SIAM J. SCI. STAT. COMPUT.
Vol. 13, No. 3, pp. 727–741, May 1992

007

SOME APPLICATIONS OF THE RANK REVEALING QR FACTORIZATION*

TONY F. CHAN† AND PER CHRISTIAN HANSEN‡

Abstract. The rank revealing QR factorization of a rectangular matrix can sometimes be used as a reliable and efficient computational alternative to the singular value decomposition for problems that involve rank determination. This is illustrated by showing how the rank revealing QR factorization can be used to compute solutions to rank deficient least squares problems, to perform subset selection, to compute matrix approximations of given rank, and to solve total least squares problems.

Key words. rank revealing QR factorization, numerical rank, rank deficient problems, subset selection, matrix approximation, total least squares

AMS(MOS) subject classifications. 65F25, 65F20

1. Introduction.

One of the more intricate problems in numerical linear algebra is to find the numerical rank of a matrix. This computational problem is the heart of many numerical methods, such as subset selection, total least squares, regularization, and matrix approximation. The *singular value decomposition* (SVD) is undoubtedly the most reliable method for computing the numerical rank. A big disadvantage of the SVD is, however, the high computational complexity of the standard SVD algorithm, as compared to a QR factorization, for example, [12, p. 248]. The same is true for SVD algorithms based on Jacobi iteration. SVD algorithms for sparse or structured matrices based on Lanczos iteration are faster, but they have problems with computation of the smallest singular values and, therefore, are not reliable for numerical rank determination.

A number of alternative, less computationally demanding, methods have been proposed. Most of these are based on a QR factorization with column pivoting [4], [18], [19], but the numerical rank computed by these methods is not entirely reliable (see [12, §5.5.7] and [1, §7]), and none of these methods are suited for sparse matrices because of the column pivoting. Similar QR-based methods specially designed for sparse matrices [16], [21], based solely on detecting small elements on the diagonal of the triangular matrix, are not reliable either. Another alternative method, the partial SVD (PSVD) [25], is as reliable as the SVD. However, a complete reduction to bidiagonal form is required, so its complexity is still higher than that of a QR factorization, and it is not well suited to general sparse matrices.

The most promising alternative to the SVD is the *rank revealing* QR *factorization* (RRQR factorization) defined by Chan [6], [7] (we note that similar ideas were proposed independently by Foster [10]). An important existence proof of RRQR factorizations is given in [17]. The RRQR factorization will reveal the numerical rank of any matrix, because it is guaranteed to capture all the small singular values of the matrix by producing reasonably tight upper and lower bounds for these singular

* Received by the editors May 14, 1990; accepted for publication April 10, 1991. This research was supported by NATO Collaborative Research grant 5-2-05/RG900098.

† Department of Mathematics, University of California, 405 Hilgard Ave., Los Angeles, California 90024 (chan@math.ucla.edu). The work of this author was supported by Army Research Office contract DAAL03-88-K-0085, by Department of Energy contract DE-FG-03-87-ER-25037, and by National Science Foundation contract NSF-DMS87-14612.

‡ UNI•C (Danish Computing Center for Research and Education), Building 305, Technical University of Denmark, DK-2800 Lyngby, Denmark (unipch@wuli.uni-c.dk).

values. In addition, the RRQR algorithm produces a set of linearly independent vectors that span a good approximation to the numerical null-space of the matrix. This information is sufficient to solve many problems in numerical linear algebra.

The computational complexity of the RRQR algorithm is only slightly larger than that of the standard QR algorithm, as long as the nullity is small compared to the dimensions of the matrix. A Fortran implementation of the algorithm is now available from ACM TOMS [22]. Moreover, the RRQR factorization of a sparse matrix can be computed efficiently without destroying the sparsity pattern of the matrix, and Bischof and Hansen [1] have demonstrated how to implement the RRQR factorization algorithm with a minimum of column interchanges. Thus, we feel that the time is ripe for using the RRQR factorization in numerical linear algebra.

Chan and Hansen [8] showed how truncated SVD solutions can be computed efficiently by means of the RRQR factorization, and Hansen, Sekhii, and Shibahashi [15] use RRQR factorizations to regularize discrete ill-posed problems. Comon and Golub [9] use RRQR factorizations in conjunction with Lanczos block-bidiagonalization. Bischof and Shroff [2] have shown how to use the null-space information from an RRQR factorization in conjunction with the "signal subspace" approach to parameter estimation in signal processing. In this paper, we illustrate several other important applications of the RRQR factorization in numerical linear algebra.

Stewart [24] has recently proposed a related factorization, namely, a rank revealing complete orthogonal decomposition, which is particularly suited to "subspace tracking" in signal processing. This factorization is computationally more expensive than the RRQR factorization, but *updating of the null-space* from Stewart's factorization is cheaper than updating the null-space from an RRQR factorization.

Our paper is organized as follows. In §2, we summarize the most important properties of the RRQR factorization. Then, we show how the RRQR factorization can be used in rank deficient least squares problems (§3), in subset selection problems (§4), and in matrix approximations (§5). Finally, we demonstrate in §6 how the RRQR factorization can be used to solve total least squares problems with full rank as well as rank deficient coefficient matrices.

We shall use two-norms almost entirely, so we use the abbreviation $\|\cdot\|$ for $\|\cdot\|_2$. The range (column space) of a matrix is denoted by $\mathcal{R}(\cdot)$. Throughout this paper, in addition to our new results, we include a few results already published in other manuscripts. We feel that the present constellation of this material will provide new insight into the applications and practical use of the RRQR factorization.

2. RRQR factorizations. Throughout the paper, we assume that the matrix A has been properly scaled, for example, such that the uncertainties in its elements are roughly of the same size [23]. The numerical rank, or ϵ-*rank*, of A with respect to the tolerance ϵ is defined by

$$k = k(A, \epsilon) \equiv \min_{\|A-B\| \le \epsilon} \operatorname{rank}(B). \tag{1}$$

In other words, the ϵ-rank of A is equal to the number of columns in A that are guaranteed to be linearly independent for any perturbation of A with norm less than or equal to the tolerance ϵ. As a guide to choosing this tolerance, it is customary to let ϵ reflect the uncertainties in A [23]. See also [11] and [12, §2.5.4] for more details.

The most reliable way of computing the ϵ-rank of A is via its SVD. Assume for

simplicity that $A \in \Re^{m \times n}$ with $m \geq n$. Then the SVD of A is

$$A = U \Sigma V^T = \sum_{i=1}^{n} \mathbf{u}_i \, \sigma_i \, \mathbf{v}_i^T, \tag{2}$$

where $U = [\mathbf{u}_1, \cdots, \mathbf{u}_n]$ and $V = [\mathbf{v}_1, \cdots, \mathbf{v}_n]$ are matrices with orthonormal columns, and $\Sigma = \mathrm{diag}(\sigma_1, \cdots, \sigma_n)$ is an $m \times n$ diagonal matrix whose diagonal entries, the *singular values* of A, are ordered such that $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n \geq 0$. From the orthonormality of the columns of U and V it follows that $\|A\mathbf{v}_i\| = \sigma_i$, $i = 1, \cdots, n$. It is straightforward to show that if k is the number of singular values strictly greater than ϵ, i.e., $\sigma_k > \epsilon \geq \sigma_{k+1}$, then k is the ϵ-rank of A as defined in (1). For every singular value $\sigma_i < \epsilon$, the corresponding right singular vector $\mathbf{v}_i$ is a numerical null-vector of A in the sense that $\|A\,\mathbf{v}_i\| \leq \epsilon$. It is therefore natural to define the *numerical null-space* of A as the space spanned by the vectors $\mathbf{v}_{k+1}$ through $\mathbf{v}_n$:

$$\mathcal{N}_k(A) \equiv \mathrm{span}\{\mathbf{v}_{k+1}, \cdots, \mathbf{v}_n\}. \tag{3}$$

We can now define a rank revealing QR factorization of A as a special QR factorization $A\,\Pi = QR$, which is guaranteed to reveal the ϵ-rank k of A in displaying elements in the lower portion of R with magnitude of the order σ_{k+1} or less. An RRQR factorization thus has the form

$$A\,\Pi = QR = Q \begin{pmatrix} R_{11} & R_{12} \\ 0 & R_{22} \end{pmatrix}, \tag{4}$$

where Π is a permutation matrix, Q has orthonormal columns, R_{11} is a $k \times k$ matrix with condition number approximately equal to σ_1/σ_k, $\|R_{22}\|$ is of the order σ_{k+1}, and k is the ϵ-rank of A. Such an RRQR factorization of A is not unique, and different RRQR algorithms may produce different factorizations. The key idea in all RRQR algorithms is, however, the same: first compute any QR factorization of A and then construct Π and Q by building up R_{22} one row at a time, starting from the bottom. Assume that a trailing $(n-i) \times (n-i)$ submatrix with small norm has already been generated. In the next step, the RRQR algorithm then proceeds as follows:

1. Compute the smallest singular value δ_i and the corresponding right null-vector $\mathbf{w}^{(i)} \in \Re^i$ of the leading $i \times i$ submatrix $R^{(i)}$ of R, such that

$$\|\mathbf{w}^{(i)}\| = 1, \qquad \|R^{(i)}\mathbf{w}^{(i)}\| = \delta_i \leq \sigma_i. \tag{5}$$

   The inequality $\delta_i \leq \sigma_i$ follows immediately from the interlacing inequality for singular values [12, Cor. 8.3.3].
2. Find the permutation that permutes the largest element in absolute value of $\mathbf{w}^{(i)}$ to the bottom.
3. Apply this permutation to the columns of $R^{(i)}$ and compute a new QR factorization of this matrix.
4. The (i,i)-element of $R^{(i)}$ is now *guaranteed* to be of the order δ_i.

This process continues until $\delta_i > \epsilon$ and then the ϵ-rank k of A, given by (1), is equal to i. The vectors $\mathbf{w}^{(i)}$ are padded with zeros and gathered in a matrix

$$W = \begin{pmatrix} W_1 \\ W_2 \end{pmatrix}$$

in such a way that $W_2 \in \Re^{(n-k)\times(n-k)}$ is upper triangular. The resulting column permutation matrix Π seeks to make W_2, produced by the RRQR algorithm, as well conditioned as possible (a priori upper bounds for $\|W_2^{-1}\|$, which depend on the particular RRQR algorithm, can be found in [1], [8], [10]). It is very important that the submatrix W_2 be well conditioned, for then we are guaranteed to obtain tight bounds for the singular values of A due to the following theorem.

THEOREM 2.1. *Let $R_{22}^{(i)}$ and $W_2^{(i)}$ denote the lower right $(n-i+1)\times(n-i+1)$ submatrices of R_{22} and W_2, respectively. Also, let δ_i denote the smallest singular value of the leading principal $i \times i$ submatrices of R. Then for $i = k+1, \cdots, n$:*

$$\frac{\sigma_i}{\sqrt{n-i+1}\,\|(W_2^{(i)})^{-1}\|} \le \delta_i \le \sigma_i \le \|R_{22}^{(i)}\| \le \sigma_i \sqrt{n-i+1}\,\|(W_2^{(i)})^{-1}\|. \tag{6}$$

Proof. See [7, Cor. 4.1]. □

Theorem 2.1 shows that the quantities δ_i and $\|R_{22}^{(i)}\|$ provide easily computed lower and upper bounds for the singular values σ_i. Moreover, the outermost bounds in (6) show that if $\|(W_2^{(i)})^{-1}\|$ is not large, then δ_i and $\|R_{22}^{(i)}\|$ are guaranteed to be tight bounds for σ_i. Therefore, the ϵ-rank of A will always be revealed from inspection of the upper and lower bounds for σ_i produced by the RRQR algorithm. In addition, Theorem 2.1 guarantees that $\|R_{22}\|$ is indeed of the order σ_{k+1}.

The matrix W produced during the RRQR algorithm is such an integral part of the RRQR that one may almost consider it a part of the factorization, the reason being that $\mathcal{R}(\Pi W)$ is a good approximation to the numerical null-space $\mathcal{N}_k(A)$. In fact, as shown in [8, Thm. 4.1], the larger the gap between σ_k and σ_{k+1}, the smaller the subspace angle between $\mathcal{R}(\Pi W)$ and $\mathcal{N}_k(A)$, i.e., the better $\mathcal{R}(\Pi W)$ approximates the numerical null-space. If a more accurate basis for $\mathcal{N}_k(A)$ is required, the columns of W can always be improved by a few inverse iterations, as shown in [8].

3. Rank deficient least squares problems. In this section we consider algorithms for solving the linear least squares problem

$$\min \|A\mathbf{x} - \mathbf{b}\|, \qquad A \in \Re^{m\times n}, \tag{7}$$

where the matrix A is very ill conditioned. The usual least squares solution, formally given by $\mathbf{x} = \Pi R^{-1} Q^T \mathbf{b}$, is then of no use because it is extremely sensitive to perturbations of $\mathbf{b}$ and it is usually dominated by highly oscillating contributions from the errors in the right-hand side $\mathbf{b}$. A standard approach to computing a least squares solution that is less sensitive to perturbations is to transform (7) into a nearby problem that is better conditioned. In practice, this process usually corresponds to damping or filtering the contributions to the least squares solution corresponding to the smallest singular values of A [13], [14]. One such approach is the *truncated* SVD (TSVD) method, in which one completely filters out all the small singular values below the ϵ-rank k. The TSVD solution $\mathbf{x}_{TSVD}$ is thus defined as

$$\mathbf{x}_{TSVD} \equiv \sum_{i=1}^{k} \frac{\mathbf{u}_i^T \mathbf{b}}{\sigma_i} \mathbf{v}_i, \qquad k = k(A, \epsilon). \tag{8}$$

The TSVD solution can always be computed from the complete SVD of A, but this is computationally expensive because a great deal of the information provided by the SVD is not used. In fact, only the ϵ-rank and information about the numerical

TABLE 1
Dominating terms of the computational effort.

	excl. RRQR factorization	incl. RRQR factorization
$\mathbf{x}_{TSVD}$	$(2q(n-k)+1)n^2$	$(2m-\frac{2}{3}n+(\frac{7}{2}+2q)(n-k)+1)n^2$
$\mathbf{x}_{TQR}$	$(2(n-k)+1)n^2$	$(2m-\frac{2}{3}n+\frac{11}{2}(n-k)+1)n^2$
$\mathbf{x}_B$	n^2	$(2m-\frac{2}{3}n+\frac{7}{2}(n-k)+1)n^2$

null-space is required. Therefore, the TSVD solution can be computed efficiently by means of the RRQR factorization of A, as shown in [8].

Instead of using the RRQR to compute $\mathbf{x}_{TSVD}$, one may define a least squares solution in terms of the RRQR factorization itself. Here, we shall analyze and compare this approach to the TSVD method. Let the RRQR factorization of A be given by (4). The standard approach [12, §5.5] is to neglect the submatrix R_{22} (which is guaranteed to have a norm of the order σ_{k+1} due to Theorem 2.1) and to then solve this modified problem. In analogy with the TSVD solution, it is natural to define the *truncated* QR (TQR) solution $\mathbf{x}_{TQR}$ as the minimum two-norm least squares solution to the modified problem. To compute $\mathbf{x}_{TQR}$, which involves the pseudoinverse of

$$\begin{pmatrix} R_{11} & R_{12} \\ 0 & 0 \end{pmatrix},$$

it is convenient to use a right orthogonal transformation P to annihilate R_{12} by means of R_{11}, i.e., $(R_{11}, R_{12})P = (\hat{R}_{11}, 0)$. Then the TQR solution is given by

$$\mathbf{x}_{TQR} \equiv \Pi P \begin{pmatrix} \hat{R}_{11}^{-1} & 0 \\ 0 & 0 \end{pmatrix} Q^T \mathbf{b}. \tag{9}$$

Alternatively, one can compute the *basic solution* $\mathbf{x}_B$, defined as the solution to another modified least squares problem where both submatrices R_{12} and R_{22} are neglected. The basic solution is given by

$$\mathbf{x}_B \equiv \Pi \begin{pmatrix} R_{11}^{-1} & 0 \\ 0 & 0 \end{pmatrix} Q^T \mathbf{b}. \tag{10}$$

We list in column one of Table 1 the dominating terms of the computational effort required to compute the three solutions $\mathbf{x}_{TSVD}$, $\mathbf{x}_{TQR}$, and $\mathbf{x}_B$, assuming that the RRQR factorization of A has been computed and that $Q^T\mathbf{b}$ is computed simultaneously with the factorization. The computational effort is measured in flops (a flop is either one addition or one multiplication). The quantity q is the number of inverse iterations used to compute accurate singular subspaces, and usually q is less than 4. The computational effort to compute the RRQR factorization itself depends on the column permutations needed during the computations, but it never exceeds $(2m-\frac{2}{3}n+\frac{7}{2}(n-k))n^2$ flops (provided that the Linpack condition estimator is used in each step). This leads to the total amount of computational effort given in column two. We note that more recent condition estimators, such as Incremental Condition Estimation and related algorithms, may be much more efficient in this context; see [1] for details.

In comparison, we can compute the complexity required by a similar technique for computing $\mathbf{x}_{TSVD}$ based on the PSVD algorithm [25] with accumulation of $Q^T\mathbf{b}$

(PSVD is not suited to computation of $\mathbf{x}_{TQR}$ or $\mathbf{x}_B$). The R-bidiagonalization of $A = U_B B V_B^T$ requires $2(m+n)n^2$ flops. Computation of the $n-k$ smallest singular values and the corresponding left and right singular vectors, using on average two "chases" per singular value, requires $24(n-k)n^2$ flops. Backsubstitution with B and orthogonalization with respect to the left and right null spaces requires $O(n)$ flops. Coordinate transformation with V_B requires $2n^2$ flops. Thus the total is $(2m+2n+24(n-k)+2)n^2$ flops. Therefore, the RRQR algorithm is always less computationally demanding than the PSVD-based algorithm.

For our numerical comparison, it is convenient to define the residual vectors corresponding to the three solutions:

$$\mathbf{r}_i = A\,\mathbf{x}_i - \mathbf{b}, \qquad i = TSVD, TQR, B. \tag{11}$$

Then we have the following results.

THEOREM 3.1. *The* TSVD *and* TQR *solutions are related by*

$$\|\mathbf{x}_{TSVD} - \mathbf{x}_{TQR}\| \le \|R_{22}\|\,\|R_{11}^{-1}\| \left(2\,\|\mathbf{x}_{TSVD}\| + \frac{\|\mathbf{r}_{TSVD}\|}{\sigma_k}\right) \tag{12}$$

and the TQR *and basic solutions are related by*

$$\|\mathbf{x}_{TQR} - \mathbf{x}_B\| \le \frac{1+\sqrt{5}}{2}\,\|R_{11}^{-1}\|^2\,\|R_{12}\|\,\|\mathbf{b}\|. \tag{13}$$

The three residual vectors satisfy

$$\|\mathbf{r}_{TSVD} - \mathbf{r}_{TQR}\| \le \|R_{22}\| \left(\|\mathbf{x}_{TSVD}\| + \frac{\|\mathbf{r}_{TSVD}\|}{\sigma_k}\right) \tag{14}$$

and

$$\|\mathbf{r}_{TQR} - \mathbf{r}_B\| \le \|R_{22}\|\,\|R_{11}^{-1}\|\,\|\mathbf{b}\|. \tag{15}$$

Proof. The TQR solution $\mathbf{x}_{TQR}$ is identical to a truncated SVD solution to the problem

$$\min \left\| Q \begin{pmatrix} R_{11} & R_{12} \\ 0 & 0 \end{pmatrix} \Pi^T \mathbf{x} - \mathbf{b} \right\| = \min \left\| \left(A - Q \begin{pmatrix} 0 & 0 \\ 0 & R_{22} \end{pmatrix} \Pi^T \right) \mathbf{x} - \mathbf{b} \right\|.$$

Thus, we can consider $\mathbf{x}_{TQR}$ a perturbation of $\mathbf{x}_{TSVD}$, with the perturbed matrix given by

$$\tilde{A} = Q \begin{pmatrix} R_{11} & R_{12} \\ 0 & 0 \end{pmatrix} \Pi^T$$

(which has rank k) and with the perturbation matrix given by

$$E = Q \begin{pmatrix} 0 & 0 \\ 0 & R_{22} \end{pmatrix} \Pi^T.$$

Note in particular that $\|E\| = \|R_{22}\|$, that $\|\tilde{A}^+\| = \|\hat{R}_{11}^{-1}\| \le \|R_{11}^{-1}\|$, and that the $(k+1)$th singular value of $\tilde{A}$ is zero. In [13, §3] Hansen derived general perturbation

bounds for TSVD problems. For this special case, these bounds become somewhat simpler:

$$\|\mathbf{x}_{TSVD} - \mathbf{x}_{TQR}\| \leq \|\tilde{A}^+\| \left(\|E\| \, \|\mathbf{x}_{TSVD}\| + \sin\theta_k \, \|\mathbf{r}_{TSVD}\|\right) + \sin\theta_k \, \|\mathbf{x}_{TSVD}\|,$$
$$\|\mathbf{r}_{TSVD} - \mathbf{r}_{TQR}\| \leq \|E\| \, \|\mathbf{x}_{TSVD}\| + \sin\theta_k \, \|\mathbf{r}_{TSVD}\|,$$

where $\sin\theta_k \leq \|E\|/\sigma_k$. Inserting this bound into the above expressions, and using the fact that $\sigma_k^{-1} \leq \delta_k^{-1} = \|R_{11}^{-1}\|$, we obtain (12) and (14). The difference between $\mathbf{x}_{TQR}$ and $\mathbf{x}_B$ satisfies

$$\|\mathbf{x}_{TQR} - \mathbf{x}_B\| \leq \left\| \begin{pmatrix} R_{11} & R_{12} \\ 0 & 0 \end{pmatrix}^+ - \begin{pmatrix} R_{11} & 0 \\ 0 & 0 \end{pmatrix}^+ \right\| \|\mathbf{b}\|.$$

Here, we can consider

$$\begin{pmatrix} 0 & R_{12} \\ 0 & 0 \end{pmatrix}$$

a perturbation of

$$\begin{pmatrix} R_{11} & 0 \\ 0 & 0 \end{pmatrix},$$

and a standard result for perturbed pseudoinverses [3, Thm. 5.3], $\|(A+E)^+ - A^+\| \leq \frac{1+\sqrt{5}}{2} \|A^+\| \, \|(A+E)^+\| \, \|E\|$, then yields (13). Finally, to prove (15), we have

$$\mathbf{r}_B - \mathbf{r}_{TQR} = A\left(\mathbf{x}_{TQR} - \mathbf{x}_B\right) = Q \begin{pmatrix} 0 & 0 \\ R_{22} P_{21} \hat{R}_{11}^{-1} & 0 \end{pmatrix} Q^T \mathbf{b},$$

where P_{21} is the bottom left $(n-k) \times k$ submatrix of the orthogonal matrix P. Taking norms and using $\|\hat{R}_{11}^{-1}\| \leq \|R_{11}^{-1}\|$, we obtain (15). □

Due to Theorem 2.1 we are guaranteed that $\|R_{22}\| \leq \sigma_{k+1}\sqrt{n-k} \, \|W_2^{-1}\|$ when R is computed via a RRQR factorization. Hence, as long as σ_{k+1} is small, Theorem 3.1 guarantees that TSVD and TQR will produce approximately the same solutions, as well as residuals. These two methods are therefore in many circumstances equally well suited for solving rank deficient least squares problems with well-determined ϵ-rank (unless, of course, the exact TSVD solution is required), and $\mathbf{x}_{TQR}$ is cheaper to compute than $\mathbf{x}_{TSVD}$.

The RRQR also leads to a basic solution $\mathbf{x}_B$ with approximately the same residual vector as both $\mathbf{r}_{TQR}$ and $\mathbf{r}_{TSVD}$, because R_{11} is guaranteed to be well conditioned. The basic solution itself, on the other hand, may be very different from both $\mathbf{x}_{TSVD}$ and $\mathbf{x}_{TQR}$. This is so because the basic solution $\mathbf{x}_B$ has a (possibly large) component in the numerical null-space of A.

To illustrate the different properties of all the above-mentioned solutions, we have carried out a series of experiments using Pro-Matlab [20]. The dimensions were $m = n = 100$, the ratio σ_1/σ_k (where k is the ϵ-rank) was 10^3 for all the matrices, and the right-hand side $\mathbf{b}$ was generated such that the TSVD residual vector satisfies $\|\mathbf{r}_{TSVD}\| = 10^{-3}\|A\,\mathbf{x}_{TSVD}\|$. Typical numerical results are shown in Table 2 for three different values of the ϵ-rank k and three different ratios σ_k/σ_{k+1}. These results clearly illustrate how the similarity of $\mathbf{x}_{TSVD}$ and $\mathbf{x}_{TQR}$, as well as the similarity of all the residual vectors, depends on the size of the gap between σ_k and σ_{k+1}, but not on the ϵ-rank k. Table 2 also illustrates that the solutions $\mathbf{x}_{TQR}$ and $\mathbf{x}_B$ generally are very different.

TABLE 2
Typical numerical results for rank deficient least squares problems with $m = n = 100$, $\sigma_1/\sigma_k = 10^3$.

Legend for each entry below	
$\|\mathbf{x}_{TSVD} - \mathbf{x}_{TQR}\|$	$\|\mathbf{r}_{TSVD} - \mathbf{r}_{TQR}\|/\|\mathbf{b}\|$
$\|\mathbf{x}_{TQR} - \mathbf{x}_B\|$	$\|\mathbf{r}_{TQR} - \mathbf{r}_B\|/\|\mathbf{b}\|$

$\frac{\sigma_k}{\sigma_{k+1}}$	$k = 50$		$k = 75$		$k = 90$	
10^6	$4.51 \cdot 10^{-10}$	$1.55 \cdot 10^{-10}$	$6.76 \cdot 10^{-10}$	$2.19 \cdot 10^{-10}$	$2.14 \cdot 10^{-10}$	$8.04 \cdot 10^{-11}$
	3.53	$1.22 \cdot 10^{-7}$	1.70	$9.59 \cdot 10^{-8}$	1.51	$5.05 \cdot 10^{-8}$
10^3	$4.34 \cdot 10^{-7}$	$1.34 \cdot 10^{-7}$	$7.31 \cdot 10^{-7}$	$2.26 \cdot 10^{-7}$	$1.93 \cdot 10^{-7}$	$7.20 \cdot 10^{-8}$
	3.53	$1.04 \cdot 10^{-4}$	1.70	$9.35 \cdot 10^{-5}$	1.51	$3.11 \cdot 10^{-5}$
10	$1.90 \cdot 10^{-3}$	$3.80 \cdot 10^{-5}$	$2.00 \cdot 10^{-3}$	$5.73 \cdot 10^{-5}$	$2.29 \cdot 10^{-5}$	$7.03 \cdot 10^{-6}$
	3.52	$9.52 \cdot 10^{-3}$	1.70	$9.10 \cdot 10^{-3}$	1.51	$2.62 \cdot 10^{-3}$

4. Subset selection problems. *Subset selection* is the problem of determining the most linearly independent columns of the matrix A. To be precise, if k is the ϵ-rank of A, then the aim is to find a column permutation Π such that the submatrix consisting of the first k columns of $A\Pi$ are as well conditioned as possible. The RRQR factorization of A obviously produces such a permutation Π. The basic solution $\mathbf{x}_B$ discussed in the previous section is in fact the least squares solution derived from this strategy by forcing to zero those elements of $\mathbf{x}_B$ that correspond to the linearly dependent columns of A. Such a solution may in some applications be preferred to the TSVD and TQR solutions. Subset selection is also interesting in its own right.

It is therefore interesting to compare the RRQR-based subset selection algorithm with the standard SVD-based algorithm proposed by Golub, Klema, and Stewart [11], [12, §12.2]. Their algorithm constructs a permutation matrix Π_{SVD} such that the bottom right $(n-k) \times (n-k)$ submatrix $\tilde{V}_{22}$ of $\Pi_{SVD}^T V$ is well conditioned, and then the first $n-k$ columns of $A\Pi_{SVD}$ are guaranteed to form a well-conditioned matrix. In other words, these columns form a linearly independent set of the columns of A.

The RRQR factorization also produces a permutation Π such that the first $n-k$ columns of $A\Pi$ are linearly independent, but this Π is constructed on the basis of information in the matrix W. The difference between these two methods therefore basically lies in the way that Π is computed. In general, we cannot guarantee that the two algorithms give identical permutations, and this makes a comparison of the two solutions difficult. On the other hand, it is more appropriate for subset selection problems to compare the subspaces spanned by the first $n-k$ columns of $A\Pi_{SVD}$ and $A\Pi$.

THEOREM 4.1. *Let* $\mathcal{R}(U_k)$ *denote the subspace* $\text{span}\{\mathbf{u}_1, \cdots, \mathbf{u}_k\}$, *and let* B_{SVD} *and* B_{RRQR} *denote the submatrices consisting of the first* $n-k$ *columns of* $A\Pi_{SVD}$ *and* $A\Pi$, *respectively. Then*

$$\sin\theta(\mathcal{R}(U_k), \mathcal{R}(B_{SVD})) \leq \sigma_{k+1} \|\tilde{V}_{22}^{-1}\| \, \sigma_k^{-1}, \tag{16}$$

$$\sin\theta(\mathcal{R}(U_k), \mathcal{R}(B_{RRQR})) \leq \sigma_{k+1} \|R_{11}^{-1}\|. \tag{17}$$

Proof. Our proof follows that given by Golub and Van Loan for [12, Thm. 12.2.2]. In their proof, they derive the upper bound $\sin\theta(\mathcal{R}(U_k), \mathcal{R}(B_{SVD})) \leq \sigma_{k+1}/\sigma_k(B_{SVD})$, where $\sigma_k(B_{SVD})$ denotes the smallest singular value of B_{SVD}. They also show in [12,

TABLE 3

Typical numerical results for the RRQR *subset selection algorithm. Each entry in the table shows the subspace angles* $\sin\theta(\mathcal{R}(U_k), \mathcal{R}(B_{RRQR}))$ *and* $\sin\theta(\mathcal{R}(B_{SVD}), \mathcal{R}(B_{RRQR}))$. *The dimensions are* $m = n = 100$.

$\frac{\sigma_k}{\sigma_{k+1}}$	$k = 50$		$k = 75$		$k = 90$	
10^6	$1.03 \cdot 10^{-6}$	$1.26 \cdot 10^{-6}$	$1.58 \cdot 10^{-6}$	$1.74 \cdot 10^{-6}$	$1.30 \cdot 10^{-6}$	$1.51 \cdot 10^{-6}$
10^3	$9.70 \cdot 10^{-4}$	$1.18 \cdot 10^{-3}$	$1.54 \cdot 10^{-3}$	$1.66 \cdot 10^{-3}$	$1.30 \cdot 10^{-3}$	$1.51 \cdot 10^{-3}$
10	$9.38 \cdot 10^{-2}$	$1.14 \cdot 10^{-1}$	$1.50 \cdot 10^{-3}$	$1.62 \cdot 10^{-1}$	$1.30 \cdot 10^{-7}$	$1.50 \cdot 10^{-1}$

Thm. 12.2.1] that $\sigma_k(B_{SVD}) \geq \sigma_k \|\tilde{V}_{22}^{-1}\|$, which yields (16). Similarly, since

$$B_{RRQR} = Q \begin{pmatrix} R_{11} \\ 0 \end{pmatrix},$$

we obtain that

$$\sin\theta(\mathcal{R}(U_k), \mathcal{R}(B_{RRQR})) \leq \frac{\sigma_{k+1}}{\sigma_k(B_{RRQR})} = \frac{\sigma_{k+1}}{\sigma_k(R_{11})},$$

which is (17). □

The submatrix R_{11} in the RRQR factorization of A is guaranteed to be well conditioned and $\|R_{11}^{-1}\|$ is of the order σ_k^{-1}. Theorem 4.1 therefore ensures that the sine of both subspace angles is of the same order as σ_{k+1}/σ_k. Moreover, if this ratio is small, then Theorem 4.1 ensures that both $\mathcal{R}(B_{SVD})$ and $\mathcal{R}(B_{RRQR})$ will be close to the subspace $\mathcal{R}(U_k)$, and the subspace angle between $\mathcal{R}(B_{SVD})$ and $\mathcal{R}(B_{RRQR})$ is therefore also bound to be small. Hence, if σ_{k+1}/σ_k is small, then SVD and RRQR yield approximately the same subspaces. We illustrate this in Table 3, where we show typical values of $\sin\theta(\mathcal{R}(U_k), \mathcal{R}(B_{RRQR}))$ and $\sin\theta(\mathcal{R}(B_{SVD}), \mathcal{R}(B_{RRQR}))$. We see that both subspace angles are indeed of the same order as the ratio σ_{k+1}/σ_k. The conclusion is that although the SVD and RRQR subset selection algorithms do not necessarily produce the same column permutations and thus the same subset of columns of A, the subspaces spanned by these two sets of columns are still almost identical whenever σ_{k+1}/σ_k is small.

5. Matrix approximation. It is well known that the best rank-k approximation A_k to the matrix A, in any unitarily invariant norm, is the matrix obtained by truncating the SVD expansion in (2) after the first k terms. That is, the matrix A_k given by

$$A_k = \sum_{i=1}^{k} \mathbf{u}_i \, \sigma_i \, \mathbf{v}_i^T, \qquad k \leq n \tag{18}$$

solves the problem $\min_{\text{rank}(X)=k} \|A - X\|$ (this is the Eckart–Young–Mirsky Theorem [12, Thm. 2.5.2]). In particular, $\|A - A_k\| = \sigma_{k+1}$ and $\|A - A_k\|_F = (\sigma_{k+1}^2 + \cdots + \sigma_n^2)^{1/2}$. As we shall demonstrate in the following theorem, neglection of the submatrix R_{22} in a RRQR factorization also yields a good rank-k approximation, provided that W_2 is well conditioned.

THEOREM 5.1. *Let*

$$B_k = Q \begin{pmatrix} R_{11} & R_{12} \\ 0 & 0 \end{pmatrix} \Pi^T$$

TABLE 4
Typical numerical results for RRQR *matrix approximations. Each entry shows* $\|A - B_k\|$ *and* $\|A_k - B_k\|$ *for matrices with* $m = n = 100$, $\sigma_1 = 1$, *and* $\sigma_{k+1} = \|A - A_k\| = 10^{-7}$.

$\frac{\sigma_k}{\sigma_{k+1}}$	$k = 50$		$k = 75$		$k = 90$	
10^6	$3.05 \cdot 10^{-7}$	$2.90 \cdot 10^{-7}$	$2.55 \cdot 10^{-7}$	$2.48 \cdot 10^{-7}$	$2.63 \cdot 10^{-7}$	$2.51 \cdot 10^{-7}$
10^3	$2.39 \cdot 10^{-7}$	$2.17 \cdot 10^{-7}$	$2.51 \cdot 10^{-7}$	$2.31 \cdot 10^{-7}$	$5.68 \cdot 10^{-7}$	$5.60 \cdot 10^{-7}$
10	$4.20 \cdot 10^{-7}$	$4.08 \cdot 10^{-7}$	$5.02 \cdot 10^{-7}$	$4.92 \cdot 10^{-7}$	$6.98 \cdot 10^{-7}$	$6.91 \cdot 10^{-7}$

denote the matrix obtained from the RRQR *factorization* (4) *by neglecting the submatrix* R_{22}. *Then*

$$\|A - B_k\| \le \sqrt{n-k}\,\|W_2^{-1}\|\,\sigma_{k+1}, \tag{19}$$

$$\|A - B_k\|_F \le \sqrt{n-k}\,\|W_2^{-1}\|\,(\sigma_{k+1}^2 + \cdots + \sigma_n^2)^{1/2}, \tag{20}$$

where A_k *denotes the truncated* SVD *matrix* (18).

Proof. Obviously, $\|A - B_k\| = \|R_{22}\|$ and $\|A - B_k\|_F = \|R_{22}\|_F$, and (19) then follows from Theorem 2.1. To get an upper bound for $\|R_{22}\|_F$, we use the fact that $\|R_{22}\|_F \le \|R_{22}\,W_2\|_F\,\|W_2^{-1}\|_F \le \|A\,\Pi\,W\|_F\,\|W_2^{-1}\|_F$. Since each column vector $\mathbf{w}_i$ of W satisfies

$$\|A\,\Pi\,\mathbf{w}_i\| = \left\| QR \begin{pmatrix} \mathbf{w}^{(i)} \\ \mathbf{0} \end{pmatrix} \right\| = \|R^{(i)}\mathbf{w}^{(i)}\| = \delta_i \le \sigma_{k+i}$$

(cf. (5)), we obtain $\|A\,\Pi\,W\|_F^2 \le \sigma_{k+1}^2 + \cdots + \sigma_n^2$. This yields (20). □

Theorem 5.1 states that matrix approximations derived from RRQR factorizations are almost as good as those derived from truncated SVD approximations. Moreover, it is trivial from Theorem 5.1 that the difference between A_k and B_k satisfies $\|A_k - B_k\| \le (1 + \sqrt{n-k}\,\|W_2^{-1}\|)\,\sigma_{k+1}$. The interesting fact about the RRQR matrix approximations is that the bounds in (19) and (20) do not depend on the gap between σ_k and σ_{k+1}. The algorithm can therefore be applied to any matrix independently of its singular value spectrum and with potential application to digital image compression. We illustrate this in Table 4, which shows typical values of $\|A - B_k\|$ and $\|A_k - B_k\|$ for random matrices with $m = n = 100$, $\sigma_1 = 1$, $\sigma_{k+1} = \|A - A_k\| = 10^{-7}$, and different values of the ϵ-rank k and the ratio σ_k/σ_{k+1}. The table confirms that $\|A - B_k\|$ and $\|A_k - B_k\|$ are both of the order σ_{k+1}, as proved in the above theorem, and are independent of σ_k/σ_{k+1}.

6. Total least squares problems. Another aspect of matrix approximation arises in connection with total least squares (TLS) problems [12, §12.3]. The key problem here is to find three matrices E, R, and X such that $\|(E, R)\|$ is small and such that $(A + E)\,X = B + R$, with $A \in \Re^{m\times n}$ and $B \in \Re^{m\times p}$ (the ordinary least squares problem corresponds to setting $E = 0$). Total least squares problems typically arise in applications where both the coefficient matrix A and the right-hand side B are contaminated with errors, in which case one can think of E and R as being residual matrices.

The classical approach to TLS is based on the SVD and is described in [12, §12.3]. Let

$$V = \begin{pmatrix} V_{11} & V_{12} \\ V_{21} & V_{22} \end{pmatrix}$$

be the matrix of right singular vectors in the SVD of the compound matrix (A, B), partitioned such that $V_{11} \in \Re^{n \times r}$, where r is the ϵ-rank of A. If A has full rank $(r = n)$, then the solution that minimizes the Frobenius norm of the compound residual matrix $\|(E, R)\|_F$ is given by $X_{SVD} = -V_{12}V_{22}^{-1}$ [12, Thm. 12.3.1]. When A is rank deficient $(r < n)$, the solution of the minimum Frobenius norm that minimizes $\|(E, R)\|_F$ is given by [27]

$$X_{SVD} = -V_{12}V_{22}^{+}. \tag{21}$$

Here, V_{22}^{+} is the pseudoinverse of the $p \times (n-r+p)$ submatrix V_{22}, and V_{22}^{+} is identical to V_{22}^{-1} when $r = n$. As long as $p \ll n$, V_{22}^{+} can easily be computed stably, e.g., by a QR factorization of V_{22}. Note that there is no guarantee that V_{22} is well conditioned. Also note that the numerical rank of the compound matrix (A, B) is not used in total least squares, only the ϵ-rank of A is required. We return to this aspect at the end of this section.

From (21) it is obvious that the complete SVD of (A, B) is not needed for computing X_{SVD}. Instead, only the right singular vectors corresponding to the smallest $n - r + p$ singular values are required. It is possible to modify the classical SVD algorithm by taking this into account. This is done in the PSVD algorithm [25], [26], developed for TLS problems, which requires $(2m + 2n + 14p + 2)(n + p)^2 + O(n)$ flops to compute X_{SVD} if the bidiagonalization part is preceded by a standard QR factorization, as described in [5].

Here we derive algorithms for TLS based on the RRQR factorization of (A, B). The key idea is to use the RRQR algorithm to compute approximate null-vectors corresponding to the $n-r+p$ smallest singular values of (A, B) (assume for the moment that r is known). We can then use the same approach as that used in [8] to compute TSVD solutions, namely, to use inverse subspace iterations to refine the numerical null-vectors in W. This approach yields the right singular vectors required to compute X_{SVD} by (21), and the accuracy of the solution depends primarily on the number of subspace iterations. Except for a single backsubstitution, which is not required to compute X_{SVD}, the dominating computational effort remains the same as that for computing $\mathbf{x}_{SVD}$; cf. §3. Hence, this approach requires $(2m - \frac{2}{3}n + (2q + 3))(n + p)^2$ flops (where q is the number of inverse iterations required to refine the null vectors), and it is actually less demanding than the PSVD approach, mainly because a reduction to bidiagonal form is avoided.

We shall now analyze an even simpler approach to TLS, based directly on the matrix W without performing any inverse iterations. Whenever the singular values σ_r and σ_{r+1} are well separated, it is shown in [8, Thm. 4.1] that the range of ΠW is a good approximation to $\mathcal{N}_r(A, B)$, in the sense that the subspace angle θ between $\mathcal{R}(\Pi W)$ and $\mathcal{N}_r(A, B)$ is small. Thus, it is natural to obtain an approximation to the last $n - r + p$ null vectors of (A, B) simply by orthonormalization of the columns of ΠW, e.g., by means of the modified Gram–Schmidt process, to obtain

$$\Pi W = \begin{pmatrix} \bar{V}_{12} \\ \bar{V}_{22} \end{pmatrix} \bar{R}, \qquad \bar{V}_{12}^T\bar{V}_{12} + \bar{V}_{22}^T\bar{V}_{22} = I_{n-r+p}, \tag{22}$$

and then define the alternative TLS solution by $X_{RRQR} = -\bar{V}_{12}\bar{V}_{22}^{-1}$ or, in the general case, by

$$X_{RRQR} = -\bar{V}_{12}\bar{V}_{22}^{+}. \tag{23}$$

As is the case in the SVD approach, we cannot guarantee that $\bar{V}_{22}$ is well conditioned. The question is then whether a small subspace angle θ ensures that X_{RRQR} is close to X_{SVD}. In Theorem 6.3 below we give a positive answer to this question for the full rank case ($r = n$), but first we need the following two lemmas.

LEMMA 6.1. *If*

$$V = \begin{pmatrix} V_{11} & V_{12} \\ V_{21} & V_{22} \end{pmatrix}$$

is orthogonal, then the norm of the Schur complement of V_{22} *satisfies*

$$\|V_{11} - V_{12}V_{22}^{-1}V_{21}\| = \|V_{22}^{-1}\|. \tag{24}$$

Proof. For simplicity, we assume $V_{11} \in \Re^{q\times q}$, $V_{22} \in \Re^{p\times p}$ with $q > p$. Then the CS decomposition [12, Thm. 2.6.1] of V is given by

$$\begin{pmatrix} V_{11} & V_{12} \\ V_{21} & V_{22} \end{pmatrix} = \begin{pmatrix} \hat{U}_1 & 0 \\ 0 & \hat{U}_2 \end{pmatrix} \begin{pmatrix} I_{q-p} & 0 & 0 \\ 0 & C & S \\ 0 & -S & C \end{pmatrix} \begin{pmatrix} \hat{V}_1^T & 0 \\ 0 & \hat{V}_2^T \end{pmatrix},$$

where $C^2 + S^2 = I_p$. It is straightforward to show that $V_{22}^{-1} = \hat{V}_2 C^{-1} \hat{U}_2^T \Rightarrow \|V_{22}^{-1}\| = \|C^{-1}\|$ and, by inserting the CS decomposition, that

$$V_{11} - V_{12}V_{22}^{-1}V_{21} = \hat{U}_1 \begin{pmatrix} I_{q-p} & 0 \\ 0 & C^{-1} \end{pmatrix} \hat{V}_1^T,$$

from which (24) immediately follows. □

LEMMA 6.2. *If* Q, V, *and* $\bar{V}$ *are orthogonal matrices such that*

$$\bar{V} = (\bar{V}_1, \bar{V}_2) = VQ = (V_1, V_2) \begin{pmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{pmatrix}, \tag{25}$$

then the subspace angle θ *between the subspaces* $\mathcal{R}(V_2)$ *and* $\mathcal{R}(\bar{V}_2)$ *satisfies*

$$\sin\theta = \|Q_{12}\|, \qquad \cos\theta = \|Q_{22}^{-1}\|^{-1}. \tag{26}$$

Proof. We have $\sin\theta = \|V_1^T \bar{V}_2\| = \|V_1^T(V_1 Q_{12} + V_2 Q_{22})\| = \|Q_{12}\|$. The relation for $\cos\theta$ follows from the CS decomposition of Q. □

THEOREM 6.3. *Let* X_{SVD} *and* X_{RRQR} *be given by* (21) *and* (23), *respectively, and assume that* A *has full rank. If* θ *denotes the subspace angle between* $\mathcal{N}_r(A, B)$ *and* $\mathcal{R}(\Pi W)$, *and if* $\tan\theta < \|V_{22}^{-1}\|^{-1}$, *then*

$$\|X_{SVD} - X_{RRQR}\| \le \tan\theta \, \|V_{22}^{-1}\|^2 \Big(1 + O(\tan\theta)\Big). \tag{27}$$

Proof. Using the notation from (25) with

$$V_2 = \begin{pmatrix} V_{12} \\ V_{22} \end{pmatrix} \quad \text{and} \quad \bar{V}_2 = \begin{pmatrix} \bar{V}_{12} \\ \bar{V}_{22} \end{pmatrix},$$

we obtain $\bar{V}_{12} = V_{11}Q_{12} + V_{12}Q_{22}$ and $\bar{V}_{22} = V_{21}Q_{12} + V_{22}Q_{22} = (V_{21}Q_{12}Q_{22}^{-1}V_{22}^{-1} + I_p)\, V_{22}Q_{22}$. If we define $\Delta = V_{21}Q_{12}Q_{22}^{-1}V_{22}^{-1}$, then $\bar{V}_{22} = (I_p + \Delta)\, V_{22}Q_{22}$. From

TABLE 5
Mean and maximum values for an experiment with 100 *matrices with dimensions* $m = 150$, $n = 100$, *and* $p = 5$. *The* ϵ*-rank of* A *is* $r = 90$.

	Mean	Maximum
$\tan\theta$	$1.92 \cdot 10^{-7}$	$1.92 \cdot 10^{-6}$
$\|\bar{V}_{22}^{+}\|$	1.20	1.30
$\tan\theta \, \|V_{22}^{+}\|^2 / \|X_{SVD}\|$	$4.07 \cdot 10^{-7}$	$3.95 \cdot 10^{-6}$
$\|X_{SVD} - X_{RRQR}\| / \|X_{SVD}\|$	$5.06 \cdot 10^{-8}$	$5.27 \cdot 10^{-7}$

Lemma 6.2 we have $\|\Delta\| \leq \tan\theta \, \|V_{22}^{-1}\|$, and since we have assumed that $\tan\theta < \|V_{22}^{-1}\|^{-1}$ such that $\|\Delta\| < 1$, we have

$$\bar{V}_{22}^{-1} = Q_{22}^{-1} V_{22}^{-1} (I_p + \Delta)^{-1} = Q_{22}^{-1} V_{22}^{-1} (I_p - \Delta + O(\Delta^2)).$$

Thus, we obtain that

$$X_{RRQR} = -\bar{V}_{12}\bar{V}_{22}^{-1} = -V_{11}Q_{12}Q_{22}^{-1}V_{22}^{-1} - V_{12}V_{22}^{-1} + (V_{11}Q_{12}Q_{22}^{-1} + V_{12})V_{22}^{-1}\Delta + O(\Delta^2).$$

Inserting the expression for Δ and using the fact that $X_{SVD} = -V_{12}V_{22}^{-1}$, we then obtain

$$X_{RRQR} - X_{SVD} = -(V_{11} - V_{12}V_{22}^{-1}V_{21}) \, Q_{12}Q_{22}^{-1}V_{22}^{-1} + V_{11}Q_{12}Q_{22}^{-1}V_{22}^{-1}\Delta + O(\Delta^2).$$

Taking norms on both sides of the above equation and using Lemmas 6.1 and 6.2, we then obtain (27). □

Theorem 6.3 states that if A has full rank, if the subspace angle θ is small, and if V_{22} is well conditioned, then the total least squares solution X_{RRQR} defined by (23) will be close to the ordinary total least squares solution X_{SVD} given by (21). Although we cannot ensure that the matrix V_{22} is well conditioned, it is our experience that it is very unlikely to be ill conditioned.

Unfortunately, we were not able to prove a similar result for the rank deficient case ($r < n$), the reason being the appearance of the pseudoinverse V_{22}^{+}, which severely complicates the relations. On the other hand, our experiments with Matlab strongly suggest that (27) holds in general: among 100 random matrices with dimensions $m = 150$, $n = 100$, and $p = 5$ and with A's ϵ-rank $r = 90$, the ratio between $\|X_{SVD} - X_{RRQR}\|$ and $\tan\theta \, \|V_{22}^{+}\|^2$ in (27) never exceeded 0.6. The results from this test are summarized in Table 5, where we list the mean and maximum values of $\tan\theta$, $\|\bar{V}_{22}^{+}\|$, $\tan\theta \, \|V_{22}^{+}\|^2 / \|X_{SVD}\|$, and $\|X_{SVD} - X_{RRQR}\| / \|X_{SVD}\|$. This confirms the fact that $\bar{V}_{22}$ is in all likelihood a well-conditioned matrix and that the upper bound in (27) is not large. The conclusion is that X_{RRQR} is indeed a good TLS solution.

Besides being faster than both the traditional SVD algorithm and the PSVD algorithm, our approach to TLS based on RRQR has one more important advantage: if the ϵ-rank of A is unknown, then one can compute it during the RRQR factorization of (A, B) with very little overhead. First compute an RRQR factorization of A,

$$A\,\Pi^{(1)} = Q^{(1)} \begin{pmatrix} R^{(1)} \\ 0 \end{pmatrix} = Q^{(1)} \begin{pmatrix} R_{11}^{(1)} & R_{12}^{(1)} \\ 0 & R_{22}^{(1)} \\ 0 & 0 \end{pmatrix}, \tag{28}$$

revealing the ϵ-rank r of A. Then append

$$\begin{pmatrix} R^{(1)} \\ 0 \end{pmatrix}$$

with $(Q^{(1)})^T B$, to form

$$C = \begin{pmatrix} R_{11}^{(1)} & R_{12}^{(1)} & \hat{B}_1 \\ 0 & R_{22}^{(1)} & \hat{B}_2 \\ 0 & 0 & \hat{B}_3 \end{pmatrix}, \qquad \begin{pmatrix} \hat{B}_1 \\ \hat{B}_2 \\ \hat{B}_3 \end{pmatrix} = (Q^{(1)})^T B$$

and compute the RRQR factorization of the lower right 2×2 block submatrix of C:

$$\begin{pmatrix} R_{22}^{(1)} & \hat{B}_2 \\ 0 & \hat{B}_3 \end{pmatrix} \Pi^{(2)} = Q^{(2)} \begin{pmatrix} R_{22}^{(2)} & R_{23}^{(2)} \\ 0 & R_{33}^{(2)} \end{pmatrix}. \tag{29}$$

Finally, apply the second set of permutations to $\begin{pmatrix} R_{12}^{(1)} & \hat{B}_2 \end{pmatrix}$ to obtain

$$\begin{pmatrix} R_{12}^{(2)} & R_{13}^{(2)} \end{pmatrix} = \begin{pmatrix} R_{12}^{(1)} & \hat{B}_1 \end{pmatrix} \Pi^{(2)}. \tag{30}$$

Then the resulting triangular factor of (A, B) is given by

$$R = \begin{pmatrix} R_{11}^{(1)} & R_{12}^{(2)} & R_{13}^{(2)} \\ 0 & R_{22}^{(2)} & R_{23}^{(2)} \\ 0 & 0 & R_{33}^{(2)} \end{pmatrix}.$$

During the second factorization (29), one can take advantage of the fact that the first $n - r$ columns are in all likelihood linear combinations within the tolerance ϵ of the remaining columns. Since $R_{11}^{(1)}$ is guaranteed to be well conditioned, the QR factorization resulting from (28)–(30) is close to being an RRQR factorization of (A, B), and the desired RRQR factorization can then be achieved by a few "backward passes" through R as described in [1, §5].

REFERENCES

[1] C. H. Bischof and P. C. Hansen, *Structure-preserving and rank-revealing QR-factorizations*, Report MCS-P100-0989, Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, IL, 1989; SIAM J. Sci. Statist. Comput., 12 (1991), pp. 1332–1350.

[2] C. H. Bischof and G. M. Shroff, *On updating signal subspaces*, IEEE Trans. Signal Process., 40 (1992), pp. 96–105.

[3] Å. Björck, *Least squares methods*, in Handbook of Numerical Analysis, Vol. I: Finite Difference Methods—Solution of Equations in R^n, P. G. Ciarlet and J. L. Lions, eds., Elsevier, Amsterdam, 1990.

[4] P. A. Businger and G. H. Golub, *Linear least squares solution by Householder transformation*, Numer. Math., 7 (1965), pp. 269–276.

[5] T. F. Chan, *An improved algorithm for computing the singular value decomposition*, ACM Trans. Math. Software, 8 (1982), pp. 72–83.

[6] ———, *Alternative to the* SVD: *Rank revealing* QR-*factorizations*, in Advanced Algorithms and Architectures for Signal Processing, J. M. Speiser, ed., SPIE Proceedings Vol. 696, 1986.

[7] ———, *Rank revealing* QR *factorizations*, Linear Algebra Appl., 88/89 (1987), pp. 67–82.

[8] T. F. Chan and P. C. Hansen, *Computing truncated* SVD *least squares solutions by rank revealing* QR *factorizations*, SIAM J. Sci. Statist. Comput., 11 (1990), pp. 519–530.

[9] P. Comon and G. H. Golub, *Tracking a few extreme singular values and vectors in signal processing*, Proc. IEEE, 78 (1990), pp. 1327–1343.

[10] L. Foster, *Rank and null space calculations using matrix decomposition without column interchanges*, Linear Algebra Appl., 74 (1986), pp. 47–71.

[11] G. H. Golub, V. Klema, and G. W. Stewart, *Rank degeneracy and least squares problems*, Report TR-456, Department of Computer Science, University of Maryland, College Park, MD, 1976.

[12] G. H. Golub and C. F. Van Loan, *Matrix Computations*, Second Edition, The Johns Hopkins University Press, Baltimore, MD, 1989.

[13] P. C. HANSEN, *The truncated* SVD *as a method for regularization*, BIT, 27 (1987), pp. 534–553.
[14] ———, *Truncated* SVD *solutions to discrete ill-posed problems with ill-determined numerical rank*, SIAM J. Sci. Statist. Comput., 11 (1990), pp. 503–518.
[15] P. C. HANSEN, T. SEKII, AND H. SHIBAHASHI, *The modified truncated* SVD *method for regularization in general form*, SIAM J. Sci. Statist. Comput., 13 (1992), to appear.
[16] M. T. HEATH, *Some extensions of an algorithm for sparse linear least squares problems*, SIAM J. Sci. Statist. Comput., 3 (1982), pp. 223–237.
[17] Y. P. HONG AND C.-T. PAN, *Rank-revealing* QR *factorizations and* SVD, Math. Comput., to appear.
[18] C. L. LAWSON AND R. J. HANSON, *Solving Least Squares Problems*, Prentice-Hall, Englewood Cliffs, NJ, 1974.
[19] T. A. MANTEUFFEL, *An interval analysis approach to rank determination in linear least squares problems*, SIAM J. Sci. Statist. Comput., 2 (1981), pp. 335–348.
[20] C. B. MOLER, J. LITTLE, AND S. BANGERT, *Pro-Matlab User's Guide*, The MathWorks, Sherborn, MA, 1987.
[21] E. NG, *A scheme for handling rank-deficiency in the solution of sparse linear least squares problems*, SIAM J. Sci. Statist. Comput., 12 (1991), pp. 1173–1183.
[22] L. REICHEL AND W. B. GRAGG, *Algorithm* 686: *Fortran subroutines for updating the* QR *decomposition*, ACM Trans. Math. Software, 16 (1990), pp. 369–377.
[23] G. W. STEWART, *Rank degeneracy*, SIAM J. Sci. Statist. Comput., 5 (1984), pp. 403–413.
[24] ———, *An updating algorithm for subspace tracking*, Report CS-TR 2494, Department of Computer Science, University of Maryland, College Park, MD, 1990; IEEE Trans. Signal Process., to appear.
[25] S. VAN HUFFEL, J. VANDEWALLE, AND A. HAEGEMANS, *An efficient and reliable algorithm for computing the singular subspace of a matrix, associated with its smallest singular values*, J. Comput. Appl. Math., 19 (1987), pp. 313–330.
[26] S. VAN HUFFEL AND J. VANDEWALLE, *The partial total least squares algorithm*, J. Comput. Appl. Math., 21 (1988), pp. 333–341.
[27] M. D. ZOLTOWSKI, *Generalized minimum norm and constrained total least squares with applications to array signal processing*, in Advanced Algorithms and Architectures for Signal Processing III, F. T. Luk, ed., SPIE Vol. 975, 1988, pp. 78–85.