & \end{pmatrix}$$



Powerful matrix preorderings: nonsymmetric case

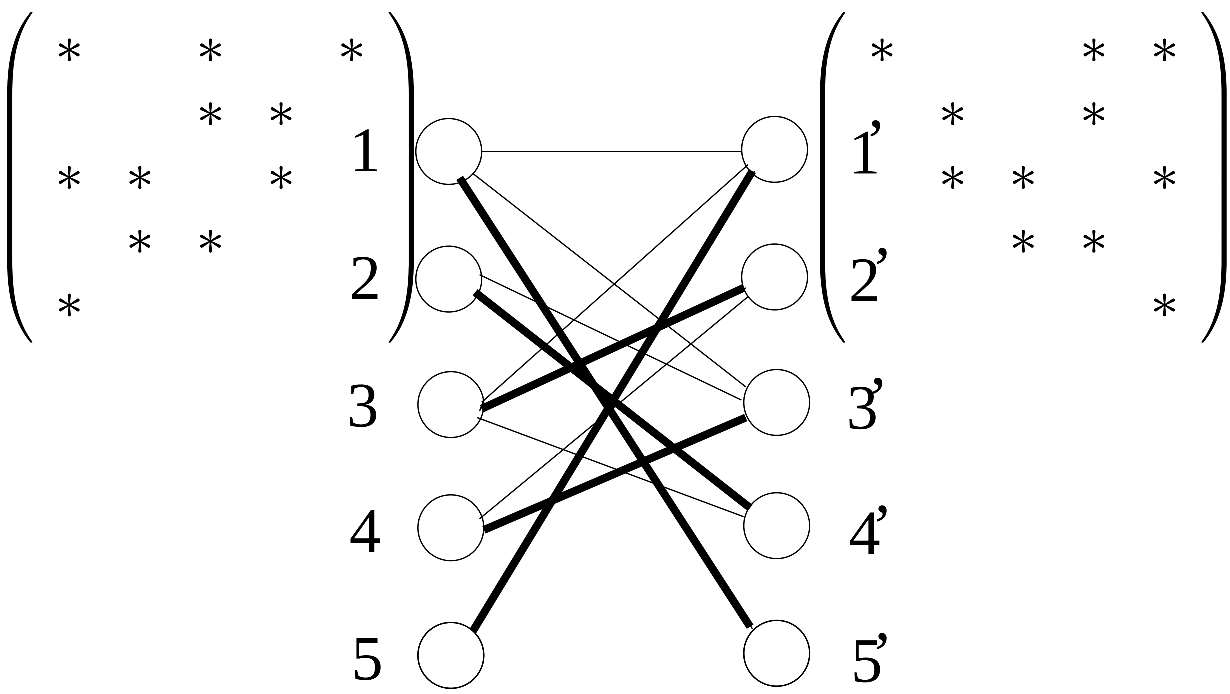
- permutation to get a nonzero diagonal – a classical technique for nonsymmetric matrices (Duff, 1977)

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



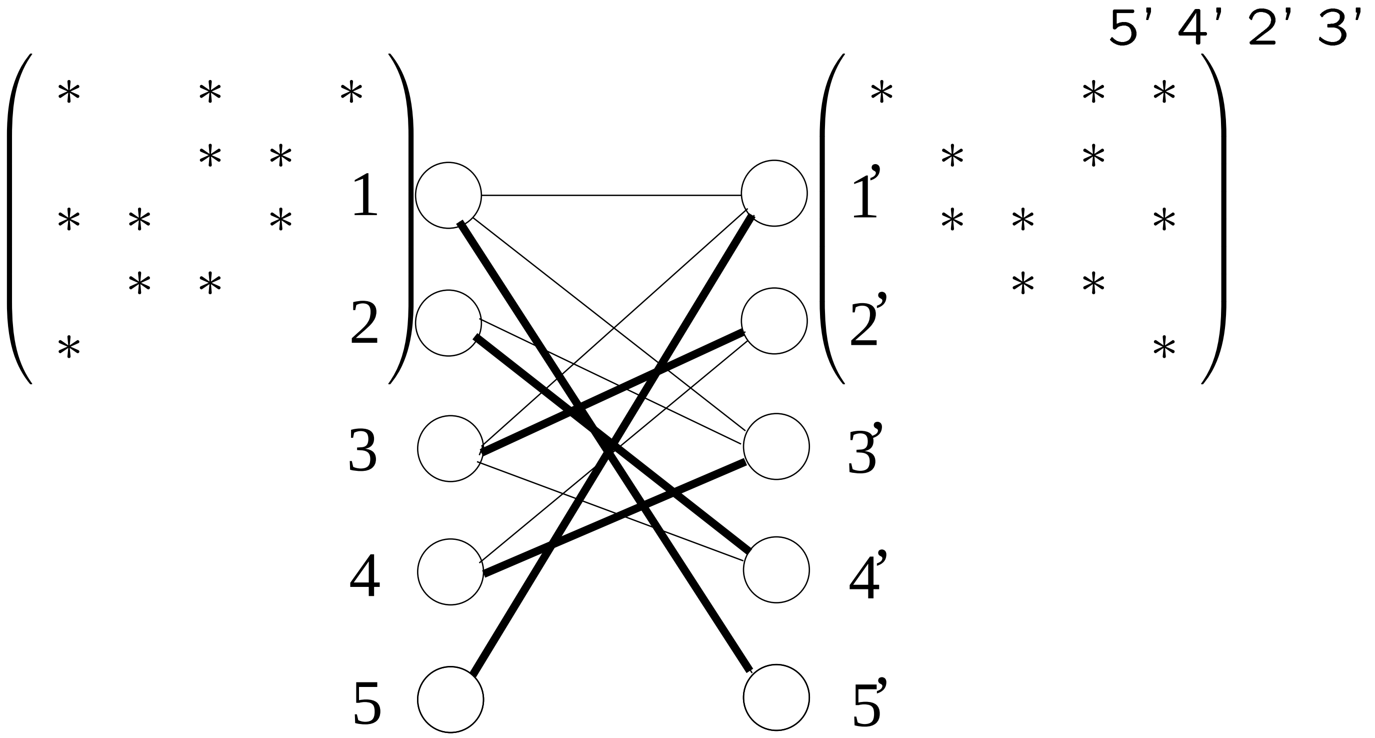
Powerful matrix preorderings: nonsymmetric case

- permutation to get a nonzero diagonal – a classical technique for nonsymmetric matrices (Duff, 1977)



Powerful matrix preorderings: nonsymmetric case

- permutation to get a nonzero diagonal – a classical technique for nonsymmetric matrices (Duff, 1977)



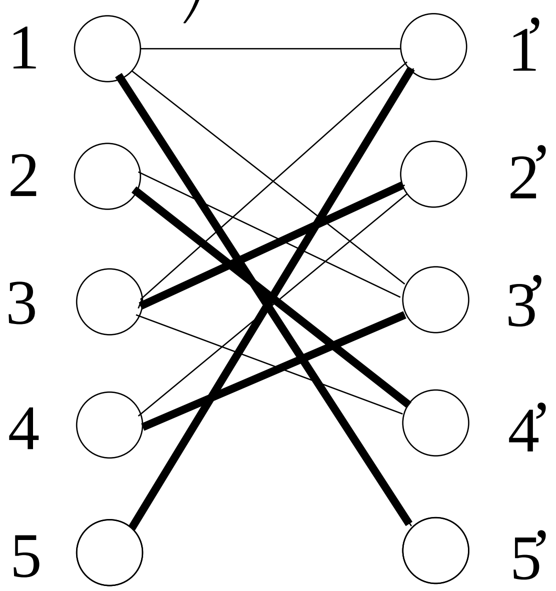
Powerful matrix preorderings: nonsymmetric case

- strengthening diagonal/block-diagonal dominance
  - e.g. **sum/product** matching problem – maximize
    sum/product of modules of transversal entries
  - Olschowka, Neumaier, 1999; Duff, Koster, 1997, 2001; Benzi, Haws, T., 1999.
  - but: permutations are generally nonsymmetric

Symmetric case

- Idea of Iain Duff and John Gilbert (2002) – split the loops of a nonsymmetric permutation
- our example:

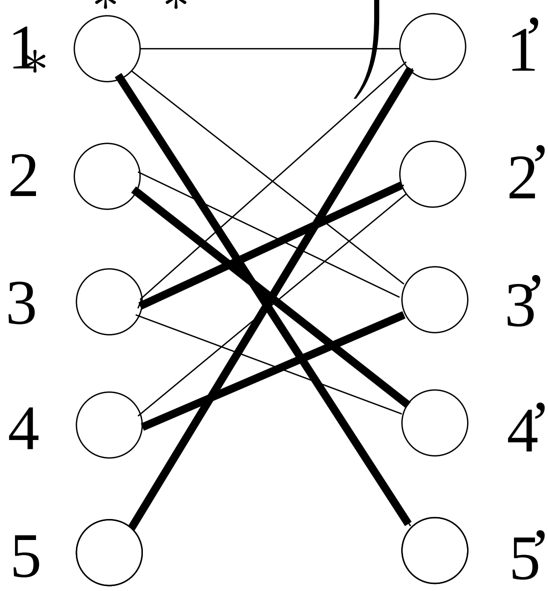
$$\begin{pmatrix} * & & * & & * \\ & & * & * & \\ * & * & & * & \\ & * & * & & \\ * & & & & \end{pmatrix}$$

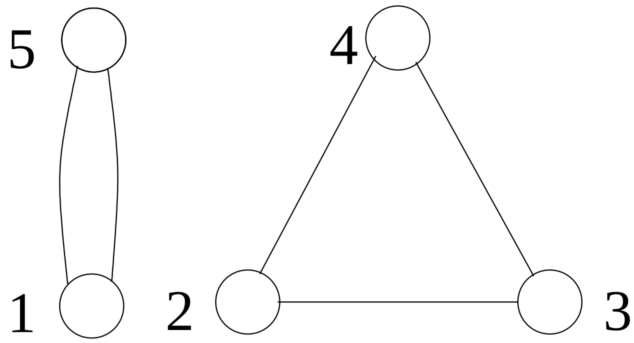


Symmetric case

- Idea of Iain Duff and John Gilbert (2002) – split the loops of a nonsymmetric permutation
- our example:

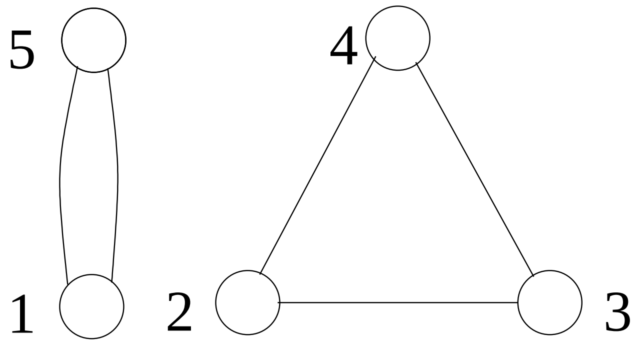
$$\begin{pmatrix} * & & * & & * \\ & & * & * & \\ * & * & & * & \\ & * & * & & \\ * & & & & \end{pmatrix}$$





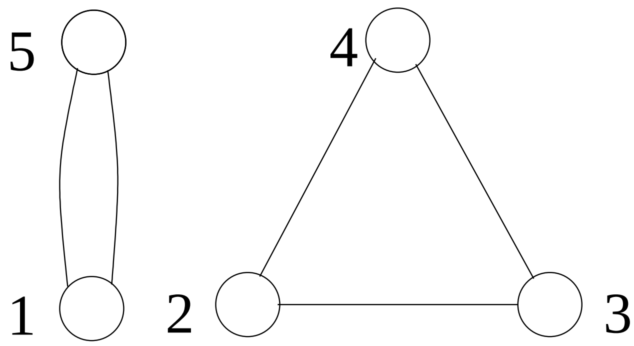
Symmetric case: symmetrized permutation based on the loops

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



Symmetric case: symmetrized permutation based on the loops

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

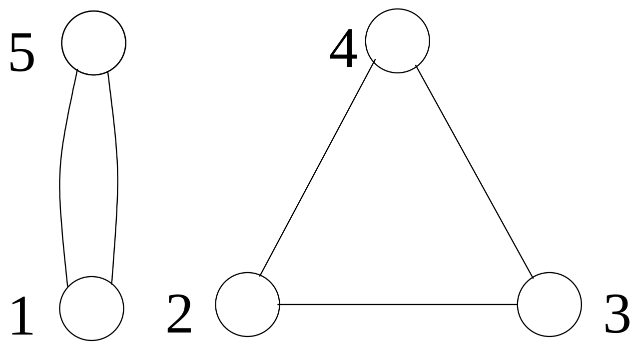


$$\Downarrow$$

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

Symmetric case: symmetrized permutation based on the loops

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



$$\Downarrow$$

$$\begin{pmatrix} \boxed{0} & \boxed{*} & & * & \\ \boxed{*} & \boxed{0} & & & \\ & & \boxed{0} & \boxed{*} & * \\ * & & \boxed{*} & \boxed{0} & * \\ & & * & * & \boxed{0} \end{pmatrix}$$

Symmetric case: symmetrized permutation based on the loops

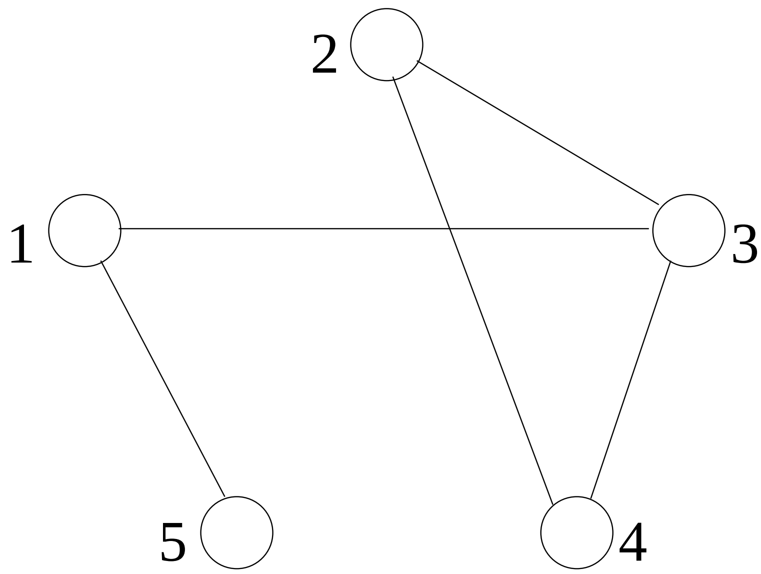
- summarized idea:
  bipartite matching → loops → general matching

- previous work based on this strategy:
  - static preordering for direct methods: Duff, Pralet, 2004; additional criterion: based on sparsity of rows/columns
  - preordering for approximate decompositions
    for preconditioning : Hagemann, Schenk, 2004

New idea: general graph matching

- avoids problems with splitting odd loops (not the problems with odd n)
- how to define graph edge weights?

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

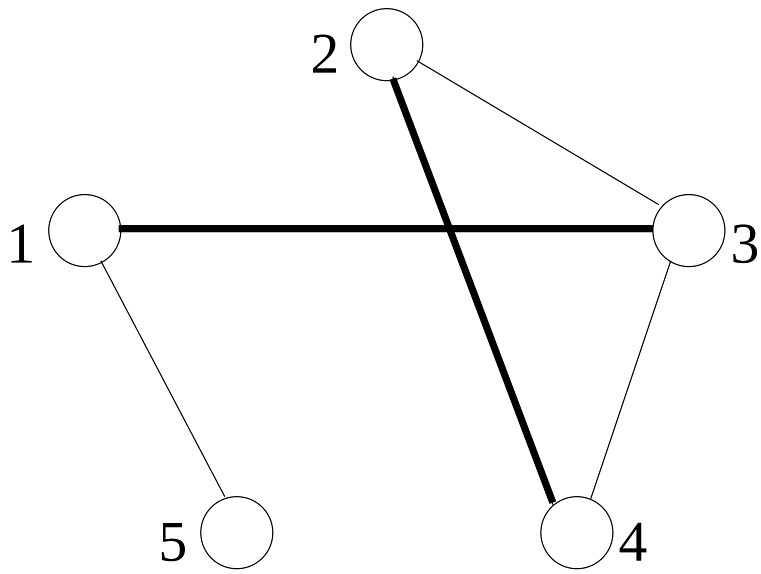


New idea: general graph matching

- avoids problems with splitting odd loops (not the problems with odd n)

- how to define graph edge weights?

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



General graph matching: how to define graph weights

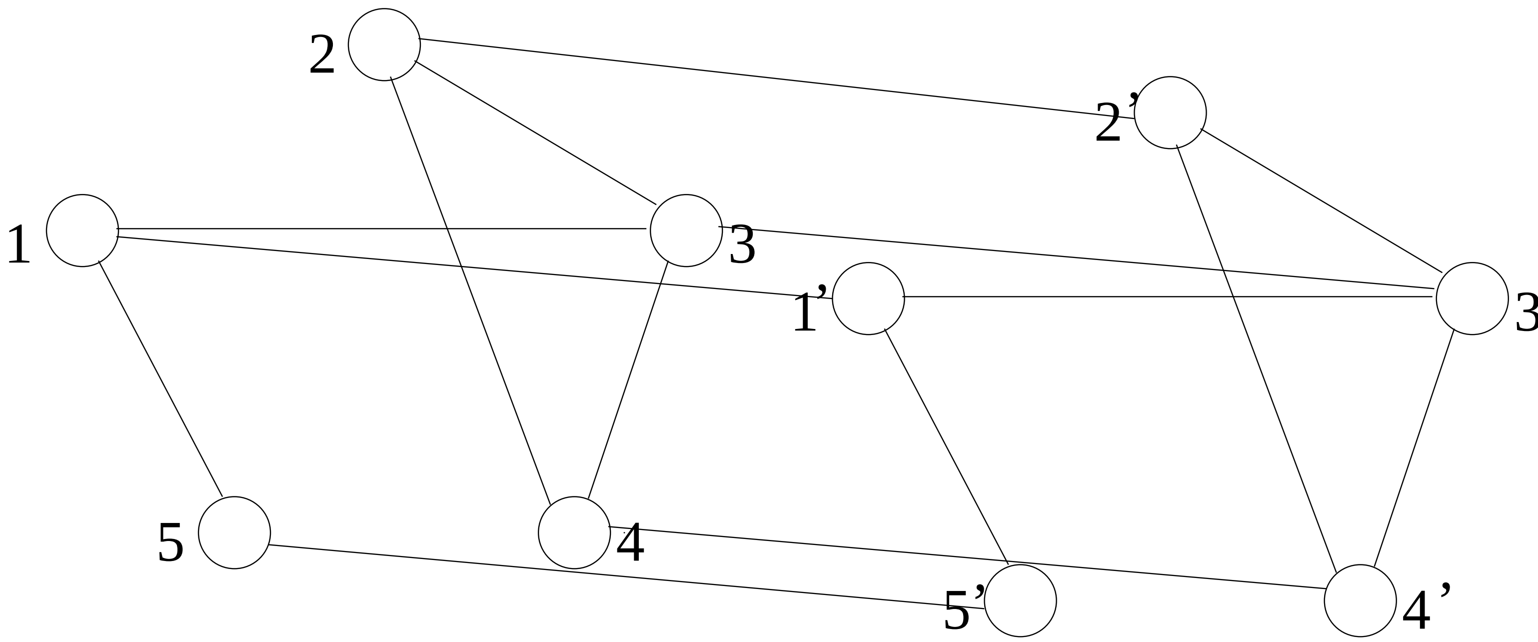
- first possibility:

$$weight_{ij} = |a_{ij}| + \alpha(|a_{ii}| + |a_{jj}|)$$

- α balances influence of diagonals and off-diagonals

General graph matching: how to define graph weights

- another approach: weights are entry sizes; derived (doubled) graph



4. Preconditioning strategies

- extended set of options with respect to Benzi, T. 2002
- right-looking (submatrix) implementation of incomplete decompositions
- Bunch-Parlett-Kaufmann family of pivoting options
  - Bunch-Parlett with various pivotings
  - Bunch-Kaufmann variations
  - bounded Bunch-Kaufmann (Ashcraft, Grimes, Lewis, 1997)
  - Bunch tridiagonal pivoting (Bunch, 1973; Hagemann, Schenk, 2004)
  - Bunch-Kaufmann pentadiagonal pivoting
  - approximate LDL^T decomposition

Other implemented preconditioners

- sparse LTL^T decomposition
  - slow (fill-in in exact case given by $\sum_{i=1}^{n} adj(T[i])$ see Ashcraft, Grimes, Lewis, 1997)
  - if incomplete, large growth in the sub-matrix
- saddle-point reconstruction
  - $A = \begin{pmatrix} \hat{A} & \hat{B} \\ \hat{B}^T & -\hat{C} \end{pmatrix}$
  - sometimes useful
  - not much improvement for strongly in-definite problems
- block diagonal, block symmetric Gauss-Seidel; blocks based on matchings or TPABLO (O'Neil, Szyld,1990)

6. Experimental results

- preconditioned MINRES (implemented as smoothed CG;
  similar results for preconditioned symmetric QMR)
- MMD on blocks/vertices,
  (see Qu, Fish, 2001; cf. Benzi, Szyld, Van Duin, 1999)
- presenting results with approximate diagonal pivoting decomposition; tridiagonal pivoting in most cases; whenever this fails, replaced by Bunch-Kaufmann pivoting
- bipartite matching: MC64 (Duff, Koster, 2001; HSL)
- general matching by greedy heuristics (tested also Blossom 3 (Cook, Rohe, 1998); SMP (Burkard, Derigs, 1980); WMATCH (Rothberg, 1973)).

- in some cases: other preprocessings are better:
  e.g., TPABLO (O'Neil, Szyld, 1990)
- stopping criterion: relative residual norm reduction by 10^{-8}
- a subset of matrices from Li, Saad (2004); Hagemann, Schenk (2004)

Tested matrices

$Matrix$	n	nz
C-41	9769	55757
C-19	2327	12072
C-64	51035	384438
C-70	68924	363955
C-71	76638	468096
traj33	20006	504090
traj27	17148	242286
stiff5	33410	177384
mass05	33410	241140
heat02	10295	90129

Results of experiments

$Matrix$	symm matching		general matching		no
	$Size\_p$	its	$Size\_p$	its	Siz
C-41	118323	315	117168	129	100
C-19	26411	7	26805	8	278
C-64	604274	63	615061	55	493
C-70	1186138	12	1162256	9	865
C-71	1421761	15	1421994	45	1388
traj33	221223	464	102629	186	102
traj27	106633	471	105819	140	104
stiff5	202983	80	217119	72	287
mass05	41817	24	41216	52	568
heat02	247912	34	410773	45	631

7. Conclusions

- approximate diagonal pivoting preconditioning is becoming a standard and reasonably reliable strategy
- block preprocessing techniques can improve the behavior; it should be developed further
- all preprocessings: still a gap before getting to be mature
- solving general indefinite systems is very difficult: in many cases only one specialized technique (Schur complement approach) works