

Common Factors in Fraction-Free Matrix Decompositions

Johannes Middeke

Research Institute for Symbolic Computation, Johannes Kepler University, Altenberger Straße 69, A-4040 Linz, Austria

David J. Jeffrey

Department of Applied Mathematics, University of Western Ontario, Middlesex College, Room 255, 1151 Richmond Street North, London, Ontario, Canada, N6A 5B7

Christoph Koutschan

Johann Radon Institute for Computational and Applied Mathematics, Austrian Academy of Sciences, Altenberger Straße 69, A-4040 Linz, Austria

Abstract

We consider LU and QR matrix decompositions using exact computations. We show that fraction free Gauß–Bareiss reduction leads to triangular matrices having a non-trivial number of common row factors. We identify two types of common factors: systematic and statistical. Systematic factors depend on the process, while statistical factors depend on the specific data. We show that existing fraction-free QR (Gram–Schmidt) algorithms create a common factor in the last column of Q . This factor always occurs and is thus a systematic factor. We relate the existence of row factors in the LU decomposition to factors appearing in the Smith–Jacobson normal form of the matrix. For statistical factors, we identify some of the mechanisms that create them and give estimates of the frequency of their occurrence. Our conclusions are tested experimentally.

Keywords: fraction-free algorithms, Gaussian elimination, exact linear system solving
2010 MSC: 00-01, 99-00

1. Introduction

Although known earlier, fraction-free methods for exact matrix computations became popular after Bareiss’s study of Gaussian elimination (Bareiss, 1968). Extensions to related topics, such as LU factoring, were considered in Lee and Saunders (1995); Nakos et al. (1997); Zhou and Jeffrey (2008). Gram–Schmidt orthogonalization and QR factoring were studied by Erlingsson et al. (1996), under the more descriptive name of “exact division”. Recent studies have looked at extending fraction-free LU factoring to non-invertible matrices (Jeffrey, 2010) and rank profiling (Dumas et al., 2015), and more generally to areas such as the Euclidean algorithm, and the

Email addresses: jmiddeke@risc.jku.at (Johannes Middeke), djeffrey@uwo.ca (David J. Jeffrey), christoph.koutschan@ricam.oeaw.ac.at (Christoph Koutschan)

Berlekamp–Massey algorithm (Kaltofen and Yuhasz, 2013). We consider matrices over an integral domain \mathbb{D} . For the purposes of giving illustrative examples and conducting computational experiments, matrices over \mathbb{Z} and $\mathbb{Q}[x]$ are used, because these domains are well established and familiar to readers. We emphasize, however, that the methods here apply for all integral domains, as opposed to methods that target specific domains, such as Giesbrecht and Storjohann (2002); Pauderis and Storjohann (2013).

The starting point for this paper is a fraction-free form for LU decomposition (Jeffrey, 2010): given a matrix A over an integral domain \mathbb{D} ,

$$A = P_r L D^{-1} U P_c,$$

where L and U are lower and upper triangular, and D is diagonal, and where the entries of L , D and U are from \mathbb{D} . The permutation matrices P_r and P_c ensure that the decomposition is always a full-rank decomposition, even if A is rectangular or rank deficient; see section 2. The decomposition is achieved by a variant of Bareiss’s algorithm (Bareiss, 1968), which can be seen as a fraction-free approach to Gaussian elimination. This is analogous to how the usual LU decomposition is achieved with Doolittle’s algorithm or Crout’s algorithm. We show in section 7 that the $LD^{-1}U$ decomposition can cover a fraction-free version of the QR decomposition as well.

The key feature of Bareiss’s algorithm is that it predicts certain common factors in rows and removes them immediately by an exact division. However, it was surprising for us to see that when computing concrete examples we still find a considerable number of common factors in the rows of the output matrix U . We find that the same holds true for the QR decomposition, as computed by the algorithm from Zhou and Jeffrey (2008). We will refer to common factors which appear owing to the decomposition algorithm being used as “systematic factors”. Note that these factors will appear in the final result even if the input matrix A does not have any common factors. On the other hand, “statistical factors” are those which are brought into the process by the input matrix. In contrast to systematic factors, statistical factors do not always arise, and their existence will depend on the particular inputs. In this paper we discuss the origins of both kinds of common factors and show that we can predict a significant proportion of them from simple considerations. It is clear that if the elements in a column of L or a row of U possess a common factor, then that factor can be removed, reducing the size of the matrix elements. This leads to a modified $LD^{-1}U$ decomposition. As there is some choice in how these factors should be treated (see Remark 5) we will introduce a more general notion of a *fraction-free LU decomposition* which will include the $LD^{-1}U$ decomposition of Jeffrey (2010) and also our decompositions with the common factors being removed.

After recalling Bareiss’s algorithm, the $LD^{-1}U$ decomposition and the algorithm from Jeffrey (2010) in section 2, we establish, in section 3, a relation between the “systematic” common row factors of U and the entries in the Smith normal form of the same input matrix A . In section 4 we propose an efficient way of identifying a considerable number of “systematic” common row factors introduced by Bareiss’s algorithm; these factors can then be easily removed by exact division. In section 5 we present a detailed analysis concerning the expected number of “statistical” common factors in the special case $\mathbb{D} = \mathbb{Z}$, and we find perfect agreement with our experimental results. In section 6 we discuss the applicability of the previous results to the solving of (possibly rank-deficient) linear systems, namely to obtain a fast method for checking compatibility conditions and for the construction of the solution; this is most useful when a given linear system has to be solved repeatedly for many different right-hand sides.

In [section 7](#) we investigate QR factoring. In this context, the orthonormal Q matrix used in floating point calculations is replaced by a Θ matrix, which is left-orthogonal, i.e. $\Theta^t\Theta$ is diagonal, but $\Theta\Theta^t$ is not. We show that, for a square matrix A , the last column of Θ , as calculated by existing algorithms, is subject to an exact division by the determinant of A , with a significant reduction in size.

Throughout the paper, we employ the following notation. We assume, unless otherwise stated, that the ring \mathbb{D} is an arbitrary integral domain. We denote the set of all m -by- n matrices over \mathbb{D} by $\mathbb{D}^{m \times n}$. We write $\mathbf{1}_n$ for the n -by- n identity matrix and $\mathbf{0}_{m \times n}$ for the m -by- n zero matrix. We shall usually omit the subscripts if no confusion is possible. For $A \in \mathbb{D}^{m \times n}$ and $1 \leq i \leq m$, $A_{i,*}$ is the i^{th} row of A . Similarly, $A_{*,j}$ is the j^{th} column of A for $1 \leq j \leq n$. If $1 \leq i_1 < i_2 \leq m$ and $1 \leq j_1 < j_2 \leq n$, we use $A_{i_1 \dots i_2, j_1 \dots j_2}$ to refer to the submatrix of A made up from the entries of the rows i_1 to i_2 and the columns j_1 to j_2 . Given elements $a_1, \dots, a_n \in \mathbb{D}$, with $\text{diag}(a_1, \dots, a_n)$ we refer to the diagonal matrix that has a_j as the entry at position (j, j) for $1 \leq j \leq n$. We will use the same notation for block diagonal matrices.

We denote the set of all column vectors of length m with entries in \mathbb{D} by \mathbb{D}^m and that of all row vectors of length n by $\mathbb{D}^{1 \times n}$. If \mathbb{D} is a unique factorisation domain and $v = (v_1, \dots, v_n) \in \mathbb{D}^{1 \times n}$, then we set $\text{gcd}(v) = \text{gcd}(v_1, \dots, v_n)$. Moreover, with $d \in \mathbb{D}$ we write $d \mid v$ if $d \mid v_1 \wedge \dots \wedge d \mid v_n$ (or, equivalently, if $d \mid \text{gcd}(v)$). We also use the same notation for column vectors.

We will sometimes write column vectors $w \in \mathbb{D}^m$ with an underline \underline{w} and row vectors $v \in \mathbb{D}^{1 \times n}$ with an overline \overline{v} if we want to emphasize the specific type of vector.

2. Bareiss's Algorithm and the $LD^{-1}U$ Decomposition

For the convenience of the reader, we start by recalling Bareiss's algorithm from [Bareiss \(1968\)](#) as well as the $LD^{-1}U$ decomposition from [Jeffrey \(2010\)](#).

Let \mathbb{D} be an integral domain, and let $A \in \mathbb{D}^{m \times n}$ be a matrix and $b \in \mathbb{D}^m$ be a vector. We are interested in computing the exact solution(s) for the system $Ax = b$ over the quotient field of \mathbb{D} . Following standard methods from linear algebra, we compute a (variation of the) LU decomposition for A . Suppose that

$$A = \begin{pmatrix} a & \dots \\ b & \dots \\ \vdots & \ddots \end{pmatrix},$$

where we assume that a and b are not zero. Choosing a as our pivot in the usual Gaussian elimination, we would subtract b/a times the first row from the second as seen in, for example, [Geddes et al. \(1992, Section 9.2\)](#). The division, however, forces us to work in the quotient field of \mathbb{D} , which is computationally expensive. One way to avoid this is to use cross-multiplication (see [Geddes et al. \(1992, Section 9.2\)](#) where it is called division-free elimination): In order to eliminate b from the second row, first multiply the second row by a and then subtract b times the first row. While this avoids dealing with fractions, when repeating this process for the entire elimination the matrix entries grow in size exponentially with the number of rows which will also need more computation time (and take up more memory). If \mathbb{D} is a unique factorisation domain, it is possible to keep the sizes of the entries as small as possible by dividing each row by its greatest common divisor after the cross-multiplication. the drawback is that computing these greatest common divisors in all steps is again computationally expensive.

Bareiss (1968) observed that after carrying out cross-multiplication for two iterations (that is, eliminating the entries from the first column except the first and eliminating the entries from the second column except the first two) every entry in the third row and below is divisible by a . More generally, if we denote the pivots during the elimination by p_1, \dots, p_r (where r is the rank of A), then after $k \geq 2$ iterations every entry below the k^{th} row is divisible by p_{k-1} . (In actual implementations one often additionally defines $p_{-1} = 1$ in order to prevent $k = 1$ from being a special case.) Bareiss's observation gives us the possibility of removing the factor p_{k-1} after the k^{th} iteration from the lower rows of the matrix by using (relatively) cheap exact division while at the same time avoiding the cost of computing the greatest common divisors. This makes Bareiss's algorithm a valuable choice for exact matrix computations (see, for example, Geddes et al. (1992, Section 9.3) where it is called the single-step fraction-free elimination scheme). We will show below that we can extract even more factors at a moderate additional cost.

In Jeffrey (2010) the idea of Bareiss's algorithm was used in order to obtain a fraction-free variant of the LU factorization. We repeat the main result from that paper here.

Theorem 1 (Jeffrey (2010, Thm. 2)). *A rectangular matrix A with elements from an integral domain \mathbb{D} , having dimensions $m \times n$ and rank r , may be factored into matrices containing only elements from \mathbb{D} in the form*

$$A = P_r L D^{-1} U P_c = P_r \begin{pmatrix} \mathcal{L} \\ \mathcal{M} \end{pmatrix} D^{-1} \begin{pmatrix} \mathcal{U} & \mathcal{V} \end{pmatrix} P_c$$

where the permutation matrix P_r is $m \times m$; the permutation matrix P_c is $n \times n$; \mathcal{L} is $r \times r$, lower triangular and invertible:

$$\mathcal{L} = \begin{pmatrix} p_1 & 0 & \cdots & 0 \\ \ell_{21} & p_2 & \ddots & \vdots \\ \vdots & \vdots & \ddots & 0 \\ \ell_{r1} & \ell_{r2} & \cdots & p_r \end{pmatrix}$$

where the $p_i \neq 0$ are the pivots in a Gaussian elimination; \mathcal{M} is $(m-r) \times r$ and could be null; D is $r \times r$ and diagonal:

$$D = \text{diag}(p_1, p_1 p_2, p_2 p_3, \dots, p_{r-2} p_{r-1}, p_{r-1} p_r);$$

\mathcal{U} is $r \times r$ and upper triangular, while \mathcal{V} is $r \times (n-r)$ and could be null:

$$\mathcal{U} = \begin{pmatrix} p_1 & u_{12} & \cdots & u_{1r} \\ 0 & p_2 & \cdots & u_{2r} \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & p_r \end{pmatrix}.$$

Inspecting the proof given in Jeffrey (2010), it is possible to extract an algorithm for the computation of the $LD^{-1}U$ decomposition. Note that the algorithm can be seen as a variant of Bareiss's algorithm (Bareiss, 1968) which will yield the same U . The difference is that Jeffrey (2010) also explains how to obtain L and D in a fraction-free way.

Algorithm 2. ($LD^{-1}U$ decomposition)

Input: A matrix $A \in \mathbb{D}^{m \times n}$.

Output: The $LD^{-1}U$ decomposition of A as in Theorem 1.

1. Initialize $p_{-1} = 1$, $P_r = \mathbf{1}_m$, $L = \mathbf{0}_{m \times m}$, $U = A$ and $P_c = \mathbf{1}_n$.
2. For each $k = 1, \dots, \min\{m, n\}$:
 - (a) Find a non-zero pivot p_k in $U_{k..m, k..n}$ and bring it to position (k, k) recording the row and column swaps in P_r and P_c . Also apply the row swaps to L accordingly. If no pivot is found, then set $r = k$ and exit the loop.
 - (b) Set $L_{k,k} = p_k$ and $L_{i,k} = U_{i,k}$ for $i = k + 1, \dots, m$. Then eliminate the entries in the k^{th} column and below the k^{th} row in U by cross-multiplication; that is, for $i > k$ set $U_{i,*}$ to $p_k U_{i,*} - U_{ik} U_{k,*}$.
 - (c) Perform division by p_{k-1} on the rows beneath the k^{th} in U ; that is, for $i > k$ set $U_{i,*}$ to $U_{i,*} / p_{k-1}$. Note that the divisions will be exact.
3. If r is not set yet, set $r = \min\{m, n\}$.
4. If $r < m$, then trim the last $m - r$ columns from L as well as the last $m - r$ rows from U .
5. Set $D = \text{diag}(p_1 p_2, \dots, p_{r-1} p_r)$.
6. Return P_r , L , D , U , and P_c .

The algorithm does not specify the choice of pivot in step 2a. Conventional wisdom (see, for example, [Geddes et al. \(1992\)](#)) is that in exact algorithms choosing the smallest possible pivot (measured in a way suitable for \mathbb{D}) will lead to the smallest output sizes. We have been able to confirm this experimentally in [Middeke and Jeffrey \(2014\)](#) for $\mathbb{D} = \mathbb{Z}$ where size was measured as the absolute value. In step 2c the divisions are guaranteed to be exact. Thus, an implementation can use more efficient procedures for this step if available (for example, for big integers using `mpz_divexact` in the GMP library which is based on [Jebelean \(1993\)](#) instead of regular division).

One of the goals of the present paper is to discuss improvements to the decomposition explained above. Throughout this paper we shall use the term $LD^{-1}U$ decomposition to mean exactly the decomposition from Theorem 1 as computed by Algorithm 2. For the variations of this decomposition we introduce the following term:

Definition 3 (Fraction-Free LU Decomposition). For a matrix $A \in \mathbb{D}^{m \times n}$ of rank r we say that $A = P_r L D^{-1} U P_c$ is a *fraction-free LU decomposition* if $P_r \in \mathbb{D}^{m \times m}$ and $P_c \in \mathbb{D}^{n \times n}$ are permutation matrices, $L \in \mathbb{D}^{m \times r}$ has $L_{ij} = 0$ for $j > i$ and $L_{ii} \neq 0$ for all i , $U \in \mathbb{D}^{r \times n}$ has $U_{ij} = 0$ for $i > j$ and $U_{ii} \neq 0$ for all i , and $D \in \mathbb{D}^{r \times r}$ is a diagonal matrix.

We will usually refer to matrices $L \in \mathbb{D}^{m \times r}$ with $L_{ij} = 0$ for $j > i$ and $L_{ii} \neq 0$ for all i as *lower triangular* and to matrices $U \in \mathbb{D}^{r \times n}$ with $U_{ij} = 0$ for $i > j$ and $U_{ii} \neq 0$ for all i as *upper triangular* even if they are not square.

As mentioned in the introduction, Algorithm 2 does result in common factors in the rows of the output U and the columns of L . In the following sections, we will explore methods to explain and predict those factors. The next result asserts that we can cancel all common factors which we find from the final output. This yields a fraction-free LU decomposition of A where the size of the entries of U (and L) are smaller than in the $LD^{-1}U$ decomposition.

Corollary 4. Given a matrix $A \in \mathbb{D}^{m \times n}$ with rank r and its decomposition $A = P_w L D^{-1} U P_c$, if $D_U = \text{diag}(d_1, \dots, d_r)$ is a diagonal matrix with $d_k \mid \gcd(U_{k,*})$, then setting $\hat{U} = D_U^{-1} U$ and $\hat{D} = D D_U^{-1}$ where both matrices are fraction-free we have the decomposition $A = P_w L \hat{D}^{-1} \hat{U} P_c$.

PROOF. By Theorem 1, the diagonal entries of U are the pivots chosen during the decomposition and they also divide the diagonal entries of D . Thus, any common divisor of $U_{k,*}$ will also divide D_{kk} and therefore both \hat{U} and \hat{D} are fraction-free. We can easily check that $A = P_w L D^{-1} D_U D_U^{-1} U = P_w L \hat{D}^{-1} \hat{U} P_c$. \square

Remark 5. If we predict common column factors of L we can cancel them in the same way. However, if we have already canceled factors from U , then there is no guarantee that $d \mid L_{*,k}$ implies $d \mid \hat{D}_{kk}$. Thus, in general we can only cancel $\gcd(d, \hat{D}_{kk})$ from $L_{*,k}$. The same holds *mutatis mutandis* if we cancel the factors from L first.

It will be an interesting discussion for future research whether it is better to cancel as many factors as possible from U or to cancel them from L .

3. LU and the Smith–Jacobson Normal Form

This section explains a connection between “systematic factors” (that is, common factors which appear in the decomposition due to the algorithm being used) and the Smith–Jacobson normal form. Given a matrix A over a principal ideal domain \mathbb{D} , we consider the $LD^{-1}U$ decomposition $A = LD^{-1}U$. Note that here we require an $LD^{-1}U$ decomposition in which the permutation matrices P_r and P_c are trivial. That is, we do not allow arbitrary pivoting. This is justified because in exact linear algebra, pivoting is not as important as in numerical linear algebra. It is only needed for avoiding exact, symbolic zeros. Hence our assumption of no pivoting is not restrictive.

Theorem 6. Let the matrix $A \in \mathbb{D}^{n \times n}$ have the Smith–Jacobson normal form $S = \text{diag}(d_1, \dots, d_n)$ where $d_1, \dots, d_n \in \mathbb{D}$. Moreover, let $A = LD^{-1}U$ be an $LD^{-1}U$ decomposition of A without permutations. Then for $k = 1, \dots, n$

$$d_k^* = \prod_{j=1}^k d_j \mid U_{k,*} \quad \text{and} \quad d_k^* \mid L_{*,k}.$$

Remark 7. The values d_1^*, \dots, d_n^* are known in the literature as the *determinantal divisors* of A .

PROOF. According to (Newman, 1972, II.15), the diagonal entries of the Smith form are quotients of the determinantal divisors, i. e., $d_1^* = d_1$ and $d_k = d_k^* / d_{k-1}^*$ for $k = 2, \dots, n$. Moreover, d_k^* is the greatest common divisor of all k -by- k minors of A for each $k = 1, \dots, n$. Thus, we only have to prove that the entries of the k^{th} row of U are k -by- k minors of A . However, this follows from (Geddes et al., 1992, Eqns (9.8), (9.12)), since the k^{th} row of U consists of the elements

$$U_{kj} = \det \begin{pmatrix} A_{1,1} & \cdots & A_{1,k-1} & A_{1,j} \\ \vdots & & \vdots & \vdots \\ A_{k,1} & \cdots & A_{k,k-1} & A_{k,j} \end{pmatrix} \quad \text{where } j = 1, \dots, n.$$

Similarly, following the algorithm in Jeffrey (2010), we see that the columns of L are just made up by copying entries from the columns of U during the reduction. More precisely, the k^{th} column of L will have the entries $a_{1k}^{(k-1)}, \dots, a_{nk}^{(k-1)}$, using the notation of Geddes et al. (1992), and these are again just k -by- k minors of A . Explicitly, the j^{th} column of L consists of the elements

$$L_{kj} = \det \begin{pmatrix} A_{1,1} & \cdots & A_{1,j} \\ \vdots & & \vdots \\ A_{j-1,1} & \cdots & A_{j-1,j} \\ A_{k,1} & \cdots & A_{k,j} \end{pmatrix} \quad \text{where } k = 1, \dots, n.$$

□

We give an example using the domain $\mathbb{Q}[x]$. Let A be the polynomial matrix

$$\begin{pmatrix} -\frac{3}{2} & -x^3 + 5x^2 + 3x - \frac{9}{2} & x^2 + x & \frac{1}{2}x^3 - x^2 \\ -3 & -2x^3 + 10x^2 + 5x - 9 & 2x^2 + 2x & x^3 - 2x^2 \\ \frac{1}{2} & x^3 + \frac{3}{2} & 0 & -\frac{1}{2}x^3 \\ -\frac{1}{2} & -x - \frac{3}{2} & 0 & \frac{1}{2}x \end{pmatrix}.$$

The Smith–Jacobson normal form S of A is

$$\text{diag}(1, x, x(x+1), x(x+1)(x-1))$$

and thus its determinantal divisors are $d_1^* = 1$, $d_2^* = x$, $d_3^* = x^2(x+1)$ and $d_4^* = x^3(x+1)^2(x-1)$. Computing the $LD^{-1}U$ decomposition of A yields $A = LD^{-1}U$ where L is

$$\begin{pmatrix} -\frac{3}{2} & 0 & 0 & 0 \\ -3 & \frac{3}{2}x & 0 & 0 \\ \frac{1}{2} & -x^3 - \frac{5}{2}x^2 - \frac{3}{2}x & \frac{1}{2}x^3 + \frac{1}{2}x^2 & 0 \\ -\frac{1}{2} & -\frac{1}{2}x^3 + \frac{5}{2}x^2 + 3x & -\frac{1}{2}x^3 - \frac{1}{2}x^2 & -\frac{1}{4}x^6 - \frac{1}{4}x^5 + \frac{1}{4}x^4 + \frac{1}{4}x^3 \end{pmatrix},$$

$D = \text{diag}(-3/2, -9/4x, 3/4x^4 + 3/4x^3, -1/8x^9 - 1/4x^8 + 1/4x^6 + 1/8x^5)$, and U is

$$\begin{pmatrix} -\frac{3}{2} & -x^3 + 5x^2 + 3x - \frac{9}{2} & x^2 + x & \frac{1}{2}x^3 - x^2 \\ 0 & \frac{3}{2}x & 0 & 0 \\ 0 & 0 & \frac{1}{2}x^3 + \frac{1}{2}x^2 & -\frac{1}{2}x^4 - \frac{1}{2}x^3 \\ 0 & 0 & 0 & -\frac{1}{4}x^6 - \frac{1}{4}x^5 + \frac{1}{4}x^4 + \frac{1}{4}x^3 \end{pmatrix}.$$

Computing the column factors of L and the row factors of U yields the list $1, x, x^2(x+1)$ and $x^3(x-1)(x+1)^2$, i. e., exactly the determinantal divisors. In general, there could be other factors as well.

4. Efficient Detection of Factors

When considering the output of Algorithm 2, we find an interesting relation between the entries of L and U which can be exploited in order to find “systematic” common factors in the

$LD^{-1}U$ decomposition. Theorem 8 below shows that it is possible to compute a lower bound for the common factors in the k^{th} row of U by looking at just three entries of L . Likewise, we obtain lower bounds for the common factors of the k^{th} column of L from three of the entries of U . Note that the theorem leads to a very efficient way of detecting common factors since we only need to compute two greatest common divisors no matter how many entries the corresponding row (or column) has. Also note that in contrast to Theorem 6 the following result remains valid in the presence of pivoting. As in the previous section, let \mathbb{D} be a principal ideal domain.

Theorem 8. *Let $A \in \mathbb{D}^{m \times n}$ and let $P_rLD^{-1}UP_c$ be the $LD^{-1}U$ decomposition of A . Then*

$$\frac{\gcd(L_{k-1,k-1}, L_{k,k-1})}{\gcd(L_{k-1,k-1}, L_{k,k-1}, L_{k-2,k-2})} \mid U_{k,*}$$

and

$$\frac{\gcd(U_{k-1,k-1}, U_{k-1,k})}{\gcd(U_{k-1,k-1}, U_{k-1,k}, U_{k-2,k-2})} \mid L_{*,k}$$

for $k = 2, \dots, m-1$ (where we use $L_{0,0} = U_{0,0} = 1$ for $k = 2$).

PROOF. Suppose that during Bareiss's algorithm after $k-1$ iterations we have reached the following state

$$A^{(k-1)} = \begin{pmatrix} T & \underline{*} & \underline{*} & \underline{*} \\ \overline{0} & p & \underline{*} & \underline{*} \\ \overline{0} & 0 & a & \underline{\bar{v}} \\ \overline{0} & 0 & b & \underline{\bar{w}} \\ \mathbf{0} & \underline{0} & \underline{*} & \underline{*} \end{pmatrix},$$

where T is an upper triangular matrix, $p, a, b \in \mathbb{D}$, $\bar{v}, \bar{w} \in \mathbb{D}^{1 \times n-k-1}$ and the other overlined quantities are row vectors and the underlined quantities are column vectors. Assume that $a \neq 0$ and that we choose it as a pivot. Continuing the computations we now eliminate b (and the entries below) by cross-multiplication

$$A^{(k-1)} \rightsquigarrow \begin{pmatrix} T & \underline{*} & \underline{*} & \underline{*} \\ \overline{0} & p & \underline{*} & \underline{*} \\ \overline{0} & 0 & a & \underline{\bar{v}} \\ \overline{0} & 0 & 0 & a\bar{w} - b\bar{v} \\ \mathbf{0} & \underline{0} & \underline{0} & \underline{*} \end{pmatrix}.$$

Here, we can see that any common factor of a and b will be a factor of every entry in that row, i. e., $\gcd(a, b) \mid a\bar{w} - b\bar{v}$. However, we still have to carry out the exact division step. This leads to

$$A^{(k-1)} \rightsquigarrow \begin{pmatrix} T & \underline{*} & \underline{*} & \underline{*} \\ \overline{0} & p & \underline{*} & \underline{*} \\ \overline{0} & 0 & a & \underline{\bar{v}} \\ \overline{0} & 0 & 0 & \frac{1}{p}(a\bar{w} - b\bar{v}) \\ \mathbf{0} & \underline{0} & \underline{0} & \underline{*} \end{pmatrix} = A^{(k)}.$$

The division by p is exact. Some of the factors in p might be factors of a or b while others are hidden in \bar{v} or \bar{w} . However, every common factor of a and b which is not also a factor of p will

still be a common factor of the resulting row. In other words,

$$\frac{\gcd(a, b)}{\gcd(a, b, p)} \mid \frac{1}{p}(a\bar{w} - b\bar{v}).$$

In fact, the factors do not need to be tracked during the $LD^{-1}U$ reduction but can be computed afterwards: All the necessary entries a , b and p of $A^{(k-1)}$ will end up as entries of L . More precisely, we shall have $p = L_{k-2, k-2}$, $a = L_{k-1, k-1}$ and $b = L_{k, k-1}$.

Similar reasoning can be used to predict common factors in the columns of L . Here, we have to take into account that the columns of L are made up from entries in U during each iteration of the computation. \square

As a typical example consider the matrix

$$A = \begin{pmatrix} 8 & 49 & 45 & -77 & 66 \\ -10 & -77 & -19 & -52 & 48 \\ 51 & 18 & -81 & 31 & 69 \\ -97 & -58 & 37 & 41 & 22 \\ -60 & 0 & -25 & -18 & -92 \end{pmatrix}.$$

This matrix has a $LD^{-1}U$ decomposition with

$$L = \begin{pmatrix} 8 & 0 & 0 & 0 & 0 \\ -10 & -126 & 0 & 0 & 0 \\ 51 & -2355 & 134076 & 0 & 0 \\ -97 & 4289 & -233176 & -28490930 & 0 \\ -60 & 2940 & -148890 & -53377713 & 11988124645 \end{pmatrix}$$

and with

$$U = \begin{pmatrix} 8 & 49 & 45 & -77 & 66 \\ 0 & -126 & 298 & -1186 & 1044 \\ 0 & 0 & 134076 & -414885 & 351648 \\ 0 & 0 & 0 & -28490930 & 55072620 \\ 0 & 0 & 0 & 0 & 11988124645 \end{pmatrix}.$$

Note that in this example pivoting is not needed, that is, we have $P_r = P_c = \mathbf{1}$. The method outlined in Theorem 8 correctly predicts the common factor 2 in the second row, the factor 3 in the third row and the factor 2 in the fourth row. However, it does not detect the additional factor 5 in the fourth row.

The example also provides an illustration of the proof of Theorem 6: The entry -414885 of U at position $(3, 4)$ is given by the determinant of the submatrix

$$\begin{pmatrix} 8 & 49 & -77 \\ -10 & -77 & -52 \\ 51 & 18 & 31 \end{pmatrix}$$

consisting of the first three rows and columns 1, 2 and 4 of A . In this particular example, however, the Smith–Jacobson Normal Form of the matrix A is $\text{diag}(1, 1, 1, 1, 11988124645)$ which does not yield any information about the common factors.

Given Theorem 8, one can ask how good this prediction actually is. Concentrating on the case of integer matrices, the following Theorem 9 shows that with this prediction we do find a common factor in roughly a quarter of all rows. Experimental data suggest a similar behavior for matrices containing polynomials in $\mathbb{F}_p[x]$ where p is prime. Moreover, these experiments also showed that the prediction was able to account for 40.17% of all the common prime factors (counted with multiplicity) in the rows of U .¹

Theorem 9. *For random integers $a, b, p \in \mathbb{Z}$ the probability that the formula in Theorem 8 predicts a non-trivial common factor is*

$$\mathbb{P}\left(\frac{\gcd(a, b)}{\gcd(p, a, b)} = 1\right) = 6 \frac{\zeta(3)}{\pi^2} \approx 26.92\%.$$

PROOF. The following calculation is due to Hare (2016); Winterhof (2016): First note that the probability that $\gcd(a, b) = n$ is $1/n^2$ times the probability that $\gcd(a, b) = 1$. Summing up all of these probabilities gives

$$\sum_{n=1}^{\infty} \mathbb{P}(\gcd(a, b) = n) = \sum_{n=1}^{\infty} \frac{1}{n^2} \mathbb{P}(\gcd(a, b) = 1) = \mathbb{P}(\gcd(a, b) = 1) \frac{\pi^2}{6}.$$

As this sum must be 1, this gives that the $\mathbb{P}(\gcd(a, b) = 1) = 6/\pi^2$, and the $\mathbb{P}(\gcd(a, b) = n) = 6/(\pi^2 n^2)$. Given that $\gcd(a, b) = n$, the probability that $n \mid c$ is $1/n$. So the probability that $\gcd(a, b) = n$ and that $\gcd(p, a, b) = n$ is $6/(\pi^2 n^3)$. So $\mathbb{P}(\gcd(a, b) / \gcd(p, a, b) = 1)$ is

$$\sum_{n=1}^{\infty} \mathbb{P}(\gcd(a, b) = n \text{ and } \gcd(p, a, b) = n) = \sum_{n=1}^{\infty} \frac{6}{\pi^2 n^3} = 6 \frac{\zeta(3)}{\pi^2}. \quad \square$$

There is another way in which common factors in integer matrices can arise: Let d be any number. Then for random a, b the probability that $d \mid a + b$ is $1/d$. That means that if $v, w \in \mathbb{Z}^{1 \times n}$ are vectors, then $d \mid v + w$ with a probability of $1/d^n$. This effect is noticeable in particular for small numbers like $d = 2, 3$ and in the last iterations of the $LD^{-1}U$ decomposition when the number of non-zero entries in the rows has shrunk. For instance, in the second last iterations we only have three rows with at most three non-zero entries each. Moreover, we know that the first non-zero entries of the rows cancel during cross-multiplication. Thus, a factor of 2 appears with a probability of 25% in one of those rows, a factor of 3 with a probability of 11.11%. In the example above, the probability for the factor 5 to appear in the fourth row was 4%.

5. Expected Number of Factors

In this section, we provide a detailed analysis of the expected number of common “statistical” factors in the rows of U , in the case when the input matrix A has integer entries, that is, $\mathbb{D} = \mathbb{Z}$. We base our considerations on a “uniform” distribution on \mathbb{Z} , e.g., by imposing a uniform

¹This experiment was carried out with random square matrices A of sizes between 5-by-5 and 125-by-125. We decomposed A into $P_r LD^{-1} U P_c$ and then computed the number of predicted prime factors in U and related that to the number of actual prime factors. We did not consider the last row of U since this contains only the determinant.

distribution on $\{-n, \dots, n\}$ for very large n . However, the only relevant property that we use is the assumption that the probability that a randomly chosen integer is divisible by p is $1/p$.

We consider a matrix $A = (A_{i,j})_{1 \leq i, j \leq n} \in \mathbb{Z}^{n \times n}$ of full rank. The assumption that A be square is made for the sake of simplicity; the results shown below immediately generalize to rectangular matrices. As before, let U be the upper triangular matrix from the $LD^{-1}U$ decomposition of A :

$$U = \begin{pmatrix} U_{1,1} & U_{1,2} & \dots & U_{1,n} \\ 0 & U_{2,2} & \dots & U_{2,n} \\ \vdots & & \ddots & \vdots \\ 0 & \dots & & U_{n,n} \end{pmatrix}.$$

Define

$$g_k := \gcd(U_{k,k}, U_{k,k+1}, \dots, U_{k,n})$$

to be the greatest common divisor of all entries in the k^{th} row of U . Counting (with multiplicities) all the prime factors of g_1, \dots, g_{n-1} , one gets the picture shown in Figure 1; g_n is omitted as it contains only the single nonzero entry $U_{n,n} = \det(A)$. Our goal is to give a probabilistic explanation for the occurrence of these common factors, whose number seems to grow linearly with the dimension of the matrix.

As we have seen in the proof of Theorem 6, the entries $U_{k,\ell}$ can be expressed as minors of the original matrix A :

$$U_{k,\ell} = \det \begin{pmatrix} A_{1,1} & A_{1,2} & \dots & A_{1,k-1} & A_{1,\ell} \\ A_{2,1} & A_{2,2} & \dots & A_{2,k-1} & A_{2,\ell} \\ \vdots & \vdots & & \vdots & \vdots \\ A_{k,1} & A_{k,2} & \dots & A_{k,k-1} & A_{k,\ell} \end{pmatrix}.$$

Observe that the entries $U_{k,\ell}$ in the k^{th} row of U are all given as determinants of the same matrix, where only the last column varies. For any integer $q \geq 2$ we have that $q \mid g_k$ if q divides all these determinants. A sufficient condition for the latter to happen is that the determinant

$$h_k := \det \begin{pmatrix} A_{1,1} & \dots & A_{1,k-1} & 1 \\ A_{2,1} & \dots & A_{2,k-1} & x \\ \vdots & \vdots & & \vdots \\ A_{k,1} & \dots & A_{k,k-1} & x^{k-1} \end{pmatrix}$$

is divisible by q as a polynomial in $\mathbb{Z}[x]$, i.e., if q divides the content of the polynomial h_k . We now aim at computing how likely it is that $q \mid h_k$ when q is fixed and when the matrix entries $A_{1,1}, \dots, A_{k,k-1}$ are chosen randomly. Since q is now fixed, we can equivalently study this problem over the finite ring \mathbb{Z}_q , which means that the matrix entries are picked randomly and uniformly from the finite set $\{0, \dots, q-1\}$. Moreover, it turns out that it suffices to answer this question for prime powers $q = p^j$.

The probability that all $k \times k$ -minors of a randomly chosen $k \times (k+1)$ -matrix are divisible by p^j , where p is a prime number and $j \geq 1$ is an integer, is given by

$$P_{p,j,k} := 1 - \left(1 + p^{1-j-k} \frac{p^k - 1}{p - 1}\right) \prod_{i=0}^{k-1} (1 - p^{-j-i}),$$

which is a special case of [Brent and McKay \(1987, Thm. 2.1\)](#). Note that this is exactly the probability that h_{k+1} is divisible by p^j . Recalling the definition of the q -Pochhammer symbol

$$(a; q)_k := \prod_{i=0}^{k-1} (1 - aq^i), \quad (a; q)_0 := 1,$$

the above formula can be written more succinctly as

$$P_{p,j,k} := 1 - \left(1 + p^{1-j-k} \frac{p^k - 1}{p - 1}\right) \left(\frac{1}{p^j}; \frac{1}{p}\right)_k.$$

Now, an interesting observation is that this probability does not, as one could expect, tend to zero as k goes to infinity. Instead, it approaches a nonzero constant that depends on p and j (see [Table 1](#)):

$$P_{p,j,\infty} := \lim_{k \rightarrow \infty} P_{p,j,k} = 1 - \left(1 + \frac{p^{1-j}}{p - 1}\right) \left(\frac{1}{p^j}; \frac{1}{p}\right)_\infty$$

p^j	$k = 1$	$k = 2$	$k = 3$	$k = 4$	$k = 5$	$k = 6$	$k = \infty$
2	0.25000	0.34375	0.38477	0.40399	0.41330	0.41789	0.42242
3	0.11111	0.14403	0.15460	0.15808	0.15923	0.15962	0.15981
4	0.06250	0.09766	0.11560	0.12461	0.12912	0.13138	0.13364
5	0.04000	0.04768	0.04920	0.04951	0.04957	0.04958	0.04958
7	0.02041	0.02326	0.02367	0.02373	0.02374	0.02374	0.02374
8	0.01563	0.02588	0.03149	0.03440	0.03588	0.03662	0.03737

Table 1: Behavior of the sequence $(P_{p,j,k})_{k \in \mathbb{N}}$ for some small values of p^j .

Using the probability $P_{p,j,k}$, one can write down the expected number of factors in the determinant h_{k+1} , i.e., the number of prime factors in the content of the polynomial h_{k+1} , counted with multiplicities:

$$\sum_{p \in \mathbb{P}} \sum_{j=1}^{\infty} P_{p,j,k},$$

where $\mathbb{P} = \{2, 3, 5, \dots\}$ denotes the set of prime numbers. The inner sum can be simplified as follows, yielding the expected multiplicity $M_{p,k}$ of a prime factor p in h_{k+1} :

$$\begin{aligned} M_{p,k} &:= \sum_{j=1}^{\infty} P_{p,j,k} = \sum_{j=1}^{\infty} \left(1 - \left(1 + p^{1-j-k} \frac{p^k - 1}{p - 1}\right) \left(\frac{1}{p^j}; \frac{1}{p}\right)_k\right) \\ &= - \sum_{j=1}^{\infty} \left(\left(\frac{1}{p^j}; \frac{1}{p}\right)_k - 1\right) - p^{1-k} \frac{p^k - 1}{p - 1} \sum_{j=1}^{\infty} \frac{1}{p^j} \left(\frac{1}{p^j}; \frac{1}{p}\right)_k \\ &= - \sum_{j=1}^{\infty} \sum_{i=1}^k (-1)^i p^{-ij - i(i-1)/2} \begin{bmatrix} k \\ i \end{bmatrix}_{1/p} - p^{1-k} \frac{p^k - 1}{p - 1} \frac{p^k}{p^{k+1} - 1} \\ &= \sum_{i=1}^k \frac{(-1)^{i-1}}{p^{i(i-1)/2} (p^i - 1)} \begin{bmatrix} k \\ i \end{bmatrix}_{1/p} + \frac{1}{p^{k+1} - 1} - \frac{1}{p - 1} \end{aligned}$$

In this derivation we have used the expansion formula of the q -Pochhammer symbol in terms of the q -binomial coefficient

$$\begin{bmatrix} n \\ k \end{bmatrix}_q := \frac{(1-q^n)(1-q^{n-1})\cdots(1-q^{n-k+1})}{(1-q^k)(1-q^{k-1})\cdots(1-q)},$$

evaluated at $q = 1/p$. Moreover, the identity that is used in the third step,

$$\sum_{j=1}^{\infty} \frac{1}{p^j} \left(\frac{1}{p^j}; \frac{1}{p} \right)_k = \frac{p^k}{p^{k+1} - 1},$$

is certified by rewriting the summand as

$$\frac{1}{p^j} \left(\frac{1}{p^j}; \frac{1}{p} \right)_k = t_{j+1} - t_j \quad \text{with} \quad t_j = \frac{p^k(p^{1-j} - 1)}{p^{k+1} - 1} \left(\frac{1}{p^j}; \frac{1}{p} \right)_k$$

and by applying a telescoping argument.

Hence, when we let k go to infinity, we obtain

$$M_{p,\infty} = \lim_{k \rightarrow \infty} \sum_{j=1}^{\infty} P_{p,j,k} = \sum_{i=1}^{\infty} \frac{(-1)^{i-1}}{p^{i(i-1)/2}(p^i - 1)} \frac{(p^{-i-1}; p^{-1})_{\infty}}{(p^{-1}; p^{-1})_{\infty}} - \frac{1}{p-1}.$$

Note that the sum converges quickly, so that one can use the above formula to compute an approximation for the expected number of factors in h_{k+1} when k tends to infinity

$$\sum_{p \in \mathbb{P}} M_{p,\infty} \approx 0.89764,$$

which gives the asymptotic slope of the function plotted in Figure 1.

As discussed before, the divisibility of h_k by some number $q \geq 2$ implies that the greatest common divisor g_k of the k^{th} row is divisible by q , but this is not a necessary condition. It may happen that h_k is not divisible by q , but nevertheless q divides each $U_{k,\ell}$ for $k \leq \ell \leq n$. The probability for this to happen is the same as the probability that the greatest common divisor of $n - k + 1$ randomly chosen integers is divisible by q . The latter obviously is $q^{-(n-k+1)}$. Thus, in addition to the factors coming from h_k , one can expect

$$\sum_{p \in \mathbb{P}} \sum_{j=1}^{\infty} \frac{1}{p^{j(n-k+1)}} = \sum_{p \in \mathbb{P}} \frac{1}{p^{n-k+1} - 1}$$

many prime factors in g_k .

Summarizing, the expected number of prime factors in the rows of the matrix U is

$$\begin{aligned} F(n) &= \sum_{k=2}^{n-1} \sum_{p \in \mathbb{P}} M_{p,k-1} + \sum_{k=1}^{n-1} \sum_{p \in \mathbb{P}} \frac{1}{p^{n-k+1} - 1} \\ &= \sum_{p \in \mathbb{P}} \left(\sum_{k=0}^{n-2} M_{p,k} + \sum_{k=0}^{n-2} \frac{1}{p^{k+2} - 1} \right) \\ &= \sum_{p \in \mathbb{P}} \sum_{k=0}^{n-2} \left(\sum_{i=1}^k \frac{(-1)^{i-1}}{p^{i(i-1)/2}(p^i - 1)} \begin{bmatrix} k \\ i \end{bmatrix}_{1/p} + \frac{1}{p^{k+2} - 1} + \frac{1}{p^{k+1} - 1} - \frac{1}{p-1} \right). \end{aligned}$$

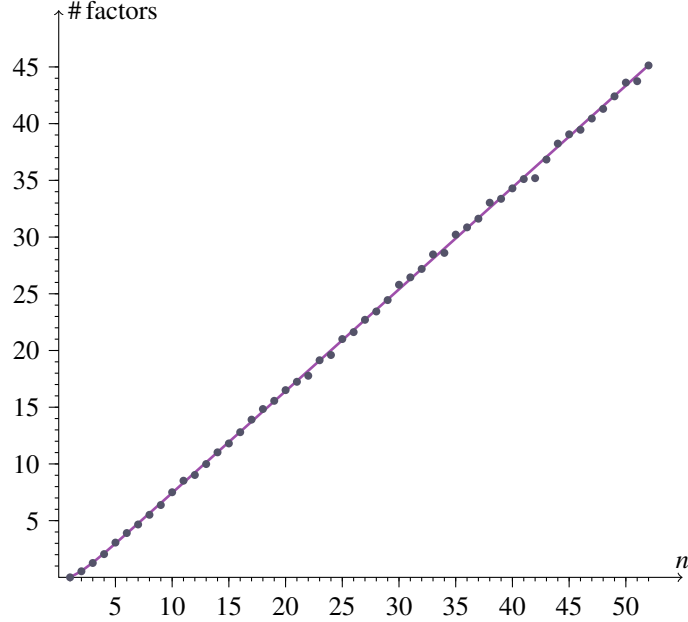


Figure 1: Number of factors depending on the size n of the matrix. The curve shows the function $F(n)$, while the dots represent experimental data: for each dimension n , 1000 matrices were generated with random integer entries between 0 and 10^9 .

From the discussion above, it follows that for large n this expected number can be approximated by a linear function as follows:

$$F(n) \approx 0.89764n - 1.53206.$$

6. Solving Linear System via $LD^{-1}U$ Decomposition

In this section we detail a method for solving linear systems in such a way that fractions are delayed until the final output. A fraction-free solving method was already discussed in [Bareiss \(1968\)](#); while it was restricted to invertible matrices, our method works for an arbitrary input matrix A . In particular, our method gives an explicit representation of the kernel of A . Moreover, the method we propose is designed for a generic right-hand side, so that the reduction has to be done only once, in the situation where several systems with different right-hand sides have to be solved. As in Cramer's rule, the denominator of the solution in Bareiss' method is just $\det A$, whereas in our formulation we obtain smaller denominators in general.

Let $A \in \mathbb{D}^{m \times n}$ and $b \in \mathbb{D}^m$, where now \mathbb{D} is only assumed to be an integral domain. We wish to solve the system $Ax = b$, seeking solutions x with entries in the field of fractions of \mathbb{D} . First, apply the $LD^{-1}U$ decomposition as in [Jeffrey \(2010\)](#) but without trimming the resulting matrices. We obtain

$$DL^{-1}P_w^t A = \begin{pmatrix} V \\ W \end{pmatrix} A = \begin{pmatrix} U & B \\ 0 & 0 \end{pmatrix} P_c \quad \text{and} \quad P_c^{-1}x = \begin{pmatrix} y \\ z \end{pmatrix},$$

where all (sub) matrices have entries in \mathbb{D} , U is an r -by- r , regular and upper triangular matrix, r is the rank of A and where y has dimension r . Then $Ax = b$ if and only if $Wb = 0$ and $Uy + Bz = Vb$.

Now, perform a second $LD^{-1}U$ decomposition on U (pivoting is not needed as all diagonal entries of U are non-zero), working from the bottom to the top, and from right to left². This will compute a regular $X \in \mathbb{D}^{r \times r}$ such that $XU = \Delta$ is a diagonal matrix. Then $Ax = b$ if and only if $Wb = 0$ and $\Delta y + XBz = XVb$.

Assume now that the compatibility condition $Wb = 0$ is fulfilled. In order to compute a particular solution x_0 of the system $Ax = b$, we can simply choose

$$x_0 = P_c^{-1} \begin{pmatrix} \Delta^{-1} XVb \\ 0 \end{pmatrix} = \tilde{\Delta}^{-1} S b \quad \text{where} \quad S = P_c \begin{pmatrix} XV \\ 0 \end{pmatrix}$$

and where $\tilde{\Delta} = P_c \text{diag}(\Delta, \mathbf{1}) P_c$ is a diagonal matrix with entries in \mathbb{D} .

Moreover, we can compute the nullspace of A in the following way: If

$$x \in \text{colspace } P_c^{-1} \begin{pmatrix} -\Delta^{-1} XB \\ \mathbf{1}_{n-r} \end{pmatrix},$$

then we can easily check that $Ax = 0$ using that $(U^t, V^t)^t$ is regular and that $\Delta^{-1}X = U^{-1}$. Since the $n - r$ columns of the matrix spanning the space are clearly linearly independent, it follows that this is already the entire nullspace of A . Thus, setting

$$K = P_c \begin{pmatrix} -XB \\ \mathbf{1} \end{pmatrix},$$

we see the nullspace of A is $\text{colspace } \tilde{\Delta}^{-1}K$, with $\tilde{\Delta}$ as defined above.

Note that S and K are both matrices over \mathbb{D} . Thus, the particular solution and the nullspace are both computed in a fraction-free way. Moreover, neither of the matrices depends on the right hand side b . Consequently, after computing W , S , $\tilde{\Delta}$ and K , we can solve the system $Ax = b$ for arbitrary b by just checking whether $Wb = 0$ and then computing $x_0 = \tilde{\Delta}^{-1}Sb$.

We summarize the method as follows:

Algorithm 10. (*Fraction-free solving of a linear system*)

Input: A matrix $A \in \mathbb{D}^{m \times n}$.

Output: Matrices W , S , and K with entries in \mathbb{D} and a diagonal matrix $\tilde{\Delta}$ with entries in \mathbb{D} such that for any $b \in \mathbb{D}^m$ if the compatibility condition $Wb = 0$ is met, then the system $Ax = b$ has the solution set $\tilde{\Delta}^{-1}Sb + \text{colspace } \tilde{\Delta}^{-1}K$.

1. Apply the $LD^{-1}U$ decomposition to obtain

$$DL^{-1}P_w^t A = \begin{pmatrix} V \\ W \end{pmatrix} A = \begin{pmatrix} U & B \\ 0 & 0 \end{pmatrix}$$

where U is upper triangular.

²More formally, let Π be the matrix of the permutation which maps i to $r + 1 - i$ and decompose $\Pi U \Pi$ in the normal way applying the same permutations to the result.

2. Use a backwards $LD^{-1}U$ decomposition on U to obtain a matrix X such that diagonal $XU = \Delta$ is a diagonal matrix.
3. Let

$$S = P_c \begin{pmatrix} XV \\ 0 \end{pmatrix}, \quad K = P_c \begin{pmatrix} -XB \\ \mathbf{1} \end{pmatrix}$$

$$\text{and } \tilde{\Delta} = P_c \text{diag}(\Delta, \mathbf{1})P_c.$$

As an example we consider the matrix

$$A = \begin{pmatrix} -370 & -62 & -101 & -3 \\ -708 & -120 & -193 & -5 \\ -304 & -50 & -83 & -3 \\ -1962 & -336 & -534 & -12 \end{pmatrix}$$

and examine the two systems below for solutions.

$$Ax = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 1 \end{pmatrix} = b_1 \quad \text{and} \quad Ax = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \end{pmatrix} = b_2.$$

Following Algorithm 10, we first compute

$$\left(\begin{array}{cccc|cccc} 1 & 0 & 0 & 0 & -3 & -62 & -101 & -370 \\ 3 & 0 & -3 & 0 & 0 & -36 & -54 & -198 \\ 110 & -36 & -50 & 0 & 0 & 0 & -12 & -12 \\ 7 & -6 & -1 & 1 & 0 & 0 & 0 & 0 \end{array} \right) A = \left(\begin{array}{ccc|c} -3 & -62 & -101 & -370 \\ 0 & -36 & -54 & -198 \\ 0 & 0 & -12 & -12 \\ 0 & 0 & 0 & 0 \end{array} \right) P_c$$

where P_c represents the permutation (1 4); and use this to define the matrices V , W , U and B . Next, we compute

$$X = \begin{pmatrix} 432 & -744 & -288 \\ 0 & -12 & 54 \\ 0 & 0 & 1 \end{pmatrix}$$

and $XU = \text{diag}(-1296, 432, -12) = \Delta$. This leads to

$$S = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 5904 & -1944 & -2664 & 0 \\ 110 & -36 & -50 & 0 \\ -33480 & 10368 & 16632 & 0 \end{pmatrix}, \quad K = \begin{pmatrix} 1 \\ -1728 \\ 12 \\ 9072 \end{pmatrix}$$

and $\tilde{\Delta} = \text{diag}(1, 432, -12, -1296)$.

We can check that $Wb_1 = 8 \neq 0$. Consequently, the system $Ax = b_1$ does not have a solution. On the other hand, $Wb_2 = 0$ and the solution set for $Ax = b_2$ is

$$\tilde{\Delta}^{-1}Sb + \text{colspace } \tilde{\Delta}^{-1}K = \begin{pmatrix} 0 \\ -37/6 \\ 25/6 \\ -77/6 \end{pmatrix} + \text{colspace} \begin{pmatrix} 1 \\ -4 \\ -1 \\ -7 \end{pmatrix}.$$

7. QR Decomposition

A fraction-free QR decomposition, which is based on the $LD^{-1}U$ decomposition, was given in [Zhou and Jeffrey \(2008\)](#). In this section, we present a refined version of this algorithm (see [Theorem 12](#)). As a first step in its proof, we will need the Cholesky decomposition, which is introduced in the following lemma.

Theorem 11. *Let $A \in \mathbb{D}^{n \times n}$ be a symmetric matrix such that its $LD^{-1}U$ decomposition can be computed without permutations; then we have $U = L^t$, that is,*

$$A = LD^{-1}L^t.$$

PROOF. Compute the decomposition $A = LD^{-1}U$ as in [Theorem 1](#). If we do not execute step 4 of [Algorithm 2](#), we obtain the decomposition

$$A = \tilde{L}\tilde{D}^{-1}\tilde{U} = \begin{pmatrix} \mathcal{L} & \mathbf{0} \\ \mathcal{M} & \mathbf{1} \end{pmatrix} \begin{pmatrix} D & \mathbf{0} \\ \mathbf{0} & \mathbf{1} \end{pmatrix}^{-1} \begin{pmatrix} \mathcal{U} & \mathcal{V} \\ \mathbf{0} & \mathbf{0} \end{pmatrix}.$$

Then because A is symmetric, we obtain

$$\tilde{L}\tilde{D}^{-1}\tilde{U} = A = A^t = \tilde{U}^t\tilde{D}^{-1}\tilde{L}^t$$

The matrices \tilde{L} and \tilde{D} have full rank which implies

$$\tilde{U}(\tilde{L}^t)^{-1}\tilde{D} = \tilde{D}\tilde{L}^{-1}\tilde{U}^t.$$

Examination of the matrices on the left hand side reveals that they are all upper triangular. Therefore also their product is an upper triangular matrix. Similarly, the right hand side is a lower triangular matrix and the equality of the two implies that they must both be diagonal. Canceling \tilde{D} and rearranging the equation yields $\tilde{U} = (\tilde{L}^{-1}\tilde{U}^t)\tilde{L}^t$ where $\tilde{L}^{-1}\tilde{U}^t$ is diagonal. This shows that the rows of \tilde{U} are just multiples of the rows of \tilde{L}^t . However, we know that the first r diagonal entries of \tilde{U} and \tilde{L} are the same, where r is the rank of \tilde{U} . This yields

$$\tilde{L}^{-1}\tilde{U}^t = \begin{pmatrix} \mathbf{1}_r & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix},$$

and hence, when we remove the unnecessary last $n - r$ rows of \tilde{U} and the last $n - r$ columns of \tilde{L} (as suggested in [Jeffrey \(2010\)](#)), we remain with $U = L^t$. \square

The following theorem is a variant of [Zhou and Jeffrey \(2008, Thm. 8\)](#), where we exploit the symmetry of A^tA by invoking [Theorem 11](#). This leads to a nicer representation of the decomposition, and we obtain more information about $\Theta^t\Theta$.

Theorem 12. *Let $A \in \mathbb{D}^{m \times n}$ with $n \leq m$ and with full column rank. Then the partitioned matrix $(A^tA \mid A^t)$ has $LD^{-1}U$ decomposition*

$$(A^tA \mid A^t) = R^tD^{-1}(R \mid \Theta^t),$$

where $\Theta^t\Theta = D$ and $A = \Theta D^{-1}R$.

PROOF. Since A has full column rank, the *Gramian matrix* A^tA will have full rank, too. By taking the first k columns of A (and the first k rows of A^t), it follows that also the k^{th} principal minor of A^tA is nonzero. Consequently, when we compute the $LD^{-1}U$ decomposition, we do not need any permutations.

Hence, by Theorem 11, we can decompose the symmetric matrix A^tA as

$$A^tA = R^tD^{-1}R.$$

Applying the same row transformations to A^t yields a matrix Θ^t , that is, we obtain $(A^tA \mid A^t) = R^tD^{-1}(R \mid \Theta^t)$. As in the proof of Zhou and Jeffrey (2008, Thm. 8), we easily compute that $A = \Theta D^{-1}R$ and that $\Theta^t\Theta = D^t(R^{-1})^tA^tAR^{-1}D = D^t(R^{-1})^tR^tD^{-1}RR^{-1}D = D$. \square

For example, let $A \in \mathbb{Z}[x]^{3 \times 3}$ be the matrix

$$A = \begin{pmatrix} x & 1 & 2 \\ 2 & 0 & -x \\ x & 1 & x+1 \end{pmatrix}.$$

Then the $LD^{-1}U$ decomposition of $A^tA = R^tD^{-1}R$ is given by

$$R = \begin{pmatrix} 2(x^2+2) & 2x & x(x+1) \\ 0 & 8 & 4(x^2+x+3) \\ 0 & 0 & 4(x-1)^2 \end{pmatrix},$$

$$D = \begin{pmatrix} 2(x^2+2) & 0 & 0 \\ 0 & 16(x^2+2) & 0 \\ 0 & 0 & 32(x-1)^2 \end{pmatrix},$$

and we obtain for the QR decomposition $A = \Theta D^{-1}R$:

$$\Theta = \begin{pmatrix} x & 4 & -4(x-1) \\ 2 & -4x & 0 \\ x & 4 & 4(x-1) \end{pmatrix}.$$

We see that the $\Theta D^{-1}R$ decomposition has some common factor in the last column of Θ . This observation is explained by the following theorem.

Theorem 13. *With full-rank $A \in \mathbb{D}^{n \times n}$ and Θ as in Theorem 12, we have for all $i = 1, \dots, n$ that*

$$\Theta_m = (-1)^{n+i} \det_{i,n} A \cdot \det A$$

where $\det_{i,n} A$ is the (i, n) minor of A .

PROOF. We use the notation from the proof of Theorem 12. From $\Theta D^{-1}R = A$ and $\Theta^t\Theta = D$ we obtain

$$\Theta^tA = \Theta^t\Theta D^{-1}R = R.$$

Thus, since A has full rank, $\Theta^t = RA^{-1}$ or, equivalently,

$$\Theta = (RA^{-1})^t = (A^{-1})^tR^t = (\det A)^{-1}(\text{adj } A)^tR^t$$

where $\text{adj } A$ is the adjugate matrix of A . Since R^t is a lower triangular matrix with $\det A^tA = (\det A)^2$ at position (n, n) , the claim follows. \square

Knowing that there is always a common factor, we can cancel it, which leads to a fraction-free QR decomposition of smaller size.

Theorem 14. *Given a square matrix A , a reduced fraction-free QR decomposition is given by $A = \hat{\Theta}\hat{D}^{-1}\hat{R}$, where $S = \text{diag}(1, 1, \dots, \det A)$ and $\hat{\Theta} = \Theta S^{-1}$, and $\hat{R} = S^{-1}R$. In addition, $\hat{D} = S^{-1}DS^{-1} = \hat{\Theta}'\hat{\Theta}$.*

PROOF. By Theorem 13, ΘS^{-1} is an exact division. The statement of the theorem follows from $A = \Theta S^{-1}S D^{-1}S S^{-1}R$. \square

If we apply Theorem 14 to our previous example, we obtain the simpler QR decomposition, where the factor $\det A = -2(x-1)$ has been removed.

$$\begin{pmatrix} x & 4 & 2 \\ 2 & -4x & 0 \\ x & 4 & -2 \end{pmatrix} \begin{pmatrix} 2(x^2+2) & 0 & 0 \\ 0 & 16(x^2+2) & 0 \\ 0 & 0 & 8 \end{pmatrix}^{-1} \begin{pmatrix} 2(x^2+2) & 2x & x(x+1) \\ 0 & 8 & 4(x^2+x+3) \\ 0 & 0 & -2(x-1) \end{pmatrix}.$$

The properties of the QR -decomposition are strong enough to guarantee a certain uniqueness of the output.

Theorem 15. *Let $A \in \mathbb{D}^{n \times n}$ have full rank. Let $A = \Theta D^{-1}R$ the decomposition from Theorem 12; and let $A = \tilde{\Theta}\tilde{D}^{-1}\tilde{R}$ be another decomposition where $\tilde{\Theta}, \tilde{D}, \tilde{R} \in \mathbb{D}^{n \times n}$ are such that \tilde{D} is a diagonal matrix, \tilde{R} is an upper triangular matrix and $\tilde{\Theta}'\tilde{\Theta}$ is a diagonal matrix. Then $\Theta'\tilde{\Theta}$ is also a diagonal matrix and $\tilde{R} = (\Theta'\tilde{\Theta})^{-1}\tilde{D}R$.*

PROOF. We have

$$\tilde{\Theta}\tilde{D}^{-1}\tilde{R} = \Theta D^{-1}R \quad \text{and thus} \quad \Theta'\tilde{\Theta}\tilde{D}^{-1}\tilde{R} = \Theta'\Theta D^{-1}R = R.$$

If R and \tilde{R} have full rank, this is equivalent to

$$\Theta'\tilde{\Theta} = R\tilde{R}^{-1}\tilde{D}.$$

Note that all the matrices on the right hand side are upper triangular. Similarly, we can compute that

$$\tilde{\Theta}'\Theta D^{-1}R = \tilde{\Theta}'\tilde{\Theta}\tilde{D}^{-1}\tilde{R} = \Delta\tilde{D}^{-1}\tilde{R}$$

which implies $\tilde{\Theta}'\Theta = \Delta\tilde{D}^{-1}\tilde{R}R^{-1}D$. Hence, also $\tilde{\Theta}'\Theta = (\Theta'\tilde{\Theta})'$ is upper triangular and consequently $\tilde{\Theta}'\Theta = T$ for some diagonal matrix T with entries from \mathbb{D} . We obtain $R = T\tilde{D}^{-1}\tilde{R}$ and thus $\tilde{R} = T^{-1}\tilde{D}R$. \square

8. Acknowledgments

J. M. was supported in part by the Austrian Science Fund (FWF): SFB50 (F5009-N15). C. K. was supported by the Austrian Science Fund (FWF): P29467-N32 and W1214.

We would like to thank Kevin G. Hare and Arne Winterhof for helpful comments and discussions.

References

- Bareiss, E. H., 1968. Sylvester's identity and multistep integer-preserving Gaussian elimination. *Mathematics of Computation* 22 (103), 565–578.
- Brent, R. P., McKay, B. D., 1987. Determinants and ranks of random matrices over \mathbb{Z}_m . *Discrete Mathematics*, 35–49.
- Dumas, J.-G., Pernet, C., Sultan, Z., 2015. Computing the rank profile matrix. In: Robertz, D. (Ed.), *Proceedings of the International Symposium on Symbolic and Algebraic Computation*. ACM Press, pp. 149–156.
URL [doi:10.1145/2755996.2756682](https://doi.org/10.1145/2755996.2756682)
- Erlingsson, Ú., Kaltofen, E., Musser, D., 1996. Generic Gram–Schmidt orthogonalization by exact division. In: *International Symposium on Symbolic and Algebraic Computation*. ACM Press, pp. 275–282.
- Geddes, K. O., Czapor, S. R., Labahn, G., 1992. *Algorithms for Computer Algebra*. Kluwer Academic Publisher.
- Giesbrecht, M. W., Storjohann, A., 2002. Computing rational forms of integer matrices. *Journal of Symbolic Computation* 34 (3), 157–172.
- Hare, K. G., 2016. Personal Communication.
- Jebelean, T., 1993. An algorithm for exact division. *J. Symbolic Computation* 15, 169–180.
- Jeffrey, D. J., 2010. *LU* factoring of non-invertible matrices. *ACM Communications in Computer Algebra* 44 (171), 1–8.
- Kaltofen, E., Yuhasz, G., 2013. A fraction free matrix Berlekamp/Massey algorithm. *Linear Algebra and its Applications* 439 (9), 2515–2526.
- Lee, H. R., Saunders, B. D., 1995. Fraction free gaussian elimination for sparse matrices. *Journal of Symbolic Computation* 19 (5), 393–402.
- Middeke, J., Jeffrey, D. J., 2014. Fraction-free factoring revisited. Poster presentation at the International Symposium on Symbolic and Algebraic Computation.
- Nakos, G. C., Turner, P. R., Williams, R. M., 1997. Fraction-free algorithms for linear and polynomial equations. *SIGSAM Bulletin* 31 (3), 11–19.
URL [doi:http://doi.acm.org/10.1145/271130.271133](http://doi.acm.org/10.1145/271130.271133)
- Newman, M., 1972. *Integral Matrices*. Vol. 45 of *Pure and Applied Mathematics*. Academic Press, New York.
- Pauderis, C., Storjohann, A., 2013. Computing the invariant structure of integer matrices: fast algorithms into practice. In: Kauers, M. (Ed.), *Proceedings of the International Symposium on Symbolic and Algebraic Computation*. ACM Press, pp. 307–314.
- Winterhof, A., 2016. Personal Communication.
- Zhou, W., Jeffrey, D. J., 2008. Fraction-free matrix factors: new forms for *LU* and *QR* factors. *Frontiers of Computer Science in China* 2 (1), 67–80.