Laboratoire de l'Informatique du Parallélisme

École Normale Supérieure de Lyon
Unité Mixte de Recherche CNRS-INRIA-ENS LYON n° 5668

CENTRE NATIONAL DE LA RECHERCHE SCIENTIFIQUE

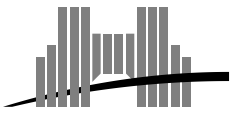
SPI

Efficient Matrix Preconditioners for Black Box Linear Algebra

L. Chen, W. Eberly, E. Kaltofen, B.D. Saunders, W.J. Turner, G. Villard

Janvier 2001

Research Report N° 2001-05



École Normale Supérieure de Lyon
46 Allée d'Italie, 69364 Lyon Cedex 07, France
Téléphone : +33(0)4.72.72.80.37
Télécopieur : +33(0)4.72.72.80.80
Adresse électronique : `lip@ens-lyon.fr`

INRIA

Efficient Matrix Preconditioners for Black Box Linear Algebra

L. Chen, W. Eberly, E. Kaltofen, B.D. Saunders, W.J. Turner, G. Villard

Janvier 2001

Abstract

The main idea of the "black box" approach in exact linear algebra is to reduce matrix problems to the computation of minimum polynomials. In most cases preconditioning is necessary to obtain the desired result. Here, good preconditioners will be used to ensure geometrical / algebraic properties on matrices, rather than numerical ones, so we do not address a condition number. We offer a review of problems for which (algebraic) preconditioning is used, provide a bestiary of preconditioning problems, and discuss several preconditioner types to solve these problems. We include new conditioners, new analyses of preconditioner performance, and results on the relations among preconditioning problems and with linear algebra problems. Thus improvements are offered for the efficiency and applicability of preconditioners. The focus is on linear algebra problems over finite fields, but most results are valid for entries from arbitrary fields.

Keywords: Linear algebra, randomized algorithms, black box matrix, sparse matrix, exact arithmetic, finite fields, linear systems, rank, preconditioner.

Résumé

L'idée principale de l'approche à base de boîtes noires en algèbre linéaire exacte est de ramener la résolution de problèmes matriciels à des calculs de polynômes minimaux. Dans la plupart des cas, un préconditionnement s'avère nécessaire pour obtenir le résultat désiré. Nous parlerons de "bons" préconditionnements quand il s'agira d'assurer des propriétés géométriques et algébriques sur les matrices plutôt que des qualités numériques, donc sans relation avec un nombre de conditionnement. Dans ce rapport nous passons en revue divers problèmes pour lesquels on utilise un préconditionnement (algébrique), proposons une classification des différents problèmes de préconditionnement et étudions plusieurs solutions. En particulier, nous envisageons de nouveaux préconditionnements, de nouvelles analyses de leurs performances et des relations pouvant être définies entre eux en rapport avec les questions d'algèbre linéaire qu'ils résolvent. Des améliorations sont donc obtenues quant à l'efficacité et l'applicabilité des préconditionnements. Ces résultats se concentrent sur des matrices à coefficients dans des corps finis mais ils s'appliquent dans le cas de corps commutatifs quelconques.

Mots-clés: Algèbre linéaire, algorithmes probabilistes, matrice boîte noire, matrice creuse, arithmétique exacte, corps finis, systèmes linéaires, rang, préconditionnement.

Efficient Matrix Preconditioners for Black Box Linear Algebra *

L. Chen[1] W. Eberly[2], E. Kaltofen[3], B.D. Saunders[1], W.J. Turner[3], G. Villard[4]

[1]Dpt. of Comp. and Infor. Sc., University of Delaware, Newark, Delaware 19716;
lchen,saunders@mail.eecis.udel.edu, http://www.cis.udel.edu/ lchen, saunders

[2]Dpt. of Comp. Sc., University of Calgary, Alberta, Canada T2N 1N4;
eberly@cpsc.ucalgary.ca, http://www.cpsc.ucalgary.ca/ eberly

[3]Dpt. of Math., North Carolina State University, Raleigh, North Carolina 27695-8205;
kaltofen,wjturner@math.ncsu.edu, http://www.math.ncsu.edu/ kaltofen, wjturner

[4] Cnrs, Projet Cnrs / Ens Lyon / Inria Arenaire, Ecole normale supérieure de Lyon;
Gilles.Villard@ens-lyon.fr, http://www.ens-lyon.fr/ gvillard

Abstract

The main idea of the "black box" approach in exact linear algebra is to reduce matrix problems to the computation of minimum polynomials. In most cases preconditioning is necessary to obtain the desired result. Here, good preconditioners will be used to ensure geometrical / algebraic properties on matrices, rather than numerical ones, so we do not address a condition number. We offer a review of problems for which (algebraic) preconditioning is used, provide a bestiary of preconditioning problems, and discuss several preconditioner types to solve these problems. We include new conditioners, new analyses of preconditioner performance, and results on the relations among preconditioning problems and with linear algebra problems. Thus improvements are offered for the efficiency and applicability of preconditioners. The focus is on linear algebra problems over finite fields, but most results are valid for entries from arbitrary fields.

1 Introduction

In the black box approach [15] one takes an external view of a matrix: It is a linear operator on a vector space. Information is derived from a series of applications of this operator to vectors. By contrast most matrix algorithms are internal, involving some sort of elimination process. The black box approach is particularly suited to the handling of large sparse or structured matrices over finite fields. This fact – well known in the numerical computation area – has led the computer algebra community to a considerable interest in black box algorithms for linear algebra. Many developments have been proposed to adapt Krylov or Lanczos methods to fast exact algorithms. Wiedemann's paper [24] was the seminal work to these developments. He showed how to solve an invertible $n \times n$ linear systems using $O(n)$ matrix-vector products, $O(n^2)$ additional arithmetic operations in the entry field, and $O(n)$ space for intermediate results. Since matrix-vector product costs at most $O(n^2)$ operations, Wiedemann's algorithm is asymptotically competitive with elimination. For many problems the operator application, the matrix-vector product Av, may be economically computed both in time and/or in space. Problems of interest may have cost $O(n \log(n))$, even $O(n)$. In these cases the black box approach is a substantial improvement over elimination. When the matrix is sparse, and elimination is

*This material is based on work supported in part by the National Science Foundation under Grants nos. CCR-9712267, CCR-9988177 and DMS-9977392 (Kaltofen and Turner), INT-9726763 (Kaltofen and Saunders), and CCR-9712362 (Chen and Saunders), by the Natural Sciences and Engineering Research Council of Canada (Eberly), and by the Centre National de la Recherche Scientifique, Action Incitative no. 5929 (Villard). This text is also available as a research report of the Institut National de Recherche en Informatique et en Automatique http://www.inria.fr.

subject to fill-in it also has the important advantage of modest space demand. Other examples are matrices that have efficient procedures for generating their entries, for instance, the Hilbert matrix. A black box algorithm never constructs such a matrix, hence is substantially more space efficient.

To solve several problems using the algorithms invented by Wiedemann and his followers the black box coefficient matrix needs to be preconditioned. As detailed in sections 2 and 3, the preconditioning allows to reduce problems to the computation of minimum polynomials and leads to faster solutions. Common preconditioners, some already known to Wiedemann, are matrix pre- and postmultipliers. These multiplier matrices must have efficient matrix-vector products in order to avoid a too high slow-down of the matrix-vector product for the resulting preconditioned matrix. Our target problems discussed are linear system solution, determinant, and rank. Solutions to additional problems such as Diophantine problems (over the integers) and Smith form computation [7, 8, 18], also involve these preconditioners. Future work may concern the use of preconditioners to compute the characteristic polynomial of a matrix and matrix normal forms as well.

We present more efficient preconditioners for most of the problems discussed above. Most of our preconditioners apply to matrices over an arbitrary field, but our focus is on matrices over a finite field. Our time cost analyses are in terms of number of arithmetic operations in the element field and our space cost is measured in number of field elements. Finite fields are categorized as large or small, depending on whether they have sufficiently many elements to support those randomized methods for which the Schwartz-Zippel lemma [21, 25] (see also [3]) is used in the probability analysis. We organize solutions around this distinction and offer new results for large and small fields.

In section 2 we offer a list of problems to which preconditioning has been applied with a discussion of the solution methods advanced to date. In section 3 the notion of a preconditioning problem and preconditioner are given precise definitions and a list of useful preconditioning problems is offered. The problems are of three general types: linear independence (localizing it), nilpotent blocks (avoiding them), and cyclicity (achieving it, for the nonzero eigenvalues). Results on relations among them are also in section 3. Notably, Wiedemann already used three kinds of preconditioner : diagonal, Beneš permutation network, and sparse preconditioners. The usefulness of diagonal conditioners is extended and their effects more thoroughly examined in section 4. Regarding Beneš' network-based preconditioners, we show that the size can be cut in half yielding a butterfly network, the individual switches can be simplified, and the network can generalized to arbitrary dimensions that are not powers of 2 (see section 5). And in section 6 we prove that Wiedemann's sparse preconditioners can be used directly for the inhomogeneous system solution problem for matrices over small finite fields without the need of binary search.

2 List of Matrix Problems and Solutions

We present our target linear algebra problems which we label as MinPoly, LinSolve0, LinSolve1, Det, and Rank. We discuss the use of various preconditioners to provide reductions between problems and list known solutions.

The objectives are to find algorithms running within the costs stated in the introduction: with $n(\log n)^{O(1)}$ black box calls, $n^2(\log n)^{O(1)}$ additional operations in the entry field and using $n(\log n)^{O(1)}$ intermediate storage [11, Open Problem 3]. Throughout the paper, when we say of a problem "the question is *open*" or "it is an *open* problem", we mean that the question has no known solution within these resource limitations (see for instance the certificate of system inconsistency or the computation of the determinant). Most of the solutions listed below are randomized. Such algorithms are *Monte Carlo* if the answer returned is possibly wrong (with quantified probability of error), and are called *Las Vegas* or are said to have a *certificate* if the solution is always correct and unluck in the random choices can only cause violation of the resource limitations promised. We say problem A is *reducible* to problem B if A may be solved by computing an instance for problem B in such a way that the overall cost is within the resource limitations assuming an algorithm for B which meets the resource limitations. Problems A and B are *equivalent* if reductions exist both ways.

For a matrix $A \in \mathbb{F}^{n\times n}$ over a field $\mathbb{F}$, and vectors $u_1, \ldots, u_\kappa \in \mathbb{F}^n$, the black box algorithms (adaptations of Krylov, Lanczos, or conjugate gradient algorithms) essentially compute minimal relations in Krylov spaces constructed from the vectors. Using block size $\kappa = 1$, we have *scalar algorithms* to compute the minimum

polynomial of the vector with respect to A [24]. With $\kappa \geq 1$, we obtain *block algorithms* to compute a matrix minimum polynomial [22]. With high probability this will give the minimum polynomial of A or a multiple of it and thus a solution to the first problem to consider:

MinPoly - Compute the minimum polynomial of A.

Over any field, a Monte Carlo solution is given by the original Wiedemann's algorithm [24]. It is not known how to *certify the result* and how to recover the minimum polynomial from the blocked versions (only a multiple is computed).

The next problem to consider is LinSolve0, the computation of a nonzero vector in the nullspace of a singular matrix. We remark that the goal of homogeneous linear systems solving is often taken to be to compute a nullspace basis. However, for sparse or structured matrices of low rank, to compute a basis for the nullspace will entail construction of many dense vectors, which will be vastly more space consuming than the original matrix. Such a project would be antithetical to the spirit of sparse methods. One may as well use a dense method if the goal is a basis and the rank is low. Thus we pose as the basic problem to compute one vector in the nullspace.

In the discussion of this and the following problems we refer to certain preconditioners (see section 3). The preconditioners are categorized by purpose and given names such as PreCondNil, PreCondCyc, etc. For example, the problem PreCondNil is to produce a matrix equivalent to A which has no nilpotent blocks in its Jordan form. These must be understood by forward reference to section 3 where the preconditioning problems are discussed in detail. Summary of the existing solutions to these preconditioning problems and presentation of improved methods for them is the central purpose of this paper. By abuse of notation we also use the label of a preconditioner problem to refer to the use of a solution to that problem in a computation.

LinSolve0 - Compute $w \neq 0$ such that $Aw = 0$.

This also gives a singularity certificate and a Monte Carlo test for nonsingularity: If any of the algorithms repeatedly fails, the matrix likely is nonsingular.

Over small fields, the block Wiedemann algorithm [2] together with tricks in [10] leads to $(1+\varepsilon)n$ or $(2+\varepsilon)n$ matrix-times-vector products. Complete analyses may be found in [10, 22, 23]. Comparisons with the block Lanczos algorithm are under development. Both may incorporate the *early termination strategy* first observed by A. Lobo. If the minimum polynomial has small degree, the solution is found without completing the sequence to the worst case length. This criterion, probabilistically correct for randomly preconditioned matrices, is incorporated in Lanczos variants [6]. Over large fields, a Lanczos variant of the block Wiedemann algorithm should be superior since [5] shows that *look-ahead* is unlikely.

Special case SymReal - If A is symmetric and $\mathbb{F}$ is a subfield of the real numbers then unblocked Lanczos should be used to solve the system $Ax = Ay$ for a randomly chosen y. With high probability, $x - y$ is a nonzero element of the nullspace if A is singular. Here n matrix-times-vector products are sufficient.

Over a field of positive characteristic problems arise due to the possibility of self-orthogonal rows in a symmetric matrix A and the possibility of nontrivial nilpotent blocks in its Jordan form. If $\mathbb{F}$ is sufficiently large and its characteristic is not two, then PreCondNil together with the above solution allows the problem to be solved with n matrix-times-vector products [5].

It is not known whether an additional black box for A^{T} can improve the above methods.

LinSolve1 - Given A and b, compute x such that $Ax = b$.

The problem of finding a random element of the nullspace (call it *random*-LinSolve0) is equivalent to LinSolve1. To solve random-LinSolve0, solve $Ax = Ay$, where y is a random vector. In the reverse direction, consider $[A \mid b]w = 0$.

The reduction of LinSolve1 to LinSolve0 is immediate if A is nonsingular. The preconditioner PreCondNil together with a block algorithm as discussed under LinSolve0 solves LinSolve1 (see [23] over large fields). A solution to Rank together with the preconditioner PreCondrxr also solves LinSolve1 [14]. A certificate for inconsistency is known only with an additional black box for A^{T} [9]. Without a transpose box, the problem is open.

DET - Compute $\det(A)$. The problem is open over small fields except for $\mathbb{F}_2$ where one may use the singularity test mentioned in LINSOLVE0. It is also open how to certify that $\det(A) \neq 0$. Over large fields, a solution to the problem MINPOLY together with the preconditioner PRECONDCYC solves DET [24].

RANK - Compute the rank of A. A Monte Carlo algorithm uses the preconditioner PRECONDSXS and the singularity test mentioned under LINSOLVE0 to find the rank by binary search [24], using $O(\log n)$ calls to the sparse solver. It is open how to avoid using $\Omega(\log n)$ of these calls. The problem is also solved over large fields with the preconditioner PRECONDCYCNIL and a solution to problem MINPOLY, see [14]. These Monte Carlo algorithms may underestimate the rank. However, the rank can be certified over real fields [20].

3 Preconditioners in General

Matrix problems on A may be reduced to simpler problems on a well chosen matrix A' called a preconditioning of A. This section is intended to define precisely what we mean by a preconditioning problem. New preconditioners for some of the problems will be given in sections 4 and 5. We also derive reductions between preconditioning problems that help in characterizing the preconditioners themselves and will lead to new preconditioners with the sparse matrices of section 6.

A *preconditioning problem* is a pair $(\mathcal{R}, \mathcal{C})$ of a *relation* $\mathcal{R}$ and a *condition* $\mathcal{C}$ on matrices in a given class $\mathcal{M}$. A solution to a preconditioning problem is a mapping $A \to A'$ on $\mathcal{M}$ such that (1) $\mathcal{C}(A')$ holds, and (2) $\mathcal{R}(A, A')$ holds. We say that A' is *good* for A with respect to the given preconditioning problem.

Generally speaking, $\mathcal{C}$ is a property desired so that the input conditions of some computational technique are satisfied, $\mathcal{R}$ is a relation needed in order for results computed concerning A' to yield information about A. For most preconditioners used in linear systems solving, the relation $\mathcal{R}$ is matrix *equivalence*: $A' = LAR$ for L and R two invertible matrices. However some of the existing preconditioners are symmetrizing products involving A^{T}, for which the relation is preservation of rank [6]. All of the preconditioners discussed in this paper are multiplicative, A' being a product involving A, nonsingular *scaling* matrices, and sometimes A^{T}.

A preconditioner $A \to A'$ is *generic* if it is good for all $A \in \mathcal{M}$. The central issue determining the usefulness of a preconditioner is usually that computation with A' be as inexpensive as possible, preferably within a constant factor of the cost with A alone. A generic preconditioner with good computational performance is generally not possible to achieve. Generic preconditioners usually involve scaling the given matrix by a multiplier whose entries are multivariate polynomials over the field of the entries of A. They are useful as a step in construction of families of preconditioners whose scalings have entries in the field of entries (or a small extension thereof). The individual members of a family of preconditioners are obtained by substitution of random field elements for the variables in a generic preconditioner. The distribution of the preconditioners in such a family should have the property that for all $A \in \mathcal{M}$ the probability that a preconditioner $A \to A'$ chosen at random is good for A is at least p, for a specified probability p. When we solve a preconditioner problem with a random family in this way we prefix the preconditioner name with "p-". For example we may speak of a p-PRECONDIND preconditioner.

3.1 Preconditioning Problems

Since we reduce problems to computing minimum polynomials, the preconditioning questions we address are related to modifications of Jordan structures of matrices. In general the purpose is to ensure diagonalizability conditions which may themselves follow from independence properties (see paragraph 3.4). We distinguish three main types of preconditioners : *linear independence* conditioners, *nilpotent block* conditioners (to avoid nontrivial ones in the Jordan form), cyclicity conditioners (to ensure cyclicity – only one Jordan block – of the nonzero eigenvalues).

Solutions to the following problems will be proposed in subsequent sections. These problems are listed with the target conditions $\mathcal{C}$ on A' and the solutions for small fields and large fields, where large means big enough for the use of the Schwartz-Zippel lemma. Generally preconditioners to be applied to LINSOLVE preserve the matrix equivalence relation. Preconditioners to be applied to DET or RANK may potentially

preserve a weaker condition. For example, in the following list PRECONDSQUFREE preserves (an unknown) rank while the others preserve matrix equivalence.

Linear Independence Preconditioning.

PRECONDIND - The r leading columns of A' are linearly independent, where r is the rank of A (see LINSOLVE1). Over small fields, see the solution of [24] presented in section 6.

PRECONDRXR - The $r \times r$ leading principal minor of A' is nonzero, where r is the rank of A (see LINSOLVE1). Over small fields, $A' = W_1 \cdot A \cdot W_2$ where W_i are the sparse matrices constructed by [24] (see also section 6). Over large fields, see PRECONDGEN but note that the failure probabilities are smaller for this condition.

PRECONDSXS - Given s, if $s \leq$ the rank of A, the $s \times s$ leading principal minor of A' is nonzero, (see RANK).

PRECONDGEN - All leading principal minors of A' of size up to and including the rank are nonzero. This condition was given the name *generic rank profile* in [12]. The question of efficient PRECONDGEN is open over small fields. Over large fields, $A' = B_1 \cdot A \cdot B_2$, where B_i encode symbolic Beneš permutation networks [24]. A new, more efficient solution is given in section 5 below. Another is $A' = T_{\text{upper}} \cdot A \cdot T_{\text{lower}}$, where T_{upper} is a random unit upper triangular Toeplitz matrix and T_{lower} is a random unit lower triangular Toeplitz matrix [14]. This is less efficient but useful for matrices of low displacement rank [10, Appendix]. PRECONDGEN may be reduced to PRECONDIND, see theorem 3.1. If A is nonsingular, the preconditioner may be reduced to a single multiplier, which may be on either side.

These independence preconditioners were used for instance in [24] to compute the rank by binary search. They are also a main ingedient to construct *nilpotent block preconditioners* basically used for LINSOLVE1 (see theorem 3.5):

Nilpotent Block Preconditioning.

PRECONDNIL - A' has no nilpotent blocks of size greater than 1 in its Jordan canonical form (see LINSOLVE1). A reduction to independence preconditioners is proposed in section 3.4. Over small fields, $A' = W_1 \cdot A \cdot W_2$ is a solution, where W_1 and W_2 are sparse matrices as shown in section 6. For large fields when A symmetric, use $A' = D \cdot A$ or $D \cdot A \cdot D$ where D is a random diagonal matrix as established in section 4.

For problems as DET or RANK, independence preconditioners are too weak, known reductions to MINPOLY need to modify the invariant structure of the matrix. The corresponding *cyclicity preconditioners* may be classified with respect to the effect they have on the nonzeroand on the zero eigenvalues:

Cyclicity Preconditioning.

PRECONDCYC - For A nonsingular, A' is nonsingular and cyclic: char-poly(A') = min-poly(A'). For problem DET, the det(A) must be easily derivable from det(A'). The question is open over small fields. Over large fields, the solution $A' = D \cdot A$ given in theorem 4.2 below improves previously known solutions that were reducing the problem to PRECONDGEN$(A) \cdot D$ [24]. The solution to PRECONDGEN based on Toeplitz matrices is also sufficient here [13].

PRECONDCYC-X - The nonsingular part of A' is cyclic: char-poly(A') = min-poly$(A') \cdot x^l = f(x) \cdot x^k$ where $f(0) \neq 0$ ($\deg(f) + k - l - 1$ is then a lower bound for the rank). Over large fields, $A' = D \cdot A$, where D is a random diagonal matrix, see theorem 4.2.

PRECONDSQUFREE-X - Same as probf PreCondCyC-x with the additional condition that f is squarefree. If the characteristic of the coefficient field is 0 or is greater than n the same solution $A' = D \cdot A$ works, see theorem 4.3.

PRECONDCYCNIL - The minimum polynomial is $f(x)\cdot x$ and the characteristic polynomial is $f(x)\cdot x^k$ where $f(0) \neq 0$. As a consequence $k = n - \text{rank}(A)$ [14] (see RANK). Over large fields and for A symmetric, $A' = D \cdot A$ or $D \cdot A \cdot D$ where D is a random diagonal matrix, see theorem 4.5. A solution will solve PRECONDNIL, PRECONDCYC-X.

PRECONDSQUFREE - Same as probf PreCondCyCNil with the additional condition that f is squarefree. One also has the same solution in the case of a symmetric matrix when the field characteristic is 0 or greater than n, see theorem 4.7. In the general case, a solution here will also solve PRECONDNIL and PRECONDSQUFREE-X. The question is open over small fields. For large fields, a solution is $A' =$ PRECONDGEN$(A)\cdot D$, where D is a random diagonal matrix [14]. If the transpose black-box is available, $A' = A^{\mathrm{T}} \cdot D \cdot A$, where D is a random diagonal matrix [6].

3.2 Reducibility: Independence Preconditioners

A PRECONDIND scaling for $\mathbb{F}^{*\times n}$ is a solution to PRECONDIND of the form $A' = AR$ with $R \in \mathbb{F}^{n\times n}$ valid for all $m \times n$ matrices: $\mathcal{M} = \bigcup_m \mathbb{F}^{m\times n}$. In this section we show that generic rank profile scaling reduces to two independence scalings.

Theorem 3.1 *Let L be a (row) p-*PRECONDIND *scaling for $\mathbb{F}^{m\times *}$ and let R be a (column) q-*PRECONDIND *scaling for $\mathbb{F}^{*\times n}$. Let $t = 1-(1-pq)n$. Then*

$$h(A) = LAR$$

*forms a pq-*PRECONDRXR *(and pq-*PRECONDSXS*) scaling and a t-*PRECONDGEN *scaling for $F^{m\times n}$. Conversely, if h is a p-*PRECONDRXR *(or* PRECONDGEN*) scaling for $\mathbb{F}^{n\times n}$, defined by $h(A) = LAR$, then R is a p-*PRECONDIND *scaling for $\mathbb{F}^{*\times n}$.*

Proof: Let $B = LA$. Then B has leading r rows independent for $r = \text{rank}(A)$ with probability at least p. For given k, with $1 \leq k \leq \text{rank}(A)$, let B_k denote the matrix consisting of the first k rows of B. Then the leading k columns of B_kR are independent with probability at least pq. This implies that the leading $k \times k$ minor of (full rank) B_kR is nonzero. This minor is also the leading $k \times k$ minor of LAR. The probability that all these principal minors are simultaneously nonzero is at least $t = 1-(1-pq)n$, since each is zero with probability at most $1-pq$.

To prove the second claim, consider a given $m \leq n$ and $A \in \mathbb{F}^{m\times n}$ of rank r. We have that the first r columns of

$$h(A) = L \begin{bmatrix} A \\ 0 \end{bmatrix} R$$

are independent. It follows immediately that the first r columns of AR are independent. ⊠

Remark 3.2 *Considering for A any $n \times n$ matrix with exactly k nonzero columns being distinct canonical vectors shows that if R is a q-*PRECONDIND *preconditioner for $\mathbb{F}^{*\times n}$, then any $k \times k$ determinant of a submatrix formed from the first k columns of R is nonzero with probability at least q.*

3.3 Reducibility: Matrices with Nonzero Minors

The property of independence preconditioners given in remark 3.2 is not sufficient. The minors in the leading k columns must themselves satisfy independence conditions. We show that simply the addition of a *diagonal scaling* will ensure these latter conditions.

Theorem 3.3 *Let Q be a matrix such that all minors in the leading k columns of Q are nonzero, Let $\mathcal{D}$ be a diagonal matrix of indeterminates. Then $\mathcal{D}Q$ is a generic* PRECONDIND *conditioner for $\mathbb{F}^{*\times n}$.*

Proof: Let I and J be sequences of k indices with $J = (1, 2, \ldots, k)$. Denote the minor in rows I columns J of matrix A by $A_{I,J}$. Let the matrix A be conditioned as $B = A' = A\mathcal{D}Q$. Then for each I, the minor $B_{I,J}$ has the expansion

$$B_{I,J} = \sum_K A_{I,K}\mathcal{D}_{K,K}Q_{K,J}.$$

As a polynomial in the indeterminates in $\mathcal{D}$, each summand is a distinct term, since the $\mathcal{D}_{K,K}$ are distinct monomials. As the $Q_{K,J}$ are nonzero if any $A_{I,K}$ is nonzero then $B_{I,J}$ is nonzero and the first k columns of B are independent. ⊠

3.4 Reducibility: Avoiding Nilpotent Blocks

For $A \in \mathbb{F}^{n\times n}$ the *generic nilpotency problem* – PRECONDNIL – is to produce an equivalent matrix A' whose minimum polynomial has valuation one (the nilpotent blocks of the Jordan form have dimension one). By *valuation* we mean the degree of the lowest term. This problem is closely related to the LINSOLVE1 problem.

Lemma 3.4 *Let $A \in \mathbb{F}^{n\times n}$. Then the minimum polynomial of A has valuation one if and only if* rank $A^2 =$ rank A.

Proof: Let $J = T^{-1}AT =$ diag $(\mathcal{J}_1, \ldots \mathcal{J}_\lambda, \mathcal{N}_1, \ldots, \mathcal{N}_\nu)$ be the Jordan normal form of A with λ blocks $\mathcal{J}_j$ having nonzero eigenvalues and ν nilpotent blocks $\mathcal{N}_j$. They satisfy rank $\mathcal{J}_j^2 =$ rank $\mathcal{J}_j$, $1 \le j \le \lambda$, and rank $\mathcal{N}_j^2 =$ rank $\mathcal{N}_j$ $(= 0)$ if and only if $\mathcal{N}_j = [\ 0\ ]$, $1 \le j \le \nu$. The assertion of the lemma follows since rank $A^2 =$ rank J^2. ⊠

This naturally leads to the fact that preconditioners ensuring independence and rank properties give preconditioners for the generic nilpotency problem.

Theorem 3.5 *Let L be a (row) p-*PRECONDIND *scaling for $\mathbb{F}^{n\times *}$ and let R be a (column) p-*PRECONDIND *scaling for $\mathbb{F}^{*\times n}$. If in addition, for A of rank r and $Q \in \mathbb{F}^{(n-r)\times r}$, the columns of*

$$AR \begin{bmatrix} I_r \\ Q \end{bmatrix} \tag{1}$$

are independent with probability at least q then LAR and ARL have rank r and their minimum polynomials have valuation one with probability at least pq.

Proof: For A of rank r and two matrices L and R with appropriate dimensions, if rank $ARLA = r$ then rank $AR = r$. Thus the column space of R together with the right nullspace of A and the one of AR generates all of $\mathbb{F}^n$ and rank $ARLAR = r$. This also implies that the row space of L together with the left nullspace of A generates all of $\mathbb{F}^n$ and rank $(LAR)^2 = r$. In the same way we deduce from rank $ARLAR = r$ that rank $(ARL)^2 = r$. Since the converse statements are true we have:

$$\begin{aligned} \text{rank } (ARL)^2 = r &\iff \text{rank } ARLA = r \\ &\iff \text{rank } (LAR)^2 = r. \end{aligned} \tag{2}$$

Now if L is such that the first r rows of LA are independent, let T be an invertible matrix such that

$$LAT = \begin{bmatrix} I_r & 0 \\ Q & 0 \end{bmatrix}.$$

Then

$$\text{rank } AR \begin{bmatrix} I_r \\ Q \end{bmatrix} = r \implies \text{rank } ARLA = r.$$

Thus for L and R the preconditioners of the theorem, rank $ARLA = r$ with probability at least pq and using (2) together with lemma 3.4 the theorem is proven. ⊠

We will establish is section 6 that a particular class of sparse matrices used in [24] fulfills the requirements of the theorem. As for remark 3.2 we have:

Remark 3.6 *Taking for A in (2) any $n \times n$ matrix with exactly k nonzero columns being distinct canonical vectors shows that if $C = RL$ is a p-*PRECONDNIL *preconditioner then any of its minors is nonzero with probability at least p.*

From independence preconditioners L and R, the additional condition (1) could be ensured up to a diagonal scaling $R\mathcal{D}$ by analogy with theorem 3.3.

4 Diagonal Preconditioners

Recall that the *invariant factors* of a matrix A are polynomials $f_1, \ldots, f_s$ such that $f_1, \ldots f_s$ is the characteristic polynomial of A, f_i divides f_{i+1} for $1 \leq i < s$, and f_s is the minimal polynomial of A. A matrix A is *cyclic up to nilpotent blocks* if the invariant factors $f_1, \ldots, f_{s-1}$ are monomials in x, that is, if the ratio of the characteristic polynomial to the minimal polynomial is a monomial in x.

Lemma 4.1 *Let A be a square matrix over an integral domain and let $\mathcal{D} = diag(\delta_1, \ldots, \delta_n)$, where $\delta_1, \ldots, \delta_n$ are distinct indeterminates over the domain. Then $\mathcal{D}A$ is cyclic up to nilpotent blocks and the minimal polynomial of $\mathcal{D}A$ is the product of a squarefree polynomial and a power of x.*

Proof: It is necessary and sufficient to prove that the characteristic polynomial $C(x)$ of $\mathcal{D}A$ has no repeated factor other than x. Let $C(x) = x^n + c_1 x^{n-1} + c_2 x^{n-2} + \cdots + c_0$. Each coefficient c_i is a sum of $i \times i$ minors of $\mathcal{D}A$ and hence is either homogeneous of degree i in $\delta_1, \ldots, \delta_n$ or is zero. Therefore $C(x)$ is homogeneous of degree n in the indeterminates $\delta_1, \ldots, \delta_n$ and x. Thus the factors of $C(x)$ are homogeneous in these indeterminates, in any factorization of this polynomial. On the other hand, each c_i is at most linear in each indeterminate δ_j, since each $i \times i$ minor of $\mathcal{D}A$ is.

Suppose now that $C(x)$ has a repeated factor $g(x)$, so that $C(x) = f(x)g(x)^2$ for some polynomial $f(x)$. No indeterminate δ_j can occur in $g(x)$ for, otherwise, $g(x)^2$ and $C(x)$ would not be linear in δ_j. Thus the repeated factor $g(x)$ must be homogeneous in $\delta_1, \ldots, \delta_n, x$ and free of $\delta_1, \ldots, \delta_n$, and must be a monomial in x. $\boxtimes$

Theorem 4.2 *Let $\mathbb{F}$ be a field, let A be an $n \times n$ matrix over $\mathbb{F}$, and let S be a finite subset of $\mathbb{F}$. If $D = diag(d_1, \ldots, d_n)$ where $d_1, \ldots, d_n$ are chosen uniformly and independently from S then DA is cyclic up to nilpotent blocks with probability at least $1 - n(n-1)/|S|$.*

Proof: Suppose $|\mathbb{F}| > n(n-1)$ — the result is trivial otherwise. By Lemma 4.1, every invariant factor of $\mathcal{D}A$ except the minimal polynomial $\hat{f}_s$ is a monomial of x, if $\mathcal{D} = diag(\delta_1, \ldots, \delta_n)$ and $\delta_1, \ldots, \delta_n$ are distinct indeterminates over $\mathbb{F}$. Let k be the degree of $\hat{f}_s$. If $\gamma_1, \ldots, \gamma_n$ are distinct indeterminates that are different from $\delta_1, \ldots, \delta_n$ and $\mathcal{Y} = [\gamma_1, \ldots, \gamma_n]^{\mathrm{T}}$, then the vectors

$$\mathcal{Y}, (\mathcal{D}A)\mathcal{Y}, \ldots, (\mathcal{D}A)^{k-1}\mathcal{Y}$$

are linearly independent, so there is a $k \times k$ submatrix of the matrix with these vectors as its columns whose determinant is a nonzero polynomial in $\delta_1, \ldots, \delta_n, \gamma_1, \ldots, \gamma_n$. This polynomial has total degree at most $k \leq n$ in the indeterminates $\gamma_1, \ldots, \gamma_n$. Therefore, if these indeterminates are replaced by uniformly and independently chosen elements of S, so that $\mathcal{Y}$ is replaced by a vector $y \in \mathbb{F}^{n \times 1}$, then this determinant becomes a nonzero polynomial in $\delta_1, \ldots, \delta_n$ with probability at least $1 - n/|S| > 0$, by the Schwartz-Zippel lemma. Fix any such vector y for which the determinant is nonzero; the determinant is now a nonzero polynomial with total degree at most $k(k-1)/2 \leq n(n-1)/2$ in $\delta_1, \ldots, \delta_n$. Thus, if values $d_1, \ldots, d_n$ for $\delta_1, \ldots, \delta_n$ are chosen uniformly and independently from S, then the determinant is a nonzero element of $\mathbb{F}$ with probability at least $1 - n(n-1)/|S|$. In this case, if we set $D = diag(d_1, \ldots, d_n)$ then the vectors $y, (DA)y, \ldots, (DA)^{k-1}y$ are linearly independent, and the invariant factors of DA are $f_1, \ldots, f_{s-1}, f_s$, where $f_1, \ldots, f_{s-1}, \hat{f}_s$ are the invariant factors of $\mathcal{D}A$ and f_s is obtained from $\hat{f}_s$ by replacing the indeterminates $\delta_1, \ldots, \delta_n$ with the values $d_1, \ldots, d_n$, respectively. $\boxtimes$

It follows that if $\mathbb{F}$ is a large field then diagonal scaling is a sufficient conditioner for PRECONDCYC. Choosing S to be a subset of $\mathbb{F} \setminus \{0\}$, one can ensure that DA is nonsingular if A is, so that the minimal polynomial and characteristic polynomial of DA agree if DA is cyclic up to nilpotent blocks.

A conditioner for PRECONDSQUFREE-X is also obtained, unless the characteristic of $\mathbb{F}$ is positive and small:

Theorem 4.3 *Let $\mathbb{F}$ be a field whose characteristic is either zero or greater than n, let A be an $n \times n$ matrix over $\mathbb{F}$, and let S be a finite subset of $\mathbb{F}$. If $D = diag(d_1, \ldots, d_n)$ where $d_1, \ldots, d_n$ are chosen uniformly and independently from S, then the characteristic polynomial of DA is the product of a squarefree polynomial and a power of x with probability at least $1 - (2n^2 - n)/|S|$.*

Proof: Once again, it follows by Lemma 4.1 that if $\mathcal{D} = diag(\delta_1, \ldots, \delta_n)$, where $\delta_1, \ldots, \delta_n$ are distinct indeterminates over $\mathbb{F}$, then the characteristic polynomial of $\mathcal{D}A$ is the product of a squarefree polynomial f such that $f(0) \neq 0$ and a power x^k of x. The coefficients of f are clearly polynomials in $\delta_1, \ldots, \delta_n$, since these are also coefficients of the characteristic polynomial.

Since the degree of f is at most n, f is squarefree, and the characteristic of $\mathbb{F}$ is either zero or greater than n, the discriminant of f with respect to x is a nonzero polynomial in $\delta_1, \ldots, \delta_n$. This polynomial has degree at most $2n - 1$ in each indeterminate δ_i so it follows, once again by the Schwartz-Zippel lemma, that if $d_1, \ldots, d_n$ are chosen uniformly and independently from S, and $D = diag(d_1, \ldots, d_n)$, then the polynomial in $\mathbb{F}[x]$ obtained from f by replacing $\delta_1, \ldots, \delta_n$ with $d_1, \ldots, d_n$, respectively, is squarefree with probability at least $1-(2n^2-n)/|S|$. In this case, the characteristic polynomial of DA is clearly the product of a squarefree polynomial and a power of x. $\boxtimes$

Suppose once again that A is an $n \times n$ matrix over $\mathbb{F}$, and let r be the rank of A. Then there exist an $(n-r) \times n$ matrix L and an $n \times (n-r)$ matrix R, each with full rank $n-r$, such that $LA = 0$ and $AR = 0$.

Lemma 4.4 *Let A, L, and R be as above. If LR is nonsingular then A has no nilpotent blocks (of size greater than one) in its Jordan normal form.*

Proof: Suppose A is a matrix with at least one nilpotent block of size greater than one in its Jordan normal form.

If X is a nonsingular matrix and $A' = X^{-1}AX$, then $L' = XL$ and $R' = X^{-1}R$ are clearly matrices with full rank $n-r$ such that $L'A' = LAX = 0$ and $A'R' = X^{-1}A = 0$. Since $L'R' = LR$, we may assume without loss of generality that A is block diagonal, with a nilpotent Jordan block of size greater than one in its lower right corner. In this case, the vector $v = [0, \ldots, 0, 1]^{\mathrm{T}}$ is a vector such that $Av = 0$ and $u^{\mathrm{T}}v = 0$ for every vector u such that $u^{\mathrm{T}}A = 0$. Thus, v is a nonzero vector in the column space of R, and $Lv = 0$. Therefore, LR is singular. $\boxtimes$

Theorem 4.5 *Let A be a symmetric $n \times n$ matrix over a field $\mathbb{F}$ and let S be a finite subset of $\mathbb{F} \setminus \{0\}$. If $d_1, \ldots, d_n$ are chosen uniformly and independently from S and $D = diag(d_1, \ldots, d_n)$, then the matrices A and DA have the same rank r, and the probability that DA has a nilpotent block of size greater than one is at most $(n-r)/|S| \leq n/|S|$.*

Proof: It is sufficient to prove that the matrix $D^{-1}A$ has no nilpotent blocks of size greater than one with high probability, since the entries of D^{-1} are clearly chosen uniformly and independently from a finite subset $S' = \{s^{-1} : s \in S\}$ with the same size as S.

Let L and R be as above, so that L and R are $(n-r) \times n$ and $n \times (n-r)$ matrices, respectively, of full rank $n-r$, such that $LA = 0$ and $AR = 0$. Since A is symmetric we may assume that $R = L^{\mathrm{T}}$. In this case, $L' = LD$ and $R' = R = L^{\mathrm{T}}$ are matrices of full rank such that $L'(D^{-1}A) = 0$ and $(D^{-1}A)R' = 0$. It is sufficient, by the above lemma, to prove that the $(n-r) \times (n-r)$ matrix $L'R' = LDL^{\mathrm{T}}$ is nonsingular with probability at least $(n-r)/|S|$.

Consider the matrix $L\mathcal{D}L^{\mathrm{T}}$, where as usual $\mathcal{D} = diag(\delta_1, \ldots, \delta_n)$ and $\delta_1, \ldots, \delta_n$ are distinct indeterminates over $\mathbb{F}$. The determinant of this matrix has total degree at most $n-r$ in these indeterminates.

Since L has full rank, it has a nonsingular $(n-r) \times (n-r)$ minor, L'. Set D' to be a diagonal matrix whose i^{th} diagonal entry is one if the i^{th} row of L is included in this minor, and whose i^{th} entry is zero otherwise. Then $D' = (D')^2$, $LD'L^{\mathrm{T}} = L(D')^2L^{\mathrm{T}} = L'(L')^{\mathrm{T}}$, and the determinant of $LD'L^{\mathrm{T}}$ is the square of that of L', which is clearly nonzero. The determinant of LDL^{T} is therefore a nonzero polynomial, and the result follows by the Schwartz-Zippel lemma. ⊠

A diagonal scaling that preserves symmetry will also be useful. Note that if A is symmetric and D is a nonsingular diagonal matrix, then DAD is a symmetric matrix with the same invariant factors (and rational Jordan form) as D^2A. The next result can therefore be established from the previous one.

Theorem 4.6 *Let A be a symmetric $n \times n$ matrix over a field $\mathbb{F}$ and let S be a finite subset of $\mathbb{F} \setminus \{0\}$. If $d_1, \ldots, d_n$ are chosen uniformly and independently from S and $D = diag(d_1, \ldots, d_n)$, then the matrices A and DAD have the same rank, and the probability that DAD has a nilpotent block of size greater than one is at most $2n/|S|$. Furthermore, if the squares of elements of S are distinct (that is, if $s^2 \neq t^2$ whenever $s, t \in S$ and $s \neq t$), then DAD has a nilpotent block of size greater than one with probability at most $n/|S|$.*

A conditioner for PRECONDSQUFREE can also be obtained unless the characteristic of $\mathbb{F}$ is small.

Theorem 4.7 *Let A be a symmetric $n \times n$ matrix over a field $\mathbb{F}$ whose characteristic is zero or greater than n and let S be a finite subset of $\mathbb{F} \setminus \{0\}$. If $d_1, \ldots, d_n$ are chosen uniformly and independently from S and $D = diag(d_1, \ldots, d_n)$, then the matrices A and DAD have the same rank, the minimal polynomial of DAD is squarefree, and the characteristic polynomial is the product of the minimal polynomial and a power of x, with probability at least $1 - 4n^2/|S|$. This probability increases to $1 - 2n^2/|S|$ if the squares of elements of S are distinct.*

Proof: Once again, it should be noted that the matrices DAD and D^2A have the same minimal polynomial.

Consider the first claim. If $|S| \leq 4n^2$ then this is trivial. Otherwise there is a subset S' of S with size greater than $2n^2$ whose squares are distinct, and one can apply Theorem 4.3 to establish the existence of a nonsingular diagonal matrix D such that the characteristic polynomial of D^2A (and DAD) is the product of a squarefree polynomial and a power of x. The argument used to prove Theorem 4.6 can now be applied, with the matrix $\mathcal{D}A\mathcal{D}$ instead of $\mathcal{D}A$, where $\mathcal{D}$ is as above, to conclude that if $d_1, \ldots, d_n$ are chosen uniformly and independently from S then the characteristic polynomial of DAD is *not* the product of a squarefree polynomial f and a power of x, with probability at most $(4n^2 - 2n)/|S|$. On the other hand, Theorem 4.6 implies that DAD has a nilpotent block of size greater than one with probability at most $2n/|S|$. Consequently, the characteristic polynomial is the product of a squarefree polynomial f such that $f(0) \neq 0$, and a power of x, and the minimal polynomial of DAD is either f or xf, with probability at $1 - (4n^2)/|S|$, as needed.

If the squares of elements of S are distinct, then the set S' of squares of elements of S is another subset of $\mathbb{F} \setminus \{0\}$ of the same size, and, since DAD and D^2A have the same minimal polynomial for any nonsingular diagonal matrix D, the likelihood that the minimal polynomial of DAD is squarefree, and that the characteristic polynomial is the product of the minimal polynomial and a power of x, is the same when the entries of D are chosen uniformly and independently from S as the likelihood that these properties hold for DA when the entries of D are chosen uniformly and independently from S'. The second claim therefore follows from Theorems 4.3 and 4.5. ⊠

5 Preconditioners based on Beneš Networks

Preconditioners based on Beneš networks work on the problem of localizing linear independence. The objective is to precondition an $n \times n$ matrix of rank r so that the first r rows of the preconditioned matrix become linearly independent. In this section, we improve on the earlier work presented in [19] and [24] in two ways. First, in section 5.1, instead of using Beneš permutation networks as in [24] we use butterflies as Parker does in [19]. However, unlike Parker, we generalize our networks to arbitrary n and are not limited to powers of 2. Then in section 5.2, we improve on [19] again by using an exchange matrix that saves one multiplication per switch over Parker's.

5.1 Butterfly Networks

Let us consider the n rows of a $n \times n$ matrix. We want to make the first r of these linearly independent. We can use a switching network to exchange rows until the first r are linearly independent. Our goal is to switch any r rows of an arbitrary number n rows to the beginning of the network. However, we must first consider the case of switching any r rows into any contiguous block for $n = 2^l$.

An l-dimensional butterfly network is a recursive network of butterfly switches with 2^l nodes at each level such that at level m the node i is merged with node $i + 2^{m-1}$. Figure 1 illustrates a 3-dimensional butterfly with 8 nodes at each level.

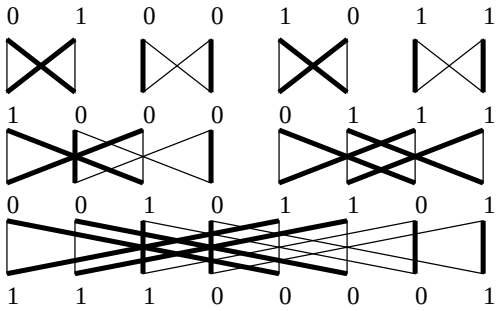


Figure 1: Butterfly network

Lemma 5.1 *Let $n = 2^l$. The l-dimensional butterfly network discussed above can switch any r indices $1 \leq i_1 < \cdots < i_r \leq n$ into any desired contiguous block of indices; wrap around outside, for our purposes, shall preserve contiguity. For example, in figure 1, the ones would be considered contiguous. Furthermore, the network contains a total of $n \log_2(n)/2$ switches.*

Proof: Let us prove this lemma by induction. For $n = 1$ the proof is trivial because no switches are required.

Suppose the lemma is true for $n/2$. Then, let us divide the n nodes in half with r_1 given such that $i_{r_1} \leq n/2 < i_{r_1+1}$. Now, we can construct butterfly networks of dimension $l - 1$ for each of these collections of $n/2$ nodes. By the lemma, each of these subnetworks can arrange the indices $i_1, \ldots, i_{r_1}$ and $i_{r_1+1}, \ldots, i_r$, respectively, to any desired contiguous blocks.

Let us consider the contiguous block desired from the network. It is either contained in the interior of the network in indices $1 \leq j, \ldots, j + r - 1 \leq n$ or it wraps around the outside of the network and can be denoted by indices $1, \ldots, j - 1$ and $n - r + j, \ldots, n$. This second situation can be converted into the first by instead thinking of switching the $n - r$ indices not originally chosen into the contiguous block $j, \ldots, j + n - r - 1$. Thus we only have to consider the first situation. This can then be further divided into the two cases when the contiguous block $j, \ldots, j + r - 1$ is contained within one half and when the block is in both halves and connected in the center.

For the first case, let us assume the desired block is completely within the first half: $j + r - 1 \leq n/2$. Then we can use the first subnetwork to place $i_1, \ldots, i_{r_1}$ so they switch into $j, \ldots, j + r_1 - 1$, and we can use the second subnetwork to position $i_{r_1+1}, \ldots, i_r$ to switch into $j + r_1, \ldots, j + r - 1$ as in figure 2. A symmetric argument holds when the desired contiguous block is contained in the second half: $j > n/2$.

For the case when $j \leq n/2$ and $j + r - 1 > n/2$, let us assume $r_1 \leq n/2 - j + 1$ and thus we need to switch $r_2 = n/2 - j - r_1 + 1$ indices from the second half to the first. Then we can use the first subnetwork to place $i_1, \ldots, i_{r_1}$ so they switch into $j, \ldots, j + r_1 - 1$, and we can use the second subnetwork to position $i_{r_1+1}, \ldots, i_r$ in a contiguous block which wraps around the outside of the subnetwork so they switch into $j + r_1, \ldots, j + r - 1$ as in figure 3. Once again, a symmetric argument holds for $r_1 \geq n/2 - j + 1$.
The switch count is that for an l-dimensional butterfly. ⊠

This means we can switch any r rows of a $n \times n$ matrix into any contiguous block for $n = 2^l$. Now we are ready to consider our original goal of switching any r rows into the first block of r rows for any n. When

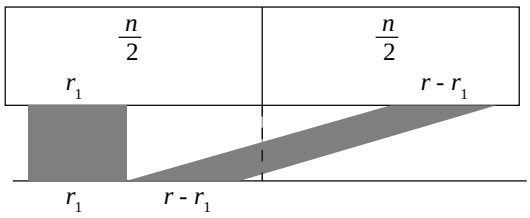


Figure 2: Butterfly network case 1

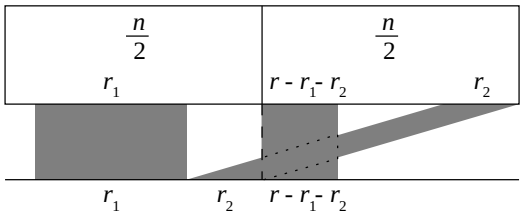


Figure 3: Butterfly network case 2

n is not a power of two, let us decompose n as

$$n = \sum_{i=1}^{k} 2^{l_i} \text{ where } l_1 < l_2 < \cdots < l_p; \text{ let } n_i = 2^{l_i}. \tag{3}$$

First we lay out butterfly networks for each of the n_i blocks. Then we build a generalized butterfly network by connecting these butterfly networks by butterfly switches recursively such that $\sum_{i=1}^{k-1} n_i$ is merged with the far right nodes of n_k as in figure 4,

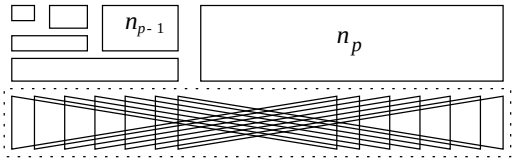


Figure 4: Generalized butterfly network

Theorem 5.2 *The generalized butterfly network discussed above can switch any r indices $1 \le i_1 < \cdots < i_r \le n$ into the contiguous block $1, 2, \ldots, r$. Furthermore, it has a depth of $\lceil \log_2(n) \rceil$ and a total of no more than $n\lceil \log_2(n) \rceil / 2$ butterfly switches.*

Proof: If $n = 2^l$, the proof follows directly from Lemma 5.1 and equality is obtained in the number of switches. Otherwise, from (3) we know $n_k > \sum_{i=1}^{k-1} n_i$. We prove the first part of this theorem by induction. If $k = 1$ the proof is directly from Lemma 5.1. Otherwise, suppose the theorem is true for $\sum_{i=1}^{k-1} n_i$, and let i_{r_1} be the last index in the left half of the network, that is, $i_{r_1} \le \sum_{i=1}^{r-1} n_i < i_{r_1+1}$. Then we can switch the indices $i_1, \ldots, i_{r_1}$ into the contiguous block $1, \ldots, r_1$ using a generalized butterfly network.

If $r \le \sum_{i=1}^{k-1} n_i$, we can use Lemma 5.1 to position the indices $i_{r_1+1}, \ldots, i_r$ so they switch into positions $r_1 + 1, \ldots, r$ as in figure 5. Otherwise let $r_2 = (\sum_{i=1}^{k-1} n_i) - r_1$, and then we can use the same lemma to position the indices as in figure 6.

The total number of butterfly switches is the number of switches for each of the subnetworks plus another $\sum_{i=1}^{p-1} n_i$ switches to combine the two. Another way of counting the switches is the sum of the number of switches for each of the n_i blocks plus the number of switches to connect these blocks:

$$s = \sum_{i=1}^{p} \frac{n_i}{2} l_i + \sum_{i=1}^{p-1} \left(\sum_{j=1}^{i} n_j \right). \tag{4}$$

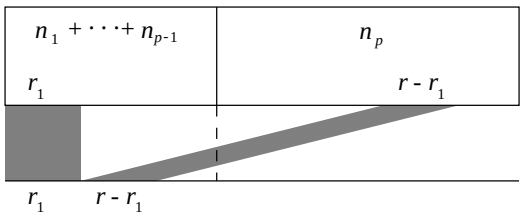


Figure 5: Generalized butterfly network case 1

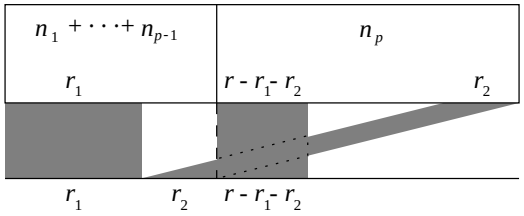


Figure 6: Generalized butterfly network case 2

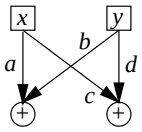
From equation (3) we know $l_i \leq l_p - (p-i)$ for $i < p$ and also $\sum_{j=1}^{i} n_j < n_{i+1}$. Thus, equation (4) gives us

$$s \leq \sum_{i=1}^{p} \frac{n_i}{2} l_p + \sum_{i=2}^{p} \frac{n_i}{2} < \frac{n}{2} l_p + \frac{n}{2} = \frac{n}{2} \lceil \log_2(n) \rceil. \tag{5}$$

Furthermore, the depth of the network is $l_p + 1 = \lceil \log_2(n) \rceil$. $\boxtimes$

5.2 Generic exchange matrices

Wiedemann [24] uses these switching networks (Beneš permutation networks in his case; butterflies were shown to suffice in [19]) for the construction of left (and right) preconditioners in the following manner. Each switch in the network implements a directed acyclic arithmetic circuit:



$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} ax + by \\ cx + dy \end{bmatrix}$$

Here a, b, c, d will be chosen appropriately later. The circuit performs the given 2×2 matrix operation. Here x and y stand for rows (columns) that need to pass through the switch. The 2×2 matrix is embedded in an $n \times n$ matrix in the fashion of an elementary matrix that can exchange row i and row j.

$$\mathcal{E}^{[i,j]}(a,b,c,d) = \begin{bmatrix} \ddots & & & & & & \\ & 1 & & & & & \\ & & a & & & b & \\ & & & 1 & & & \\ & & & & \ddots & & \\ & & & & & 1 & \\ & & c & & & d & \\ & & & & & & 1 \\ & & & & & & & \ddots \end{bmatrix}.$$

Similarly to Wiedemann, one observes that by setting $a = d = 1$ and $b = c = 0$ the circuit passes the rows straight through, while by setting $a = d = 0$ and $b = c = 1$ the circuit exchanges the rows. We consider the

preconditioner

$$\mathcal{L} = \prod_{k=1}^{s} \mathcal{E}_k(\alpha_k, \beta_k, \gamma_k, \delta_k)$$

where $\mathcal{E}_k$ implements the k^{th} switch in the generalized butterfly network of s switches, and where $\alpha_k, \beta_k, \gamma_k, \delta_k$ are symbols. Let A be a fixed $n \times n$ matrix of rank r. Then the first r rows of $\mathcal{L}A$ are linearly independent over $\mathbb{F}(\alpha_1, \ldots, \alpha_s, \beta_1, \ldots, \beta_s, \gamma_1, \ldots, \gamma_s, \delta_1, \ldots, \delta_s)$ because one may evaluate the symbols in such a manner that the generalized butterfly network switches r linearly independent rows to the top. In [17] the exchange matrix is reduced to a single variable, namely $\begin{bmatrix} 1-a & a \\ a & 1-a \end{bmatrix}$. Wiedemann actually gives a unimodular matrix, namely $\begin{bmatrix} 1 & a \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ b & 1 \end{bmatrix} \begin{bmatrix} 1 & c \\ 0 & 1 \end{bmatrix}$, where the row exchange is accomplished by $a = 1$, $b = -1$, and $c = 1$.

The preconditioner matrix L, where the symbols have been evaluated at fixed random values, is used as a black box matrix and the expense for L times a vector needs to be optimized. We will show that for symbolic matrices of the form

$$\hat{\mathcal{E}}(\alpha) = \begin{bmatrix} 1 & \alpha \\ 1 & 1+\alpha \end{bmatrix} \text{ with action } \hat{\mathcal{E}}(a) \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} x + ay \\ y + (x + ay) \end{bmatrix} \tag{6}$$

the first r rows of $(\prod_{k=1}^{s} \hat{\mathcal{E}}_k(\alpha_k))A$ are linearly independent over $\mathbb{F}(\alpha_1, \ldots, \alpha_s)$. By (6), each switch requires 2 additions and 1 multiplication. For contrast, Parker [19] uses an exchange matrix of the form $\begin{bmatrix} a & b \\ a & -b \end{bmatrix}$ which requires 2 additions and 2 multiplications, one more multiplication than $\hat{\mathcal{E}}(a)$.

The proof is by induction on the levels of the generalized butterfly network, where we follow the routing of r linearly independent rows. On each level, these rows have been placed in certain row positions in the matrix. In Figure 7 we depict the route of row x_i through the network. We will set the switches by evaluating the symbols α_k to certain values using the mixing DAG (6). The goal is to show that along the route of the generalized butterfly network that brings the r linearly independent rows to the front, the now arithmetically mixed rows, which originally correspond to the routed r linearly independent rows, remain linearly independent. We simply prove this from one level to the next, and denote by $x_i^{[j]}$ the row in the position of the original row i at level j. The induction hypothesis is that the r rows $x_{i_1}^{[j]}, \ldots, x_{i_r}^{[j]}$ are linearly independent over $\mathbb{F}(\alpha_1, \ldots, \alpha_s)$. In the network, they are placed at certain designated positions (at level j), which we have marked by squares in Figure 7.

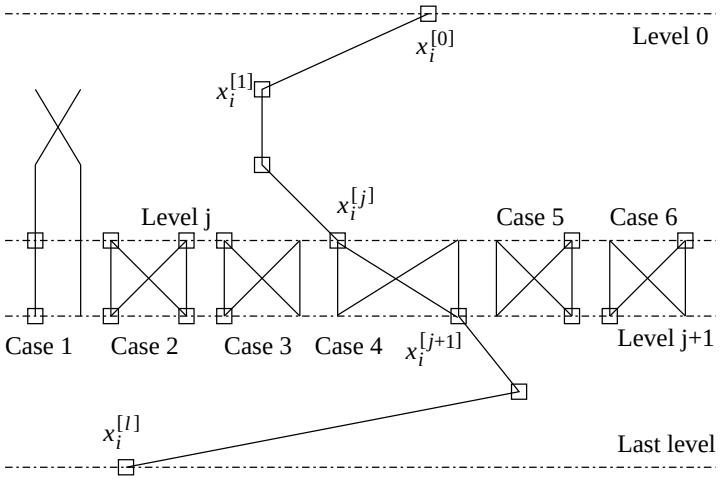


Figure 7: Illustration of proof

Each position at level $j+1$ that holds a designated row has a mixture of the rows above. There are six cases, depicted from left to right in Figure 7. Case 1 is where the row is routed straight through without a switch. This may be done at the bottom of the network if n is not a power of 2. Nothing needs to be done, as the row remains untouched. Case 2 is where the switch mixes two designated rows. This case is surprisingly easy: we set the corresponding symbol $\alpha_k = 0$. By (6) the new rows are x and $x+y$. They span the same two-dimensional subspace and the overall linear independence of the r designated rows remains unaffected. In the remaining four cases, a linearly independent row is mixed with a dependent one. In Cases 3 and 4, the designated row is on the left side of the switch, and in Cases 5 and 6 on the right side. The former is easier: In Case 3 we again set $\alpha_k = 0$ and in Case 4 we set $\alpha_k = -1$ with the effect that the designated row gets routed through the switch unchanged. In both Cases 5 and 6 we retain α_k as a symbol. We now have fresh symbolic weights on these rows on the next level, where they appear in linear the combination $\alpha_k y + x + y$ (Case 5) or $\alpha_k y + x$ (Case 6).

The argument is concluded as follows. Select r columns in the linearly independent rows $x_{i_1}^{[j]}, \ldots, x_{i_r}^{[j]}$ on level j such that the $r \times r$ submatrix formed by the rows and those columns is nonsingular. Now consider the same column selection on level $j+1$. The coefficient of the term $\prod_{k_t} \alpha_{k_t}$ in the corresponding minor, where α_{k_t} are the retained new symbols of the Cases 5 and 6, is the minor (of the submatrix) on level j, hence nonzero. Thus the new designated rows on level $j+1$ are linearly independent over $\mathbb{F}(\alpha_1, \ldots, \alpha_s)$.

Theorem 5.3 *Let $\mathbb{F}$ be a field, let A be an $n \times n$ matrix over $\mathbb{F}$ with r linearly independent rows, let s be the number of butterfly switches in the generalized butterfly network from Theorem 5.2, and let S be a finite subset of $\mathbb{F}$. If $a_1, \ldots, a_s$ are randomly chosen uniformly and independently from S then the first r rows of*

$$\left(\prod_{k=1}^{s} \hat{\mathcal{E}}_k(a_k) \right) A$$

are linearly independent with probability no less than

$$1 - \frac{r\lceil \log_2(n) \rceil}{|S|} \geq 1 - \frac{n\lceil \log_2(n) \rceil}{|S|}.$$

Proof: The matrix A is over the field $\mathbb{F}$, so each row of A is a row vector of polynomials in $\alpha_1, \ldots, \alpha_s$ of degree zero. Each switch in the generalized butterfly network increases the degree of the polynomials by one, and the depth of the network is $\lceil \log_2(n) \rceil$. So, the rows of $(\prod_{k=1}^{s} \hat{\mathcal{E}}_k(\alpha_k))A$ are vectors of polynomials in $\alpha_1, \ldots, \alpha_s$ of degree $\lceil \log_2(n) \rceil$. Thus, the determinant of an $r \times r$ submatrix of this preconditioned matrix is a polynomial of degree $r\lceil \log_2(n) \rceil$.

Given that A has r linearly independent rows, we can designate these rows to be switched by the generalized butterfly network of Theorem 5.2 to the first r rows of the preconditioned matrix $(\prod_{k=1}^{s} \hat{\mathcal{E}}_k(\alpha_k))A$. The argument above shows at every level in the network the r designated rows remain linearly independent over $\mathbb{F}(\alpha_1, \ldots, \alpha_s)$. In particular, the designated rows in the last level, namely the first r rows of the preconditioned matrix are linearly independent over $\mathbb{F}(\alpha_1, \ldots, \alpha_s)$. This means, there is an $r \times r$ submatrix of the first r rows of $(\prod_{k=1}^{s} \hat{\mathcal{E}}_k(\alpha_k))A$ whose determinant is not identically zero. Because this is a polynomial of degree $r\lceil \log_2(n) \rceil$, the Schwartz/Zippel lemma tells us that $(a_1, \ldots, a_s)$ is a root of it with probability no greater than $r\lceil \log_2(n) \rceil / |S|$. With probability no less than $1 - r\lceil \log_2(n) \rceil / |S|$, it is not a root of the polynomial, and thus we have an $r \times r$ submatrix of the first r rows of $(\prod_{k=1}^{s} \hat{\mathcal{E}}_k(a_k))A$ whose determinant is not zero. Therefore, the first r rows of $(\prod_{k=1}^{s} \hat{\mathcal{E}}_k(a_k))A$ are linearly independent with probability no less than

$$1 - \frac{r\lceil \log_2(n) \rceil}{|S|} \geq 1 - \frac{n\lceil \log_2(n) \rceil}{|S|}. \quad \boxtimes$$

6 Sparse Matrix Preconditioners

For matrices over fields $\mathbb{F}$ with a small number of elements compared to the matrix dimension n or to n^2, the preconditioners of sections 4 and 5 may not be usable directly. Their proofs — based on the Schwartz-Zippel lemma — require a field extension with logarithmic degree over $\mathbb{F}$. An extra $O(\log n)$ factor may be involved

in the costs of the resulting algorithms. We show here that a special probability distribution on sparse matrices with entries in $\mathbb{F}$, proposed in [24], also provides preconditioners for p-PRECONDNIL. This avoids the need of field extensions, for instance to solve LINSOLVE1 using the algorithm in [23], and may be useful for practical implementations.

In the following, for given parameters $w_{i,j} \in [0,1]$, $1 \leq i,j \leq n$, the preconditioner distributions are defined by a random $n \times n$ matrix whose entry (i,j) is a uniform randomly chosen nonzero element of $\mathbb{F}$ (or of a subset of $\mathbb{F}$) with probability $w_{i,j}$ and zero otherwise. For $q = |\mathbb{F}|$ and $w_{i,j} = w = 1 - 1/q$ it is well known that such matrices are invertible with probability

$$\tau_q(n) = (1-1/q)(1-1/q^2)\ldots(1-1/q^n) \geq \sqrt{2}/5 > 1/4 \tag{7}$$

(the bound $\sqrt{2}/5$ is proven in [4]). For $w_{i,j} = w$, the expected rank considered as a function of w decreases monotonically in the range $1-1/q \geq w \geq 0$, its value is $n - O(1)$ for $w_{i,j} = \log(n)/n$ [1]. To get PRECONDIND scalings with $w_{i,j}$ a function w_j of j only, remark 3.2 thus indicates that w_j has to be greater than $(\log j)/j$.

Definition 6.1 [24] *For any given subset S of $\mathbb{F}$ with $\sigma \geq 2$ elements and containing zero and for $\kappa \geq 1$, the distribution defined by*

$$w_{i,j} = w_j = \min\{1 - 1/\sigma, \quad \kappa(\log n)/j\}$$

is called the Wiedemann distribution.

Wiedemann has shown that his distribution gives PRECONDIND p-preconditioners for $S = \mathbb{F}$ [24, Theorem 1]. Actually it also satisfies the additional assumption (1) of theorem 3.5:

Proposition 6.2 *Let $A \in \mathbb{F}^{*\times n}$ be of rank r and let Q be in $\mathbb{F}^{(n-r)\times r}$. Let W be chosen from the Wiedemann distribution. If $W^{(r)}$ and $\overline{W}$ respectively denote the first r and the last $n-r$ columns of W then W satisfies (1):*

$$\text{rank } A(W^{(r)} + \overline{W}Q) = \text{rank } A$$

with probability at least

$$(1 - 1/n^\kappa)^r \cdot \prod_{j=1}^{r}(1 - 1/\sigma^j). \tag{8}$$

Proof: We follow the arguments in [24, pp. 56-57] and detail only what is needed to show the additional property (1). The property is satisfied if and only if $W^{(r)} + \overline{W}Q$ together with the right nullspace of A generates all of $\mathbb{F}^n$. Since the entries of W are independent it is sufficient to prove that the columns of $W^{(r)} + C$ for any $n \times r$ matrix C together with any given subspace V_{n-r} of dimension $n-r$ generates all of $\mathbb{F}^n$ with the announced probability.

Let V_k be a subspace of dimension k. For a given vector c let $a[i]$ be the number of vectors u having i nonzero entries in S and such that $u + c \in V_k$. With no loss of generality, the set of restrictions of vectors in V_k to the k first coordinates is of dimension k. Two different vectors $u_1 + c$ and $u_2 + c$ in V_k have different restrictions to these coordinates and the same is true for the restrictions of u_1 and u_2. Each restriction is a vector of length k with i nonzero coordinates chosen between $\sigma - 1$ values thus:

$$\textstyle\sum_{i=0}^{j} a[i] \leq \sum_{i=0}^{j} \binom{k}{i} (\sigma-1)^i.$$

This coincides with the bound used in [24] for the number of vectors u, with at most j nonzero entries, which belong to a given V_k. For any given C, the probability that the j-th column $W_j + C_j$ of $W^{(r)} + C$ lies in a given subspace of dimension $n-j$ is thus less than $(1-w_j)^j$ [24, p. 56]. The probability that it does not belong to the subspace generated by V_{n-r} and the columns $W_l + C_l$, $r \geq l \geq j+1$, is thus greater than $1 - (1-w_j)^j$. By doing the product, the probability that W satisfies (1) is thus at least

$$\prod_{j=1}^{r}(1 - (1-w_j)^j).$$

Let $J = \kappa(\log n)(\sigma-1)/\sigma$. For $1 \le j \le J$, $(1-w_j)^j = 1/\sigma^j$. Otherwise, $(1-w_j)^j = (1-\kappa(\log n)/j)^j \le \exp(-\kappa \log n) \le 1/n^\kappa$. The probability that W is good is thus at least

$$(1-1/n^\kappa)^{r-\min\{J,r\}} \cdot \prod_{j=1}^{\min\{J,r\}} (1-1/\sigma^j),$$

this gives the announced bound. ⊠

Let us notice that for $\kappa \ge 2$ and large n, bound (8) will be very close to bound (7) with $q = \sigma$. The expected number of nonzero entries in W is less than $n \sum_j w_j$ which is less than $\kappa n(\log n)(1+\log n)$.

Corollary 6.3 *For any $A \in \mathbb{F}^{n\times n}$, matrices R and L chosen from the Wiedemann distribution and the transposed one give scaling preconditioners for the generic nilpotency problem (*PreCondNil*) ($A' = LAR$ or $A' = ARL$ as in section 3.4) - each with at most $2n(\log n)(1+\log n) + hn$ nonzero entries – with probability at least $(1-2/n)\tau_\sigma^2(n) - 1/(2h^2)$. The probability is thus bounded from below by a constant even for $\{0,1\}$-preconditioners.*

Proof: Theorem 3.5 and proposition 6.2 with $\kappa = 2$ give the first term of the probability bound. Following [24, Theorem 1], the variance of the total number of nonzero entries in both preconditioners is $2n^2/4$. Therefore, by Chebyshev's inequality, the probability that the expected number of nonzero entries is exceeded by hn is less than $2n^2/(4h^2n^2) = 1/(2h^2)$. ⊠

If the rank r of the matrix A to precondition is known, then preconditioners over any field with an expected number of nonzero entries $O(n \log n)$ instead of $O(n(\log n)^2)$ may be constructed using the distribution in [24, Theorem 1']. It may be possible to show that it also satisfies (1).

References

Note: many of the authors' publications are accessible through links in their Internet homepages listed under the title.

[1] Blömer, J., Karp, R., and Welzl, E. The rank of sparse random matrices over finite fields. *Random Structures and Algorithms 10* (1997), 407–419.

[2] Coppersmith, D. Solving homogeneous linear equations over GF(2) via block Wiedemann algorithm. *Math. Comput. 62*, 205 (1994), 333–350.

[3] DeMillo, R. A., and Lipton, R. J. A probabilistic remark on algebraic program testing. *Information Process. Letters 7*, 4 (1978), 193–195.

[4] Eberly, W. Processor-efficient parallel matrix inversion over abstract fields: two extensions. In *Proc. Second Internat. Symp. Parallel Symbolic Comput. PASCO '97* (New York, N. Y., 1997), M. Hitz and E. Kaltofen, Eds., ACM Press, pp. 38–45.

[5] Eberly, W. Avoidance of look-ahead in Lanczos by random projections, 2000. Manuscript in preparation.

[6] Eberly, W., and Kaltofen, E. On randomized Lanczos algorithms. In Küchlin [16], pp. 176–183.

[7] Giesbrecht, M. Fast computation of the Smith normal form of a sparse integer matrix. In *Algorithms in Number Theory Symposium* (1996), LNCS 1122, pp. 173–186.

[8] Giesbrecht, M. Efficient parallel solution of sparse systems of linear diophantine equations. In *Second International Symposium on Parallel Symbolic Computation (PASCO'97), Maui, Hawaii, USA* (Jul 1997), pp. 1–10.

[9] Giesbrecht, M., Lobo, A., and Saunders, B. D. Certifying inconsistency of sparse linear systems. In *ISSAC 98 Proc. 1998 Internat. Symp. Symbolic Algebraic Comput.* (New York, N. Y., 1998), O. Gloor, Ed., ACM Press, pp. 113–119.

[10] Kaltofen, E. Analysis of Coppersmith's block Wiedemann algorithm for the parallel solution of sparse linear systems. *Math. Comput. 64*, 210 (1995), 777–806.

[11] Kaltofen, E. Challenges of symbolic computation my favorite open problems. *J. Symbolic Comput. 29*, 6 (2000), 891–919. With an additional open problem by R. M. Corless and D. J. Jeffrey.

[12] Kaltofen, E., and Lobo, A. On rank properties of Toeplitz matrices over finite fields. In *ISSAC 96 Proc. 1996 Internat. Symp. Symbolic Algebraic Comput.* (New York, N. Y., 1996), Lakshman Y. N., Ed., ACM Press, pp. 241–249.

[13] Kaltofen, E., and Pan, V. Processor-efficient parallel solution of linear systems II: the positive characteristic and singular cases. In *Proc. 33rd Annual Symp. Foundations of Comp. Sci.* (Los Alamitos, California, 1992), IEEE Computer Society Press, pp. 714–723.

[14] Kaltofen, E., and Saunders, B. D. On Wiedemann's method of solving sparse linear systems. In *Proc. AAECC-9* (Heidelberg, Germany, 1991), H. F. Mattson, T. Mora, and T. R. N. Rao, Eds., vol. 539 of *Lect. Notes Comput. Sci.*, Springer Verlag, pp. 29–38.

[15] Kaltofen, E., and Trager, B. Computing with polynomials given by black boxes for their evaluations: Greatest common divisors, factorization, separation of numerators and denominators. *J. Symbolic Comput. 9*, 3 (1990), 301–320.

[16] Küchlin, W., Ed. *ISSAC 97 Proc. 1997 Internat. Symp. Symbolic Algebraic Comput.* (New York, N. Y., 1997), ACM Press.

[17] Lobo, A. *Matrix-Free Linear System Solving and Applications to Symbolic Computation.* PhD thesis, Rensselaer Polytechnic Instit., Troy, New York, Dec. 1995.

[18] Mulders, T., and Storjohann, A. Diophantine linear system solving. In *International Symposium on Symbolic and Algebraic Computation, Vancouver, BC, Canada* (Jul 1999), pp. 181–188.

[19] Parker, D. S. Random butterfly transformations with applications in computational linear algebra. Technical Report CSD-950023, UCLA, Computer Science Dept., 1995. See `http://www.cs.ucla.edu/~stott/ge/`.

[20] Saunders, B. D., Storjohann, A., and Villard, G. Rank certificates for sparse matrices, June 1998. Reported to the LinBox group: to be published.

[21] Schwartz, J. Fast probabilistic algorithms for verification of polynomial identities. *J. ACM 27* (1980), 701–717.

[22] Villard, G. Further analysis of Coppersmith's block Wiedemann algorithm for the solution of sparse linear systems. In Küchlin [16], pp. 32–39.

[23] Villard, G. Block solution of sparse linear systems over GF(q): the singular case. *SIGSAM Bulletin 32*, 4 (1998), 10–12.

[24] Wiedemann, D. Solving sparse linear equations over finite fields. *IEEE Trans. Inf. Theory* IT-*32* (1986), 54–62.

[25] Zippel, R. Probabilistic algorithms for sparse polynomials. In *Proc. EUROSAM '79* (Heidelberg, Germany, 1979), vol. 72 of *Lect. Notes Comput. Sci.*, Springer Verlag, pp. 216–226.