Rank Revealing Gaussian Elimination by the Maximum Volume Concept

Lukas Schork* Jacek Gondzio†

School of Mathematics, University of Edinburgh, Edinburgh EH9 3FD, Scotland, UK

Technical Report ERGO 18-002, revision 1, March 17, 2019

Abstract

A Gaussian elimination algorithm is presented that reveals the numerical rank of a matrix and identifies a square nonsingular submatrix of maximum dimension. The bounds on the singular values of the submatrix and its Schur complement are similar to the best known bounds for rank revealing *LU* factorization, but in contrast to existing methods the algorithm does not make use of the normal matrix. An implementation for dense matrices is described whose computational cost is roughly twice the cost of an *LU* factorization with complete pivoting. Because of its flexibility in choosing pivot elements, the algorithm is amenable to implementation with blocked memory access and for sparse matrices.

1 Introduction

This paper is concerned with the problem to determine the rank of a matrix in the numerical sense and to identify a square nonsingular submatrix of maximum dimension. Given $A \in \mathbb{R}^{m\times n}$ and a tolerance $\varepsilon > 0$, the task is to determine an index r such that $\sigma_r \geq \varepsilon$ and $\sigma_{r+1} = O(\varepsilon)$, where $\sigma_1 \geq \ldots \geq \sigma_d \geq 0$ ($d = \min(m,n)$) are the singular values of A and $\sigma_{d+1} := 0$. Our definition of numerical rank relaxes the condition $\sigma_r \geq \varepsilon > \sigma_{r+1}$, which would be achievable only by computing the singular values. In addition to the rank r, we want to identify an $r \times r$ submatrix of A whose minimum singular value is not too much smaller than σ_r.

The stated problem arises, for example, in computing null space bases of large and sparse matrices. Suppose that $A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}$ has rank r and that A_{11} is $r \times r$ nonsingular. Then the Schur complement $A/A_{11} = A_{22} - A_{21}A_{11}^{-1}A_{12}$ is all zero and the columns of $Z := \begin{bmatrix} -A_{11}^{-1}A_{12} \\ I_{n-r} \end{bmatrix}$ form a basis for the null space of A. (Here and in the sequel I_k denotes the identity matrix of dimension k.) When instead A has numerical rank r, one may wish to find a submatrix A_{11} of dimension r so that $\|A/A_{11}\| = O(\varepsilon)$, making Z a null space basis for an ε-perturbation of A. This allows to extend the null space method for solving KKT systems [11, Section 16.3] to the case where the constraint matrix is numerically rank

*L.Schork@ed.ac.uk
†J.Gondzio@ed.ac.uk

deficient. Such systems arise from rank deficient least squares problems and in nonlinear optimization when the constraint normals at the current point are (close to) linearly dependent. Because the matrices in these applications are typically large and sparse, methods based on orthogonal factorizations can be unacceptably expensive in terms of computation time and memory requirement. We are therefore interested in solving the above problem by Gaussian elimination, which is generally better suited for sparse matrices.

It is well known that Gaussian elimination with complete pivoting may not detect a near singularity. For the example from [13],

$$A = \begin{bmatrix} 1 & -1 & \cdots & -1 & -1 \\ & 1 & & & -1 \\ & & \ddots & & \vdots \\ & & & 1 & -1 \\ & & & & 1 \end{bmatrix} \in \mathbb{R}^{m \times m}, \tag{1}$$

complete pivoting allows to choose the diagonal elements as pivots, so that no eliminations are needed and A is determined to be of full rank. It is not revealed that $\sigma_m(A) = O(2^{-m})$ (see [12, Section 5]) and the numerical rank of A to be $m-1$ for m moderately large.

A method based on Gaussian elimination that does reveal the rank and identifies a submatrix with the desired properties was described by Pan [12]. The method employs the "maximum volume" criterion, firstly to choose a column subset of A and then to choose a square submatrix within these columns. The concept of maximum volume has been used before in rank revealing factorizations and related topics, see [10, 9, 12, 8, 7] and the references therein. The issue with Pan's method is that choosing the column subset requires operations with the inverse of normal matrices $A_{\mathcal{J}}^T A_{\mathcal{J}}$, where $\mathcal{J}$ is an index set that is updated by a pivot rule and $A_{\mathcal{J}}$ denotes the matrix composed of the corresponding columns of A. Although it is possible to implement these operations in the sparse case through Cholesky factorization [3], the computations can be much more expensive than operations with the inverse of a square submatrix of A.

The algorithm presented in this paper avoids operations with normal matrices. Instead, it selects an $m \times m$ "basis" matrix $\mathbb{A}_{\mathcal{B}}$ of local maximum volume in $\mathbb{A} = \begin{bmatrix} A & \beta I_m \end{bmatrix}$ for a suitably chosen scalar $\beta > 0$. A square submatrix of A will then be defined by means of the columns of A and the columns of βI_m which compose $\mathbb{A}_{\mathcal{B}}$. The advantage of the new method is that it can be implemented by updating a sparse LU factorization after column changes to the basis matrix, which is a common operation in linear programming for which computationally efficient techniques exist [4].

It will be shown that the obtained submatrix A_{11} satisfies

$$\|A/A_{11}\|_{\max} \le \beta \quad \text{and} \quad \left\|A_{11}^{-1}\right\|_{\max} \le \beta^{-1},$$

where $\|\cdot\|_{\max}$ is the maximum absolute entry of a matrix. Furthermore, for $\beta = \min(m,n)\varepsilon$ the dimension r of A_{11} will reveal the numerical rank of A. We will derive a lower bound on $\sigma_{\min}(A_{11})$ and an upper bound on $\|A/A_{11}\|_2$ in terms of $\sigma_r(A)$ and $\sigma_{r+1}(A)$, which are very similar to the bounds proved for Pan's method [12]. Applied to the matrix (1), the algorithm selects the upper right $(m-1) \times (m-1)$ block as A_{11}, for which $\sigma_{\min}(A_{11}) \approx \sigma_{m-1}(A)$ and $\|A/A_{11}\|_2 = O(2^{-m})$.

Throughout the paper A is an $m \times n$ matrix and A_{11} is a square nonsingular submatrix. It is assumed that A has been permuted so that

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}. \tag{2}$$

The Schur complement of A_{11} in A is

$$A/A_{11} = A_{22} - A_{21}A_{11}^{-1}A_{12}.$$

$\sigma_k(\cdot)$ and $\sigma_{\min}(\cdot)$ denote the k-th largest and the minimum singular value of a matrix. $\|\cdot\|_2$ and $\|\cdot\|_{\max}$ are the maximum singular value and the maximum absolute entry of a matrix, which satisfy the relation

$$\|A\|_{\max} \le \|A\|_2 \le \sqrt{mn}\,\|A\|_{\max}.$$

Ordered index sets are denoted by calligraphic letters. $\mathcal{J}_p$ means the p-th index in the set and $A_{\mathcal{J}}$ is the matrix composed of the columns of A indexed by $\mathcal{J}$. A basis $\mathcal{B}$ for $\mathbb{A} \in \mathbb{R}^{m\times(n+m)}$ is an index set such that the basis matrix $\mathbb{A}_{\mathcal{B}}$ is square and nonsingular (requiring that $\mathbb{A}$ has rank m). Associated with $\mathcal{B}$ is the nonbasic set $\mathcal{N} = \{1,\dots,n+m\} \setminus \mathcal{B}$. Vectors are notated in bold lower case, where $\boldsymbol{e}_j$ is the j-th unit vector. Expressions like $|A|$ and $|\boldsymbol{b}|$ are meant componentwise.

2 Maximum Volume Concept

The volume of a matrix of arbitrary dimension and rank was introduced in [1]. This paper uses the definition from [12], which differs in that the volume of a rank deficient matrix is zero.

Definition 2.1. *For $A \in \mathbb{R}^{m\times n}$ with singular values $\sigma_1 \ge \ldots \ge \sigma_d \ge 0$ ($d = \min(m,n)$), the volume of A is defined by*

$$\operatorname{vol}(A) = \sigma_1 \cdots \sigma_d.$$

In particular, the volume of a square matrix is the absolute value of its determinant.

Definition 2.2. *Let $A \in \mathbb{R}^{m\times n}$ and $\rho \ge 1$.*

(i) Let B be a $k\times k$ submatrix of A. $\operatorname{vol}(B) (\neq 0)$ is said to be a global ρ-maximum volume in A if

$$\rho \operatorname{vol}(B) \ge \operatorname{vol}(B') \tag{3}$$

for all $k \times k$ submatrices B' of A.

(ii) Let B be formed by k columns (rows) of A. $\operatorname{vol}(B) (\neq 0)$ is said to be a local ρ-maximum volume in A if (3) holds for any B' that is obtained by replacing one column (row) of B by a column (row) of A which is not in B.

(iii) Let B be a $k \times k$ submatrix ($k < \min(m,n)$) of A. $\operatorname{vol}(B) (\neq 0)$ is said to be a local ρ-maximum volume in A if it is a global ρ-maximum volume in all $(k+1)\times(k+1)$ submatrices of A which contain B.

The important concept in the theory of rank revealing factorizations is the local maximum volume. The definition 2.2(ii) is from [12] and 2.2(iii) is the natural extension to square submatrices of any dimension. It is equivalent to saying that A_{11} has local ρ-maximum volume in (2) if the volume of the $(1,1)$ block cannot be increased by more than a factor ρ by interchanging two columns and/or two rows.

Finding a submatrix of local maximum volume will make use of column and row exchanges. The following lemmas provide fomulas for the change of volume when a column and/or row is replaced in a square nonsingular matrix.

Lemma 2.3. *Let A_{11} be $k \times k$ nonsingular and A_{11}' be obtained by replacing column j by the vector $\boldsymbol{b}$. Then*

$$\frac{\operatorname{vol}(A_{11}')}{\operatorname{vol}(A_{11})} = |A_{11}^{-1}\boldsymbol{b}|_j.$$

In particular, A_{11} in (2) *has local ρ-maximum volume in its block row and block column if and only if $\left\|A_{11}^{-1}A_{12}\right\|_{\max} \leq \rho$ and $\left\|A_{21}A_{11}^{-1}\right\|_{\max} \leq \rho$, respectively.*

Proof. Follows immediately from Cramer's rule, by which the column replacement changes $\det(A_{11})$ by the factor $\left(A_{11}^{-1}\boldsymbol{b}\right)_j$. □

Lemma 2.4. *Let $\hat{A}$ be square and nonsingular and B be obtained by removing row i and column j. Then*

$$\frac{\operatorname{vol}(B)}{\operatorname{vol}(\hat{A})} = |\hat{A}^{-1}|_{ji}.$$

In particular, B has ρ-maximum volume in $\hat{A}$ if and only if $\rho|\hat{A}^{-1}|_{ji} \geq \left\|\hat{A}^{-1}\right\|_{\max}$.

Proof. By Cramer's rule

$$(\hat{A}^{-1})_{ji} = (\hat{A}^{-1}\boldsymbol{e}_i)_j = \frac{\det(\hat{A} - \hat{A}\boldsymbol{e}_j\boldsymbol{e}_j^T + \boldsymbol{e}_i\boldsymbol{e}_j^T)}{\det(\hat{A})}.$$

Because the matrix whose determinant is taken in the numerator has unit column $\boldsymbol{e}_i$ in position j, by Laplace's formula

$$\det(\hat{A} - \hat{A}\boldsymbol{e}_j\boldsymbol{e}_j^T + \boldsymbol{e}_i\boldsymbol{e}_j^T) = (-1)^{i+j}\det(B).$$

Substituting into the previous expression and taking absolute values completes the proof. □

Lemma 2.5. *Let A_{11} be $k \times k$ nonsingular and*

$$\hat{A} = \begin{bmatrix} A_{11} & \boldsymbol{b} \\ \boldsymbol{c}^T & \alpha \end{bmatrix}. \tag{4}$$

Let $\gamma = \hat{A}/A_{11}$ and A_{11}'' be the leading $k \times k$ block of $\hat{A}$ after interchanging columns $k+1$ and j $(1 \leq j \leq k)$ and rows $k+1$ and i $(1 \leq i \leq k)$. Then

$$\frac{\operatorname{vol}(A_{11}'')}{\operatorname{vol}(A_{11})} = |\gamma(A_{11}^{-1})_{ji} + (A_{11}^{-1}\boldsymbol{b})_j(A_{11}^{-T}\boldsymbol{c})_i|. \tag{5}$$

Proof. First, consider that $\hat{A}$ is singular, in which case $\operatorname{rank}(\hat{A}) = k$ and $\gamma = 0$. If $|A_{11}^{-1}\boldsymbol{b}|_j = 0$, then the first k columns of $\hat{A}$ after the interchanges have rank $k-1$. Hence A_{11}'' must be singular and both sides of (5) are zero. Otherwise let A_{11}' be obtained from A_{11} by replacing column j by the vector $\boldsymbol{b}$. Then, by Lemma 2.3,

$$\operatorname{vol}(A_{11}') = \operatorname{vol}(A_{11})|A_{11}^{-1}\boldsymbol{b}|_j.$$

Let $\boldsymbol{c}'$ be obtained from $\boldsymbol{c}$ be replacing the j-th entry by α. Because $\hat{A}$ is singular,

$$(A_{11}')^{-T}\boldsymbol{c}' = A_{11}^{-T}\boldsymbol{c}.$$

Therefore, by Lemma 2.3,

$$\begin{aligned}\operatorname{vol}(A_{11}'') &= \operatorname{vol}(A_{11}')|(A_{11}')^{-T}\boldsymbol{c}'|_i \\ &= \operatorname{vol}(A_{11})|A_{11}^{-1}\boldsymbol{b}|_j|A_{11}^{-T}\boldsymbol{c}|_i.\end{aligned}$$

Second, consider that $\hat{A}$ is nonsingular, in which case $\gamma \neq 0$. Then

$$\hat{A}^{-1} = \begin{bmatrix} H & \boldsymbol{f} \\ \boldsymbol{g}^T & \gamma^{-1} \end{bmatrix}, \tag{6}$$

where

$$\begin{aligned} \boldsymbol{f} &= -\gamma^{-1} A_{11}^{-1} \boldsymbol{b}, \\ \boldsymbol{g} &= -\gamma^{-1} A_{11}^{-T} \boldsymbol{c}, \\ H &= A_{11}^{-1} + \gamma \boldsymbol{f} \boldsymbol{g}^T. \end{aligned}$$

It follows from Lemma 2.4 that

$$\begin{aligned} \operatorname{vol}(A_{11}) &= |\gamma^{-1}| \operatorname{vol}(\hat{A}), \\ \operatorname{vol}(A''_{11}) &= |H_{ji}| \operatorname{vol}(\hat{A}). \end{aligned}$$

Therefore

$$\begin{aligned} \frac{\operatorname{vol}(A''_{11})}{\operatorname{vol}(A_{11})} &= \frac{|(A_{11}^{-1})_{ji} + \gamma^{-1}(A_{11}^{-1}\boldsymbol{b})_j (A_{11}^{-T}\boldsymbol{c})_i|}{|\gamma^{-1}|} \\ &= |\gamma (A_{11}^{-1})_{ji} + (A_{11}^{-1}\boldsymbol{b})_j (A_{11}^{-T}\boldsymbol{c})_i|. \end{aligned}$$

□

3 Rank Revealing Algorithm

In this section we present the algorithm for selecting the submatrix A_{11} and prove that the dimension of A_{11} reveals the numerical rank of A. Instead of selecting the row and column subsets directly, the algorithm selects a basis matrix of $\mathbb{A} = \begin{bmatrix} A & \beta I_m \end{bmatrix}$ for a given $\beta > 0$. The columns of A and βI_m in $\mathbb{A}$ are termed "structural" and "logical", respectively. Letting $\mathcal{B}$ and $\mathcal{N}$ be a basic-nonbasic partitioning of the columns of $\mathbb{A}$, we can partition $\mathbb{A}_{\mathcal{B}}$ and $\mathbb{A}_{\mathcal{N}}$ into

$$\mathbb{A}_{\mathcal{B}} = \begin{bmatrix} A_{11} & 0 \\ A_{21} & \beta I_{m-k} \end{bmatrix}, \quad \mathbb{A}_{\mathcal{N}} = \begin{bmatrix} A_{12} & \beta I_k \\ A_{22} & 0 \end{bmatrix}, \tag{7}$$

where the rightmost $m-k$ and k columns are logical. (The rows of $\mathbb{A}$ and the indices in $\mathcal{B}$ and $\mathcal{N}$ can always be permuted to obtain that form.) The partitioning uniquely determines A_{11}. Therefore any basis for $\mathbb{A}$ determines a square nonsingular A_{11}.

To obtain A_{11} with the desired properties, it will turn out that $\mathbb{A}_{\mathcal{B}}$ must have local ρ-maximum volume in $\mathbb{A}$. An algorithm for finding a basis matrix of local maximum volume is given in [7]. Algorithm 1 is a generic version that leaves some flexibility to the implementation by not specifying how to choose (p, q) in line 4 in case there is more than one candidate. In particular, it is not necessary to scan the entire matrix $\mathbb{A}_{\mathcal{B}}^{-1}\mathbb{A}_{\mathcal{N}}$ in every iteration or even to compute it explicitly.

Lemma 3.1. *Algorithm 1 terminates in a finite number of iterations. The resulting A_{11} has local $(2\rho^2)$-maximum volume in A and*

$$\|A/A_{11}\|_{\max} \leq \rho\beta, \quad \left\|A_{11}^{-1}\right\|_{\max} \leq \rho\beta^{-1}. \tag{8}$$

Algorithm 1

Input: $A \in \mathbb{R}^{m\times n}$, $\rho \geq 1$, $\beta > 0$.

Output: Square submatrix A_{11} of local $(2\rho^2)$-maximum volume in A.

1: Build $\mathbb{A} = \begin{bmatrix} A & \beta I_m \end{bmatrix}$.
2: Initialize $\mathcal{B} = \{n+1, \ldots, n+m\}$, $\mathcal{N} = \{1, \ldots, n\}$.
3: **while** $\left\|\mathbb{A}_{\mathcal{B}}^{-1}\mathbb{A}_{\mathcal{N}}\right\|_{\max} > \rho$ **do**
4: Choose (p, q) such that $|\mathbb{A}_{\mathcal{B}}^{-1}\mathbb{A}_{\mathcal{N}}|_{pq} > \rho$.
5: Exchange indices $\mathcal{B}_p$ and $\mathcal{N}_q$ between $\mathcal{B}$ and $\mathcal{N}$.
6: **end while**
7: Build A_{11} from (7).

Proof. Each basis update in Algorithm 1 increases the volume of $\mathbb{A}_{\mathcal{B}}$ by a factor greater than 1. Therefore a basis cannot repeat and the algorithm terminates in a finite number of iterations. When the algorithm terminates, all entries of

$$\mathbb{A}_{\mathcal{B}}^{-1}\mathbb{A}_{\mathcal{N}} = \begin{bmatrix} A_{11}^{-1}A_{12} & \beta A_{11}^{-1} \\ \beta^{-1}A/A_{11} & -A_{21}A_{11}^{-1} \end{bmatrix} \tag{9}$$

are bounded by ρ in absolute value. This means that A_{11} has local ρ-maximum volume in its block row and block column, and $\|A/A_{11}\|_{\max} \leq \rho\beta$ and $\left\|A_{11}^{-1}\right\|_{\max} \leq \rho\beta^{-1}$. When A_{11} has dimension $\min(m, n)$, then A_{11} has local ρ-maximum volume in A. Otherwise consider any submatrix of A of the form (4). The right-hand side in (5) is bounded by

$$|\gamma(A_{11}^{-1})_{ji} + (A_{11}^{-1}\boldsymbol{b})_j(A_{11}^{-T}\boldsymbol{c})_i| \leq \rho\beta\rho\beta^{-1} + \rho\rho = 2\rho^2.$$

Therefore A_{11} has local $(2\rho^2)$-maximum volume in A. □

The following lemma proves that for a certain choice of β the dimension of A_{11} reveals the numerical rank of A.

Lemma 3.2. *Let $\varepsilon > 0$ and $\rho \geq 1$ be given parameters. Let A_{11} be the $r \times r$ submatrix determined by Algorithm 1 for $\beta = \min(m, n)\varepsilon\rho$. Then r is the numerical rank of A under tolerance ε.*

Proof. We need to show that $\sigma_r(A) \geq \varepsilon$ and $\sigma_{r+1}(A) = O(\varepsilon)$. For the first part, the interlacing property of the singular values [6, Corollary 8.6.3] implies that $\sigma_r(A_{11}) \leq \sigma_r(A)$ for any $r \times r$ submatrix A_{11} of A. Therefore

$$\frac{1}{\sigma_r(A)} \leq \frac{1}{\sigma_{\min}(A_{11})} = \left\|A_{11}^{-1}\right\|_2 \leq r\left\|A_{11}^{-1}\right\|_{\max} \leq r\rho\beta^{-1} \leq \frac{1}{\varepsilon},$$

so that $\sigma_r(A) \geq \varepsilon$.

The second part is trivial if $r = \min(m, n)$. Otherwise we use from [12, Theorem 2.7] that for any nonsingular $r \times r$ submatrix A_{11} of A, $\sigma_{r+1}(A) \leq \|A/A_{11}\|_2$. Therefore

$$\begin{aligned} \sigma_{r+1}(A) &\leq \|A/A_{11}\|_2 \leq \|A/A_{11}\|_{\max}\sqrt{(m-r)(n-r)} \leq \beta\rho\sqrt{(m-r)(n-r)} \\ &= \varepsilon\rho^2\min(m, n)\sqrt{(m-r)(n-r)}. \end{aligned}$$

It follows that $\sigma_{r+1}(A)$ is bounded in terms of ε for fixed dimension of A. □

In contrast to the singular value decomposition, Algorithm 1 cannot determine r such that $\sigma_r(A) \geq \varepsilon > \sigma_{r+1}(A)$. It can only guarantee the first inequality and a bound on

$\sigma_{r+1}(A)$ in terms of ε and the dimension of A. In our definition this is sufficient for a rank revealing factorization. Notice from (8) how β balances between keeping the inverse of A_{11} bounded and getting the Schur complement small. The first condition achieves that $r \geq \operatorname{rank}(A)$ and the second condition that $r \leq \operatorname{rank}(A)$ in the numerical sense. Including ρ in the definition of β in Lemma 3.2 guarantees that $\sigma_r(A) \geq \varepsilon$ at the cost that the bound on $\sigma_{r+1}(A)$ grows with ρ^2.

The following lemma and corollary provide a bound on the iteration count of Algorithm 1 in the case that $\rho > 1$ and that entries of maximum absolute value in $\mathbb{A}_{\mathcal{B}}^{-1}\mathbb{A}_{\mathcal{N}}$ are chosen as pivots.

Lemma 3.3. *Suppose that (p, q) in line 4 of Algorithm 1 is chosen such that $|\mathbb{A}_{\mathcal{B}}^{-1}\mathbb{A}_{\mathcal{N}}|_{pq} = \left\|\mathbb{A}_{\mathcal{B}}^{-1}\mathbb{A}_{\mathcal{N}}\right\|_{\max}$. Then after $l \geq 1$ exchanges*

$$\operatorname{vol}(A_{11}) \geq \rho^{(l-k)} \frac{\sigma_1(A)\cdots\sigma_k(A)}{\sqrt{mn}\cdots\sqrt{(m-k+1)(n-k+1)}}, \tag{10}$$

where A_{11} is the submatrix defined by (7) *for the current basis and k denotes its dimension.*

Proof. The proof is by induction over l. After one exchange A_{11} is a 1×1 matrix consisting of an entry of A of maximum absolute value. Hence

$$\operatorname{vol}(A_{11}) = \|A\|_{\max} \geq \frac{\|A\|_2}{\sqrt{mn}} = \rho^0 \frac{\|A\|_2}{\sqrt{mn}} = \rho^{(l-k)} \frac{\sigma_1(A)}{\sqrt{mn}},$$

so that (10) holds for $l = 1$.

Now assume that (10) holds after $1, \ldots, l$ exchanges and consider exchange $l+1$. Denote $(\mathcal{B}, \mathcal{N}, A_{11})$ the basis and submatrix before the exchange, and $(\mathcal{B}', \mathcal{N}', A'_{11})$ the same quantities after swapping $\mathcal{B}_p$ and $\mathcal{N}_q$. Let k and k' be the dimensions of A_{11} and A'_{11}, respectively. We have to distinguish three cases:

(i) Either $\mathcal{B}_p$ and $\mathcal{N}_q$ are both structural or both logical. Then the dimension of A_{11} remains unchanged, so that

$$\operatorname{vol}(A'_{11}) = \frac{\operatorname{vol}(\mathbb{A}_{\mathcal{B}'})}{\beta^{(m-k)}} > \frac{\rho \operatorname{vol}(\mathbb{A}_{\mathcal{B}})}{\beta^{(m-k)}} = \rho \operatorname{vol}(A_{11}).$$

Combining this with the induction assumption for l shows that (10) holds for $l+1$.

(ii) $\mathcal{B}_p$ is logical and $\mathcal{N}_q$ is structural. Then $k' = k+1$ and A_{11} is a submatrix of A'_{11}. From the determinant property of the Schur complement [2] we have

$$\operatorname{vol}(A'_{11}) = \operatorname{vol}(A_{11}) \operatorname{vol}(A'_{11}/A_{11}).$$

Because (p, q) was chosen as a maximum absolute entry of (9) and because A'_{11}/A_{11} is an entry of A/A_{11}, we must have $|A'_{11}/A_{11}| = \|A/A_{11}\|_{\max}$. Consequently

$$|A'_{11}/A_{11}| = \|A/A_{11}\|_{\max} \geq \frac{\|A/A_{11}\|_2}{\sqrt{(m-k)(n-k)}} \geq \frac{\sigma_{k+1}(A)}{\sqrt{(m-k)(n-k)}},$$

where the last inequality is proved in [12, Theorem 2.7]. Combining the two previous expressions and using the induction assumption gives

$$\begin{aligned}
\operatorname{vol}(A'_{11}) &= \operatorname{vol}(A_{11}) |A'_{11}/A_{11}| \\
&\geq \operatorname{vol}(A_{11}) \frac{\sigma_{k+1}(A)}{\sqrt{(m-k)(n-k)}} \\
&\geq \rho^{(l-k)} \frac{\sigma_1(A)\cdots\sigma_k(A)}{\sqrt{mn}\cdots\sqrt{(m-k+1)(n-k+1)}} \frac{\sigma_{k+1}(A)}{\sqrt{(m-k)(n-k)}} \\
&= \rho^{(l+1-k')} \frac{\sigma_1(A)\cdots\sigma_{k+1}(A)}{\sqrt{mn}\cdots\sqrt{(m-k)(n-k)}}.
\end{aligned}$$

(iii) $\mathcal{B}_p$ is structural and $\mathcal{N}_q$ is logical. Then $k' = k - 1$. Let $l'' < l$ be maximum such that the submatrix A''_{11} after l'' exchanges had dimension $k - 1$, and denote $\mathcal{B}''$ the basis after l'' exchanges. Because each of the exchanges $l'' + 1, \ldots, l + 1$ increased the volume of the basis matrix by at least a factor ρ, we have

$$\mathrm{vol}(A'_{11}) = \frac{\mathrm{vol}(\mathbb{A}_{\mathcal{B}'})}{\beta^{(m-k')}} \geq \frac{\rho^{(l+1-l'')}\,\mathrm{vol}(\mathbb{A}_{\mathcal{B}''})}{\beta^{(m-k')}} = \rho^{(l+1-l'')}\,\mathrm{vol}(A''_{11}).$$

Combining this with the induction assumption for l'' gives

$$\begin{aligned}\mathrm{vol}(A'_{11}) &\geq \rho^{(l+1-l'')}\,\mathrm{vol}(A''_{11}) \\ &\geq \rho^{(l+1-l'')}\rho^{(l''-k')}\frac{\sigma_1(A)\cdots\sigma_{k'}(A)}{\sqrt{mn}\cdots\sqrt{(m-k'+1)(n-k'+1)}} \\ &= \rho^{(l+1-k')}\frac{\sigma_1(A)\cdots\sigma_{k'}(A)}{\sqrt{mn}\cdots\sqrt{(m-k'+1)(n-k'+1)}}.\end{aligned}$$

In each of the cases (i)–(iii) (10) holds after $l + 1$ exchanges, which completes the proof. □

Corollary 3.4. *Suppose that $\rho > 1$ and that (p, q) in line 4 of Algorithm 1 is chosen as in Lemma 3.3. Then Algorithm 1 terminates after at most $r + \lfloor r \log_\rho \sqrt{mn} \rfloor$ iterations, where $r \leq \min(m, n)$ is the dimension of the final submatrix A_{11}.*

Proof. Let l be the iteration count of Algorithm 1. By the interlacing property of the singular values [6, Corollary 8.6.3] we have $\mathrm{vol}(A_{11}) \leq \sigma_1(A)\cdots\sigma_r(A)$. It follows from Lemma 3.3 that

$$\rho^{(l-r)} \leq \sqrt{mn}\cdots\sqrt{(m-r+1)(n-r+1)},$$

which implies that

$$l - r \leq \log_\rho\left(\sqrt{mn}\cdots\sqrt{(m-r+1)(n-r+1)}\right) \leq \log_\rho\left(\sqrt{mn}^{\,r}\right) = r\log_\rho(\sqrt{mn}).$$

Because the left-hand side is integer, we can round the right-hand side of the inequality toward zero, which yields the claim. □

A similar bound was proved in [10] for a particular variant of a maximum volume algorithm. Whether the iteration count remains bounded polynomially in mn for $\rho = 1$ is an open question. The numerical experiments in Section 6 will show that in practice the actual iteration count is much lower than the bound predicts.

4 Bounds on $\sigma_{\min}(A_{11})$ and $\|A/A_{11}\|_2$

The discussion so far has shown that A_{11} that satisfies (8) reveals the numerical rank of A. It remains to be shown that the minimum singular value of A_{11} is close to $\sigma_r(A)$ for A_{11} obtained from Algorithm 1. This section derives bounds on $\sigma_{\min}(A_{11})$ and $\|A/A_{11}\|_2$ in terms of the singular values of A that hold for any local maximum volume submatrix. More specifically, the following theorem is proved.

Theorem 4.1. *Let A_{11} be $k \times k$ nonsingular and have local $(2\rho^2)$-maximum volume in A. Then*

$$\sigma_k(A) \geq \sigma_{\min}(A_{11}) \geq \frac{1}{2\rho^2 k\sqrt{(m-k+1)(n-k+1)}}\sigma_k(A), \tag{11}$$

$$\sigma_{k+1}(A) \leq \|A/A_{11}\|_2 \leq 2\rho^2(k+1)\sqrt{(m-k)(n-k)}\sigma_{k+1}(A). \tag{12}$$

The first inequalities in (11) and (12) hold true for any $k \times k$ submatrix of A, whereas the second inequalities require the maximum volume property. (12) is proved in [8] under the assumption that A_{11} has global ρ-maximum volume in A. Interestingly, the proof given there goes through unchanged if A_{11} has local ρ-maximum volume as defined in this paper. The proof is given here for completeness. The proof for (11) is new to the authors.

Lemma 4.2. *Let $A \in \mathbb{R}^{m\times n}$ and A_{11} be a nonsingular $k \times k$ submatrix ($k < \min(m,n)$) of local ρ-maximum volume. Then*

$$\|A/A_{11}\|_{\max} \le \rho(k+1)\sigma_{k+1}(A).$$

Proof (from [8, Theorem 2.1]). Consider any $(k+1)\times(k+1)$ submatrix of A of the form

$$\hat{A} = \begin{bmatrix} A_{11} & \boldsymbol{b} \\ \boldsymbol{c}^T & \alpha \end{bmatrix}.$$

Then $\gamma = \alpha - \boldsymbol{c}^T A_{11}^{-1}\boldsymbol{b}$ is an entry of A/A_{11} and each entry of A/A_{11} has this form for a particular $\hat{A}$. Therefore it suffices to show that $|\gamma| \le \rho(k+1)\sigma_{k+1}(A)$.

If $\hat{A}$ is singular, then $\gamma = 0$ and the claim is trivial. Otherwise, because A_{11} has ρ-maximum volume in $\hat{A}$, by Lemma 2.4 and (6),

$$\rho|\gamma^{-1}| \ge \left\|\hat{A}^{-1}\right\|_{\max}.$$

It follows that

$$|\gamma| \le \rho\frac{1}{\left\|\hat{A}^{-1}\right\|_{\max}} \le \rho\frac{k+1}{\left\|\hat{A}^{-1}\right\|_{2}} = \rho(k+1)\sigma_{k+1}(\hat{A}) \le \rho(k+1)\sigma_{k+1}(A),$$

where the last inequality comes from the interlacing property of singular values [6, Corollary 8.6.3]. □

Corollary 4.3. *Let $A \in \mathbb{R}^{m\times n}$ and A_{11} be a nonsingular $k \times k$ submatrix ($k < \min(m,n)$) of local ρ-maximum volume. Then*

$$\sigma_{k+1}(A) \le \|A/A_{11}\|_2 \le \rho(k+1)\sqrt{(m-k)(n-k)}\sigma_{k+1}(A).$$

Proof. The first inequality is proved in [12, Theorem 2.7]. The second inequality follows from Lemma 4.2. □

Lemma 4.4. *Let $A \in \mathbb{R}^{m\times n}$ and A_{11} be a nonsingular $k \times k$ submatrix of local ρ-maximum volume. Then*

$$\sigma_k(A) \le \rho k\sqrt{(m-k+1)(n-k+1)}\sigma_k(A_{11}).$$

Proof. If $k = 1$, then A_{11} is scalar and because of local ρ-maximum volume it satisfies $\rho|A_{11}| \ge \|A\|_{\max}$. Therefore

$$\sigma_1(A) \le \sqrt{mn}\,\|A\|_{\max} \le \sqrt{mn}\rho|A_{11}| = \rho\sqrt{mn}\sigma_1(A_{11}).$$

If $k > 1$, let B be a $(k-1)\times(k-1)$ submatrix of A_{11} with maximum volume in A_{11}. In particular B is nonsingular. Consider any $k \times k$ submatrix of A of the form

$$A_{11}'' = \begin{bmatrix} B & \boldsymbol{b} \\ \boldsymbol{c}^T & \alpha \end{bmatrix}.$$

Because A''_{11} differs from A_{11} by at most one row and one column, and because A_{11} has local ρ-maximum volume in A,

$$\rho \operatorname{vol}(A_{11}) \geq \operatorname{vol}(A''_{11}).$$

From the determinant property of the Schur complement [2], $\det(A_{11}) = \det(B)\det(A_{11}/B)$, it follows for the scalars A_{11}/B and A''_{11}/B that

$$\rho|A_{11}/B| = \rho\frac{\operatorname{vol}(A_{11})}{\operatorname{vol}(B)} \geq \frac{\operatorname{vol}(A''_{11})}{\operatorname{vol}(B)} = |A''_{11}/B|.$$

Because A''_{11}/B is an entry of A/B and each entry of A/B has this form for a particular A''_{11}, it follows that

$$\rho|A_{11}/B| \geq \|A/B\|_{\max}.$$

Therefore

$$\begin{aligned}\sigma_k(A) \leq \|A/B\|_2 &\leq \sqrt{(m-k+1)(n-k+1)}\,\|A/B\|_{\max} \\ &\leq \rho\sqrt{(m-k+1)(n-k+1)}|A_{11}/B| \\ &\leq \rho\sqrt{(m-k+1)(n-k+1)}k\sigma_k(A_{11}),\end{aligned}$$

where the first inequality is from [12, Theorem 2.7] and the last inequality from Lemma 4.2 and the fact that B was chosen to have maximum volume in A_{11}. □

Corollary 4.5. *Let $A \in \mathbb{R}^{m\times n}$ and A_{11} be a nonsingular $k\times k$ submatrix of local ρ-maximum volume. Then*

$$\sigma_k(A) \geq \sigma_{\min}(A_{11}) \geq \frac{1}{\rho k\sqrt{(m-k+1)(n-k+1)}}\sigma_k(A).$$

Proof. The first inequality comes from the interlacing property of singular values [6, Corollary 8.6.3]. The second inequality follows from Lemma 4.4. □

Theorem 4.1 follows from Corollaries 4.5 and 4.3.

5 Comparison to Pan's Method

Pan [12] uses the maximum volume concept in combination with LU factorization to find a submatrix A_{11} that has very similar properties to the submatrix obtained from Algorithm 1. This section compares the two methods regarding their use of the maximum volume property and possible implementations.

Given $A \in \mathbb{R}^{m\times n}$, $k \leq \operatorname{rank}(A)$ and $\rho \geq 1$, Pan's method proceeds in two steps:

(1) It applies Algorithm 2 with $\bar{\rho} = \rho^2$ to A^TA to obtain a principal $k \times k$ submatrix of local ρ^2-maximum volume; i. e. a submatrix of the form $A_{\mathcal{J}}^T A_{\mathcal{J}}$ whose volume cannot be increased by more than a factor ρ^2 by replacing one index in $\mathcal{J}$. The column slice $A_{\mathcal{J}}$ then has ρ-maximum volume in A.

(2) It uses Algorithm 3 to find a $k \times k$ submatrix of local ρ-maximum volume in $A_{\mathcal{J}}$.

Algorithms 2 and 3 are generic versions of Algorithms 1 and 2 in [12] which leave some flexibility to the choice of the pivot element. Lemmas 2.5 and 2.3 prove that the submatrix determined by each algorithm has indeed the desired maximum volume property.

Let us say that the submatrix A_{11} obtained from the above procedure has "normal" ρ-maximum volume in A to distinguish it from a local maximum volume in our definition.

Algorithm 2

Input: $C \in \mathbb{R}^{n\times n}$ symmetric positive semidefinite, $k \leq \operatorname{rank}(C)$, $\bar{\rho} \geq 1$.
Output: Principal $k \times k$ submatrix C_{11} of local $\bar{\rho}$-maximum volume.

1: Compute the partial Cholesky factorization

$$PCP^T = \begin{bmatrix} C_{11} & C_{21}^T \\ C_{21} & C_{22} \end{bmatrix} = \begin{bmatrix} L_{11} & \\ L_{21} & I_{n-k} \end{bmatrix} \begin{bmatrix} L_{11}^T & L_{21}^T \\ & S \end{bmatrix}, \tag{13}$$

where P is a chosen permutation matrix and L_{11} is a $k \times k$ lower triangular matrix with nonzero diagonal entries.

2: **while** there exist $1 \leq p \leq k$ and $1 \leq q \leq n-k$ with $|S_{qq}\left(C_{11}^{-1}\right)_{pp} + \left(C_{21}C_{11}^{-1}\right)_{qp}^2| > \bar{\rho}$ **do**

3: Choose such (p, q) and interchange rows p and $k+q$ of P.

4: Restore the factorization (13) for the updated P.

5: **end while**

Algorithm 3

Input: $A \in \mathbb{R}^{m\times k}$ of rank k, $\rho \geq 1$.
Output: $k \times k$ submatrix A_1 of local ρ-maximum volume.

1: Compute the LU factorization

$$PAQ = \begin{bmatrix} A_1 \\ A_2 \end{bmatrix} Q = \begin{bmatrix} L_1 \\ L_2 \end{bmatrix} U, \tag{14}$$

where P and Q are chosen permutation matrices, L_1 is a $k \times k$ lower triangular matrix with nonzero diagonal entries and U is unit upper triangular.

2: **while** there exist $1 \leq p \leq k$ and $1 \leq q \leq m-k$ with $|\left(A_2 A_1^{-1}\right)_{qp}| > \rho$ **do**

3: Choose such (p, q) and interchange rows p and $k+q$ of P.

4: Restore the factorization (13) for the updated P and a newly chosen Q.

5: **end while**

Theorem 3.8 in [12] proves the following bounds on the singular values of A_{11} and A/A_{11} for $m = n$:

$$\sigma_k(A) \geq \sigma_{\min}(A_{11}) \geq \frac{1}{k(n-k)\rho^2 + 1}\sigma_k(A),$$
$$\sigma_{k+1}(A) \leq \|A/A_{11}\|_2 \leq \left(k(n-k)\rho^2 + 1\right)\sigma_{k+1}(A).$$

These bounds are almost identical to those in Theorem 4.1. Although the setting in [12] assumes the dimension of A_{11} to be given, choosing it dynamically by means of a tolerance β as in Algorithm 1 can be easily incorporated into Algorithm 2. Therefore Pan's method and our method provide the same functionality.

Pan's complexity analysis provides operation counts for updating the matrix factorizations in Algorithms 2 and 3, but does not consider the number of exchanges required. An upper bound can be derived along the lines of Lemma 3.3 and Corollary 3.4, however.

Lemma 5.1. *(i) Let $\bar{\rho} > 1$ in Algorithm 2 and let the partial Cholesky factorization in line 1 be computed with complete pivoting; i. e. in each elimination step the pivot element is chosen to be a diagonal entry of maximum absolute value from the active submatrix. Then the subsequent while-loop terminates after at most $\lfloor k \log_{\bar{\rho}} n \rfloor$ iterations.*

(ii) Let $\rho > 1$ in Algorithm 3 and let the LU factorization in line 1 be computed with complete pivoting; i. e. in each elimination step the pivot element is chosen to be of maximum absolute value from the active submatrix. Then the subsequent while-loop terminates after at most $\lfloor k \log_{\rho} m \rfloor$ iterations.

Proof. For part (i) consider the Cholesky factorization in line 1 of Algorithm 2. Denote $C^{(s)}$ the active submatrix prior to the s-th elimination step ($1 \leq s \leq k$) and let γ_s be the chosen pivot element from the diagonal of $C^{(s)}$. Because $C^{(s)}$ is positive semidefinite, its maximum absolute entry is found on the diagonal and is nonnegative (see [6, Section 4.1]). Hence, by the pivot choice, we have $\gamma_s = \left\|C^{(s)}\right\|_{\max}$. It follows from [12, Theorem 2.7] that

$$\sigma_s(C) \leq \left\|C^{(s)}\right\|_2 \leq \gamma_s(n - s + 1),$$

and by taking the product over s we obtain

$$\frac{\prod_{s=1}^k \sigma_s(C)}{\prod_{s=1}^k \gamma_s} \leq n^k.$$

From the interlacing property of the singular values [6, Corollary 8.6.3] we know that the volume of any $k \times k$ submatrix of C is bounded by $\prod_{s=1}^k \sigma_s(C)$. Because each interchange in the while-loop increases the volume of C_{11} by at least a factor $\bar{\rho}$, and because the volume of the initial C_{11} is $\prod_{s=1}^k \gamma_s$, the number of interchanges is bounded by

$$\left\lfloor \log_{\bar{\rho}} \left(n^k\right) \right\rfloor = \left\lfloor k \log_{\bar{\rho}} n \right\rfloor.$$

The proof of part (ii) is analogous. □

These bounds are similar to those for our algorithm (Corollary 3.4), which chooses the row and column subsets simultaneously. It is not of much practical relevance to compare the algorithms in terms of the derived complexity bounds, however, since (at least in our case) the bounds are far too convervative as the experiments in Section 6 will show.

The important difference between Pan's and our method concerns implementation. For dense matrices both algorithms can be implemented efficiently, but there seems to be no advantage over using rank revealing QR factorization [9] for choosing the column subset, followed by Algorithm 3 for choosing the row subset. For sparse matrices, Algorithm 1 can be implemented by maintaining an LU factorization of $\mathbb{A}_{\mathcal{B}}$ after column changes. This is a common operation in linear programming for which established sparse matrix methods exist (see, for example, [4]). Implementing Algorithm 2 requires maintaining a Cholesky (or LU) factorization of $A_{\mathcal{J}}^T A_{\mathcal{J}}$ after symmetric row and column changes. While this is possible in the sparse case [3], these operations can be much more expensive in terms of memory requirement and computation time than maintaining an LU factorization of $\mathbb{A}_{\mathcal{B}}$. To illustrate, assume that our method and Pan's method yield the same submatrix A_{11}. Then the final iteration of Algorithm 2 requires a Cholesky factorization of $A_{11}^T A_{11} + A_{21}^T A_{21}$, whereas the final iteration of Algorithm 1 requires an LU factorization of $\begin{bmatrix} A_{11} & 0 \\ A_{21} & I \end{bmatrix}$. In the latter case we only need to factorize A_{11}, which typically leads to much sparser factors than operating on the normal matrix.

Initially, the authors suspected that local and normal maximum volume might be the same property, but this conjecture turned out to be false. Two examples are given to prove that neither property implies the other. First, consider

$$A = \begin{bmatrix} 1 & & & 1 \\ & 1 & & 1 \\ & & 1 & 1 \\ 1 & 1 & 1 & \\ 1 & & & \\ & 1 & & \\ & & 1 & \end{bmatrix}$$

and let A_{11} be the leading 3×3 block. It can be computed analytically that the singular values of the submatrix formed by any three columns of A are $\left(\sqrt{5}, \sqrt{2}, \sqrt{2}\right)$, so that the first three columns have local maximum volume in A. From Lemma 2.3 it is obvious that A_{11} has local maximum volume within the first three columns. Hence it has normal maximum volume in A. However, it can be verified from Lemma 2.5 that A_{11} does not have maximum volume in the leading 4×4 block and therefore does not have local maximum volume in A.

For the opposite part consider

$$A = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & -1 & -1 \\ -1 & 1 & -1 \end{bmatrix} \tag{15}$$

and let A_{11} be the leading 2×2 block. It can be verified from Lemma 2.5 that A_{11} has local maximum volume in A. By computing singular values we obtain the volume of the matrix composed of columns 1 and 2 to be $\sqrt{5}$ and the volume of the matrix composed of columns 1 and 3 to be $\sqrt{6}$. Hence the first two columns do not have local maximum volume in A, and neither does A_{11} have normal maximum volume in A.

6 Implementation and Results

We have implemented a simplicial version of Algorithm 1 for dense matrices in C code[1]. By "simplicial" we mean that the implementation does not work on block submatrices and

[1] `http://www.maths.ed.ac.uk/ERGO/LURank`

makes no use of optimized BLAS. It therefore is slower than an optimized singular value decomposition. Our interest is to examine the number of pivot operations required and to verify the reliability of the method. Discussing an optimized implementation is beyond the scope of the paper.

Initially the matrix $W = \begin{bmatrix} A & I_m \end{bmatrix}$ is stored. Logical columns are not explicitly scaled by β to avoid values with very different orders of magnitude in the computation. Instead, multiplications with β and β^{-1} are applied on the fly when logical columns are involved.

In each iteration the algorithm chooses a pivot element in the following order:

(i) If $|W|$ has entries corresponding to block A_{11}^{-1} in (9) that are larger than $\rho\beta^{-1}$, then the maximum such entry is chosen as pivot.

(ii) If $|W|$ has entries corresponding to block $A_{11}^{-1}A_{12}$ or $-A_{21}A_{11}^{-1}$ in (9) that are larger than ρ, then the maximum such entry is chosen as pivot.

(iii) If $|W|$ has entries corresponding to block A/A_{11} in (9) that are larger than $\rho\beta$, then the maximum such entry is chosen as pivot.

If a pivot is found, then its column is transformed into a unit column by applying row operations to W. If none of the cases (i)–(iii) yields a pivot element, the algorithm terminates.

The implementation, in the following called RRGE for "rank revealing Gaussian elimination", has been tested on matrices from the San Jose State University Singular Matrix Database [5]. We used the 327 matrices (as of January 2018) for which $\min(m,n) \leq 1000$. The matrices were transposed if necessary so that $m \leq n$. For comparison a singular value decomposition (SVD) of each matrix A was computed and the numerical rank of A was determined as the largest index s such that

$$\sigma_s(A) \geq \max(m,n)\varepsilon_{\text{mach}}\sigma_1(A),$$

where $\varepsilon_{\text{mach}} \approx 2 \cdot 10^{-16}$ is the relative machine precision. This is the criterion used by the MATLAB `cond` function. All matrices in the test set were rank deficient by this definition. For RRGE β was chosen comparably as

$$\beta = \max(m,n)\varepsilon_{\text{mach}} \|A\|_{\max}$$

and ρ was initially set to 2.0. Denote r the dimension of the final submatrix A_{11}, which is the estimated rank of A.

For 56 matrices the numerical ranks determined by SVD and RRGE differed. This is legitimate if there is no unique large gap in the spectrum of A. To verify that the rank determined by RRGE is acceptable with respect to the singular values of A, Figure 1 shows the ratios $\sigma_r(A)/\sigma_s(A)$ and $\sigma_{r+1}(A)/\sigma_{s+1}(A)$ for those matrices where $r \neq s$. Because $\sigma_r(A)/\sigma_s(A) \leq 3.0$, RRGE certainly does not underestimate the rank of A. Furthermore, because $\sigma_r(A)/\sigma_s(A) \approx \sigma_{r+1}(A)/\sigma_{s+1}(A)$, there cannot be a clear gap of the form $\sigma_s(A) \gg \sigma_{s+1}(A) \approx \sigma_m(A)$ if $r > s$. For example, if $r = s+1$, then $\sigma_s(A)$, $\sigma_{s+1}(A)$ and $\sigma_{s+2}(A)$ must be equally apart, so that determining the rank of A as $s+1$ is legitimate. Hence it can be concluded that with the above choice of β, the rank computed by RRGE is in accordance with the spectrum of A for all matrices in our test set.

Table 1 categorizes the 327 matrices into buckets by means of the number of pivot operations of RRGE. Because our implementation started from the all logical basis, a minimum of r pivots was required. For $\rho = 2.0$ the number of pivots was almost always within 5% of the optimum. In this case the computational cost of RRGE was roughly twice the cost of an LU factorization of $\mathbb{A}$ with complete pivoting. For $\rho = 1.1$ the number of pivots increased significantly on many matrices. However, we have not found a relevant increase in the ratios $\sigma_r(A_{11})/\sigma_r(A)$, meaning that the quality of the submatrix A_{11} did not improve. Hence there seems to be no advantage of choosing ρ particularly close to 1.0 in practice.

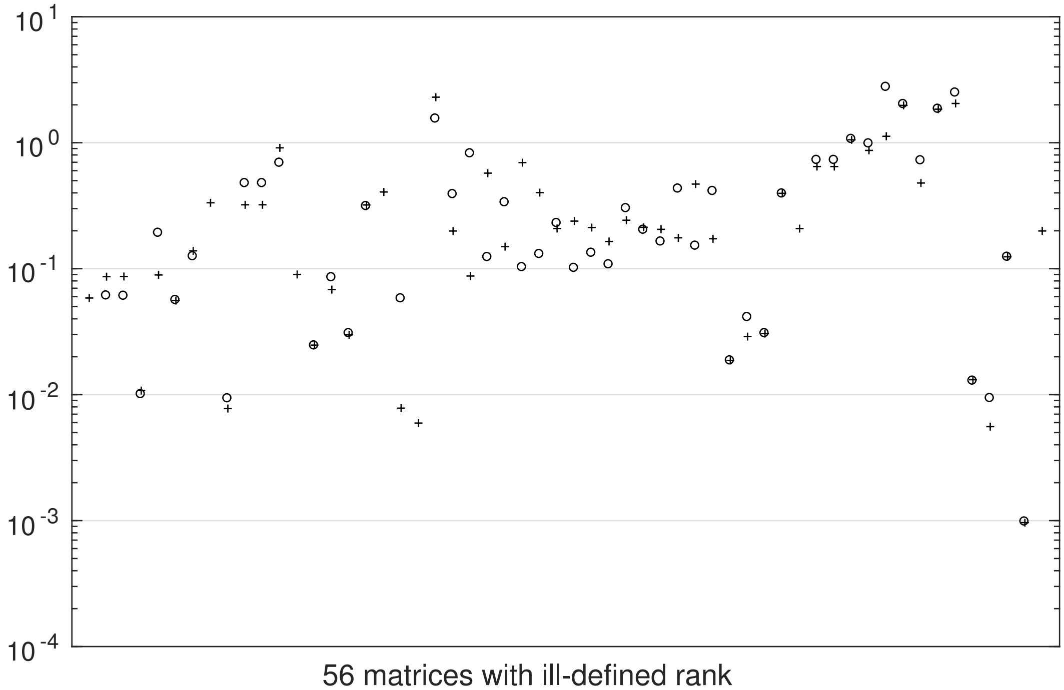

Figure 1: Ratios $\sigma_r(A)/\sigma_s(A)$ ("+") and $\sigma_{r+1}(A)/\sigma_{s+1}(A)$ ("o") for matrices with $r \neq s$. The "o" marker is missing when $r = m$ (7 matrices).

pivots/r	$\rho = 2.0$	$\rho = 1.1$
$[1.00, 1.05)$	325	159
$[1.05, 1.50)$	2	124
$[1.5, 4.0)$	0	37
$[4.0, 5.0)$	0	7

Table 1: Matrices categorized by the number of pivot operations of RRGE.

Despite its simplicity, the implementation proved to be numerically stable in our experiments. One reason for this is the order in which pivot elements were chosen. Note that the dimension of A_{11} increases only in case (iii) above, which occurs only when none of the other three blocks in (9) contains a pivot element. Therefore A_{11} remains as well conditioned as possible throughout, keeping the entries in $|W|$ bounded by $O(\beta^{-1})$. A more efficient implementation of Algorithm 1 needs to avoid searching the entire matrix W after each exchange, so that the above pivot rule is not applicable. In particular for sparse matrices we can only compute one column or one row of $\mathbb{A}_{\mathcal{B}}^{-1}\mathbb{A}_{\mathcal{N}}$ at a time and need to choose a pivot within that column or row. Whether the computations can still be made numerically stable remains to be investigated.

7 Conclusions

We have presented an algorithm for revealing the numerical rank of A by Gaussian elimination on the matrix $\begin{bmatrix} A & \beta I_m \end{bmatrix}$. The bounds on the revealed singular values are very similar to those given in [12], but our algorithm does not make use of the normal matrix. A prototype implementation has shown that the number of pivot operations required in practice is only slightly larger than the rank of A. Because the algorithm allows some flexibility in choosing pivot elements, it can be implemented with blocked memory access to achieve high floating point performance on dense matrices. Developing a rank revealing factorization for sparse matrices based on the results from this paper is a topic for further research.

References

[1] A. Ben-Israel. A volume associated with $m \times n$ matrices. *Linear Algebra Appl.*, 167:87–111, 1992.

[2] R. W. Cottle. Manifestations of the Schur complement. *Linear Algebra Appl.*, 8:189–211, 1974.

[3] T. A. Davis and W. W. Hager. Row modifications of a sparse Cholesky factorization. *SIAM J. Matrix Anal. Appl.*, 26(3):621–639, 2005.

[4] J. J. H. Forrest and J. A. Tomlin. Updated triangular factors of the basis to maintain sparsity in the product form simplex method. *Math. Programming*, 2:263–278, 1972.

[5] L. Foster. San Jose State University singular matrix database. `http://www.math.sjsu.edu/singular/matrices/`. Accessed: Jan 17, 2018.

[6] G. H. Golub and C. F. Van Loan. *Matrix computations*. Johns Hopkins Studies in the Mathematical Sciences. Johns Hopkins University Press, Baltimore, MD, third edition, 1996.

[7] S. A. Goreinov, I. V. Oseledets, D. V. Savostyanov, E. E. Tyrtyshnikov, and N. L. Zamarashkin. How to find a good submatrix. In *Matrix methods: theory, algorithms and applications*, pages 247–256. World Sci. Publ., Hackensack, NJ, 2010.

[8] S. A. Goreinov and E. E. Tyrtyshnikov. The maximal-volume concept in approximation by low-rank matrices. In *Structured matrices in mathematics, computer science, and engineering, I (Boulder, CO, 1999)*, volume 280 of *Contemp. Math.*, pages 47–51. Amer. Math. Soc., Providence, RI, 2001.

[9] M. Gu and S. C. Eisenstat. Efficient algorithms for computing a strong rank-revealing QR factorization. *SIAM J. Sci. Comput.*, 17(4):848–869, 1996.

[10] D. E. Knuth. Semioptimal bases for linear dependencies. *Linear and Multilinear Algebra*, 17(1):1–4, 1985.

[11] J. Nocedal and S. J. Wright. *Numerical optimization.* Springer Series in Operations Research and Financial Engineering. Springer, New York, second edition, 2006.

[12] C.-T. Pan. On the existence and computation of rank-revealing *LU* factorizations. *Linear Algebra Appl.*, 316(1-3):199–222, 2000.

[13] G. Peters and J. H. Wilkinson. The least squares problem and pseudo-inverses. *The Computer Journal*, 13:309–316, 1970.