

A BACKWARD STABLE ALGORITHM FOR COMPUTING THE CS DECOMPOSITION VIA THE POLAR DECOMPOSITION*

EVAN S. GAWLIK[†], YUJI NAKATSUKASA[‡], AND BRIAN D. SUTTON[§]

Abstract. We introduce a backward stable algorithm for computing the CS decomposition of a partitioned $2n \times n$ matrix with orthonormal columns, or a rank-deficient partial isometry. The algorithm computes two $n \times n$ polar decompositions (which can be carried out in parallel) followed by an eigendecomposition of a judiciously crafted $n \times n$ Hermitian matrix. We prove that the algorithm is backward stable whenever the aforementioned decompositions are computed in a backward stable way. Our algorithm can also be adapted to compute the complete CS decomposition of a square orthogonal or unitary matrix. Since the polar decomposition and the symmetric eigendecomposition are highly amenable to parallelization, the algorithm inherits this feature. We illustrate this fact by invoking recently developed algorithms for the polar decomposition and symmetric eigendecomposition that leverage Zolotarev’s best rational approximations of the sign function. Numerical examples demonstrate that the resulting algorithm for computing the CS decomposition enjoys excellent numerical stability.

Key words. CS decomposition, polar decomposition, eigendecomposition, Zolotarev, generalized singular value decomposition, simultaneous diagonalization, backward stability

AMS subject classifications. 65F30, 15A23, 65F15, 15A18, 65G50

DOI. 10.1137/18M1182747

1. Introduction. The CS decomposition [7, section 2.5.4] allows any partitioned $2n \times n$ matrix

$$A = \begin{pmatrix} A_1 \\ A_2 \end{pmatrix}, \quad A_1, A_2 \in \mathbb{C}^{n \times n},$$

with orthonormal columns to be factorized as

$$A = \begin{pmatrix} U_1 & 0 \\ 0 & U_2 \end{pmatrix} \begin{pmatrix} C \\ S \end{pmatrix} V_1^*,$$

where $U_1, U_2, V_1 \in \mathbb{C}^{n \times n}$ are unitary matrices and $C, S \in \mathbb{R}^{n \times n}$ are diagonal matrices with nonnegative entries satisfying $C^2 + S^2 = I$. In other words, $A_1 = U_1 C V_1^*$ and $A_2 = U_2 S V_1^*$ have highly correlated singular value decompositions: they share the same right singular vectors, and the singular values of A_1 and A_2 are the cosines and sines, respectively, of angles $0 \leq \theta_1 \leq \theta_2 \leq \dots \leq \theta_n \leq \frac{\pi}{2}$. An analogous factorization holds for $(m_1 + m_2) \times n$ ($m_1, m_2 \geq n$) matrices with orthonormal columns [7, section 2.5.4].

By writing

$$(1.1) \quad A_1 = (U_1 V_1^*)(V_1 C V_1^*)$$

and

$$(1.2) \quad A_2 = (U_2 V_1^*)(V_1 S V_1^*),$$

*Received by the editors April 24, 2018; accepted for publication (in revised form) by D. Szyld July 27, 2018; published electronically DATE.

<http://www.siam.org/journals/simax/x-x/M118274.html>

[†]Department of Mathematics, University of California, San Diego, La Jolla, CA 92093 (egawlik@ucsd.edu).

[‡]National Institute of Informatics, 2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo 101-8430, Japan (nakatsukasa@nii.ac.jp).

[§]Department of Mathematics, Randolph-Macon College, Ashland, VA 23005 (bsutton@rmc.edu).

another perspective emerges. Since $W_1 := U_1V_1^*$ and $W_2 := U_2V_1^*$ are unitary and $H_1 := V_1CV_1^*$ and $H_2 := V_1SV_1^*$ are Hermitian positive semidefinite, the *polar decompositions* $A_i = W_iH_i$, $i = 1, 2$, are highly correlated. Specifically, the matrices H_1 and H_2 are simultaneously diagonalizable with eigenvalues $\{\cos \theta_i\}_{i=1}^n$ and $\{\sin \theta_i\}_{i=1}^n$, respectively.

In this paper, we leverage the preceding observation to construct a backward stable algorithm for the CS decomposition. The algorithm computes two polar decompositions $A_1 = W_1H_1$ and $A_2 = W_2H_2$ followed by an eigendecomposition of a judiciously crafted Hermitian matrix $B \in \mathbb{C}^{n \times n}$. As it turns out, the stability of the algorithm depends critically on the choice of B . Obvious candidates, such as H_1 , H_2 , or even $H_1 + H_2$, lead to unstable algorithms. The choice $B = H_2 - H_1$, on the other hand, leads to a backward stable algorithm, assuming that the two polar decompositions and the eigendecomposition are computed in a backward stable way. A central aim of this paper is to prove this assertion.

One of the hallmarks of this approach is its simplicity: it is built entirely from a pair of standard matrix decompositions for which a wealth of highly optimized algorithms are available. In particular, one can compute the CS decomposition in a parallel, communication-efficient way by invoking off-the-shelf algorithms for the polar decomposition and symmetric eigendecomposition. We illustrate this fact by invoking recently developed algorithms that leverage Zolotarev's best rational approximations of the sign function [11].

With a small modification, our algorithm enables the computation of a generalization of the CS decomposition that is applicable to *partial isometries*. Recall that $A \in \mathbb{C}^{m \times n}$ is a partial isometry if $AA^*A = A$; equivalently, every singular value of A is either 1 or 0. We introduce this generalized CS decomposition in section 2 and prove backward stability of the algorithm in this generalized setting. The algorithm also extends easily (see remark (3.viii)) to an algorithm for computing the *complete* 2×2 CS decomposition

$$A = \begin{pmatrix} U_1 & 0 \\ 0 & U_2 \end{pmatrix} \begin{pmatrix} C & -S \\ S & C \end{pmatrix} \begin{pmatrix} V_1 & 0 \\ 0 & V_2 \end{pmatrix}^*$$

of a unitary matrix $A \in \mathbb{C}^{2n \times 2n}$, where $U_1, U_2, V_1, V_2 \in \mathbb{C}^{n \times n}$ are unitary, $C, S \in \mathbb{R}^{n \times n}$ are diagonal with nonnegative entries, and $C^2 + S^2 = I$ [7, section 2.5.4].

Stably computing the CS decomposition of a matrix with orthonormal columns is a notoriously delicate task. The difficulties stem from the fact that the columns of V_1 serve simultaneously as the right singular vectors of A_1 and A_2 . In the presence of roundoff errors, choosing the columns of V_1 to satisfy both roles simultaneously is nontrivial, particularly when $\{\theta_i\}_{i=1}^n$ contains clusters of nearby angles. Early algorithms for the CS decomposition include [16, 21]; both algorithms obtain V_1 by computing an SVD of either A_1 or A_2 and then modifying it. Recent algorithms have focused on simultaneously diagonalizing A_1 and A_2 , using either simultaneous QR iteration or a divide-and-conquer strategy after A_1 and A_2 have been simultaneously reduced to bidiagonal form [17, 18, 19]. There are also general-purpose algorithms for computing the generalized SVD [2], of which the CS decomposition is a special case.

Applications of the CS decomposition are widespread. It can be used to help compute principal angles between subspaces [7, section 6.4.3], the logarithm on the Grassmannian manifold [6], the generalized SVD [7, section 6.1.6], and decompositions of quantum circuits [20]. Good overviews of these and other applications are given in [14, 1].

Organization. This paper is organized as follows. After introducing the generalized CS decomposition for partial isometries in section 2, we detail in section 3 our proposed algorithm for computing it via two polar decompositions and a symmetric eigendecomposition. We prove backward stability of the algorithm in section 4 under mild hypotheses on the eigensolver and the algorithm used to compute the polar decomposition. In section 5, we highlight a specific pair of algorithms—Zolo-pd and Zolo-eig—for computing the polar decomposition and symmetric eigendecomposition. The resulting algorithm for the CS decomposition—Zolo-csd—is tested in section 6 on several numerical examples.

2. Preliminaries. In this section, we introduce a generalization of the CS decomposition that is applicable to partial isometries. We then discuss a few issues concerning partial isometries in finite precision arithmetic.

To begin, recall that every matrix $A \in \mathbb{C}^{m \times n}$ admits a unique *canonical polar decomposition* $A = UH$, where $U \in \mathbb{C}^{m \times n}$ is a partial isometry (that is, $UU^*U = U$), H is Hermitian positive semidefinite, and $\text{range}(U^*) = \text{range}(H)$ [8, Theorem 8.3]. If $A = P\Sigma Q^*$ is the compact SVD of A , i.e., $P \in \mathbb{C}^{m \times r}$ and $Q \in \mathbb{C}^{n \times r}$ have orthonormal columns and $\Sigma \in \mathbb{C}^{r \times r}$ is diagonal with positive diagonal entries (where $r = \text{rank}(A)$), then $U = PQ^*$ and $H = Q\Sigma Q^* = (A^*A)^{1/2}$.

If $m \geq n$, then $A \in \mathbb{C}^{m \times n}$ also admits a *polar decomposition* $A = WH$, where $W \in \mathbb{C}^{m \times n}$ has orthonormal columns and H is Hermitian positive semidefinite. In general, the canonical polar decomposition $A = UH$ differs from the polar decomposition $A = WH$, which is only defined for $m \geq n$. When $m \geq n$, the two H 's coincide, and if A has rank n , then W is uniquely determined, $W = U$, and H is positive definite.

We make use of both decompositions in this paper; the latter is used in the following theorem.

THEOREM 2.1. *Let $A \in \mathbb{C}^{m \times n}$ ($m \geq 2n$) be a partial isometry of rank r . For any partition*

$$A = \begin{pmatrix} A_1 \\ A_2 \end{pmatrix}, \quad A_1 \in \mathbb{C}^{m_1 \times n}, A_2 \in \mathbb{C}^{m_2 \times n}, m_1, m_2 \geq n, m_1 + m_2 = m,$$

there exist $U_1 \in \mathbb{C}^{m_1 \times n}$, $U_2 \in \mathbb{C}^{m_2 \times n}$, and $C, S, V_1 \in \mathbb{C}^{n \times n}$ such that U_1, U_2 , and V_1 have orthonormal columns, C and S are diagonal with nonnegative entries, $C^2 + S^2 = \begin{pmatrix} I_r & 0 \\ 0 & 0_{n-r} \end{pmatrix}$, and

$$(2.1) \quad A = \begin{pmatrix} U_1 & 0 \\ 0 & U_2 \end{pmatrix} \begin{pmatrix} C \\ S \end{pmatrix} V_1^*.$$

Proof. For $i = 1, 2$, let $A_i = W_i H_i$ be a polar decomposition of A_i . Observe that since $A_1 A_1^* A = A_1$,

$$\begin{aligned} A_1 A_2^* A_2 &= A_1 (A^* A - A_1^* A_1) \\ &= A_1 - A_1 A_1^* A_1. \end{aligned}$$

Thus,

$$\begin{aligned} H_1^2 H_2^2 &= (A_1^* A_1)(A_2^* A_2) \\ &= A_1^* (A_1 A_2^* A_2) \\ &= A_1^* A_1 - (A_1^* A_1)^2 \end{aligned}$$

and

$$\begin{aligned} H_2^2 H_1^2 &= (A_2^* A_2)(A_1^* A_1) \\ &= (A_1 A_2^* A_2)^* A_1 \\ &= A_1^* A_1 - (A_1^* A_1)^2. \end{aligned}$$

This shows that H_1^2 and H_2^2 are commuting Hermitian positive semidefinite matrices, so they are simultaneously diagonalizable [7, section 8.7.2]: $H_1^2 = X \Lambda_1 X^*$ and $H_2^2 = X \Lambda_2 X^*$ for some unitary $X \in \mathbb{C}^{n \times n}$ and diagonal $\Lambda_1, \Lambda_2 \in \mathbb{C}^{n \times n}$ with nonnegative entries. Moreover,

$$\begin{aligned} X(\Lambda_1 + \Lambda_2)X^* &= H_1^2 + H_2^2 \\ &= A_1^* A_1 + A_2^* A_2 \\ &= A^* A. \end{aligned}$$

The matrix $\Lambda_1 + \Lambda_2$, being similar to $A^* A$, has r eigenvalues equal to 1 and $n - r$ equal to 0. Since it is diagonal, we may order the columns of X so that

$$\Lambda_1 + \Lambda_2 = \begin{pmatrix} I_r & 0 \\ 0 & 0_{n-r} \end{pmatrix}.$$

The theorem follows from taking $U_1 = W_1 X$, $U_2 = W_2 X$, $C = \Lambda_1^{1/2}$, $S = \Lambda_2^{1/2}$, and $V_1 = X$. \square

Note that (2.1) also proves the existence of an ‘‘economical’’ rank-deficient CS decomposition

$$(2.2) \quad A = \begin{pmatrix} U_{1r} & 0 \\ 0 & U_{2r} \end{pmatrix} \begin{pmatrix} C_r \\ S_r \end{pmatrix} V_{1r}^*,$$

where the subscript r indicates the submatrices consisting of the leading r columns (and rows for C_r, S_r): $U_{1r} \in \mathbb{C}^{m_1 \times r}$, $U_{2r} \in \mathbb{C}^{m_2 \times r}$, $C_r, S_r \in \mathbb{C}^{r \times r}$, and $V_{1r} \in \mathbb{C}^{n \times r}$.

2.1. Approximate partial isometries. In finite precision arithmetic, we will be interested in computing the CS decomposition of matrices that are approximate partial isometries, in the sense that $\|AA^*A - A\|$ is small. The next pair of lemmas show that if A is a matrix for which $\|AA^*A - A\|$ is small, then A is close to a partial isometry U whose rank coincides with the numerical rank of A .

We begin with a few definitions. For a given unitarily invariant norm $\|\cdot\|$ and a given number $\varepsilon > 0$, we define the ε -rank of a matrix $A \in \mathbb{C}^{m \times n}$ to be

$$(2.3) \quad \text{rank}_\varepsilon(A) = \min_{\substack{B \in \mathbb{C}^{m \times n} \\ \|A-B\| \leq \varepsilon}} \text{rank}(B).$$

Note that if $A = \sum_{k=1}^{\min(m,n)} \sigma_k u_k v_k^*$ is the singular value decomposition of A in summation form, then the minimizer of (2.3) is given by

$$(2.4) \quad A_r = \sum_{k=1}^r \sigma_k u_k v_k^*,$$

where $r = \text{rank}_\varepsilon(A)$.

Let

$$(2.5) \quad d(A) = \min_{\substack{U \in \mathbb{C}^{m \times n} \\ UU^*U = U}} \|A - U\|.$$

It can be shown [9] in the spectral norm that

$$(2.6) \quad d(A) = \max_{1 \leq i \leq \min(m, n)} \min(\sigma_i(A), |1 - \sigma_i(A)|),$$

and this minimum is achieved by the factor U in the canonical polar decomposition $A_r = UH$, where A_r is given by (2.4) and r is the largest integer such that $\sigma_r \geq 1/2$.

Throughout this paper, we make use of the fact that in any unitarily invariant norm,

$$(2.7) \quad \|ABC\| \leq \min\{\sigma_1(A)\sigma_1(B)\|C\|, \sigma_1(A)\|B\|\sigma_1(C), \|A\|\sigma_1(B)\sigma_1(C)\}$$

for any matrices A, B, C whose product ABC is defined [8, equation (B.7)].

The following lemma extends [8, Lemma 8.17] to approximate partial isometries having exact rank r .

LEMMA 2.2. *Let $A \in \mathbb{C}^{m \times n}$ have canonical polar decomposition $A = UH$. If $\text{rank}(A) = r$, then*

$$(2.8) \quad \frac{\|AA^*A - A\|}{\sigma_1(A)(1 + \sigma_1(A))} \leq \|A - U\| \leq \frac{\|AA^*A - A\|}{\sigma_r(A)(1 + \sigma_r(A))}$$

in any unitarily invariant norm.

Proof. Let $A = P\Sigma Q^*$ be the compact singular value decomposition of A with $\Sigma \in \mathbb{R}^{r \times r}$, so that $U = PQ^*$. Then

$$\begin{aligned} \|AA^*A - A\| &= \|\Sigma^3 - \Sigma\| \\ &= \|\Sigma(\Sigma - I)(\Sigma + I)\| \\ &\leq \sigma_1(A)\|\Sigma - I\|(\sigma_1(A) + 1) \\ &= \sigma_1(A)\|A - U\|(\sigma_1(A) + 1). \end{aligned}$$

On the other hand,

$$\begin{aligned} \|A - U\| &= \|\Sigma - I\| \\ &= \|\Sigma^{-1}(\Sigma^3 - \Sigma)(\Sigma + I)^{-1}\| \\ &\leq \sigma_1(\Sigma^{-1})\|\Sigma^3 - \Sigma\|\sigma_1((\Sigma + I)^{-1}) \\ &= \frac{1}{\sigma_r(A)}\|AA^*A - A\| \frac{1}{\sigma_r(A) + 1}. \quad \square \end{aligned}$$

The next lemma handles the setting in which $\|AA^*A - A\|$ is small and A has ε -rank r rather than exact rank r .

LEMMA 2.3. *Let $A \in \mathbb{C}^{m \times n}$ have ε -rank r with respect to a unitarily invariant norm $\|\cdot\|$. Then there exists a partial isometry $U \in \mathbb{C}^{m \times n}$ of rank r satisfying*

$$(2.9) \quad \|A - U\| \leq \varepsilon + \frac{\|AA^*A - A\| + \varepsilon(1 + 3\sigma_1(A)^2)}{\sigma_r(A)(1 + \sigma_r(A))}.$$

Proof. Let A_r be as in (2.4), and let $A_r = UH$ be the canonical polar decomposition of A_r . By Lemma 2.2,

$$\|A - U\| \leq \|A - A_r\| + \frac{\|A_r A_r^* A_r - A_r\|}{\sigma_r(A_r)(1 + \sigma_r(A_r))}.$$

Now since $\sigma_1(A_r) = \sigma_1(A)$, the inequality (2.7) implies

$$\begin{aligned} \|A_r A_r^* A_r - A_r\| &\leq \|A A^* A - A\| + \|A_r - A\| + \|A_r A_r^* A_r - A A^* A\| \\ &\leq \|A A^* A - A\| + \|A_r - A\| + \|(A_r - A) A_r^* A_r\| + \|A(A_r - A)^* A_r\| \\ &\quad + \|A A^*(A_r - A)\| \\ &\leq \|A A^* A - A\| + (1 + 3\sigma_1(A)^2) \|A_r - A\|. \end{aligned}$$

The result then follows from the relations $\sigma_r(A_r) = \sigma_r(A)$ and $\|A - A_r\| \leq \varepsilon$. \square

Note that Lemmas 2.2 and 2.3 can be used to estimate $d(A)$ in (2.5).

3. Algorithm. In this section, we detail a general algorithm for computing the CS decomposition via two polar decompositions and a symmetric eigendecomposition. We focus on the motivation behind the algorithm first, postponing an analysis of its stability to section 4.

The observations made in section 1 immediately suggest the following general strategy for computing the CS decomposition of a matrix $A = \begin{pmatrix} A_1 \\ A_2 \end{pmatrix}$ with $A_1, A_2 \in \mathbb{C}^{n \times n}$ and $A^* A = I$. One can compute polar decompositions $A_1 = W_1 H_1$ and $A_2 = W_2 H_2$, diagonalize H_1 and/or H_2 to obtain V_1, C and S such that $H_1 = V_1 C V_1^*$ and $H_2 = V_1 S V_1^*$, and set $U_1 = W_1 V_1, U_2 = W_2 V_1$. For such an approach to be viable, it is critical that V_1 be computed in a stable way. The following MATLAB example illustrates the pitfalls of a naive strategy: diagonalizing H_1 .

```
theta = [1e-8 2e-8 3e-8];
C = diag(cos(theta));
S = diag(sin(theta));
V1 = [2 -1 2; 2 2 -1; 1 -2 -2]/3;
H1 = V1*C*V1';
H2 = V1*S*V1';
[V1,C] = eig(H1);
S = V1'*H2*V1;
max(max(abs(S-diag(diag(S)))))
ans = 2.8187e-09
```

Here, the computed V_1 (which we will denote by \widehat{V}_1) provides a poor approximation of the eigenvectors of H_2 , as evidenced by the size of the off-diagonal entry of $\widehat{V}_1^* H_2 \widehat{V}_1$ with the largest absolute value. Ideally, in double-precision arithmetic, the latter quantity should be a small multiple of the unit roundoff $u = 2^{-53} \approx 10^{-16}$.

One possible remedy is to perform simultaneous diagonalization [4] to obtain V_1 from H_1 and H_2 . In this paper, we consider a different approach that exploits the special structure of H_1 and H_2 . The idea is to obtain V_1 by computing the eigendecomposition of $H_2 - H_1$, whose eigenvectors are the same as those of H_1 and H_2 . The advantages of this approach, though not obvious at first glance, are easily illustrated. In the MATLAB example above, replacing `eig(H1)` with `eig(H2-H1)` yields `ans = 1.7806e-17`.

To give more insight, we explain what goes wrong if we obtain V_1 via an eigendecomposition of H_1 . Since $\cos \theta_i \approx 1 - \frac{1}{2}\theta_i^2$ for small θ_i , we have $|\cos \theta_i - \cos \theta_j| \approx$

$\frac{1}{2}(\theta_i + \theta_j)|\theta_i - \theta_j| \ll |\theta_i - \theta_j|$ for small θ_i and θ_j , rendering the eigenvectors of H_1 very sensitive to perturbation if two or more angles are close to zero. A standard eigendecomposition algorithm for H_1 still gives a backward stable decomposition: $\|\widehat{V}_1^* H_1 \widehat{V}_1 - \widehat{\Lambda}_1\| \leq \epsilon \|H_1\|$, where $\widehat{\Lambda}_1$ is diagonal and ϵ is a small multiple of u . However, inaccuracies in the columns of \widehat{V}_1 manifest themselves when we use the same \widehat{V}_1 to compute the eigendecomposition of H_2 : For a computed eigenpair $(\widehat{\lambda}_i, \widehat{v}_i)$ with $\|\widehat{v}_i\|_2 = 1$ obtained in a backward stable manner, we have $H_1 \widehat{v}_i = \widehat{\lambda}_i \widehat{v}_i + \epsilon$, where $\|\epsilon\| = O(u)$, where u is the unit roundoff. Expanding $\widehat{v}_i = \sum_{j=1}^n c_j v_j$, where v_j are the exact eigenvectors of H_1 , we have $|v_j^* \epsilon| = |c_j| |\widehat{\lambda}_i - \lambda_j|$. Now, the same \widehat{v}_i taken as an approximate eigenvector of H_2 gives $H_2 \widehat{v}_i = \widehat{\lambda}_{i,2} \widehat{v}_i + \epsilon_2$, where the choice $\widehat{\lambda}_{i,2} = \widehat{v}_i^* H_2 \widehat{v}_i$ minimizes $\|\epsilon_2\|$. We then have $|v_j^* \epsilon_2| = |\widehat{\lambda}_{i,2} - \lambda_{j,2}| |c_j|$ for each j . Using the above relation $|c_j| = \frac{|v_j^* \epsilon|}{|\widehat{\lambda}_i - \lambda_j|}$, we see that $|v_j^* \epsilon_2| = |v_j^* \epsilon| \frac{|\widehat{\lambda}_{i,2} - \lambda_{j,2}|}{|\widehat{\lambda}_i - \lambda_j|}$ for each j . The crucial observation is that for each j , the v_j -component of ϵ is magnified by the factor $\frac{|\widehat{\lambda}_{i,2} - \lambda_{j,2}|}{|\widehat{\lambda}_i - \lambda_j|}$, the ratio in the eigenvalue gap. In the above setting, $\lambda_i = \cos \theta_i$ and $\lambda_{i,2} = \sin \theta_i$, and since the eigenvalues have $O(\|\epsilon\|^2)$ accuracy [15], no essence is lost in taking $\widehat{\lambda}_i = \lambda_i$ and $\widehat{\lambda}_{i,2} = \lambda_{i,2}$. Thus, since $|\sin \theta_i - \sin \theta_j| \approx |\theta_i - \theta_j|$ and $|\cos \theta_i - \cos \theta_j| \approx \frac{1}{2}(\theta_i + \theta_j)|\theta_i - \theta_j|$ for small θ_i and θ_j , the relative gap $|\sin \theta_i - \sin \theta_j| / |\cos \theta_i - \cos \theta_j| \approx 2/(\theta_i + \theta_j)$ can be arbitrarily large, in which case \widehat{V}_2 does not give a backward stable eigendecomposition for H_2 : $\|\widehat{V}_1^* H_2 \widehat{V}_1 - \widehat{\Lambda}_2\| \gg u \|H_2\|$.

A similar problem occurs if V_1 is obtained via an eigendecomposition of H_2 . If two or more angles are close to $\frac{\pi}{2}$, their sines are closely spaced, rendering the eigenvectors of H_2 very sensitive to perturbation. In this scenario, numerical experiments show that $\widehat{V}_1^* H_1 \widehat{V}_1$ can have off-diagonal entries with unacceptably large magnitude. The essence of the problem is that for $\theta_i \neq \theta_j$ near $\frac{\pi}{2}$, the ratio $|\cos \theta_i - \cos \theta_j| / |\sin \theta_i - \sin \theta_j| \approx 2/(\pi - \theta_i - \theta_j)$ can be arbitrarily large.

Obtaining V_1 via an eigendecomposition of $H_2 - H_1$ sidesteps these difficulties for the following reason. The function $g(\theta) = \sin \theta - \cos \theta$ has derivative $g'(\theta) \geq 1$ on $[0, \frac{\pi}{2}]$, from which it is easy to show that $|\cos \theta_i - \cos \theta_j| / |g(\theta_i) - g(\theta_j)| \leq 1$ and $|\sin \theta_i - \sin \theta_j| / |g(\theta_i) - g(\theta_j)| \leq 1$ for every $\theta_i, \theta_j \in [0, \frac{\pi}{2}]$ with $\theta_i \neq \theta_j$. In other words, the eigenvalues of H_1 and H_2 are spaced no further apart than the eigenvalues of $H_2 - H_1$. As a result, the arguments in the preceding paragraphs suggest that the numerically computed eigenvectors of $H_2 - H_1$ likely provide a backward stable approximation of the eigenvectors of both H_1 and H_2 . As an aside, note that another seemingly natural alternative—computing the eigendecomposition of $H_1 + H_2$ —is not viable since the derivative of $\cos \theta + \sin \theta$ vanishes at $\theta = \frac{\pi}{4}$.

Extension to partial isometries. With one caveat, all of the arguments in the preceding paragraph carry over to the more general setting in which $A = \begin{pmatrix} A_1 \\ A_2 \end{pmatrix}$ is a partial isometry with $A_1 \in \mathbb{C}^{m_1 \times n}$, $A_2 \in \mathbb{C}^{m_2 \times n}$, and $m_1, m_2 \geq n$. The caveat is that if A is rank-deficient and has principal angle(s) θ_i equal to $\pi/4$, then it may be impossible to distinguish between two eigenspaces of H_1 and H_2 : the eigenspace \mathcal{V}_0 corresponding to the eigenvalue 0, and the eigenspace $\mathcal{V}_{1/\sqrt{2}}$ corresponding to the eigenvalue $\cos(\pi/4) = \sin(\pi/4) = 1/\sqrt{2}$. Indeed, both of these eigenspaces correspond to the zero eigenvalue of $H_2 - H_1$. Even if $\theta_i \neq \pi/4$ for every i , numerical instabilities can still arise if any angle θ_i is close to $\pi/4$.

Fortunately, there is a simple remedy to this problem. When A is rank-deficient, then instead of computing the eigendecomposition of $H_2 - H_1$, one can compute the eigendecomposition of $B = H_2 - H_1 + \mu(I - A^*A)$, where $\mu > 1$ is a scalar. This has

the effect of shifting the eigenvalue corresponding to \mathcal{V}_0 away from all of the other eigenvalues of B . Indeed, if $H_1 = V_1 \begin{pmatrix} C_r & 0 \\ 0 & 0_{n-r} \end{pmatrix} V_1^*$ and $H_2 = V_1 \begin{pmatrix} S_r & 0 \\ 0 & 0_{n-r} \end{pmatrix} V_1^*$, then

$$B = V_1 \begin{pmatrix} S_r - C_r & 0 \\ 0 & \mu I_{n-r} \end{pmatrix} V_1^*,$$

and the diagonal entries of $S_r - C_r$ lie in the interval $[-1, 1] \not\ni \mu$.

Algorithm summary. The algorithm that results from these considerations is summarized below. In what follows, we use diag to denote (as in MATLAB) the operator that, if applied to a matrix $X \in \mathbb{C}^{n \times n}$, returns a vector $x \in \mathbb{C}^n$ with $x_i = X_{ii}$, and, if applied to a vector $x \in \mathbb{C}^n$, returns $X \in \mathbb{C}^{n \times n}$ with $X_{ii} = x_i$ and $X_{ij} = 0$ for $i \neq j$.

Algorithm 3.1 CS decomposition of a partial isometry $A = \begin{pmatrix} A_1 \\ A_2 \end{pmatrix}$, $A_1 \in \mathbb{C}^{m_1 \times n}$, $A_2 \in \mathbb{C}^{m_2 \times n}$, $m_1, m_2 \geq n$.

- 1: $W_1 H_1 = A_1$ (polar decomposition)
 - 2: $W_2 H_2 = A_2$ (polar decomposition)
 - 3: **if** $\text{rank}(A) = n$ **then** $\mu = 0$ **else** $\mu = 2$ **end if**
 - 4: $B = H_2 - H_1 + \mu(I - A^* A)$
 - 5: $V_1 \Lambda V_1^* = B$ (symmetric eigendecomposition)
 - 6: $U_1 = W_1 V_1$
 - 7: $U_2 = W_2 V_1$
 - 8: $C = \text{diag}(\text{diag}(V_1^* H_1 V_1))$
 - 9: $S = \text{diag}(\text{diag}(V_1^* H_2 V_1))$
 - 10: **return** U_1, U_2, C, S, V_1
-

We conclude this section with a few remarks.

- (3.i) The algorithm treats A_1 and A_2 in a symmetric way, in the sense that when $\mu = 0$, exchanging the roles of A_1 and A_2 merely negates B and Λ (thereby sending $\theta_i = \arctan(S_{ii}/C_{ii})$ to $\pi/2 - \theta_i$ for each i).
- (3.ii) Lines 1–2 and lines 6–9 can each be carried out in parallel. Furthermore, if $A^* A$ is needed in line 4, then it can be computed in parallel with lines 1–2.
- (3.iii) As a postprocessing step, one can compute $\theta_i = \arctan(S_{ii}/C_{ii})$, $i = 1, 2, \dots, n$, and overwrite C and S with $\text{diag}(\cos \theta)$ and $\text{diag}(\sin \theta)$, respectively (with the obvious modifications for rank-deficient A). It is not hard to verify that this has the effect of reducing $\|C^2 + S^2 - \begin{pmatrix} I_r & 0 \\ 0 & 0_{n-r} \end{pmatrix}\|$ without disrupting the backward stability of the algorithm.
- (3.iv) An alternative, cheaper way to compute C and S in lines 8–9 is to solve the equation $\sin \theta_i - \cos \theta_i = \Lambda_{ii}$ for θ_i , $i = 1, 2, \dots, n$, and set $C = \text{diag}(\cos \theta)$, $S = \text{diag}(\sin \theta)$ (with the obvious modifications for rank-deficient A). Our numerical experiments suggest that this approach is generally less accurate than lines 8–9, but it still renders the algorithm backward stable. Our analysis will focus on the use of lines 8–9 to obtain C and S , but it can be easily modified to treat the alternative approach.
- (3.v) Let us examine the arithmetic cost in flop counts. The steps that require $O(n^3)$ flops are two polar decompositions (lines 1–2) and a symmetric eigendecomposition (line 5), in addition to matrix-product operations whose flop counts are clear: $A^* A$ (line 4, costing $(m_1 + m_2)n^2$ flops exploiting symmetry) and U_1, U_2 (lines 6–7, $2m_i n^2$ flops for each $i = 1, 2$), and the diagonal elements of C and S (lines 8–9, $2n^3$ flops each). The costs of the polar and eigenvalue decompositions depend on the algorithm used. When Zolopd and Zolo-eig are used, they are $64m_i n^2 + \frac{8}{3}n^3$ flops for each $i = 1, 2$

($8 \max\{m_1, m_2\}n^2 + \frac{1}{3}n^3$ along the critical path) for Zolo-pd and about $55n^3$ flops ($16n^3$) for Zolo-eig [11, Table 5.1,5.2]. Zolo-csd thus requires a total of about $67(m_1+m_2)n^2+64n^3$ flops ($10 \max\{m_1, m_2\}n^2+16n^3$ along the critical path). Clearly, the polar and eigenvalue decompositions form the majority of the computation. When a classical algorithm is used for these decompositions (via the SVD for polar, costing $8m_i n^2+20n^3$ and $9n^3$ for the eigendecomposition), the overall cost is $11(m_1+m_2)n^2+53n^3$ flops ($10 \max\{m_1, m_2\}n^2+29n^3$ along the critical path). It is worth noting that these flop counts usually do not accurately reflect the actual running time, particularly in a massively parallel computing environment; they are presented here for reference purposes.

- (3.vi) In applications, we expect that users will know in advance whether A is full-rank or not. In the rare situation in which it is not known until runtime, an inexpensive approach is to compute $\|A\|_F$, noting that assuming $d(A) \ll 1$, we have $\|A\|_F \approx \sqrt{\text{rank}(A)}$. Another alternative is to perform a rank-revealing QR factorization (for instance) in line 3. Needless to say, one should measure the ε -rank of A for a suitable tolerance $\varepsilon > 0$, not the exact rank of A .
- (3.vii) In the rank-deficient case $r < n$, an economical CS decomposition (2.2) can be obtained by a simple modification as follows: After line 5, we extract the eigenvalues $\Lambda_{ii} \in [-1, 1]$ (there should be r of them) and their corresponding eigenvectors $V_{1r} \in \mathbb{C}^{n \times r}$. (The remaining $n - r$ columns of V_1 are the null vectors of A .) We have $H_2 - H_1 = V_{1r}\Lambda_r V_{1r}^*$, where $\Lambda_r \in \mathbb{C}^{r \times r}$ is diagonal with the r eigenvalues of Λ lying in $[-1, 1]$ on its diagonal (assuming A is an exact partial isometry). Finally, we let $U_i := W_i V_{1r} \in \mathbb{C}^{m_i \times r}$, and $C = \text{diag}(\text{diag}(V_{1r}^* H_1 V_{1r})) \in \mathbb{C}^{r \times r}$, $S = \text{diag}(\text{diag}(V_{1r}^* H_2 V_{1r})) \in \mathbb{C}^{r \times r}$ to obtain the rank-deficient CS decomposition $A_1 = U_1 C V_{1r}^*$, $A_2 = U_2 S V_{1r}^*$. (We output $V_1 := V_{1r}$.)
- (3.viii) It is well known that any unitary $A = \begin{pmatrix} A_1 & A_3 \\ A_2 & A_4 \end{pmatrix} \in \mathbb{C}^{2n \times 2n}$ with $A_i \in \mathbb{C}^{n \times n}$, $i = 1, 2, 3, 4$, admits a *complete* 2×2 CS decomposition

$$A = \begin{pmatrix} U_1 & 0 \\ 0 & U_2 \end{pmatrix} \begin{pmatrix} C & -S \\ S & C \end{pmatrix} \begin{pmatrix} V_1 & 0 \\ 0 & V_2 \end{pmatrix}^*$$

where $U_1, U_2, V_1, V_2 \in \mathbb{C}^{n \times n}$ are unitary, $C, S \in \mathbb{R}^{n \times n}$ are diagonal with nonnegative entries, and $C^2 + S^2 = I$ [7, section 2.5.4]. Algorithm 3.1 applied to $\begin{pmatrix} A_1 \\ A_2 \end{pmatrix}$ computes all of these matrices except V_2 . If V_2 is desired, we advocate using the following strategy from [19, section 4.11]: compute $X = -A_3^* U_1 S + A_4^* U_2 C$ and its QR decomposition $X = QR$, and set $V_2 = Q$. (The QR decomposition may be unnecessary, because if A is exactly unitary, then so is X .) An argument similar to the proof of [19, Theorem 18] shows that this algorithm for the 2×2 CS decomposition is backward stable if Algorithm 3.1 is backward stable.

- (3.ix) More generally, the CS decomposition of an arbitrarily partitioned unitary matrix $A = \begin{pmatrix} A_1 & A_3 \\ A_2 & A_4 \end{pmatrix} \in \mathbb{C}^{m \times m}$ with $A_1 \in \mathbb{C}^{m_1 \times n_1}$, $A_2 \in \mathbb{C}^{m_2 \times n_1}$, $A_3 \in \mathbb{C}^{m_1 \times n_2}$, and $A_4 \in \mathbb{C}^{m_2 \times n_2}$, (assuming $m_1 \geq n_1$ and $m_1 \geq m_2$ for simplicity) is [7, Theorem 2.5.3]

$$(3.1) \quad \begin{pmatrix} U_1 & 0 \\ 0 & U_2 \end{pmatrix}^* A \begin{pmatrix} V_1 & 0 \\ 0 & V_2 \end{pmatrix} = \left(\begin{array}{cc|ccc} I_p & 0 & 0 & 0 & 0 \\ 0 & C & -S & 0 & 0 \\ 0 & 0 & 0 & I_{m_1-n_1} & 0 \\ \hline 0 & S & C & 0 & 0 \\ 0 & 0 & 0 & 0 & I_q \end{array} \right),$$

where $p = \max(0, n_1 - m_2)$ and $q = \max(0, m_2 - n_1)$, and $C, S \in \mathbb{R}^{(n_1-p) \times (n_1-p)}$, $U_1 \in \mathbb{C}^{m_1 \times m_1}$, $V_1 \in \mathbb{C}^{n_1 \times n_1}$. By transposing A and/or permuting its columns, we may assume without loss of generality that $n_1 = \min\{n_1, n_2, m_1, m_2\}$, so that $p = 0$ and $q = m_2 - n_1$. To compute the CS decomposition under this assumption, we first proceed exactly as in remark (3.viii). We use Algorithm 3.1 to compute $U_1 \in \mathbb{C}^{m_1 \times n_1}$, $U_2 \in \mathbb{C}^{m_2 \times n_1}$, and $C, S, V_1 \in \mathbb{C}^{n_1 \times n_1}$ satisfying $A_1 = U_1 C V_1^*$ and $A_2 = U_2 S V_1^*$, and we set $V_2 \in \mathbb{C}^{n_2 \times n_1}$ equal to the factor Q in the (thin) QR decomposition $X = QR$ of $X = -A_3^* U_1 S + A_4^* U_2 C$. We then append orthonormal columns arbitrarily to U_1 , U_2 , and V_2 until they are square ($U_1 \in \mathbb{C}^{m_1 \times m_1}$, $U_2 \in \mathbb{C}^{m_2 \times m_2}$, and $V_2 \in \mathbb{C}^{n_2 \times n_2}$) and unitary. In exact arithmetic,

$$(3.2) \quad \begin{pmatrix} U_1 & 0 \\ 0 & U_2 \end{pmatrix}^* A \begin{pmatrix} V_1 & 0 \\ 0 & V_2 \end{pmatrix} = \left(\begin{array}{c|cc} C & -S & 0 \\ 0 & 0 & Y_1 \\ \hline S & C & 0 \\ 0 & 0 & Y_2 \end{array} \right),$$

where $Y_1 \in \mathbb{C}^{(m_1-n_1) \times (m-2n_1)}$, $Y_2 \in \mathbb{C}^{(m_2-n_1) \times (m-2n_1)}$, and $Y = \begin{pmatrix} Y_1 \\ Y_2 \end{pmatrix}$ is unitary. Overwriting V_2 with $V_2 \begin{pmatrix} I_{n_1} & 0 \\ 0 & Y^* \end{pmatrix}$ yields the decomposition (3.1). (Recall that $p = 0$.) A slight improvement of this algorithm is achieved if one appends the correct columns to V_2 at the outset, rather than choosing them arbitrarily and overwriting them. A short calculation verifies that appending $(A_3^* U_{1\perp} \ A_4^* U_{2\perp})$ accomplishes this goal, where $U_{1\perp} \in \mathbb{C}^{m_1 \times (m_1-n_1)}$ and $U_{2\perp} \in \mathbb{C}^{m_2 \times (m_2-n_1)}$ consist of the columns appended to U_1 and U_2 , respectively.

4. Backward stability. In this section, we prove that Algorithm 3.1 is backward stable, provided that the polar decompositions in lines 1–2 and the eigendecomposition in line 5 are computed in a backward stable way.

Throughout this section, we continue to use $\|\cdot\|$ to denote any unitarily invariant norm. We denote by c_n the absolute condition number of the map \mathcal{H} sending a matrix $A \in \mathbb{C}^{m \times n}$ ($m \geq n$) to the Hermitian positive semidefinite factor $H = \mathcal{H}(A)$ in the polar decomposition $A = WH$; that is,

$$(4.1) \quad c_n = \max_{\substack{A, \Delta A \in \mathbb{C}^{m \times n} \\ \Delta A \neq 0}} \frac{\|\mathcal{H}(A + \Delta A) - \mathcal{H}(A)\|}{\|\Delta A\|}.$$

It is easy to check that this number depends on n and $\|\cdot\|$, but not m . In the Frobenius norm, $c_n = \sqrt{2}$ is constant [8, Theorem 8.9]. In the 2-norm, it is known that $b_n \leq c_n \leq 1 + 2b_n$, where $b_n \sim \frac{2}{\pi} \log n$ [10, Corollary 4.3]. We will also make use of the fact that

$$\|\text{diag}(\text{diag}(A))\| \leq \|A\|$$

in any unitarily invariant norm [3, p. 152].

THEOREM 4.1. *Let $A = \begin{pmatrix} A_1 \\ A_2 \end{pmatrix} \in \mathbb{C}^{m \times n}$ with $A_1 \in \mathbb{C}^{m_1 \times n}$, $A_2 \in \mathbb{C}^{m_2 \times n}$, $m_1, m_2 \geq n$, and $m = m_1 + m_2$. Suppose that Algorithm 3.1 computes the following quantities*

with indicated errors:

$$(4.2) \quad \widehat{W}_1 \widehat{H}_1 = A_1 + \Delta A_1,$$

$$(4.3) \quad \widehat{W}_2 \widehat{H}_2 = A_2 + \Delta A_2,$$

$$(4.4) \quad \widehat{B} = \widehat{H}_2 - \widehat{H}_1 + \mu(I - A^* A) + \Delta B_1,$$

$$(4.5) \quad \widehat{V}_1 \widehat{\Lambda} \widehat{V}_1^* = \widehat{B} + \Delta B_2,$$

$$(4.6) \quad \widehat{C} = \text{diag}(\text{diag}(\widehat{V}_1^* \widehat{H}_1 \widehat{V}_1)) + \Delta C,$$

$$(4.7) \quad \widehat{S} = \text{diag}(\text{diag}(\widehat{V}_1^* \widehat{H}_2 \widehat{V}_1)) + \Delta S,$$

where $\widehat{\Lambda}$, \widehat{C} , and \widehat{S} are real and diagonal, \widehat{H}_1 and \widehat{H}_2 are Hermitian, and $\mu \geq 0$. Assume that $\|\widehat{W}_1^* \widehat{W}_1 - I\|$, $\|\widehat{W}_2^* \widehat{W}_2 - I\|$, $\|\widehat{V}_1^* \widehat{V}_1 - I\|$, $\min_{G=G^* \geq 0} \|\widehat{H}_1 - G\|$, and $\min_{G=G^* \geq 0} \|\widehat{H}_2 - G\|$ are each bounded above by a number δ . If (2.5) has no minimizer of rank n , assume further that $\mu > 1$. Then

$$(4.8) \quad \begin{aligned} \|\widehat{W}_1 \widehat{V}_1 \widehat{C} \widehat{V}_1^* - A_1\| &\leq (4c_n + 1)\|\Delta A_1\| + 2c_n\|\Delta A_2\| + 2\|\Delta B_1\| + 2\|\Delta B_2\| + \|\Delta C\| \\ &\quad + (9c_n + 10 + 2 \max\{\mu, 1\})\delta + (6c_n + 4\mu)d(A) + o(\eta), \end{aligned}$$

$$(4.9) \quad \begin{aligned} \|\widehat{W}_2 \widehat{V}_1 \widehat{S} \widehat{V}_1^* - A_2\| &\leq 2c_n\|\Delta A_1\| + (4c_n + 1)\|\Delta A_2\| + 2\|\Delta B_1\| + 2\|\Delta B_2\| + \|\Delta S\| \\ &\quad + (9c_n + 10 + 2 \max\{\mu, 1\})\delta + (6c_n + 4\mu)d(A) + o(\eta), \end{aligned}$$

asymptotically as

$$(4.10) \quad \eta := \max\{\delta, d(A), \|\Delta A_1\|, \|\Delta A_2\|, \|\Delta B_1\|, \|\Delta B_2\|, \|\Delta C\|, \|\Delta S\|\} \rightarrow 0.$$

Before proving the theorem, we make a few remarks. First, the smallness of the quantities $\|\Delta A_i\|$, $\|\widehat{W}_i^* \widehat{W}_i - I\|$, and $\min_{G=G^* \geq 0} \|\widehat{H}_i - G\|$ is equivalent to the condition that the polar decompositions $A_i \approx \widehat{W}_i \widehat{H}_i$, $i = 1, 2$, are computed in a backward stable way [12]. Second, the smallness of $\|\Delta B_2\|$ and $\|\widehat{V}_1^* \widehat{V}_1 - I\|$ corresponds to the condition that the eigendecomposition of \widehat{B} is computed in a backward stable way. The smallness of $\|\Delta B_1\|$, $\|\Delta C\|$, and $\|\Delta S\|$ is automatic in floating point arithmetic. We also note that A is not assumed to be exactly a partial isometry; its deviation is measured by $d(A)$. Theorem 4.1 thus says that Algorithm 3.1 is backward stable whenever $d(A)$ is small and the polar decompositions in lines 1–2 and the symmetric eigendecomposition in line 5 are computed in a backward stable way. We give examples of backward stable algorithms for the polar decomposition and symmetric eigendecomposition in section 5.

The estimates (4.8)–(4.9) have been written in full detail to make clear the contribution of each source of error. Coarser estimates of a more memorable form are easy to write down. Consider, for example, the setting in which A has nearly orthonormal columns, so that μ can be taken equal to zero. Then (4.8)–(4.9) imply that

$$(4.11) \quad \|\widehat{W}_1 \widehat{V}_1 \widehat{C} \widehat{V}_1^* - A_1\|_2 \lesssim \left(39 + \frac{84}{\pi} \log n\right) \eta + o(\eta),$$

$$(4.12) \quad \|\widehat{W}_2 \widehat{V}_1 \widehat{S} \widehat{V}_1^* - A_2\|_2 \lesssim \left(39 + \frac{84}{\pi} \log n\right) \eta + o(\eta),$$

in the 2-norm, and

$$(4.13) \quad \|\widehat{W}_1 \widehat{V}_1 \widehat{C} \widehat{V}_1^* - A_1\|_F \leq (18 + 21\sqrt{2})\eta + o(\eta),$$

$$(4.14) \quad \|\widehat{W}_2 \widehat{V}_1 \widehat{S} \widehat{V}_1^* - A_2\|_F \leq (18 + 21\sqrt{2})\eta + o(\eta),$$

in the Frobenius norm, where η is given by (4.10). (Our numerical experiments suggest that these are pessimistic estimates.)

Proof of Theorem 4.1. To prove the theorem, let $\tilde{A} \in \mathbb{C}^{m \times n}$ be a partial isometry of maximal rank such that $\|A - \tilde{A}\| = d(A)$. Let $r = \text{rank}(\tilde{A})$, and let $\tilde{A}_1 \in \mathbb{C}^{m_1 \times n}$ and $\tilde{A}_2 \in \mathbb{C}^{m_2 \times n}$ be such that $\tilde{A} = \begin{pmatrix} \tilde{A}_1 \\ \tilde{A}_2 \end{pmatrix}$. Let

$$\begin{pmatrix} \tilde{A}_1 \\ \tilde{A}_2 \end{pmatrix} = \begin{pmatrix} U_1 & 0 \\ 0 & U_2 \end{pmatrix} \begin{pmatrix} C \\ S \end{pmatrix} V_1^*$$

be a CS decomposition of \tilde{A} . Let $0 \leq \theta_1 \leq \theta_2 \leq \dots \leq \theta_r \leq \frac{\pi}{2}$ be the corresponding angles such that $C = \begin{pmatrix} C_r & 0 \\ 0 & 0_{n-r} \end{pmatrix}$ and $S = \begin{pmatrix} S_r & 0 \\ 0 & 0_{n-r} \end{pmatrix}$ with $C_r = \text{diag}(\cos \theta)$ and $S_r = \text{diag}(\sin \theta)$. Define $W_1 = U_1 V_1^*$, $H_1 = V_1 C V_1^*$, $W_2 = U_2 V_1^*$, and $H_2 = V_1 S V_1^*$, so that $W_i H_i$ is a polar decomposition of \tilde{A}_i for each i . \square

LEMMA 4.2. *Let $\Delta H_i = \widehat{H}_i - H_i$, $i = 1, 2$. Then, to leading order,*

$$(4.15) \quad \|\Delta H_i\| \leq \delta + c_n \left(\|A_i - \tilde{A}_i\| + \|\Delta A_i\| + \frac{3}{2}\delta \right),$$

where c_n is given by (4.1).

Proof. Let $X_i = \arg \min_{X^* X = I} \|\widehat{W}_i - X\|$ and $G_i = \arg \min_{G=G^* \geq 0} \|\widehat{H}_i - G\|$. Then

$$\begin{aligned} X_i G_i &= \widehat{W}_i \widehat{H}_i + (X_i - \widehat{W}_i) \widehat{H}_i + X_i (G_i - \widehat{H}_i) \\ &= \tilde{A}_i + \Delta \tilde{A}_i, \end{aligned}$$

where $\Delta \tilde{A}_i = (A_i - \tilde{A}_i) + \Delta A_i + (X_i - \widehat{W}_i) \widehat{H}_i + X_i (G_i - \widehat{H}_i)$. Since X_i is unitary and G_i is Hermitian positive semidefinite, we have

$$\begin{aligned} \|G_i - H_i\| &= \|\mathcal{H}(\tilde{A}_i + \Delta \tilde{A}_i) - \mathcal{H}(\tilde{A}_i)\| \\ &\leq c_n (\|A_i - \tilde{A}_i\| + \|\Delta A_i\| + \|X_i - \widehat{W}_i\| \sigma_1(\widehat{H}_i) + \|G_i - \widehat{H}_i\|). \end{aligned}$$

By assumption, $\|G_i - \widehat{H}_i\| \leq \delta$ and $\|\widehat{W}_i^* \widehat{W}_i - I\| \leq \delta$, so $\|X_i - \widehat{W}_i\| \sim \frac{1}{2}\delta$ as $\delta \rightarrow 0$ [8, Lemma 8.17]. Moreover, $\sigma_1(\widehat{H}_i) \sim \sigma_1(A_i + \Delta A_i) \leq \sigma_1(A + (\Delta A_1^*, \Delta A_2^*)^*) \sim 1$. These facts, together with the inequality

$$\|\widehat{H}_i - H_i\| \leq \|G_i - \widehat{H}_i\| + \|G_i - H_i\|,$$

prove (4.15). \square

Now let

$$B = H_2 - H_1 + \mu(I - \tilde{A}^* \tilde{A}).$$

Then

$$\widehat{B} - B = \Delta B_1 + \Delta H_2 - \Delta H_1 + \mu(\tilde{A} - A)^* A + \mu \tilde{A}^* (\tilde{A} - A),$$

so the preceding lemma implies

$$(4.16) \quad \begin{aligned} \|\widehat{B} - B\| &\leq \|\Delta B_1\| + c_n (\|A_1 - \tilde{A}_1\| + \|A_2 - \tilde{A}_2\| + \|\Delta A_1\| + \|\Delta A_2\|) \\ &\quad + (3c_n + 2)\delta + \mu(\sigma_1(A) + 1)d(A). \end{aligned}$$

The forthcoming analysis will rely on a certain pair of functions f and g with the property that $f(B) = H_1$, $g(B) = H_2$, f and g have bounded Fréchet derivative at B , and f and g are analytic on a complex neighborhood of the spectrum of B . Consider first the case in which $r < n$, so that $\mu > 1$. (In our algorithm we always take $\mu = 2$ when $r < n$.) Let

$$\rho = \begin{cases} \frac{1+\mu}{\sqrt{2}} & \text{if } 1 < \mu < 2\sqrt{2} - 1, \\ \sqrt{2} & \text{otherwise,} \end{cases}$$

and define

$$f(z) = \begin{cases} \frac{1}{2}(-z + \sqrt{2 - z^2}) & \text{if } |z| < \rho, \\ 0 & \text{if } |z| \geq \rho, \end{cases}$$

and

$$g(z) = \begin{cases} \frac{1}{2}(z + \sqrt{2 - z^2}) & \text{if } |z| < \rho, \\ 0 & \text{if } |z| \geq \rho. \end{cases}$$

The functions f and g satisfy $f(\mu) = g(\mu) = 0$, and $f(\sin \theta - \cos \theta) = \cos \theta$ and $g(\sin \theta - \cos \theta) = \sin \theta$ for every $\theta \in [0, \frac{\pi}{2}]$. Moreover, they are analytic on $\mathbb{C} \setminus \{z : |z| = \rho\}$. This is an open set containing the spectrum of B , since the equality

$$B = V_1 \begin{pmatrix} S_r - C_r & 0 \\ 0 & \mu I_{n-r} \end{pmatrix} V_1^*$$

shows that B has spectrum contained in $[-1, 1] \cup \{\mu\}$. The functions $f : \mathbb{C}^{n \times n} \rightarrow \mathbb{C}^{n \times n}$ and $g : \mathbb{C}^{n \times n} \rightarrow \mathbb{C}^{n \times n}$ are thus well-defined in a neighborhood of B with

$$f(B) = V_1 \begin{pmatrix} f(S_r - C_r) & 0 \\ 0 & f(\mu I_{n-r}) \end{pmatrix} V_1^* = V_1 \begin{pmatrix} C_r & 0 \\ 0 & 0_{n-r} \end{pmatrix} V_1^* = H_1$$

and

$$g(B) = V_1 \begin{pmatrix} g(S_r - C_r) & 0 \\ 0 & g(\mu I_{n-r}) \end{pmatrix} V_1^* = V_1 \begin{pmatrix} S_r & 0 \\ 0 & 0_{n-r} \end{pmatrix} V_1^* = H_2.$$

Since B is Hermitian and $\sup_{z \in [-1, 1] \cup \{\mu\}} |f'(z)| = 1$, it follows [8, Corollary 3.16] that the Fréchet derivative $L_f(B, \cdot) : \mathbb{C}^{n \times n} \rightarrow \mathbb{C}^{n \times n}$ of f at B satisfies

$$(4.17) \quad \|L_f(B, E)\| \leq \|E\|$$

for every $E \in \mathbb{C}^{n \times n}$. Similarly,

$$(4.18) \quad \|L_g(B, E)\| \leq \|E\|$$

for every $E \in \mathbb{C}^{n \times n}$.

For the case in which $r = n$, we instead simply define $f(z) = \frac{1}{2}(z + \sqrt{2 - z^2})$ and $g(z) = \frac{1}{2}(-z + \sqrt{2 - z^2})$. Arguments analogous to those above show that in this setting (regardless of the value of μ), $f(B) = H_1$, $g(B) = H_2$, and L_f and L_g satisfy (4.17)–(4.18).

We will now show that the backward error $\widehat{W}_1 \widehat{V}_1 \widehat{C} \widehat{V}_1^* - A_1$ is small. To begin, denote $D = \text{diag}(\text{diag}(\widehat{V}_1^* \widehat{H}_1 \widehat{V}_1))$ and observe that

$$\begin{aligned} \widehat{W}_1 \widehat{V}_1 \widehat{C} \widehat{V}_1^* &= \widehat{W}_1 \widehat{V}_1 (D + \Delta C) \widehat{V}_1^* \\ &= \widehat{W}_1 \widehat{V}_1 \widehat{V}_1^* \widehat{H}_1 \widehat{V}_1 \widehat{V}_1^* + \widehat{W}_1 \widehat{V}_1 (D - \widehat{V}_1^* \widehat{H}_1 \widehat{V}_1 + \Delta C) \widehat{V}_1^* \\ &= \widehat{W}_1 \widehat{H}_1 + \widehat{W}_1 (\widehat{V}_1 \widehat{V}_1^* - I) \widehat{H}_1 \widehat{V}_1 \widehat{V}_1^* + \widehat{W}_1 \widehat{H}_1 (\widehat{V}_1 \widehat{V}_1^* - I) \\ &\quad + \widehat{W}_1 \widehat{V}_1 (D - \widehat{V}_1^* \widehat{H}_1 \widehat{V}_1 + \Delta C) \widehat{V}_1^*. \end{aligned}$$

Since $\widehat{W}_1 \widehat{H}_1 = A_1 + \Delta A_1$, it follows that to leading order,

$$(4.19) \quad \|\widehat{W}_1 \widehat{V}_1 \widehat{C} \widehat{V}_1^* - A_1\| \leq \|\Delta A_1\| + 2\delta\sigma_1(\widehat{H}_1) + \|D - \widehat{V}_1^* \widehat{H}_1 \widehat{V}_1\| + \|\Delta C\|.$$

The next lemma estimates $\|D - \widehat{V}_1^* \widehat{H}_1 \widehat{V}_1\|$.

LEMMA 4.3. *To leading order,*

$$\|D - \widehat{V}_1^* \widehat{H}_1 \widehat{V}_1\| \leq 2(\|B - \widehat{B}\| + \|\Delta B_2\| + (1 + \max\{\mu, 1\})\delta + \|\Delta H_1\|).$$

Proof. Since $f(B) = H_1$, we have

$$(4.20) \quad \begin{aligned} \widehat{V}_1^* \widehat{H}_1 \widehat{V}_1 &= \widehat{V}_1^* f(B) \widehat{V}_1 + \widehat{V}_1^* (\widehat{H}_1 - H_1) \widehat{V}_1 \\ &= \widehat{V}_1^* (f(B) - f(B + E)) \widehat{V}_1 + \widehat{V}_1^* f(B + E) \widehat{V}_1 + \widehat{V}_1^* \Delta H_1 \widehat{V}_1 \end{aligned}$$

for any $E \in \mathbb{C}^{n \times n}$. Choosing $E = (\widehat{B} + \Delta B_2) \widehat{V}_1^{-*} \widehat{V}_1^{-1} - B$, so that $\widehat{V}_1^{-1}(B + E) \widehat{V}_1 = \widehat{V}_1^{-1}(\widehat{B} + \Delta B_2) \widehat{V}_1^{-*} = \widehat{\Lambda}$, we find that the second term in (4.20) is equal to

$$\begin{aligned} \widehat{V}_1^* f(B + E) \widehat{V}_1 &= \widehat{V}_1^* \widehat{V}_1 \widehat{V}_1^{-1} f(B + E) \widehat{V}_1 \\ &= \widehat{V}_1^* \widehat{V}_1 f\left(\widehat{V}_1^{-1}(B + E) \widehat{V}_1\right) \\ &= \widehat{V}_1^* \widehat{V}_1 f(\widehat{\Lambda}). \end{aligned}$$

It follows that

$$f(\widehat{\Lambda}) - \widehat{V}_1^* \widehat{H}_1 \widehat{V}_1 = \widehat{V}_1^* (f(B + E) - f(B)) \widehat{V}_1 + (I - \widehat{V}_1^* \widehat{V}_1) f(\widehat{\Lambda}) - \widehat{V}_1^* \Delta H_1 \widehat{V}_1.$$

By (4.17), the first term above is bounded by

$$\begin{aligned} \|\widehat{V}_1^* (f(B + E) - f(B)) \widehat{V}_1\| &\leq \|E\| \\ &= \left\| (\widehat{B} - B + \Delta B_2) \widehat{V}_1^{-*} \widehat{V}_1^{-1} + B (\widehat{V}_1^{-*} \widehat{V}_1^{-1} - I) \right\| \\ &\leq \|\widehat{B} - B\| + \|\Delta B_2\| + \delta\sigma_1(B) \end{aligned}$$

to leading order in δ and $\|E\|$. Thus,

$$\begin{aligned} \|f(\widehat{\Lambda}) - \widehat{V}_1^* \widehat{H}_1 \widehat{V}_1\| &\leq \|\widehat{B} - B\| + \|\Delta B_2\| + \delta\sigma_1(B) + \delta\sigma_1(f(\widehat{\Lambda})) + \|\Delta H_1\| \\ &\leq \|\widehat{B} - B\| + \|\Delta B_2\| + (1 + \max\{\mu, 1\})\delta + \|\Delta H_1\|, \end{aligned}$$

where we have used the fact that $\sigma_1(B) \leq \max\{\mu, 1\}$ and $\sup_{z \in \mathbb{R} \setminus \{\rho\}} |f(z)| = 1$, so

$\sigma_1(f(\widehat{\Lambda})) \leq 1$. The conclusion follows from the inequality above and the bound

$$\begin{aligned} \|D - \widehat{V}_1^* \widehat{H}_1 \widehat{V}_1\| &\leq \|D - f(\widehat{\Lambda})\| + \|\widehat{V}_1^* \widehat{H}_1 \widehat{V}_1 - f(\widehat{\Lambda})\| \\ &= \|\text{diag}(\text{diag}(\widehat{V}_1^* \widehat{H}_1 \widehat{V}_1 - f(\widehat{\Lambda})))\| + \|\widehat{V}_1^* \widehat{H}_1 \widehat{V}_1 - f(\widehat{\Lambda})\| \\ &\leq 2\|\widehat{V}_1^* \widehat{H}_1 \widehat{V}_1 - f(\widehat{\Lambda})\|. \quad \square \end{aligned}$$

The proof of (4.8) is completed by combining Lemma 4.3 with (4.15), (4.16), and (4.19), invoking the asymptotic estimates $\sigma_1(A) \sim 1$, $\sigma_1(\widehat{H}_1) \sim 1$, and invoking the inequalities $\|A_1 - \widetilde{A}_1\| \leq d(A)$, $\|A_2 - \widetilde{A}_2\| \leq d(A)$. The proof of (4.9) is almost identical.

Remarks. Let us recall the discussion in section 3 and reconsider an algorithm based on the eigendecomposition of other choices of B , such as H_1 or $H_1 + H_2$. An attempt to follow the same argument as above to establish stability breaks down, because the resulting functions f, g have unbounded derivatives.

For later use, we mention now a subtle generalization of Theorem 4.1. A careful reading of the proof above shows that (4.8)–(4.9) continue to hold if, for each i , the condition $\|\widehat{W}_i^* \widehat{W}_i - I\| \leq \delta$ is relaxed to the following pair of conditions:

$$(4.21) \quad \sigma_1(\widehat{W}_i) = 1 + o(1),$$

$$(4.22) \quad \|(X_i - \widehat{W}_i) \widehat{H}_i\| \leq \frac{1}{2} \delta + o(\eta),$$

where $X_i = \arg \min_{X^* X = I} \|\widehat{W}_i - X_i\|$. In other words, near orthonormality of \widehat{W}_i is not strictly necessary if (4.21)–(4.22) can be verified. This observation will be used in section 5.2.

5. Algorithms for the polar decomposition and symmetric eigendecomposition. A number of iterative algorithms are available for computing the polar decomposition [8, section 8]. Among them, the scaled Newton and QDWH algorithms are known to be backward stable [12]. Tests on the Zolo-pd algorithm indicate that it is a backward stable, but proving its stability is still an open problem [11].

In virtually all algorithms for computing the polar decomposition $A = WH$, the iterates X_k have the same singular vectors as the original A , that is, $X_k = U r_k(\Sigma) V^*$, where $A = U \Sigma V^*$ is the SVD, where r_k is a (odd and usually rational) function. The goal is to find r_k that maps all $\sigma_i(A)$ to 1, as then we have $X_k = UV^* = W$, as required (then $H = W^* A$). Different methods employ different functions r_k to achieve this. Here we focus on the Zolo-pd algorithm, as it offers high parallelizability, building upon basic matrix operations (multiplication, QR, and Cholesky).

In Zolo-pd, $r_k(\Sigma)$ is taken to be the type $((2p+1)^k, (2p+1)^k - 1)$ best rational approximation to the sign function on $[-\sigma_{\max}(A), -\sigma_{\min}(A)] \cup [\sigma_{\min}(A), \sigma_{\max}(A)]$. (The extremal singular values are estimated unless specified by the user.) Such functions are called Zolotarev's functions, and they have the significant property that r_k can be obtained by appropriately composing two Zolotarev functions of types $((2p+1)^{k-1}, (2p+1)^{k-1} - 1)$ and $(2p+1, 2p)$. Combining this fact with a partial fraction representation of rational functions, Zolo-pd requires just two iterations for convergence in double precision, each iteration involving $p(\leq 8)$ QR or Cholesky factorizations, which can be executed in parallel. For details, see [11]. QDWH is a special case where $p = 1$, which minimizes the flop count but has the largest number of iterations (and less parallelizability).

For the symmetric eigenvalue decomposition, a spectral divide-and-conquer algorithm [13] can be developed also based on the polar decomposition. The idea is that for a symmetric matrix B , the unitary polar factor W is equivalent to the matrix sign decomposition and has eigenvalues ± 1 since the SVD and eigendecomposition for a symmetric B are related by $B = U\Sigma V^* = (US)(S\Sigma)V^* = V\Lambda V^*$ for $S = \text{diag}(\pm 1)$, so $U = VS$, and hence $W = UV^* = VSV^* =: [V_+ \ V_-] \begin{bmatrix} I_{n_+} & \\ & -I_{n-n_+} \end{bmatrix} [V_+ \ V_-]^*$, where n_+ is the number of positive eigenvalues in B (we work with a shifted matrix $B - sI$ so that $n_+ \approx n/2$). Thus $\frac{1}{2}(W + I) = V_+ V_+^*$ is a partial isometry onto the eigenspace corresponding to the positive eigenvalues of B . From this we can obtain an orthogonal transformation $\tilde{V} := [V_+ \ V_-]$ that block diagonalizes B . We perform this process recursively on the decoupled diagonal blocks to diagonalize the matrix, resulting in the full eigendecomposition. It transpires that the overall process is backward stable if each polar decomposition is computed in a backward stable way, analogous to Theorem 4.1.

Of course, classical algorithms for symmetric eigendecomposition based on reduction to tridiagonal form [7, Chapter 8] are both effective and stable. An advantage of algorithms based on the polar decomposition is that they can be implemented in a communication-minimizing manner using only BLAS-3 operations such as matrix multiplication, QR factorization, and Cholesky decomposition.

5.1. Dealing with ill-conditioned matrices. The matrices A_i can be ill-conditioned, and this impacts the polar decompositions in lines 1–2 of Algorithm 3.1: $W_1 H_1 = A_1$ is ill-conditioned when there exists $\theta_i \approx \frac{\pi}{2}$, and $W_2 H_2 = A_2$ when $\theta_i \approx 0$. When the conditioning is $O(u^{-1})$ or larger, Zolo-pd becomes expensive and even unreliable, as Zolotarev's function of type at most $(17^2, 17^2 - 1)$ may not be enough to map the $O(u)$ singular values to 1. Below we assume the use of double precision arithmetic; the value of p (here $p = 8$) will depend on the unit roundoff u (though rather weakly, like $\sqrt{|\log(u)|}$).

Here we discuss a remedy for such situations. When computing the polar decompositions, we apply the Zolo-pd algorithm, but working with the narrower interval $[-1, -\epsilon] \cup [\epsilon, 1]$, rather than $[-1, -\sigma_{\min}(A_1)] \cup [\sigma_{\min}(A_1), 1]$. We choose ϵ to be a modest multiple of unit roundoff; here $\epsilon = 10^{-15}$. The resulting modified Zolo-pd computes $\tilde{W}_i := U_i r_2(\Sigma_i) V_1^*$, where r_2 is the Zolotarev function of type $(17^2, 17^2 - 1)$ on $[-1, -\epsilon] \cup [\epsilon, 1]$. In particular, this gives $r_2(x) = 1 - O(u)$ for $x \in [\epsilon, 1]$, and $0 \leq r_2(x) \leq 1$ on $[0, 1]$. It follows that $\tilde{H}_i := \tilde{W}_i^* A_i = V_1 \Sigma_i r_2(\Sigma_i) V_1^*$ has eigenvalues $\lambda_j(\tilde{H}_i) \in \sigma_j(A_i) + [-\tilde{\epsilon}, 0]$, where $\tilde{\epsilon} = O(u)$. We also have

$$(5.1) \quad \|\tilde{H}_i - H_i\| = \|\Sigma_i r_2(\Sigma_i) - \Sigma_i\|,$$

where $\Sigma_i r_2(\Sigma_i) - \Sigma_i$ is diagonal with (j, j) element $a_j := \sigma_j(A_i) r_2(\sigma_j(A_i)) - \sigma_j(A_i)$. We claim that $|a_j| = O(u)$ for all j : indeed, for j such that $\sigma_j(A_i) > \epsilon$, we have $r_2(\sigma_j(A_i)) = 1 - O(u)$, so $|a_j| = O(u)$. If $\sigma_j(A_i) \leq \epsilon$, then $|a_j| \leq |\sigma_j(A_i)| \leq \epsilon = O(u)$. Together with (5.1) we obtain $\|\tilde{H}_i - H_i\| = O(u)$. Moreover, we have $\|\tilde{W}_i \tilde{H}_i - A\| = \|r_2(\Sigma_i)^2 \Sigma_i - \Sigma_i\|$, which is also $O(u)$ by a similar argument. Summarizing, we have

$$(5.2) \quad \|\tilde{H}_i - H_i\| = O(u), \quad \|\tilde{W}_i \tilde{H}_i - A\| = O(u),$$

even though $\tilde{W}_i \tilde{H}_i$ is not a polar decomposition since \tilde{W}_i is not orthogonal.

In view of the first equation in (5.2), we proceed to lines 3–5 in Algorithm 3.1, which gives results that are backward stable. The only issue lies in lines 6–7, where

we would compute $\widetilde{W}_i \widetilde{V}_1$: since \widetilde{W}_i does not have orthonormal columns, neither does $\widetilde{W}_i \widetilde{V}_1$.

To overcome this issue, we make the following observation: the R -factors in the QR factorizations of A_i and H_i are identical, that is, $A_i = Q_i R_i$ and $H_i = Q_{i,H} R_i$. (We adopt the convention that the diagonal elements of R are nonnegative; this makes the QR factorization unique [7, section 5.2.1].) It follows that in the QR and polar decompositions $A_i = Q_i R_i = W_i H_i$, we have the relation $W_i = Q_i Q_{i,H}^*$.

Recalling from (5.2) that $\widetilde{H}_i = H_i + O(u)$, this suggests the following: before line 6 of Algorithm 3.1, compute the QR factorizations $A_i = Q_i R_i$ and $\widetilde{H}_i = \widetilde{Q}_{i,H} \widetilde{R}_i$, and redefine $W_i := Q_i \widetilde{Q}_{i,H}^*$. We summarize the process in Algorithm 5.1.

Algorithm 5.1 Modification to Algorithm 3.1 when A_1 and/or A_2 is ill-conditioned.

- 1: In place of lines 1–2 of Algorithm 3.1, compute $A_i \approx \widetilde{W}_i \widetilde{H}_i$, obtained by a modified Zolo-pd mapping the interval $[-1, -\epsilon] \cup [\epsilon, 1]$ to 1 ($\epsilon = 10^{-15}$ in double precision)
 - 2: Compute QR factorizations $A_i = Q_i R_i$ and $\widetilde{H}_i = \widetilde{Q}_{i,H} \widetilde{R}_i$
 - 3: Set $W_i = Q_i \widetilde{Q}_{i,H}^*$, $H_i = \widetilde{H}_i$
 - 4: Proceed with lines 3 onward of Algorithm 3.1
-

In practice, since it is usually unknown a priori if A_i is ill-conditioned, we execute Zolo-pd as usual, in which one of the preprocessing step is to estimate $\sigma_{\min}(A_i)$: if it is larger than 10^{-15} , we continue with standard Zolo-pd; otherwise A_i is ill-conditioned, and we run Algorithm 5.1. (This is necessary only for i for which A_i is ill-conditioned.)

An important question is whether this process is stable. Since $\|\widetilde{H}_i - H_i\| = O(u)$ by (5.2) and the resulting W_i is orthogonal to working precision by construction, the main question is whether $W_i \widetilde{H}_i$ still gives a backward stable polar decomposition for A_i , that is, whether $\|W_i \widetilde{H}_i - A\| = O(u)$ holds or not. To examine this, note that

$$\|W_i \widetilde{H}_i - A\| = \|Q_i \widetilde{Q}_{i,H}^* \widetilde{H}_i - Q_i R_i\| = \|Q_i \widetilde{R}_i - Q_i R_i\| = \|\widetilde{R}_i - R_i\|.$$

The question therefore becomes whether $\|\widetilde{R}_i - R_i\| = O(u)$ holds. (Note that this is an inexpensive condition to check.) In general, the triangular factor in the QR factorization can be ill-conditioned; however, it is known to be usually much better conditioned than the original matrix [5]. Indeed in all our experiments with ill-conditioned A_i , we observed that $\|\widetilde{R}_i - R_i\|$ was a small multiple of unit roundoff, indicating Algorithm 5.1 is an effective workaround for ill-conditioned problems.

In the rare event that $\|\widetilde{R}_i - R_i\|$ is unacceptably large, a conceptually simple and robust (but expensive) workaround is to compute the polar decompositions via the SVD, which is unconditionally backward stable.

5.2. Dealing with rank deficiency. When A is a rank deficient partial isometry ($r < n$) rather than having orthonormal columns, a natural goal is to compute the economical decomposition (2.2), as it saves memory requirements and allows efficient operations such as multiplications.

Recall that the rank r of A can be computed via $\|A\|_F \approx \sqrt{r}$; here we assume that $r < n$. When $r < n$, both A_1 and A_2 are rank-deficient. As described in [13], QDWH and Zolo-pd are capable of computing the canonical polar decomposition, and in exact arithmetic it computes $A_i = W_i H_i$, where W_i are partial isometries. In finite-precision arithmetic, however, roundoff error usually causes the zero singular values of A_i to get mapped to nonzero values. These eventually converge to 1 in QDWH (in six iterations), but Zolo-pd, which terminates in two iterations, faces the

same difficulty discussed above, usually producing a W_i that has $n - r$ singular values that are between 0 and 1. Then the computed \widetilde{W}_i does not have singular values that are all close to 0 or 1, and hence neither does the resulting U_i . Here we discuss a modification of Zolo-pd to deal with such issues.

The first step is identical to the previous subsection: we invoke the modified Zolo-pd to map singular values in $[\epsilon, 1]$ to 1. Recall that the resulting H_i are correct to working precision. We then compute $B = H_2 - H_1 + \mu(I - A^*A)$ with $\mu = 2$, and its eigendecomposition $B = V_1 \Lambda V_1^*$ as usual, and proceed as described in the second-to-last remark after Algorithm 3.1 to extract the relevant matrices. We summarize the process in Algorithm 5.2.

Algorithm 5.2 Modification to Algorithm 3.1 when A is rank deficient.

- 1: Compute $A_i \approx \widetilde{W}_i \widetilde{H}_i$ for $i = 1, 2$, obtained by a modified Zolo-pd mapping the interval $[-1, -\epsilon] \cup [\epsilon, 1]$ to 1
 - 2: $W_i = \widetilde{W}_i, H_i = \widetilde{H}_i$
 - 3: $B = H_2 - H_1 + 2(I - A^*A)$
 - 4: $V_1 \Lambda V_1^* = B$ (symmetric eigendecomposition)
 - 5: Find $V_{1r} \in \mathbb{C}^{n \times r}$, the eigenvectors corresponding to eigenvalues in $[-1, 1]$
 - 6: $U_1 = W_1 V_{1r}$
 - 7: $U_2 = W_2 V_{1r}$
 - 8: $C = \text{diag}(\text{diag}(V_{1r}^* H_1 V_{1r}))$
 - 9: $S = \text{diag}(\text{diag}(V_{1r}^* H_2 V_{1r}))$
 - 10: **return** $U_1, U_2, C, S, V_1 := V_{1r}$
-

Note that the modified Zolo-pd is used in exactly the same way in Algorithms 5.1 and 5.2. This lets us easily treat the situation where both issues are present: A is rank-deficient and C or S is ill-conditioned. In this case, we replace line 2 of Algorithm 5.2 by lines 2–3 of Algorithm 5.1. We also note that in spectral divide-and-conquer algorithms such as Zolo-eig, it is straightforward to modify the algorithm to compute only eigenpairs lying in the prescribed interval $[-1, 1]$ (by splitting, e.g., at 1.1), thus saving some cost.

A nontrivial question here is this: does each U_i have orthonormal columns? This is not obvious because W_i is not orthonormal. To examine this, we momentarily abuse notation by writing U_i as $U_{i,r}$ to distinguish it from the U_i from earlier sections ($U_{i,r}$ consists of the first r columns of U_i). Now recall that $W_i = U_{i,r} r_2(\Sigma_i) V_{1r}^*$, which we rewrite as

$$W_i = [U_{i,r}, U_{i,r}^\perp] \begin{pmatrix} r_2(\Sigma_{i,r}) & \\ & r_2(\Sigma_{i,>r}) \end{pmatrix} [V_{1r}, V_{1r}^\perp]^*,$$

where $\Sigma_{i,r}$ are the r (nonzero) leading singular values of A_i . Assuming that they are larger than ϵ (when this is violated we invoke Algorithm 5.1 as mentioned above), we have $r_2(\Sigma_{i,r}) = I_r + O(u)$. Therefore $W_i V_{1r} = U_{i,r} + O(u)$, which is orthonormal to working precision, as required. Our experiments illustrate that U_i are indeed orthonormal to working precision.

A second question that must be addressed is whether Algorithm 5.2 is still backward stable, given that $\|W_i^* W_i - I\|$ may be much greater than u . To answer this question, we recall the remark made at the end of section 4: backward stability is still ensured if $\sigma_1(W_i) \leq 1 + O(u)$ and $\|(X_i - W_i)H_i\| = O(u)$, where $X_i = \arg \min_{X^* X = I} \|X - W_i\|$. The first of these conditions is clearly satisfied. For the second, observe that $H_i = W_i^* A = V_{1r} r_2(\Sigma_i) \Sigma_i V_{1r}^*$ and by [8, Theorem 8.4],

$X_i = U_i V_1^*$. Thus,

$$\begin{aligned} & (X_i - W_i)H_i \\ &= U_i(I - r_2(\Sigma_i))r_2(\Sigma_i)\Sigma_i V_1^* \\ &= U_i \begin{pmatrix} (I_r - r_2(\Sigma_{i,r}))r_2(\Sigma_{i,r})\Sigma_{i,r} & \\ & (I_{n-r} - r_2(\Sigma_{i,>r}))r_2(\Sigma_{i,>r})\Sigma_{i,>r} \end{pmatrix} V_1^*. \end{aligned}$$

In the block diagonal matrix above, the upper left block is $O(u)$ because $\|I_r - r_2(\Sigma_{i,r})\| = O(u)$ and $\|\Sigma_{i,r}r_2(\Sigma_{i,r})\| \leq 1 + O(u)$, whereas the lower right block is $O(u)$ because $\|\Sigma_{i,>r}\| = O(u)$ and $\|(I_{n-r} - r_2(\Sigma_{i,>r}))r_2(\Sigma_{i,>r})\| \leq 1 + O(u)$. We conclude that $\|(X_i - W_i)H_i\| = O(u)$, as desired.

We emphasize that all of the arguments above hinge upon the assumption that $r_2(A_i)$ is computed in a backward stable manner in the Zolo-pd algorithm. This is supported by extensive numerical evidence but not yet by a proof [11].

6. Numerical examples. We tested Algorithm 3.1 on the following examples adapted from [19]. Below, $\text{nint}(x)$ denotes the nearest integer to a real number x , and $St(n, m) = \{A \in \mathbb{C}^{m \times n} \mid A^*A = I\}$ denotes the complex Stiefel manifold.

1. (Haar) A $2n \times n$ matrix sampled randomly from the Haar measure on $St(n, 2n)$.
2. (Clustered) A $2n \times n$ matrix $A = \begin{pmatrix} U_1 C V_1^* \\ U_2 S V_1^* \end{pmatrix}$, where $U_1, U_2, V_2 \in \mathbb{C}^{n \times n}$ are sampled randomly from the Haar measure on $St(n, n)$, and C and S are generated with the following MATLAB commands:

```
delta = 10^(-18*rand(n+1), 1);
theta = pi/2*cumsum(delta(1:n))/sum(delta);
C = diag(cos(theta));
S = diag(sin(theta));
```

This code tends to produce principal angles $\theta_1, \theta_2, \dots, \theta_n$ that are highly clustered.

3. (Rank-deficient, Haar) A $2n \times n$ matrix of rank $r = \text{nint}(3n/4)$ given by $A = XY^*$, where $X \in \mathbb{C}^{2n \times r}$ and $Y \in \mathbb{C}^{n \times r}$ are sampled randomly from the Haar measure on $St(r, 2n)$ and $St(r, n)$, respectively.
 4. (Rank-deficient, clustered) A $2n \times n$ matrix of rank $r = \text{nint}(3n/4)$ generated in the same way as in (2), but with C_{ii} and S_{ii} replaced by zero for $n - r$ random indices $i \in \{1, 2, \dots, n\}$.
- 1'-4'. (Noisy) Tests (1-4), each perturbed by $1e-10*(\text{randn}(2*n,n)+i*\text{randn}(2*n,n))$.

We ran these tests with $n = \text{nint}(30 \cdot 2^{j/2})$, $j = 0, 1, \dots, 9$. In all of the tests, we performed the postprocessing procedure suggested in remark (3.iii).

Tables 1-2 report the scaled residuals $\frac{\|\hat{A}-A\|_2}{d(A)}$ and scaled orthogonality measures $\frac{\|\hat{U}_1^* \hat{U}_1 - I\|_2}{u}$, $\frac{\|\hat{U}_2^* \hat{U}_2 - I\|_2}{u}$, $\frac{\|\hat{V}_1^* \hat{V}_1 - I\|_2}{u}$ for each test, where $\hat{A} = \begin{pmatrix} \hat{U}_1 \hat{C} \hat{V}_1^* \\ \hat{U}_2 \hat{S} \hat{V}_1^* \end{pmatrix}$, $d(A)$ is given by (2.5), and $u = 2^{-53}$ is the unit roundoff. For comparison, the results obtained with LAPACK's `csd` function are recorded in Table 1 as well. (Results from LAPACK are not shown in Table 2, since LAPACK's `csd` function applies only to full-rank matrices.) Inspection of Table 1 reveals that in most of the tests involving full-rank A , the residuals and orthogonality measures were closer to zero for Zolo-csd than for LAPACK.

Acknowledgments. We thank the referees and the editor for their helpful suggestions, particularly for a comment that led to (3.ix).

TABLE 1

Residuals and orthogonality measures for Zolo-csd (Z) and LAPACK (L) on the test matrices (1), (1)', (2), and (2)'.

Test	n	$d(A)$	$\frac{\ \widehat{A}-A\ _2}{d(A)}$		$\ \widehat{U}_1^* \widehat{U}_1 - I\ _2$		$\ \widehat{U}_2^* \widehat{U}_2 - I\ _2$		$\ \widehat{V}_1^* \widehat{V}_1 - I\ _2$	
			Z	L	Z	L	Z	L	Z	L
1	30	$4.4 \cdot 10^{-16}$	3.34	20.63	14.91	26.45	14.73	40.06	4.89	23.23
	42	$6.7 \cdot 10^{-16}$	2.23	15.86	17.84	32.32	14.35	36.48	5.51	42.12
	60	$4.4 \cdot 10^{-16}$	3.42	24.43	12.10	43.96	13.26	37.12	5.39	48.55
	85	$5.6 \cdot 10^{-16}$	3.68	14.65	19.82	51.03	20.54	43.64	5.97	52.56
	120	$4.4 \cdot 10^{-16}$	4.79	26.80	17.64	52.09	22.86	62.29	7.62	57.64
	170	$1.8 \cdot 10^{-15}$	1.41	7.70	29.31	64.29	28.63	72.25	8.55	72.11
	240	$2.2 \cdot 10^{-15}$	1.20	6.92	24.01	83.93	23.94	72.90	10.20	102.09
	339	$3.8 \cdot 10^{-15}$	0.75	4.26	30.54	83.15	33.81	99.92	9.78	92.59
	480	$4.2 \cdot 10^{-15}$	0.70	4.68	24.53	97.90	26.10	128.97	11.28	113.41
	679	$3.3 \cdot 10^{-15}$	0.95	12.15	22.78	157.24	29.36	118.41	11.45	148.04
1'	30	$1.0 \cdot 10^{-9}$	1.12	1.65	13.68	27.99	12.59	29.93	4.64	26.63
	42	$1.2 \cdot 10^{-9}$	1.12	1.64	16.47	34.00	12.92	29.40	5.13	32.33
	60	$1.5 \cdot 10^{-9}$	1.12	1.61	17.34	36.49	19.18	44.51	5.70	42.80
	85	$1.8 \cdot 10^{-9}$	1.12	1.66	15.08	48.39	13.93	46.39	6.14	48.70
	120	$2.1 \cdot 10^{-9}$	1.13	1.66	18.39	56.58	20.43	67.83	7.42	57.91
	170	$2.5 \cdot 10^{-9}$	1.13	1.64	19.00	62.15	18.10	73.14	8.74	76.03
	240	$3.1 \cdot 10^{-9}$	1.12	1.61	25.99	72.08	23.78	89.14	9.90	77.96
	339	$3.6 \cdot 10^{-9}$	1.13	1.66	19.58	88.79	22.96	97.81	9.85	102.62
	480	$4.3 \cdot 10^{-9}$	1.13	1.66	25.08	96.85	29.18	114.03	11.62	115.54
	679	$5.2 \cdot 10^{-9}$	1.13	1.63	25.00	117.16	22.59	129.55	11.48	130.97
2	30	$5.6 \cdot 10^{-16}$	4.01	15.11	19.31	28.52	22.95	21.81	4.68	21.44
	42	$5.6 \cdot 10^{-16}$	5.67	16.85	9.71	28.92	16.05	23.76	4.84	27.75
	60	$4.4 \cdot 10^{-16}$	9.52	17.15	22.90	37.85	6.41	42.61	5.53	37.05
	85	$4.4 \cdot 10^{-16}$	10.51	54.97	17.92	44.89	7.06	42.79	6.02	39.91
	120	$4.4 \cdot 10^{-16}$	11.80	60.01	18.24	51.92	8.86	48.47	7.78	46.76
	170	$1.3 \cdot 10^{-15}$	4.01	18.56	31.75	72.35	22.20	63.85	9.15	72.10
	240	$2.2 \cdot 10^{-15}$	3.64	10.39	24.84	74.19	11.48	74.36	9.95	68.58
	339	$2.7 \cdot 10^{-15}$	2.81	14.91	27.60	114.44	20.81	195.52	9.74	101.61
	480	$3.2 \cdot 10^{-15}$	3.09	10.16	12.47	108.85	12.38	118.05	11.30	137.93
	679	$3.3 \cdot 10^{-15}$	3.05	9.41	33.61	126.26	12.63	149.26	11.52	139.84
2'	30	$9.8 \cdot 10^{-10}$	1.28	1.69	22.99	27.13	11.25	20.80	5.23	21.23
	42	$1.2 \cdot 10^{-9}$	1.30	1.68	18.05	30.55	11.99	30.92	5.00	30.14
	60	$1.5 \cdot 10^{-9}$	1.25	1.69	16.63	40.00	12.80	37.03	5.47	38.47
	85	$1.8 \cdot 10^{-9}$	1.14	1.61	14.87	53.31	13.57	50.00	6.52	48.85
	120	$2.1 \cdot 10^{-9}$	1.26	1.66	20.44	49.26	16.41	67.56	7.58	48.72
	170	$2.6 \cdot 10^{-9}$	1.16	1.66	25.63	63.21	19.44	65.30	8.62	65.34
	240	$3.0 \cdot 10^{-9}$	1.18	1.66	23.27	82.38	22.94	85.60	9.94	86.37
	339	$3.6 \cdot 10^{-9}$	1.16	1.65	26.45	109.93	21.10	101.70	9.89	85.35
	480	$4.3 \cdot 10^{-9}$	1.17	1.66	25.76	116.44	27.44	120.71	11.27	104.99
	679	$5.2 \cdot 10^{-9}$	1.14	1.64	23.76	140.07	29.24	130.63	11.67	156.60

TABLE 2
Residuals and orthogonality measures for Zolo-csd on the test matrices (3), (3)', (4), and (4)'.
Residuals and orthogonality measures for Zolo-csd on the test matrices (3), (3)', (4), and (4)'.

Test	n	$d(A)$	$\frac{\ \widehat{A}-A\ _2}{d(A)}$	$\frac{\ \widehat{U}_1^* \widehat{U}_1 - I\ _2}{u}$	$\frac{\ \widehat{U}_2^* \widehat{U}_2 - I\ _2}{u}$	$\frac{\ \widehat{V}_1^* \widehat{V}_1 - I\ _2}{u}$
3	30	$6.7 \cdot 10^{-16}$	7.28	4.03	4.53	4.28
	42	$4.6 \cdot 10^{-16}$	15.99	5.26	5.34	4.37
	60	$8.9 \cdot 10^{-16}$	7.98	5.23	5.17	5.38
	85	$6.1 \cdot 10^{-16}$	21.78	5.73	5.99	5.42
	120	$7.1 \cdot 10^{-16}$	52.89	6.76	6.70	6.66
	170	$8.0 \cdot 10^{-16}$	62.66	8.25	8.32	7.80
	240	$8.9 \cdot 10^{-16}$	34.87	9.70	9.44	8.53
	339	$1.1 \cdot 10^{-15}$	30.22	9.21	9.43	8.31
	480	$1.9 \cdot 10^{-15}$	27.90	10.61	10.71	9.65
	679	$2.6 \cdot 10^{-15}$	84.96	11.06	11.12	10.06
3'	30	$1.2 \cdot 10^{-9}$	2.31	23.53	22.59	4.22
	42	$1.3 \cdot 10^{-9}$	2.51	24.88	25.77	4.43
	60	$1.7 \cdot 10^{-9}$	2.43	25.75	23.80	4.89
	85	$2.0 \cdot 10^{-9}$	2.47	26.57	26.13	5.80
	120	$2.4 \cdot 10^{-9}$	2.45	26.68	27.33	6.97
	170	$2.9 \cdot 10^{-9}$	2.41	28.57	28.36	7.36
	240	$3.4 \cdot 10^{-9}$	2.47	30.63	30.44	8.74
	339	$4.2 \cdot 10^{-9}$	2.45	29.87	29.68	8.46
	480	$5.0 \cdot 10^{-9}$	2.40	31.80	31.51	9.35
	679	$5.9 \cdot 10^{-9}$	2.47	31.56	31.71	10.18
4	30	$4.4 \cdot 10^{-16}$	10.24	4.23	4.64	4.78
	42	$6.7 \cdot 10^{-16}$	11.62	4.23	4.61	4.39
	60	$6.1 \cdot 10^{-16}$	22.43	5.13	5.30	5.12
	85	$6.7 \cdot 10^{-16}$	22.75	6.12	5.64	5.76
	120	$7.2 \cdot 10^{-16}$	23.36	6.49	7.11	6.18
	170	$8.1 \cdot 10^{-16}$	18.64	8.31	8.44	7.45
	240	$9.6 \cdot 10^{-16}$	31.04	9.51	9.27	8.91
	339	$1.1 \cdot 10^{-15}$	41.15	9.29	9.43	8.56
	480	$1.8 \cdot 10^{-15}$	27.76	10.63	10.39	9.92
	679	$1.6 \cdot 10^{-15}$	33.13	10.90	10.98	10.19
4'	30	$1.1 \cdot 10^{-9}$	2.19	21.78	27.26	3.98
	42	$1.3 \cdot 10^{-9}$	3.21	22.12	17.88	4.89
	60	$1.6 \cdot 10^{-9}$	2.67	27.89	20.36	5.31
	85	$2.0 \cdot 10^{-9}$	2.24	19.58	27.85	5.50
	120	$2.4 \cdot 10^{-9}$	2.41	29.12	26.10	6.34
	170	$2.9 \cdot 10^{-9}$	2.60	31.46	26.97	7.86
	240	$3.4 \cdot 10^{-9}$	2.36	29.08	31.94	8.87
	339	$4.1 \cdot 10^{-9}$	2.43	27.06	27.09	8.50
	480	$4.9 \cdot 10^{-9}$	2.50	31.65	31.68	9.75
	679	$5.9 \cdot 10^{-9}$	2.55	33.87	31.08	10.08

REFERENCES

- [1] Z. BAI, *CSD, GSVD, Their Applications and Computations*, Technical report IMA preprint series 958, Institute for Mathematics and its Applications, University of Minnesota, 1992.
- [2] Z. BAI AND J. W. DEMMEL, *Computing the generalized singular value decomposition*, SIAM J. Sci. Comput., 14 (1993), pp. 1464–1486.
- [3] R. BHATIA, M.-D. CHOI, AND C. DAVIS, *Comparing a matrix to its off-diagonal part*, in The Gohberg Anniversary Collection, Springer, New York, 1989, pp. 151–164.
- [4] A. BUNSE-GERSTNER, R. BYERS, AND V. MEHRMANN, *Numerical methods for simultaneous diagonalization*, SIAM J. Matrix Anal. Appl., 14 (1993), pp. 927–949.
- [5] X.-W. CHANG, C. C. PAIGE, AND G. W. STEWART, *Perturbation analyses for the QR factorization*, SIAM J. Matrix Anal. Appl., 18 (1997), pp. 775–791.
- [6] K. A. GALLIVAN, A. SRIVASTAVA, X. LIU, AND P. VAN DOOREN, *Efficient algorithms for inferences on Grassmann manifolds*, in Proceedings of the IEEE Workshop on Statistical Signal Processing, 2003, pp. 315–318.
- [7] G. H. GOLUB AND C. F. VAN LOAN, *Matrix Computations*, Vol. 3, Johns Hopkins University Press, Baltimore, MD, 2012.
- [8] N. J. HIGHAM, *Functions of Matrices: Theory and Computation*, SIAM, Philadelphia, 2008.
- [9] B. LASZKIEWICZ AND K. ZIETAK, *Approximation of matrices and a family of Gander methods for polar decomposition*, BIT, 46 (2006), pp. 345–366.
- [10] R. MATHIAS, *The Hadamard operator norm of a circulant and applications*, SIAM J. Matrix Anal. Appl., 14 (1993), pp. 1152–1167.
- [11] Y. NAKATSUKASA AND R. W. FREUND, *Computing fundamental matrix decompositions accurately via the matrix sign function in two iterations: The power of Zolotarev’s functions*, SIAM Rev., 58 (2016), pp. 461–493.
- [12] Y. NAKATSUKASA AND N. J. HIGHAM, *Backward stability of iterations for computing the polar decomposition*, SIAM J. Matrix Anal. Appl., 33 (2012), pp. 460–479.
- [13] Y. NAKATSUKASA AND N. J. HIGHAM, *Stable and efficient spectral divide and conquer algorithms for the symmetric eigenvalue decomposition and the SVD*, SIAM J. Sci. Comput., 35 (2013), pp. A1325–A1349.
- [14] C. C. PAIGE AND M. WEI, *History and generality of the CS decomposition*, Linear Algebra Appl., 208 (1994), pp. 303–326.
- [15] B. N. PARLETT, *The Symmetric Eigenvalue Problem*, SIAM, Philadelphia, 1998.
- [16] G. W. STEWART, *Computing the CS decomposition of a partitioned orthonormal matrix*, Numer. Math., 40 (1982), pp. 297–306.
- [17] B. D. SUTTON, *Computing the complete CS decomposition*, Numer. Algorithms, 50 (2009), pp. 33–65.
- [18] B. D. SUTTON, *Stable computation of the CS decomposition: Simultaneous bidiagonalization*, SIAM J. Matrix Anal. Appl., 33 (2012), pp. 1–21.
- [19] B. D. SUTTON, *Divide and conquer the CS decomposition*, SIAM J. Matrix Anal. Appl., 34 (2013), pp. 417–444.
- [20] R. R. TUCCI, *A Rudimentary Quantum Compiler*, (2nd ed.), preprint, arXiv:quant-ph/9902062, 1999.
- [21] C. F. VAN LOAN, *Computing the CS and the generalized singular value decompositions*, Numer. Math., 46 (1985), pp. 479–491.