Principal Angles between Subspaces in an A -Based Scalar Product: Algorithms and Perturbation Estimates

Article *in* SIAM Journal on Scientific Computing · January 2002

DOI: 10.1137/S1064827500377332

CITATIONS
141

READS
177

2 authors, including:

Andrew Knyazev
University of Colorado
199 PUBLICATIONS **2,108** CITATIONS

SEE PROFILE

SIAM J. SCI. COMPUT.
Vol. 23, No. 6, pp. 2009–2041

PRINCIPAL ANGLES BETWEEN SUBSPACES IN AN *A*-BASED SCALAR PRODUCT: ALGORITHMS AND PERTURBATION ESTIMATES*

ANDREW V. KNYAZEV† AND MERICO E. ARGENTATI‡

Abstract. Computation of principal angles between subspaces is important in many applications, e.g., in statistics and information retrieval. In statistics, the angles are closely related to measures of dependency and covariance of random variables. When applied to column-spaces of matrices, the principal angles describe canonical correlations of a matrix pair. We highlight that all popular software codes for canonical correlations compute only cosine of principal angles, thus making impossible, because of round-off errors, finding small angles accurately. We review a combination of sine and cosine based algorithms that provide accurate results for all angles. We generalize the method to the computation of principal angles in an A-based scalar product for a symmetric and positive definite matrix A. We provide a comprehensive overview of interesting properties of principal angles. We prove basic perturbation theorems for absolute errors for sine and cosine of principal angles with improved constants. Numerical examples and a detailed description of our code are given.

Key words. principal angles, canonical correlations, subspaces, scalar product, orthogonal projection, algorithm, accuracy, round-off errors, perturbation analysis

AMS subject classifications. 65F30, 65G50, 68P20

PII. S1064827500377332

1. Introduction.

Let us consider two real-valued matrices F and G, each with n rows, and their corresponding column-spaces $\mathcal{F}$ and $\mathcal{G}$, which are subspaces in $\mathbf{R^n}$, assuming that

$$p = \dim\mathcal{F} \geq \dim\mathcal{G} = q \geq 1.$$

Then the *principal angles*

$$\theta_1, \ldots, \theta_q \in [0, \pi/2]$$

between $\mathcal{F}$ and $\mathcal{G}$ may be defined, e.g., [17, 13], recursively for $k = 1, \ldots, q$ by

$$\cos(\theta_k) = \max_{u \in \mathcal{F}} \max_{v \in \mathcal{G}} u^T v \;=\; u_k^T v_k$$

subject to

$$\|u\| = \|v\| = 1, \qquad u^T u_i = 0, \qquad v^T v_i = 0, \qquad i = 1, \ldots, k-1.$$

The vectors $u_1, \ldots, u_q$ and $v_1, \ldots, v_q$ are called principal vectors. Here and below $\|\cdot\|$ denotes the standard Euclidean norm of a vector or, when applied to a matrix, the corresponding induced operator norm, also called the spectral norm, of the matrix.

*Received by the editors August 23, 2000; accepted for publication (in revised form) November 6, 2001; published electronically February 27, 2002.
http://www.siam.org/journals/sisc/23-6/37733.html

†Department of Mathematics, University of Colorado at Denver, P.O. Box 173364, Campus Box 170, Denver, CO 80217-3364 (andrew.knyazev@cudenver.edu, http://www-math.cudenver.edu/~aknyazev)

‡Department of Mathematics, University of Colorado at Denver, P.O. Box 173364, Campus Box 170, Denver, CO 80217-3364 (rargenta@math.cudenver.edu).

According to [26, 23], the notion of canonical angles between subspaces goes back to Jordan (1875). Principal angles between subspaces, and particularly the smallest and the largest angles, serve as important tools in functional analysis (see books [1, 12, 18] and a survey [9]) and in perturbation theory of invariant subspaces, e.g., [6, 26, 24, 19, 22].

Computation of principal angles between subspaces is needed in many applications. For example, in statistics, the angles are closely related to measures of dependency and covariance of random variables; see a canonical analysis of [5]. When applied to column-spaces $\mathcal{F}$ and $\mathcal{G}$ of matrices F and G, the principal angles describe canonical correlations $\sigma_k(F,G)$ of a matrix pair, e.g., [17, 15], which is important in applications such as system identification and information retrieval. Principal angles between subspaces also appear naturally in computations of eigenspaces, e.g., [20, 21], where angles provide information about solution quality and need to be computed with high accuracy.

In such large-scale applications, it is typical that $n \gg p$; in other words, informally speaking, we are dealing with a small number of vectors having a large number of components. Because of this, we are interested in "matrix-free" methods; i.e., no n-by-n matrices need to be stored in memory in our algorithms.

A singular value decomposition (SVD)-based algorithm [11, 3, 4, 13, 15] for computing cosines of principal angles can be formulated as follows. Let columns of matrices $Q_F \in \mathbf{R}^{\mathbf{n}\times\mathbf{p}}$ and $Q_G \in \mathbf{R}^{\mathbf{n}\times\mathbf{q}}$ form orthonormal bases for the subspaces $\mathcal{F}$ and $\mathcal{G}$, respectively. The reduced SVD of $Q_F^T Q_G$ is

$$Y^T Q_F^T Q_G Z \;=\; \mathrm{diag}(\sigma_1, \sigma_2, \ldots, \sigma_q), \qquad 1 \geq \sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_q \geq 0, \tag{1.1}$$

where $Y \in \mathbf{R}^{\mathbf{p}\times\mathbf{q}}$, $Z \in \mathbf{R}^{\mathbf{q}\times\mathbf{q}}$ both have orthonormal columns. Then the principal angles can be computed as

$$\theta_k = \arccos(\sigma_k), \qquad k = 1, \ldots, q, \tag{1.2}$$

where

$$0 \leq \theta_1 \leq \cdots \leq \theta_q \leq \frac{\pi}{2},$$

while principal vectors are given by

$$u_k = Q_F y_k, \quad v_k = Q_G z_k, \quad k = 1, \ldots, q.$$

The equivalence [3, 13] of the original geometric definition of principal angles and the SVD-based approach follows from the next general theorem on an equivalent representation of singular values.

THEOREM 1.1. *If $M \in \mathbf{R}^{\mathbf{m}\times\mathbf{n}}$, then the singular values of M are defined recursively by*

$$\sigma_k = \max_{y \in \mathbf{R}^{\mathbf{m}}} \max_{z \in \mathbf{R}^{\mathbf{n}}} y^T M z \;=\; y_k^T M z_k, \qquad k = 1, \ldots, \min\{m,n\}, \tag{1.3}$$

subject to

$$\|y\| = \|z\| = 1, \quad y^T y_i = 0, \; z^T z_i = 0, \quad i = 1, \ldots, k-1. \tag{1.4}$$

The vectors y_i and z_i are, respectively, left and right singular vectors.

Proof. The proof of the theorem is straightforward if based on Allakhverdiev's representation (see [12]) of singular numbers,

$$\sigma_k = \left\| M - \sum_{i=1}^{k-1} v_i u_i^T \sigma_i \right\|,$$

and using the well-known formula of the induced Euclidean norm of a matrix as the norm of the corresponding bilinear form. ∎

To apply the theorem to principal angles, one takes $M = Q_F^T Q_G$.

In the most recent publication on the subject, [10], the SVD-based algorithm for cosine is proved to be mixed stable, and QR factorizations with the complete pivoting are recommended for computing Q_F and Q_G.

The SVD-based algorithm for cosine is considered as the standard one at present and is implemented in software packages, e.g., in MATLAB, version 5.3, 2000, code SUBSPACE.m, revision 5.5,[1] where $Q_F \in \mathbf{R^{n\times p}}$ and $Q_G \in \mathbf{R^{n\times q}}$ are computed using the QR factorization.

However, this algorithm cannot provide accurate results for small angles because of the presence of round-off errors. Namely, when using the standard double-precision arithmetic with $EPS \approx 10^{-16}$ the algorithm fails to accurately compute angles smaller than 10^{-8} (see section 2). The problem has been highlighted in the classical paper [3], as well as a cure has been suggested (see also publications on cosine-sine (CS) decomposition methods [25, 28, 26, 23]), but apparently it did not attract enough attention.

In statistics, most software packages include a code for computing $\sigma_k = cos(\theta_k)$, which are called *canonical correlations*; see, e.g., CANCOR Fortran code in FIRST MDS Package of AT&T, CANCR (DCANCR) Fortran subroutine in IMSL STAT/ LIBRARY, G03ADF Fortran code in NAG package, CANCOR subroutine in Splus, and CANCORR procedure in SAS/STAT Software. While accurately computing the cosine of principal angles in corresponding precision, these codes do not compute the sine. However, the cosine simply equals one in double precision for all angles smaller than 10^{-8} (see next section). Therefore, it is impossible in principal to observe an improvement in canonical correlations for angles smaller than 10^{-8} in double precision. It might not be typically important when processing experimental statistical data because the expected measurement error may be so great that a statistician would deem the highly correlated variable essentially redundant and therefore not useful as a further explanatory variable in their model. Statistical computer experiments are different, however, as there is no measurement error, so accurate computation of very high correlations may be important in such applications.

The largest principal angle is related to the notion of distance, or a gap, between equidimensional subspaces. If $p = q$, the distance is defined [1, 12, 13, 18] as

$$\text{gap}(\mathcal{F}, \mathcal{G}) = \|P_F - P_G\| = \sin(\theta_q) = \sqrt{1 - (\cos(\theta_q))^2}, \tag{1.5}$$

where P_F and P_G are orthogonal projectors onto $\mathcal{F}$ and $\mathcal{G}$, respectively.

This formulation provides insight into a possible alternative algorithm for computing the sine of principal angles. The corresponding algorithm, described in [3],

[1] Revision 5.8 of SUBSPACE.m in the current MATLAB release 12.1, version 6.1.0.450, May 18, 2001, is still identical to revision 5.5, which we have used for numerical tests in the present paper.

while being mathematically equivalent to the previous one in exact arithmetic, is accurate for small angles in computer arithmetic as it computes the sine of principal angles directly, without using SVD (1.1) leading to the cosine. We review the algorithm of [3] based on a general form of (1.5) in section 3 and suggest an improved version, similar to the CS decomposition algorithm of [28], with the second SVD of the reduced size.

The CS decomposition methods, e.g., [25, 28, 26, 23], one of which we just mentioned, provide a well-known and popular alternative approach for computing principal angles between subspaces given by selected p (q) columns of orthogonal matrices of the size n. For example, if the matrix $Q_{F^\perp}$, with orthonormal columns that span the subspace $\mathcal{F}^\perp$, the orthogonal complement of $\mathcal{F}$, is available to us, the CS decomposition methods compute the SVD of $(Q_F)^T Q_G$ together with the SVD of $(Q_{F^\perp})^T Q_G$, thus providing cosine and sine of principal angles. When p is of the same order as $n/2$, matrix $Q_{F^\perp}$ is about of the same size as matrix Q_F, and the CS decomposition methods are effective and are recommended for practical computations. However, when $n \gg p$, the CS decomposition methods, explicitly using matrix $Q_{F^\perp}$ of the size n-by-$n-p$, will be less efficient compared to "matrix-free" methods we consider in the present paper. Let us highlight that the cosine- and sine-based methods of [3] that we investigate here in section 3, while different algorithmically from the CS decomposition methods, are very close mathematically to them.

A different sine-based approach, using eigenvalues of $P_F - P_G$, is described in [4, 23]; see a similar statement of Theorem 3.4. It is also not attractive numerically, when $n \gg p$, as it requires computing an n-by-n matrix and finding all its nonzero eigenvalues.

In some applications, e.g., when solving symmetric generalized eigenvalue problems [20], the default scalar product $u^T v$ cannot be used and needs to be replaced with an A-based scalar product $(u, v)_A = u^T A v$, where A is a symmetric positive definite matrix. In statistics, a general scalar product for computing canonical correlations gives a user an opportunity, for example, to take into account a priori information that some vector components are more meaningful than others. In a purely mathematical setting, generalization to A-based scalar products brings nothing really new. In practical computations, however, it carries numerous algorithmic and numerical problems, especially for ill-conditioned cases, which are important in applications.

In section 4, we propose extension of the algorithms to an A-based scalar product and provide the corresponding theoretical justification.

In section 5, we turn our attention to perturbation estimates, which generalize the following trigonometric inequalities: if an angle $\theta \in [0, \pi/2]$ is perturbed by $\epsilon \in [0, \pi/2]$ such that $\theta + \epsilon \in [0, \pi/2]$, then

$$0 \leq \cos(\theta) - \cos(\theta + \epsilon) \leq \sin(\theta + \epsilon) \sin(\epsilon) \leq \sin(\epsilon),$$

$$0 \leq \sin(\theta + \epsilon) - \sin(\theta) \leq \cos(\theta) \sin(\epsilon) \leq \sin(\epsilon).$$

We prove new absolute perturbation estimates for the sine and cosine of principal angles computed in the A-based scalar product. When $A = I$, our estimates are similar to those of [3, 29, 27, 15, 14], but the technique we use is different. More importantly, our constants are somewhat better, in fact, in the same way the constants in the middle terms of the trigonometric inequalities above are less than one.

We consider particular implementation of algorithms used in our MATLAB code SUBSPACEA.m in section 6, with emphasis on the large-scale case, $n \gg p$, and sparse

ill-conditioned matrix A, which may be specified only as a function that multiplies A by a given vector. When matrices F and G are sparse, our code can still be used even though it performs orthogonalization of columns of matrices F and G that increases the fill-in; cf. [15]. Also, we do not even touch here upon a practically important issue of the possibility of recomputing the correlations with an increase in the data; see again [15].

Finally, numerical results, presented in section 7, demonstrate the practical robustness of our code.

For simplicity, we discuss only real spaces and real scalar products; however, all results can be trivially generalized to cover complex spaces as well. In fact, our code SUBSPACEA.m is written for the general complex case.

As pointed out by an anonymous referee, several natural questions are left unanswered here.

- Our algorithms are based on SVD. How does SVD accuracy (cf. [2, 8, 7, 10]), especially for small singular values, or in ill-conditioned cases, affect the results?
- In [10], a formal stability analysis is done for the SVD-based algorithm for cosine, which is proved to be mixed stable. In our numerical tests, practical robustness of our algorithms is encouraging. Are our methods accurate and stable theoretically, e.g., see [16]?
- For A-based scalar products, how does the increase of the condition number of A influence the accuracy? Which parts of our algorithm are responsible for the main error growth?

We feel, however, that investigating these matters is not within the limited scope of the present paper, which is already quite long. They may rather serve as interesting directions for future research.

2. Inaccuracy in the cosine-based algorithm. Let d be a constant and

$$\mathcal{F} = \operatorname{span}\left\{(1\ 0)^T\right\}, \quad \mathcal{G} = \operatorname{span}\left\{(1\ d)^T\right\}.$$

Then the angle between the one-dimensional subspaces $\mathcal{F}$ and $\mathcal{G}$ can be computed as

$$\theta = \arcsin\left(\frac{d}{\sqrt{1+d^2}}\right). \tag{2.1}$$

In the table below d varies from one to small values. Formula (2.1) is accurate for small angles, so we use it as an "exact" answer in the second column of the table. We use the MATLAB built-in function SUBSPACE.m (revision 5.5) which implements (1.1) to compute values for the third column of the table.

It is apparent that SUBSPACE.m returns inaccurate results for $d \le 10^{-8}$, which is approximately $\sqrt{EPS}$ for double precision.

d	Formula (2.1)	SUBSPACE.m
1.0	7.853981633974483e-001	7.853981633974483e-001
1.0e-04	9.999999966666666e-005	9.999999986273192e-005
1.0e-06	9.999999999996666e-007	1.000044449242271e-006
1.0e-08	1.000000000000000e-008	-6.125742274543099e-017
1.0e-10	1.000000000000000e-010	-6.125742274543099e-017
1.0e-16	9.999999999999998e-017	-6.125742274543099e-017
1.0e-20	9.999999999999998e-021	-6.125742274543099e-017
1.0e-30	1.000000000000000e-030	-6.125742274543099e-017

In this simple one-dimensional example the algorithm of SUBSPACE.m is reduced to computing

$$\theta = \arccos\left(\frac{1}{\sqrt{1+d^2}}\right).$$

This formula clearly shows that the inability to compute accurately small angles is integrated in the standard algorithm and cannot be fixed without changing the algorithm itself. The cosine, that is, a canonical correlation, is computed accurately and simply equals to one for all positive $d \leq 10^{-8}$. However, one cannot determine small angles from a cosine accurately in the presence of round-off errors. In statistical terms, it illustrates the problem we already mentioned above that the canonical correlation itself does not show any improvement in correlation when d is smaller than 10^{-8} in double precision.

In the next section, we consider a formula [3] that directly computes the sine of principal angles as in (2.1).

3. Properties of principal angles and a sine-based algorithm. We first review known sine-based formulas for the largest principal angle. Results of [1, 18] concerning the aperture of two linear manifolds give

(3.1)
$$\|P_F - P_G\| = \max\left\{\max_{x\in\mathcal{G},\|x\|=1} \|(I-P_F)x\|,\ \max_{y\in\mathcal{F},\|y\|=1} \|(I-P_G)y\|\right\}.$$

Let columns of matrices $Q_F \in \mathbf{R^{n\times p}}$ and $Q_G \in \mathbf{R^{n\times q}}$ form orthonormal bases for the subspaces $\mathcal{F}$ and $\mathcal{G}$, respectively. Then orthogonal projectors on $\mathcal{F}$ and $\mathcal{G}$ are $P_F = Q_F Q_F^T$ and $P_G = Q_G Q_G^T$, respectively, and

$$\|P_F - P_G\| = \max\{\|(I-Q_F Q_F^T)Q_G\|, \|(I-Q_G Q_G^T)Q_F\|\}. \tag{3.2}$$

If $p \neq q$, then expression of (3.2) is always equal to one; e.g., if $p > q$, then the second term under the maximum is one. If $p = q$, then both terms are the same and yield $\sin(\theta_q)$ by (1.5). Therefore, under our assumption $p \geq q$, only the first term is interesting to analyze. We note that the first term is the largest singular value of $(I - Q_F Q_F^T)Q_G$. What is the meaning of other singular values of the matrix?

This provides an insight into how to find a sine-based formulation to obtain the principal angles, which is embodied in the following theorem [3].

THEOREM 3.1. *Singular values $\mu_1 \leq \mu_2 \leq \cdots \leq \mu_q$ of the matrix $(I - Q_F Q_F^T)Q_G$ are given by $\mu_k = \sqrt{1-\sigma_k^2}$, $k = 1, \ldots, q$, where σ_k are defined in* (1.1). *Moreover, the principal angles satisfy the equalities $\theta_k = \arcsin(\mu_k)$.*

The right principal vectors can be computed as

$$v_k = Q_G z_k, \qquad k = 1, \ldots, q,$$

where z_k are corresponding orthonormal right singular vectors of matrix $(I - Q_F Q_F^T)Q_G$. The left principal vectors are then computed by

$$u_k = Q_F Q_F^T v_k/\sigma_k \quad \text{if } \sigma_k \neq 0, \quad k = 1, \ldots, q.$$

Proof. Our proof is essentially the same as that of [3]. We reproduce it here for completeness as we use a similar proof later for a general scalar product.

Let $B = (I - P_F)Q_G = (I - Q_F Q_F^T)Q_G$. Using the fact that $I - P_F$ is a projector and that $Q_G^T Q_G = I$, we have

$$\begin{aligned} B^T B &= Q_G^T (I - P_F)(I - P_F) Q_G = Q_G^T (I - P_F) Q_G \\ &= I - Q_G^T Q_F Q_F^T Q_G. \end{aligned}$$

Utilizing the SVD (1.1), we obtain $Q_F^T Q_G = Y \Sigma Z^T$, where $\Sigma = \mathrm{diag}(\sigma_1, \sigma_2, \ldots, \sigma_q)$; then

$$Z^T B^T B Z = I - \Sigma^2 = \mathrm{diag}(1 - \sigma_1^2, 1 - \sigma_2^2, \ldots, 1 - \sigma_q^2).$$

Thus, the singular values of B are given by $\mu_k = \sqrt{1 - \sigma_k^2}$, $k = 1, \ldots, q$, and the formula for the principal angles $\theta_k = \arcsin(\mu_k)$ follows directly from (1.2). □

We can now use the theorem to formulate an algorithm for computing all the principal angles. This approach meets our goal of a sine-based formulation, which should provide accurate computation of small angles. However, for large angles we keep the cosine-based algorithm.

ALGORITHM 3.1: **Modified SUBSPACE.m.**

Input: *real matrices F and G with the same number of rows.*

1. *Compute orthonormal bases $Q_F = \mathrm{orth}(F)$, $Q_G = \mathrm{orth}(G)$ of column-spaces of F and G.*
2. *Compute SVD for cosine: $[Y, \Sigma, Z] = \mathrm{svd}(Q_F^T Q_G)$, $\Sigma = \mathrm{diag}(\sigma_1, \ldots, \sigma_q)$.*
3. *Compute matrices of left $U_{\cos} = Q_F Y$ and right $V_{\cos} = Q_G Z$ principal vectors.*
4. *Compute matrix* $B = \begin{cases} Q_G - Q_F(Q_F^T Q_G) & \text{if } \mathrm{rank}(Q_F) \geq \mathrm{rank}(Q_G); \\ Q_F - Q_G(Q_G^T Q_F) & \text{otherwise.} \end{cases}$
5. *Compute SVD for sine: $[Y, \mathrm{diag}(\mu_1, \ldots, \mu_q), Z] = \mathrm{svd}(B)$.*
6. *Compute matrices $U_{\sin}$ and $V_{\sin}$ of left and right principal vectors:*
   $V_{\sin} = Q_G Z$, $U_{\sin} = Q_F(Q_F^T V_{\sin})\Sigma^{-1}$ *if* $\mathrm{rank}(Q_F) \geq \mathrm{rank}(Q_G)$;
   $U_{\sin} = Q_F Z$, $V_{\sin} = Q_G(Q_G^T U_{\sin})\Sigma^{-1}$ *otherwise.*
7. *Compute the principal angles, for $k = 1, \ldots, q$:*
   $\theta_k = \begin{cases} \arccos(\sigma_k) & \text{if } \sigma_k^2 < 1/2; \\ \arcsin(\mu_k) & \text{if } \mu_k^2 \leq 1/2. \end{cases}$
8. *Form matrices U and V by picking up corresponding columns of $U_{\sin}$, $V_{\sin}$ and $U_{\cos}$, $V_{\cos}$, according to the choice for θ_k above.*

Output: *Principal angles $\theta_1, \ldots, \theta_q$ between column-spaces of matrices F and G, and corresponding matrices U and V of left and right principal vectors, respectively.*

REMARK 3.1. *In step* 1 *of the algorithm, the orthogonalization can be performed using the QR method or the SVD. In our actual code, an SVD-based built-in MATLAB function ORTH.m is used for the orthogonalization.*

It is pointed out in [10] *that errors in computing Q_F and Q_G, especially expected for ill-conditioned F and G, may lead to an irreparable damage in final answers. A proper column scaling of F and G could in some cases significantly reduce condition numbers of F and G. We highlight that an explicit columnwise normalization of matrices F and G is not required prior to orthogonalization if a particular orthogonalization algorithm used here is invariant under column scaling in finite precision arithmetic. Our numerical tests show that the explicit column scaling is not needed if we utilize a built-in MATLAB function QR.m for orthonormalization. However, the explicit column scaling apparently helps to improve the accuracy when the SVD-based*

built-in MATLAB function ORTH.m is used for orthonormalization. In [10]*, QR factorizations with complete pivoting are recommended for computing* Q_F *and* Q_G*.*

REMARK 3.2. *A check* $\mathrm{rank}(Q_F) \geq \mathrm{rank}(Q_G)$ *in steps* 4 *and* 6 *of the algorithm removes the need for our assumption* $p = \mathrm{rank}(Q_F) \geq \mathrm{rank}(Q_G) = q$*.*

REMARK 3.3. *We replace here in step* 4

$$(I - Q_F Q_F^T)Q_G = Q_G - Q_F(Q_F^T Q_G), \quad (I - Q_G Q_G^T)Q_F = Q_F - Q_G(Q_G^T Q_F)$$

to avoid storing in memory any n*-by-*n *matrices in the algorithm, which allows us to compute principal angles efficiently for large* $n \gg p$ *as well.*

If the matrix $Q_{F^\perp}$*, with orthonormal columns that span the subspace* $\mathcal{F}^\perp$*, the orthogonal complement of* $\mathcal{F}$*, was available to us when* $p \geq q$*, we could take here*

$$B = Q_{F^\perp} Q_G,$$

as in the CS decomposition methods; see [25, 28, 26, 23]*. Under our assumption* $n \gg p$*, however, the matrix* $Q_{F^\perp}$ *is essentially of the size* n *and thus shall be avoided.*

The 1/2 threshold used in Algorithm 3.1 in steps 7 and 8 to separate small and large principal angles and corresponding vectors seems to be a natural choice. However, such an artificial fixed threshold may cause troubles with orthogonality in the resulting choice of vectors in step 8 if there are several angles close to each other but on different sides of the threshold. The problem is that the corresponding principal vectors, picked up from two orthogonal sets computed by different algorithms, may not be orthogonal. A more accurate approach would be to identify such possible cluster of principal angles around the original threshold and to make sure that all principal vectors corresponding to the cluster are chosen according to either step 3, or step 6, but not both.

ALGORITHM 3.2: **Modified and Improved SUBSPACE.m.**
Input: *real matrices* F *and* G *with the same number of rows.*

1. *Compute orthonormal bases* $Q_F = \mathrm{orth}(F)$, $Q_G = \mathrm{orth}(G)$ *of column-spaces of* F *and* G*.*
2. *Compute SVD for cosine:* $[Y, \Sigma, Z] = \mathrm{svd}(Q_F^T Q_G)$, $\Sigma = \mathrm{diag}(\sigma_1, \ldots, \sigma_q)$*.*
3. *Compute matrices of left* $U_{\cos} = Q_F Y$ *and right* $V_{\cos} = Q_G Z$ *principal vectors.*
4. *Compute large principal angles, for* $k = 1, \ldots, q$*:*
   $\theta_k = \arccos(\sigma_k)$ *if* $\sigma_k^2 < 1/2$*.*
5. *Form parts of matrices* U *and* V *by picking up corresponding columns of* $U_{\cos}$*,* $V_{\cos}$*, according to the choice for* θ_k *above. Put columns of* $U_{\cos}$*,* $V_{\cos}$*, which are left, in matrices* R_F *and* R_G*. Collect the corresponding* σ*'s in a diagonal matrix* Σ_R*.*
6. *Compute the matrix* $B = R_G - Q_F(Q_F^T R_G)$*.*
7. *Compute SVD for sine:* $[Y, \mathrm{diag}(\mu_1, \ldots, \mu_q), Z] = \mathrm{svd}(B)$*.*
8. *Compute matrices* $U_{\sin}$ *and* $V_{\sin}$ *of left and right principal vectors:*
   $V_{\sin} = R_G Z$*,* $U_{\sin} = R_F(R_F^T V_{\sin})\Sigma_R^{-1}$*.*
9. *Recompute the small principal angles, for* $k = 1, \ldots, q$*:*
   $\theta_k = \arcsin(\mu_k)$ *if* $\mu_k^2 \leq 1/2$*.*
10. *Complete matrices* U *and* V *by adding columns of* $U_{\sin}$*,* $V_{\sin}$*.*

Output: *Principal angles* $\theta_1, \ldots, \theta_q$ *between column-spaces of matrices* F *and* G*, and corresponding matrices* U *and* V *of left and right principal vectors, respectively.*

Let us repeat that, in exact arithmetic, the sine and cosine based approaches give the same results; e.g., columns of $U_{\sin}$ and $V_{\sin}$ must be the same as those of $U_{\cos}$

and $V_{\cos}$. Why do we need to recompute essentially the same vectors a second time? What if we compute only $U_{\cos}$, $V_{\cos}$ and then recompute just small principal angles using, e.g., the obvious formula

$$\mu_k = \|u_k - \sigma_k v_k\|? \tag{3.3}$$

An anonymous referee recommended this approach and suggested that it would resolve the inaccuracy in the cosine-based algorithm illustrated in the previous section, without the need for the second SVD.

The answer is that the cosine-based algorithm fails to compute accurately not only the small principal angles but also the corresponding principal vectors. The reason for this is that singular values computed in step 2 of Algorithm 3.1 are the cosines of principal angles, while singular values of the matrix B in step 5 are the sines of principal angles. Thus, the distribution of singular values is different in steps 2 and 5; e.g., singular values corresponding to small angles are much better separated in step 5 than in step 2. For example, angles 10^{-10} and 10^{-12} will produce a multiple singular value 1 in step 2 in double precision but will produce two distinct small singular values in step 5. This means that singular vectors, corresponding to small principal angles, might not be computed accurately in computer arithmetic using only SVD in step 2, which will also lead to inaccurate computation of the small principal angles by formula (3.3). Our numerical tests support this conclusion.

There is some obvious redundancy in Algorithm 3.1. Indeed, we do not need to calculate columns of $U_{\sin}$ and $V_{\sin}$, corresponding to large sines, and columns of $U_{\cos}$ and $V_{\cos}$, corresponding to large cosines, as they are computed inaccurately in computer arithmetic and we just discard them later in the algorithm. However, first, for large-scale applications with $n \gg p$ that we are interested in, the redundancy is insignificant. Second, this allows us to perform steps 2–3 and steps 4–5 of Algorithm 3.1 independently in parallel. We note that Σ must be invertible in Algorithm 3.1.

For sequential computations, we now describe Algorithm 3.2. Here, we reduce computational costs of the second SVD by using already computed vectors $U_{\cos}$ and $V_{\cos}$ for the cosines. The cosine-based algorithm computes inaccurately individual principal vectors corresponding to small principal angles. However, it may find accurately the corresponding invariant subspaces spanned by all these vectors. Thus, the idea is that, using $U_{\cos}$ and $V_{\cos}$, we can identify invariant subspaces in $\mathcal{F}$ and $\mathcal{G}$, which correspond to all small principal angles. Then, we perform the second SVD only for these subspaces, computing only columns of $U_{\sin}$ and $V_{\sin}$ that we actually need, which may significantly reduce the size of the matrix in the second SVD. This idea is used in the CS decomposition algorithm of [28].

We keep steps 1–3 of Algorithm 3.1 unchanged but modify accordingly later steps to obtain Algorithm 3.2. Such changes may significantly reduce the size of matrix B and, thus, the costs, if large principal angles outnumber small ones; e.g., if there are no small principal angles at all, the algorithm simply stops at step 3. We note that matrix Σ_R is always invertible, unlike matrix Σ.

REMARK 3.4. *By construction, matrices R_F and R_G have the same number of already orthogonal columns, which removes the need for orthogonalization and for comparing, in step* 6*, their ranks.*

REMARK 3.5. *We have three, equivalent in exact arithmetic, possibilities to compute matrix B:*

$$B = (I - R_F R_F^T) R_G = R_G - R_F (R_F^T R_G) = R_G - Q_F (Q_F^T R_G).$$

The first formula is ruled out to avoid storing in memory any n-by-n matrices in the algorithm. Our numerical tests show that the third expression, though somewhat more expensive than the second one, often provides more accurate results in the presence of round-off errors.

To summarize, Algorithms 3.1 and 3.2 use the cosine-based formulation (1.1), (1.2) for large angles and the sine-based formulation of Theorem 3.1 for small angles, which allows accurate computation of all angles. The algorithms are reasonably efficient for large-scale applications with $n \gg p$ and are more robust than the original cosine-based only version.

In the rest of the section, we describe some useful properties of principal angles not yet mentioned. In the present paper, we follow [19] and make use of an orthogonal projectors technique. For an alternative approach, popular in matrix theory, which is based on representation of subspaces in a canonical CS form, we refer to [26].

Theorem 3.1 characterizes singular values of the product $(I - P_F)Q_G$, which are the sine of the principal angles. What are singular values of the matrix $P_F Q_G$? A trivial modification of the previous proof leads to the following not really surprising result that these are the cosine of the principal angles.

THEOREM 3.2. *Singular values of the matrix $Q_F Q_F^T Q_G$ are exactly the same as σ_k, defined in* (1.1).

We conclude this section with other simple and known (e.g., [29, 26, 30]) sine and cosine representations of principal angles and principal vectors, this time using orthogonal projectors P_F and P_G on subspaces $\mathcal{F}$ and $\mathcal{G}$, respectively.

THEOREM 3.3. *Let assumptions of Theorem* 3.1 *be satisfied. Then $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_q$ are the q largest singular values of the matrix $P_F P_G$; in particular,*

$$\sigma_1 = \|P_F P_G\|.$$

Other $n - q$ singular values are all equal to zero.

REMARK 3.6. *As singular values of $P_F P_G$ are the same as those of $P_G P_F$, subspaces $\mathcal{F}$ and $\mathcal{G}$ play symmetric roles in Theorem* 3.3*; thus, our assumption that $p = \dim\mathcal{F} \geq \dim\mathcal{G} = q$ is irrelevant here.*

THEOREM 3.4. *Let assumptions of Theorem* 3.1 *be satisfied. Then $\mu_1 \leq \mu_2 \leq \cdots \leq \mu_q$ are the q largest singular values of the matrix $(I - P_F)P_G$; in particular,*

$$\mu_q = \|(I - P_F)P_G\|.$$

Other $n - q$ singular values are all equal to zero.

REMARK 3.7. *Comparing Theorems* 3.3 *and* 3.4 *shows trivially that sine of principal angles between $\mathcal{F}$ and $\mathcal{G}$ are the same as cosine of principal angles between $\mathcal{F}^\perp$ and $\mathcal{G}$ because $I - P_F$ is an orthogonal projector on $\mathcal{F}^\perp$. If $p > n/2 > q$, it may be cheaper to compute principal angles between $\mathcal{F}^\perp$ and $\mathcal{G}$ instead of principal angles between $\mathcal{F}$ and $\mathcal{G}$.*

What can we say about singular values of the matrix $(I - P_G)P_F$? In other words, how do cosine of principal angles between subspaces $\mathcal{F}^\perp$ and $\mathcal{G}$ compare to cosine of principal angles between their orthogonal complements $\mathcal{F}$ and $\mathcal{G}^\perp$? If $p = q$, they are absolutely the same; in particular, the minimal angle between subspaces $\mathcal{F}^\perp$ and $\mathcal{G}$ is in this case the same as the minimal angle between their orthogonal complements $\mathcal{F}$ and $\mathcal{G}^\perp$, e.g., in [9], and, in fact, is equal to $\text{gap}(\mathcal{F}, \mathcal{G}) = \|P_F - P_G\|$ as we already discussed. When $p > q$, subspaces $\mathcal{F}$ and $\mathcal{G}^\perp$ must have a nontrivial intersection because the sum of their dimensions is too big; thus, the minimal angle between subspaces $\mathcal{F}$ and $\mathcal{G}^\perp$ must be zero in this case, which corresponds to $\|(I - P_G)P_F\| = 1$,

while $\|(I - P_F)P_G\|$ may be less than one. To be more specific, $\dim(\mathcal{F} \cap \mathcal{G}^\perp) \geq p - q$; thus, at least $p - q$ singular values of the matrix $(I - P_G)P_F$ are equal to one. Then, we have the following statement, which completely clarifies the issue of principal angles between orthogonal complements; cf. Ex. 1.2.6 of [4].

THEOREM 3.5. *The set of singular values of $(I - P_G)P_F$, when $p > q$, consists of $p - q$ ones, q singular values of $(I - P_F)P_G$, and $n - p$ zeros.*

In particular, this shows that the smallest positive sine of principal angles between $\mathcal{F}$ and $\mathcal{G}$, called the *minimum gap*, is the same as that between $\mathcal{F}^\perp$ and $\mathcal{G}^\perp$; see [18]. This theorem can also be used to reduce the costs of computing the principal angles between subspaces $\mathcal{F}$ and $\mathcal{G}$, when their dimensions p and q are greater than $n/2$, by replacing $\mathcal{F}$ and $\mathcal{G}$ with their orthogonal complements.

Let us finally mention a simple property of principal vectors, emphasized in [29], which helps us to understand a geometric meaning of pairs of corresponding principal vectors from different subspaces.

THEOREM 3.6. *We have*

$$P_F v_k = \sigma_k u_k, \quad P_G u_k = \sigma_k v_k, \quad k = 1, \ldots, q,$$

and

$$u_i^T v_j = (P_F u_i)^T v_j = u_i^T P_F v_j = \sigma_j u_i^T u_j = \sigma_j \delta_{ij}, \qquad i, j = 1, \ldots, q.$$

In other words, a chosen pair u_k, v_k spans a subspace, invariant with respect to orthoprojectors P_F and P_G and orthogonal to all other such subspaces. The kth principal angle θ_k is simply the angle between u_k and v_k; see (3.3).

Moreover, the subspace $\operatorname{span}\{u_k, v_k\}$ *is also invariant with respect to orthoprojectors $I - P_F$ and $I - P_G$. Let us define two other unit vectors in this subspace:*

$$u_k^\perp = (v_k - \sigma_k u_k)/\mu_k \in \mathcal{F}^\perp, \quad v_k^\perp = (u_k - \sigma_k v_k)/\mu_k \in \mathcal{G}^\perp$$

such that $u_k^T u_k^\perp = v_k^T v_k^\perp = 0$. Then

- *u_k, v_k are principal vectors for subspaces $\mathcal{F}$ and $\mathcal{G}$;*
- *$u_k^\perp, v_k$ are principal vectors for subspaces $\mathcal{F}^\perp$ and $\mathcal{G}$;*
- *$u_k, v_k^\perp$ are principal vectors for subspaces $\mathcal{F}$ and $\mathcal{G}^\perp$;*
- *$u_k^\perp, -v_k^\perp$ are principal vectors for subspaces $\mathcal{F}^\perp$ and $\mathcal{G}^\perp$,*

which concludes the description of all cases.

In the next section, we deal with an arbitrary scalar product.

4. Generalization to an A-based scalar product. Let $A \in \mathbf{R}^{\mathbf{n}\times\mathbf{n}}$ be a fixed symmetric positive definite matrix. Let $(x, y)_A = (x, Ay) = y^T A x$ be an A-based scalar product, $x, y \in \mathbf{R}^{\mathbf{n}}$. Let $\|x\|_A = \sqrt{(x, x)_A}$ be the corresponding vector norm and let $\|B\|_A$ be the corresponding induced matrix norm of a matrix $B \in \mathbf{R}^{\mathbf{n}\times\mathbf{n}}$. We note that $\|x\|_A = \|A^{1/2}x\|$ and $\|B\|_A = \|A^{1/2}BA^{-1/2}\|$.

In order to define principal angles based on this scalar product, we will follow arguments of [3, 13] but in an A-based scalar product instead of the standard Euclidean scalar product. Again, we will assume for simplicity of notation that $p \geq q$.

Principal angles

$$\theta_1, \ldots, \theta_q \in [0, \pi/2]$$

between subspaces $\mathcal{F}$ and $\mathcal{G}$ in the A-based scalar product $(\cdot, \cdot)_A$ are defined recursively for $k = 1, \ldots, q$ by analogy with the previous definition for $A = I$ as

$$\cos(\theta_k) = \max_{u \in \mathcal{F}} \max_{v \in \mathcal{G}} (u, v)_A = (u_k, v_k)_A \tag{4.1}$$

subject to

$$\|u\|_A = \|v\|_A = 1, \quad (u, u_i)_A = 0, \quad (v, v_i)_A = 0, \quad i = 1, \dots, k-1. \tag{4.2}$$

The vectors $u_1, \dots, u_q$ and $v_1, \dots, v_q$ are called principal vectors relative to the A-based scalar product.

The following theorem justifies the consistency of the definition above and provides a cosine-based algorithm for computing the principal angles in the A-based scalar product. It is a direct generalization of the cosine-based approach of [3, 13].

THEOREM 4.1. *Let columns of $Q_F \in \mathbf{R}^{\mathbf{n}\times\mathbf{p}}$ and $Q_G \in \mathbf{R}^{\mathbf{n}\times\mathbf{q}}$ now be A-orthonormal bases for the subspaces F and G, respectively. Let $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_q$ be singular values of $Q_F^T A Q_G$ with corresponding left and right singular vectors y_k and z_k, $k = 1, \dots, q$. Then the principal angles relative to the scalar product $(\cdot, \cdot)_A$ as defined in* (4.1) *and* (4.2) *are computed as*

$$\theta_k = \arccos(\sigma_k), \qquad k = 1, \dots, q, \tag{4.3}$$

where

$$0 \leq \theta_1 \leq \cdots \leq \theta_q \leq \frac{\pi}{2},$$

while the principal vectors are given by

$$u_k = Q_F y_k, \quad v_k = Q_G z_k, \quad k = 1, \dots, q.$$

Proof. We first rewrite definition (4.1) and (4.2) of principal angles in the following equivalent form. For $k = 1, \dots, q$,

$$\cos(\theta_k) = \max_{y \in \mathbf{R}^{\mathbf{p}}} \max_{z \in \mathbf{R}^{\mathbf{q}}} y^T Q_F^T A Q_G z \;=\; y_k^T Q_F^T A Q_G z_k$$

subject to

$$\|y\| = \|z\| = 1, \quad y^T y_i = 0, \quad z^T z_i = 0, \quad i = 1, \dots, k-1,$$

where $u = Q_F y \in \mathcal{F}$, $v = Q_G z \in \mathcal{G}$ and $u_k = Q_F y_k \in \mathcal{F}$, $v_k = Q_G z_k \in \mathcal{G}$.

Since Q_F and Q_G have A-orthonormal columns, $Q_F^T A Q_F = I$ and $Q_G^T A Q_G = I$. This implies

$$\|u\|_A^2 = y^T Q_F^T A Q_F y = y^T y = \|y\|^2 = 1$$

and

$$\|v\|_A^2 = z^T Q_G^T A Q_G z = z^T z = \|z\|^2 = 1.$$

For $i \neq j$, we derive

$$(u_i, u_j)_A = y_i^T Q_F^T A Q_F y_j = y_i^T y_j = 0$$

and

$$(v_i, v_j)_A = z_i^T Q_G^T A Q_G z_j = z_i^T z_j = 0.$$

Now, let the reduced SVD of $Q_F^T A Q_G$ be

$$Y^T Q_F^T A Q_G Z \;=\; \mathrm{diag}(\sigma_1, \sigma_2, \dots, \sigma_q), \tag{4.4}$$

where $Y \in \mathbf{R}^{\mathbf{p}\times\mathbf{q}}$, $Z \in \mathbf{R}^{\mathbf{q}\times\mathbf{q}}$ both have orthonormal columns.

Then, by Theorem 1.1 with $M = Q_F^T A Q_G$, the equality $\cos(\theta_k) = \sigma_k$, $k = 1, \ldots, q$, just provides two equivalent representations of the singular values of $Q_F^T A Q_G$, and y_z and z_k can be chosen as columns of matrices Y and Z, respectively. The statement of the theorem follows. ∎

Let us now make a trivial but important observation that links principal angles in the A-based scalar product with principal angles in the original standard scalar product, when a factorization of $A = K^T K$, e.g., $K = A^{1/2}$, is available. We formulate it as the following theorem.

THEOREM 4.2. *Let $A = K^T K$. Under assumptions of Theorem* 4.1 *the principal angles between subspaces $\mathcal{F}$ and $\mathcal{G}$ relative to the scalar product $(\cdot,\cdot)_A$ coincide with the principal angles between subspaces $K\mathcal{F}$ and $K\mathcal{G}$ relative to the original scalar product $(\cdot,\cdot)$.*

Proof. One way to prove this is to notice that our definition of the principal angles between subspaces $\mathcal{F}$ and $\mathcal{G}$ relative to the scalar product $(\cdot,\cdot)_A$ turns into a definition of the principal angles between subspaces $K\mathcal{F}$ and $K\mathcal{G}$ relative to the original scalar product $(\cdot,\cdot)$ if we make substitutions $Ku \mapsto u$ and $Kv \mapsto v$.

Another proof is to use the representation

$$Q_F^T A Q_G = \left(K Q_F\right)^T K Q_G,$$

where columns of matrices KQ_F and KQ_G are orthonormal with respect to the original scalar product $(\cdot,\cdot)$ and span subspaces $K\mathcal{F}$ and $K\mathcal{G}$, respectively. Now Theorem 4.1 is equivalent to the traditional SVD theorem on cosine of principal angles between subspaces $K\mathcal{F}$ and $K\mathcal{G}$ relative to the original scalar product $(\cdot,\cdot)$, formulated in the introduction. ∎

The A-orthogonal projectors on subspaces $\mathcal{F}$ and $\mathcal{G}$ are now defined by formulas

$$P_F = Q_F Q_F^{*_A} = Q_F Q_F^T A \text{ and } P_G = Q_G Q_G^{*_A} = Q_G Q_G^T A,$$

where $*_A$ denotes the A-adjoint.

To obtain a sine-based formulation in the A-based scalar product that is accurate for small angles, we first adjust (1.5) and (3.1) to the new A-based scalar product:

(4.5)
$$\begin{aligned}\operatorname{gap}_A(\mathcal{F},\mathcal{G}) &= \|P_F - P_G\|_A \\ &= \max\left\{\max_{x\in\mathcal{G},\|x\|_A=1} \|(I-P_F)x\|_A,\ \max_{y\in\mathcal{F},\|y\|_A=1} \|(I-P_G)y\|_A\right\}.\end{aligned}$$

If $p = q$, this equation will yield $\sin(\theta_q)$, consistent with Theorem 4.1. Similar to the previous case $A = I$, only the first term under the maximum is of interest under our assumption that $p \geq q$. Using the fact that

$$\|x\|_A = \|Q_G z\|_A = \|z\| \ \ \forall x \in \mathcal{G}, \qquad x = Q_G z,\ z \in \mathbf{R}^{\mathbf{q}},$$

the term of interest can be rewritten as

$$\max_{x\in\mathcal{G},\|x\|_A=1} \|(I-P_F)x\|_A = \|A^{1/2}(I - Q_F Q_F^T A)Q_G\|. \tag{4.6}$$

Here we use the standard induced Euclidean norm $\|\cdot\|$ for computational purposes. Similar to our arguments in the previous section, we obtain a more general formula for all principal angles in the following.

THEOREM 4.3. *Let $S = (I - Q_F Q_F^T A)Q_G$. Singular values $\mu_1 \le \mu_2 \le \cdots \le \mu_q$ of matrix $A^{1/2}S$ are given by $\mu_k = \sqrt{1-\sigma_k^2}$, $k = 1, \ldots, q$, where σ_k are defined in* (4.4). *Moreover, the principal angles satisfy the equalities $\theta_k = \arcsin(\mu_k)$. The right principal vectors can be computed as*

$$v_k = Q_G z_k, \qquad k = 1, \ldots, q,$$

where z_k are corresponding orthonormal right singular vectors of matrix $A^{1/2}S$. The left principal vectors are then computed by

$$u_k = Q_F Q_F^T A v_k / \sigma_k \quad \text{if } \sigma_k \neq 0, \quad k = 1, \ldots, q.$$

Proof. We first notice that squares of the singular values μ_k of the matrix $A^{1/2}S$, which appear in (4.6), coincide with eigenvalues $\nu_k = \mu_k^2$ of the product $S^T AS$. Using the fact that $Q_F^T A Q_F = I$ and $Q_G^T A Q_G = I$, we have

$$\begin{aligned} S^T AS &= Q_G^T (I - AQ_F Q_F^T) A (I - Q_F Q_F^T A) Q_G \\ &= I - Q_G^T A Q_F Q_F^T A Q_G. \end{aligned}$$

Utilizing the SVD (4.4), we obtain $Q_F^T A Q_G = Y \Sigma Z^T$, where $\Sigma = \operatorname{diag}(\sigma_1, \sigma_2, \ldots, \sigma_q)$; then

$$Z^T S^T AS Z = I - \Sigma^2 = \operatorname{diag}(1-\sigma_1^2, 1-\sigma_2^2, \ldots, 1-\sigma_q^2).$$

Thus, the eigenvalues of $S^T AS$ are given by $\nu_k = 1 - \sigma_k^2$, $k = 1, \ldots, q$, and the formula for the principal angles follows directly from (4.3). ∎

For computational reasons, when n is large, we need to avoid dealing with the square root $A^{1/2}$ explicitly. Also, A may not be available as a matrix but only as a function performing the multiplication of A by a given vector. Fortunately, the previous theorem can be trivially reformulated as follows to resolve this issue.

THEOREM 4.4. *Eigenvalues $\nu_1 \le \nu_2 \le \cdots \le \nu_q$ of matrix $S^T AS$, where $S = (I - Q_F Q_F^T A)Q_G$, are equal to $\nu_k = 1 - \sigma_k^2$, $k = 1, \ldots, q$, where σ_k are defined in* (4.4). *Moreover, the principal angles satisfy the equalities $\theta_k = \arcsin\left(\sqrt{\nu_k}\right)$, $k = 1, \ldots, q$. The right principal vectors can be computed as*

$$v_k = Q_G z_k, \qquad k = 1, \ldots, q,$$

where z_k are corresponding orthonormal right eigenvectors of matrix $S^T AS$. The left principal vectors are then computed by

$$u_k = Q_F Q_F^T A v_k / \sigma_k \quad \text{if } \sigma_k \neq 0, \quad k = 1, \ldots, q.$$

We can easily modify the previous proof to obtain the following analogue of Theorem 3.2.

THEOREM 4.5. *Singular values of the matrix $A^{1/2} Q_F Q_F^T A Q_G = A^{1/2} P_F Q_G$ coincide with σ_k, defined in* (4.4).

It is also useful to represent principal angles using exclusively A-orthogonal projectors P_F and P_G on subspaces $\mathcal{F}$ and $\mathcal{G}$, respectively, similarly to Theorems 3.3 and 3.4.

THEOREM 4.6. *Under assumptions of Theorem* 4.1, *$\sigma_1 \ge \sigma_2 \ge \cdots \ge \sigma_q$ are the q largest singular values of the matrix $A^{1/2} P_F P_G A^{-1/2}$; in particular,*

$$\sigma_1 = \|P_F P_G\|_A.$$

Other $n-q$ singular values are all equal to zero.

Proof. First, we rewrite

$$\begin{aligned} A^{1/2}P_FP_GA^{-1/2} &= A^{1/2}Q_FQ_F^TAQ_GQ_G^TAA^{-1/2} \\ &= A^{1/2}Q_F\left(A^{1/2}Q_F\right)^TA^{1/2}Q_G\left(A^{1/2}Q_G\right)^T. \end{aligned}$$

As columns of matrices $A^{1/2}Q_F$ and $A^{1/2}Q_G$ are orthonormal with respect to the original scalar product $(\cdot,\cdot)$ bases of subspaces $A^{1/2}\mathcal{F}$ and $A^{1/2}\mathcal{G}$, respectively, the last product is equal to the product of orthogonal (not A-orthogonal!) projectors $P_{A^{1/2}\mathcal{F}}$ and $P_{A^{1/2}\mathcal{G}}$ on subspaces $A^{1/2}\mathcal{F}$ and $A^{1/2}\mathcal{G}$.

Second, we can now use Theorem 3.3 to state that cosine of principal angles between subspaces $A^{1/2}\mathcal{F}$ and $A^{1/2}\mathcal{G}$ with respect to the original scalar product $(\cdot,\cdot)$ are given by the q largest singular values of the product $P_{A^{1/2}\mathcal{F}}P_{A^{1/2}\mathcal{G}} = A^{1/2}P_FP_GA^{-1/2}$.

Finally, we use Theorem 4.2 to conclude that these singular values are, in fact, $\sigma_k,\ k=1,\dots,q$, i.e., the cosine of principal angles between subspaces $\mathcal{F}$ and $\mathcal{G}$ with respect to the A-based scalar product. ∎

THEOREM 4.7. *Let assumptions of Theorem* 4.4 *be satisfied. Then $\mu_1 \le \mu_2 \le \cdots \le \mu_q$ are the q largest singular values of the matrix $A^{1/2}(I-P_F)P_GA^{-1/2}$; in particular,*

$$\mu_q = \|(I-P_F)P_G\|_A.$$

The other $n-q$ singular values are all equal to zero.

Proof. We rewrite

$$\begin{aligned} A^{1/2}(I-P_F)P_GA^{-1/2} &= \left(I - A^{1/2}Q_F\left(A^{1/2}Q_F\right)^T\right)A^{1/2}Q_G\left(A^{1/2}Q_G\right)^T \\ &= \left(I-P_{A^{1/2}\mathcal{F}}\right)P_{A^{1/2}\mathcal{G}} \end{aligned}$$

and then follow arguments similar to those of the previous proof, but now using Theorem 3.4 instead of Theorem 3.3. ∎

Remarks 3.6–3.7 and Theorems 3.5–3.6 for the case $A = I$ hold in the general case, too, with obvious modifications.

Our final theoretical results are perturbation theorems in the next section.

5. Perturbation of principal angles in the A-based scalar product. In the present section, for simplicity, we *always assume* that matrices F, G and their perturbations $\tilde{F}$, $\tilde{G}$ have the same rank; thus, in particular, $p = q$.

We notice that F and G appear symmetrically in the definition of the principal angles, under our assumption that they and their perturbations have the same rank. This means that we do not have to analyze the perturbation of F and G together at the same time. Instead, we first study only a perturbation in G.

Before we start with an estimate for cosine, let us introduce a new notation $\ominus$ using an example:

$$(\mathcal{G}+\tilde{\mathcal{G}})\ominus\mathcal{G} = (\mathcal{G}+\tilde{\mathcal{G}})\cap\mathcal{G}^\perp,$$

where $\ominus$ and the orthogonal complement to $\mathcal{G}$ are understood in the A-based scalar product.

LEMMA 5.1. *Let $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_q$ and $\hat{\sigma}_1 \geq \hat{\sigma}_2 \geq \cdots \geq \hat{\sigma}_q$ be cosine of principal angles between subspaces $\mathcal{F}$, $\mathcal{G}$ and $\mathcal{F}$, $\tilde{\mathcal{G}}$, respectively, computed in the A-based scalar product. Then, for $k = 1, \ldots, q$,*

(5.1)
$$|\sigma_k - \hat{\sigma}_k| \leq \max\{\cos(\theta_{\min}\{(\mathcal{G}+\tilde{\mathcal{G}}) \ominus \mathcal{G}, \mathcal{F}\}); \cos(\theta_{\min}\{(\mathcal{G}+\tilde{\mathcal{G}}) \ominus \tilde{\mathcal{G}}, \mathcal{F}\})\}\mathrm{gap}_A(\mathcal{G}, \tilde{\mathcal{G}}),$$

where $\theta_{\min}$ is the smallest angle between corresponding subspaces, measured in the A-based scalar product.

Proof. The proof is based on the following identity:

(5.2)
$$A^{1/2}Q_F Q_F^T A Q_{\tilde{G}} = A^{1/2}Q_F Q_F^T A Q_G Q_G^T A Q_{\tilde{G}} + A^{1/2}Q_F Q_F^T A (I - Q_G Q_G^T A) Q_{\tilde{G}},$$

which is a multidimensional analogue of the trigonometric formula for the cosine of the sum of two angles. Now we use two classical theorems on perturbation of singular values with respect to addition:

$$s_k(T+S) \leq s_k(T) + \|S\|, \tag{5.3}$$

and with respect to multiplication:

$$s_k(TS^T) \leq s_k(T)\|S^T\|, \tag{5.4}$$

where T and S are matrices of corresponding sizes. We first need to take $T = A^{1/2}Q_F Q_F^T A Q_G Q_G^T A Q_{\tilde{G}}$ and $S = A^{1/2}Q_F Q_F^T A (I - Q_G Q_G^T A) Q_{\tilde{G}}$ in (5.3) to get

$$\begin{aligned}\hat{\sigma}_k = s_k(A^{1/2}Q_F Q_F^T A Q_{\tilde{G}}) \leq s_k(&A^{1/2}Q_F Q_F^T A Q_G Q_G^T A Q_{\tilde{G}}) \\ &+ \|A^{1/2}Q_F Q_F^T A (I - Q_G Q_G^T A) Q_{\tilde{G}}\|,\end{aligned}$$

where the first equality follows from Theorem 4.5. In the second term in the sum on the right, we need to estimate a product, similar to a product of *three* orthoprojectors. We notice that column vectors of $(I - Q_G Q_G^T A) Q_{\tilde{G}}$ belong to the subspace $(\mathcal{G}+\tilde{\mathcal{G}}) \ominus \mathcal{G}$. Let $P_{(\mathcal{G}+\tilde{\mathcal{G}})\ominus\mathcal{G}}$ be an A-orthogonal projector on the subspace. Then the second term can be rewritten, also using the projector $Q_F Q_F^T A = P_F$, as

$$A^{1/2}Q_F Q_F^T A (I - Q_G Q_G^T A) Q_{\tilde{G}} = A^{1/2}Q_F Q_F^T A P_{(\mathcal{G}+\tilde{\mathcal{G}})\ominus\mathcal{G}} (I - Q_G Q_G^T A) Q_{\tilde{G}}$$

$$= \left(A^{1/2} P_F P_{(\mathcal{G}+\tilde{\mathcal{G}})\ominus\mathcal{G}} A^{-1/2}\right) A^{1/2} (I - Q_G Q_G^T A) Q_{\tilde{G}};$$

therefore, it can be estimated as

$$\|A^{1/2}Q_F Q_F^T A (I - Q_G Q_G^T A) Q_{\tilde{G}}\| \leq \|A^{1/2} P_F P_{(\mathcal{G}+\tilde{\mathcal{G}})\ominus\mathcal{G}} A^{-1/2}\| \|A^{1/2}(I - Q_G Q_G^T A) Q_{\tilde{G}}\|.$$

The first multiplier in the last product equals

$$\|A^{1/2} P_F P_{(\mathcal{G}+\tilde{\mathcal{G}})\ominus\mathcal{G}} A^{-1/2}\| = \|P_F P_{(\mathcal{G}+\tilde{\mathcal{G}})\ominus\mathcal{G}}\|_A = \cos(\theta_{\min}\{(\mathcal{G}+\tilde{\mathcal{G}}) \ominus \mathcal{G}, \mathcal{F}\}),$$

similar to (4.6) and using Theorem 4.6 for subspaces $(\mathcal{G}+\tilde{\mathcal{G}}) \ominus \mathcal{G}$ and $\mathcal{F}$; while the second multiplier is $\mathrm{gap}_A(\mathcal{G}, \tilde{\mathcal{G}})$, because of our assumption $\dim\mathcal{F} = \dim\mathcal{G} = \dim\tilde{\mathcal{G}}$.

To estimate the first term in the sum, we apply (5.4) with $T = A^{1/2}Q_F Q_F^T A Q_G$ and $S^T = Q_G^T A Q_{\tilde{G}}$:

$$\begin{aligned} s_k(A^{1/2}Q_F Q_F^T A Q_G Q_G^T A Q_{\tilde{G}}) &\le s_k(A^{1/2}Q_F Q_F^T A Q_G)\|Q_G^T A Q_{\tilde{G}}\| \\ &\le s_k(A^{1/2}Q_F Q_F^T A Q_G) = \sigma_k, \end{aligned}$$

simply because the second multiplier here is the cosine of an angle between $\mathcal{G}$ and $\tilde{\mathcal{G}}$ in the A-based scalar product, which is, of course, bounded by one from above. Thus, we just proved

$$\hat{\sigma}_k \le \sigma_k + \cos(\theta_{\min}\{(\mathcal{G}+\tilde{\mathcal{G}}) \ominus \mathcal{G},\, \mathcal{F}\})\mathrm{gap}_A(\mathcal{G}, \tilde{\mathcal{G}}).$$

Changing places of $Q_{\tilde{G}}$ and Q_G, we obtain

$$\sigma_k \le \hat{\sigma}_k + \cos(\theta_{\min}\{(\mathcal{G}+\tilde{\mathcal{G}}) \ominus \tilde{\mathcal{G}},\, \mathcal{F}\})\mathrm{gap}_A(\mathcal{G}, \tilde{\mathcal{G}})$$

and come to the statement of the lemma. ∎

REMARK 5.1. *Let us try to clarify the meaning of constants appearing in the statement of Lemma* 5.1. *Let us consider, e.g.,* $\cos(\theta_{\min}\{(\mathcal{G}+\tilde{\mathcal{G}}) \ominus \mathcal{G}, \mathcal{F}\})$. *The cosine takes its maximal value, one, when at least one direction of the perturbation of* $\mathcal{G}$ *is* A*-orthogonal to* $\mathcal{G}$ *and parallel to* $\mathcal{F}$ *at the same time. It is small, on the contrary, when a part of the perturbation,* A*-orthogonal to* $\mathcal{G}$*, is also* A*-orthogonal to* $\mathcal{F}$. *As* $(\mathcal{G}+\tilde{\mathcal{G}}) \ominus \mathcal{G} \subseteq \mathcal{G}^{\perp}$, *we have*

$$\cos(\theta_{\min}\{(\mathcal{G}+\tilde{\mathcal{G}}) \ominus \mathcal{G}, \mathcal{F}\}) \le \cos\left(\theta_{\min}\{\mathcal{G}^{\perp}, \mathcal{F}\}\right) = \sin\left(\theta_{\max}\{\mathcal{G}, \mathcal{F}\}\right) = \mathrm{gap}_A(\mathcal{G}, \mathcal{F}),$$

which is the constant of the asymptotic perturbation estimate of [3] *(where* $A = I$*). The latter constant is small if subspaces* $\mathcal{G}$ *and* $\mathcal{F}$ *are close to each other, which can be considered more as a cancellation prize as in this case cosine of all principal angles is almost one, and a perturbation estimate for the cosine does not help much because of the cancellation effect.*

REMARK 5.2. *A natural approach similar to that of* [15] *with* $A = I$ *involves a simpler identity:*

$$Q_F^T A Q_{\tilde{G}} = Q_F^T A Q_G + Q_F^T A (Q_{\tilde{G}} - Q_G),$$

where a norm of the second term is then estimated. Then (5.3) *gives an estimate of singular values using* $\|A^{1/2}(Q_{\tilde{G}} - Q_G)\|$. *As singular values are invariant with respect to particular choices of matrices* $Q_{\tilde{G}}$ *and* Q_G *with* A*-orthonormal columns, as far as they provide ranges* $\tilde{\mathcal{G}}$ *and* $\mathcal{G}$*, respectively, we can choose them to minimize the norm of the difference, which gives*

$$\inf_{Q} \|A^{1/2}(Q_G - Q_{\tilde{G}} Q)\|, \tag{5.5}$$

where Q *is an arbitrary* q*-by-*q *orthogonal matrix. This quantity appears in* [15] *with* $A = I$ *as a special type of the Procrustes problem. In* [15]*, it is estimated in terms of the gap between subspaces* $\tilde{\mathcal{G}}$ *and* $\mathcal{G}$ *(using an extra assumption that* $2q \le n$*). Repeating similar arguments, we derive*

$$|\sigma_k - \hat{\sigma}_k| \le \inf_{Q} \|A^{1/2}(Q_G - Q_{\tilde{G}} Q)\|_A \le \sqrt{2}\,\mathrm{gap}_A(\mathcal{G}, \tilde{\mathcal{G}}), \qquad k = 1, \ldots, q. \tag{5.6}$$

Lemma 5.1 *furnishes estimates of the perturbation of singular values in terms of the gap directly, which gives a much better constant, consistent with that of the asymptotic estimate of* [3] *for* $A = I$*; see the previous remark.*

Now we prove a separate estimate for sine.

LEMMA 5.2. *Let* $\mu_1 \le \mu_2 \le \cdots \le \mu_q$ *and* $\hat{\mu}_1 \le \hat{\mu}_2 \le \cdots \le \hat{\mu}_q$ *be sine of principal angles between subspaces* $\mathcal{F}$, $\mathcal{G}$, *and* $\mathcal{F}$, $\tilde{\mathcal{G}}$, *respectively, computed in the* A*-based scalar product. Then, for* $k = 1, \ldots, q$,

(5.7)
$$|\mu_k - \hat{\mu}_k| \le \max\{\sin(\theta_{\max}\{(\mathcal{G}+\tilde{\mathcal{G}}) \ominus \mathcal{G}, \mathcal{F}\}); \sin(\theta_{\max}\{(\mathcal{G}+\tilde{\mathcal{G}}) \ominus \tilde{\mathcal{G}}, \mathcal{F}\})\} \mathrm{gap}_A(\mathcal{G}, \tilde{\mathcal{G}}),$$

where $\theta_{\max}$ *is the largest angle between corresponding subspaces, measured in the* A*-based scalar product.*

Proof. The proof is based on the following identity:

$$\begin{aligned} A^{1/2}(I - Q_F Q_F^T A)Q_{\tilde{G}} = {} & A^{1/2}(I - Q_F Q_F^T A)Q_G Q_G^T A Q_{\tilde{G}} \\ & + A^{1/2}(I - Q_F Q_F^T A)(I - Q_G Q_G^T A)Q_{\tilde{G}}, \end{aligned}$$

which is a multidimensional analogue of the trigonometric formula for the sine of the sum of two angles. The rest of the proof is similar to that of Lemma 5.1.

We first need to take $T = A^{1/2}(I - Q_F Q_F^T A)Q_G Q_G^T A Q_{\tilde{G}}$ and $S = A^{1/2}(I - Q_F Q_F^T A)(I - Q_G Q_G^T A)Q_{\tilde{G}}$ and use (5.3) to get

$$\begin{aligned} s_k(A^{1/2}(I - Q_F Q_F^T A)Q_{\tilde{G}}) \le {} & s_k(A^{1/2}(I - Q_F Q_F^T A)Q_G Q_G^T A Q_{\tilde{G}}) \\ & + \|A^{1/2}(I - Q_F Q_F^T A)(I - Q_G Q_G^T A)Q_{\tilde{G}}\|. \end{aligned}$$

In the second term in the sum on the right, $Q_F Q_F^T A = P_F$ and we deduce

$$A^{1/2}(I - Q_F Q_F^T A)(I - Q_G Q_G^T A)Q_{\tilde{G}} = A^{1/2}(I - P_F)P_{(\mathcal{G}+\tilde{\mathcal{G}})\ominus\mathcal{G}}(I - Q_G Q_G^T A)Q_{\tilde{G}}$$

$$= \left(A^{1/2}(I - P_F)P_{(\mathcal{G}+\tilde{\mathcal{G}})\ominus\mathcal{G}} A^{-1/2}\right)\left(A^{1/2}(I - Q_G Q_G^T A)Q_{\tilde{G}}\right),$$

using notation $P_{(\mathcal{G}+\tilde{\mathcal{G}})\ominus\mathcal{G}}$ for the A-orthogonal projector on the subspace $(\mathcal{G}+\tilde{\mathcal{G}})\ominus\mathcal{G}$, introduced in the proof of Lemma 5.1. Therefore, the second term can be estimated as

$$\begin{aligned} & \|A^{1/2}(I - Q_F Q_F^T A)(I - Q_G Q_G^T A)Q_{\tilde{G}}\| \\ & \qquad \le \|A^{1/2}(I - P_F)P_{(\mathcal{G}+\tilde{\mathcal{G}})\ominus\mathcal{G}} A^{-1/2}\| \|A^{1/2}(I - Q_G Q_G^T A)Q_{\tilde{G}}\|. \end{aligned}$$

The first multiplier is

$$\|A^{1/2}(I - P_F)P_{(\mathcal{G}+\tilde{\mathcal{G}})\ominus\mathcal{G}} A^{-1/2}\| = \|(I - P_F)P_{(\mathcal{G}+\tilde{\mathcal{G}})\ominus\mathcal{G}}\|_A = \sin(\theta_{\max}\{(\mathcal{G}+\tilde{\mathcal{G}})\ominus\mathcal{G}, \mathcal{F}\})$$

by Theorem 4.7 as $\dim\mathcal{F} \ge \dim((\mathcal{G}+\tilde{\mathcal{G}})\ominus\mathcal{G})$, while the second multiplier is simply $\mathrm{gap}_A(\mathcal{G}, \tilde{\mathcal{G}})$ because of our assumption $\dim\mathcal{G} = \dim\tilde{\mathcal{G}}$.

To estimate the first term in the sum, we take with $T = A^{1/2}(I - Q_F Q_F^T A)Q_G$ and $S^T = Q_G^T A Q_{\tilde{G}}$ and apply (5.4):

$$\begin{aligned} s_k(A^{1/2}(I - Q_F Q_F^T A)Q_G Q_G^T A Q_{\tilde{G}}) & \le s_k(A^{1/2}(I - Q_F Q_F^T A)Q_G)\|Q_G^T A Q_{\tilde{G}}\| \\ & \le s_k(A^{1/2}(I - Q_F Q_F^T A)Q_G), \end{aligned}$$

using exactly the same arguments as in the proof of Lemma 5.1.

Thus, we have proved

$$\hat{\mu}_k \le \mu_k + \sin(\theta_{\max}\{(\mathcal{G}+\tilde{\mathcal{G}}) \ominus \mathcal{G},\, \mathcal{F}\})\mathrm{gap}_A(\mathcal{G},\tilde{\mathcal{G}}).$$

Changing the places of $Q_{\tilde{G}}$ and Q_G, we get

$$\mu_k \le \hat{\mu}_k + \sin(\theta_{\max}\{(\mathcal{G}+\tilde{\mathcal{G}}) \ominus \tilde{\mathcal{G}},\, \mathcal{F}\})\mathrm{gap}_A(\mathcal{G},\tilde{\mathcal{G}}).$$

The statement of the lemma follows. ∎

REMARK 5.3. *Let us also highlight that simpler estimates,*

$$|\mu_k - \hat{\mu}_k| \le \mathrm{gap}_A(\mathcal{G},\tilde{\mathcal{G}}), \quad |\sigma_k - \hat{\sigma}_k| \le \mathrm{gap}_A(\mathcal{G},\tilde{\mathcal{G}}), \quad k = 1,\dots,q,$$

which are not as sharp as those we prove in Lemmas 5.1 *and* 5.2*, can be derived almost trivially using orthoprojectors (see* [29, 27, 14]*), where this approach is used for the case $A = I$. Indeed, we start with identities*

$$A^{1/2}P_F P_{\tilde{G}} A^{-1/2} = A^{1/2}P_F P_G A^{-1/2} + \left(A^{1/2}P_F A^{-1/2}\right)\left(A^{1/2}(P_{\tilde{G}} - P_G)A^{-1/2}\right)$$

for the cosine and

$$\begin{aligned} A^{1/2}(I-P_F)P_{\tilde{G}}A^{-1/2} &= A^{1/2}(I-P_F)P_G A^{-1/2} \\ &\quad + \left(A^{1/2}(I-P_F)A^{-1/2}\right)\left(A^{1/2}(P_{\tilde{G}} - P_G)A^{-1/2}\right) \end{aligned}$$

for the sine, and use (5.3) *and Theorems* 4.6 *and* 4.7*. A norm of the second term is then estimated from above by* $\mathrm{gap}_A(\mathcal{G},\tilde{\mathcal{G}})$*, using the fact that for an A-orthoprojector P_F we have $\|P_F\|_A = \|I - P_F\|_A = 1$.*

Instead of the latter, we can use a bit more sophisticated approach, as in [14]*, if we introduce the A-orthogonal projector $P_{\mathcal{G}+\tilde{\mathcal{G}}}$ on the subspace $\mathcal{G}+\tilde{\mathcal{G}}$. Then the norm of second term is bounded by* $\mathrm{gap}_A(\mathcal{G},\tilde{\mathcal{G}})$ *times $\|P_F P_{\mathcal{G}+\tilde{\mathcal{G}}}\|_A$ for the cosine and times $\|(I-P_F)P_{\mathcal{G}+\tilde{\mathcal{G}}}\|_A$ for the sine, where we can now use Theorem* 4.6 *to provide a geometric interpretation of these two constants. This leads to estimates similar to those of* [14] *for $A = I$:*

$$|\sigma_k - \hat{\sigma}_k| \le \cos(\theta_{\min}\{\mathcal{F},\, \mathcal{G}+\tilde{\mathcal{G}}\})\mathrm{gap}_A(\mathcal{G},\tilde{\mathcal{G}}), \qquad k = 1,\dots,q,$$

and

$$|\mu_k - \hat{\mu}_k| \le \cos(\theta_{\min}\{\mathcal{F}^{\perp},\, \mathcal{G}+\tilde{\mathcal{G}}\})\mathrm{gap}_A(\mathcal{G},\tilde{\mathcal{G}}), \qquad k = 1,\dots,q.$$

However, the apparent constant "improvement" in the second estimate, for the sine, is truly misleading as

$$\cos(\theta_{\min}\{\mathcal{F}^{\perp},\, \mathcal{G}+\tilde{\mathcal{G}}\}) = 1$$

simply because $\dim\mathcal{F} < \dim(\mathcal{G}+\tilde{\mathcal{G}})$ *in all cases except for the trivial possibility $\mathcal{G} = \tilde{\mathcal{G}}$, so subspaces $\mathcal{F}^{\perp}$ and $\mathcal{G}+\tilde{\mathcal{G}}$ must have a nontrivial intersection.*

The first estimate, for the cosine, does give a better constant (compare to one), but our constant is sharper; e.g.,

$$\cos(\theta_{\min}\{\mathcal{F},\, (\mathcal{G}+\tilde{\mathcal{G}}) \ominus G\}) \le \cos(\theta_{\min}\{\mathcal{F},\, \mathcal{G}+\tilde{\mathcal{G}}\}).$$

Our more complex identities used to derive perturbation bounds provide an extra projector in the error term, which allows us to obtain better constants.

We can now establish an estimate of absolute sensitivity of cosine and sine of principal angles with respect to absolute *perturbations of subspaces.*

THEOREM 5.3. *Let $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_q$ and $\tilde{\sigma}_1 \geq \tilde{\sigma}_2 \geq \cdots \geq \tilde{\sigma}_q$ be cosine of principal angles between subspaces $\mathcal{F}$, $\mathcal{G}$, and $\tilde{\mathcal{F}}$, $\tilde{\mathcal{G}}$, respectively, computed in the A-based scalar product. Then*

$$|\sigma_k - \tilde{\sigma}_k| \leq c_1 \mathrm{gap}_A(\mathcal{F}, \tilde{\mathcal{F}}) + c_2 \mathrm{gap}_A(\mathcal{G}, \tilde{\mathcal{G}}), \qquad k = 1, \ldots, q, \tag{5.8}$$

where

$$c_1 = \max\{\cos(\theta_{\min}\{(\mathcal{G} + \tilde{\mathcal{G}}) \ominus \mathcal{G}, \mathcal{F}\}); \cos(\theta_{\min}\{(\mathcal{G} + \tilde{\mathcal{G}}) \ominus \tilde{\mathcal{G}}, \mathcal{F}\})\},$$

$$c_2 = \max\{\cos(\theta_{\min}\{(\mathcal{F} + \tilde{\mathcal{F}}) \ominus \mathcal{F}, \tilde{\mathcal{G}}\}); \cos(\theta_{\min}\{(\mathcal{F} + \tilde{\mathcal{F}}) \ominus \tilde{\mathcal{F}}, \tilde{\mathcal{G}}\})\},$$

where $\theta_{\min}$ is the smallest angle between corresponding subspaces in the A-based scalar product.

Proof. First, by Lemma 5.1, for $k = 1, \ldots, q$,

$$|\sigma_k - \hat{\sigma}_k| \leq \max\{\cos(\theta_{\min}\{(\mathcal{G} + \tilde{\mathcal{G}}) \ominus \mathcal{G}, \mathcal{F}\}); \cos(\theta_{\min}\{(\mathcal{G} + \tilde{\mathcal{G}}) \ominus \tilde{\mathcal{G}}, \mathcal{F}\})\} \mathrm{gap}_A(\mathcal{G}, \tilde{\mathcal{G}}).$$

Second, we apply a similar statement to cosine of principal angles between subspaces $\mathcal{F}$, $\tilde{\mathcal{G}}$ and $\tilde{\mathcal{F}}$, $\tilde{\mathcal{G}}$, respectively, computed in the A-based scalar product:

$$|\tilde{\sigma}_k - \hat{\sigma}_k| \leq \max\{\cos(\theta_{\min}\{(\mathcal{F} + \tilde{\mathcal{F}}) \ominus \mathcal{F}, \tilde{\mathcal{G}}\}); \cos(\theta_{\min}\{(\mathcal{F} + \tilde{\mathcal{F}}) \ominus \tilde{\mathcal{F}}, \tilde{\mathcal{G}}\})\} \mathrm{gap}_A(\mathcal{F}, \tilde{\mathcal{F}}).$$

The statement of the theorem now follows from the triangle inequality. ∎

THEOREM 5.4. *Let $\mu_1 \leq \mu_2 \leq \cdots \leq \mu_q$ and $\tilde{\mu}_1 \leq \tilde{\mu}_2 \leq \cdots \leq \tilde{\mu}_q$ be sine of principal angles between subspaces $\mathcal{F}$, $\mathcal{G}$, and $\tilde{\mathcal{F}}$, $\tilde{\mathcal{G}}$, respectively, computed in the A-based scalar product. Then*

$$|\mu_k - \tilde{\mu}_k| \leq c_3 \mathrm{gap}_A(\mathcal{F}, \tilde{\mathcal{F}}) + c_4 \mathrm{gap}_A(\mathcal{G}, \tilde{\mathcal{G}}), \qquad k = 1, \ldots, q, \tag{5.9}$$

where

$$c_3 = \max\{\sin(\theta_{\max}\{(\mathcal{G} + \tilde{\mathcal{G}}) \ominus \mathcal{G}, \mathcal{F}\}); \sin(\theta_{\max}\{(\mathcal{G} + \tilde{\mathcal{G}}) \ominus \tilde{\mathcal{G}}, \mathcal{F}\})\},$$

$$c_4 = \max\{\sin(\theta_{\max}\{(\mathcal{F} + \tilde{\mathcal{F}}) \ominus \mathcal{F}, \tilde{\mathcal{G}}\}); \sin(\theta_{\max}\{(\mathcal{F} + \tilde{\mathcal{F}}) \ominus \tilde{\mathcal{F}}, \tilde{\mathcal{G}}\})\},$$

where $\theta_{\max}$ is the largest angle between corresponding subspaces in the A-based scalar product.

Proof. First, by Lemma 5.2, for $k = 1, \ldots, q$,

$$|\mu_k - \hat{\mu}_k| \leq \max\{\sin(\theta_{\max}\{(\mathcal{G} + \tilde{\mathcal{G}}) \ominus \mathcal{G}, \mathcal{F}\}); \sin(\theta_{\max}\{(\mathcal{G} + \tilde{\mathcal{G}}) \ominus \tilde{\mathcal{G}}, \mathcal{F}\})\} \mathrm{gap}_A(\mathcal{G}, \tilde{\mathcal{G}}).$$

Second, we apply a similar statement to sine of principal angles between subspaces $\mathcal{F}$, $\tilde{\mathcal{G}}$ and $\tilde{\mathcal{F}}$, $\tilde{\mathcal{G}}$, respectively, computed in the A-based scalar product:

$$|\tilde{\mu}_k - \hat{\mu}_k| \leq \max\{\sin(\theta_{\max}\{(\mathcal{F} + \tilde{\mathcal{F}}) \ominus \mathcal{F}, \tilde{\mathcal{G}}\}); \sin(\theta_{\max}\{(\mathcal{F} + \tilde{\mathcal{F}}) \ominus \tilde{\mathcal{F}}, \tilde{\mathcal{G}}\})\} \mathrm{gap}_A(\mathcal{F}, \tilde{\mathcal{F}}).$$

The statement of the theorem now follows from the triangle inequality. ∎

Finally, we want a perturbation analysis in terms of matrices F and G that generate subspaces $\mathcal{F}$ and $\mathcal{G}$. For that, we have to estimate the sensitivity of a column-space of a matrix, for example, matrix G.

LEMMA 5.5. *Let*

$$\kappa_A(G) = \frac{s_{\max}(A^{1/2}G)}{s_{\min}(A^{1/2}G)}$$

denote the corresponding A-based condition number of G, where $s_{\max}$ and $s_{\min}$ are, respectively, largest and smallest singular values of the matrix $A^{1/2}G$. Let $\mathcal{G}$ and $\tilde{\mathcal{G}}$ be column-spaces of matrices G and $\tilde{G}$, respectively. Then

$$\operatorname{gap}_A(\mathcal{G}, \tilde{\mathcal{G}}) \le \kappa_A(G) \frac{\|A^{1/2}(G - \tilde{G})\|}{\|A^{1/2}G\|}. \tag{5.10}$$

Proof. Here, we essentially just adopt the corresponding proof of [29] for the A-based scalar product using the same approach as in Theorem 4.2.

Let us consider the polar decompositions

$$A^{1/2}G = A^{1/2}Q_G T_G \text{ and } A^{1/2}\tilde{G} = A^{1/2}Q_{\tilde{G}}T_{\tilde{G}},$$

where matrices $A^{1/2}Q_G$ and $A^{1/2}Q_{\tilde{G}}$ have orthonormal columns and matrices T_G and $T_{\tilde{G}}$ are q-by-q symmetric positive definite; e.g., $T_G = (Q_G Q_G^T)^{1/2}$. Singular values of T_G and $T_{\tilde{G}}$ are, therefore, the same as singular values of $A^{1/2}G$ and $A^{1/2}\tilde{G}$, respectively. Then,

$$(I - P_{\tilde{G}})(G - \tilde{G}) = (I - P_{\tilde{G}})Q_G T_G.$$

Therefore,

$$A^{1/2}(I - P_{\tilde{G}})Q_G = \left(A^{1/2}(I - P_{\tilde{G}})A^{-1/2}\right) A^{1/2}(G - \tilde{G})T_G^{-1},$$

and

$$\operatorname{gap}_A(\mathcal{G}, \tilde{\mathcal{G}}) \le \|A^{1/2}(G - \tilde{G})\|\|T_G^{-1}\| = \frac{\|A^{1/2}(G - \tilde{G})\|}{s_{\min}(A^{1/2}G)},$$

as $\|A^{1/2}(I - P_{\tilde{G}})A^{-1/2}\| = \|I - P_{\tilde{G}}\|_A \le 1$. The statement of the lemma follows. ∎

REMARK 5.4. *Some matrices allow improvement of their condition numbers by column scaling, which trivially does not change the column range. Our simple Lemma* 5.5 *does not capture this property. A more sophisticated variant can be easily obtained using a technique developed in* [15, 14].

Our cosine theorem follows next. It generalizes results of [27, 14, 15] to A-based scalar products and somewhat improves the constant.

THEOREM 5.6. *Under assumptions of Theorem* 5.3,

$$|\sigma_k - \tilde{\sigma}_k| \le c_1 \kappa_A(F) \frac{\|A^{1/2}(F - \tilde{F})\|}{\|A^{1/2}F\|} + c_2 \kappa_A(G) \frac{\|A^{1/2}(G - \tilde{G})\|}{\|A^{1/2}G\|}, \qquad k = 1, \dots, q. \tag{5.11}$$

The theorem above does not provide an accurate estimate for small angles. To fill the gap, we suggest the following perturbation theorem in terms of sine of principal angles; cf. [27, 14] for $A = I$.

THEOREM 5.7. *Under assumptions of Theorem* 5.4,

$$|\mu - \tilde{\mu}_k| \leq c_3 \kappa_A(F) \frac{\|A^{1/2}(F - \tilde{F})\|}{\|A^{1/2}F\|} + c_4 \kappa_A(G) \frac{\|A^{1/2}(G - \tilde{G})\|}{\|A^{1/2}G\|}, \qquad k = 1, \ldots, q. \tag{5.12}$$

Finally, let us underline that all sensitivity results of the present paper are for *absolute* errors. Golub and Zha in [14] observe that relative errors of sine and cosine of principal angles are not, in general, bounded by the perturbation.

We consider algorithms of computing principal angles with respect to an A-based scalar product in the next section.

6. Algorithm implementation. In this section, we provide a detailed description of our code SUBSPACEA.m and discuss the algorithm implementation.

Theorem 4.4 is our main theoretical foundation for a sine-based algorithm for computing principal angles with respect to an A-based scalar product. However, a naive implementation, using the SVD of the matrix $S^T AS$, may not produce small angles accurately in computer arithmetic. We now try to explain informally this fact, which is actually observed in numerical tests.

Let, for simplicity of notation, all principal angles be small.

Let us consider a particular case, where columns of matrices Q_F and Q_G are already principal vectors in exact arithmetic. In reality, this is the situation we will face in Algorithm 6.2. Then, in exact arithmetic, columns of S are A-orthogonal and their A-norms are exactly the sine of principal angles. Thus, if there are several small angles different in orders of magnitude, the columns of S are badly scaled. When we take the norms squared, by explicitly computing the product $S^T AS$, we make the scaling even worse, as the diagonal entries of this diagonal matrix are now the sine of the principal angles squared, in exact arithmetic. In the presence of round-off errors, the matrix $S^T AS$ is usually not diagonal; thus, principal angles smaller than 10^{-8} will lead to an underflow effect in double precision, which cannot be cured by taking square roots of its singular values later in the algorithm.

To resolve this, we want to be able to compute the SVD of the matrix $S^T AS$ without using either $S^T AS$ itself or $A^{1/2}S$. One possibility is suggested in the following lemma.

LEMMA 6.1. *The SVD of the matrix $A^{1/2}S$ coincides with the SVD of the matrix $Q_S^T AS$, where Q_S is a matrix with A-orthonormal columns, which span the same column-space as columns of matrix S.*

Proof. We have

$$(Q_S^T AS)^T Q_S^T AS = S^T A Q_S Q_S^T AS = S^T A P_S S = S^T AS,$$

where $P_S = Q_S Q_S^T A$ is the A-orthogonal projector on the column-space of Q_S, which is the same, by definition, as the column-space of S, so that $P_S S = S$. ∎

This contributes to the accuracy of our next Algorithm 6.1, based on Lemma 6.1, to be more reliable in the presence of round-off errors, when several principal angles are small.

By analogy with Algorithms 3.1 and 3.2, we can remove the restriction that matrix Σ is invertible and somewhat improve the costs in Algorithm 6.1 by reducing the size of the matrix S in step 4, which leads to Algorithm 6.2.

ALGORITHM 6.1: **SUBSPACEA.m.**

Input: *real matrices F and G with the same number of rows, and a symmetric positive definite matrix A for the scalar product, or a device to compute Ax for a given vector x.*

1. *Compute A-orthonormal bases* $Q_F = \mathrm{ortha}(F)$, $Q_G = \mathrm{ortha}(G)$ *of column-spaces of* F *and* G.
2. *Compute SVD for cosine* $[Y, \Sigma, Z] = \mathrm{svd}(Q_F^T A Q_G)$, $\Sigma = \mathrm{diag}(\sigma_1, \ldots, \sigma_q)$.
3. *Compute matrices of left* $U_{\cos} = Q_F Y$ *and right* $V_{\cos} = Q_G Z$ *principal vectors.*
4. *Compute the matrix* $S = \begin{cases} Q_G - Q_F(Q_F^T A Q_G) & \text{if } \mathrm{rank}(Q_F) \geq \mathrm{rank}(Q_G), \\ Q_F - Q_G(Q_G^T A Q_F) & \text{otherwise.} \end{cases}$
5. *Compute A-orthonormal basis* $Q_S = \mathrm{ortha}(S)$ *of the column-space of* S.
6. *Compute SVD for sine:* $[Y, \mathrm{diag}(\mu_1, \ldots, \mu_q), Z] = \mathrm{svd}(Q_S^T A S)$.
7. *Compute matrices* $U_{\sin}$ *and* $V_{\sin}$ *of left and right principal vectors:*
   $V_{\sin} = Q_G Z$, $U_{\sin} = Q_F(Q_F^T A V_{\sin})\Sigma^{-1}$ *if* $\mathrm{rank}(Q_F) \geq \mathrm{rank}(Q_G)$;
   $U_{\sin} = Q_F Z$, $V_{\sin} = Q_G(Q_G^T A U_{\sin})\Sigma^{-1}$ *otherwise.*
8. *Compute the principal angles, for* $k = 1, \ldots, q$:
   $\theta_k = \begin{cases} \arccos(\sigma_k) & \text{if } \sigma_k^2 < 1/2, \\ \arcsin(\mu_k) & \text{if } \mu_k^2 \leq 1/2. \end{cases}$
9. *Form matrices U and V by picking up corresponding columns of* $U_{\sin}$, $V_{\sin}$ *and* $U_{\cos}$, $V_{\cos}$, *according to the choice for* θ_k *above.*

Output: *Principal angles $\theta_1, \ldots, \theta_q$ between column-spaces of matrices F and G in the A-based scalar product, and corresponding matrices of left, U, and right, V, principal vectors.*

ALGORITHM 6.2: **Improved SUBSPACEA.m.**

Input: *real matrices F and G with the same number of rows, and a symmetric positive definite matrix A for the scalar product, or a device to compute Ax for a given vector x.*

1. *Compute A-orthogonal bases* $Q_F = \mathrm{ortha}(F)$, $Q_G = \mathrm{ortha}(G)$ *of column-spaces of* F *and* G.
2. *Compute SVD for cosine* $[Y, \Sigma, Z] = \mathrm{svd}(Q_F^T A Q_G)$, $\Sigma = \mathrm{diag}(\sigma_1, \ldots, \sigma_q)$.
3. *Compute matrices of left* $U_{\cos} = Q_F Y$ *and right* $V_{\cos} = Q_G Z$ *principal vectors.*
4. *Compute large principal angles for* $k = 1, \ldots, q$:
   $\theta_k = \arccos(\sigma_k)$ *if* $\sigma_k^2 < 1/2$.
5. *Form parts of matrices U and V by picking up corresponding columns of* $U_{\cos}$, $V_{\cos}$, *according to the choice for* θ_k *above. Put columns of* $U_{\cos}$, $V_{\cos}$, *which are left, in matrices* R_F *and* R_G. *Collect the corresponding σ's in a diagonal matrix* Σ_R.
6. *Compute the matrix* $S = R_G - Q_F(Q_F^T A R_G)$.
7. *Compute A-orthogonal basis* $Q_S = \mathrm{ortha}(S)$ *of the column-space of* S.
8. *Compute SVD for sine:* $[Y, \mathrm{diag}(\mu_1, \ldots, \mu_q), Z] = \mathrm{svd}(Q_S^T A S)$.
9. *Compute matrices* $U_{\sin}$ *and* $V_{\sin}$ *of left and right principal vectors:*
   $V_{\sin} = R_G Z$, $U_{\sin} = R_F(R_F^T A V_{\sin})\Sigma_R^{-1}$.
10. *Compute the missing principal angles, for* $k = 1, \ldots, q$:
    $\theta_k = \arcsin(\mu_k)$ *if* $\mu_k^2 \leq 1/2$.
11. *Complete matrices U and V by adding columns of* $U_{\sin}$, $V_{\sin}$.

Output: *Principal angles $\theta_1, \ldots, \theta_q$ between column-spaces of matrices F and G, and corresponding matrices U and V of left and right principal vectors, respectively.*

Previous remarks of section 3 for the algorithms with $A = I$ are applicable to the present algorithms with self-evident changes. A few additional A-specific remarks follow.

REMARK 6.1. *In step* 1 *of Algorithms* 6.1 *and* 6.2*, we use our SVD-based function ORTHA.m for the A-orthogonalization. Specifically, computing an A-orthogonal basis Q of the column-space of a matrix X is done in three steps in ORTHA.m. First, we orthonormalize X, using SVD-based built-in MATLAB code ORTH.m with a preceding explicit column scaling; see Remark* 3.1 *on whether the scaling is actually needed. Second, we compute* $[U, S, V] = \mathrm{svd}(X^T AX)$*, using MATLAB's built-in SVD code. Finally, we take* $Q = XUS^{-1/2}$*. If A is ill-conditioned, an extra cycle may be performed to improve the accuracy. While formal stability and accuracy analysis of this method is yet to be done, ORTHA.m demonstrates practical robustness in our numerical tests. A detailed description and investigation of the algorithm used in ORTHA.m will be reported elsewhere.*

REMARK 6.2. *We note that, when* $n \gg p$*, the computational costs of SVDs of p-by-p matrices are negligible; it is multiplication by A, which may be very computationally expensive. Therefore, we want to minimize the number of multiplications by A. In the present version* 4.0 *of our code SUBSPACEA.m, based on Algorithm* 6.2*, we multiply matrix A by a vector* $2p + q$ *times in the worst-case scenario of all angles being small, in steps* 1 *and* 7*. We can avoid multiplying by A on steps* 2*,* 8*, and* 9 *by using appropriate linear combinations of earlier computed vectors instead.*

REMARK 6.3. *Our actual code is written for a more general complex case, where we require matrix A to be Hermitian.*

Let us finally underline that in a situation when a matrix K from the factorization $A = K^T K$ is given rather than the matrix A itself, we do not advise using Algorithms 6.1 and 6.2. Instead, we recommend multiplying matrices F and G by K on the right and using simpler Algorithms 3.1 and 3.2, according to Theorem 4.2.

7. Numerical examples. Numerical tests in the present section were performed using version 4.0 of our SUBSPACEA.m code, based on Algorithms 3.2 and 6.2. Numerical results presented were obtained, unless indicated otherwise, on Red Hat 6.1 LINUX Dual Pentium-III 500, running MATLAB release 12, version 6.0.0.18. Tests were also made on Compaq Dual Alpha server DS 20, running MATLAB release 12, version 6.0.0.18 and on several Microsoft Windows Pentium-III systems, running MATLAB version 5.1–5.3. In our experience, Intel PIII-based LINUX systems typically provided more accurate results, apparently utilizing the extended 80 bit precision of FPU registers of PIII; see the discussion at the end of the section.

The main goal of our numerical tests is to check a practical robustness of our code, using the following argument. According to our analysis of section 5 and similar results of [3, 27, 14, 15], an absolute change in cosine and sine of principal angles is bounded by perturbations in matrices F and G, with the constant, proportional to their condition numbers taken after a proper column scaling; see Theorems 5.6 and 5.12 and Remark 5.4. Assuming a perturbation of entries of matrices F and G at the level of double precision, $EPS \approx 10^{-16}$, we expect a similar perturbation in cosine and sine of principal angles, when matrices F and G are well-conditioned after column scaling. We want to check if our code achieves this accuracy in practice.

We concentrate here on testing our sine-based algorithms, i.e., for principal angles smaller than $\pi/4$. The cosine-based algorithm with $A = I$ is recently studied in [10].

Our first example is taken from [3] with $p = 13$ and $m = 26$. Matrices F and G were called A and B in [3]. F was orthogonal, while G was an m-by-p Vandermonde

matrix with $\mathrm{cond}(G) \approx 10^4$. Matrix G was generated in double precision and then rounded to single precision.

According to our theory and a perturbation analysis of [3, 27, 14, 15], in this example an absolute change in principal angles is bounded by a perturbation in matrix G times its condition number. Thus, we should expect sine and cosine of principal angles computed in [3] to be accurate with approximately four decimal digits.

In our code, all computations are performed in double precision; therefore, answers in Table 7.1 should be accurate up to twelve decimal digits. We observe, as expected, that our results are consistent with those of [3] within four digits.

TABLE 7.1
Computed sine and cosine of principal angles of the example of [3].

k	$\sin(\theta_k)$	$\cos(\theta_k)$
1	0.00000000000	1.00000000000
2	0.05942261363	0.99823291519
3	0.06089682091	0.99814406635
4	0.13875176720	0.99032719194
5	0.14184708183	0.98988858230
6	0.21569434797	0.97646093022
7	0.27005046021	0.96284617096
8	0.33704307148	0.94148922881
9	0.39753678833	0.91758623677
10	0.49280942462	0.87013727135
11	0.64562133627	0.76365770483
12	0.99815068733	0.06078820101
13	0.99987854229	0.01558527040

In our next series of tests, we assume n to be even and $p = q \le n/2$. Let D be a diagonal matrix of the size p:

$$D = \mathrm{diag}(d_1, \ldots, d_p), \quad d_k > 0, \quad k = 1, \ldots, p.$$

We first define n-by-p matrices

$$F_1 = [I\ 0]^T, \quad G_1 = [I\ D\ 0]^T, \tag{7.1}$$

where I is the identity matrix of the size p and 0 are zero matrices of appropriate sizes. We notice that condition numbers of F_1 and G_1 are, respectively, one and

$$\mathrm{cond}\, G_1 = \sqrt{\frac{1 + (\max\{\mathrm{diag}(D)\})^2}{1 + (\min\{\mathrm{diag}(D)\})^2}}.$$

Thus, the condition number may be large only when large diagonal entries in D are present. Yet in this case, the condition number of G_1 can be significantly reduced by column scaling; see Remark 5.4.

The exact values of sine and cosine of principal angles between column-spaces of matrices F_1 and G_1 are obviously given by

$$\mu_k = \frac{d_k}{\sqrt{1 + d_k^2}}, \quad \sigma_k = \frac{1}{\sqrt{1 + d_k^2}}, \quad k = 1, \ldots, p, \tag{7.2}$$

respectively, assuming that d_k's are sorted in the increasing order. The collective error in principal angles is measured as the following sum:

$$\sqrt{(\mu_1 - \tilde{\mu}_1)^2 + \cdots + (\mu_p - \tilde{\mu}_p)^2} + \sqrt{(\sigma_1 - \tilde{\sigma}_1)^2 + \cdots + (\sigma_p - \tilde{\sigma}_p)^2}, \tag{7.3}$$

where μ's are the sine and σ's are the cosine of principal angles, and the tilde sign $\tilde{}$ is used for actual computed values.

We multiply matrices F_1 and G_1 by a random orthogonal matrix U of the size n on the left to get

$$F_2 = U * F_1, \quad G_2 = U * G_1. \tag{7.4}$$

This transformation does not change angles and condition numbers. It still allows for improving the condition number of G_2 by column scaling.

Finally, we multiply matrices by random orthogonal p-by-p matrices T_F and T_G, respectively, on the right:

$$F_3 = F_2 * T_F, \quad G_3 = G_2 * T_G. \tag{7.5}$$

This transformation does not change angles or condition numbers. It is likely, however, to remove the possibility of improving the condition number of G_3 by column scaling. Thus, if G_3 is ill-conditioned, we could expect a loss of accuracy; see Theorems 5.6 and 5.12 and Remark 5.4.

We start by checking scalability of the code for well-conditioned cases. We increase the size of the problem n and plot the collective error, given by (7.3), for the principal angles between F_3 and G_3, against the value $n/2$. We solve the same problem two times, using Algorithm 3.2 and Algorithm 6.2 with $A = I$.

In the first two tests, diagonal entries of D are chosen as uniformly distributed random numbers *rand* on the interval $(0, 1)$. On Figure 7.1 (top) we fix $p = q = 20$. We observe that the average error grows approximately two times with a ten times increase in the problem size. On Figure 7.1 (bottom), we also raise $p = q = n/2$. This time, the error grows with the same pace as the problem size. Please note different scales used for errors on Figure 7.1.

To test our methods for very small angles with $p = q = 20$, we first choose

$$p = 20, \quad d_k = 10^{-k}, \quad k = 1, \dots, p.$$

We observe a similar pattern of the absolute error as that of Figure 7.1 (top) and do not reproduce this figure here because of the space limitations.

In our second test for small angles, we set $p = q = n/2$ as on Figure 7.1 (bottom) and select every diagonal element of D in the form $10^{-17 \cdot rand}$, where *rand* is again a uniformly distributed random number on the interval $(0, 1)$. The corresponding figure, not shown here for the sake of brevity, looks the same as Figure 7.1 (bottom), except that the error grows a bit faster and reaches the level $\approx 4 \cdot 10^{-14}$ (compare to $\approx 3 \cdot 10^{-14}$ value on Figure 7.1 (bottom)).

In all these and our other analogous tests, Algorithm 3.2 and Algorithm 6.2 with $A = I$ behave very similarly, so that Figure 7.1 provides a typical example.

In our next series of experiments, we fix a small $p = q$ and $n = 100$, and compute angles between F_2 and G_2 and between F_3 and G_3 500 times, changing only the random matrices used in the construction of our F_i and G_i. Instead of the collective error, given by (7.3), we now compute errors for individual principal angles as

$$|\mu_k - \tilde{\mu}_k| + |\sigma_k - \tilde{\sigma}_k|, \qquad k = 1, \dots, p.$$

We tested several different combinations of angles less than $\pi/4$. In most cases, the error was only insignificantly different from $EPS \approx 10^{-16}$. The worst-case scenario found numerically corresponds to

$$D = \operatorname{diag}\{1, 0.5, 10^{-11}, 10^{-12}, 10^{-13}, 5 \cdot 10^{-15}, 2 \cdot 10^{-15}, 10^{-15}, 10^{-16}, 0\}$$

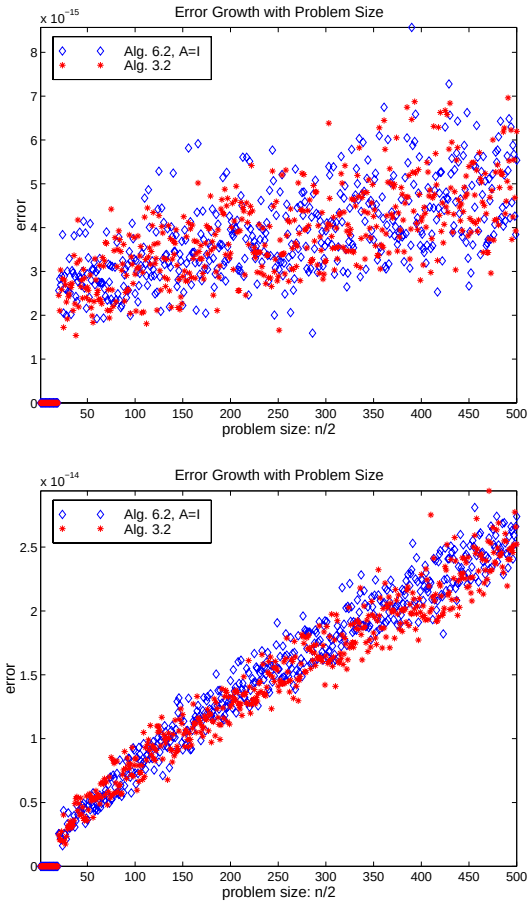

FIG. 7.1. *Errors in principal angles as functions of* $n/2$*:* $p = 20$ *(top) and* $p = n/2$ *(bottom).*

and is presented on Figure 7.2. Figure 7.2 (top) shows the distribution of the error for individual angles between F_3 and G_3 in Algorithm 3.2 in 500 runs. Figure 7.2 (bottom) demonstrates the performance of Algorithm 6.2 with $A = I$, for the same problem. The numeration of angles is reversed for technical reasons; i.e., smaller angles are further away from the viewer. Also the computed distribution of the error for individual angles between F_2 and G_2 are very similar and, because of that, are not shown here.

We detect that for such small values of p and n most of the angles are computed essentially within the double precision accuracy. Only multiple small angles present a slight challenge, more noticeable on Figure 7.2 (bottom), which uses a somewhat different scale for the error to accommodate a larger error. Nevertheless, all observed errors in all 500 runs are bounded from above by $\approx 6{\cdot}10^{-15}$ on Figure 7.2, which seems to be a reasonable level of accuracy for accumulation of round-off errors appearing in computations with 200-by-10 matrices in double precision.

To test our code for an ill-conditioned case, we add two large values to the previous choice of D to obtain

$$D = \mathrm{diag}\{10^{10}, 10^{8}, 1, 0.5, 10^{-11}, 10^{-12}, 10^{-13}, 5 \cdot 10^{-15}, 2 \cdot 10^{-15}, 10^{-15}, 10^{-16}, 0\},$$

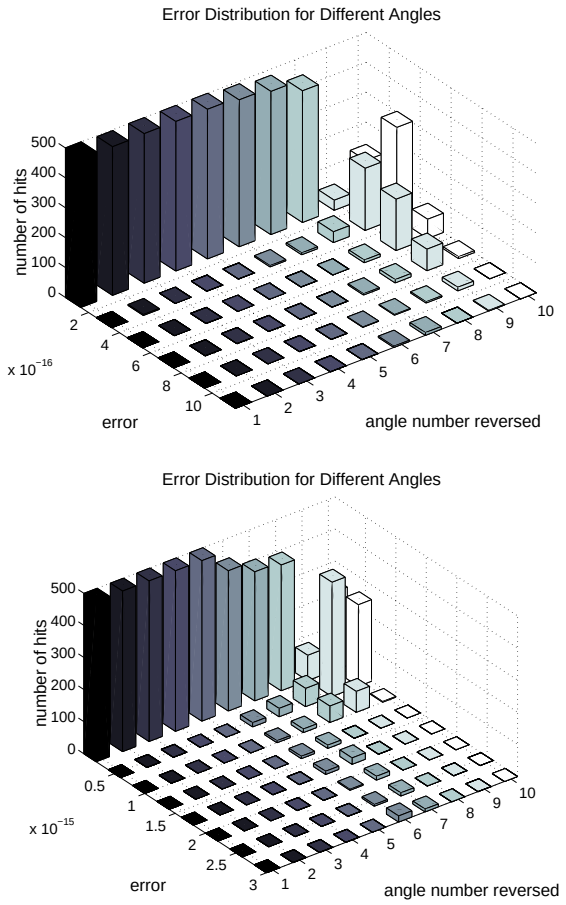

FIG. 7.2. *Errors in individual angles in Algorithms* 3.2 *(top) and* 6.2 *with* $A = I$ *(bottom).*

which leads to $\operatorname{cond} G_1 \approx 10^{10}$.

Figure 7.3 (top) shows the distribution of the error for individual angles between F_2 and G_2 in Algorithm 6.2 with $A = I$ after 500 runs. The numeration of angles is again reversed; i.e., smaller angles are further away from the viewer. There is no visible difference between the new Figure 7.3 (top) and the previous Figure 7.2 for the well-conditioned case, which confirms results of [15, 14] on column scaling of ill-conditioned matrices; see Remark 5.4. Namely, our ill-conditioned matrix G_2 can be made well-conditioned by column scaling. Thus, perturbations in the angles should be small. Figure 7.3 (top) therefore demonstrates that our code SUBSPACEA.m is able to take advantage of this.

As we move to computing angles between F_3 and G_3 in this example (see Figure 7.3 (bottom)), where the distribution of the error for individual angles in Algorithm 6.2 with $A = I$ after 500 runs is shown, the situation changes dramatically. In general, it is not possible to improve the condition number $\operatorname{cond}(G_3) \approx 10^{10}$ by column scaling. Thus, according to [3, 15] and our perturbation analysis of Theorems 5.6 and 5.12 and Remark 5.4, we should expect the absolute errors in angles to grow $\approx 10^{10}$ times (compare to the errors on Figure 7.3 (bottom)), i.e., up to the level 10^{-5}. On Figure

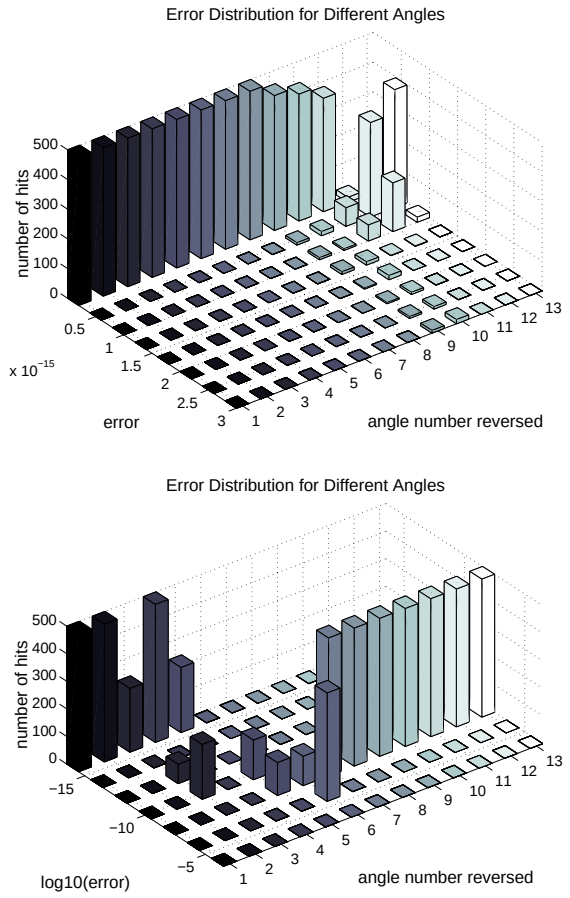

FIG. 7.3. *Errors in individual angles between F_2 and G_2 (top) and F_3 and G_3 (bottom).*

7.3 (bottom), which shows actual errors obtained during 500 runs of the code, we indeed observe absolute errors in angles at the level up to 10^{-5}, as just predicted. Surprisingly, the absolute error for angles with small cosines is much better. We do not have an explanation of such good behavior, as the present theory does not provide individual perturbation bounds for different angles.

Our concluding numerical results illustrate performance of our code for ill-conditioned scalar products.

We take G to be the first ten columns of the identity matrix of size twenty, and F to be the last ten columns of the Vandermonde matrix of size twenty with elements $v_{i,j} = i^{20-j}$, $i, j = 1, \ldots, 20$. Matrix F is ill-conditioned, $\mathrm{cond}F \approx 10^{13}$. We compute principal angles and vectors between F and G in an A-based scalar product for the following family of matrices:

$$A = A_l = 10^{-l} I + H, \qquad l = 1, \ldots, 16,$$

where I is identity and H is the Hilbert matrix of the order twenty, whose elements are given by $h_{i,j} = 1/(i + j - 1)$, $i, j = 1, \ldots, 20$. Our subspaces $\mathcal{F}$ and $\mathcal{G}$ do not change with l; only the scalar product that describes the geometry of the space changes.

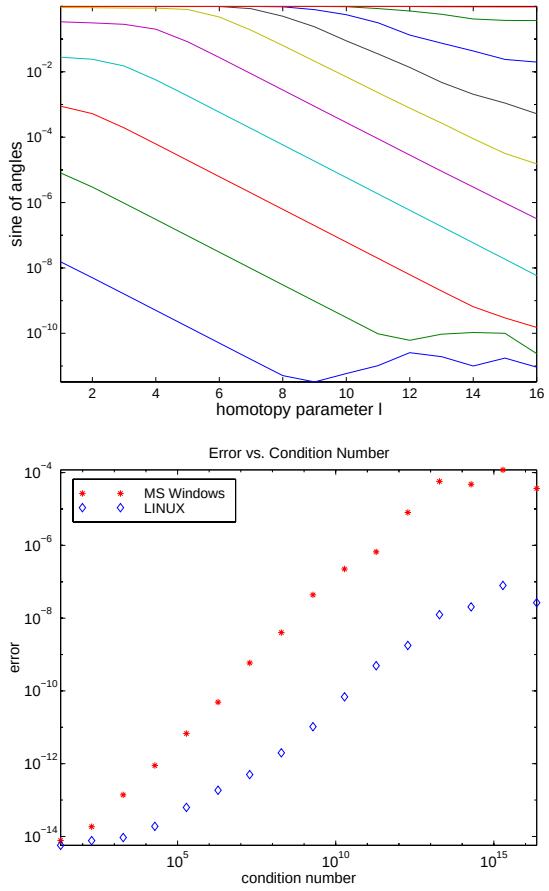

FIG. 7.4. *μ_k as functions of l (top). Errors as functions of condition number (bottom).*

When $l = 1$, we observe three angles with cosine less than 10^{-3} and three angles with sine less than 10^{-3}. When l increases, we are getting closer to the Hilbert matrix, which emphasizes first rows in matrices F and G, effectively ignoring last rows. By construction of F, its large elements, which make subspace $\mathcal{F}$ to be further away from subspace $\mathcal{G}$, are all in last rows. Thus, we should expect large principal angles between $\mathcal{F}$ and $\mathcal{G}$ to decrease monotonically as l grows. We observe this in our numerical tests (see Figure 7.4 (top)), which plots in logarithmic scale sine of all ten principal angles as functions of l.

Of course, such change in geometry that makes sine of an angle to decrease 10^4 times, means that matrix A_l, describing the scalar product, gets more and more ill-conditioned, as it gets closer to Hilbert matrix H, namely, $\text{cond}(A) \approx 10^l$ in our case. It is known that ill-conditioned problems usually lead to a significant increase of the resulting error, as ill-conditioning amplifies round-off errors. To investigate this effect for our code, we introduce the error as the following sum:

$$\text{error} = \|V^T AV - I\| + \|U^T AU - I\| + \|\Sigma - U^T AV\|,$$

where the first two terms control orthogonality of principal vectors and the last term

measures the accuracy of cosine of principal angles. We observe in our experiments that different terms in the sum are close to each other, and none dominates. The accuracy of sine of principal angles is not crucial in this example as angles are not small enough to cause concerns. As U and V are constructed directly from columns of F and G, they span the same subspaces with high accuracy independently of condition number of A, as we observe in the tests.

We plot the error on the y-axis of Figure 7.4 (bottom) for a Pentium III 500 PC running two different operating systems: Microsoft Windows NT 4.0 SP6 (red stars) and RedHat LINUX 6.1 (blue diamonds), where the x-axis presents condition number of A; both axes are in logarithmic scale. The MATLAB Version 5.3.1.29215a (R11.1) is the same on both operating systems.

We see, as expected, that the error grows, apparently linearly, with the condition number. We also observe, now with quite a surprise, that the error on LINUX is much smaller than the error on MS Windows!

As the same MATLAB's version and the same code SUBSPACEA.m are run on the same hardware, this fact deserves an explanation. As a result of a discussion with Nabeel and Lawrence Kirby at the News Group *sci.math.num-analysis*, it has been found that MATLAB was apparently compiled on LINUX to take advantage of extended 80 bit precision of FPU registers of PIII, while Microsoft compilers apparently set the FPU to 64 bit operations. To demonstrate this, Nabeel suggested the following elegant example: compute scalar product

$$(1\ 10^{-19}\ -1)^T\ (1\ 1\ 1).$$

On MS Windows, the result is zero, as it should be in double precision, while on LINUX the result is 1.084210^{-19}.

Figure 7.4 (bottom) shows that our algorithm turns this difference into a significant error improvement for an ill-conditioned problem.

Finally, our code SUBSPACEA.m has been used since 1999 in the code LOBPCG.m (see [20, 21]) to control accuracy of invariant subspaces of large symmetric generalized eigenvalue problems, and thus has been tested for a variety of large-scale practical problems.

8. Availability of the software. See http://www.mathworks.com/support/ftp/linalgv5.shtml for our code SUBSPACEA.m and the function ORTHA.m it uses, as well as our fix for SUBSPACE.m, as submitted to MathWorks. They are also publicly available at the Web page maintained by the first author: http://www-math.cudenver.edu/~aknyazev/software.

9. Conclusion. Let us formulate here the main points of the present paper:

- A bug in the cosine-based algorithm for computing principal angles between subspaces, which prevents one from computing small angles accurately in computer arithmetic, is illustrated.
- An algorithm is presented that computes all principal angles accurately in computer arithmetic and is proved to be equivalent to the standard algorithm in exact arithmetic.
- A generalization of the algorithm to an arbitrary scalar product given by a symmetric positive definite matrix is suggested and justified theoretically.
- Perturbation estimates for absolute errors in cosine and sine of principal angles, with improved constants and for an arbitrary scalar product, are derived.

- A description of the code is given as well as results of numerical tests. The code is robust in practice and provides accurate angles for large-scale and ill-conditioned cases we tested numerically. It is also reasonably efficient for large-scale applications with $n \gg p$.
- Our algorithms are "matrix-free"; i.e., they do not require storing in memory any matrices of the size n and are capable of dealing with A, which may be specified only as a function that multiplies A by a given vector.

Acknowledgments. The authors thank CU-Denver graduate students Sean Jenson, who helped to test our code SUBSPACEA.m, and Chan-Chai Aniwathananon and Saulo Oliveira, who participated in testing the code ORTHA.m.

The authors are also indebted to anonymous referees for their creative and constructive comments.

REFERENCES

[1] N. I. Akhiezer and I. M. Glazman, *Theory of Linear Operators in Hilbert Space*, Dover, New York, 1993.
[2] J. Barlow and J. Demmel, *Computing accurate eigensystems of scaled diagonally dominant matrices*, SIAM J. Numer. Anal., 27 (1990), pp. 762–791.
[3] Å. Björck and G. H. Golub, *Numerical methods for computing angles between linear subspaces*, Math. Comp., 27 (1973), pp. 579–594.
[4] F. Chatelin, *Eigenvalues of Matrices*, John Wiley and Sons, Chichester, UK, 1993.
[5] J. Dauxois and G. M. Nkiet, *Canonical analysis of two Euclidean subspaces and its applications*, Linear Algebra Appl., 264 (1997), pp. 355–388.
[6] C. Davis and W. M. Kahan, *The rotation of eigenvectors by a perturbation.* III, SIAM J. Numer. Anal., 7 (1970), pp. 1–46.
[7] J. Demmel, M. Gu, S. Eisenstat, I. Slapničar, K. Veselić, and Z. Drmač, *Computing the singular value decomposition with high relative accuracy*, Linear Algebra Appl., 299 (1999), pp. 21–80.
[8] J. Demmel and K. Veselić, *Jacobi's method is more accurate than QR*, SIAM J. Matrix Anal. Appl., 13 (1992), pp. 1204–1245.
[9] F. Deutsch, *The angle between subspaces of a Hilbert space*, in Approximation Theory, Wavelets and Applications, Kluwer Academic Publishers, Dordrecht, The Netherlands, 1995, pp. 107–130.
[10] Z. Drmač, *On principal angles between subspaces of Euclidean space*, SIAM J. Matrix Anal. Appl., 22 (2000), pp. 173–194.
[11] D. A. Flanders, *Angles between flat subspaces of a real n-dimensional Euclidean space*, in Studies and Essays Presented to R. Courant on his 60th Birthday, January 8, 1948, Interscience Publishers, New York, 1948, pp. 129–138.
[12] I. C. Gohberg and M. G. Kreĭn, *Introduction to the Theory of Linear Nonselfadjoint Operators*, Transl. Math. Monogr. 18, AMS, Providence, RI, 1969.
[13] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 3rd ed., Johns Hopkins University Press, Baltimore, MD, 1996.
[14] G. H. Golub and H. Y. Zha, *Perturbation analysis of the canonical correlations of matrix pairs*, Linear Algebra Appl., 210 (1994), pp. 3–28.
[15] G. H. Golub and H. Y. Zha, *The canonical correlations of matrix pairs and their numerical computation*, in Linear Algebra for Signal Processing, Springer-Verlag, New York, 1995, pp. 27–49.
[16] N. J. Higham, *Accuracy and Stability of Numerical Algorithms*, SIAM, Philadelphia, PA, 1996.
[17] H. Hotelling, *Relation between two sets of variables*, Biometrica, 28 (1936), pp. 322–377.
[18] T. Kato, *Perturbation Theory for Linear Operators*, Springer-Verlag, New York, 1976.
[19] A. V. Knyazev, *New estimates for Ritz vectors*, Math. Comp., 66 (1997), pp. 985–995.
[20] A. V. Knyazev, *Preconditioned eigensolvers—an oxymoron?*, Electron. Trans. Numer. Anal., 7 (1998), pp. 104–123.
[21] A. V. Knyazev, *Toward the optimal preconditioned eigensolver: Locally optimal block preconditioned conjugate gradient method*, SIAM J. Sci. Comput., 23 (2001), pp. 517–541.
[22] R.-C. Li, *Relative perturbation theory:* II. *Eigenspace and singular subspace variations*, SIAM J. Matrix Anal. Appl., 20 (1998), pp. 471–492.

[23] C. C. Paige and M. Wei, *History and generality of the CS decomposition*, Linear Algebra Appl., 208/209 (1994), pp. 303–326.
[24] B. N. Parlett, *The Symmetric Eigenvalue Problem*, SIAM, Philadelphia, PA, 1997.
[25] G. W. Stewart, *Computing the CS decomposition of a partitioned orthonormal matrix*, Numer. Math., 40 (1982), pp. 297–306.
[26] G. W. Stewart and J. G. Sun, *Matrix Perturbation Theory*, Academic Press, Boston, MA, 1990.
[27] J. G. Sun, *Perturbation of angles between linear subspaces*, J. Comput. Math., 5 (1987), pp. 58–61.
[28] C. Van Loan, *Computing the CS and the generalized singular value decompositions*, Numer. Math., 46 (1985), pp. 479–491.
[29] P. A. Wedin, *On angles between subspaces of a finite-dimensional inner product space*, in Matrix Pencils, Bo Kågström and Axel Ruhe, eds., Springer-Verlag, Berlin, 1983, pp. 263–285.
[30] H. K. Wimmer, *Canonical angles of unitary spaces and perturbations of direct complements*, Linear Algebra Appl., 287 (1999), pp. 373–379.