Recursive calculation of dominant singular subspaces.

Chahlaoui, Younes and Gallivan,
Kyle A and Van Dooren, Paul

2003

MIMS EPrint: **2008.14**

Manchester Institute for Mathematical Sciences
School of Mathematics

The University of Manchester

Reports available from: http://eprints.maths.manchester.ac.uk/
And by contacting: The MIMS Secretary
School of Mathematics
The University of Manchester
Manchester, M13 9PL, UK

ISSN 1749-9097

SIAM J. MATRIX ANAL. APPL.
Vol. 25, No. 2, pp. 445–463

RECURSIVE CALCULATION OF DOMINANT SINGULAR SUBSPACES*

Y. CHAHLAOUI†, K. GALLIVAN‡, AND P. VAN DOOREN†

Abstract. In this paper we show how to compute recursively an approximation of the left and right dominant singular subspaces of a given matrix. In order to perform as few as possible operations on each column of the matrix, we use a variant of the classical Gram–Schmidt algorithm to estimate this subspace. The method is shown to be particularly suited for matrices with many more rows than columns. Bounds for the accuracy of the computed subspace are provided. Moreover, the analysis of error propagation in this algorithm provides new insights in the loss of orthogonality typically observed in the classical Gram–Schmidt method.

Key words. singular value decomposition, Gram–Schmidt, dominant subspace

AMS subject classifications. 15A18, 15A42, 65Y20

DOI. 10.1137/S0895479803374657

1. Introduction.

In many problems one needs to compute the projector on the dominant subspace of a given data matrix A of dimension $m \times n$. The type of application we are thinking of here implies $m \gg n$, and for the sake of simplicity we will assume A to be real. In addition, we assume that the matrix A is produced incrementally, so all of the columns are not available simultaneously. Several applications have this property. For example, approximating a matrix A in which each column represents an image of a given sequence amounts to an SVD-based compression [5]. Such an approximation is also used in the context of observation-based model reduction for dynamical systems. The so-called proper orthogonal decomposition (POD) approximation uses the dominant left space of a matrix A where a column consists of a time instance of the solution of an evolution equation, e.g., the flow field from a fluid dynamics simulation. Since these flow fields tend to be very large only a small number can be stored efficiently during the simulation, and therefore an incremental approach is useful [11]. Finally, the dominant space approximation is also used in text retrieval to encode document/term information and avoid certain types of semantic noise. The incremental form is required when documents are added or when the entire matrix is not available at one point in time and space [3].

In each of these applications, one can interpret the columns of the matrix A as "data vectors" with some "energy" equal to their 2-norm. Finding the dominant space of dimension $k < \min(m, n)$ amounts to finding the k first columns of the matrix U

*Received by the editors June 29, 2000; accepted for publication (in revised form) by J. M. Hyman March 3, 2003; published electronically September 9, 2003. This paper presents research supported by NSF contract CCR-99-12415 and by the Belgian Programme on Inter-University Poles of Attraction, initiated by the Belgian State, Prime Minister's Office for Science, Technology and Culture. The scientific responsibility rests with its authors.

http://www.siam.org/journals/simax/25-2/37465.html

†Department of Mathematical Engineering, Université Catholique de Louvain, CESAME, avenue Georges Lemaître 4, B-1348 Louvain-la-Neuve, Belgium (chahlaoui@csam.ucl.ac.be, vdooren@csam.ucl.ac.be). The work of the first author was partially carried out within the framework of a collaboration agreement between CESAME (Université Catholique de Louvain, Belgium) and LINMA of the Faculty of Sciences (Université Chouaib Doukkali, Morocco), funded by the Secretary of the State for Development Cooperation and by the CIUF (Conseil Interuniversitaire de la Communauté Française, Belgium).

‡School of Computational Science and Information Technology, Florida State University, Tallahassee, FL 32306 (gallivan@csit.fsu.edu).

in the singular value decomposition of A:

$$A = U\Sigma V^T, \quad U^TU = I_n, \quad VV^T = V^TV = I_n, \quad \Sigma = \mathrm{diag}\{\sigma_1, \ldots \sigma_n\}, \tag{1.1}$$

and where the diagonal elements σ_i of Σ are nonnegative and nonincreasing. This decomposition in fact expresses that the orthogonal transformation V applied to the columns of A yields a new matrix $AV = U\Sigma$ with orthogonal columns of nonincreasing norm. The "dominant" columns of this transformed matrix are obviously the k leading ones. A block version of this decomposition makes this more explicit:

$$A = U\Sigma V^T = \begin{bmatrix} U_1 & U_2 \end{bmatrix} \begin{bmatrix} \Sigma_{1,1} & \\ & \Sigma_{2,2} \end{bmatrix} \begin{bmatrix} V_1 & V_2 \end{bmatrix}^T, \tag{1.2}$$

where U_1 and V_1 have k columns and $\Sigma_{1,1}$ is $k \times k$. An orthogonal basis for the corresponding space is then given by U_1, which is also equal to $AV_1\Sigma_{1,1}^{-1}$. The cost of this decomposition including the construction of U is $14mn^2 + O(n^3)$. For an additional $O(n^3)$ operations it is also possible to compute an orthogonal basis for the columns of V_1, which is required in several applications.

A cheaper procedure is to first perform a QR decomposition of A, followed by a singular value decomposition of the smaller matrix R [4]:

$$A = QR, \quad R = U\Sigma V^T. \tag{1.3}$$

From these equations it is easy to see that $AV = QU\Sigma$, and again this has orthogonal columns of nonincreasing norms. This decomposition costs typically $6mn^2 + O(n^3)$ [8]. It is even more economical to use the normal equations (or covariance matrix) of A. Its eigenvalue decomposition gives

$$A^TA = V\Lambda V^T, \tag{1.4}$$

and comparing this with (1.1) shows that the same matrix V is constructed and that

$$(AV)^T(AV) = \Lambda = \Sigma^T\Sigma.$$

This algorithm requires mn^2 operations to construct A^TA and $mnk + O(n^3)$ operations to obtain $U_1 = AV_1\Sigma_{1,1}^{-1}$. Unfortunately, using the covariance matrix is not recommended because it is more sensitive to rounding errors [8].

In this paper we consider applications where m is huge, and where every column operation on A or on the basis U not only is costly in operations but also involves swapping data from the main memory, which will slow down the algorithm significantly. We present an algorithm that yields an approximate decomposition but requires only $8mnk + O(nk^3)$ operations and also works recursively on the columns of A; i.e., the columns of A (or data vectors) can be produced recursively and A need not be stored in its entirety.

The paper is organized as follows. In sections 2 and 3 we derive an economical sequential procedure to approximate a matrix A by a low-rank factorization. In section 4 we derive bounds for the residual error and compare our method with the "optimal" singular value decomposition approach. In section 5 we illustrate these bounds via numerical experiments. In section 6 we study the effect of round-off and prove backward stability as well as preservation of orthogonality of our computed basis vectors under some mild conditions. This surprising feature (of a classical Gram–Schmidt-like method) is explained and illustrated numerically in the last section.

2. A recursive procedure. In this section we propose a recursive procedure to estimate the dominant subspace of a given matrix A using a sequential (and incremental) processing of the columns of A. Bounds for the accuracy of this decomposition are derived later. The algorithm is based on an efficient calculation of the dominant k-dimensional space of an $m \times (k+1)$ matrix M. Assume that a QR decomposition of M is available:

$$M = QR. \tag{2.1}$$

Then compute the smallest singular vector u_{k+1} of R (i.e., $Rv_{k+1} = u_{k+1}\mu_{k+1}$) and construct an orthogonal transformation G_u such that $G_u^T u_{k+1} = e_{k+1}$. Now apply G_u^T to the rows of R and let G_v be an orthogonal transformation putting $G_u^T R$ back in triangular form:

$$G_u^T R G_v = R_{up}.$$

In this new coordinate system the right singular vector u_{k+1} becomes e_{k+1}, a unit vector with 1 in the $(k+1)$ element, and v_{k+1} is transformed to a new vector $\hat{v}_{k+1}$. Therefore,

$$R_{up} e_{k+1} = \mu_{k+1} \hat{v}_{k+1}, \quad R_{up}^T \hat{v}_{k+1} = \mu_{k+1} e_{k+1}.$$

It easily follows that R_{up} has the form

$$R_{up} = \begin{bmatrix} R_{1,1} & 0 \\ 0 & \mu_{k+1} \end{bmatrix}. \tag{2.2}$$

We therefore have the updated QR decomposition

$$MG_v = Q_{up} R_{up} = (QG_u)(G_u^T R G_v),$$

and since R_{up} has the required block form (1.2) we have found a basis for the dominant k-dimensional subspace of M in the form of the first k columns of Q_{up}.

Both matrices G_u and G_v can be constructed as a product of k 2×2 Givens transformations, allowing an elegant update of R using only $O(k^2)$ operations. But the costly part of the algorithm is the update of Q, and hence it is preferable to choose G_u to be a Householder transformation. When retriangularizing $G_u^T R$ one then needs to perform again a QR factorization, which requires $O(k^3)$ operations, but since $k < n \ll m$, this is of no concern. The cost of the update of Q to Q_{up} is that of a Householder transformation applied to an $m \times (k+1)$ matrix and is thus $4m(k+1)$ operations. The vector u_{k+1} can be computed with a few steps of inverse iteration or with a shifted inverse iteration. The cost of this calculation as well as the update of R is thus $O(k^3)$ and hence negligible with respect to the update of Q. A more involved technique uses modified Givens transformations since their complexity is the same as that of Householder transformations for the product QG_u, and is of $O(k^2)$ when used for forming the product $G_u^T R G_v$. Unfortunately, this requires storing and updating additional diagonal scaling matrices, which typically hurt the performance of codes used for parallel machines.

How is this now applied to finding the dominant subspace of A? We start with a QR factorization of the first k columns of A:

$$A(:, 1\colon k) = Q_{(k)} R_{(k)}. \tag{2.3}$$

Then we recursively apply the following update and downdate of this decomposition. For $i = k+1$ to n, append the next column $a_i \doteq A(:,i)$ to the current matrix decomposition and perform a QR decomposition of it. The formulas for this are standard. Define $r_i = Q_{(i-1)}^T a_i$; then $\hat{a}_i \doteq a_i - Q_{(i-1)} r_i$ is orthogonal to $Q_{(i-1)}$. Define ρ_i as its norm, and $\hat{q}_i = \hat{a}_i/\rho_i$. Then

$$\begin{bmatrix} Q_{(i-1)}R_{(i-1)} & a_i \end{bmatrix} = \begin{bmatrix} Q_{(i-1)} & \hat{q}_i \end{bmatrix} \begin{bmatrix} R_{(i-1)} & r_i \\ 0 & \rho_i \end{bmatrix}. \tag{2.4}$$

Update this matrix decomposition to "deflate" its smallest singular value as above,

$$\begin{bmatrix} Q_{(i-1)} & \hat{q}_i \end{bmatrix} G_u \cdot G_u^T \begin{bmatrix} R_{(i-1)} & r_i \\ 0 & \rho_i \end{bmatrix} G_v = \begin{bmatrix} Q_{(i)} & q_i \end{bmatrix} \cdot \begin{bmatrix} R_{(i)} & 0 \\ 0 & \mu_i \end{bmatrix}, \tag{2.5}$$

and delete the last columns to obtain the new $Q_{(i)}$ and $R_{(i)}$. The complexity of this algorithm is $10mkn + O((n-k)k^3)$ when using Givens transformations for G_u and $8mkn + O((n-k)k^3)$ when using a Householder transformation or modified Givens transformations for G_u. This is clearly cheaper than all earlier algorithms if $m \gg n \gg k$.

The algorithm thus computes at each step a decomposition that "deflates" the smallest singular vector of the current $m \times (k+1)$ matrix and then appends to it the next column of A. All columns of A therefore are passed through once and compared with the current best estimate of this dominant subspace. At first sight this is a very heuristic algorithm, but in the next section we show that quite good bounds can be obtained for the quality of this basis.

REMARK 2.1. *Although we do not consider in this paper the updating problem to dimension $k+l$ for $l > 1$, it can be done in a very similar manner. If appropriately implemented, this "block" version still has $\theta(mkn)$ complexity. Convergence results are essentially the same and good performance can be expected on parallel architectures (see also* [2]*).*

3. Updating a two-sided decomposition. The algorithm above yields at step i an approximation $Q_{(i)}$ of the dominant left singular subspace of $A(:, 1:i)$, but in several applications it makes sense to update simultaneously an approximation of the corresponding right singular subspace of this matrix. This can be done with little extra cost.

We start from the notation introduced in (2.3), which we rewrite as

$$A(:, 1:k)V_{(k)} = Q_{(k)}R_{(k)}, \tag{3.1}$$

where $V_{(k)} = I_k$. We show by induction that at each step $i \geq k$ we have a decomposition

$$A(:, 1:i)V_{(i)} = Q_{(i)}R_{(i)}, \tag{3.2}$$

where $V_{(i)} \in \mathbb{R}^{i \times k}$ satisfies $V_{(i)}^T V_{(i)} = I_k$. From (3.1) it is obvious that this holds for $i = k$. For the induction step we start by assuming that it holds for $i-1$:

$$A(:, 1:(i-1))V_{(i-1)} = Q_{(i-1)}R_{(i-1)}.$$

We then append a column a_i to $A(:, 1:i-1)$ to get $A(:, 1:i)$ and obviously

$$A(:, 1:i) \begin{bmatrix} V_{(i-1)} & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} Q_{(i-1)}R_{(i-1)} & a_i \end{bmatrix}. \tag{3.3}$$

Now use (2.4), (2.5) to update this to

$$A(:,1\colon i)\begin{bmatrix} V_{(i-1)} & 0 \\ 0 & 1 \end{bmatrix} G_v = \begin{bmatrix} Q_{(i)}R_{(i)} & q_i\mu_i \end{bmatrix}. \tag{3.4}$$

Taking the first k columns of both sides of this equation yields (3.1) with

$$V_{(i)} = \begin{bmatrix} V_{(i-1)} & 0 \\ 0 & 1 \end{bmatrix} G_v \begin{bmatrix} I_k \\ 0 \dots 0 \end{bmatrix} \in \mathbb{R}^{i\times k}, \tag{3.5}$$

which obviously satisfies $V_{(i)}^T V_{(i)} = I_k$. The additional work for updating the approximation $V_{(i)}$ is just the multiplication (3.5), which requires $6ik$ flops and hence leads to a total of

$$\sum_{i=k}^{n} 6ik \approx 3k(n+k)(n-k+1)$$

additional flops for the full decomposition. This additional work can be neglected if $m \gg k$.

We terminate this section by writing a decomposition for the matrix $A(:,1:i)$ if we would not delete the last column at each step. There exists an orthogonal matrix $V_i \in \mathbb{R}^{i\times i}$ embedding $V_{(i)}$:

$$V_i = \begin{bmatrix} V_{(i)} & V_{(i)}^{\perp} \end{bmatrix}.$$

Choosing appropriate basis vectors for $V_{(i)}^{\perp}$, we obtain a decomposition of the type

$$A(:,1\colon i)V_i = \begin{bmatrix} Q_{(i)}R_{(i)} & \tilde{q}_i & \dots & \tilde{q}_n \end{bmatrix}, \tag{3.6}$$

where $\tilde{q}_j = q_j\mu_j$ and $\|\tilde{q}_j\|_2 = \mu_j$. From this we obtain the additive decomposition

$$A(:,1\colon i) = Q_{(i)}R_{(i)}V_{(i)}^T + \begin{bmatrix} \tilde{q}_i & \dots & \tilde{q}_n \end{bmatrix} V_{(i)}^{\perp T}, \tag{3.7}$$

which will be used later on to derive error bounds.

4. Accuracy bounds. It is clear that after the first step $i = k+1$ we obtain a decomposition

$$[A(:,1\colon k{+}1)]\, G_v^T = \begin{bmatrix} Q_{(k+1)} & q_{k+1} \end{bmatrix} \cdot \begin{bmatrix} R_{(k+1)} & 0 \\ 0 & \mu_{k+1} \end{bmatrix}. \tag{4.1}$$

Let σ_i, $i = 1,\dots,n$, be the singular values of A and $\hat{\sigma}_i^{(j)}$, $i = 1,\dots,k$, those of $R(j)$. Then according to the above decomposition, $A(:,1\colon k+1)$ has singular values

$$\hat{\sigma}_1^{(k+1)}, \dots, \hat{\sigma}_k^{(k+1)}, \mu_{k+1}.$$

But since this is a submatrix of A obtained by deleting a number of columns, we have the inequalities [8]

$$\hat{\sigma}_1^{(k+1)} \le \sigma_1, \quad \dots, \quad \hat{\sigma}_k^{(k+1)} \le \sigma_k, \quad \mu_{k+1} \le \sigma_{k+1}. \tag{4.2}$$

Similarly one easily shows that each intermediate matrix

$$\begin{bmatrix} Q_{(i)} & q_i \end{bmatrix} \cdot \begin{bmatrix} R_{(i)} & 0 \\ 0 & \mu_i \end{bmatrix} \tag{4.3}$$

with singular values

$$\hat{\sigma}_1^{(i)}, \ldots, \hat{\sigma}_k^{(i)}, \mu_i$$

is also orthogonally equivalent to a submatrix of A. Therefore we have in general

$$\hat{\sigma}_1^{(i)} \leq \sigma_1, \quad \ldots, \quad \hat{\sigma}_k^{(i)} \leq \sigma_k, \quad \mu_i \leq \sigma_{k+1}. \tag{4.4}$$

Finally, since the matrix

$$\begin{aligned} \left[\begin{array}{cc} A(:, 1\colon (i-1)) & a_i \end{array} \right] &= \left[\begin{array}{cc} Q_{(i-1)} R_{(i-1)} & Q_{(i-1)} r_i + \hat{q}_i \rho_i \end{array} \right] \\ &= \left[\begin{array}{cc} Q_{(i)} & q_i \end{array} \right] \left[\begin{array}{cc} R_{(i)} & 0 \\ 0 & \mu_i \end{array} \right] G_v^T \end{aligned} \tag{4.5}$$

has singular values $\hat{\sigma}_1^{(i)}, \ldots, \hat{\sigma}_k^{(i)}, \mu_i$ and $Q_{(i-1)} R_{(i-1)}$ is its submatrix, we have the inequalities

$$\hat{\sigma}_1^{(i-1)} \leq \hat{\sigma}_1^{(i)}, \quad \ldots \ , \quad \hat{\sigma}_k^{(i-1)} \leq \hat{\sigma}_k^{(i)}. \tag{4.6}$$

All this says that the singular values μ_i that are dismissed at each step are all smaller than σ_{k+1} and that the singular values $\hat{\sigma}_j^{(i)}$, $j = 1, \ldots, k$, that are updated increase monotonically towards the first k singular values of A. To obtain bounds at the end of the iterative procedure we need to relate A to the computed quantities. For this, we point out that there exists an orthogonal column transformation V which relates A and the intermediate results of the recursive algorithm:

$$AV_n = \left[\begin{array}{cccc} Q_{(n)} R_{(n)} & \mu_{k+1} q_{k+1} & \ldots & \mu_n q_n \end{array} \right]. \tag{4.7}$$

The transformation V_n indeed consists of all the smaller transformations G_v and appropriately chosen permutations to obtain (4.7). Using the singular value decomposition of $R_{(n)}$,

$$R_{(n)} = \hat{U}_n \Sigma \hat{V}_n^T,$$

one then constructs orthogonal transformations such that

$$AV_n \left[\begin{array}{cc} \hat{V}_n & 0 \\ 0 & I \end{array} \right] = \left[\begin{array}{cc} Q_{(n)} \hat{U}_n & Q_{(n)}^{\perp} \end{array} \right] \left[\begin{array}{cc} \hat{\Sigma} & A_{1,2} \\ 0 & A_{2,2} \end{array} \right], \tag{4.8}$$

where $Q_{(n)}^{\perp}$ is orthogonal to $Q_{(n)}$ and where the columns of $A_2 \doteq \left[\begin{smallmatrix} A_{1,2} \\ A_{2,2} \end{smallmatrix} \right]$ have 2-norms μ_i. The Frobenius norm of this submatrix is therefore equal to $\| [\mu_{k+1}, \ldots, \mu_n] \|_2$. From (4.8) one already finds a bound for the accuracy of the computed singular values. The singular values of A are also those of $M \doteq \left[\begin{smallmatrix} \hat{\Sigma} & A_{1,2} \\ 0 & A_{2,2} \end{smallmatrix} \right]$. Applying the Wielandt–Hoffman theorem for singular values to this [8] yields

$$\sum_{i=1}^{k} (\sigma_i - \hat{\sigma}_i^{(n)})^2 \leq \|A_2\|_F^2 = \sum_{i=k+1}^{n} (\mu_i)^2 \leq (n-k) \cdot \sigma_{k+1}^2. \tag{4.9}$$

If we know the singular values have a considerable gap $\gamma \doteq \sigma_k - \sigma_{k+1}$, then this bound says that the k largest singular values are well approximated. If γ is large, the space

spanned by the corresponding singular vectors is also insensitive to perturbations. Moreover, one can improve the bounds for the singular value perturbations provided by the Wielandt–Hoffman theorem. To analyze this in more detail we use the following theorem proven in [10].

THEOREM 4.1. *Let $\hat{H}$ and E be square Hermitian matrices partitioned as*

$$\hat{H} = \begin{bmatrix} \hat{H}_{1,1} & 0 \\ 0 & \hat{H}_{2,2} \end{bmatrix}, \quad E = \begin{bmatrix} E_{1,1} & E_{1,2} \\ E_{2,1} & E_{2,2} \end{bmatrix},$$

and define $\epsilon = \|E_{1,2}\|_2$ and $\delta = \min|\lambda(\hat{H}_{1,1}) - \lambda(\hat{H}_{2,2})| - \|E_{1,1}\|_2 - \|E_{2,2}\|_2$.

If $\delta > 2\epsilon$, then there exists a unitary matrix X of the form

$$X = \begin{bmatrix} I_k & -P^T \\ P & I_{n-k} \end{bmatrix} \begin{bmatrix} (I + P^T P)^{-1/2} & 0 \\ 0 & (I + PP^T)^{-1/2} \end{bmatrix}$$

such that

$$H \doteq X^T(\hat{H} + E)X = \begin{bmatrix} H_{1,1} & 0 \\ 0 & H_{2,2} \end{bmatrix},$$

where $\|P\|_2 < 2\epsilon/\delta$.

This theorem is used to estimate the accuracy of both the left and right dominant subspaces of A as follows. Suppose

$$\hat{H}_u = \begin{bmatrix} \hat{\Sigma}^2 & 0 \\ 0 & 0 \end{bmatrix} \tag{4.10}$$

is the current "approximation" of the eigenvalue decomposition of

$$H_u \doteq MM^T = \begin{bmatrix} \hat{\Sigma}^2 & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} A_{1,2} \\ A_{2,2} \end{bmatrix} \begin{bmatrix} A_{1,2}^T & A_{2,2}^T \end{bmatrix}. \tag{4.11}$$

The left dominant "singular" subspace of M is also the dominant eigensubspace of H_u. The dominant eigensubspace of the nearby matrix $\hat{H}_u$ is clearly $\mathrm{Im}\left[\begin{smallmatrix} I_k \\ 0 \end{smallmatrix}\right]$ and the corresponding eigenvalues are the diagonal elements $\hat{\sigma}_1^{(n)}, \ldots, \hat{\sigma}_k^{(n)}$ of $\hat{\Sigma}^2$. But due to the perturbations $A_{1,2}$ and $A_{2,2}$ these are incorrect. After transforming MM^T to $X_u^T MM^T X_u$ we obtain its true eigenvalues (i.e., the squared singular values of M) in the matrix $H_{1,1}$ and the true dominant subspace as $\mathrm{Im}\left[\begin{smallmatrix} I_k \\ P_u \end{smallmatrix}\right]$. The norm of P_u is a measure for the angular rotation of this subspace, and it is bounded by $2\epsilon_u/\delta_u$. The largest canonical angle θ_k between the spaces $\mathrm{Im}\left[\begin{smallmatrix} I_k \\ 0 \end{smallmatrix}\right]$ and $\mathrm{Im}\left[\begin{smallmatrix} I_k \\ P_u \end{smallmatrix}\right]$ in fact satisfies [10]

$$\cos\theta_k = 1/\sqrt{1 + \|P_u\|^2}, \quad \sin\theta_k = \|P_u\|/\sqrt{1 + \|P_u\|^2}, \quad \tan\theta_k = \|P_u\|$$

and measures the "rotation" of the dominant subspace with respect to its approximation.

Clearly here $\epsilon_u = \|A_{1,2}A_{2,2}^T\|_2$ and $\delta_u = (\hat{\sigma}_k^{(n)})^2 - \|A_{1,1}\|_2^2 - \|A_{2,2}\|_2^2$. Notice that $\|A_2\|_F^2 = \sum_i \mu_i^2$ and that we actually compute these values during our recursive calculations. It would therefore be convenient to bound $2\epsilon_u/\delta_u$ in terms of these "discarded" singular values μ_i. One easily derives the bounds

$$\|A_{1,2}A_{2,2}^T\|_2 \leq \frac{1}{2}\left\| \begin{bmatrix} A_{1,2} \\ A_{2,2} \end{bmatrix} \begin{bmatrix} A_{1,2}^T & A_{2,2}^T \end{bmatrix} \right\|_2 = \frac{1}{2}\left\| \underbrace{\begin{bmatrix} A_{1,2} \\ A_{2,2} \end{bmatrix}}_{A_2} \right\|_2^2$$

and

$$\|A_2\|_2^2 \leq \|A_{1,2}A_{1,2}^T\|_2 + \|A_{2,2}A_{2,2}^T\|_2 = \|A_{1,2}^T A_{1,2}\|_2 + \|A_{2,2}^T A_{2,2}\|_2 \leq 2\|A_2\|_2^2.$$

Defining

$$\mu \doteq \left\| \begin{bmatrix} A_{1,2} \\ A_{2,2} \end{bmatrix} \right\|_2 \tag{4.12}$$

we then have

$$\epsilon_u \leq \mu^2/2, \quad (\hat{\sigma}_k^{(n)})^2 - \mu^2 \geq \delta_u \geq (\hat{\sigma}_k^{(n)})^2 - 2\mu^2, \tag{4.13}$$

and provided that $\hat{\sigma}_k^{(n)} \geq \sqrt{3}\mu$ we obtain

$$\delta_u \geq 2\epsilon_u \Rightarrow \|P_u\|_2 \leq 2\epsilon_u/\delta_u.$$

For the right dominant singular subspace of M we must consider

$$H_v \doteq M^T M = \begin{bmatrix} \hat{\Sigma}^2 & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & \hat{\Sigma}A_{1,2} \\ A_{1,2}^T\hat{\Sigma} & A_{1,2}^T A_{1,2} + A_{2,2}^T A_{2,2} \end{bmatrix}. \tag{4.14}$$

For the quantities ϵ_v and δ_v corresponding to Theorem 4.1, we find

$$\epsilon_v \doteq \|\hat{\Sigma}A_{1,2}\|_2 \leq \mu\|A\|_2, \quad \delta_v \doteq \min|\lambda(\hat{\Sigma}^2)| - \|A_2\|_2^2 = (\hat{\sigma}_k^{(n)})^2 - \mu^2.$$

Provided that $(\hat{\sigma}_k^{(n)})^2 \geq \frac{16}{7}\mu\|A\|_2$ we obtain

$$\delta_v \geq 2\epsilon_v \Rightarrow \|P_v\|_2 \leq 2\epsilon_v/\delta_v.$$

Applying the same reasoning as above we denote the true dominant subspace as $\mathrm{Im}\left[\begin{smallmatrix} I_k \\ P_v \end{smallmatrix}\right]$. The norm of P_v is then a measure for the angular rotation of this subspace, and it is bounded by $2\epsilon_v/\delta_v$. The corresponding largest canonical angle ϕ_k satisfies again [10]

$$\cos\phi_k = 1/\sqrt{1+\|P_v\|^2}, \quad \sin\phi_k = \|P_v\|/\sqrt{1+\|P_v\|^2}, \quad \tan\phi_k = \|P_v\|$$

and measures the "rotation" of the right dominant singular subspace with respect to its approximation. We summarize this discussion in the following theorem.

THEOREM 4.2. *Let*

$$\hat{M} = \begin{bmatrix} \hat{\Sigma} & 0 \\ 0 & 0 \end{bmatrix}, \quad M = \begin{bmatrix} \hat{\Sigma} & A_{1,2} \\ 0 & A_{2,2} \end{bmatrix}, \quad \mu \doteq \left\| \begin{bmatrix} A_{1,2} \\ A_{2,2} \end{bmatrix} \right\|_2.$$

Then the angles θ_k and ϕ_k between the k-dimensional left and right singular subspaces of M and $\hat{M}$, respectively, satisfy the bounds

$$\tan\theta_k < \mu^2/((\sigma_k^{(n)})^2 - 2\mu^2) \qquad \text{if} \quad \mu < \sigma_k^{(n)}/\sqrt{3}$$

and

$$\tan\phi_k < \mu\|M\|_2/((\sigma_k^{(n)})^2 - \mu^2) \quad \text{if} \quad \mu < 7(\sigma_k^{(n)})^2/16\|A\|_2.$$

These are also the angles of the left and right singular subspaces of $Q_{(i)}R_{(i)}V_{(i)}^T$ and A.

Unfortunately, we do not compute the matrices $A_{1,2}$ and $A_{2,2}$, and so we have to estimate μ. Bounding μ^2 in terms of the Frobenius norm

$$\mu^2 \leq \sum_i \mu_i^2$$

would yield serious overestimates since δ may become negative. Therefore we have to make some simplifying assumptions. The ith column of A_2 at step i of the recursive calculation contains what could be considered "residual noise vectors," and we assume therefore that they are randomly distributed. It is shown in [7] that an $(n-k) \times n$ matrix B with elements chosen independently from a standard Gaussian distribution has column norms tending to $\sqrt{n}$ and a spectral norm $\|B\|_2$ tending to $\sqrt{n}(1 + \sqrt{(n-k)/n})$ as n becomes large. If our matrix A_2 has equal column norms (hence equal to $\max_i \mu_i$ rather than $\sqrt{n}$), we then obtain the approximation

$$\max_i \mu_i \leq \mu \leq c. \max_i \mu_i, \quad c \approx (1 + \sqrt{(n-k)/n}).$$

On the other hand, if the columns are of very different norm, one gets closer to the lower bound since the number of relevant columns entering the above analysis becomes smaller than $(n-k)$, and thus c tends to 1. We will simply use $\hat{\mu} = \max_i \mu_i$ and $\hat{\sigma}_1^{(n)}$, respectively, as estimates of μ and $\|A\|_2$, which leads to the following approximations for our bounds:

$$\hat{\epsilon}_u \approx \hat{\mu}^2/2, \quad \hat{\delta}_u \approx (\hat{\sigma}_k^{(n)})^2 - \hat{\mu}^2, \quad \hat{\epsilon}_v \approx \hat{\mu}\hat{\sigma}_1^{(n)}, \quad \hat{\delta}_v \approx (\hat{\sigma}_k^{(n)})^2 - \hat{\mu}^2.$$

Notice that these approximations have the advantage that $\hat{\delta}_u$ and $\hat{\delta}_v$ will always be positive since $\sigma_k^{(n)} \geq \sigma_{k+1}^{(i)} = \mu_i$. The resulting estimates for the norm of P_u and P_v then become

$$\|P_u\|_2 \approx \tan\hat{\theta}_k \doteq 2\frac{\hat{\epsilon}_u}{\hat{\delta}_u} = \frac{\hat{\mu}^2}{(\hat{\sigma}_k^{(n)})^2 - \hat{\mu}^2}, \tag{4.15}$$

$$\|P_v\|_2 \approx \tan\hat{\phi}_k \doteq 2\frac{\hat{\epsilon}_u}{\hat{\delta}_u} = \frac{\hat{\mu}\hat{\sigma}_1^{(n)}}{(\hat{\sigma}_k^{(n)})^2 - \hat{\mu}^2}. \tag{4.16}$$

It is possible to estimate the quality of the computed singular values using a simpler analysis. From Theorem 4.1 it follows that

$$N \begin{bmatrix} I + P^T \end{bmatrix} \left(\begin{bmatrix} \hat{\Sigma}^2 & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} A_{1,2} \\ A_{2,2} \end{bmatrix} \begin{bmatrix} A_{1,2}^T & A_{2,2}^T \end{bmatrix} \right) \begin{bmatrix} I \\ P \end{bmatrix} N = H_{1,1}, \tag{4.17}$$

where

$$N = (I + P^T P)^{-\frac{1}{2}}, \quad N = N^T \leq I.$$

This yields the residual equation

$$H_{1,1} - N\hat{\Sigma}^2 N = R \doteq N \begin{bmatrix} I & P^T \end{bmatrix} \begin{bmatrix} A_{1,2} \\ A_{2,2} \end{bmatrix} \begin{bmatrix} A_{1,2}^T & A_{2,2}^T \end{bmatrix} \begin{bmatrix} I \\ P \end{bmatrix} N,$$

and since

$$N\hat{\Sigma}^2 N \leq \hat{\Sigma}^2$$

we have

$$H_{1,1} - \hat{\Sigma}^2 \leq H_{1,1} - N\hat{\Sigma}^2 N = R.$$

But

$$\|R\|_2 = \left\| \begin{bmatrix} A_{1,2} \\ A_{2,2} \end{bmatrix} \right\|_2^2 = \mu^2,$$

from which we obtain the strict bound

$$|\sigma_i^2 - (\hat{\sigma}_i^{(n)})^2| \leq \|H_{1,1} - \hat{\Sigma}^2\|_2 \leq \mu^2.$$

This analysis is very simple and does not take into account any information about P, which can be used to improve the bound. Instead, we replace μ by its estimate $\hat{\mu}$, which yields

$$|\sigma_i - \hat{\sigma}_i^{(n)}| \approx \hat{\mu}^2/(\sigma_i + \hat{\sigma}_i^{(n)}) \leq \hat{\mu}^2/2\hat{\sigma}_i^{(n)}. \tag{4.18}$$

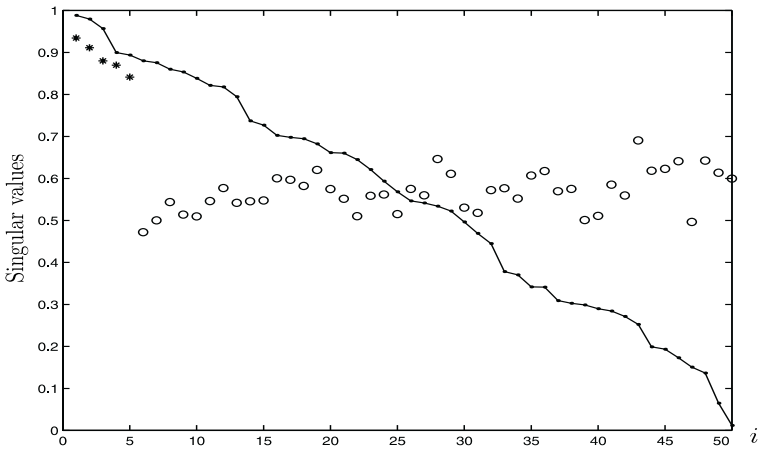
We point out that all of the estimates are quadratic in $\hat{\mu}$, which should give very accurate results if $\hat{\mu} \ll \hat{\sigma}_i^{(n)}$. This is the case if the gap γ at the kth singular value is large, and the quality of the estimate should be expected to deteriorate when this gap becomes small. We illustrate the quality of these bounds in the examples of the next section.

REMARK 4.1. *If A has rank k, then this approach produces an exact decomposition since each submatrix $A_{(i)}$ has rank less than or equal to k and hence $\mu_i = 0$ at each step.*

5. Numerical tests of the approximation. We generated random matrices of dimension $m = 1000$ by $n = 50$ and attempted to track the $k = 5$ dominant singular values and vectors. At every step we keep at most $k + 1 = 6$ vectors in our basis. We thus update to a subspace of dimension 6 and then deflate the smallest singular value to fall back to a space of dimension 5 at each step.

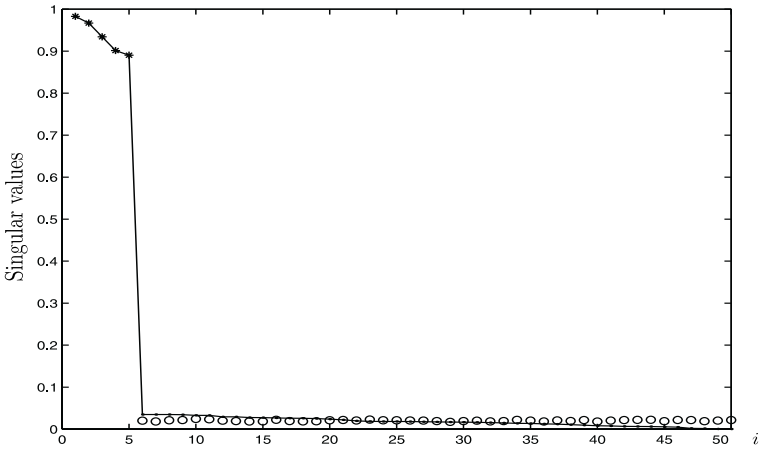
In Figures 1 and 2, the true singular values σ_i $(i = 1, \ldots, n)$ are represented by the solid line, the approximations $\sigma_i^{(n)}$ of the $i = 1, \ldots, k$ leading singular values are the asterisks, and the dismissed singular values μ_i $(i = k + 1, \ldots, n)$ are the circles. Two different gaps are used to illustrate the trend of a larger gap improving the quality of the approximations. Both figures are accompanied by a table (see Tables 1 and 2) listing the singular values σ_i, their approximations $\hat{\sigma}_i^{(n)}$, the corresponding errors $|\sigma_i - \hat{\sigma}_i^{(n)}|$ and their estimate $\hat{\mu}^2/(2\hat{\sigma}_i^{(n)})$, and finally the cosines of the canonical angles $\cos\theta_i$ and $\cos\phi_i$, the smallest of which indicate the rotation of the dominant left and right singular subspaces versus their approximation, and the estimated angles $\cos\hat{\theta}_k$ and $\cos\hat{\phi}_k$. We also give the true value of μ, its estimate $\hat{\mu}$, and finally the $k + 1$ singular value.

From these examples it appears that the method works reasonably well. It should be pointed out that Theorem 4.2 applies only to the second example and that the estimates are very good. Nevertheless the estimates are still acceptable even when the conditions of this theorem do not apply, as is shown by the first example, which has virtually *no* gap! Notice that $\mu/\hat{\mu}$ remains smaller than 2, as suggested by the statistical arguments of section 4. We also analyzed intermediate values of γ, which confirmed the remarks made above.



—— true sv's $\sigma_i(A)$, $*$ approximated sv's $\hat{\sigma}_1^{(n)}, \ldots, \hat{\sigma}_k^{(n)}$, $\circ$ dismissed sv's $\mu_{k+1}, \ldots, \mu_n$

FIG. 1. *Matrix with small gap* $\gamma = 0.01375$.



—— true sv's $\sigma_i(A)$, $*$ approximated sv's $\hat{\sigma}_1^{(n)}, \ldots, \hat{\sigma}_k^{(n)}$, $\circ$ dismissed sv's $\mu_{k+1}, \ldots, \mu_n$

FIG. 2. *Matrix with large gap* $\gamma = 0.85541$.

6. The effect of round-off. In this section we analyze the propagation of round-off in the proposed algorithm. The first aim is to prove some kind of backward stability of the algorithm. We show that at each step i the algorithm produces "approximate" matrices $\bar{V}_{(i)}, \bar{Q}_{(i)}$, and $\bar{R}_{(i)}$ that satisfy exactly the perturbed equations

$$[A(:, 1 \ : i) + E]\bar{V}_{(i)} = \bar{Q}_{(i)}\bar{R}_{(i)}, \quad (\bar{V}_{(i)} + F)^T(\bar{V}_{(i)} + F) = I_k, \tag{6.1}$$

where

$$\|E\|_F \leq \epsilon_e \|A\|_2, \ \ \epsilon_e \approx u, \quad \|F\|_F \leq \epsilon_f \approx u,$$

TABLE 1

σ_i	$\hat{\sigma}_i^{(n)}$	$\lvert\sigma_i - \hat{\sigma}_i\rvert$	$\frac{\hat{\mu}^2}{(2\hat{\sigma}_i^{(n)})}$	$\cos\theta_i$	$\cos\hat{\theta}_i$	$\cos\phi_i$	$\cos\hat{\phi}_i$
0.98833	0.93436	0.05398	0.27320	0.97419	0.36164	0.95272	0.34189
0.97975	0.91122	0.06852	0.28725	0.94833	0.11482	0.91511	0.10679
0.95684	0.87986	0.07698	0.30809	0.88082	0.04148	0.84415	0.03815
0.89977	0.86969	0.03008	0.31534	0.80644	0.11320	0.75753	0.10941
0.89390	0.84136	0.05253	0.33693	0.16487	0.27966	0.14274	0.26322

$\mu = 0.97905$	$\hat{\mu} = 0.69067$	$\sigma_{k+1} = 0.88014$

TABLE 2

σ_i	$\hat{\sigma}_i^{(n)}$	$\lvert\sigma_i - \hat{\sigma}_i\rvert$	$\frac{\hat{\mu}^2}{(2\hat{\sigma}_i^{(n)})}$	$\cos\theta_i$	$\cos\hat{\theta}_i$	$\cos\phi_i$	$\cos\hat{\phi}_i$
0.98299	0.98299	$2.0\ 10^{-7}$	0.00030	0.99999	0.99999	0.99999	0.99999
0.96689	0.96689	$1.0\ 10^{-7}$	0.00032	0.99999	0.99999	0.99999	0.99999
0.93424	0.93424	$1.0\ 10^{-7}$	0.00034	0.99999	0.99999	0.99999	0.99999
0.90161	0.90161	$0.5\ 10^{-7}$	0.00036	0.99999	0.99999	0.99999	0.99999
0.89032	0.89032	$1.5\ 10^{-7}$	0.00037	0.99999	0.99999	0.99999	0.99999

$\mu = 0.03491$	$\hat{\mu} = 0.02430$	$\sigma_{k+1} = 0.03491$

in which u is the so-called unit round-off of the IEEE floating point standard (see, e.g., [9]). This is used to prove that the effect of round-off remains small despite the fact that this is a classical Gram–Schmidt procedure.

The proof of the following theorem is given in the appendix.

THEOREM 6.1. *The recursive algorithm described in sections* 2 *and* 3 *produces "approximate" matrices* $\bar{V}_{(i)}, \bar{Q}_{(i)}$, *and* $\bar{R}_{(i)}$ *that satisfy* exactly *the perturbed equation* (6.1) *with the bounds (up to* $O(u^2)$ *terms)*

$$\|E\|_F \leq \epsilon_e \|A\|_2, \quad \epsilon_e \leq 26k^{\frac{3}{2}}nu, \quad \|F\|_F \leq \epsilon_f \leq 9k^{\frac{3}{2}}nu.$$

We point out here that these bounds do *not* depend on m, the largest dimension of A. Moreover, if one uses Householder transformations rather than Givens transformations, the results are very similar.

REMARK 6.1. *Although Theorem* 6.1 *indicates that the error* $\|E\|_F$ *grows with the number of columns* n, *it does not seem to grow in actual experiments. This can be explained as follows. Assume that at step* i *we have the perturbed equation*

$$\begin{bmatrix} Q_{(i-1)} + E_{(i-1)} & \hat{q}_i + e_i \end{bmatrix} G_u = \begin{bmatrix} Q_{(i)} + E_{(i)} & q_i + g_i \end{bmatrix}, \tag{6.2}$$

where $E_{(i)}$ *accounts for the loss of orthogonality in* $Q_{(i)}$, *and* e_i *is the local error in the vector* $\hat{q}_i$, *and* g_i *is the resulting error in the vector* q_i. *If we assume the errors in the right-hand side of* (6.2) *to be evenly distributed over the matrix, then it follows that*

$$\|E_{(i)}\|_F^2 \leq \frac{k}{(k+1)}\|E_{(i-1)}\|_F^2 + \|e_i\|_2^2, \tag{6.3}$$

which for growing i *tends to a limit*

$$\|E\|_F^2 \leq (k+1)\max_i \|e_i\|_2^2$$

that is independent of n. The same reasoning can be applied to the error $\|F\|_F$. *The corresponding bounds of Theorem* 6.1 *become*

$$\epsilon_e \le 26k^2u, \quad \epsilon_f \le 9k^2u.$$

We now turn our attention to the loss of orthogonality in the computed matrix $\bar{Q}$. This can be bounded using a perturbation result for the QR factorization of

$$(A+E)\bar{V} = A\bar{V} + E\bar{V} \doteq A\bar{V} + G,$$

where, using the bounds of Theorem 6.1, we have

$$\|G\|_F = \epsilon_g\|A\|_2, \quad \epsilon_g \le \epsilon_e + O(\epsilon_e\epsilon_f) \approx u.$$

THEOREM 6.2. *Let (a given matrix)* $\bar{V} \in \mathcal{R}^{n\times k}$ *"select" k columns of the matrix* $A \in \mathcal{R}^{m\times n}$, *and let*

$$A\bar{V} = QR, \quad Q^TQ = I_k,$$

with R upper triangular, be its exact QR factorization. Let

$$A\bar{V} + G = \bar{Q}\bar{R}, \quad \|G\|_F = \epsilon_g\|A\|_2 \approx u\|A\|_2 \tag{6.4}$$

be a "computed" version, where $\bar{Q} = Q + \Delta_Q$, $\bar{R} = R + \Delta_R$. *Then under a mild assumption, namely, condition* (6.6), *we can bound the loss of orthogonality in* $\bar{Q}$ *as follows:*

$$\|\bar{Q}^T\bar{Q} - I_k\|_F \le \sqrt{2}\epsilon_g\kappa_2(R)\kappa_R(A\bar{V}) \le 2\epsilon_g\kappa_2^2(R), \quad \epsilon_g \approx u.$$

Proof. Since $\bar{Q}$ is not necessarily orthogonal we first compute its QR factorization:

$$\bar{Q} = Q_0R_0, \quad Q_0^TQ_0 = I_k.$$

So we can consider the perturbation of the QR decomposition of $A\bar{V}$:

$$A\bar{V} = QR, \quad A\bar{V} + G = Q_0(R_0\bar{R}). \tag{6.5}$$

The loss of orthogonality in $\bar{Q}$ can be measured by R_0 since

$$\bar{Q}^T\bar{Q} - I_k = R_0^TQ_0^TQ_0R_0 - I_k = R_0^TR_0 - I_k.$$

To measure this, we first use a perturbation analysis of [6] for (6.5) to obtain

$$\|R_0\bar{R} - R\|_F \le \epsilon_g\kappa_R(A\bar{V})\|R\|_2,$$

where $\kappa_R(A\bar{V})$ is the "refined" condition number of the factor R of the QR factorization (6.5) of $A\bar{V}$ [6]. If we define $\Delta_0 \doteq R_0 - I_k$, we then have

$$R_0\bar{R} - R = (I_k + \Delta_0)(R + \Delta_R) - R = \Delta_0\bar{R} + \Delta_R \approx \Delta_0R + \Delta_R$$

and, hence,

$$\|\Delta_0R + \Delta_R\|_F \approx \|\Delta_0\bar{R} + \Delta_R\|_F \le \epsilon_g\kappa_2(R)\|R\|_2.$$

We now assume that there are no strong cancellations between $\|\Delta_R\|_F$ (measuring the perturbation of R) and $\|\Delta_0 R\|_F$ (measuring the perturbation in Q) and hence that $\|\Delta_0 R\|_F$ and $\|\Delta_R + \Delta_0 R\|_F$ are of the same order of magnitude:

$$\|\Delta_0 R\|_F \approx \|\Delta_R + \Delta_0 R\|_F. \tag{6.6}$$

From $\|\Delta_0 R\|_F \leq \epsilon_g \kappa_R(A\bar{V})\|R\|_2$ it then follows that

$$\|\Delta_0\|_F \leq \epsilon_g \kappa_R(A\bar{V})\|R\|_2\|R^{-1}\|_2.$$

This can now be used to bound $\|R_0^T R_0 - I_k\|_F = \|\Delta_0 + \Delta_0^T + \Delta_0^T\Delta_0\|_F \approx \sqrt{2}\|\Delta_0\|_F$, which yields

$$\|R_0^T R_0 - I_k\|_F \leq \sqrt{2}\epsilon_g \kappa_2(R)\kappa_R(A\bar{V}). \tag{6.7}$$

Using the overestimate $\kappa_R(A\bar{V}) \leq \sqrt{2}\kappa_2(R)$ of [6] we approximate this finally by

$$\|R_0^T R_0 - I_k\|_F \leq 2\epsilon_g \kappa_2^2(R). \qquad \square \tag{6.8}$$

REMARK 6.2. *Assumption* (6.6) *is crucial to the proof of Theorem* 6.2. *It is easy to see that any factorization of the type* (6.4) *will not yield the bounds* (6.7) *or* (6.8)*: consider, e.g., the factorization*

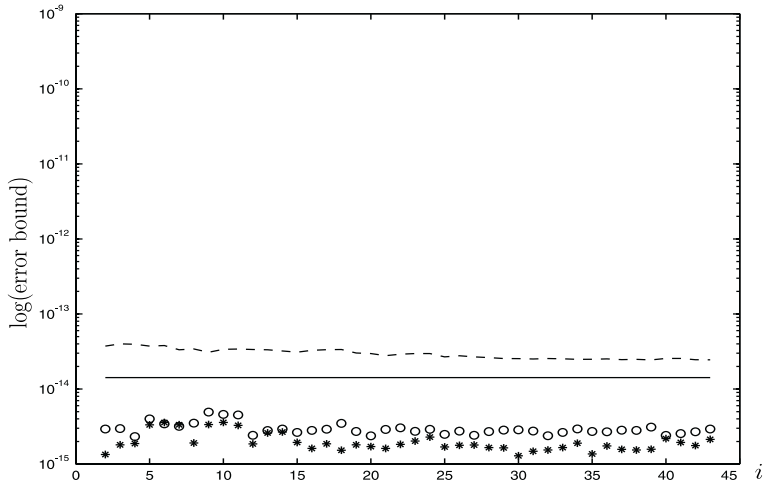
$$A\bar{V} + G = (\bar{Q}U)(U^{-1}\bar{R}),$$

where U is any invertible upper triangular matrix. This clearly satisfies the conditions of the theorem, except for assumption (6.6). *The critical quantity for this new factorization then becomes $\|U^T R_0^T R_0 U - I_k\|_F$, and since U can be chosen arbitrarily, it is impossible to bound it. Assumption* (6.6) *is therefore crucial, and we show in the next section that it indeed holds in practice.*

7. Numerical tests for the error propagation. In this section we present numerical evidence that the analysis of the previous section can be applied to the tracking problem of the dominant spaces of a given matrix. The numerical experiments we ran show that the loss of orthogonality in the computed matrix $\bar{Q}_{(i)}$ of (6.1) remains bounded by the condition number squared of the matrix R that we are "tracking."

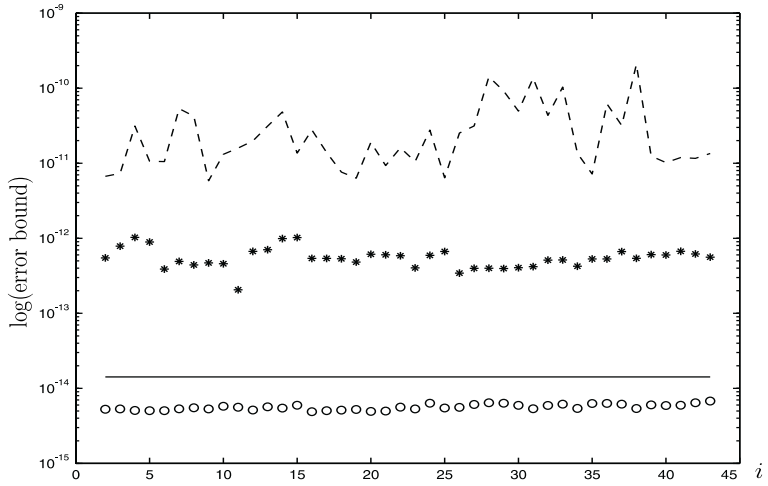
We show in Figures 3 and 4 two plots that compare the loss of orthogonality in the proposed algorithm based on the classical Gram–Schmidt method (labeled CGS) and a "fully orthogonal" method, which we obtain by performing *two* steps of CGS, rather than one, at each iteration. This second method, labeled CGS2, was analyzed in [1] and shown to yield a Q factor that is close to orthogonal. We chose this as an alternative to the Householder method because in the iterative scheme considered in this paper, CGS2 involves significantly fewer operations than the Householder method.

As suggested by Remark 6.1, the backward error $E_{(i)}$ and the quantity ϵ_e can be bounded independently of the step i. We therefore compare the loss of orthogonality $\|R_0^T R_0 - I_k\|_F$ with the quantities $uk^2\kappa_2(R_{(i)})\kappa_R(A(:, 1:i)\bar{V}_{(i)})$ for the CGS method and uk^2 for the CGS2 method. These "simplified" quantities are indicators to show that the loss of orthogonality is of the order of magnitude predicted by our error analysis. To show the effect of the condition number of the triangular factor $R_{(i)}$, we let it grow in the two examples by choosing a growing condition number for A.



– – CGS bound $uk^2\kappa_2(R_{(i)})\kappa_R(A(:,1:i)\bar{V}_{(i)}))$, —— CGS2 bound uk^2,
$*$ loss of orthogonality in CGS method, $\circ$ loss of orthogonality in CGS2 method

FIG. 3. $\kappa_2(A) = 41.806$, $\kappa_2(R_{(n)}) = 1.156$, $\kappa_R(A\bar{V}_{(n)})) = 1.492$.



– – CGS bound $uk^2\kappa_2(R_{(i)})\kappa_R(A(:,1:i)\bar{V}_{(i)}))$, —— CGS2 bound uk^2,
$*$ loss of orthogonality in CGS method, $\circ$ loss of orthogonality in CGS2 method

FIG. 4. $\kappa_2(A) = 6928$, $\kappa_2(R_{(n)}) = 134.7$, $\kappa_R(A\bar{V}_{(n)}) = 7.028$.

The following observations can be derived from these experiments:

- The condition numbers $\kappa_2(R_{(i)})$ and $\kappa_R(A(:,1:i)\bar{V}_{(i)})$ do not affect the loss of orthogonality of the CGS2 method, as expected from the analysis of [1]. (The product $\kappa_2(R_{(i)})\kappa_R(A(:,1:i)\bar{V}_{(i)})$ can be inferred from the gap between the CGS and CGS2 bounds.)
- The statistical assumption of Remark 6.1 seems to hold since there is no growth in the loss of orthogonality of the computed matrices $\bar{Q}_{(i)}$: this should

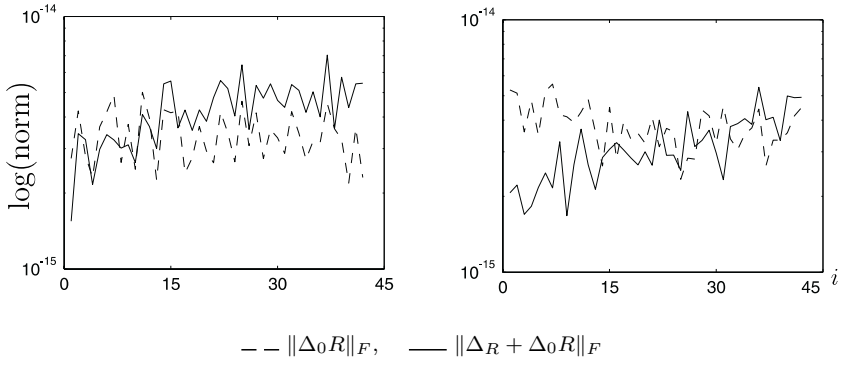


FIG. 5. *Verification of assumption* (6.6) *for examples Figures* 3 *and* 4.

depend on the backward error $E_{(i)}$, which does not depend on i if the assumption of Remark 6.1 holds

- Assumption (6.6) made in Theorem 6.2 was verified in these experiments and validates the resulting bounds (6.7), (6.8) of that theorem; the graphs in Figure 5 give the norms of the two quantities for the two examples given earlier and illustrate that the assumption that those quantities are of the same order of magnitude is reasonable.
- The loss of orthogonality remains very reasonable when the condition number $\kappa_2(R_{(i)})$ is not too large, which is a reasonable assumption in applications where a "dominant matrix" $R_{(i)}$ is being tracked.

We observed no difference in the computed spaces for the CGS or CGS2 methods. We conclude from our analysis and the experimental evidence that the cheapest version of the algorithm (CGS) can be used safely for the applications represented by the experiments and mentioned in section 1. By this we mean that the angles $\cos\theta_k$ and $\cos\phi_k$ for both methods were equal in the first four digits despite a very small loss of orthogonality in the CGS method.

8. Conclusions. In this paper we presented an analysis of an efficient incremental algorithm to compute the dominant subspace of a given matrix A. Although similar algorithms have been discussed in the literature [5], we have given here a more efficient implementation along with a fairly tight bound on its accuracy and estimators that can be used in practice to monitor that accuracy.

The contributions of this paper are the following:

- A CGS-like algorithm of complexity close to $8mnk$ flops was derived for computing a rank k approximation of an $m \times n$ matrix A.
- A posteriori bounds for the accuracy of the approximation error were presented and their reliability was illustrated.
- The effect of round-off was studied, and it was shown that the algorithm behaves much better than what can be expected for CGS. An explanation of this phenomenon was given and illustrated by numerical experiments. The effect of propagation of round-off errors was also analyzed and shown to be negligible for the applications considered in this paper.

Appendix. In this section we give the proof of Theorem 6.1. This result is obtained by analyzing one step i of the recursive algorithm. We first analyze the local errors in that step and hence assume all quantities at the beginning of step i to be

exact. For the computations of step i we use $\bar{x}$ to denote the "computed version" of x that is actually stored in the computer.

The first part of step i is the Gram–Schmidt update, which corresponds to

$$\bar{r}_i = fl(\bar{Q}_{(i-1)}^T a_i), \tag{A.1}$$

$$\bar{\tilde{q}}_i = fl(a_i - \bar{Q}_{(i-1)}^T \bar{r}_i), \tag{A.2}$$

$$\bar{\rho}_i = fl\left(\sqrt{\bar{\tilde{q}}_i^T \bar{\tilde{q}}_i}\right), \tag{A.3}$$

$$\bar{q}_i = fl(\bar{\tilde{q}}_i / \bar{\rho}_i). \tag{A.4}$$

From (A.2), (A.4), and standard error analysis results it follows that

$$\bar{\tilde{q}}_i = a_i + d_i - [\bar{Q}_{(i-1)} + \delta Q_{(i-1)}]\bar{r}_i = \bar{\rho}_i[\bar{q}_i + f_i], \tag{A.5}$$

where (up to order u^2) we have the elementwise inequalities

$$|[f_i]_j| \le u|[\bar{q}_i]_j|, \quad |[d_i]_j| \le ku|[a_i]_j|, \quad |[\delta Q_{(i-1)}]_{jl}| \le (k-l+2)u|[\bar{Q}_{(i-1)}]_{jl}|.$$

To obtain this result we assumed that the loop on the columns of the Gram–Schmidt orthogonalization (A.2) progresses from left to right. We can then equate this as follows:

$$a_i + e_i = \begin{bmatrix} \bar{Q}_{(i-1)} & \bar{q}_i \end{bmatrix} \begin{bmatrix} \bar{r}_i \\ \rho_i \end{bmatrix}, \quad e_i = d_i - \delta Q_{(i-1)}\bar{r}_i + f_i\bar{\rho}_i. \tag{A.6}$$

We also assume that

$$\| \begin{bmatrix} \bar{Q}_{(i-1)} & \bar{q}_i \end{bmatrix} - \begin{bmatrix} Q_{(i-1)} & q_i \end{bmatrix} \|_2 = K.u \ll 1, \tag{A.7}$$

i.e., there is no *complete* loss of orthogonality, which allows us to approximate the 2-norm of $\begin{bmatrix} \bar{Q}_{(i-1)} & \bar{q}_i \end{bmatrix}$ or any of its columns by $1 + O(u)$. We then obtain the inequalities

$$\begin{aligned}
\|e_i\|_2 &\le \|d_i\|_2 + \|f_i\bar{\rho}_i\|_2 + \sum_l \||\delta Q_{(i-1)}|_{:,l}\|_2 . |\bar{r}_i|_l + O(u^2) \\
&\le u\left[k\|a_i\|_2 + \|\bar{q}_i\|\bar{\rho}_i + \sum_l \||Q_{(i-1)}|_{:,l}\|_2 . (k-l+2)|r_i|_l\right] + O(u^2) \\
&\le u\left[k\|a_i\|_2 + \left(|\bar{\rho}_i| + \sum_l (k-l+2)|\bar{r}_i|_l\right)\right] + O(u^2) \\
&\le u(k\|a_i\|_2 + \|[1,2,\ldots,k+1]\|_2\|a_i\|_2) + O(u^2) \\
&\le u\left(k + \sqrt{\frac{(k+2)^3}{3}}\right)\|a_i\|_2 + O(u^2),
\end{aligned} \tag{A.8}$$

where the next-to-last line was obtained by Cauchy–Schwarz. Notice that all errors due to this part are superposed on column a_i. Therefore the error matrix E_1 of this first part satisfies $\|E_1\|_F = \|e_i\|_2$.

The second part of step i consists of the transformations G_v and G_u in (9), which we assume are each implemented with a sequence of k Givens rotations. For this we

will use Lemma 18.8 of [9], which we recall in a slightly modified form. (We refer to [9] for the details of the implementation and construction of each Givens rotation.)

LEMMA A.1. *Consider the sequence of Givens transformations*

$$M_k = G_k \cdot \ldots \cdot G_1 M = G \cdot M.$$

Then there exists a perturbation ΔM of M such that the computed matrix $\bar{M}_k$ satisfies

$$\bar{M}_k = G(M + \Delta M), \quad \|\Delta M\|_F \leq 6k\sqrt{2}u\|M\|_F + O(u^2).$$

Applying this to the products $Q_{up} \cdot R_{up} = (QG_u^T) \cdot (G_u R G_v^T)$ and $V_{up} = (VG_v^T)$ we obtain

$$\begin{aligned} \bar{Q}_{up}\bar{R}_{up} &= (Q + \Delta Q)G_u^T \cdot G_u(R + \Delta R)G_v^T \doteq QRG_v^T + E_2, \\ \bar{V}_{up} &= (V + \Delta V)G_v^T \doteq VG_v^T + F, \end{aligned}$$

where

$$\begin{aligned} E_2 &\doteq (\Delta QR + Q\Delta R + \Delta Q\Delta R)G_v^T, \\ \|\Delta Q\|_F &\leq 6\sqrt{2}ku\|Q\|_F + O(u^2) = 6k\sqrt{2(k+1)}u + O(u^2), \\ \|\Delta R\|_F &\leq 12\sqrt{2}ku\|R\|_F + O(u^2) = 12k\sqrt{2(k+1)}u\|A\|_2 + O(u^2), \end{aligned}$$

and

$$\begin{aligned} F &\doteq (\Delta V)G_v^T, \\ \|\Delta V\|_F &\leq 6\sqrt{2}ku\|V\|_F + O(u^2) = 6k\sqrt{2(k+1)}u + O(u^2). \end{aligned}$$

The norms of E_2 and F can then be bounded by

$$\begin{aligned} \|E_2\|_F &\leq \|Q\|_2\|\Delta R\|_F + \|R\|_2\|\Delta Q\|_F + O(u^2) \\ &\leq 18k\sqrt{2(k+1)}u\|A\|_2 + O(u^2), \\ \|F\|_F &\leq 6k\sqrt{2(k+1)}u + O(u^2). \end{aligned}$$

Combining the bounds for E_1 and E_2 yields the bound

$$\|E\|_F \ \leq \ 26uk^{\frac{3}{2}}\|A\|_2 + O(u^2)$$

for the local error E in step i. Similarly, the error matrix F on $V_{(i)}$ corresponding to the local errors of step i can be bounded by

$$\|F\|_F \ \leq \ 9uk^{\frac{3}{2}} + O(u^2).$$

In order to sum up these errors over the $n-k$ steps of the algorithm, we can neglect the second order effects and then only need to multiply these bounds by $(n-k)$. This then yields the bounds of Theorem 6.1.

Acknowledgment. We would like to thank A. Edelman for pointing out the work of Gemam to us.

REFERENCES

[1] N. Abdelmalek, *Round-off error analysis for Gram-Schmidt method and solution of linear least squares problems*, BIT, 11 (1971), pp. 45–68.

[2] C. G. Baker, *An Incremental Block Algorithm for Tracking Dominant Singular Subspaces*, Technical Report FSU-CSIT-03-03, CSIT, Florida State University, Tallahassee, FL, 2003.

[3] Y. Chahlaoui, K. Gallivan, and P. Van Dooren, *An incremental method for computing dominant singular spaces*, in Computational Information Retrieval, M. W. Berry, ed., SIAM, Philadelphia, 2001, pp. 53–62.

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

[5] S. Chandrasekaran, B. S. Manjunath, Y. F. Wang, J. Winkeler, and H. Zhang, *An eigenspace update algorithm for image analysis*, Graph. Models Image Process., 59 (1997), pp. 321–332.

[6] X.-W. Chang, C. C. Paige, and G. W. Stewart, *Perturbation analyses for the QR factorization*, SIAM J. Matrix Anal. Appl., 18 (1997), pp. 775–791.

[7] S. Geman, *A limit theorem for the norm of random matrices*, Ann. Probab., 8 (1980), pp. 252–261.

[8] G. H. Golub and C. Van Loan, *Matrix Computations*, Johns Hopkins University Press, Baltimore, MD, 1983.

[9] N. J. Higham, *Accuracy and Stability of Numerical Algorithms*, SIAM, Philadelphia, 1996.

[10] G. W. Stewart and J.-G. Sun, *Matrix Perturbation Theory*, Academic Press, San Diego, 1990.

[11] P. Van Dooren, *Gramian based model reduction of large-scale dynamical systems*, in Numerical Analysis 1999, Chapman Hall/CRC Press, London, 2000, pp. 231–247.