Sparse PCA on fixed-rank matrices *

Alberto Del Pia †

January 7, 2022

Abstract

Sparse PCA is the optimization problem obtained from PCA by adding a sparsity constraint on the principal components. Sparse PCA is NP-hard and hard to approximate even in the single-component case. In this paper we settle the computational complexity of sparse PCA with respect to the rank of the covariance matrix. We show that, if the rank of the covariance matrix is a fixed value, then there is an algorithm that solves sparse PCA to global optimality, whose running time is polynomial in the number of features. We also prove a similar result for the version of sparse PCA which requires the principal components to have disjoint supports.

Key words: principal component analysis; sparsity; polynomial-time algorithm; global optimum; constant-rank quadratic function

1 Introduction

Principal component analysis is one of the oldest and most popular dimensionality reduction techniques and it is used in a wide array of scientific disciplines. In principal component analysis, we are given a positive integer d and an $n \times m$ *data matrix* Q, where each column represents an independent sample from data population, and each row gives a particular kind of feature. Our task is to find d linear combinations of the n features, called *principal components*, that correspond to directions of maximal variance in the data. The d principal components typically explain most of the variance present in the data, even if the number d is chosen to be much lower than the number of features n in the original dataset. Typically, principal component analysis is formulated in terms of the *covariance matrix*, which is the $n \times n$ positive semidefinite matrix $K := 1/m \cdot (Q - E[Q])(Q - E[Q])^{\mathsf{T}}$. Formally, in principal component analysis we are given an $n \times n$ positive semidefinite matrix K, a positive integer d smaller than n, and we seek an optimal solution to the optimization problem

$$\max_{X \in \mathbb{R}^{n \times d},\ X^{\mathsf{T}} X = I_d} \quad \operatorname{trace}(X^{\mathsf{T}} K X), \tag{PCA}$$

*This work is supported by ONR grant N00014-19-1-2322. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the Office of Naval Research.

†Department of Industrial and Systems Engineering & Wisconsin Institute for Discovery, University of Wisconsin-Madison, Madison, WI, USA. E-mail: `delpia@wisc.edu`.

where I_d denotes the $d \times d$ identity matrix. The d principal components correspond to the d columns of an optimal solution X. It is well-known that PCA can be efficiently solved. In fact, an optimal solution is the matrix X whose columns are the d eigenvectors of K corresponding to the largest d eigenvalues. This optimal solution to PCA can be found in $O(n^3)$ time by computing an eigenvalue decomposition of K. We refer the reader to [29] for an introduction to principal component analysis.

1.1 Sparse PCA

A potential disadvantage of PCA is that the principal components are usually linear combinations of all features. This often makes the derived principal components difficult to interpret. *Sparse principal component analysis* overcomes this disadvantage by requiring the principal components to be linear combinations of just a few features. A direct consequence is that sparse principal component analysis generally provides higher data interpretability as well as better generalization error [7, 18, 16, 35, 6]. A natural formulation of sparse principal component analysis is obtained by adding to PCA a sparsity constraint on the principal components. Formally, in sparse principal component analysis we are given an $n \times n$ positive semidefinite matrix K, positive integers d, s smaller than n, and we seek an optimal solution to the optimization problem

$$\max_{\substack{X \in \mathbb{R}^{n \times d} \\ X^\top X = I_d,\ |\operatorname{supp}(X)| \leq s}} \operatorname{trace}(X^\top K X), \tag{SPCA}$$

where $\operatorname{supp}(X)$ denotes the index set of the nonzero rows of the matrix X. Throughout this paper we will often discuss the special cases of PCA and SPCA with $d = 1$. We refer to these cases, where we only seek one principal component, as the *single-component* cases.

SPCA is NP-hard and hard to approximate [8, 24] even in the single component case. Successful approaches for SPCA include replacing the ℓ_0-norm constraint with an ℓ_1-norm constraint or ℓ_1 penalty [18, 35, 32], branch-and bound [25, 4], semidefinite programming [11, 9, 34, 10], and convex integer programming [12]. A number of other specialized algorithms have been proposed in, e.g., [30, 17, 19, 6, 1, 33, 27]. Only few of these papers directly deal with the general version of SPCA as defined in this paper [6]. In fact, most known algorithms are based on an iterative approach where the principal components are estimated in a one-at-a-time fashion with some sort of deflation step between iterations [23].

The main challenge in solving SPCA to global optimality lies in identifying an optimal support of SPCA among all the $\binom{n}{s}$ index sets of cardinality s, where an *optimal support* of SPCA is defined as an index set $S^* \subseteq \{1, \ldots, n\}$ of cardinality s such that $\operatorname{supp}(X^*) \subseteq S^*$ for an optimal solution X^* to SPCA. Asteris et al. [2] show that, in the single-component case, it is possible to design an algorithm that identifies $O(n^r)$ candidate supports in $O(n^{r+1})$ time, where r denotes the rank of the matrix K, among which lies an optimal support. Therefore, if one considers matrices K whose rank r is a fixed value, both the number of candidate supports constructed and the running time of the algorithm are polynomial in n. In this paper, we confirm that fixing the rank r of K is key in solving

SPCA in polynomial time, and not just in the single-component case, but for any number d of principal components. Next, we formally state our first main result.

Theorem 1. *There is an algorithm that finds an optimal solution to SPCA in time*

$$O\left(n^{\min\{d,r\}(r^2+r)}(\min\{d,r\}nr^2 + n\log n)\right),$$

where r denotes the rank of the input matrix K. In particular, the algorithm constructs $O(n^{\min\{d,r\}(r^2+r)})$ candidate supports among which lies an optimal support.

If the rank r of K is a fixed value, then both the number of candidate supports constructed and the running time of the algorithm are polynomial in n. Theorem 1 constitutes the first polynomial-time algorithm for SPCA, for any fixed value of r. We remark that the running time exponential dependence on r is expected, since SPCA is NP-hard in its full generality. The proof of Theorem 1 is given in Section 3.

1.2 Sparse PCA with disjoint supports

In this paper, we study also *sparse principal component analysis with disjoint supports,* which is a different version of sparse principal component analysis which has been considered in the literature (see, e.g., [3]). Also in this model each principal component is a linear combination of at most s features, but here no feature can be used by two different principal components. Given a matrix X, we denote by x_i its ith column. Furthermore, for a nonnegative integer d, we let $[d] := \{1, \dots, d\}$. With this notation, we can denote by $\mathcal{X}$ the set of feasible matrices

$$\begin{aligned}\mathcal{X} := \{X \in \mathbb{R}^{n\times d} \ : \ & |\text{supp}(x_i)| \le s, \ \|x_i\|_2 = 1, \ \forall i \in [d], \\ & \text{supp}(x_i) \cap \text{supp}(x_{i'}) = \emptyset, \ \forall i \neq i' \in [d]\},\end{aligned}$$

Formally, in sparse principal component analysis with disjoint supports we are given an $n \times n$ positive semidefinite matrix K, positive integers d, s smaller than n, and we seek an optimal solution to the optimization problem

$$\max_{X\in\mathcal{X}} \ \text{trace}(X^\mathsf{T} K X). \qquad \text{(SPCA-DS)}$$

We remark that single-component SPCA is also a special case of SPCA-DS, obtained by setting $d = 1$. Therefore, also SPCA-DS is NP-hard and hard to approximate.

Similarly to SPCA, the main difficulty in SPCA-DS consists in finding an optimal support of SPCA-DS among all the $O(n^{ds})$ families of d index sets of cardinality at most s, where an *optimal support* of SPCA-DS is defined as a family of index sets $\{S_i^*\}_{i\in[d]}$ with $S_i^* \subseteq [n]$, $|S_i^*| \le s$, $\forall i \in [d]$, $S_i^* \cap S_{i'}^* = \emptyset$, $\forall i \neq i' \in [d]$, and such that $\text{supp}(x_i^*) \subseteq S_i^*$, $\forall i \in [d]$, for an optimal solution X^* to SPCA-DS. Our second main result, stated below, implies that we can construct $O((dn)^{d^2(r^2+r)/2})$ candidate supports, among which lies an optimal one.

Theorem 2. *There is an algorithm that finds an optimal solution to SPCA-DS in time*

$$O\left((dn)^{d^2(r^2+r)/2}(dnr^2 + d^3n^5\log n)\right),$$

where r denotes the rank of the input matrix K. In particular, the algorithm constructs $O((dn)^{d^2(r^2+r)/2})$ candidate supports, among which lies an optimal support.

If r and d are fixed values, then both the number of candidate supports constructed and the running time of the algorithm are polynomial in n. Theorem 2 then yields the first polynomial-time algorithm for SPCA-DS, for any fixed values of r and d. To the best of our knowledge, the only other algorithm for SPCA-DS with theoretical guarantees is given in [3], where the authors propose an algorithm that finds an ϵ-approximate solution with running time polynomial in n and $1/\epsilon$, provided that r and d are fixed. The proof of Theorem 2 can be found in Section 5.

1.3 Techniques

We briefly explain the main techniques used in our two algorithms. To simplify the exposition, we assume that r is a fixed value in SPCA, and that both r and d are fixed in SPCA-DS.

The first technique that we introduce is a dimensionality reduction approach which allows us, in both problems, to replace our original matrix X of variables with a new matrix Y of variables which has the advantage of having only a fixed number of entries. This approach can be seen as a multi-component generalization of the auxiliary unit vector technique [22, 31, 26, 21, 20, 2], and has strong connections with procedures used in principal component analysis when the original dimensionality n of the data is much larger than the number of data vectors (see Section 23.1.1 in [29]).

The next technique is a tool from discrete geometry known as the *hyperplane arrangement theorem*. A set $\mathcal{H}$ of p hyperplanes in a q-dimensional Euclidean space determines a partition of the space called the *arrangement of $\mathcal{H}$*. The hyperplane arrangement theorem states that this arrangement consists of $O(p^q)$ full-dimensional polyhedra and can be constructed in time $O(p^q)$. For more details, we refer the reader to [13], and in particular to Theorem 3.3 therein. In both our algorithms, this theorem is employed to partition an extended version of the space of variables Y in a polynomial number of polyhedra. Each one will correspond to a candidate support that we construct, and at least one of them will be optimal to the problem.

Finally, in the proof of Theorem 2, we reduce a restricted version of SPCA-DS to a maximum-profit integer circulation problem. This allows us to make use of the optimality conditions for this problem and of the strongly polynomial-time algorithm by Goldberg and Tarjan [14, 15]. First, the optimality conditions are exploited to obtain the arrangement discussed above. Next, for each polyhedron in the arrangement, we select a vector in its interior and apply Goldberg and Tarjan's algorithm to the corresponding instance. The output of the algorithm allows us to obtain the candidate support $\{S_i\}_{i\in[d]}$ associated with the polyhedron.

1.4 Computational complexity and practical applicability of our algorithms

We remark that we do not expect that a direct implementation of our algorithms will lead to practical algorithms for solving SPCA and SPCA-DS. Rather, our results demonstrate that these problems are efficiently solvable from a theoretical point of view in the settings considered. This is important, because once a problem is shown to be efficiently solvable, usually practical algorithms follow (see, e.g., [5]).

We remark that our analysis of the algorithms can be improved in several ways to obtain marginally better running times. For example, the hyperplane arrangement theorem is always used with a set $\mathcal{H}$ of p hyperplanes *that pass through the origin* in a q-dimensional Euclidean space. In this special case, it is known that the arrangement consists of $O((p-1)^{q-1})$ full-dimensional polyhedra and can be constructed in time $O((p-1)^{q-1})$.

2 A useful lemma

Before proving our main results, we present a lemma that uses standard eigenvalue arguments. This lemma plays a crucial role in the dimensionality reduction performed by both our algorithms. In particular, it implies that the optimal value of a PCA problem with an input matrix of fixed rank can be obtained by solving a different PCA problem with an input matrix of fixed dimensions. In this paper, we denote by $\|\cdot\|_F$ the Frobenius norm.

Lemma 1. *Let M be an $s \times r$ matrix, let d be a positive integer, and let $d' := \min\{d, r\}$. Then*

$$\max_{\substack{X \in \mathbb{R}^{s \times d} \\ X^\mathsf{T} X = I_d}} \|M^\mathsf{T} X\|_F^2 = \max_{\substack{X \in \mathbb{R}^{s \times d'} \\ X^\mathsf{T} X = I_{d'}}} \|M^\mathsf{T} X\|_F^2 = \max_{\substack{Y \in \mathbb{R}^{r \times d'} \\ Y^\mathsf{T} Y = I_{d'}}} \|MY\|_F^2.$$

Proof. Denote by λ_j, for $j \in [s]$, the eigenvalues of the $s \times s$ positive semidefinite matrix MM^T, and assume without loss of generality that $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_s \geq 0$. Then

$$\max_{\substack{X \in \mathbb{R}^{s \times d} \\ X^\mathsf{T} X = I_d}} \|M^\mathsf{T} X\|_F^2 = \max_{\substack{X \in \mathbb{R}^{s \times d} \\ X^\mathsf{T} X = I_d}} \operatorname{trace}(X^\mathsf{T} M M^\mathsf{T} X) = \sum_{j=1}^{d} \lambda_j, \tag{1}$$

where in the first equality we used the definition of Frobenius norm and the second is well known (see, e.g., [29]).

Symmetrically, we obtain

$$\max_{\substack{X \in \mathbb{R}^{s \times d'} \\ X^\mathsf{T} X = I_{d'}}} \|M^\mathsf{T} X\|_F^2 = \max_{\substack{X \in \mathbb{R}^{s \times d'} \\ X^\mathsf{T} X = I_{d'}}} \operatorname{trace}(X^\mathsf{T} M M^\mathsf{T} X) = \sum_{j=1}^{d'} \lambda_j. \tag{2}$$

Since the nonzero eigenvalues of MM^T are at most $\operatorname{rank}(MM^\mathsf{T}) = \operatorname{rank}(M) \leq r$, we have $\sum_{j=1}^{d} \lambda_j = \sum_{j=1}^{d'} \lambda_j$. Thus (1) and (2) coincide and we have shown the first equality in the statement of the lemma.

Denote by μ_k, for $k \in [r]$, the eigenvalues of the $r \times r$ positive semidefinite matrix $M^\mathsf{T} M$, and assume without loss of generality that $\mu_1 \geq \mu_2 \geq \cdots \geq \mu_r \geq 0$. Similarly to our previous derivations, we have

$$\max_{\substack{Y \in \mathbb{R}^{r \times d'} \\ Y^\mathsf{T} Y = I_{d'}}} \|MY\|_F^2 = \max_{\substack{Y \in \mathbb{R}^{r \times d'} \\ Y^\mathsf{T} Y = I_{d'}}} \operatorname{trace}(Y^\mathsf{T} M^\mathsf{T} M Y) = \sum_{k=1}^{d'} \mu_k. \tag{3}$$

Since the nonzero eigenvalues of MM^T and $M^\mathsf{T}M$ are the same, we have $\sum_{j=1}^{d'} \lambda_j = \sum_{k=1}^{d'} \mu_k$. Thus (2) and (3) coincide and we have shown the second equality in the statement of the lemma. □

3 Proof of Theorem 1

Consider SPCA where the input matrix $K \in \mathbb{R}^{n\times n}$ has rank r. Since the matrix K is positive semidefinite, it is well known that we can compute an $n \times r$ matrix R such that $K = RR^\mathsf{T}$ in $O(n^3)$ time, for instance using the Cholesky decomposition with complete pivoting. Using the definition of Frobenius norm, SPCA takes the form

$$\max_{\substack{X\in\mathbb{R}^{n\times d}\\ X^\mathsf{T}X=I_d,\ |\mathrm{supp}(X)|\le s}} \quad \|R^\mathsf{T}X\|_F^2. \tag{4}$$

We introduce some notation that will be used in this proof. For $j \in [n]$, we denote by R_j the jth row of R. Similarly, for $S \subseteq [n]$, R_S denotes the $|S| \times r$ submatrix of R containing only the rows indexed by S. We also denote by $d' := \min\{d, r\}$.

As discussed in Section 1.1, the main difficulty in solving Problem (4) consists in finding an optimal support S^* of Problem (4). In fact, once S^* is determined, an optimal solution X^* to Problem (4) can be obtained by setting to zero the rows of X^* with indices not in S^*, while the other rows of X^* can be obtained by solving the optimization problem

$$\max_{\substack{X\in\mathbb{R}^{s\times d}\\ X^\mathsf{T}X=I_d}} \|R_{S^*}^\mathsf{T}X\|_F^2. \tag{5}$$

This is a PCA problem with an $s\times s$ input matrix. In particular, the input matrix $R_{S^*}R_{S^*}^\mathsf{T}$ can be constructed in $O(s^2r)$ time and an optimal solution can be found in $O(s^3)$ time. Based on this discussion, in the remainder of the proof it suffices to find an optimal support S^* of Problem (4).

The next claim uses Lemma 1 to replace our matrix of variables $X \in \mathbb{R}^{n\times d}$ in Problem (4) with an $r \times d'$ matrix of variables, that we denote by Y. In the claim we consider the following two optimization problems:

$$\max_{\substack{S\subseteq[n]\\ |S|=s}} \max_{\substack{X\in\mathbb{R}^{s\times d}\\ X^\mathsf{T}X=I_d}} \|(R_S)^\mathsf{T}X\|_F^2, \tag{6}$$

$$\max_{\substack{S\subseteq[n]\\ |S|=s}} \max_{\substack{Y\in\mathbb{R}^{r\times d'}\\ Y^\mathsf{T}Y=I_{d'}}} \|R_SY\|_F^2. \tag{7}$$

We say that S^* is an *optimal support* of Problem (6) if there exists X^* such that (S^*, X^*) is an optimal solution to Problem (6). Similarly, we say that S^* is an *optimal support* of Problem (7) if there exists Y^* such that (S^*, Y^*) is an optimal solution to Problem (7).

Claim 1. *The optimal supports of Problems* (4), (6), (7) *coincide.*

Proof of claim. Lemma 1, applied with $M := R_S$, implies that the optimal supports of Problems (6) and (7) coincide. Thus we only need to show that the optimal supports of Problems (4) and (6) coincide. To do so, it suffices to prove the following two statements: (i) For every feasible solution (S, X) to Problem (6) with objective function value γ, there is a feasible solution $\tilde{X}$ to Problem (4) with objective function value γ such that $\operatorname{supp}(\tilde{X}) \subseteq S$; (ii) For every feasible solution $\tilde{X}$ to Problem (4) with objective function value γ, there is a feasible solution (S, X) to Problem (6) with objective function value γ such that $\operatorname{supp}(\tilde{X}) \subseteq S$.

(i). Let (S, X) be a feasible solution to Problem (6) with objective function value γ. Let $\tilde{X} \in \mathbb{R}^{n\times d}$ be obtained from X by adding zero rows corresponding to the indices not in S. Then $\tilde{X}$ is a feasible solution to Problem (4) with objective function value γ such that $\operatorname{supp}(\tilde{X}) \subseteq S$.

(ii). Let $\tilde{X}$ be a feasible solution to Problem (4) with objective function value γ. Let S be a subset of $[n]$ of cardinality s containing $\operatorname{supp}(\tilde{X})$, and let X be obtained from $\tilde{X}$ by dropping the (zero) rows with indices not in S. Then (S, X) is a feasible solution to Problem (6) with objective function value γ such that $\operatorname{supp}(\tilde{X}) \subseteq S$. $\diamond$

Due to Claim 1, in the rest of the proof our goal will be finding an optimal support of Problem (7). Next, we define a restricted version of Problem (7), where we fix the matrix of variables $Y \in \mathbb{R}^{r\times d'}$:

$$\max_{\substack{S\subseteq[n]\\|S|=s}} \|R_S Y\|_F^2.$$

We denote this restricted problem by RST(Y). The next claim gives a simple characterization of the optimal solutions to Problem RST(Y).

Claim 2. *Let $Y \in \mathbb{R}^{r\times d'}$ be given. Then S^* is an optimal solution to Problem* RST(Y) *if and only if $S^* \subseteq [n]$, $|S^*| = s$, and $\|R_j Y\|_2^2 \geq \|R_{j'} Y\|_2^2$, $\forall j \in S^*, \forall j' \in [n] \setminus S^*$.*

Proof of claim. This claim follows trivially by writing Problem RST(Y) in the form

$$\max_{\substack{S\subseteq[n]\\|S|=s}} \sum_{j\in S} \|R_j Y\|_2^2.$$

$\diamond$

Claim 2 implies that in order to find an optimal solution to Problem RST(Y), it is sufficient to order all values $\|R_j Y\|_2^2$, for $j \in [n]$. Therefore, our next task is to partition all matrices $Y \in \mathbb{R}^{r\times d'}$ based on the order of the values $\|R_j Y\|_2^2$, for every $j \in [n]$, that they yield. Each $\|R_j Y\|_2^2$, for $j \in [n]$, is a quadratic polynomial in the entries of Y and every monomial is a constant times the product of two variables in the same column of Y, i.e., $y_{ki}y_{k'i}$, for $k, k' \in [r]$, $i \in [d']$. Since we wish to obtain a polyhedral partition, we introduce a new space of variables that allows us to write each $\|R_j Y\|_2^2$, for $j \in [n]$, as a linear function. Formally, we define the space $\mathcal{E}$ that contains one variable for each $y_{ki}y_{k'i}$, for $k, k' \in [r]$, $i \in [d']$. The dimension of the space $\mathcal{E}$ is therefore $d' \cdot (r^2 + r)/2$.

Note that, for each $Y \in \mathbb{R}^{r\times d'}$, there exists a unique corresponding point in $\mathcal{E}$, that we denote by $\text{ext}(Y)$, obtained by computing all the products $y_{ki}y_{k'i}$, for $k, k' \in [r]$, $i \in [d']$. For each $j \in [n]$, we can now write in time $O(d'r^2)$ a linear function $\ell_j : \mathcal{E} \to \mathbb{R}$ such that $\ell_j(\text{ext}(Y)) = \|R_j Y\|_2^2$ for every matrix $Y \in \mathbb{R}^{r\times d'}$.

Claim 3. *There exist a finite index set T of cardinality $O(n^{d'(r^2+r)})$, full-dimensional polyhedra $P^t \subseteq \mathcal{E}$, for $t \in T$, that cover $\mathcal{E}$, and index sets S^t, for $t \in T$, with the following property: For every $t \in T$, and for every Y such that $\text{ext}(Y) \in P^t$, S^t is an optimal solution to Problem* $\text{RST}(Y)$. *The polyhedra P^t, for $t \in T$, can be constructed in $O(n^{d'(r^2+r)})$ time. Furthermore, for each $t \in T$, S^t can be computed in $O(d'nr^2 + n \log n)$ time.*

Proof of claim. For every two distinct indices $j, j' \in [n]$, the hyperplane

$$H_{j,j'} := \{z \in \mathcal{E} \ : \ \ell_j(z) = \ell_{j'}(z)\} \tag{8}$$

partitions all points $z \in \mathcal{E}$ based on which of the two values $\ell_j(z)$ and $\ell_{j'}(z)$ is larger. By considering the hyperplane $H_{j,j'}$ for all distinct pairs of indices $j, j' \in [n]$, we obtain a set $\mathcal{H}$ of $(n^2 - n)/2 \le n^2$ hyperplanes in $\mathcal{E}$. By the hyperplane arrangement theorem, the arrangement of $\mathcal{H}$ consists of $O((n^2)^{\dim \mathcal{E}}) = O(n^{d'(r^2+r)})$ full-dimensional polyhedra, and can be constructed in $O(n^{d'(r^2+r)})$ time. We denote by P^t, for $t \in T$, the polyhedra in the arrangement, where T is a finite index set of cardinality $O(n^{d'(r^2+r)})$. From the definition of the hyperplanes (8) we have that, if for some $t \in T$ there exists a vector $z^t \in P^t$ that satisfies $\ell_j(z^t) > \ell_{j'}(z^t)$ for two distinct indices $j, j' \in [n]$, then every vector $z \in P^t$ must satisfy $\ell_j(z) \ge \ell_{j'}(z)$.

Next, we explain how the index sets S^t, for $t \in T$, are constructed. To do so, we fix one polyhedron P^t, for some $t \in T$, until the end of the proof of the claim. The hyperplane arrangement theorem also returns explicitly a vector z^t in the interior of P^t [13]. We then compute $\ell_j(z^t)$ for every $j \in [n]$ in time $O(d'nr^2)$. Since z^t is in the interior of P^t, in time $O(n \log n)$ we can find an ordering $j_1^t, j_2^t, \dots, j_n^t$ of the indices $1, \dots, n$ such that

$$\ell_{j_1^t}(z^t) > \ell_{j_2^t}(z^t) > \cdots > \ell_{j_n^t}(z^t).$$

From the property of the polyhedra in the arrangement we have that, for every z with $z \in P^t$,

$$\ell_{j_1^t}(z) \ge \ell_{j_2^t}(z) \ge \cdots \ge \ell_{j_n^t}(z).$$

In particular, for every Y with $\text{ext}(Y) \in P^t$, we have

$$\ell_{j_1^t}(\text{ext}(Y)) \ge \ell_{j_2^t}(\text{ext}(Y)) \ge \cdots \ge \ell_{j_n^t}(\text{ext}(Y)),$$

thus

$$\|R_{j_1^t} Y\|_2^2 \ge \|R_{j_2^t} Y\|_2^2 \ge \cdots \ge \|R_{j_n^t} Y\|_2^2.$$

Claim 2 then implies that for each Y such that $\text{ext}(Y) \in P^t$, the set $S^t := \{j_1^t, j_2^t, \dots, j_s^t\}$ is an optimal solution to Problem $\text{RST}(Y)$. $\diamond$

Let $\mathcal{S}$ be the family of all index sets S^t obtained in Claim 3, namely

$$\mathcal{S} := \{S^t\}_{t \in T}.$$

Claim 4. *The family $\mathcal{S}$ contains an optimal support of Problem* (7).

Proof of claim. Let (S^*, Y^*) be an optimal solution to Problem (7). Then S^* is an optimal solution to the restricted Problem RST(Y^*). Let P^t, for $t \in T$, be a polyhedron such that $\text{ext}(Y^*) \in P^t$, and let $S^t \in \mathcal{S}$ be the corresponding index set. From Claim 3, S^t is an optimal solution to Problem RST(Y^*). This implies that the solution (S^t, Y^*) is also optimal to Problem (7). $\diamond$

Claim 4 implies that, in order to find an optimal support of Problem (7), it suffices to solve the $|T|$ optimization problems

$$\max_{\substack{Y \in \mathbb{R}^{r \times d'} \\ Y^\mathsf{T} Y = I_{d'}}} \|R_{S^t} Y\|_F^2 \qquad \forall t \in T. \tag{9}$$

In fact, an index set S^t, for $t \in T$, which yields the maximum optimal value among Problems (9) is then an optimal support of Problem (7). Each Problem (9) is a PCA problem with an $r \times r$ input matrix. In particular, the input matrix $R_{S^t}^\mathsf{T} R_{S^t}$ can be constructed in $O(sr^2)$ time and an optimal solution can be found in $O(r^3)$ time. This completes the description of the algorithm and the proof of its correctness.

Next, we analyze the total running time of the algorithm presented. The matrix R is computed in $O(n^3)$ time, the linear functions ℓ_j, for $j \in [n]$, are obtained in $O(d'nr^2)$ time, the polyhedra P^t, for $t \in T$, are constructed $O(|T|)$ time, the sets S^t, for $t \in T$, are computed in $O(|T|(d'nr^2 + n \log n))$ time, the $|T|$ PCA Problems (9) are solved in $O(|T|(sr^2 + r^3))$ time, and the PCA Problem (5) is solved in $O(s^2 r + s^3)$ time. The total running time is therefore

$$O\left(|T|(d'nr^2 + n \log n)\right) = O\left(n^{d'(r^2+r)}(d'nr^2 + n \log n)\right).$$

This concludes the proof of Theorem 1. $\square$

4 The maximum-profit integer circulation problem

In the proof of Theorem 2 we will consider the maximum-profit integer circulation problem. Hence, before proceeding with the proof, we give a brief overview of this problem and we present optimality conditions and a strongly polynomial-time algorithm to solve it.

Let $D = (V, A)$ be a directed graph. A vector $f \in \mathbb{R}^A$ is called a *circulation* if $f(\delta^{\text{in}}(v)) = f(\delta^{\text{out}}(v))$ for each vertex $v \in V$, where $\delta^{\text{in}}(v) = \{wv \in A\}$ and $\delta^{\text{out}}(v) = \{vw \in A\}$. A circulation f is said to be *integer* if f has all integer entries. In a *maximum-profit integer circulation problem* we are given a directed graph $D = (V, A)$, arc capacities $u \in \mathbb{Z}_+^A$, and arc profits $p \in \mathbb{Q}^A$. We say that f is a *feasible circulation* if f is an integer circulation in the directed graph D subject to $0 \le f \le u$. The *profit* of a feasible circulation f is $p^\mathsf{T} f$. The goal of the maximum-profit integer circulation problems is that of finding an *optimal circulation,* which is a feasible circulation of maximum profit. We refer the reader to Chapters 11 and 12 in [28] for a thorough presentation of circulations problems. We refer the reader to the same book [28] for standard graph theory definitions including that of directed circuit and undirected circuit.

To state the optimality conditions for a maximum-profit integer circulation problem, it will be useful to consider the *residual directed graph* $D_f = (V, A_f)$ of a circulation f, where

$$A_f := \{a \ : \ a \in A, \ f_a < u_a\} \cup \{a^{\leftarrow} \ : \ a \in A, \ f_a > 0\}.$$

Here $a^{\leftarrow} := wu$ if $a = uw$. For a directed circuit C in D_f, we define $\chi^C \in \{0, \pm 1\}^A$ by:

$$\chi^C_a := \begin{cases} 1 & \text{if } C \text{ traverses } a, \\ -1 & \text{if } C \text{ traverses } a^{\leftarrow}, \\ 0 & \text{if } C \text{ traverses neither } a \text{ nor } a^{\leftarrow}. \end{cases}$$

We then define, for every directed circuit C in D_f its *profit* as

$$p(C) = \sum_{a \in A} \chi^C_a p_a.$$

We are now ready to state the optimality conditions, which follow, for example, from Theorem 12.1 in [28].

Proposition 1. *A feasible circulation f is optimal if and only if each directed circuit in D_f has nonpositive profit.*

The above optimality conditions are at the basis of Goldberg and Tarjan's strongly polynomial-time algorithm to solve the maximum-profit integer circulation problem [14, 15]. We refer the reader to Section 12.3 in [28] for a description of the algorithm.

Proposition 2 (Corollary 12.2a in [28])**.** *An optimal circulation can be found in $O(|V|^2|A|^3 \log |V|)$ time.*

5 Proof of Theorem 2

Consider SPCA-DS where the input matrix $K \in \mathbb{R}^{n \times n}$ has rank r. Since the matrix K is positive semidefinite, we can compute an $n \times r$ matrix R such that $K = RR^\mathsf{T}$ in $O(n^3)$ time, for example using the Cholesky decomposition with complete pivoting. The objective function of SPCA-DS can then be written as $\operatorname{trace}(X^\mathsf{T} K X) = \|R^\mathsf{T} X\|_F^2 = \sum_{i=1}^d \|R^\mathsf{T} x_i\|_2^2$ and SPCA-DS takes the form

$$\max_{X \in \mathcal{X}} \sum_{i=1}^d \|R^\mathsf{T} x_i\|_2^2. \tag{10}$$

In this proof we use some of the notation introduced in the proof of Theorem 1. Namely, for $j \in [n]$, R_j denotes the jth row of R and, for $S \subseteq [n]$, R_S denotes the $|S| \times r$ submatrix of R containing only the rows indexed by S.

As discussed in Section 1.2, the main difficulty in solving Problem (10) consists in finding an optimal support $\{S_i^*\}_{i\in[d]}$ of Problem (10). In fact, once $\{S_i^*\}_{i\in[d]}$ is determined, each optimal vector x_i^*, for $i \in [d]$, can be obtained by setting to zero the entries of x_i^* with indices not in S_i^*, while the other entries of x_i^* can be obtained by solving the optimization problem

$$\max_{\substack{x_i \in \mathbb{R}^{|S_i^*|} \\ \|x_i\|_2 = 1}} \|R_{S_i^*}^{\mathsf{T}} x_i\|_2^2. \tag{11}$$

This is a single-component PCA problem with an input matrix of dimension at most $s \times s$. In particular, the input matrix $R_{S_i^*} R_{S_i^*}^{\mathsf{T}}$ can be constructed in $O(s^2 r)$ time and an optimal solution can be found in $O(s^3)$ time. Based on this discussion, in the remainder of the proof it suffices to find an optimal support $\{S_i^*\}_{i\in[d]}$ of Problem (10).

The next claim uses Lemma 1 to replace each vector of variables $x_i \in \mathbb{R}^n$ in Problem (10) with a vector of variables $y_i \in \mathbb{R}^r$. In the claim we consider the following two optimization problems:

$$\max_{\substack{S_i \subseteq [n], |S_i| \le s,\ \forall i \in [d] \\ S_i \cap S_{i'} = \emptyset,\ \forall i \ne i' \in [d]}} \quad \max_{\substack{x_i \in \mathbb{R}^{|S_i|}, \|x_i\|_2 = 1, \\ \forall i \in [d]}} \sum_{i=1}^{d} \|(R_{S_i})^{\mathsf{T}} x_i\|_2^2, \tag{12}$$

$$\max_{\substack{S_i \subseteq [n], |S_i| \le s,\ \forall i \in [d] \\ S_i \cap S_{i'} = \emptyset,\ \forall i \ne i' \in [d]}} \quad \max_{\substack{y_i \in \mathbb{R}^{r}, \|y_i\|_2 = 1, \\ \forall i \in [d]}} \sum_{i=1}^{d} \|R_{S_i} y_i\|_2^2. \tag{13}$$

We say that $\{S_i^*\}_{i\in[d]}$ is an *optimal support* of Problem (12) if there exist x_i^*, for $i \in [d]$, such that $\{(S_i^*, x_i^*)\}_{i\in[d]}$ is an optimal solution to Problem (12). Similarly, we say that $\{S_i^*\}_{i\in[d]}$ is an *optimal support* of Problem (13) if there exist y_i^*, for $i \in [d]$, such that $\{(S_i^*, y_i^*)\}_{i\in[d]}$ is an optimal solution to Problem (13).

Claim 5. *The optimal supports of Problems* (10)*,* (12)*,* (13) *coincide.*

Proof of claim. Lemma 1, applied d times with $M := R_{S_i}$, for $i \in [d]$, implies that the optimal supports of Problems (12) and (13) coincide. Thus we only need to show that the optimal supports of Problems (10) and (12) coincide. To do so, it suffices to prove the following two statements: (i) For every feasible solution $\{(S_i, x_i)\}_{i\in[d]}$ to Problem (12) with objective function value γ, there is a feasible solution $\{\tilde{x}_i\}_{i\in[d]}$ to Problem (10) with objective function value γ such that $\operatorname{supp}(\tilde{x}_i) \subseteq S_i$ $\forall i \in [d]$; (ii) For every feasible solution $\{\tilde{x}_i\}_{i\in[d]}$ to Problem (10) with objective function value γ, there is a feasible solution $\{(S_i, x_i)\}_{i\in[d]}$ to Problem (12) with objective function value γ such that $\operatorname{supp}(\tilde{x}_i) \subseteq S_i$ $\forall i \in [d]$.

(i). Let $\{(S_i, x_i)\}_{i\in[d]}$ be a feasible solution to Problem (12) with objective function value γ. For each $i \in [d]$, let $\tilde{x}_i \in \mathbb{R}^n$ be obtained from x_i by adding zero entries corresponding to the indices not in S_i. Then $\{\tilde{x}_i\}_{i\in[d]}$ is a feasible solution to Problem (10) with objective function value γ such that $\text{supp}(\tilde{x}_i) \subseteq S_i \ \forall i \in [d]$.

(ii). Let $\{\tilde{x}_i\}_{i\in[d]}$ be a feasible solution to Problem (10) with objective function value γ. Let $S_i := \text{supp}(\tilde{x}_i)$, for every $i \in [d]$. Let x_i be obtained from $\tilde{x}_i$ by dropping the (zero) entries with indices not in S_i. Then $\{(S_i, x_i)\}_{i\in[d]}$ is a feasible solution to Problem (12) with objective function value γ such that $\text{supp}(\tilde{x}_i) \subseteq S_i \ \forall i \in [d]$. $\diamond$

Due to Claim 5, in the rest of the proof our goal will be finding an optimal support of Problem (13).

5.1 The restricted problem

In this section we study the restricted version of Problem (13) obtained by fixing the d vectors of variables $y_i \in \mathbb{R}^r$, for $i \in [d]$. We denote this restricted problem by $\text{RST}(\{y_i\}_{i\in[d]})$, and formally define it as

$$\max_{\substack{S_i\subseteq[n], |S_i|\le s, \ \forall i\in[d] \\ S_i\cap S_{i'}=\emptyset, \ \forall i\neq i'\in[d]}} \sum_{i=1}^{d} \|R_{S_i} y_i\|_2^2.$$

Our next goal is to provide a characterization of the optimal solutions to Problem $\text{RST}(\{y_i\}_{i\in[d]})$ based on a maximum-profit integer circulation problem. We refer the reader to Section 4 for a brief introduction to the maximum-profit integer circulation problem.

In the remainder of the proof, we denote by $D = (V, A)$ the directed graph with vertices $V = U \cup W \cup \{t\}$, where $U = \{u_1, \ldots, u_d\}$, $W = \{w_1, \ldots, w_n\}$, and with arcs $A = A_0 \cup A_U \cup A_W$, where $A_0 = \{u_i w_j : i \in [d], j \in [n]\}$, $A_U = \{tu_i : i \in [d]\}$, $A_W = \{w_j t : j \in [n]\}$. The directed graph D is depicted in Figure 1. We define arc

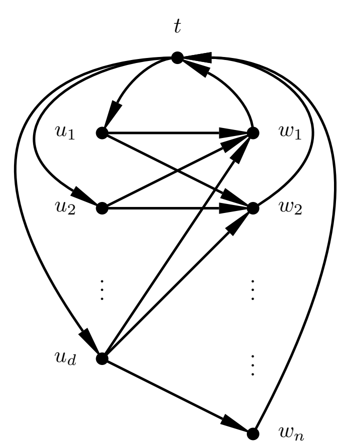

Figure 1: The directed graph $D = (V, A)$ considered in Section 5.1.

capacities $u \in \mathbb{Z}_+^A$ as $u_a := 1$ if $a \in A_0 \cup A_W$, $u_a := s$ if $a \in A_U$. We also define arc profits $p \in \mathbb{Q}^A$ by $p_a := (R_j y_i)^2$ if $a = u_i w_j \in A_0$, $p_a := 0$ if $a \in A_U \cup A_W$. We then define Problem $\mathrm{CRC}(\{y_i\}_{i\in[d]})$ as the maximum-profit integer circulation problem on the directed graph $D = (V, A)$, with arc capacities u and arc profits p. We remark that in Problem $\mathrm{CRC}(\{y_i\}_{i\in[d]})$, only the arc profits depend on $\{y_i\}_{i\in[d]}$. The next claim provides a characterization of the optimal solutions to Problem $\mathrm{RST}(\{y_i\}_{i\in[d]})$ in terms of optimal circulations to Problem $\mathrm{CRC}(\{y_i\}_{i\in[d]})$.

Claim 6. *Let $\{y_i\}_{i\in[d]}$ be given. Then $\{S_i^*\}_{i\in[d]}$ is an optimal solution to Problem* $\mathrm{RST}(\{y_i\}_{i\in[d]})$ *if and only if $S_i^* := \{j \in [n] : f^*_{u_i w_j} = 1\}$, for $i \in [d]$, where f^* is an optimal circulation to Problem* $\mathrm{CRC}(\{y_i\}_{i\in[d]})$.

Proof of claim. To prove the claim, it suffices to prove the following two statements: (i) For every feasible solution $\{S_i\}_{i\in[d]}$ to Problem $\mathrm{RST}(\{y_i\}_{i\in[d]})$ with objective function value γ, there is a feasible circulation f to $\mathrm{CRC}(\{y_i\}_{i\in[d]})$ with profit γ such that $S_i = \{j \in [n] : f_{u_i w_j} = 1\}$ $\forall i \in [d]$; (ii) For every feasible circulation f to $\mathrm{CRC}(\{y_i\}_{i\in[d]})$ with profit γ, the solution $\{S_i\}_{i\in[d]}$ defined by $S_i := \{j \in [n] : f_{u_i w_j} = 1\}$ $\forall i \in [d]$, is feasible to Problem $\mathrm{RST}(\{y_i\}_{i\in[d]})$ and has objective function value γ. In the following, we first discuss the mapping between solutions and circulations in (i) and (ii), and then we discuss the correspondence of objective function values and profits in both (i) and (ii).

(i). Let $\{S_i\}_{i\in[d]}$ be a feasible solution to Problem $\mathrm{RST}(\{y_i\}_{i\in[d]})$, i.e., $S_i \subseteq [n]$, $|S_i| \le s$, $\forall i \in [d]$, and $S_i \cap S_{i'} = \emptyset$, $\forall i \neq i' \in [d]$. For every pair i, j such that $j \in S_i$, define $f_{u_i w_j} := 1$, $f_{w_j t} := 1$, and set $f_a := 0$ for every other $a \in A_0 \cup A_W$. For every $i \in [d]$, define $f_{t u_i} := |S_i|$. It can be easily checked that f is a feasible circulation to $\mathrm{CRC}(\{y_i\}_{i\in[d]})$ such that $S_i = \{j \in [n] : f_{u_i w_j} = 1\}$ $\forall i \in [d]$.

(ii). Viceversa, let f be a feasible circulation to $\mathrm{CRC}(\{y_i\}_{i\in[d]})$. Since $u_a = 1$ for every $a \in A_0$, we have $f_a \in \{0, 1\}$ for every $a \in A_0$. For every $i \in [d]$, define $S_i := \{j \in [n] : f_{u_i w_j} = 1\}$. $u_a = 1$ for every $a \in A_W$ implies that no $j \in [n]$ is in more than one set S_i, thus $S_i \cap S_{i'} = \emptyset$, $\forall i \neq i' \in [d]$. Since $u_a = s$ for every $a \in A_U$, we also have $|S_i| \le s$ for every $i \in [d]$. Therefore, $\{S_i\}_{i\in[d]}$ is a feasible solution to Problem $\mathrm{RST}(\{y_i\}_{i\in[d]})$.

The claim follows since objective function values and profits coincide in both mappings (i) and (ii):

$$p^\mathsf{T} f = \sum_{i=1}^{d} \sum_{j=1}^{n} p_{u_i w_j} f_{u_i w_j} = \sum_{i=1}^{d} \sum_{j \in S_i} p_{u_i w_j} = \sum_{i=1}^{d} \sum_{j \in S_i} (R_j y_i)^2 = \sum_{i=1}^{d} \|R_{S_i} y_i\|_2^2. \qquad \diamond$$

5.2 A polynomial arrangement

Claim 6 implies that in order to find an optimal solution to Problem $\mathrm{RST}(\{y_i\}_{i\in[d]})$, it is sufficient to find an optimal circulation to Problem $\mathrm{CRC}(\{y_i\}_{i\in[d]})$. Thus we now focus on the latter problem. The optimality conditions stated in Proposition 1 imply that in order to understand an optimal circulation to Problem $\mathrm{CRC}(\{y_i\}_{i\in[d]})$, it is important to understand the sign of the profits of all directed circuits in D_f, for any feasible circulation f. Note that any directed circuit C in D_f, for a feasible circulation f, gives an undirected circuit C' in D. For an undirected circuit C' in D, we define $\chi^{C'} \in \{0, \pm 1\}^A$ by:

$$\chi_a^{C'} := \begin{cases} 1 & \text{if } C' \text{ traverses } a \text{ forward,} \\ -1 & \text{if } C' \text{ traverses } a \text{ backward,} \\ 0 & \text{if } C' \text{ does not traverse } a. \end{cases}$$

We then define, for every undirected circuit C' in D, its *profit* as

$$p(C') = \sum_{a \in A} \chi_a^{C'} p_a.$$

In this way we obtain that, if a directed circuit C in D_f, for some feasible circulation f, gives the undirected circuit C' in D, then we have $p(C) = p(C')$. From the above discussion, in order to understand the sign of the profits of all directed circuits in D_f, for any feasible circulation f, it suffices to understand the signs of the profits of all undirected circuits in D. From now on, we denote by $\mathcal{C}$ the set of undirected circuits in D. The structure of the directed graph D implies that each undirected circuit in $\mathcal{C}$ can contain at most d vertices in W. Thus we obtain $|\mathcal{C}| = O((dn)^d)$.

Our next task is to partition the dr-dimensional space of all d vectors $\{y_i\}_{i \in [d]}$, where each y_i is in $\mathbb{R}^r$, based on the sign of the values $p(C')$, for every $C' \in \mathcal{C}$, that they yield. Each $p(C')$, for $C' \in \mathcal{C}$, can be written as a linear function of arc profits

$$p(C') = \sum_{a \in A} \chi_a^{C'} p_a = \sum_{u_i w_j \in A_0} \chi_{u_i w_j}^{C'} p_{u_i w_j}.$$

Each arc profit $p_{u_i w_j} = (R_j y_i)^2$, for $i \in [d]$, $j \in [n]$, is a quadratic polynomial in the entries of the vector y_i, and every monomial is a constant times the product of two variables in the vector y_i, i.e., $(y_i)_k (y_i)_{k'}$, for $k, k' \in [r]$. Since we wish to obtain a polyhedral partition, we introduce a new space of variables that allows us to write each $p(C')$, for $C' \in \mathcal{C}$, as a linear function. Formally, we define the space $\mathcal{E}$ that contains one variable for each $(y_i)_k (y_i)_{k'}$, for $i \in [d]$, $k, k' \in [r]$. The dimension of the space $\mathcal{E}$ is therefore $d \cdot (r^2 + r)/2$. Note that, for every d vectors $\{y_i\}_{i \in [d]}$, where each y_i is in $\mathbb{R}^r$, there exists a unique corresponding point in $\mathcal{E}$, that we denote by $\operatorname{ext}(\{y_i\}_{i \in [d]})$, obtained by computing all the products $(y_i)_k (y_i)_{k'}$, for $i \in [d]$, $k, k' \in [r]$. For each arc $u_i w_j$, $i \in [d]$, $j \in [n]$, we can now write in time $O(r^2)$ a linear function $\ell_{u_i w_j} : \mathcal{E} \to \mathbb{R}$ such that $\ell_{u_i w_j}(\operatorname{ext}(\{y_i\}_{i \in [d]})) = (R_j y_i)^2$ for every $\{y_i\}_{i \in [d]}$. As a consequence, for each $C' \in \mathcal{C}$, we can write a linear function $\ell_{C'} : \mathcal{E} \to \mathbb{R}$ such that $\ell_{C'}(\operatorname{ext}(\{y_i\}_{i \in [d]})) = p(C')$ for every $\{y_i\}_{i \in [d]}$. Note that all these linear functions can be constructed in time $O(dnr^2 + dr^2|\mathcal{C}|) = O(dnr^2 + d^{d+1} r^2 n^d)$.

Claim 7. *There exist a finite index set T of cardinality $O((dn)^{d^2(r^2+r)/2})$, full-dimensional polyhedra $P^t \subseteq \mathcal{E}$, for $t \in T$, that cover $\mathcal{E}$, and index sets $\{S_i^t\}_{i \in [d]}$, for $t \in T$, with the following property: For every $t \in T$, and for every $\{y_i\}_{i \in [d]}$ such that $\operatorname{ext}(\{y_i\}_{i \in [d]}) \in P^t$, $\{S_i^t\}_{i \in [d]}$ is an optimal solution to Problem $\operatorname{RST}(\{y_i\}_{i \in [d]})$. The polyhedra P^t, for $t \in T$, can be constructed in $O((dn)^{d^2(r^2+r)/2})$ time. Furthermore, for each $t \in T$, $\{S_i^t\}_{i \in [d]}$ can be computed in $O(dnr^2 + d^3 n^5 \log n)$ time.*

Proof of claim. For every $C' \in \mathcal{C}$, the hyperplane

$$H_{C'} := \{z \in \mathcal{E} \ : \ \ell_{C'}(z) = 0\} \tag{14}$$

partitions all points $z \in \mathcal{E}$ based on the sign of $\ell_{C'}(z)$. By considering the hyperplane $H_{C'}$ for all $C' \in \mathcal{C}$, we obtain a set $\mathcal{H}$ of $|\mathcal{C}| = O((dn)^d)$ hyperplanes in $\mathcal{E}$. By the hyperplane arrangement theorem, the arrangement of $\mathcal{H}$ consists of $O((dn)^{d \cdot \dim \mathcal{E}}) = O((dn)^{d^2(r^2+r)/2})$ full-dimensional polyhedra, and can be constructed in $O((dn)^{d^2(r^2+r)/2})$ time. We denote by P^t, for $t \in T$, the polyhedra in the arrangement, where T is a finite index set of cardinality $O((dn)^{d^2(r^2+r)/2})$. From the definition of the hyperplanes (14) we have that, if for some $t \in T$ there exists a vector $z^t \in P^t$ that satisfies $\ell_{C'}(z^t) < 0$ for some $C' \in \mathcal{C}$, then every vector $z \in P^t$ must satisfy $\ell_{C'}(z) \leq 0$.

Next, we explain how the index sets $\{S_i^t\}_{i \in [d]}$, for $t \in T$, are constructed. To do so, we fix one polyhedron P^t, for some $t \in T$, until the end of the proof of the claim. Due to Claim 6, it suffices to show that we can construct a circulation f that is an optimal circulation to every Problem CRC($\{y_i\}_{i \in [d]}$) for all $\{y_i\}_{i \in [d]}$ with $\text{ext}(\{y_i\}_{i \in [d]}) \in P^t$. To obtain this optimal circulation we will use a vector z^t in the interior of P^t, which is returned explicitly by the hyperplane arrangement theorem [13]. Then, we define Problem CRC(z^t) as the problem obtained from Problem CRC($\{y_i\}_{i \in [d]}$) for any $\{y_i\}_{i \in [d]}$ with $\text{ext}(\{y_i\}_{i \in [d]}) \in P^t$, by replacing the arc profits with the one induced by z^t. Precisely, Problem CRC(z^t) is the maximum-profit integer circulation problem on the directed graph $D = (V, A)$ defined in Section 5.1, with arc capacities $u \in \mathbb{Z}_+^A$ defined in Section 5.1, and arc profits $p^t \in \mathbb{Q}^A$ defined by $p_a^t := \ell_{u_i w_j}(z^t)$ if $a = u_i w_j \in A_0$, $p_a^t := 0$ if $a \in A_U \cup A_W$. Note that these arc profits can be computed in time $O(dnr^2)$.

From Proposition 2, an optimal circulation f^* to Problem CRC(z^t) can be found in $O(|V|^2 |A|^3 \log |V|)$ time. Since $|V| = O(n)$ and $|A| = O(dn)$, we can obtain f^* in $O(d^3 n^5 \log n)$ time. We now show that f^* is an optimal circulation to every Problem CRC($\{y_i\}_{i \in [d]}$) for all $\{y_i\}_{i \in [d]}$ with $\text{ext}(\{y_i\}_{i \in [d]}) \in P^t$. So we fix an arbitrary $\{\bar{y}_i\}_{i \in [d]}$ with $\text{ext}(\{\bar{y}_i\}_{i \in [d]}) \in P^t$. In the remainder of the proof we will denote by p^t the profits in Problem CRC(z^t) and by $\bar{p}$ the profits in Problem CRC($\{\bar{y}_i\}_{i \in [d]}$). Since f^* is a feasible circulation to Problem CRC(z^t), it is also a feasible circulation to Problem CRC($\{\bar{y}_i\}_{i \in [d]}$). This is because the two problems share the same directed graph and the same arc capacities. Furthermore, the residual directed graph D_{f^*} is the same in both problems. From the optimality conditions stated in Proposition 1, we know that $p^t(C) \leq 0$ for every directed circuit C in D_{f^*}. From the definition of the hyperplanes (14) and the fact that z^t is in the interior of P^t, we obtain that $p^t(C) < 0$ for every directed circuit C in D_{f^*}. Since $\text{ext}(\{\bar{y}_i\}_{i \in [d]}) \in P^t$, we then have $\bar{p}(C) \leq 0$ for every directed circuit C in D_{f^*}. Again from the optimality conditions in Proposition 1, we obtain that f^* is an optimal circulation to Problem CRC($\{\bar{y}_i\}_{i \in [d]}$). We have thereby shown that f^* is an optimal circulation to every Problem CRC($\{y_i\}_{i \in [d]}$) for all $\{y_i\}_{i \in [d]}$ with $\text{ext}(\{y_i\}_{i \in [d]}) \in P^t$. An optimal solution to all Problems RST($\{y_i\}_{i \in [d]}$) for all $\{y_i\}_{i \in [d]}$ with $\text{ext}(\{y_i\}_{i \in [d]}) \in P^t$ can then be obtained as described in Claim 6. The total running time to compute $\{S_i^t\}_{i \in [d]}$ is $O(dnr^2 + d^3 n^5 \log n)$ $\diamond$

Let $\mathcal{S}$ be the family of all index sets $\{S_i^t\}_{i \in [d]}$ obtained in Claim 7, namely

$$\mathcal{S} := \{\{S_i^t\}_{i \in [d]}\}_{t \in T}.$$

Claim 8. *The family $\mathcal{S}$ contains an optimal support of Problem* (13)*.*

Proof of claim. Let $\{(S_i^*, y_i^*)\}_{i\in[d]}$ be an optimal solution to Problem (13). Then $\{S_i^*\}_{i\in[d]}$ is an optimal solution to the restricted Problem $\mathrm{RST}(\{y_i^*\}_{i\in[d]})$. Let P^t, for $t \in T$, be a polyhedron such that $\mathrm{ext}(\{y_i^*\}_{i\in[d]}) \in P^t$, and let $\{S_i^t\}_{i\in[d]} \in \mathcal{S}$ be the corresponding index sets. From Claim 7, $\{S_i^t\}_{i\in[d]}$ is an optimal solution to Problem $\mathrm{RST}(\{y_i^*\}_{i\in[d]})$. This implies that the solution $\{S_i^t, y_i^*\}_{i\in[d]}$ is also optimal to Problem (13). ◇

Claim 8 implies that, in order to find an optimal support of Problem (13), it suffices to solve the $|T|$ optimization problems

$$\max_{\substack{y_i\in\mathbb{R}^r, \|y_i\|_2=1,\\ \forall i\in[d]}} \sum_{i=1}^{d} \|R_{S_i^t} y_i\|_2^2 \qquad \forall t \in T. \tag{15}$$

In fact, a $\{S_i^t\}_{i\in[d]}$, for $t \in T$, which yields the maximum optimal value among Problems (15) is then an optimal support of Problem (13). Each Problem (15) can be decomposed into the d optimization problems

$$\max_{\substack{y_i\in\mathbb{R}^r\\ \|y_i\|_2=1}} \|R_{S_i^t} y_i\|_2^2 \qquad \forall i \in [d]. \tag{16}$$

Each Problem (16) is a single-component PCA problem with an $r \times r$ input matrix. In particular, the input matrix $R_{S_i^t}^\mathsf{T} R_{S_i^t}$ can be constructed in $O(sr^2)$ time and an optimal solution can be found in $O(r^3)$ time. This completes the description of the algorithm and the proof of its correctness.

Next, we analyze the total running time of the algorithm presented. The matrix R is computed in $O(n^3)$ time, the linear functions $\ell_{C'}$, for $C' \in \mathcal{C}$, are constructed in $O(dnr^2 + d^{d+1}r^2n^d)$ time, the polyhedra P^t, for $t \in T$, are constructed $O(|T|)$ time, the sets $\{S_i^t\}_{i\in[d]}$, for $t \in T$, are computed in $O(|T|(dnr^2 + d^3n^5 \log n))$ time, the $|T|d$ PCA Problems (16) are solved in $O(|T|d(sr^2 + r^3))$ time, and the d PCA Problems (11) are solved in $O(d(s^2r + s^3))$ time. The total running time is therefore

$$O\left(|T|(dnr^2 + d^3n^5 \log n)\right) = O\left((dn)^{d^2(r^2+r)/2}(dnr^2 + d^3n^5 \log n)\right).$$

This concludes the proof of Theorem 2. □

References

[1] Asteris, M., Papailiopoulos, D., Karystinos, G.: Sparse principal component of a rank-deficient matrix. In: Proceedings of ISIT (2011)

[2] Asteris, M., Papailiopoulos, D., Karystinos, G.: The sparse principal component of a constant-rank matrix. IEEE Transactions on Information Theory pp. 2281–2290 (2014)

[3] Asteris, M., Papailiopoulos, D., Kyrillidis, A., Dimakis, A.: Sparse PCA via bipartite matchings. In: Proceedings of NIPS (2015)

[4] Berk, L., Bertsimas, D.: Certifiably optimal sparse principal component analysis. Mathematical Programming Computation **11**, 381–420 (2019)

[5] Bertsimas, D., Tsitsiklis, J.: Introduction to Linear Optimization. Athena Scientific, Belmont, MA (1997)

[6] Boutsidis, C., Drineas, P., Magdon-Ismail, M.: Sparse features for PCA-like linear regression. In: Proceedings of NIPS, pp. 2285–2293 (2011)

[7] Cadima, J., Jolliffe, I.: Loading and correlations in the interpretation of principle compenents. Journal of Applied Statistics **22**(2), 203–214 (1995)

[8] Chan, S., Papailiopoulos, D., Rubinstein, A.: On the worst-case approximability of sparse PCA. Proceedings of COLT (2016)

[9] d'Aspremont, A., Bach, F., Ghaoui, L.: Optimal solutions for sparse principal component analysis. The Journal of Machine Learning Research **9**, 1269–1294 (2008)

[10] d'Aspremont, A., Bach, F., Ghaoui, L.: Approximation bounds for sparse principal component analysis. Mathematical Programming, Series B pp. 89–110 (2014)

[11] d'Aspremont, A., El Ghaoui, L., Jordan, M., Lanckriet, G.: A direct formulation for sparse PCA using semidefinite programming. SIAM review **49**(3), 434–448 (2007)

[12] Dey, S., Mazumder, R., Wang, G.: A convex integer programming approach for optimal sparse PCA. arXiv preprint arXiv:1810.09062 (2018)

[13] Edelsbrunner, H., O'Rourke, J., Seidel, R.: Constructing arrangements of lines and hyperplanes with applications. SIAM Journal on Computing **15**(2), 341–363 (1986)

[14] Goldberg, A., Tarjan, R.: Finding minimum-cost circulations by canceling negative cycles. In: Proceedings of STOC, pp. 388–397 (1988)

[15] Goldberg, A., Tarjan, R.: Finding minimum-cost circulations by canceling negative cycles. Journal of the Association for Computing Machinery **36**, 873–886 (1989)

[16] Hastie, T., Tibshirani, R., Wainwright, M.: Statistical learning with sparsity. CRC press (2015)

[17] He, Y., Monteiro, R., Park, H.: An efficient algorithm for rank-1 sparse PCA. working paper (2010)

[18] Jolliffe, I., Trendafilov, N., Uddin, M.: A modified principal component technique based on the lasso. Journal of Computational and Graphical Statistics **12**(3), 531–547 (2003)

[19] Journée, M., Nesterov, Y., Richtárik, P., Sepulchre, R.: Generalized power method for sparse principal component analysis. The Journal of Machine Learning Research **11**, 517–553 (2010)

[20] Karystinos, G., Liavas, A.: Efficient computation of the binary vector that maximizes a rank-deficient quadratic form. IEEE Transactions on Information Theory **56**(7), 3581–3593 (2010)

[21] Karystinos, G., Pados, D.: Rank-2-optimal adaptive design of binary spreading codes. IEEE Transactions on Information Theory **53**(9), 3075–3080 (2007)

[22] Mackenthun, K.: A fast algorithm for multiple-symbol differential detection of MPSK. IEEE Transactions on Communications **42**(2/3/4), 1471–1474 (1994)

[23] Mackey, L.: Deflation methods for sparse PCA. In: Proceedings of NIPS, vol. 21, pp. 1017–1024 (2009)

[24] Magdon-Ismail, M.: NP-hardness and inapproximability of sparse PCA. Information Processing Letters pp. 35–38 (2017)

[25] Moghaddam, B., Weiss, Y., Avidan, S.: Spectral bounds for sparse PCA: Exact and greedy algorithms. In: Proceedings of NIPS, vol. 18, p. 915 (2006)

[26] Motedayen, I., Krishnamoorthy, A., Anastasopoulos, A.: Optimal joint detection/estimation in fading channels with polynomial complexity. IEEE Transactions on Information Theory **53**(1), 209–223 (2007)

[27] Papailiopoulos, D., Dimakis, A., Korokythakis, S.: Sparse PCA through low-rank approximations. In: Proceedings of ICML (2013)

[28] Schrijver, A.: Combinatorial Optimization. Polyhedra and Efficiency. Springer-Verlag, Berlin (2003)

[29] Shalev-Shwartz, S., Ben-David, S.: Understanding Machine Learning. Cambridge University Press (2014)

[30] Sigg, C., Buhmann, J.: Expectation-maximization for sparse and non-negative PCA. In: Proceedings of ICML, pp. 960–967 (2008)

[31] Sweldens, W.: Fast block noncoherent decoding. IEEE Communications Letters **5**(4), 132–134 (2001)

[32] Vu, V., Lei, J.: Minimax rates of estimation for sparse PCA in high dimensions. In: Proceedings of AIStats, pp. 1278–1286 (2012)

[33] Yuan, X., Zhang, T.: Truncated power method for sparse eigenvalue problems. Journal of Machine Learning Research **14**, 899–925 (2013)

[34] Zhang, Y., d'Aspremont, A., L., G.: Sparse PCA: Convex relaxations, algorithms and applications. In: Handbook on Semidefinite, Conic and Polynomial Optimization, pp. 915–940. Springer (2012)

[35] Zou, H., Hastie, T., Tibshirani, R.: Sparse principal component analysis. Journal of computational and graphical statistics **15**(2), 265–286 (2006)