Proceedings of Machine Learning Research vol 178:1–59, 2022 35th Annual Conference on Learning Theory

# Fast algorithm for overcomplete order-3 tensor decomposition

**Jingqiu Ding** JINGQIU.DING@INF.ETHZ.CH
**Tommaso d'Orsi** TOMMASO.DORSI@INF.ETHZ.CH
**Chih-Hung Liu** CHIH-HUNG.LIU@INF.ETHZ.CH
**David Steurer** DAVID.STEURER@INF.ETHZ.CH
**Stefan Tiegel** STEFAN.TIEGEL@INF.ETHZ.CH
*Universitätstrasse 6, 8006 Zürich, Switzerland*

**Editors:** Po-Ling Loh and Maxim Raginsky

## Abstract

We develop the first *fast* spectral algorithm to decompose a random third-order tensor over $\mathbb{R}^d$ of rank up to $O(d^{3/2}/\operatorname{polylog}(d))$. Our algorithm only involves simple linear algebra operations and can recover all components in time $O(d^{6.05})$ under the current matrix multiplication time.

Prior to this work, comparable guarantees could only be achieved via sum-of-squares [Ma, Shi, Steurer 2016]. In contrast, fast algorithms [Hopkins, Schramm, Shi, Steurer 2016] could only decompose tensors of rank at most $O(d^{4/3}/\operatorname{polylog}(d))$.

Our algorithmic result rests on two key ingredients. A clean lifting of the third-order tensor to a sixth-order tensor, which can be expressed in the language of tensor networks. A careful decomposition of the tensor network into a sequence of rectangular matrix multiplications, which allows us to have a fast implementation of the algorithm.

**Keywords:** Overcomplete tensor decomposition, spectral algorithms, tensor networks

## 1. Introduction

Tensor decomposition is a widely studied problem in statistics and machine learning Rabanser et al. (2017); Sidiropoulos et al. (2017); Bacciu and Mandic (2020). Techniques that recover the hidden components of a given tensor have a wide range of applications such as dictionary learning Barak et al. (2015); Ma et al. (2016), clustering Hsu and Kakade (2013), or topic modeling Anandkumar et al. (2012). From an algorithmic perspective, third-order tensors –which do not admit a natural unfolding[1]– essentially capture the challenges of the problem. Given

$$\mathbf{T} = \sum_{i\in[n]} a_i^{\otimes 3} \quad \in (\mathbb{R}^d)^{\otimes 3}\,, \tag{1.1}$$

we aim to approximately recover the unknown components $\{a_i\}$. While, in general, decomposing Eq. (1.1) is NP-hard Hillar and Lim (2013), under natural (distributional) assumptions, polynomial time algorithms are known to accurately recover the components. When $n \leqslant d$, the problem is said to be undercomplete and when $n > d$ it is called overcomplete. In the undercomplete settings, a classical algorithm Harshman (1970); Leurgans et al. (1993) called simultaneous diagonalization can efficiently decompose the input tensor when the hidden vectors are linearly independent. In stark difference from the matrix settings, tensor decompositions remain unique even when the

1. That is, a natural mapping to squared matrices

number of factors $n$ is larger than the ambient dimension $d$, making the problem suitable for applications where matrix factorizations are insufficient. This observation has motivated a flurry of work Lathauwer et al. (2007); Barak et al. (2015); Ge and Ma (2015); Anandkumar et al. (2015); Ma et al. (2016); Hopkins et al. (2016, 2019) in an effort to design algorithms for overcomplete tensor decompositions.

When the hidden vectors are sampled uniformly from the unit sphere[2], the best guarantees in terms of number of components with respect to the ambient dimension, corresponding to $\tilde{\Omega}(n^{2/3}) \leqslant d$,[3] have been achieved through semidefinite-programming Ma et al. (2016). The downside of this algorithm is that it is virtually impossible to be effectively used in practice due to the high order polynomial running time. For this reason, obtaining efficient algorithms for overcomplete tensor decomposition has remained a pressing research question. This is also the focus of our work.

For $\tilde{\Omega}(n^{2/3}) \leqslant d$, the canonical tensor power iteration Anandkumar et al. (2015, 2017) is known to converge to one of the hidden vectors –in nearly linear time[4]– given an initialization vector with *non-trivial* correlation to one of the components Anandkumar et al. (2015). Unfortunately, this does not translate to any speed up with respect to the aforementioned sum-of-squares algorithm, as that remains the only efficient algorithm known to obtain such an initialization vector. Inspired by the insight of previous sum-of-squares algorithms Ge and Ma (2015); Hopkins et al. (2016) proposed the first subquadratic spectral algorithm for overcomplete order-3 tensor decomposition. This algorithm, successfully recovers the hidden vectors as long as $\tilde{\Omega}(n^{3/4}) \leqslant d$, but falls short of the $\tilde{\Omega}(n^{2/3}) \leqslant d$ guarantees obtained via sum-of-squares.

In the related context of fourth order tensors, under algebraic assumptions satisfied by random vectors, Lathauwer et al. (2007); Hopkins et al. (2019) could recover up to $n \leqslant \tilde{O}(d^2)$ components in subquadratic time. Given an initialization vector satisfying very mild condition, under weak assumption on the components, recent work Batselier and Wong (2016); Kileel and Pereira (2019); Kileel et al. (2021) develops improved fast power methods which can work for $\tilde{O}(d^2)$ components. These results and approaches, however, are inherently hard to generalize to third-order tensors. [5]

A natural question arises:

**Question** For third-order random overcomplete tensor decomposition, can we develop a fast spectral algorithm which recovers when $\tilde{\Omega}(n^{2/3}) \geqslant d$?

In this work, we present the first *fast* spectral algorithm that provably recovers all the hidden components as long as $\tilde{\Omega}(n^{2/3}) \leqslant d$, under natural distributional assumptions. To the best of our knowledge, this is the first algorithm with a *practical* running time that provides guarantees comparable to SDP-based algorithms. More concretely we prove the following theorem.

**Theorem 1 (Fast overcomplete tensor decomposition)** *Let* $\mathbf{T} \in \left(\mathbb{R}^d\right)^{\otimes 3}$ *be a tensor of the form*

$$\mathbf{T} = \sum_{i \in [n]} a_i^{\otimes 3},$$

2. It is understood that similar reasoning applies to i.i.d. Gaussian vectors and other subgaussian symmetric distributions.
3. We hide constant factors with the notation $O(\cdot), \Omega(\cdot)$ and multiplicative *polylogarithmic* factors in the ambient dimension $d$ by $\tilde{O}(\cdot), \tilde{\Omega}(\cdot)$.
4. Hence it requires $\tilde{O}(n \cdot d^3)$ time to recover all components.
5. For general tensor decomposition, there are some other algorithm methods Kolda (2015); Nie (2014) which achieve recovery empirically or under certain practical assumptions. But they do not provably give comparable guarantees in random overcomplete tensor decomposition to our knowledge.

*where $a_1, \ldots, a_n$ are i.i.d. vectors sampled uniformly from the unit sphere in $\mathbb{R}^d$ and $\tilde{\Omega}\left(n^{2/3}\right) \leqslant d$. There exists a randomized algorithm that, given $\mathbf{T}$, with high probability recovers all components within error $\tilde{O}(\sqrt{n}/d)$ in time $\tilde{O}\left(d^{2\omega\left(1+\frac{\log n}{2\log d}\right)}\right)$, where $d^{\omega(k)}$ is the time required to multiply a $(d^k \times d)$ matrix with a $(d \times d)$ matrix.*[6]

In other words, Theorem 1 states that there exists an algorithm that, in time $\tilde{O}\left(d^{2\omega\left(1+\frac{\log n}{2\log d}\right)}\right)$, outputs vectors $b_1, \ldots, b_n \in \mathbb{R}^d$ such that

$$\forall i \in [n]\,, \quad \left\|a_i - b_{\pi[i]}\right\| \leqslant \tilde{O}\left(\frac{\sqrt{n}}{d}\right)\,,$$

for some permutation $\pi : [n] \to [n]$.

The distributional assumptions of Theorem 1 are the same of Hopkins et al. (2016); Ma et al. (2016). In contrast to Hopkins et al. (2016), our result can deal with the inherently harder settings of $\tilde{\Omega}(n^{2/3}) \leqslant d \leqslant \tilde{O}(n^{3/4})$. In comparison to the sum-of-squares algorithm in Ma et al. (2016), which runs in time $\tilde{O}(nd)^C$, for a large constant $C \geqslant 12$, our algorithm provides significantly better running time. For $\tilde{\Omega}(n^{2/3}) \leqslant d$, it holds that $\omega\left(1 + \frac{\log n}{2\log d}\right) \leqslant \omega(1.75)$. Current upper bounds on rectangular matrix multiplication constants show that $\omega(1.75) \leqslant 3.021591$ and thus, the algorithm runs in time at most $\tilde{O}\left(d^{6.043182}\right)$. Moreover, with the current upper bounds on $\omega(\frac{5}{3})$, the algorithm even runs in subquadratic time for $\tilde{\Omega}(n^{3/4}) \leqslant d$.

## 2. Preliminaries

**Organization** The paper is organized as follows. We present the main ideas in Section 3. In Section 4 we present the algorithm for fast overcomplete third-order tensor decomposition. We prove its correctness through Appendix A, Appendix B, and Appendix C. In section Appendix D we analyze the running time of the algorithm. Finally, Appendix B contains a proof for robust order-6 tensor decomposition which is essentially standard, but instrumental for our result.

**Notations for matrices** Throughout the paper, we denote matrices by non-bold capital letters $M \in \mathbb{R}^{d\times d}$ and vectors $v \in \mathbb{R}^d$ by lower-case letters. Given a matrix $M \in \mathbb{R}^{d^2\times d^2}$, at times we denote its entries with the indices $i, j, k, \ell \in [d]$. $M_{i,j,k,\ell}$ is the $(i \cdot j)$-$(k \cdot \ell)$-th entry of $M$. We then write $M_{\{1,2,3\}\{4\}}$ for the $d^3$-by-$d$ matrix obtained reshaping $M$, so that $\left(M_{\{1,2,3\}\{4\}}\right)_{i,j,k,\ell} = M_{i,j,k,\ell}$. Analogously, we express reshapings of matrices in $\mathbb{R}^{d^3\times d^3}$. We denote the identity matrix in $\mathbb{R}^{m\times m}$ by $\mathrm{Id}_m$. For any matrix $M$, we denote its Moore-Penrose inverse as $M^+$, its spectral norm as $\|M\|$ and its Frobenius norm as $\|M\|_{\mathrm{F}}$.

**Notations for tensors** Throughout the paper we denote tensors by boldface capital letters $\mathbf{T} \in \left(\mathbb{R}^d\right)^{\otimes t}$. For simplicity, for a vector $v \in \mathbb{R}^d$, we denote by $v^{\otimes t} \in \left(\mathbb{R}^d\right)^{\otimes t}$ both the tensor $v \underbrace{\otimes \ldots \otimes}_{t \text{ times}} v$ and its vectorization $v^{\otimes t} \in \mathbb{R}^{d^t}$, we also write $\left(v^{\otimes \ell}\right)\left(v^{\otimes t-\ell}\right)^{\mathsf{T}} \in \mathbb{R}^{d^\ell \times d^{t-\ell}}$ for the $d^\ell$-by-$d^{t-\ell}$ matrix flattening of $v^{\otimes t}$. If this is denoted by a boldface capital letter it is taken to be a tensor and if it is denoted by a non-bold capital letter as a matrix. We expect the meaning to be clear from context.

6. In Appendix I we provide a table containing current upper bounds on rectangular matrix multiplication constants.

For a tensor $\mathbf{T} \in \left(\mathbb{R}^d\right)^{\otimes t}$ and a partition of its modes into ordered sets $S_1, \dots, S_\ell \subseteq \{1, \dots, t\}$ we denote by $\mathbf{T}_{S_1,\dots,S_\ell}$ its flattening into an $\ell$-th order tensor. For example, for $A, B \subseteq \{1, \dots, t\}$ with $A \cup B = \{1, \dots, t\}$ and $A \cap B = \emptyset$, $\mathbf{T}_{A,B}$ is a $d^{|A|}$-by-$d^{|B|}$ matrix flattening of $\mathbf{T}$. We remark that the order of the modes matter. For a tensor $\mathbf{T} \in (\mathbb{R}^d)^{\otimes 3}$ and a vector $v \in \mathbb{R}^d$, we denote by $\mathbf{T}(v, \cdot, \cdot)$ or $(v \otimes \mathrm{Id}_d \otimes \mathrm{Id}_d)\, T$ the matrix obtain contracting the first mode of $\mathbf{T}$ with $v$. A similar notation will be used for higher order tensors. Given a tensor $\mathbf{T} \in (\mathbb{R}^d)^{\otimes 6}$, we sometimes write $\mathbf{T}_{\{1,2\}\{3,4\}\{5,6\}}$ as its reshaping to a $d^2 \times d^2 \times d^2$ tensor.

**Notations for probability and asymptotic bounds** We hide constant factors with the notation $O(\cdot), \Omega(\cdot)$ and multiplicative *polylogarithmic* factors in the ambient dimension $d$ by $\tilde{O}(\cdot), \tilde{\Omega}(\cdot)$.

We denote the standard Gaussian distribution by $N(0, \mathrm{Id}_m)$. We say an event happens with high probability if it happens with probability $1 - o(1)$. We say an event happens with overwhelming probability (or *w.ov.p*) if it happens with probability $1 - d^{-\omega(1)}$.

**Tensor networks** There are many different ways one can multiply tensors together. An expressive tool that can be used to represent some specific tensor multiplication is that of tensor networks. A tensor network is a diagram with nodes and edges (or legs). Nodes represent tensors and edges between nodes represent contractions. Edges can be dangling and need not be between pairs of nodes. Thus a third order tensor $\mathbf{T} \in (\mathbb{R}^d)^{\otimes 3}$ corresponds to a node with three dangling legs. Further examples are shown in the picture below. For a more detailed discussion we direct the reader to Moitra and Wein (2019).

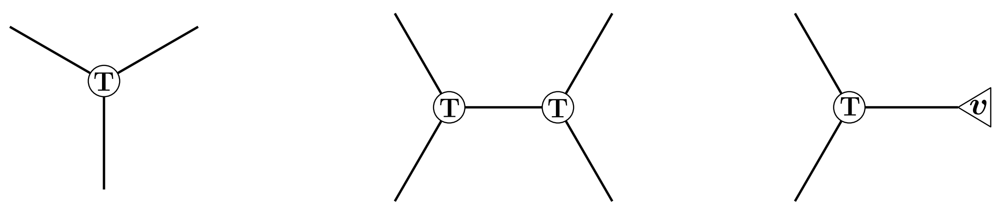


Figure 1: Fig 1.(a) represents a single third-order tensor. Fig 1.(b) depicts two tensor contracted via one mode. Fig 1.(c) represents a tensor contracted on one mode with a vector.

## 3. Techniques

Here we present the main ideas behind our result. Throughout the section we assume to be given a tensor $\mathbf{T} = \sum_{i \in [n]} a_i^{\otimes 3} \in (\mathbb{R}^d)^{\otimes 3}$ with components $a_1, \dots, a_n \in \mathbb{R}^d$ independently and uniformly sampled from the unit sphere.

**From $\tilde{\Omega}(n^{3/4}) \leqslant d$ to $\tilde{\Omega}(n^{2/3}) \leqslant d$: a first matrix with large spectral gap** To understand how to recover the components for $\tilde{\Omega}\left(n^{2/3}\right) \leqslant d$, it is useful to revisit the spectral algorithm in Hopkins et al. (2016). For a random vector $g \sim N(0, \mathrm{Id}_d)$, the matrix can be described by the tensor network in Fig. 2(a), and is a contraction between $(g \otimes \mathrm{Id} \otimes \mathrm{Id})\, \mathbf{T}$ and $\mathbf{T} \otimes \mathbf{T}$ up to reshaping. It can be written as:

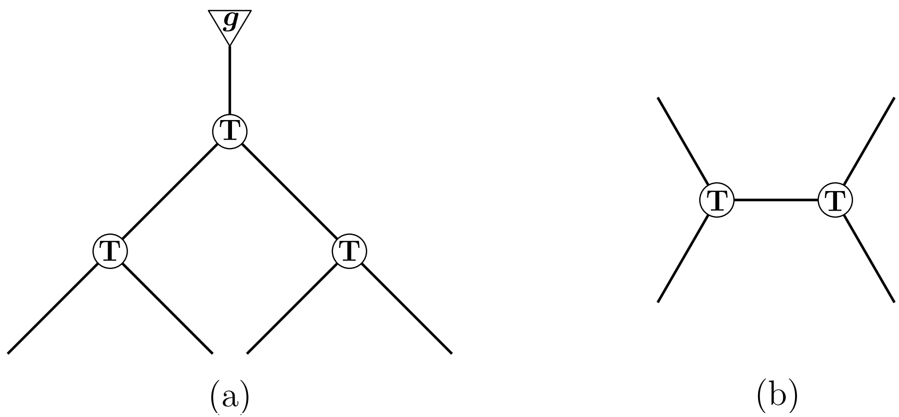


Figure 2: (a ) The tensor network for the algorithm in Hopkins et al. (2016) where $g \sim N(0, \mathrm{Id}_d)$. (b) A simple tensor network with signal-to-noise ratio $\tilde{\Omega}\left(d^{3/2}/n\right)$.

$$\begin{aligned}\sum_{i,j\in[n]} \langle g \otimes a_i \otimes a_j, \mathbf{T}\rangle \left(a_i \otimes a_j\right)\left(a_i \otimes a_j\right)^{\mathsf{T}} = &\sum_{i\in[n]} \langle g, a_i\rangle \left(a_i^{\otimes 2}\right)\left(a_i^{\otimes 2}\right)^{\mathsf{T}} \\ &+ \underbrace{\sum_{i,j\in[n],i\neq j} \langle g, \mathbf{T}(a_i \otimes a_j)\rangle \left(a_i \otimes a_j\right)\left(a_i \otimes a_j\right)^{\mathsf{T}}}_{:=\boldsymbol{E}}\end{aligned} \tag{3.1}$$

Roughly speaking, the algorithm amounts to computing the $n$ leading eigenvectors of such matrices. Since $\left\|\sum_{i\in[n]}\langle g, a_i\rangle \left(a_i^{\otimes 2}\right)\left(a_i^{\otimes 2}\right)^{\mathsf{T}}\right\| = \tilde{\Theta}(1)$, as long as the spectral norm of the noise $E$ is significantly smaller, the signal-to-noise ratio stays bounded away from zero and we can hope to recover the components. By decoupling inequalities similar to those in Ge and Ma (2015), w.h.p., it holds that $\langle g \otimes a_i \otimes a_j, \mathbf{T}\rangle \leqslant \tilde{O}(\sqrt{n}/d)$, and the derivations in Hopkins et al. (2016) further show that $\|E\| \leqslant \tilde{O}(n^{3/2}/d^2)$. Hence, this algorithm can recover the components as long as $\tilde{O}(n^{3/4}) \leqslant d$.

To improve over this result, the first key observation to make is that the term $\langle g, \mathbf{T}(a_i \otimes a_j)\rangle$ is unnecessarily large. In fact, for $n > d$, it is significantly larger (in absolute value) than the inner product $|\langle a_i, a_j\rangle| \leqslant \tilde{O}(1/\sqrt{d})$, which appears to be a reasonable yardstick for the scalar values at play in the computation, as we try to exploit the near orthogonality of the components. This suggest that even simply replacing $\langle g \otimes a_i \otimes a_j, \mathbf{T}\rangle$ by the inner product $\langle a_i, a_j\rangle$ could increase the spectral gap between the components we are trying to retrieve and the noise. Indeed, this can be achieved by considering the tensor network in Fig. 2(b), corresponding to the matrix

$$\sum_{i,j\in[n]} \langle a_i, a_j\rangle \left(a_i \otimes a_j\right)\left(a_i \otimes a_j\right)^{\mathsf{T}} = \sum_{i\in[n]} \left(a_i^{\otimes 2}\right)\left(a_i^{\otimes 2}\right)^{\mathsf{T}} + \underbrace{\sum_{i,j\in[n],i\neq j} \langle a_i, a_j\rangle \left(a_i \otimes a_j\right)\left(a_i \otimes a_j\right)^{\mathsf{T}}}_{:=E}.$$

On the one hand, with high probability, the spectral norm of the signal part satisfies $\left\|\sum_{i\in[n]}\left(a_i^{\otimes 2}\right)\left(a_i^{\otimes 2}\right)^{\mathsf{T}}\right\| = \Omega(1)$. On the other hand by (Ge and Ma, 2015, Lemma 13), with high probability, the spectral norm of $E$ is $\tilde{O}(n/d^{3/2})$. Thus, this simple tensor network provides the noise with the spectral norm we are looking for, i.e., $o(1)$ as long as $n \leqslant \tilde{O}(d^{3/2})$.[7]

The problem with the fourth order tensor network above is that it is not clear how one could directly extract even a single component. The canonical recipe, namely: (i) apply a random contraction $g \sim N(0, \mathrm{Id}_{d^2})$, (ii) recover the top eigenvector; does not work as after contracting the tensor we would end up with a rank $d$ matrix, while we wish to recover $n > d$ vectors. A natural workaround to this issue consists of lifting the fourth order tensor to a higher dimensional space and *then* applying the canonical recipe.

**Lifting to a higher order using tensor networks** It is straightforward to phrase lifting to higher orders in the language of tensor networks. For example, consider the following network (Fig. 3):

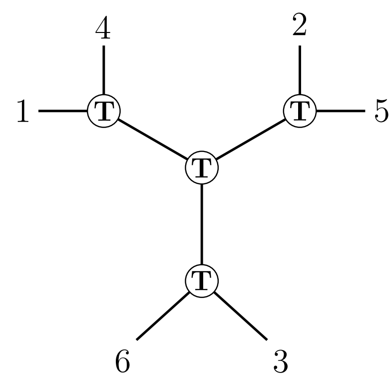


Figure 3: Lifting of the tensor network in Fig. 2(b). The numbers attached to the dangling edges can be used to keep track of the flattenings we will use throughout the paper.

In a similar spirit to Fig. 2(b), this tensor network can be flattened as the $d^3$-by-$d^3$ matrix

$$T_6 = \sum_{i\in[n]} \left(a_i^{\otimes 3}\right)\left(a_i^{\otimes 3}\right)^{\mathsf{T}} + \underbrace{\sum_{\substack{\{i,j,k,\ell\}\in[n]^4\\ i,j,k,\ell \text{ not all equal}}} \langle a_i, a_j\rangle\langle a_i, a_k\rangle\langle a_i, a_\ell\rangle(a_j\otimes a_k\otimes a_k)(a_j\otimes a_\ell\otimes a_\ell)^{\mathsf{T}}}_{=:E}\,.$$

Here $E$ is a sum of $O\left(n^4\right)$ dependent random matrices and thus, a priori, it is not clear how to study its spectrum. In particular there are many different terms in $E$ with distinct, but possibly aligning, spectra. To overcome this obstacle, we partition the terms in $E$ based on their index patterns. Mapping each index to a color, this essentially amounts to considering all the non-isomorphic 2-, 3- or 4-colorings of the tensor network in Fig. 3 (picking one arbitrary representative per class). Since the number of such non-isomorphic colorings is constant, we can bound each set in the partition separately, knowing that this triangle inequality will be tight up to constant factors.

To build some intuition consider as an example the case in which $i \neq j = k = l$. This corresponds to the coloring in which we assign a given color to the center node and a different

7. We remark that the tensor network in Fig. 2(b) was implicitly considered in Ge and Ma (2015) in the analysis of their quasi-polynomial time SoS algorithm.

one to all the leaves. Let $E'$ denote the error matrix corresponding to this case. Then, using a decoupling inequality similar to the one used for the analysis of the networks in Fig. 2 and standard Matrix Rademacher bounds, we obtain

$$\|E'\| = \left\| \sum_{i,j\in[n],i\neq j} \langle a_i, a_j\rangle^3 (a_j^{\otimes 3})(a_j^{\otimes 3})^\top \right\| \leqslant \tilde{O}\left(\sqrt{n}\cdot \frac{1}{\sqrt{d^3}}\right) \cdot \left\| \sum_{j\in[n]} (a_j^{\otimes 3})(a_j^{\otimes 3})^\top \right\| ,$$

where we also used again that for $i \neq j$ it holds that $|\langle a_i, a_j\rangle| \leqslant \tilde{O}(1/\sqrt{d})$. Since the spectral norm of the sum on the right-hand side can be bounded by $\tilde{O}(1)$, it follows that $\|E'\|_2 = \tilde{O}(\sqrt{n/d^3}) = \tilde{O}(\sqrt{n^2/d^3})$. Using arguments in a similar spirit, we can also bound the spectral norm of the other colorings by $\tilde{O}(\sqrt{n^2/d^3})$ as desired. This allows us to show that overall the noise has also spectral norm bounded by $\tilde{O}(\sqrt{n^2/d^3})$, implying that the signal-to-noise ratio has not increased.

**Recovering one component from the tensor network** To recover a single component form this network, we can do the following: Contracting (an appropriately flattened version of) $T_6$ with a random vector $g \sim N(0, \mathrm{Id}_{d^2})$ results in the matrix

$$\sum_{i\in[n]} \langle g, a_i^{\otimes 2}\rangle \left(a_i^{\otimes 2}\right) \left(a_i^{\otimes 2}\right)^\mathsf{T} + \sum_{i,j\in[n]} g_{ij} E_{ij} \,. \tag{3.2}$$

Compared to Eq. (3.1), the good news is that the contraction has broken the symmetry of the signal. However, well-known facts about Gaussian matrix series assert that the spectral norm of the randomly contracted error term behaves like the norm of a $d^4$-by-$d^2$ flattening of $E$, which necessarily satisfies the inequality

$$\left\|E_{\{1,2,3,4\}\{5,6\}}\right\|^2 \geqslant \frac{\|E\|_\mathrm{F}^2}{\mathrm{rank}(E_{\{1,2,3,4\}\{5,6\}})} \geqslant \tilde{\Omega}(n/d)\,,$$

thus jeopardizing our efforts of having a large spectral signal-to-noise ratio. We can overcome this issue with two preprocessing steps. (i) Truncate $T_6$ to its best rank-$n$ approximation $T_6^{\leqslant n}$ recovering its $n$ leading eigenvectors, so to have $\|E\|_\mathrm{F} \leqslant \sqrt{n}\cdot \tilde{O}(n/d^{3/2})$. (ii) Project the truncated matrix onto the space of matrices with bounded spectral norm after rectangular reshapings[8]

$$\left\|\left(T_6^{\leqslant n}\right)_{\{1,2,3,4\}\{5,6\}}\right\| \leqslant 1\,, \qquad \left\|\left(T_6^{\leqslant n}\right)_{\{1,2,5,6\}\{3,4\}}\right\| \leqslant 1\,.$$

After this sequence of projections, we can take a random contraction. In the resulting matrix

$$\tilde{T}_4 = \sum_{i\in[n]} \langle g, a_i^{\otimes 2}\rangle \left(a_i^{\otimes 2}\right) \left(a_i^{\otimes 2}\right)^\mathsf{T} + \tilde{E}\,,$$

the noise satisfies $\left\|\tilde{E}\right\| \leqslant \Theta(1)$ and $\left\|\tilde{E}\right\|_\mathrm{F} \leqslant \|E\|\cdot\sqrt{n} \leqslant \tilde{O}\left(\frac{n^2}{d^3}\cdot\sqrt{n}\right)$. We can thus approximately recover the components not hidden by the noise. This approach for partially recovering the components is similar in spirit to Schramm and Steurer (2017). However, for recovering all of the components, additional steps and a finer analysis are needed compared to Schramm and Steurer (2017), since the input tensor is overcomplete.

8. It can be observed that each of these projection does not destroy the properties ensured by the others. In other words two projections are enough to ensure the resulting matrix is in the intersection of the desired subspaces.

**Recovering all components from the tensor network** While the noise in $\tilde{T}_4$ is not adversarial, it has become difficult to manipulate after the pre-processing steps outlined above. The issue is that, without looking into $\mathbf{E}$, we cannot guarantee that its eigenvectors are spread enough and do not *cancel out* a fraction of the components, making full recovery impossible. Nevertheless the above reasoning ensures we can obtain $\tilde{O}(n/d^{3/2})$-close approximation vectors $b_1, \ldots, b_m \in \mathbb{R}^d$ of components $a_1, \ldots, a_m$ for some $\Omega(n) \leqslant m < n$.

Now, a natural approach to recover all components would be that of subtracting the learned components

$$T_6' = T_6 - \sum_{i \in [m]} \left(b_i^{\otimes 3}\right) \left(b_i^{\otimes 3}\right)^{\mathsf{T}}$$

and repeat the algorithm on $T_6'$. The approximation error here is

$$\left\| \sum_{i \in [m]} \left(a_i^{\otimes 3}\right) \left(a_i^{\otimes 3}\right)^{\mathsf{T}} - \sum_{i \in [m]} \left(b_i^{\otimes 3}\right) \left(b_i^{\otimes 3}\right)^{\mathsf{T}} \right\| \approx \tilde{O}\left(\sqrt{m} \cdot (n/d^{3/2})^3\right)$$

and so if indeed $n = o(d^{8/7})$ we could simply rewrite

$$T_6' = \sum_{m+1 \leqslant i \leqslant n} \left(a_i^{\otimes 3}\right) \left(a_i^{\otimes 3}\right)^{\mathsf{T}} + E', \qquad \text{where } \left\|E'\right\| \leqslant O(1/\operatorname{polylog}(d)).$$

For $n = \omega(d^{(8/7)})$, however the approximation error of our estimates is too large and this strategy fails.

We work around this obstacle boosting the accuracy of our estimates. We use each $b_i$ has a *warm start* and perform tensor power iteration Anandkumar et al. (2015). For each estimate this yield a new vector $\tilde{b}_i$ satisfying

$$1 - \langle a_i, \tilde{b}_i \rangle \leqslant \tilde{O}(\sqrt{n}/d).$$

Since now

$$\left\| \sum_{i \in [m]} \left(a_i^{\otimes 3}\right) \left(a_i^{\otimes 3}\right)^{\mathsf{T}} - \sum_{i \in [m]} \left(\tilde{b}_i^{\otimes 3}\right) \left(\tilde{b}_i^{\otimes 3}\right)^{\mathsf{T}} \right\| \approx O\left(\sqrt{m} \cdot (\sqrt{n}/d)^3\right),$$

as $\tilde{\Omega}(n^{2/3}) \leqslant d$ and $m \leqslant n$, we can subtract these estimates from $T_6$ and repeat the algorithm.

**Speeding up the computation via tensor network decomposition** The algorithm outlined above is particularly natural and streamlined, however a naïve implementation would require running time significantly larger than the result in Theorem 1. For example, naïvely computing the first $n$ eigenvectors of $T_6$ already requires time $O(n \cdot d^6)$. To speed up the algorithm we carefully compute an implicit (approximate) representation of $T_6$ in terms of its $n$ leading eigenvectors. Then use Gaussian rounding on this approximate representation of the data. Since the signal part $\sum_{i \in [n]} \left(a_i^{\otimes 3}\right) \left(a_i^{\otimes 3}\right)^{\mathsf{T}}$ has rank $n$, this approximation should loose little information about the components. This implicit representation is similar to the one used in Hopkins et al. (2019), however our path to computing it presents different challenges and thus differs significantly from previous work.

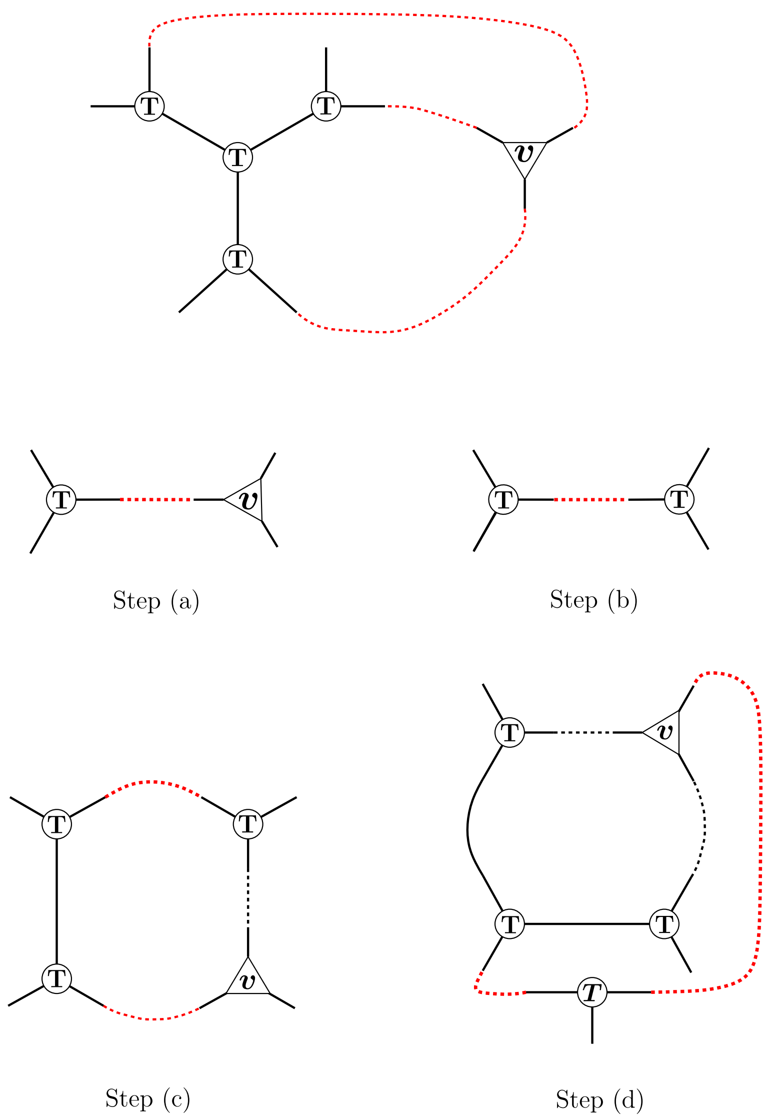


Figure 4: Step $a$ and $b$ can be seen as $(d^2 \times d)$ times $(d \times d)$ matrix multiplications. Similarly, step $c$ (the bottleneck) and step $d$ can be computed respectively as $(d^2 \times d^2)$ times $(d^2 \times d^2)$ and $(d^2 \times d^2)$ times $(d^2 \times d)$ matrix multiplications.

Our strategy is to use power iteration over $T_6$. The running time of such an approach is bounded by the time required to contract $T_6$ with a vector $v$ in $\mathbb{R}^{d^3}$. However, since we have access to $\mathbf{T}$, by *carefully decomposing* the tensor network we can perform this matrix-vector multiplication in a significantly smaller number of operations. In particular, as shown in Fig. 4, we may rewrite

$$\begin{aligned}
T_6 v &= \sum_{\{i,j,k,\ell\} \in [n]^4} \langle a_i, a_j \rangle \langle a_i, a_k \rangle \langle a_i, a_\ell \rangle (a_j \otimes a_k \otimes a_k)(a_j \otimes a_\ell \otimes a_\ell)^\mathsf{T} v \\
&= \left[ \left( \mathbf{T}_{\{1,2\}\{3\}} \mathbf{T}_{\{3\}\{1,2\}} \right) \left( \mathbf{T}_{\{1,2\}\{3\}} v_{\{3\}\{1,2\}} \right) \right]_{\{1,3\}\{2,4\}} \mathbf{T}_{\{1,2\}\{3\}} \,.
\end{aligned}$$

In other words we may compute $T_6 v$ using only a constant number of rectangular matrix multiplications, each of which has at most the complexity of a $d \times d^2$ times $d^2 \times d$ matrix multiplication![9] This approach can be even parallelized to compute the top $n$ eigenvectors of $T_6$ at the same time.

Upon obtaining this representation, we can perform basic operations (such as tensor contractions) required in the second part of the algorithm more quickly, further reducing the running time of the algorithm. Indeed, using the speed up described above, the algorithm based on the tensor network in Fig. 3 can be implemented in time $\tilde{O}\left(d^{2\omega\left(1+\frac{\log n}{2\log d}\right)}\right)$, which for $n = \Theta(d^{3/2}/\operatorname{polylog}(d))$ can be bounded by $\tilde{O}\left(d^{6.043182}\right)$.

**Remark 2** *We observe that applying the robust fourth-order tensor decomposition algorithm in Hopkins et al. (2019) on the tensor network in Fig. 2(b) can recover* "a constant fraction, bounded away from 1," *of the components, but not all of them, in* $\tilde{O}(d^{6.5})$ *time; see Appendix E. In contrast, our algorithm based on the tensor network in Fig. 3 can recover* "all" *the components in* $\tilde{O}(d^{6.043182})$ *time.*

## 4. Fast and simple algorithm for third-order overcomplete tensor decomposition

In this section, we present our fast algorithm for overcomplete tensor decomposition, which will be used to prove Theorem 1. Formally the algorithm is the following.

9. Rectangular matrix multiplications of the form $d^k \times d^k$ times $d^k \times d$ can be reduced to rectangular matrix multiplication with dimension $d^k \times d$ Gall and Urrutia (2018).

**Algorithm 3 (Fast order-3 overcomplete tensor decomposition)**

***Input:*** *Tensor* $\mathbf{T} = \sum_{i\in[n]} a_i^{\otimes 3}$.

***Output:*** *Unit vectors* $b_1, \dots, b_n \in \mathbb{R}^d$.

1. ***Lifting:*** *Compute (as in Algorithm 33) the best rank-$n$ approximation $\hat{M}$ of the flattening $\mathbf{M}_{\{1,2,3\},\{4,5,6\}}$ of the tensor network (Fig. 3)*

$$\mathbf{M} = \sum_{i,j,k,\ell\in[n]} \langle a_i, a_j\rangle \cdot \langle a_i, a_k\rangle \cdot \langle a_i, a_\ell\rangle \cdot (a_j {a_j}^{\mathsf{T}}) \otimes (a_k {a_k}^{\mathsf{T}}) \otimes (a_\ell {a_\ell}^{\mathsf{T}}) .$$

2. ***Recovery:*** *Repeat $O(\log n)$ times:*

   (a) ***Pre-processing:*** *Project $\hat{M}$ into the space of matrices in $\mathbb{R}^{d^3\times d^3}$ satisfying*

$$\left\| \hat{M}_{\{1,2,3,4\}\{5,6\}} \right\| \leqslant 1 , \quad \left\| \hat{M}_{\{1,2,5,6\}\{3,4\}} \right\| \leqslant 1 .$$

   (b) ***Rounding:*** *Run $\tilde{O}(d^2)$ independent trials of Gaussian Rounding on $\hat{M}$ contracting its first two modes to obtain a set of $0.99n$ candidate vectors $b_1, \dots, b_{0.99n}$ (see Algorithm 12).*

   (c) ***Accuracy boosting:*** *Boost the accuracy of each candidate $b_i$ via tensor power iteration.*

   (d) ***Peeling of recovered components:***

   – *Set $\hat{M}$ to be the best rank-$0.01n$ approximation of $\hat{M} - \sum_{i\leqslant 0.99n} \left(b_i^{\otimes 3}\right) \left(b_i^{\otimes 3}\right)^{\top}$*

   – *Update $n \leftarrow 0.01n$.*

3. *Return all the candidate vectors $b_1, \dots, b_n$ obtained above.*

As discussed before, the goal of the Lifting step is to compute an approximation of the sixth-order tensor $\sum_{i=1}^n a_i^{\otimes 6}$ and the goal of the Recovery step is to use this to recover the components. To prove Theorem 1, we will first prove that these two steps are correct and then argue about their running time. Concretely, regarding the correctness of Algorithm 3 we prove the following two theorems:

**Theorem 4 (Correctness of the Lifting step)** *Let $a_1, \dots, a_n$ be i.i.d. vectors sampled uniformly from the unit sphere in $\mathbb{R}^d$ and consider*

$$\mathbf{M} = \sum_{i,j,k,\ell\in[n]} \langle a_i, a_j\rangle \cdot \langle a_i, a_k\rangle \cdot \langle a_i, a_\ell\rangle \cdot (a_j {a_j}^{\mathsf{T}}) \otimes (a_k {a_k}^{\mathsf{T}}) \otimes (a_\ell {a_\ell}^{\mathsf{T}}) .$$

*Then, if $n \leqslant O(d^{3/2}/\operatorname{polylog} d)$ with overwhelming probability*

$$\mathbf{M}_{\{1,2,3\},\{4,5,6\}} = \sum_{i\in[n]} a_i^{\otimes 3} \left(a_i^{\otimes 3}\right)^{\top} + E, \quad \textit{where} \quad \|E\| \leqslant \frac{1}{\operatorname{polylog} d}.$$

*Moreover, let $\hat{M}$ be the best rank-$n$ approximation of $\mathbf{M}_{\{1,2,3\},\{4,5,6\}}$ then*

$$\hat{M} = \sum_{i\in[n]} a_i^{\otimes 3}\left(a_i^{\otimes 3}\right)^\top + \hat{E}, \quad \textit{where} \quad \|\hat{E}\|_{\mathrm{F}} \leqslant \sqrt{8n}\cdot\|E\|, \textit{ and } \|\hat{E}\| \leqslant 2\cdot\|E\|.$$

**Remark 5** *Note that in the first display we identify $\mathbf{M}$ as a tensor and in the second display $M$ as a matrix. This should not lead to confusion as it should be clear from context which is meant and also from whether we use a bold or non-bold letter to denote it which is meant.*

**Theorem 6 (Correctness of the Recovery step)** *Let $a_1, \ldots, a_n$ be i.i.d. vectors sampled uniformly from the unit sphere in $\mathbb{R}^d$. Given as input*

$$\mathbf{T} = \sum_{i=1}^n a_i^{\otimes 3} \quad \textit{and} \quad \hat{M} = \sum_{i=1}^n a_i^{\otimes 3}\left(a_i^{\otimes 3}\right)^\top + E\,, \textit{with } \|E\| \leqslant \varepsilon \textit{ and } \|E\|_{\mathrm{F}} \leqslant \varepsilon\sqrt{n}\,,$$

*the Recovery step of Algorithm 3 returns unit norm vectors $b_1, b_2, \ldots, b_n$ satisfying*

$$\|a_i - b_{\pi(i)}\| \leqslant \tilde{O}\left(\frac{\sqrt{n}}{d}\right),$$

*for some permutation $\pi : [n] \to [n]$.*

Regarding the running time of the algorithm, we prove the result below.

**Theorem 7** *Algorithm 3 can be implemented in time $\tilde{O}\left(d^{2\omega\left(1+\frac{\log n}{2\log d}\right)} + nd^4\right)$, where $d^{\omega(k)}$ is the time required to multiply a $(d^k \times d)$ matrix with a $(d\times d)$ matrix.*

Combining the above three results directly yields a proof of Theorem 1. We will prove Theorem 4 in Appendix A and Theorem 6 over the course of Sections B and C, where Appendix B analyzes Steps 2(a) and 2(b) and Appendix C the rest. Finally, in Appendix D we will prove Theorem 7.

## Acknowledgments

This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No 815464)

## References

Zeyuan Allen-Zhu and Yuanzhi Li. Lazysvd: Even faster svd decomposition yet without agonizing pain. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016. URL https://proceedings.neurips.cc/paper/2016/file/c6e19e830859f2cb9f7c8f8cacb8d2a6-Paper.pdf.

Anima Anandkumar, Dean P. Foster, Daniel J. Hsu, Sham M. Kakade, and Yi-Kai Liu. A spectral algorithm for latent dirichlet allocation. In *Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems 2012. Proceedings of a meeting held December 3-6, 2012, Lake Tahoe, Nevada, United States*, 2012.

Animashree Anandkumar, Rong Ge, and Majid Janzamin. Learning overcomplete latent variable models through tensor methods. In Peter Grünwald, Elad Hazan, and Satyen Kale, editors, *Proceedings of The 28th Conference on Learning Theory*, volume 40, pages 36–112, 2015.

Animashree Anandkumar, Rong Ge, and Majid Janzamin. Analyzing tensor power method dynamics in overcomplete regime. *J. Mach. Learn. Res.*, 18(1):752–791, jan 2017. ISSN 1532-4435.

Davide Bacciu and Danilo P. Mandic. Tensor decompositions in deep learning. In *Proceedings of the 28th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN20)*, pages 441–450, 2020.

Boaz Barak, Jonathan A. Kelner, and David Steurer. Dictionary learning and tensor decomposition via the sum-of-squares method. In *Proceedings of the Forty-Seventh Annual ACM on Symposium on Theory of Computing, STOC 2015, Portland, OR, USA, June 14-17, 2015*, 2015.

Kim Batselier and Ngai Wong. Symmetric tensor decomposition by an iterative eigendecomposition algorithm. *J. Comput. Appl. Math.*, 308(C):69–82, dec 2016. ISSN 0377-0427. doi: 10.1016/j.cam.2016.05.024. URL https://doi.org/10.1016/j.cam.2016.05.024.

Victor H. de la Peña and S. J. Montgomery-Smith. Decoupling inequalities for the tail probabilities of multivariate $U$-statistics. *Annals of Probability*, 23(2):806–816, 1995.

Francois Le Gall and Florent Urrutia. Improved rectangular matrix multiplication using powers of the coppersmith-winograd tensor. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA18)*, pages 1029–1046, 2018.

Rong Ge and Tengyu Ma. Decomposing overcomplete 3rd order tensors using sum-of-squares algorithms. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM 2015, August 24-26, 2015, Princeton, NJ, USA*, 2015.

Moritz Hardt and Eric Price. The noisy power method: A meta algorithm with applications. In *Proceedings of the 2014 Conference on Neural Information Processing Systems (NeurIPS2014)*, pages 2861–2869, 2014.

Richard A Harshman. Foundations of the parafac procedure: Models and conditions for an "explanatory" multi-modal factor analysis. 1970.

Christopher J. Hillar and Lek-Heng Lim. Most tensor problems are np-hard. *J. ACM*, 60(6), 2013.

Samuel B. Hopkins, Tselil Schramm, Jonathan Shi, and David Steurer. Fast spectral algorithms from sum-of-squares proofs: tensor decomposition and planted sparse vectors. In *Proceedings of the 48th IEEE Annual Symposium on Foundations of Computer Science (STOC16)*, pages 178–191, 2016. The full version is in https://arxiv.org/abs/1512.02337.

Samuel B. Hopkins, Tselil Schramm, and Jonathan Shi. A robust spectral algorithm for overcomplete tensor decomposition. In *Proceedings of the 32nd Conference on Learning Theory (COLT19)*, pages 1683–1722, 2019.

Daniel J. Hsu and Sham M. Kakade. Learning mixtures of spherical gaussians: moment methods and spectral decompositions. In *Innovations in Theoretical Computer Science, ITCS '13, Berkeley, CA, USA, January 9-12, 2013*, 2013.

Joe Kileel and João M. Pereira. Subspace power method for symmetric tensor decomposition and generalized pca, 2019. URL https://arxiv.org/abs/1912.04007.

Joe Kileel, Timo Klock, and João M. Pereira. Landscape analysis of an improved power method for tensor decomposition. In Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang, and Jennifer Wortman Vaughan, editors, *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 6253–6265, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/31784d9fc1fa0d25d04eae50ac9bf787-Abstract.html.

Tamara G. Kolda. Numerical optimization for symmetric tensor decomposition. *Math. Program.*, 151(1):225–248, jun 2015. ISSN 0025-5610. doi: 10.1007/s10107-015-0895-0. URL https://doi.org/10.1007/s10107-015-0895-0.

Lieven De Lathauwer, Joséphine Castaing, and Jean-François Cardoso. Fourth-order cumulant-based blind identification of underdetermined mixtures. *IEEE Trans. Signal Process.*, 55(6-2), 2007.

S. E. Leurgans, R. T. Ross, and R. B. Abel. A decomposition for three-way arrays. *SIAM Journal on Matrix Analysis and Applications*, 14(4):1064–1083, 1993. doi: 10.1137/0614071. URL https://doi.org/10.1137/0614071.

Tengyu Ma, Jonathan Shi, and David Steurer. Polynomial-time tensor decompositions with sum-of-squares. In *Proceedings of the 57th IEEE Annual Symposium on Foundations of Computer Science (FOCS16)*, pages 438–446, 2016. The full version is in https://arxiv.org/abs/1610.01980.

Ankur Moitra and Alexander S. Wein. Spectral methods from tensor networks. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing, STOC 2019, Phoenix, AZ, USA, June 23-26, 2019*, pages 926–937, 2019.

Jiawang Nie. Generating polynomials and symmetric tensor decompositions, 2014. URL https://arxiv.org/abs/1408.5664.

Stephan Rabanser, Oleksandr Shchur, and Stephan Günnemann. Introduction to tensor decompositions and their applications in machine learning. *CoRR*, abs/1711.10781, 2017. URL http://arxiv.org/abs/1711.10781.

Tselil Schramm and David Steurer. Fast and robust tensor decomposition with applications to dictionary learning. In Satyen Kale and Ohad Shamir, editors, *Proceedings of the 30th Conference on Learning Theory (COLT17)*, pages 1760–1793, 2017.

Nicholas D. Sidiropoulos, Lieven De Lathauwer, Xiao Fu, Kejun Huang, Evangelos E. Papalexakis, and Christos Faloutsos. Tensor decomposition for signal processing and machine learning. *IEEE Transactions on Signal Processing*, 65(13):3551–3582, 2017.

Joel A. Tropp. An introduction to matrix concentration inequalities. *ArXiv*, abs/1501.01571, 2015.

## Appendix A. Lifting via tensor networks

In this section, we analyze the lifting part of our algorithm using tensor networks. More precisely, we prove that the tensor network in Fig. 3 has a large signal-to-noise ratio in the spectral norm sense, and that the noise of its corresponding top-$n$ eigenspace has a small Frobenius norm. Recall that our goal is to prove Theorem 4:

**Theorem 8 (Restatement of Theorem 4)** *Let $a_1, \ldots, a_n$ be i.i.d. vectors sampled uniformly from the unit sphere in $\mathbb{R}^d$ and consider*

$$\mathbf{M} = \sum_{i,j,k,\ell \in [n]} \langle a_i, a_j \rangle \cdot \langle a_i, a_k \rangle \cdot \langle a_i, a_\ell \rangle \cdot (a_j {a_j}^\mathsf{T}) \otimes (a_k {a_k}^\mathsf{T}) \otimes (a_\ell {a_\ell}^\mathsf{T}) .$$

*Then, if $n \leqslant O(d^{3/2} / \operatorname{polylog} d)$ with overwhelming probability*

$$\mathbf{M}_{\{1,2,3\},\{4,5,6\}} = \sum_{i \in [n]} a_i^{\otimes 3} \left(a_i^{\otimes 3}\right)^\top + E, \quad \textit{where} \quad \|E\| \leqslant \frac{1}{\operatorname{polylog} d}.$$

*Moreover, let $\hat{M}$ be the best rank-$n$ approximation of $\mathbf{M}_{\{1,2,3\},\{4,5,6\}}$ then*

$$\hat{M} = \sum_{i \in [n]} a_i^{\otimes 3} \left(a_i^{\otimes 3}\right)^\top + \hat{E}, \quad \textit{where} \quad \|\hat{E}\|_{\mathrm{F}} \leqslant \sqrt{8n} \cdot \|E\|, \textit{ and } \|\hat{E}\| \leqslant 2 \cdot \|E\|.$$

In Appendix A.1 we will prove its first part and in Appendix A.2, we analyze the best rank-$n$ approximation of $M$ to prove the second part.

### A.1. Spectral gap of the ternary-tree tensor network

In this section, we will prove the first part of Theorem 4.

**Lemma 9** *Consider the setting of Theorem 4: If $n \leqslant O(d^{3/2} / \operatorname{polylog} d)$, then with overwhelming probability*

$$\mathbf{M}_{\{1,2,3\},\{4,5,6\}} = \sum_{i \in [n]} a_i^{\otimes 3} \left(a_i^{\otimes 3}\right)^\top + E, \quad \textit{where} \quad \|E\| \leqslant \frac{1}{\operatorname{polylog} d}.$$

**Proof** For ease of notation we denote by $M = \mathbf{M}_{\{1,2,3\},\{4,5,6\}}$. To proof the theorem, we will split the sum into the part where some of the indices disagree and the part where all are equal. This second term (where $i = j = k = l$) gives exactly $\sum_{i \in [n]} a_i^{\otimes 3} \left(a_i^{\otimes 3}\right)^\top$. Hence, $E$ is the remaining part of the quadruple sum where not all indices are equal. We will analyze the spectral norm of this by further splitting the sum into parts where only some of the indices are equal. A clean way to conceptualize how we do this is as follows: Notice that each index in the sum comes from one node in the tensor network. Hence, we can think of *coloring* the four nodes of the ternary tree tensor network using four colors. We map a giving coloring to a part of the sum as follows: If two nodes share the same color, we will take this to mean that the corresponding indices in the sum are equal, whereas if they have different colors, this should mean that the indices are different. For example, the coloring that all the four nodes share the same color corresponds to the matrix

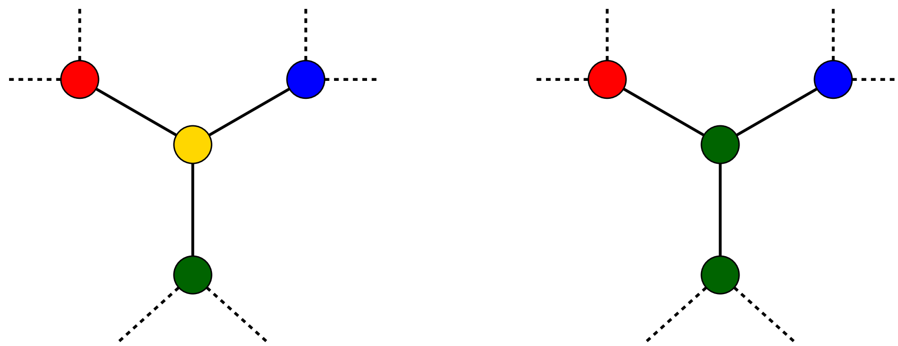


Figure 5: All leaves have a different color.

$\sum_{i\in[n]} a_i^{\otimes 3}\left(a_i^{\otimes 3}\right)^\top$. Whereas the one where say the middle node and one of the leaves have the same color and the remaining two leaves have two different colors (cf. Fig. 5 (b)) corresponds to

$$\sum_{i\in[n]} \|a_i\|^2 a_i a_i^\top \otimes \sum_{k\neq i} \langle a_i, a_k\rangle a_k a_k^\top \otimes \sum_{\ell\neq k,i} \langle a_i, a_\ell\rangle a_\ell a_\ell^\top$$

Therefore, each coloring corresponds to a matrix, and if we ignore permutations of colors (e.g. all nodes blue or all nodes red are identified as the same), since there are a constant number of colorings of the four nodes, the error matrix $E$ can be represented as a sum of a constant number of matrices, each of which corresponds to one coloring - again ignoring permutations of the colors. To bound the spectral norm of $E$, we can then bound each of the colorings independently. The colorings fall into three categories which we will analyze one by one.

1. All leaves have different colors (see Fig. 5)
2. Two leaves share the same color, but the other leaf doesn't (see Fig. 6)
3. All leaves share the same color, but the internal note has a different color

**First category.** We start with a detailed analysis for the coloring that all the four tensor nodes have different colors (Fig. 5(a)). This coloring corresponds to the following matrix

$$M_{\text{diff}} = \sum_{i\in[n]} \sum_{j\in[n],\, j\neq i} \langle a_i, a_j\rangle \cdot a_j a_j^\top \otimes \left( \sum_{k\in[n],\, k\neq i,\, j} \langle a_i, a_k\rangle \cdot a_k a_k^\top \otimes \left( \sum_{\ell\in[n],\, \ell\neq i,\, j,\, k} \langle a_i, a_\ell\rangle \cdot a_\ell a_\ell^\top \right)\right).$$

To bound its spectral norm, we will use a decoupling argument: Let $s_1, \ldots, s_n$ be $n$ independent random signs. Since $a_i$ and $s_i \cdot a_i$ share the same distribution, analyzing $M_{\text{diff}}$ is equivalent to analyzing

$$\sum_{i\in[n]} s_i \cdot \sum_{j\in[n],\, j\neq i} s_j \cdot \langle a_i, a_j\rangle \cdot a_j a_j^\top \otimes \left( \sum_{k\in[n],\, k\neq i,\, j} s_k \cdot \langle a_i, a_k\rangle \cdot a_k a_k^\top \otimes \left( \sum_{\ell\in[n],\, \ell\neq i,\, j,\, k} s_\ell \cdot \langle a_i, a_\ell\rangle \cdot a_\ell a_\ell^\top \right)\right).$$

To decouple the random signs in the above matrix, let $t_{i,j}$ for $1 \leqslant i \leqslant 4$ and $1 \leqslant j \leqslant n$ be $4n$ independent random signs, and define the following matrix

$$\tilde{M}_{\text{diff}} = \sum_{i\in[n]} t_{1,i}\cdot \sum_{j\in[n],\, j\neq i} t_{2,j}\cdot\langle a_i, a_j\rangle\cdot a_j a_j^\top \otimes\left(\sum_{k\in[n],\, k\neq i,\, j} t_{3,k}\cdot\langle a_i, a_k\rangle\cdot a_k a_k^\top \otimes\left(\sum_{\ell\in[n],\, \ell\neq i,\, j,\, k} t_{4,\ell}\cdot\langle a_i, a_\ell\rangle\cdot a_\ell a_\ell^\top\right)\right)$$

By Theorem 57, w.ov.p.,

$$\|M_{\text{diff}}\| = \tilde{O}\left(\left\|\tilde{M}_{\text{diff}}\right\|\right). \tag{A.1}$$

It hence suffices to analyze $\left\|\tilde{M}_{\text{diff}}\right\|$. To simplify notation, define the following matrices

$$\begin{aligned}
N_{i,j,k} &\coloneqq \sum_{\ell\in[n],\, \ell\neq i,\, j,\, k} t_{4,\ell}\cdot\langle a_i, a_\ell\rangle\cdot a_\ell a_\ell^\top \\
N_{i,j} &\coloneqq \sum_{k\in[n],\, k\neq i,\, j} t_{3,k}\cdot\langle a_i, a_k\rangle\cdot a_k a_k^\top \otimes N_{i,j,k} \\
N_i &\coloneqq \sum_{j\in[n],\, j\neq i} t_{2,j}\cdot\langle a_i, a_j\rangle\cdot a_j a_j^\top \otimes N_{i,j}
\end{aligned}$$

First, by a Matrix Rademacher bound (Theorem 54) and by Triangle inequality we get

$$\left\|\tilde{M}_{\text{diff}}\right\| = \sum_{i\in[n]} t_{1,i}\cdot N_i \overset{w.ov.p}{\leqslant} \tilde{O}\left(\left\|\sum_{i\in[n]} N_i^2\right\|\right)^{1/2} \leqslant \tilde{O}\left(\sqrt{n}\right)\cdot\max_{i\in[n]}\|N_i\|. \tag{A.2}$$

Second, by Theorem 55 and by Theorem 48(a)-(b) we have that for all $i$,

$$\begin{aligned}
\|N_i\| &= \left\|\sum_{j\in[n],\, j\neq i} t_{2,j}\cdot\langle a_i, a_j\rangle\cdot a_j a_j^\top \otimes N_{i,j}\right\| \\
&\overset{w.ov.p.}{\leqslant} \tilde{O}\left(\left(\max_{j\in[n],\, j\neq i}\|N_{i,j}\|\right)\cdot\left\|\sum_{j\in[n],\, j\neq i}\left(\langle a_i, a_j\rangle\cdot a_j a_j^\top\right)^2\right\|^{1/2}\right) \\
&= \max_{j\in[n],\, j\neq i}\|N_{i,j}\|\cdot\tilde{O}\left(\left\|\sum_{j\in[n],\, j\neq i}\langle a_i, a_j\rangle^2\cdot a_j a_j^\top\right\|^{1/2}\right) \\
&\leqslant \max_{j\in[n],\, j\neq i}\|N_{i,j}\|\cdot\tilde{O}\left(\max_{j\in[n],\, j\neq i}|\langle a_i, a_j\rangle|\cdot\left\|\sum_{j\in[n],\, j\neq i} a_j a_j^\top\right\|^{1/2}\right) \\
&\overset{w.ov.p.}{\leqslant} \max_{j\in[n],\, j\neq i}\|N_{i,j}\|\cdot\tilde{O}\left(\sqrt{\frac{n}{d^2}}\right)
\end{aligned} \tag{A.3}$$

By the same reasoning as above we get that for all $i \neq j$,

$$\|N_{i,j}\| \overset{w.ov.p.}{\leqslant} \tilde{O}\left(\sqrt{\frac{n}{d^2}}\right)\cdot\max_{k\in[n],\, k\neq i,\, j}\|N_{i,j,k}\| \overset{w.ov.p.}{\leqslant} \tilde{O}\left(\sqrt{\frac{n^2}{d^4}}\right) \tag{A.4}$$

where the last inequality follows from a Matrix Rademacher bound, similar steps as above, and a union bound over all $k \neq i, j$.

Combining Eq. (A.1), Eq. (A.2), Eq. (A.3) and Eq. (A.4) and two more union bounds over $i$ and $j \neq i$ (i.e., $\max$ in Eq. (A.2) and Eq. (A.3)), we finally obtain,

$$\|M_{\text{diff}}\| \overset{w.ov.p.}{\leqslant} \tilde{O}\left(\sqrt{n}\right) \cdot \tilde{O}\left(\sqrt{\frac{n}{d^2}}\right) \cdot \tilde{O}\left(\sqrt{\frac{n^2}{d^4}}\right) = \tilde{O}\left(\sqrt{\frac{n^4}{d^6}}\right) = \frac{1}{\operatorname{polylog} d} \tag{A.5}$$

Next, we discuss the second coloring in the first category. As seen before the matrix corresponding to Fig. 5(b) looks as follows:

$$\sum_{i\in[n]} \|a_i\|^2 a_i a_i^\top \otimes \sum_{k\neq i} \langle a_i, a_k\rangle a_k a_k^\top \otimes \sum_{\ell\neq k,i} \langle a_i, a_\ell\rangle a_\ell a_\ell^\top$$

Again considering $s_i a_i$ instead of $a_i$ for independent random signs and invoking Theorem 57 it suffices to bound the spectral norm of

$$\sum_{i\in[n]} a_i a_i^\top \otimes \sum_{k\neq i} t_{1,k} \langle a_i, a_k\rangle a_k a_k^\top \otimes \sum_{\ell\neq k,i} t_{2,\ell} \langle a_i, a_\ell\rangle a_\ell a_\ell^\top$$

where $t_{i,j}$ for $i = 1, 2, j \in [n]$ are independent random signs. Similarly as before and overloading notation, we define $N_{i,k} := \sum_{\ell\neq k,i} t_{2,\ell} \langle a_i, a_\ell\rangle a_\ell a_\ell^\top$ and $N_i := \sum_{k\neq i} t_{1,k} \langle a_i, a_k\rangle a_k a_k^\top \otimes N_{i,k}$. First, using Lemma 56 with the fact that $a_i a_i^\top$ is a psd matrix we get that the spectral norm of this is at most

$$\left\|\sum_{i\in[n]} a_i a_i^\top \otimes N_i\right\| \leqslant \left(\max_{i\in[n]} \|N_i\|\right) \cdot \left\|\sum_{i\in[n]} a_i a_i^\top\right\|^{1/2} \leqslant \tilde{O}\left(\sqrt{\frac{n}{d}}\right) \cdot \max_{i\in[n]} \|N_i\|$$

where the last inequality follows by Lemma 47 (b). Using the same reasoning as in Eq. (A.3) and a union bound over all $i$ we get that

$$\max_{i\in[n]} \|N_i\| \leqslant \tilde{O}\left(\sqrt{\frac{n}{d^2}}\right) \cdot \max_{k\in[n],k\neq i} \|N_{i,k}\| \leqslant \tilde{O}\left(\sqrt{\frac{n}{d^2}}\right) \cdot \tilde{O}\left(\sqrt{\frac{n}{d^2}}\right) = \tilde{O}\left(\sqrt{\frac{n^2}{d^4}}\right) = \frac{1}{\operatorname{polylog} d}$$

where the last inequality again uses a Matrix Rademacher bound (and a union bound over all $k$). Putting things together, we get that the spectral norm we wanted to bound originally is at most $\tilde{O}(\frac{n}{d^{3/2}})$.

For completeness we will also supply the proofs for the second and third category although they are very similar to the above.

**Second category.** Since we will always first multiply the $a_i$'s by random sign and then apply the decoupling theorem we will omit this step below. We will also us analogous notation. Fig. 6 shows the three cases for the second category with which we will start. For (a), the matrix looks as follows:

$$\sum_{i\in[n]} t_{1,i} \sum_{j\in[n],j\neq i} t_{2,j} \langle a_i, a_j\rangle a_j a_j \otimes \sum_{k\in[n],k\neq i,j} \langle a_i, a_k\rangle^2 (a_k^{\otimes 2})(a_k^{\otimes 2})^\top$$

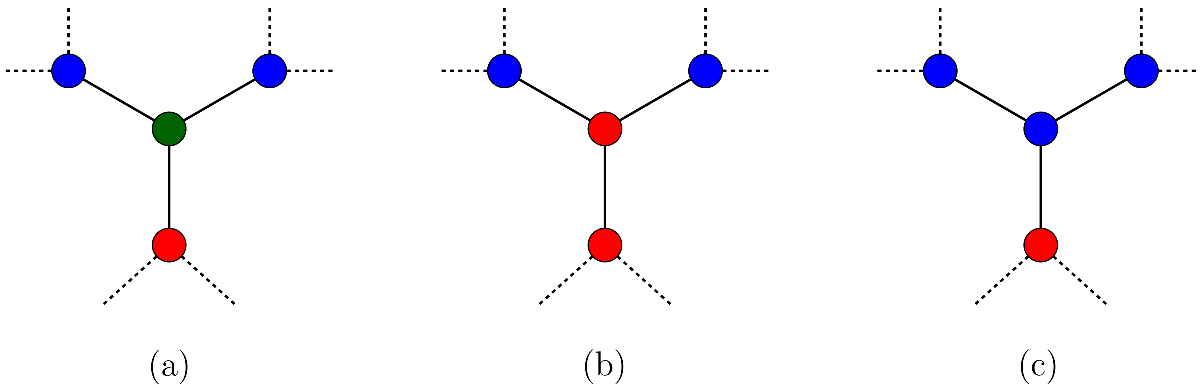


Figure 6: Two leaves share the same color but the other leaf does not.

Define $N_{i,j} \coloneqq \sum_{k\in[n],k\neq i,j}\langle a_i, a_k\rangle^2 (a_k^{\otimes 2})(a_k^{\otimes 2})^\top$ and $N_i \coloneqq \sum_{j\in[n],j\neq i} t_{2,j}\langle a_i, a_j\rangle a_j a_j \otimes N_{i,j}$. Then similarly as before, we get

$$\left\|\sum_{i\in[n]} t_{1,i} N_i\right\| \leqslant \tilde{O}\left(\sqrt{n}\right)\cdot \max_{i\in[n]}\|N_i\| \leqslant \tilde{O}\left(\sqrt{n}\right)\cdot \tilde{O}\left(\sqrt{\frac{n}{d^2}}\right)\cdot \max_{i,j\in[n],i\neq j}\|N_{i,j}\|$$

To bound the last term, we notice that for each $i \neq j$ we have that w.ov.p.

$$\|N_{i,j}\| \leqslant \max_{k\in[n],k\neq i,j}\langle a_i, a_k\rangle^2 \left\|\sum_{k\in[n],k\neq i,j}(a_k^{\otimes 2})(a_k^{\otimes 2})^\top\right\| \leqslant \tilde{O}\left(\frac{1}{d}\cdot\frac{n}{d}\right) = \tilde{O}\left(\frac{n}{d^2}\right)$$

Using a last union bound, we get that the spectral norm of the term corresponding to this coloring is at most $\tilde{O}\left(\frac{n^2}{d^3}\right) = \frac{1}{\operatorname{polylog} d}$.

For Fig. 6 (b) the matrix looks like

$$\sum_{i\in[n]} a_i a_i^\top \otimes \sum_{j\in[n],j\neq i}\langle a_i, a_j\rangle^2 (a_j^{\otimes 2})(a_j^{\otimes 2})^\top$$

Defining $N_i \coloneqq \sum_{j\in[n],j\neq i}\langle a_i, a_j\rangle^2 (a_j^{\otimes 2})(a_j^{\otimes 2})^\top$ and using Lemma 56 we can bound the spectral norm of this as

$$\left\|\sum_{i\in[n]} a_i a_i^\top\right\|\cdot \max_{i\in[n]}\|N_i\| \leqslant \tilde{O}\left(\frac{n}{d}\right)\cdot\left(\max_{i,j\in[n],i\neq j}\langle a_i, a_j\rangle^2\right)\cdot\left\|\sum_{j\in[n],j\neq i}(a_j^{\otimes 2})(a_j^{\otimes 2})^\top\right\|$$
$$\leqslant \tilde{O}\left(\frac{n}{d}\cdot\frac{1}{d}\cdot\frac{n}{d}\right) = \tilde{O}\left(\frac{n^2}{d^3}\right) = \frac{1}{\operatorname{polylog} d}$$

For Fig. 6 (c) the matrix resulting matrix is

$$\sum_{i\in[n]}(a_i^{\otimes 2})(a_i^{\otimes 2})^\top \otimes \sum_{j\in[n],j\neq i} t_{1,j}\langle a_i, a_j\rangle a_j a_j^\top$$

Again using Lemma 56 and a Matrix Rademacher bound we bound the spectral norm of this term as follows:

$$\left\|\sum_{i\in[n]}(a_i^{\otimes 2})(a_i^{\otimes 2})^\top\right\|\cdot\max_{i\in[n]}\left\|\sum_{j\in[n],j\neq i} t_{1,j}\langle a_i,a_j\rangle a_j a_j^\top\right\| \leqslant \tilde{O}\left(\frac{n}{d}\right)\cdot\left(\max_{i,j\in[n],i\neq j}\langle a_i,a_j\rangle\right)\cdot\max_{i\in[n]}\left\|\sum_{j\in[n],j\neq i} a_j a_j^\top\right\|^{1/2}$$
$$\leqslant \tilde{O}\left(\frac{n}{d}\cdot\frac{1}{\sqrt{d}}\cdot\sqrt{\frac{n}{d}}\right) = \tilde{O}\left(\sqrt{\frac{n^2}{d^3}}\right) = \frac{1}{\operatorname{polylog} d}$$

**Third category.** The last missing case is the one in the third category, where all three leaves have the same color but the internal node has a different one. In this case, the matrix we consider is

$$\sum_{i\in[n]} t_{1,i}\sum_{j\in[n],j\neq i}\langle a_i,a_j\rangle^3(a_j^{\otimes 3})(a_j^{\otimes 3})^\top$$

Using a Matrix Rademacher bound, Triangle Inequality, and Lemma 47 (c) we bound its spectral norm by

$$\tilde{O}(\sqrt{n})\cdot\max_{i\in[n]}\left\|\sum_{j\in[n],j\neq i}\langle a_i,a_j\rangle^6(a_j^{\otimes 3})(a_j^{\otimes 3})^\top\right\|^{1/2} \leqslant \tilde{O}(\sqrt{n})\cdot\max_{i,j\in[n],i\neq j}|\langle a_i,a_j\rangle|^3\cdot\max_{i\in[n]}\left\|\sum_{j\in[n],j\neq i}(a_j^{\otimes 3})(a_j^{\otimes 3})^\top\right\|$$
$$\leqslant \tilde{O}\left(\sqrt{n}\cdot\frac{1}{\sqrt{d^3}}\cdot 1\right) = \tilde{O}\left(\sqrt{\frac{n}{d^3}}\right) = \frac{1}{\operatorname{polylog} d}$$

■

### A.2. From spectral norm error to frobenius norm error

In this section our goal is to prove the second part of Theorem 4. More precisely, we will show the following lemma:

**Lemma 10** *Let $M = \sum_{i\in[n]} a_i^{\otimes 3}\left(a_i^{\otimes 3}\right)^\top + E$, where $a_1,\dots,a_n$ are i.i.d. vectors uniformly sampled from the unit sphere in $\mathbb{R}^d$ and $\|E\|\leqslant\varepsilon$. Let $\hat{M}$ be the best rank-$n$ approximation of $M$, i.e., $\hat{M} = \sum_{i\in[n]}\lambda_i v_i v_i^\top$ where $\lambda_i$'s are the top $n$ eigenvalues of $M$ and $v_i$'s are the corresponding eigenvectors. Then*

$$\hat{M} = \sum_{i\in[n]} a_i^{\otimes 3}\left(a_i^{\otimes 3}\right)^\top + \hat{E}, \quad \textit{where} \quad \|\hat{E}\|_{\mathrm{F}}\leqslant\sqrt{8n}\cdot\|E\| \quad \textit{and} \quad \|\hat{E}\|\leqslant 2\cdot\|E\|$$

**Proof** Define $S = \sum_{i\in[n]} a_i^{\otimes 3}\left(a_i^{\otimes 3}\right)^\top$, then $M = S + E$. Also, define $\hat{E} = \hat{M} - S$, then our goal will be to bound $\|\hat{E}\|$ and $\|\hat{E}\|_{\mathrm{F}}$. Since $\hat{M}$ is the best rank-$n$ approximation of $M$ we know that $\|M-\hat{M}\|\leqslant\|M-S\| = \|E\|$. We hence get

$$\|\hat{E}\| = \|\hat{M}-S\| \leqslant \|\hat{M}-M\| + \|M-S\| \leqslant 2\cdot\|E\|$$

Further, since both $S$ and $\hat{M}$ have rank $n$, the rank of $\hat{M}-S$ is at most $2n$, and it follows that

$$\|\hat{E}\|_{\mathrm{F}} = \|\hat{M}-S\|_{\mathrm{F}} \leqslant \sqrt{2n}\cdot\|\hat{M}-S\| \leqslant \sqrt{8n}\cdot\|E\|$$

■

## Appendix B. Recovering a constant fraction of the components using robust order-6 tensor decomposition

The goal of this section is to prove that in each iteration of the Recovery step in Algorithm 3, Steps 2(a) and 2(b) recover a $0.99$ fraction of the remaining components up to constant correlation. More precisely, we will show the following theorem:

**Theorem 11 (Recovery for constant fraction of component vectors)** *Let $n \leqslant O\left(d^{3/2} / \operatorname{polylog}(d)\right)$, let $a_1, a_2, \ldots, a_n \in \mathbb{R}^d$ be independently and uniformly sampled from the unit sphere, and let $\varepsilon \leqslant \frac{1}{\operatorname{polylog}(d)}$. There exists an algorithm (Algorithm 12 below) that with high probability over $a_1, a_2, \ldots, a_n$, for $d \leqslant n' \leqslant n$, for any subset $S_0 \subseteq [n]$ of size $n'$ and for a matrix $\hat{M}$ satisfying*

$$\left\| \hat{M} - \sum_{i \in S_0} a_i^{\otimes 3} \left(a_i^{\otimes 3}\right)^\top \right\|_{\mathrm{F}} \leqslant \varepsilon \sqrt{n'} = \frac{\sqrt{n'}}{\operatorname{polylog} d},$$

*returns unit vectors $b_1, b_2, \ldots, b_m \in \mathbb{R}^d$ for $m \geqslant 0.99n'$ such that for each $j \in [m]$ there exists a unique $i \in S_0$ with $\langle b_j, a_i \rangle \geqslant 0.99$.*

The algorithm looks as follows:

**Algorithm 12 (Rounding step)**
***Input:*** *A matrix* $\hat{M} \in \mathbb{R}^{d^3 \times d^3}$ *such that*

$$\left\| \hat{M} - \sum_{i \in S_0} a_i^{\otimes 3} \left(a_i^{\otimes 3}\right)^\top \right\| \leqslant \varepsilon \sqrt{n}$$

*where* $a_1, \dots, a_n$ *are i.i.d. sampled uniformly from the unit sphere,* $S_0 \subseteq [n]$ *of size* $n'$*, and* $\varepsilon = \frac{1}{\operatorname{polylog}(d)}$.
***Output:*** *A set* $S$ *of unit vectors* $b_1, \dots, b_m$ *where* $m \geqslant 0.99n'$

**Spectral truncation** *(*Corresponds to Step 2(a) of Algorithm 3*)*

*(1). Compute* $\hat{M}'$ *the projection of* $\hat{M}_{\{1,2,3,4\}\{5,6\}}$ *into the set of* $d^4 \times d^2$ *matrices with spectral norm bounded by* 1.

*(2). Compute* $M^{\leqslant 1}$ *the projection of* $\hat{M}'_{\{1,2,5,6\}\{3,4\}}$ *into the set of* $d^4 \times d^2$ *matrices with spectral norm bounded by* 1.

**Gaussian rounding**

*Initialize* $C \leftarrow \emptyset$*. Repeat* $\tilde{O}(d^2)$ *times:*

*(1). Sample* $g \sim N(0, \mathrm{Id}_{d^2})$ *and compute* $M_g = (g \otimes \mathrm{Id}_{d^2} \otimes \mathrm{Id}_{d^2})\, \mathbf{M}^{\leqslant 1}_{\{1,2\}\{3,4\}\{5,6\}}$.

*(2). Compute the top right singular vector of* $M_g$ *denoted by* $u \in \mathbb{R}^{d^2}$ *and flatten it into square matrix* $U \in \mathbb{R}^{d \times d}$.

*(3). Compute the top left and right singular vectors of* $U$ *denoted by* $v_l, v_r \in \mathbb{R}^d$.

*(4). For* $b \in \{\pm v_L, \pm v_R\}$*:*
*If* $\langle T, b^{\otimes 3} \rangle \geqslant 1 - \frac{1}{\operatorname{polylog}(n)}$
*Add* $b$ *to* $C$

*(5). For* $b \in C$*:*
*if* $\langle b, b' \rangle \geqslant 0.99$ *for all* $b' \in S$
*add* $b$ *to* $S$

*Output* $S$

We will prove Theorem 11 in several steps. Our strategy will be to apply so-called Gaussian rounding, a version of Jennrich's algorithm. However, to make this succeed in the presence of the noise matrix $E$, we will need control the spectral norm of this reshaping. In Appendix B.1 we will show that this can be done by truncating all large singular values of the respective reshapings, Concretely, we will show the following:

**Lemma 13 (Spectral truncation)** *Let* $n \leqslant O\left(d^{3/2}/\operatorname{polylog}(d)\right)$*, let* $a_1, a_2, \dots, a_n \in \mathbb{R}^d$ *be independently and uniformly sampled from the unit sphere, and let* $\varepsilon \leqslant \frac{1}{\operatorname{polylog}(d)}$*. Then, for* $d \leqslant n' \leqslant n$*, for every* $S_0 \subseteq [n]$ *of size* $n'$ *and for a matrix* $\hat{M} \in \mathbb{R}^{d^3 \times d^3}$ *satisfying*

$\|\hat{M} - \sum_{i\in S_0} a_i^{\otimes 3}\left(a_i^{\otimes 3}\right)^\top\|_{\mathrm{F}} \leqslant \varepsilon\sqrt{n'}$, *the Spectral truncation step of Algorithm 12 transforms $\hat{M}$ into tensor $\mathbf{M}^{\leqslant 1}$ such that*

- *the spectral norm of rectangular flattening is bounded by* $1$*:*

$$\left\|\mathbf{M}^{\leqslant 1}_{\{1,2,3,4\}\{5,6\}}\right\| \leqslant 1 \quad \textit{and} \quad \left\|\mathbf{M}^{\leqslant 1}_{\{1,2,5,6\}\{3,4\}}\right\| \leqslant 1,$$

- *and for $R = \sqrt{2}\cdot\left(\mathbb{E}_{a\sim N(0,\mathrm{Id}_d)}(aa^\top)^{\otimes 2}\right)^{+1/2}$, with high probability over $a_1, a_2, \ldots, a_n$, $\mathbf{M}^{\leqslant 1}$ is close to $\mathbf{S} = \sum_{i\in S_0}\left(Ra_i^{\otimes 2}\right)^{\otimes 3}$ in Frobenius norm: $\left\|\mathbf{M}^{\leqslant 1} - \mathbf{S}\right\|_F \leqslant 3\varepsilon\sqrt{n'}$.*

Given this, we will prove the correctness of the rounding part in Appendix B.2 and prove the following lemma:

**Lemma 14** *Let $n \leqslant O\left(d^{3/2}/\operatorname{polylog}(d)\right)$, let $a_1, a_2, \ldots, a_n \in \mathbb{R}^d$ be independently and uniformly sampled from the unit sphere, and let $\varepsilon \leqslant \frac{1}{\operatorname{polylog}(d)}$. Then, with high probability over $a_1, a_2, \ldots, a_n$, for $d \leqslant n' \leqslant n$ and for any $S_0 \subseteq [n]$ of size $n'$, given any $\mathbf{M}^{\leqslant 1} \in \mathbb{R}^{d^2\times d^2\times d^2}$ such that*

$$\|\mathbf{M}^{\leqslant 1} - \sum_{i\in S_0}\left((Ra_i^{\otimes 2}\right)^{\otimes 3}\|_{\mathrm{F}} \leqslant \varepsilon\sqrt{n'} \quad \textit{and} \quad \left\|\mathbf{M}^{\leqslant 1}_{\{1,2,3,4\}\{5,6\}}\right\|, \left\|\mathbf{M}^{\leqslant 1}_{\{1,2,5,6\}\{3,4\}}\right\| \leqslant 1,$$

*the Gaussian rounding step of Algorithm 12 outputs unit vectors $b_1, b_2, \ldots, b_m \in \mathbb{R}^d$ for $m \geqslant 0.99n'$ such that for each $j \in [m]$ there exists a unique $i \in S_0$ with $\langle b_j, a_i\rangle \geqslant 0.99$.*

Combining the two above theorems directly proves Theorem 11. However, there are two technical subtleties in the proof.

**Subsets of components need not be independent.** Second, it might be the case that a selected subset of the algorithm of independent random vectors are not independent. To overcome this difficulty, we instead introduce the following more general definition:

**Definition 15 (Nicely-separated vectors)** *Let $R = \sqrt{2}\left(\mathbb{E}_{a\sim N(0,\mathrm{Id}_d)}\left(aa^\top\right)^{\otimes 2}\right)^{+1/2}$. The set of vectors $a_1, a_2, \ldots, a_n'$ is called $(n,d)$-nicely-separated if all of the following are satisfied.*

1. $\left\|\sum_{i\in[n']} a_i^{\otimes 3}\left(a_i^{\otimes 3}\right)^\top\right\| = 1 \pm o(1)$
2. $\left\|\sum_{i\in[n']} a_i^{\otimes 2}\left(a_i^{\otimes 2}\right)^\top\right\| = \tilde{O}\left(\frac{n}{d}\right)$
3. $\left\|\sum_{i\in[n']} a_i a_i^\top\right\| = \tilde{O}\left(\frac{n}{d}\right)$
4. *For any $S \subseteq [n']$ with size at least $d$,*

$$\left\|\sum_{i\in S} Ra_i^{\otimes 2}\left(Ra_i^{\otimes 2}\right)^\top - \Pi\right\| = 1 \pm \tilde{O}\left(\frac{n}{d^{3/2}}\right)$$

   *, where $\Pi$ is the projection matrix into the span of $\left\{Ra_i^{\otimes 2} : i \in S\right\}$*

5. *For each* $j \in [n']$, $\sum_{i\in[n']\setminus\{j\}} \left\langle Ra_i^{\otimes 2}, Ra_j^{\otimes 2}\right\rangle^2 \leqslant \tilde{O}\left(\frac{n}{d^2}\right)$
6. *For* $i \in [n']$, $\left\|Ra_i^{\otimes 2} - a_i^{\otimes 2}\right\|^2 = \tilde{O}\left(\frac{1}{d}\right)$
7. *For* $i \in [n']$, $\|a_i\| = 1 \pm \tilde{O}\left(\frac{1}{\sqrt{d}}\right)$
8. *For* $i, j \in [n']$, $\langle a_i, a_j\rangle^2 \leqslant \tilde{O}\left(\frac{1}{d}\right)$

It can be verified that with high probability, when the component vectors are independently and uniformly sampled from the unit sphere, with high probability any subset of them is nicely-separated. In fact, we prove the following lemma in Appendix J.2.

**Lemma 16 (Satisfaction of separation assumptions)** *With probability at least* $1 - o(1)$ *over the random vectors* $a_1, a_2, \ldots, a_n \in \mathbb{R}^d$ *independently and uniformly sampled from the unit sphere, for every* $S \subseteq [n]$*, the set of vectors* $\{a_i : i \in S\}$ *is* $(n, d)$*-nicely separated.*

It is hence enough to proof Theorem 11 for the case when the subset of components indexed by $S_0$ is $(n, d)$-nicely separated.

**Isotropic components.** First, for this analysis to work we need to assume that the squared components ($a_i^{\otimes 2}$) are in isotropic position. That is, we would like to rewrite the tensor $\sum_{i\in S_0} a_i^{\otimes 6}$ as $\sum_{i\in S_0} (Ra_i^{\otimes 2})^{\otimes 3}$ where $\sqrt{2}\cdot\left(\mathbb{E}_{a\sim N(0,\mathrm{Id}_d)}\, a^{\otimes 2}\left(a^{\otimes 2}\right)^{\top}\right)^{+1/2}$. The following theorem shows that we can do this without loss of generality.

**Lemma 17** *Let* $n \leqslant O\left(d^{3/2}/\operatorname{polylog}(d)\right)$*, let* $n' \leqslant n$*, let* $a_1, a_2, \ldots, a_{n'} \in \mathbb{R}^d$ *be* $(n, d)$*-nicely-separated, and let* $R = \sqrt{2}\cdot\left(\mathbb{E}_{a\sim N(0,\mathrm{Id}_d)}\, a^{\otimes 2}\left(a^{\otimes 2}\right)^{\top}\right)^{+1/2}$*. For any tensor* $\hat{\mathbf{M}} = \sum_{i=1}^n a_i^{\otimes 6} + \mathbf{E}$ *with* $\|\mathbf{E}\|_F \leqslant \tilde{O}\left(\frac{n}{d^{3/2}}\right)\cdot\sqrt{n'}$*, we have*

$$\left\|\hat{\mathbf{M}} - \sum_{i=1}^{n'}\left(Ra_i^{\otimes 2}\right)^{\otimes 3}\right\|_F \leqslant \tilde{O}\left(\frac{n}{d^{3/2}}\right)\cdot\sqrt{n'}\,.$$

We will give a proof in Appendix J.1

### B.1. Spectral truncation

The goal of this section is to prove Lemma 13 which we restate below:

**Lemma 18 (Restatement of Lemma 13)** *Let* $n \leqslant O\left(d^{3/2}/\operatorname{polylog}(d)\right)$*, let* $a_1, a_2, \ldots, a_n \in \mathbb{R}^d$ *be independently and uniformly sampled from the unit sphere, and let* $\varepsilon \leqslant \frac{1}{\operatorname{polylog}(d)}$*. Then, for* $d \leqslant n' \leqslant n$*, for every* $S_0 \subseteq [n]$ *of size* $n'$ *and for a matrix* $\hat{M} \in \mathbb{R}^{d^3\times d^3}$ *satisfying* $\|\hat{M} - \sum_{i\in S_0} a_i^{\otimes 3}\left(a_i^{\otimes 3}\right)^{\top}\|_{\mathrm{F}} \leqslant \varepsilon\sqrt{n'}$*, the Spectral truncation step of Algorithm 12 transforms* $\hat{M}$ *into tensor* $\mathbf{M}^{\leqslant 1}$ *such that*

– *the spectral norm of rectangular flattening is bounded by* $1$*:*

$$\left\|\mathbf{M}^{\leqslant 1}_{\{1,2,3,4\}\{5,6\}}\right\| \leqslant 1 \quad \textit{and} \quad \left\|\mathbf{M}^{\leqslant 1}_{\{1,2,5,6\}\{3,4\}}\right\| \leqslant 1,$$

– *and for* $R = \sqrt{2}\cdot\left(\mathbb{E}_{a\sim N(0,\mathrm{Id}_d)}(aa^\top)^{\otimes 2}\right)^{+1/2}$*, with high probability over* $a_1, a_2, \ldots, a_n$*,* $\mathbf{M}^{\leqslant 1}$ *is close to* $\mathbf{S} = \sum_{i\in S_0}\left(Ra_i^{\otimes 2}\right)^{\otimes 3}$ *in Frobenius norm:* $\left\|\mathbf{M}^{\leqslant 1} - \mathbf{S}\right\|_F \leqslant 3\varepsilon\sqrt{n'}$*.*

**Proof** W.l.o.g. assume that $S_0 = [n']$. By Lemma 16 we know that the set $\{a_1, \ldots, a_{n'}\}$ is $(n,d)$-nicely separated. For each $i \in [n']$, we denote $b_i \coloneqq Ra_i^{\otimes 2}$. First by Lemma 17, we have

$$\left\|\hat{\mathbf{M}} - \sum_{i=1}^{n'} b_i\left(b_i^{\otimes 2}\right)^\top\right\|_{\mathrm{F}} \leqslant 2\varepsilon\sqrt{n}$$

Then by Theorem 52, with high probability we have

$$\|S_{\{1,2\},\{3\}}\| = \left\|\sum_{i=1}^{n'} b_i\left(b_i^{\otimes 2}\right)^\top\right\| \leqslant 1 + \tilde{O}\left(\frac{n}{d^{3/2}}\right)$$

We denote $\mathbf{S}' \coloneqq \frac{\mathbf{S}}{\|S_{\{1,2\},\{3\}}\|}$. Since the square flattenings of $\mathbf{S}'$ and $\mathbf{S}$ both have rank $n'$ it follows that

$$\|\mathbf{S} - \mathbf{S}'\|_{\mathrm{F}} \leqslant \tilde{O}\left(\frac{n}{d^{3/2}}\right)\cdot\sqrt{n'}$$

and $\left\|S'_{\{1,2\}\{3\}}\right\| = \left\|S'_{\{1,3\}\{2\}}\right\| = 1$.

We denote $\mathbf{E}' \coloneqq \hat{\mathbf{M}} - \mathbf{S}'$, then we have

$$\mathbf{T} = \mathbf{S} + \mathbf{E} = \mathbf{S}' + \mathbf{E}'$$

and further

$$\|\mathbf{E}'\|_F \leqslant \|\mathbf{E}\|_F + \left\|\mathbf{S}' - \mathbf{S}\right\|_{\mathrm{F}} \leqslant 2\varepsilon\sqrt{n'}$$

Denote $\mathcal{O}$ as the set of $d^2 \times d^4$ matrices with singular values at most 1. Since $S'_{\{1,2\}\{3\}} \in \mathcal{O}$, and $S'_{\{1,3\}\{2\}} \in \mathcal{O}$, we have

$$\left\|\mathbf{M}^{\leqslant 1} - \mathbf{S}'\right\|_F \leqslant \left\|\hat{\mathbf{M}}' - \mathbf{S}'\right\|_{\mathrm{F}} \leqslant \left\|\hat{\mathbf{M}} - \mathbf{S}'\right\|_{\mathrm{F}} \leqslant 2\varepsilon\sqrt{n'}.$$

And thus $\left\|\hat{\mathbf{M}}' - \mathbf{S}\right\|_{\mathrm{F}} \leqslant \|\mathbf{S} - \mathbf{S}'\|_{\mathrm{F}} + 2\varepsilon\sqrt{n'} \leqslant 3\varepsilon\sqrt{n'}$

Trivially, we then have $\|M^{\leqslant 1}_{\{1,3\}\{2\}}\| \leqslant 1$ so what remains to show is that the second projection didn't increase the spectral norm of the $\{1,2\}\{3\}$-flattening: I.e., that $\|M^{\leqslant 1}_{\{1,2\}\{3\}}\| = \|M^{\leqslant 1}_{\{1,2\}\{3\}}\| \leqslant 1$ as well. To see this, we notice the following: Let $U\Sigma V^\top$ be a SVD of $\hat{M}'_{\{1,3\}\{2\}}$ and $P = V\Theta V^\top$, where $\Theta_{i,i} = 1/\Sigma_{i,i}$ if $\Sigma_{i,i} > 1$ and 1 otherwise. Clearly, we have that $M^{\leqslant 1}_{\{1,3\}\{2\}} = \hat{M}'_{\{1,3\}\{2\}}P$. So $M^{\leqslant 1}_{\{1,2\}\{3\}}$ is obtained by starting with $\hat{M}'_{\{1,3\}\{2\}}$, switching modes 2 and 3, right-multiplying by $P$ and switching back modes 2 and 3. This is in fact equivalent to left-multiplying $(\mathrm{Id}\otimes P)$ and hence we have $\|M^{\leqslant 1}_{\{1,2\}\{3\}}\| = \|M^{\leqslant 1}_{\{1,2\}\{3\}}\| = \|(\mathrm{Id}\otimes P)\hat{M}'\| \leqslant \|\hat{M}'\|$ since the spectral norm of $P$ is at most 1. To see why this is equivalent, write $\hat{M}'$ as an $\mathbb{R}^{d^2\times d}$ matrix with $d$ blocks $B_1, \ldots, B_d \in \mathbb{R}^{d\times d}$. Exchanging modes 2 and 3 then yields the matrix with blocks $B_1^\top, \ldots B_d^\top$. So that right-multiplying with P and exchanging back modes 2 and 3 yields the matrix with $PB_1, \ldots PB_d$ which equals $(P\otimes \mathrm{Id})\hat{M}'$ (note that $P$ is symmetric). ∎

### B.2. Gaussian rounding

The goal of this section is to prove Lemma 14 which we restate below.

**Lemma 19 (Restatement of Lemma 14)** *Let $n \leqslant O\left(d^{3/2}/\operatorname{polylog}(d)\right)$, let $a_1, a_2, \ldots, a_n \in \mathbb{R}^d$ be independently and uniformly sampled from the unit sphere, and let $\varepsilon \in \frac{1}{\operatorname{polylog}(d)}$. Then, with high probability over $a_1, a_2, \ldots, a_n$, for $d \leqslant n' \leqslant n$ and for any $S_0 \subseteq [n]$ of size $n'$, given any $\mathbf{M}^{\leqslant 1} \in \mathbb{R}^{d^2 \times d^2 \times d^2}$ such that*

$$\|\mathbf{M}^{\leqslant 1} - \sum_{i \in S_0} \left((Ra_i^{\otimes 2}\right)^{\otimes 3}\|_{\mathrm{F}} \leqslant \varepsilon\sqrt{n'} \quad \text{and} \quad \left\|\mathbf{M}^{\leqslant 1}_{\{1,2,3,4\}\{5,6\}}\right\|, \left\|\mathbf{M}^{\leqslant 1}_{\{1,2,5,6\}\{3,4\}}\right\| \leqslant 1,$$

*the Gaussian rounding step of Algorithm 12 outputs unit vectors $b_1, b_2, \ldots, b_m \in \mathbb{R}^d$ for $m \geqslant 0.99n'$ such that for each $j \in [m]$ there exists a unique $i \in S_0$ with $\langle b_j, a_i\rangle \geqslant 0.99$.*

We also restate the relevant part of Algorithm 12 here:

**Algorithm 20 (Restatement of Gaussian Rounding step of Algorithm 12)**

- *Initialize $C \leftarrow \emptyset$*
- *Repeat $\tilde{O}(d^2)$ times:*
  1. *Sample $g \sim N(0, \mathrm{Id}_{d^2})$ and compute $d^2 \times d^2$ matrix $M_g = (g \otimes \mathrm{Id}_{d^2} \otimes \mathrm{Id}_{d^2})\, \mathbf{M}^{\leqslant 1}_{\{1,2\}\{3,4\}\{5,6\}}$.*
  2. *Compute the top right singular vector of $M_g$ denoted by $u \in \mathbb{R}^{d^2}$ and flatten it into square matrix $U \in \mathbb{R}^{d \times d}$.*
  3. *Compute the top left and right singular vectors of $U$ denoted by $v_l, v_r \in \mathbb{R}^d$.*
  4. *For $b \in \{\pm v_L, \pm v_R\}$:*
     *If $\langle T, b^{\otimes 3}\rangle \geqslant 1 - \frac{1}{\operatorname{polylog}(n)}$*
     *Add $b$ to $C$*
  5. *For $b \in C$:*
     *if $\langle b, b'\rangle \leqslant 0.99$ for all $b' \in S$*
     *add $b$ to $S$*
- *Output $S$.*

To prove Lemma 14 we will proceed in several steps. For the sake of presentation we will only outline the proofs and move the more technical steps to Appendix J. First, we will show that the subroutine in Step 1 in Algorithm 20 recovers one of the components up to constant correlation with probability at least $\tilde{\Theta}(d^{-2})$. Concretely, we will show the following lemma:

**Lemma 21** *Consider the setting of Lemma 14. Let $S_0 \subseteq [n]$ be of size $d \leqslant n' \leqslant n$ and assume that the set $\{a_i \mid i \in S_0\}$ is $(n,d)$-nicely separated. Consider $v_l$ and $v_r$ in Algorithm 20, then there exists a set $S \subseteq S_0$ of size $m \geqslant 0.99n'$ such that for each $i \in S$ it holds with probability $\tilde{\Theta}(d^{-2})$ that $\max_{v \in \{\pm v_l, \pm v_r\}} \langle v, a_i\rangle \geqslant 1 - \frac{1}{\operatorname{polylog}(d)}$.*

This will follow by the following sequence of lemmas. The first one show that the top singular vector of the matrix $M_g$ in Algorithm 20 is correlated with one of the components and that it further admits a spectral gap.

**Lemma 22** *Consider the setting of Lemma 14. Let $R = \sqrt{2} \cdot \left(\mathbb{E}_{a \sim \mathrm{Id}_d} \left(aa^\top\right)^{\otimes 2}\right)^{+1/2}$, let $S_0 \subseteq [n]$ be of size $n'$ where $d \leqslant n' \leqslant n$, and assume that the set $\{a_i \mid i \in S_0\}$ is $(n,d)$-nicely separated. Further, let $\hat{\mathbf{M}}$ be such that*

$$\|\mathbf{M}^{\leqslant 1} - \sum_{i \in S_0} (Ra_i^{\otimes 2})^{\otimes 3}\|_{\mathrm{F}} \leqslant \varepsilon\sqrt{n'} \quad \text{and} \quad \left\|\mathbf{M}^{\leqslant 1}_{\{1,2,3,4\}\{5,6\}}\right\|, \left\|\mathbf{M}^{\leqslant 1}_{\{1,2,5,6\}\{3,4\}}\right\| \leqslant 1.$$

*Consider the matrix $M_g = (g \otimes \mathrm{Id}_{d^2} \otimes \mathrm{Id}_{d^2})\, \mathbf{M}^{\leqslant 1}_{\{1,2\}\{3,4\}\{5,6\}}$ in Algorithm 20. Then there exists a subset $S \subseteq S_0$ of size $m \geqslant 0.99n'$, such that for each $i \in S$, and $v = Ra_i^{\otimes 2}$, with probability at least $1/d^{2(1+1/\log n)}$ over $g$, we have $M = cvv^\top + N$ where*

- $\|cvv^\top\| \geqslant (1 + \frac{1}{\log d})\|N\|$
- $\|Nv\|, \|vN\| \leqslant \varepsilon c\|v\|^2$

The proof of this lemma resembles Lemma 4.6 in Schramm and Steurer (2017), and we defer to J.3.1.

Next, we will show how to use this spectral gap to recover one of the components up to accuracy $1 - \frac{1}{\operatorname{polylog} d}$:

**Lemma 23** *Consider the setting of Lemma 14. Let $R = \sqrt{2} \cdot \left(\mathbb{E}_{a \sim \mathrm{Id}_d} \left(aa^\top\right)^{\otimes 2}\right)^{+1/2}$, let $S_0 \subseteq [n]$ be of size $d \leqslant n' \leqslant n$, and assume that the set $\{a_i \mid i \in S_0\}$ is $(n,d)$-nicely separated. Consider the matrix $M_g$ and its top right singular vector $u_r \in \mathbb{R}^{d^2}$ obtained in one iteration of Algorithm 20. Then, there exists a set $S \subseteq S_0$ with size at least $0.99n'$, such that for each $i \in S$, it holds with probability $\tilde{\Theta}(d^{-2})$ that*

- $\langle u_r, Ra_i^{\otimes 2}\rangle \geqslant 1 - \frac{1}{\operatorname{polylog} d}$.
- *the ratio between largest and second largest singular values of $M_g$ is larger than $1 + \frac{1}{\operatorname{polylog} d}$*

**Lemma 24** *Consider the setting of Lemma 14. Suppose for some unit norm vector $a \in \mathbb{R}^d$ and some unit vector $u \in \mathbf{R}^{d^2}$, $\langle u, Ra^{\otimes 2}\rangle \geqslant 1 - \frac{1}{\operatorname{polylog}(d)}$. Then flattening $u$ into a $d \times d$ matrix $U$, the top left or right singular vector of $U$ denoted by $v$ will satisfy $\langle a, v\rangle^2 \geqslant 1 - \frac{1}{\operatorname{polylog}(d)}$.*

The proof of Lemma 23 is essentially the same as Lemma 4.7 in Schramm and Steurer (2017). The proof of Lemma 24 essentially the same as Lemma 19 in Hopkins et al. (2019). We defer the proofs of these two lemmas to section J.3.2.

With this in place, it follows that the list of vectors $C = \{b_1, \ldots, b_L\}$ for $L = \tilde{O}(d^2)$ obtained by Algorithm 20 satisfies the following where $S$ is the subset of components of Lemma 21:

$$\forall i \in S\colon \max_{b \in \mathcal{C}} |\langle b, a_i\rangle| \geqslant 1 - \frac{1}{\operatorname{polylog}(d)}$$

and

$$\forall b \in C\colon\ \max_{i\in S} |\langle b, a_i\rangle| \geqslant 1 - \frac{1}{\operatorname{polylog}(d)}$$

The first equation follows by the Coupon Collector problem, Lemma 21, and the fact that we repeat the inner loop of Algorithm 20 $\tilde{O}(d^2)$ times. The second equation follows since by Lemma 58, we have $\langle T, v^{\otimes 3}\rangle \geqslant 1 - \frac{1}{\operatorname{polylog}(d)}$ if and only if $\langle v, a_i\rangle \geqslant 1 - \frac{1}{\operatorname{polylog}(d)}$.

Finally, the following lemma (proved in section J.3.3) states that Step 3 of Algorithm 20 outputs a set of vectors satisfying the conclusion of Lemma 14:

**Lemma 25** *Let $S_0 \subseteq [n]$ be of size $n' \geqslant 0.99n$ and assume that the set $\{a_i \mid i \in S_0\}$ is $(n,d)$-nicely separated. Further, let $S$ be the set of vector computed in Step 3 of Algorithm 20 and let $S'$ be the subset of components of Lemma 21. Then, for each $b \in S$, there exists a unique $i \in S'$ such that $\langle b, a_i\rangle \geqslant 1 - \frac{1}{\operatorname{polylog} d}$.*

## Appendix C. Full recovery algorithm

In the previous section, we proved that the Gaussian Rounding subroutine (Step 2(a) and Step 2(b)) in the Recovery step of Algorithm 3 recovers a $0.99$ fraction of the components. In this section, we will show how to build on this to recover all components. More precisely, we will prove Theorem 6 which we restate below.

**Theorem 26 (Restatement of Theorem 6)** *Let $a_1, \ldots, a_n$ be i.i.d. vectors sampled uniformly from the unit sphere in $\mathbb{R}^d$. For $\varepsilon = \frac{1}{\operatorname{polylog}(d)}$, given as input*

$$\mathbf{T} = \sum_{i=1}^{n} a_i^{\otimes 3} \quad \textit{and} \quad \hat{M} = \sum_{i=1}^{n} a_i^{\otimes 3}\left(a_i^{\otimes 3}\right)^{\top} + E\,, \textit{with } \|E\| \leqslant \varepsilon \textit{ and } \|E\|_{\mathrm{F}} \leqslant \varepsilon\sqrt{n}\,,$$

*Algorithm 3 returns unit norm vectors $b_1, b_2, \ldots, b_n$ satisfying*

$$\|a_i - b_{\pi(i)}\| \leqslant \tilde{O}\left(\frac{\sqrt{n}}{d}\right),$$

*for some permutation $\pi : [n] \to [n]$.*

For completeness, we also restate the relevant part of Algorithm 3 here:'

**Algorithm 27 (Restatement of the Recovery step in Algorithm 3)**

***Input:*** *A matrix $\hat{M}$ such that for some $\varepsilon = \frac{1}{\operatorname{polylog}(d)}$:*

$$\|\hat{M} - \sum_{i=1}^{n} (a_i^{\otimes 3})(a_i^{\otimes 3})^\top\| \leqslant \varepsilon \quad \text{and} \quad \|\hat{M} - \sum_{i=1}^{n} (a_i^{\otimes 3})(a_i^{\otimes 3})^\top\|_{\mathrm{F}} \leqslant \varepsilon \cdot \sqrt{n}$$

***Output:*** *Unit vectors $b_1, \ldots, b_n \in \mathbb{R}^d$.*

– *Repeat $O(\log n)$ times:*

(a) ***Pre-processing:*** *Project $\hat{M}$ into the space of matrices in $\mathbb{R}^{d^3 \times d^3}$ satisfying*

$$\left\|\hat{M}_{\{1,2,3,4\}\{5,6\}}\right\| \leqslant 1\,, \quad \left\|\hat{M}_{\{1,2,5,6\}\{3,4\}}\right\| \leqslant 1\,.$$

(b) ***Rounding:*** *Run $\tilde{O}(d^2)$ independent trials of Gaussian Rounding on $\hat{M}$ contracting its first two modes (as in Algorithm 12) to obtain a set of $0.99n$ candidate vectors $b_1, \ldots, b_{0.99n}$.*

(c) ***Accuracy boosting:*** *Boost the accuracy of each candidate $b_i$ via tensor power iteration.*

(d) ***Peeling of recovered components:***

- *Set $\hat{M}$ to be the best rank-$0.01n$ approximation of $\hat{M} - \sum_{i \leqslant 0.99n} \left(b_i^{\otimes 3}\right)\left(b_i^{\otimes 3}\right)^\top$*
- *Update $n \leftarrow 0.01n$.*

– *Return all the candidate vectors $b_1, \ldots, b_n$ obtained above.*

Our main goal will be to show that in each iteration the matrix $\hat{M}$ satisfies the assumption of Theorem 11 and then use an induction argument. To show this, we will proceed using following steps:

– By Theorem 11 we recover at least a 0.99 fraction of the remaining components up to accuracy 0.99.

– We will show that using tensor power iteration we can boost this accuracy to $1 - \tilde{O}\left(\frac{\sqrt{n}}{d}\right)$.

– In a last step we prove that after the removal step (Step 2(d)) the resulting matrix satisfies the assumptions of Theorem 11.

We will discuss the boosting step in Appendix C.1 and the removal step in Appendix C.2. In Appendix C.3 we will show how to combine the two to prove Theorem 6.

### C.1. Boosting the recovery accuracy by tensor power iteration

Given the relatively coarse estimation of part of the components, we use tensor power iteration in Anandkumar et al. (2015) to boost the accuracy.

**Lemma 28 (Lemma 2 in Anandkumar et al. (2015))** *Let $T = \sum_{i=1}^n a_i^{\otimes 3}$, where $a_1, a_2, \dots, a_n$ are independently and uniformly sampled from $d$-dimensional unit sphere. Then with high probability over $a_1, a_2, \dots, a_n$, for any unit norm vector $v$ such that $\langle v, a_1 \rangle \geqslant 0.99$, , the tensor power iteration algorithm gives unit norm vector $b_1$ such that $\langle a_1, b_1 \rangle \geqslant 1 - \tilde{O}\left(\frac{n}{d^2}\right)$ and runs in $\tilde{O}(d^3)$ time.*

By running tensor power iteration on the vectors obtained in the last subsection, we thus get the following guarantee:

**Corollary 29** *Given tensor $T = \sum_{i=1}^n a_i^{\otimes 3}$, where $a_1, a_2, \dots, a_n$ are independently and uniformly sampled from $d$-dimensional unit sphere. Suppose for a set $S \subseteq [n]$ with size $m$, we are given vectors $b_1, b_2, \dots, b_m$ such that for each $i \in S$,*

$$\max_{j \in [m]} \langle a_i, b_j \rangle \geqslant 0.99$$

*Then in $\tilde{O}(nd^3)$ time, we can get unit norm vectors $c_1, c_2, \dots, c_m$ s.t for each $i \in S$,*

$$\max_{j \in [m]} \langle a_i, c_j \rangle \geqslant 1 - \tilde{O}\left(\frac{n}{d^2}\right) .$$

### C.2. Removing recovered components

In this part, we mainly prove that we can remove the recovered components as in Step 2(d) of Algorithm 3, without increasing spectral norm of noise by more than poly $\left(\frac{n}{d^{3/2}}\right)$.

**Lemma 30** *Let $m \geqslant d$ and $d \leqslant n = O\left(d^{3/2} / \operatorname{polylog}(d)\right)$. Let $a_1, a_2, \dots, a_n \in \mathbb{R}^d$ be i.i.d random unit vectors sampled uniform from the sphere. Then with high probability over $a_1, a_2, \dots, a_n$, for any $S = \{s_1, s_2, \dots, s_m\} \subseteq [n]$, and $b_1, b_2, \dots, b_m$ satisfying $\|a_{s_i} - b_i\| \leqslant \tilde{O}\left(\sqrt{n}/d\right)$, we have*

$$\left\| \sum_{i \in S} \left(a_i^{\otimes 3}\right)\left(a_i^{\otimes 3}\right)^\top - \sum_{i \in [m]} \left(b_i^{\otimes 3}\right)\left(b_i^{\otimes 3}\right)^\top \right\| \leqslant \frac{1}{\operatorname{polylog}(d)}$$

We first prove the same result under the deterministic assumption that $\{a_i : i \in S\}$ are $(n, d)$ nicely-separated. Then combining with Lemma 16, the Lemma 30 follows as a corollary.

**Lemma 31** *Let $m \geqslant d$ and $d \leqslant n = O\left(d^{3/2} / \operatorname{polylog}(d)\right)$. Let $a_1, a_2, \dots, a_m \in \mathbb{R}^d$ be vectors satisfying the $(n, d)$ nicely-separated assumptions of Theorem 15. Suppose unit norm vectors $b_1, b_2, \dots, b_m$ satisfies that $\|a_i - b_i\| \leqslant O\left(\sqrt{n}/d\right)$. Then we have*

$$\left\| \sum_{i \in [m]} \left(a_i^{\otimes 3}\right)\left(a_i^{\otimes 3}\right)^\top - \sum_{i \in [m]} \left(b_i^{\otimes 3}\right)\left(b_i^{\otimes 3}\right)^\top \right\| \leqslant \frac{1}{\operatorname{polylog}(d)}$$

**Proof** We denote the matrix $U \in \mathbb{R}^{d^3 \times m}$ with the $i$-th column given by $a_i^{\otimes 3}$, and the matrix $V \in \mathbb{R}^{d^3 \times n}$ with the $i$-th column given by $b_i^{\otimes 3}$. Then

$$\left\| \sum_{i \in [m]} \left(a_i^{\otimes 3}\right)\left(a_i^{\otimes 3}\right)^\top - \sum_{i \in [m]} \left(b_i^{\otimes 3}\right)\left(b_i^{\otimes 3}\right)^\top \right\| = \left\| UU^\top - VV^\top \right\|$$

Now since $\left\|UU^\top - VV^\top\right\| = \left\|(U-V)U^\top + V(U-V)^\top\right\|$ and $\|U\| \leqslant 1$ with high probability, it's suffcient to show that $\|U-V\| \leqslant \frac{n^2}{d^3}$, which is equivalent to $\sqrt{\|(U-V)^\top(U-V)\|}$.

We denote $W := (U-V)^\top(U-V)$, and let $W = W_1 + W_2$ where $W_1$ be the diagonal part of the matrix $W$ and $W_2$ be the non-diagonal part. Then for $i \in [n]$, the diagonal entries of $W$ are given by

$$W_{ii} = \left(a_i^{\otimes 3} - b_i^{\otimes 3}\right)^\top \left(a_i^{\otimes 3} - b_i^{\otimes 3}\right) = \left\|a_i^{\otimes 3} - b_i^{\otimes 3}\right\|^2$$

Now since

$$\left\|a_i^{\otimes 3} - b_i^{\otimes 3}\right\|^2 \leqslant 2 - 2\langle a_i, b_i\rangle^3 = 2 - 2\cdot\frac{\left(2-\|a_i-b_i\|^2\right)^3}{8} \leqslant 2 - (2 - 6\cdot\|a_i - b_i\|) = 6\cdot\|a_i-b_i\|^2$$

it follows that $\left\|a_i^{\otimes 3} - b_i^{\otimes 3}\right\| \leqslant \tilde{O}(\sqrt{n}/d)$. Since $W_1$ is a diagonal matrix, we have $\|W_1\| \leqslant \tilde{O}(\sqrt{n}/d)$.

Next we bound $\|W_2\|_F$. We denote $c_i = a_i - b_i$. Then by assumption we have $\|c_i\| \leqslant O\left(\sqrt{n}/d\right)$. Now we have

$$\begin{aligned}\langle a_i^{\otimes 3} - b_i^{\otimes 3}, a_j^{\otimes 3} - b_j^{\otimes 3}\rangle &= \langle a_i^{\otimes 3} - (a_i + c_i)^{\otimes 3}, a_j^{\otimes 3} - (a_j + c_j)^{\otimes 3}\rangle \\ &= \sum_{\substack{g_i^{(1)}, g_i^{(2)}, g_i^{(3)} \\ g_j^{(4)}, g_j^{(5)}, g_j^{(6)}}} \langle g_i^{(1)}, g_j^{(4)}\rangle\langle g_i^{(2)}, g_j^{(5)}\rangle\langle g_i^{(3)}, g_j^{(6)}\rangle\end{aligned}$$

where for $k \in [6]$ and $i \in [m]$, $g_i^{(k)} \in \{a_i, c_i\}$. Now we rewrite $W_2 = \sum_g M_g$, where $M_{g,i,j} = \langle g_i^{(1)}, g_j^{(4)}\rangle\langle g_i^{(2)}, g_j^{(5)}\rangle\langle g_i^{(3)}, g_j^{(6)}\rangle$. Since there are less than $2^3$ choices for $g = \left\{g^{(1)}, g(2), \ldots, g(6)\right\}$, By Lemma 75, for every choice of $g$, we have $\|M_g\|_{\mathrm{F}} \leqslant \tilde{O}\left(\sqrt{\frac{n}{d^{3/2}}}\right) \leqslant \frac{1}{\mathrm{polylog}(d)}$. By applying triangle inequality, we have $\|W_2\|_F \leqslant \frac{1}{\mathrm{polylog}(d)}$.

It follows that

$$\left\|(U-V)^\top(U-V)\right\| = \|W\| \leqslant \|W_1\| + \|W_2\| \leqslant \frac{1}{\mathrm{polylog}(d)},$$

which concludes the proof. ■

### C.3. Putting things together

**Proof** [Proof of Theorem 6] We show that, if the events in Theorem 11, Theorem 29, and Lemma 30 happen, then Algorithm 3 returns unit norm vectors $b_1, b_2, \ldots, b_n$ satisfying

$$\|a_i - b_{\pi(i)}\| \leqslant \tilde{O}\left(\frac{\sqrt{n}}{d}\right),$$

for some permutation $\pi : [n] \to [n]$. Since by Theorem 11, Theorem 29, and Lemma 30, these events happen with high probability over random unit vectors $a_1, a_2, \ldots, a_n$, the theorem thus follows.

Let $\delta = \frac{1}{\log^{10}(n)}$, and $n \leqslant d^{3/2}/\log^{10000} n$. For $t \leqslant O(\log n)$, we prove by mathematical induction that after $t$-th iteration of the Recovery step in Algorithm 3, for a subset $S_t \subseteq [n]$, we have

$$\left\| M - \sum_{i \in S} a_i^{\otimes 3} \left(a_i^{\otimes 3}\right)^\top \right\| \leqslant (t+1)\delta\,.$$

Further we have $S_{t+1} \leqslant 0.01 S_t$

As base case after the Lifting step of Algorithm 3, we have

$$\left\| M - \sum_{i=1}^{n} a_i^{\otimes 3} \left(a_i^{\otimes 3}\right)^\top \right\| \leqslant \tilde{O}\left(\frac{n}{d^{3/2}}\right) \leqslant \delta\,.$$

For induction step, we suppose for some $S_t \subseteq [n]$,

$$\left\| M - \sum_{i \in S_t} a_i^{\otimes 3} \left(a_i^{\otimes 3}\right)^\top \right\| \leqslant t\delta\,.$$

Since we condition that the statement in Theorem 11 holds, for some $m \geqslant 0.99\,|S_t|$ and $S_t' \subseteq n$ with size $m$, Step 2(b) of Algorithm 3 outputs unit norm vectors $b_1, b_2, \ldots, b_m$ such that for each $i \in S_t'$,

$$\max_{j \in [m]} \langle a_i, b_j \rangle \geqslant 1 - \frac{1}{\operatorname{polylog}(d)}$$

Then combining Theorem 29 and Lemma 30, before Step 2(d) of $t$-th iteration of the Recovery step, we have

$$\left\| \sum_{i \in S_t'} a_i^{\otimes 3} \left(a_i^{\otimes 3}\right)^\top - \sum_{i \in [m]} b_i^{\otimes 3} \left(b_i^{\otimes 3}\right)^\top \right\| \leqslant \delta$$

By triangle inequality, after removal step (d), it follows that

$$\left\| M - \sum_{i \in S_t \setminus S_t'} a_i^{\otimes 3} \left(a_i^{\otimes 3}\right)^\top \right\| \leqslant (t+1)\delta$$

By setting $S_{t+1} = S_t \setminus S_t'$, we have $|S_{t+1}| \leqslant 0.01\,|S_t|$, and

$$\left\| M - \sum_{i \in S_{t+1}} a_i^{\otimes 3} \left(a_i^{\otimes 3}\right)^\top \right\| \leqslant (t+1)\delta$$

The induction step is thus finished.

Now putting the recovery vectors obtained in all the iterations, we finish the proof. ■

## Appendix D. Implementation and running time analysis

We prove here Theorem 7 concerning the running time of Algorithm 3.

**Remark 32 (On the bit complexity of the algorithm)** *We assume that the vectors $a_1, \ldots, a_n \in \mathbb{R}^d$ have polynomially (in the dimension) bounded norm. We can then represent each of the vectors, matrices and tensor considered to polynomially small precision with logarithmically many bits (per entry). This representation does not significantly impact the overall running time of the algorithm, while also not invalidating its error guarantees (with high probability). For this reason we ignore the bit complexity aspects of the problem.*

### D.1. Running time analysis of the lifting step

For a matrix $A \in \mathbb{R}^{d\times d}$, we say that $L$ is the best rank-$m$ approximation of $A$ if

$$L = \arg\min \left\{ \|A - L\|_{\mathrm{F}} \;\middle|\; L \in \mathbb{R}^{d\times d}, \operatorname{rank}(L) \leqslant m \right\} .$$

We will consider the following algorithm:

**Algorithm 33 (Compute implicit representation)**

***Input:*** *Tensor* $\mathbf{T} = \sum\limits_{i\in[n]} a_i^{\otimes 3}$ .

***Output:*** $U, V \in \mathbb{R}^{d^3\times n}$.

1. *Use the $n$-dimensional subspace power method Hardt and Price (2014) on the $\{1,2,3\}\,\{4,5,6\}$ flattening of*

$$\mathbf{M} = \sum_{i,j,k,\ell\in[n]} \langle a_i, a_j\rangle \cdot \langle a_i, a_k\rangle \cdot \langle a_i, a_\ell\rangle \cdot (a_j {a_j}^{\mathsf{T}}) \otimes (a_k {a_k}^{\mathsf{T}}) \otimes (a_\ell {a_\ell}^{\mathsf{T}}) \,, \tag{D.1}$$

   *decomposing contractions with $\mathbf{M}_{\{1,2,3\}\{4,5,6\}}$ as shown in Fig. 4 and using the fast rectangular matrix multiplication algorithm of Gall and Urrutia (2018).*

2. *Return $U, V \in \mathbb{R}^{d^3\times n}$ computed from the resulting (approximate) $n$ eigenvectors and eigenvalues.*

**Lemma 34** *Let $a_1, \ldots, a_n$ be i.i.d. vectors uniformly sampled from the unit sphere in $\mathbb{R}^d$. Consider the flattening $\mathbf{M}_{\{1,2,3\}\{4,5,6\}}$ of $\mathbf{M}$ as in Eq. (D.1). Let $U'\Sigma' U'^{\mathsf{T}}$ with $U' \in \mathbb{R}^{d^3\times n}, \Sigma' \in \mathbb{R}^{n\times n}$, be its best rank-$n$ approximation. Then, there exists an algorithm (Algorithm 33) that, given $\mathbf{T}$, computes $U, V \in \mathbb{R}^{d^3\times n}$ such that*

$$\left\| UV^{\mathsf{T}} - U'\Sigma U'^{\mathsf{T}} \right\|_{\mathrm{F}} \leqslant d^{-100} \,.$$

*Moreover, the algorithm runs in time $\tilde{O}\left(d^{2\cdot\omega(1+\log n/2\log d)}\right)$, where $\omega(k)$ is the time required to multiply a $(d^k \times d)$ matrix with a $(d\times d)$ matrix.*[10]

10. See Fig. 7 in Appendix I.

**Proof** *It suffices to show how to approximately compute the top $n$ eigenvectors and eigenvalues of $\mathbf{M}_{\{1,2,3\}\{4,5,6\}}$ as then deriving $U, V$ from there is trivial.*

*We start by explaining how to use the structure of the tensor network to multiply $\mathbf{M}$ by a vector $v$ more efficiently, then extend this idea to the subspace power method Hardt and Price (2014), and finally apply the rectangular matrix multiplication method Gall and Urrutia (2018).*

*To efficiently multiply a vector $v \in \mathbb{R}^{d^3}$ by $\mathbf{M}_{\{1,2,3\}\{4,5,6\}}$, we partition the multiplication into four steps by cutting the tensor network "cleverly." Fig. 4 presents the four-step multiplication. The multiplication time is $O(d^{2w})$ as explained as following. Step (a) multiplies a $d^2 \times d$ matrix with a $d \times d^2$ matrix, and thus takes $O\left(d^{1+\omega}\right)$ time. Step (b) multiplies a $d^2 \times d$ matrix with a $d \times d^2$ matrix, and thus takes $O\left(d^{1+\omega}\right)$ time. Step (c) multiplies a $d^2 \times d^2$ matrix with a $d^2 \times d^2$ matrix, and thus takes $O\left(d^{2\omega}\right)$ time. Step (d) multiplies a $d^2 \times d^2$ matrix with $d^2 \times d$ matrix, and thus takes $O\left(d^{2+\omega}\right)$ time.*

*Each iteration of the subspace power method Hardt and Price (2014) multiplies $n$ vectors by $\mathbf{M}_{\{1,2,3\}\{4,5,6\}}$ simultaneously. Therefore, $v$ in the above 4-step multiplication is replaced with a $d^3 \times n$ matrix. Then, Step (a) becomes multiplying a $nd^2 \times d$ matrix with a $d \times d^2$ matrix, Step (c) becomes multiplying a $nd^2 \times d^2$ matrix with a $d^2 \times d^2$ matrix, and Step (d) becomes multiplying a $nd^2 \times d^2$ matrix with $d^2 \times d$ matrix.*

*The rectangular multiplication algorithm Gall and Urrutia (2018) takes $O(d^{w(k)})$ time to multiply a $d^k \times d$ matrix by a $d \times d^k$ matrix. Note that the time complexities of the following three problems are the same: multiplying a $d^k \times d$ matrix by a $d \times d^k$ matrix, multiplying a $d \times d$ matrix by a $d^k \times d$ matrix, and multiplying a $d \times d^k$ matrix by a $d \times d$ matrix. By the rectangular multiplication algorithm, Step (a) takes $O\left(n \cdot d^{2 \cdot \omega(0.5)}\right) = O\left(n \cdot d^{4.093362}\right)$ time, Step (c) takes $O\left(d^{2 \cdot \omega(\log_{d^2}(d^2 n))}\right) = O\left(d^{2 \cdot \omega(1 + \log n / \log d^2)}\right)$ time, and Step (d) takes$O\left(nd \cdot d^{\omega(2)}\right) = O\left(n \cdot d^{4.256689}\right)$ time.*

*Since the time of Step (c) dominates that of Step (a), Step (b) and Step (d), one iteration of the subspace power method takes $O\left(n \cdot d^{2 \cdot \omega(1 + \log n / \log d^2)}\right)$ time. By Lemma 9, $\lambda_{n+1}/\lambda_n \leqslant 1/\operatorname{polylog} d$, so the subspace power method takes $\operatorname{polylog} d$ iterations. To conclude, computing the top $n$ eigenvectors of $\mathbf{M}_{\{1,2,3\}\{4,5,6\}}$ takes $\tilde{O}\left(d^{2 \cdot \omega(1 + \log n / \log d^2)}\right)$ time.* ■

### D.2. Running time analysis for the pre-processing step

In this section we show that the implicit representation of tensor $M^{\leqslant 1}$ in Lemma 13 can be computed in a fast way. By Lemma 17 we may assume our matrix $UV^\mathsf{T}$ is close to a matrix flattening of $\sum_{i=1}^n \left(Ra_i^{\otimes 2}\right)^{\otimes 3}$, where $R = \sqrt{2} \cdot \left(\mathbb{E}_{a \sim N(0, \mathrm{Id}_d)} a^{\otimes 2} \left(a^{\otimes 2}\right)^\top\right)^{+1/2}$.

**Lemma 35 (Running time of the pre-processing step)** *Let $a_1, \ldots, a_n$ be a subset of i.i.d. vectors uniformly sampled from the unit sphere in $\mathbb{R}^d$. Let $R = \sqrt{2} \cdot \left(\mathbb{E}_{a \sim N(0, \mathrm{Id}_d)} a^{\otimes 2} \left(a^{\otimes 2}\right)^\top\right)^{+1/2}$ and denote $\mathbf{S}_3 = \sum_{i=1}^n \left(Ra_i^{\otimes 2}\right)^{\otimes 3}$. There exists an algorithm that, given matrices $U, V \in \mathbb{R}^{d^3 \times n}$ satisfying*

$$\|UV^\mathsf{T} - (\mathbf{S}_3)_{\{1,2,3\}\{4,5,6\}}\|_{\mathrm{F}} \leqslant \varepsilon \sqrt{n}\,,$$

*computes matrices $U', V' \in \mathbb{R}^{d^3 \times 2n}$ satisfying*

$$\|U'V'^\mathsf{T} - (\mathbf{S}_3)_{\{1,2,3\}\{4,5,6\}}\|_{\mathrm{F}} \leqslant \varepsilon \sqrt{n}\,, \quad \|(U'V'^\mathsf{T})_{\{5,6\}\{1,2,3,4\}}\| \leqslant 1\,, \quad \|U'V'^\mathsf{T}_{\{3,4\}\{1,2,5,6\}}\| \leqslant 1\,.$$

*Moreover, the algorithm runs in time* $\tilde{O}(d \cdot n^{\omega\left(\frac{2\log d}{\log n}\right)} + nd^4) \leqslant \tilde{O}(d^{5.05} + nd^4)$.

The algorithm used to compute these fast projections consists of two subsequent application of the following procedure (symmetrical with respect to the two distinct flattenings).

**Algorithm 36 (Fast projection)**

***Input:*** *Matrices* $U, V \in \mathbb{R}^{d^3 \times n}$.
***Output:*** *Matrices* $U', V' \in \mathbb{R}^{d^3 \times n}$.

1. *Denote* $N = (UV^{\mathsf{T}})_{\{5,6\}\{1,2,3,4\}}$.
2. *Compute the* $nd \times d^2$ *reshaping* $Z$ *and the* $d^2 \times nd$ *reshaping* $\tilde{V}$ *of* $V$.
3. *Compute* $W = Z^{\mathsf{T}}(U^{\mathsf{T}}U \otimes \mathrm{Id}_d)Z$.
4. *Compute* $H = (\mathrm{Id}_{d^2} - W^{-1/2})^{>0}$.
5. *Compute* $L = \tilde{V}^{\mathsf{T}}H$.
6. *Reshape* $L$ *and compute* $N^{\leqslant 1} = U'V'^{\mathsf{T}} = UV^{\mathsf{T}} - U(L \otimes \mathrm{Id}_d)$.
7. *Return the resulting matrices* $U', V'$.

Before presenting the proof, we first introduce some notation:

**Definition 37** *For arbitrary matrix* $M \in \mathbb{R}^{d\times d}$ *with eigenvalue decomposition* $M = U\Sigma U^\top$, *we denote* $M^{>t} := U\Sigma^{>t}U^\top$, *where* $\Sigma^{>t}$ *is same as* $\Sigma$ *except for truncating entries larger than* $t$ *to* $0$.

Next we prove that the spectral truncation can be done via matrix multiplication.

**Lemma 38** *Consider matrices* $N \in \mathbb{R}^{d^4\times d^2}$ *and* $M := N^\top N$. *Then* $N^{\leqslant 1} := N\left(\mathrm{Id}_{d^2} - \left(\mathrm{Id}_{d^2} - M^{-1/2}\right)^{>0}\right)$ *is the projection of* $N$ *into the set of* $d^4 \times d^2$ *matrices with spectral norm bounded by* $1$

**Proof** Indeed suppose $N$ has singular value decomposition $N = P\Sigma Q^\top$, then $M^{-1/2} = Q\tilde{\Sigma}^{-1}Q^\top$, where $\Sigma$ is a $d^4 \times d^2$ diagonal matrix and $\tilde{\Sigma} = (\Sigma^\top\Sigma)^{1/2}$. It follows that

$$
\begin{aligned}
N\left(\mathrm{Id}_{d^2} - \left(\mathrm{Id}_{d^2} - M^{-1/2}\right)^{>0}\right) &= P\Sigma Q^\top\left(\mathrm{Id}_{d^2} - Q\left(\mathrm{Id}_{d^2} - \tilde{\Sigma}^{-1}\right)^{>0}Q^\top\right) \\
&= P\Sigma\left(\mathrm{Id}_{d^2} - \left(\mathrm{Id}_{d^2} - \tilde{\Sigma}^{-1}\right)^{>0}\right)Q^\top \\
&= P\Sigma' Q^\top
\end{aligned}
$$

where $\Sigma' := \Sigma\left(\mathrm{Id}_{d^2} - \left(\mathrm{Id}_{d^2} - \tilde{\Sigma}^{-1}\right)^{>0}\right)$. Now we note that for each $i$, if $\Sigma_{ii} > 1$, then $\Sigma'_{ii} = \Sigma_{ii} \cdot \Sigma_{ii}^{-1} = 1$; otherwise $\Sigma'_{ii} = \Sigma_{ii}$. Therefore $P\Sigma' Q^\top$ is exactly the projection of $N$ into the set of $d^4 \times d^2$ matrices with spectral norm bounded by $1$. ∎

We are now ready to prove Lemma 35.

**Proof** [Proof of Lemma 35] Without loss of generality, we consider the flattening $\hat{M}_{\{5,6\},\{1,2,3,4\}}$. For simplicity, we denote $N := \hat{M}_{\{1,2,3,4\},\{5,6\}}$. Let $Z$ be an appropriate $nd \times d^2$ reshaping of $V$. Since for any vector $y \in \mathbb{R}^{d^2}$, we have that $Ny$ is the flattening of $UV^\top(y \otimes \mathrm{Id}_d)$ into a $d^4$ dimensional vector and $Ny = (U \otimes \mathrm{Id}_d)Zy$, it follows that $N = (U \otimes \mathrm{Id}_d)Z$. Further, we denote $W := N^\top N = Z^\top \left(U^\top U \otimes \mathrm{Id}_d\right) Z$. Then the $i$-th singular value of $W$ is given by the square of the $i$-th singular value of $N$.

We show that matrix $W$ can be computed in a fast way. Since $U \in \mathbb{R}^{d^3 \times n}$, we can compute $U^\top U$ in time $n^{\omega\left(\frac{3\log d}{\log n}\right)}$. When $n \leqslant d^{3/2}$, this is bounded by $d^{\frac{3}{2}\omega(2)} \leqslant d^5$. Then since $U^\top U$ is an $n \times n$ matrix, and $Z$ is a $nd \times d^2$ matrix, $\left(U^\top U \otimes \mathrm{Id}_d\right) Z$ requires $d$ distinct multiplications each between an $n \times n$ and an $n \times d^2$ matrices. Each of these multiplications takes time $O\left(n^{\omega(\frac{2\log d}{\log n})}\right)$. When $n \leqslant d^{3/2}$, this is bounded by $O(d^{5.05})$.

By Lemma 38, the projection matrix is given by $N^{\leqslant 1} = N\left(\mathrm{Id}_{d^2} - \left(\mathrm{Id}_{d^2} - W^{-1/2}\right)^{>0}\right)$. Now we claim that with high probability the matrix $\left(\mathrm{Id}_{d^2} - W^{-1/2}\right)^{>0}$ has rank at most $n$. Indeed since matrix $N$ has Frobenius norm at most $2\sqrt{n}$, it has at most $2n$ eigenvalues at least 1. Since $W = N^\top N$, it has at most $2n$ eigenvalues at least 1 as well. We then can compute the eigenvalue decomposition $H := \left(\mathrm{Id}_{d^2} - W^{-1/2}\right)^{>0} = P\Lambda^{-1/2}P^\top$ in time $O(nd^4)$.

Using this low rank representation, we show that we can compute matrices $U', V' \in \mathbb{R}^{d^3 \times n}$ such that $N^{\leqslant 1} = UV^\top - U'V'^\top$. Indeed, since $N^{\leqslant 1} = UV^\top - UV^\top(H \otimes \mathrm{Id}_d)$, it's sufficient to calculate $V^\mathsf{T}(H \otimes \mathrm{Id}_d)$. For this, we first reshape $V$ into a $d^2 \times nd$ matrix $\tilde{V}$ and then do the matrix multiplication $\tilde{V}^\top H = \tilde{V}^\top P\Lambda^{-1/2}P^\top$. Tthen we can reshape $\tilde{V}^\top H$ into an appropriate $d^3 \times n$ matrix $V'$. For $U' = U$ we then have $UV^\top(H \otimes \mathrm{Id}_d) = U'V'^\top$. Since $P \in \mathbb{R}^{d^2 \times n}$ and $\tilde{V} \in \mathbb{R}^{d^2 \times nd}$, when $n \leqslant d^{3/2}$, it takes time $O(d \cdot n^{\omega(4/3)}) \leqslant d^5$.

All in all, the total running time is bounded by $O(d^{5.05} + nd^4)$.

∎

### D.3. Running time analysis of Gaussian rounding

**Lemma 39 (Running time of the rounding step)** *In each iteration of the recovery step in algorithm Algorithm 3, the rounding step takes time at most* $O\left(n \cdot d^4 + d^{\omega\left(\frac{1}{2} + \frac{\log n}{2\log d}\right)}\right) \leqslant O(n \cdot d^4 + d^{5.25})$.

**Proof** We divide the discussion in three steps.

**Running time for a random contraction and taking top eigenvectors** We sample $\ell = \tilde{O}(d^2)$ independent random Gaussian vectors $g_1, g_2, \ldots, g_\ell \sim N(0, \mathrm{Id}_{d^2})$. In Algorithm 12, we use power method to obtain the top right singular vectors of $M_t(g)$ for all $t \in [\ell]$. We first take random initialization vectors $x_1, x_2, \ldots, x_\ell$. Then we do $\tilde{O}(1)$ power iterations. In each iteration, we update $x_i \leftarrow (x_i \otimes \mathrm{Id} \otimes g_i)\hat{\mathbf{M}}$.

Since for arbitrary vectors $x_1, x_2, \ldots, x_\ell \in \mathbb{R}^{d^2}$, by Lemma 61, we can obtain $(x_i \otimes \mathrm{Id} \otimes g_i)\hat{\mathbf{M}}$ for $i \in [\ell]$ in $O\left(n \cdot d^4 + d^{2\omega\left(\frac{1+\log_d n}{2}\right)}\right)$ time. Thus combining all iterations, the total running time is bounded by $\tilde{O}\left(n \cdot d^4 + d^{2\omega(5/4)}\right) \leqslant \tilde{O}\left(n \cdot d^4 + d^{5.25}\right)$ time.

Next we show it's sufficient to run $\tilde{O}(1)$ power iterations to get accurate approximation of top singular vectors. Consider the setting of Lemma 23. Suppose the matrix $(\mathrm{Id}_{d^2} \otimes \mathrm{Id}_{d^2} \otimes g_i)\hat{\mathbf{M}}$ satisfy the conditions that

– the top singular vector $u$ recovers some component vector $a_i$: $\left|\langle u, a_i^{\otimes 2}\rangle\right| \geqslant 1 - \frac{1}{\operatorname{polylog} d}$

– the ratio between the largest and second largest singular value of $M_g$ is larger than $1/\log\log n$.

Then by the second condition, after $\operatorname{polylog}(n)$ power iterations, we will get $|\langle x_i, u\rangle| \geqslant 1 - \frac{1}{\operatorname{polylog}(n)}$.

Then for these top eigenvectors, we flatten them into $d \times d$ matrices $B_1, B_2, \ldots, B_\ell \in \mathbb{R}^{d^2}$, and then take top singular vectors of these matrices. This takes time at most $\tilde{O}\left(\ell \cdot d^2\right) = \tilde{O}\left(d^4\right)$. As a result, we obtain $O(\ell)$ candidate recovery vectors.

**Running time for checking candidate recovery vectors** In Algorithm 12 for each of the $\ell$ candidate recovery vectors $v$, we check the value of $\langle T, v^{\otimes 3}\rangle$. This requires $\tilde{O}(\ell \cdot d^3) = \tilde{O}(d^5)$ time.

**Running time for removing redundant vectors** We consider the running time of , which is a detailed exposition of the relevant step in Algorithm 12. In each of the $\tilde{O}(d^2)$ iterations, we need to check the correlation of $b_i$ with each vector in $S'$. Since $S'$ has size at most $n$, this takes time at most $O(nd)$. Therefore the total running time is bounded by $\tilde{O}(nd^3)$.

Thus in all the running time is given by $\tilde{O}\left(n \cdot d^4 + d^{5.25}\right)$. ∎

### D.4. Running time analysis of accuracy boosting

**Lemma 40** *In each iteration of the Recovery step in algorithm Algorithm 3, the accuracy boosting step takes time at most* $\tilde{O}(n \cdot d^3)$.

**Proof** In each iteration we perform the accuracy boosting step for at most $0.99n$ vectors. For each such vector we need to run $O(\log d)$ rounds of tensor power iterationsAnandkumar et al. (2015). Since each round of tensor power iteration takes $\tilde{O}(d^3)$ time, the total running time is bounded by $\tilde{O}(n \cdot d^3)$. ∎

### D.5. Running time analysis of peeling

The last operation in each iteration of Recovery step in algorithm Algorithm 3 consists of "peeling off" the components just learned and obtain an implicit representation of the modified data. $UV^\top - \sum_{i=1}^{0.99n} b_i^{\otimes 3}\left(b_i^{\otimes 3}\right)^\top$, and obtain the implicit representation.

**Lemma 41** *Let $\varepsilon, \delta > 0$ and let $m < n$ be positive integers. Let $a_1, \ldots, a_n \in \mathbb{R}^d$ be any subset of i.i.d. vectors uniformly sampled from the unit sphere in $\mathbb{R}^d$. Let $U, V \in \mathbb{R}^{d^3 \times n}$ be such that*

$$\left\| UV^\mathsf{T} - \sum_{i\in[n]} (a_i^{\otimes 3})(a_i^{\otimes 3})^\mathsf{T} \right\| \leqslant \varepsilon\,.$$

*Let $b_1 \ldots, b_m \in \mathbb{R}^d$ be such that*

$$\forall i \in [m]\,, \quad \langle a_i, b_i\rangle \geqslant 1 - 1/\operatorname{polylog}(d)\,.$$

*Then there exists an algorithm (a slight variation of Algorithm 33) that, given $b_1\,, \ldots, b_m\,, U, V$, computes $U', V' \in \mathbb{R}^{d^3 \times n-m}$ satisfying*

$$\left\| U'(V')^{\mathsf{T}} - \sum_{i>m}^{n} (a_i^{\otimes 3})(a_i^{\otimes 3})^{\mathsf{T}} \right\| \leqslant O(\varepsilon)\,.$$

*Moreover, the algorithm runs in time $\tilde{O}\left(d^{2\cdot\omega(1+\log n/\log d^2)}\right)$, where $\omega(k)$ is the time required to multiply a $(d^k \times d)$ matrix with a $(d \times d)$ matrix.*[11]

**Proof** $\sum_{i\in[m]}(b_i^{\otimes 3})(b_i^{\otimes 3})^{\mathsf{T}}$ *can be written as tensor networks as in Fig. 3. On the other hand multiplying $UV^{\mathsf{T}}$ by a $d^3$-dimensional vector takes time at most $\tilde{O}(n^{\omega(2)}) \leqslant \tilde{O}(d^{4.9})$. Thus, as in Lemma 34, we can compute the top $n-m$ eigenspace of their difference in time $\tilde{O}\left(d^{2\cdot\omega(1+\log n/2\log d)}\right)$. By Lemma 30 the result follows.* ■

### D.6. Putting things together

We are now ready to prove Theorem 7.

**Proof** [Proof of Theorem 7] By lemma Lemma 34, the lifting step of Algorithm 3 can done in $\tilde{O}\left(d^{2\cdot\omega(1+\log n/2\log d)}\right)$ time. Combining Lemma 35, Lemma 39 Lemma 40, and Lemma 41, each iteration of the step 2 in Algorithm 3 can be done in time $O\left(n \cdot d^4 + d^{2\cdot\omega(1+\log n/2\log d)}\right)$. There are at most $O(\log n)$ iterations, and thus the total running time of the loop is bounded by $\tilde{O}\left(d^{2\cdot\omega(1+\log n/2\log d)} + nd^4\right)$. ■

## Appendix E. Partial recovery from reducing to robust fourth-order decomposition

We observed that the tensor network in Fig. 2(b) allows us to partially reduce the problem of third-order tensor decomposition to the problem of robust fourth-order tensor decomposition. A natural idea would thus be to apply existing algorithms, e.g., Hopkins et al. (2019), to this latter problem. However, such a black-box reduction faces several issues: First, the spectral norm of the noise of the network in Fig. 2(b) can only be bounded by $1/\operatorname{polylog}(d)$. For this amount of noise, the algorithm in Hopkins et al. (2019) can only recover a constant fraction, bounded away from 1, of the components, but not all of them. It is unclear, if their analysis can be adapted to handle larger amount of noise, since they deal with the inherently harder setting of adversarial instead of random noise. Second, the running time of this black-box reduction would be $\tilde{O}(n \cdot d^5)$,[12] which is $\tilde{O}(d^{6.5})$ for $n = \Theta(d^{3/2}/\operatorname{polylog}(d))$. This is even slower than our nearly-quadratic running time of $\tilde{O}\left(d^{6.043182}\right)$. Lastly, their analysis is quite involved and we argue that the language of tensor networks captures the essence of the third-order problem and thus yields a considerably simpler algorithm than this black-box reduction.

11. See Fig. 7 in Appendix I.
12. We remark that the main result in Hopkins et al. (2019) contains a minor imprecision concerning the running time. In particular, their algorithm runs in time $\tilde{O}(n \cdot d^5)$ while their result states $\tilde{O}(n^2 d^3)$ time. In the context of our interest this is a meaningful difference as $n/d^2 = o(1/\sqrt{d})$.

## Appendix F. Boosting to arbitrary accuracy

Given good initialization vector for every component, it is shown in Anandkumar et al. (2015) that we can get arbitrarily accurate estimation of the components by combining the tensor power iteration algorithm and residual error removal:

**Theorem 42 (Theorem 1 in Anandkumar et al. (2015))** *Suppose we are given tensor* $T = \sum_{i=1}^n a_i^{\otimes 3}$*, where* $n = O\left(d^{3/2}/\operatorname{polylog}(d)\right)$ *and* $a_1, a_2, \ldots, a_n$ *are independent and uniformly sampled from the unit sphere and* $\lambda_i = 1 \pm o(1)$*. Then given vectors* $b_1, b_2, \ldots, b_n$ *s.t* $\langle a_i, b_i \rangle \geqslant 0.99$*, there is a polynomial time algorithm outputting unit norm vectors* $c_1, c_2, \ldots, c_n$ *s.t*

$$\langle c_i, a_i \rangle \geqslant 1 - \varepsilon$$

Combining with Theorem 6 in this section, we thus get the following corollary

**Corollary 43** *Suppose we are given tensor* $T = \sum_{i=1}^n a_i^{\otimes 3}$*, where* $n = O\left(d^{3/2}/\operatorname{polylog}(d)\right)$ *and* $a_1, a_2, \ldots, a_n$ *are independently and uniformly sampled from the dimension* $d$ *unit sphere, then there is a* $\operatorname{poly}(d)$*-time algorithm outputting unit norm vectors* $b_1, b_2, \ldots, b_n \in \mathbb{R}^d$ *such that probability* $1 - o(1)$ *over* $a_1, a_2, \ldots, a_n$*, for each* $i \in [n]$, $\max_{j \in [n]} \langle a_i, b_j \rangle \geqslant (1 - 2^{-n}) \|a_i\|$.

## Appendix G. Concentration bounds

### G.1. Concentration of Gaussian polynomials

**Fact 44** *[Lemma A.4 in Hopkins et al. (2016)] Let* $X \sim \mathcal{N}(0,1)$*. Then for* $t > 0$*,*

$$\mathbb{P}(X > t) \leqslant \frac{e^{-t^2/2}}{t\sqrt{2\pi}}$$

*and*

$$\mathbb{P}(X > t) \geqslant \frac{e^{-t^2/2}}{\sqrt{2\pi}} \cdot \left(\frac{1}{t} - \frac{1}{t^3}\right)$$

**Proof** We record their proof for completeness. For the first statement, we have

$$\begin{aligned}
\mathbb{P}(X > t) &= \frac{1}{\sqrt{2\pi}} \int_t^\infty e^{-x^2/2} dx \\
&\leqslant \frac{1}{\sqrt{2\pi}} \int_t^\infty \frac{x}{t} e^{-x^2/2} dx \\
&= \frac{e^{-t^2/2}}{t\sqrt{2\pi}}
\end{aligned}$$

For the second statement, we have

$$\begin{aligned}
\mathbb{P}(X > t) &= \frac{1}{\sqrt{2\pi}} \int_t^\infty e^{-x^2/2} dx \\
&= \frac{1}{\sqrt{2\pi}} \int_t^\infty \frac{1}{x} \cdot x e^{-x^2/2} dx \\
&= \frac{1}{\sqrt{2\pi}} \left[ -\frac{1}{x} e^{-x^2/2}. \right]_t^\infty - \frac{1}{\sqrt{2\pi}} \int_t^\infty \frac{1}{x^2} \cdot e^{-x^2/2} dx \\
&\geqslant \frac{1}{\sqrt{2\pi}} \left[ -\frac{1}{x} e^{-x^2/2}. \right]_t^\infty - \frac{1}{\sqrt{2\pi}} \int_t^\infty \frac{x}{t^3} \cdot e^{-x^2/2} dx \\
&= \frac{1}{\sqrt{2\pi}} \left( \frac{1}{t} - \frac{1}{t^3} \right) e^{-t^2/2}
\end{aligned}$$

■

**Lemma 45 (Lemma A.5 in Hopkins et al. (2016))** *For each $\ell \geqslant 1$ there is a universal constant $c_\ell > 0$ such that for every $f$ a degree- $\ell$ polynomial of standard Gaussian random variables $X_1, \ldots, X_m$ and $t \geqslant 2$*

$$\mathbb{P}(|f(X)| > t\mathbb{E}|f(X)|) \leqslant e^{-c_\ell t^{2/\ell}}$$

*The same holds (with a different constant $c_\ell$ ) if $\mathbb{E}|f(x)|$ is replaced by $\left(\mathbb{E} f(x)^2\right)^{1/2}$.*

**Lemma 46 (Fact C.1 in Hopkins et al. (2016))** *Suppose $a_1, a_2, \ldots, a_n$ are independently sampled from $N(0, \frac{1}{d}\mathrm{Id}_d)$, then with probability $1 - n^{-\omega(1)}$, we have*

*(a) for each $i \in n$, $\|a_i\|^2 = 1 \pm \tilde{O}\left(\frac{1}{\sqrt{d}}\right)$*

*(b) for each $i, j \in n$, $i \neq j$, we have $\langle a_i, a_j \rangle^2 = \tilde{O}\left(\frac{1}{d}\right)$*

## G.2. Concentration of random matrices

**Lemma 47** *For $n \leqslant d^{3/2} / \operatorname{polylog} d$, let $a_1, \cdots, a_n$ be $n$ i.i.d. random unit vectors*

*(a) For any $i \neq j$,*

$$|\langle a_i, a_j \rangle| \overset{w.ov.p}{=} \tilde{O}\left(\frac{1}{\sqrt{d}}\right).$$

*(b)*

$$\left\| \sum_{i=1}^n a_i a_i^\top \right\| \overset{w.ov.p}{=} \tilde{O}\left(\frac{n}{d}\right).$$

*(c)*

$$\left\| \sum_{i=1}^n a_i^{\otimes 2} \left(a_i^{\otimes 2}\right)^\top \right\| \overset{w.ov.p}{=} \tilde{O}\left(\frac{n}{d}\right).$$

*(d)*

$$\left\|\sum_{i=1}^n a_i^{\otimes 3}\left(a_i^{\otimes 3}\right)^\top\right\| \overset{w.ov.p}{=} 1 \pm \tilde{O}\left(\frac{n}{d^{3/2}}\right).$$

**Proof**

(a) We rewrite $a_i = \frac{b_i}{\|b_i\|}$, where $b_1, b_2, \ldots, b_n \sim N(0, \frac{1}{d}\mathrm{Id}_d)$ are independent. Then $\langle a_i, a_j\rangle = \frac{\langle b_i, b_j\rangle}{\|b_i\|\|b_j\|}$. Now using lemma 46, we have the claim.

(b) We rewrite $a_i = \frac{b_i}{\|b_i\|}$, where $b_1, b_2, \ldots, b_n \sim N(0, \frac{1}{d}\mathrm{Id}_d)$ are independent. Then by fact C.2 in Hopkins et al. (2016), with overwhelming probability, we have $\left\|\sum_{i=1}^n b_i b_i^\top\right\| \leqslant \tilde{O}\left(\frac{n}{d}\right)$. Now by lemma 46, we have

$$\left\|\sum_{i=1}^n a_i a_i^\top\right\| \leqslant \tilde{O}\left(\frac{n}{d}\right)$$

(c) Let $U \in \mathbb{R}^{d^2 \times n}$ be a matrix with $i$-th row given by $a_i^{\otimes 2}$, then we have

$$\left\|\sum_{i=1}^n a_i^{\otimes 2}\left(a_i^{\otimes 2}\right)^\top\right\| = \left\|UU^\top\right\| = \left\|U^\top U\right\|$$

Now we have $(U^\top U)_{ii} = \langle a_i, a_i\rangle^2 = 1$, and by (a) $(U^\top U)_{ij} = \langle a_i, a_j\rangle^2 = \tilde{O}\left(\frac{1}{d}\right)$. Thus by Gershgorin circle theorem, we have

$$\|U^\top U\| \leqslant \max_{i\in[d^2]} \sum_{j\in[d^2]} \left|(U^\top U)_{ij}\right| = \tilde{O}\left(\frac{n}{d}\right)$$

(d) Let $U \in \mathbb{R}^{d^3 \times n}$ be a matrix with $i$-th row given by $a_i^{\otimes 3}$, then we have

$$\left\|\sum_{i=1}^n a_i^{\otimes 3}\left(a_i^{\otimes 3}\right)^\top\right\| = \left\|UU^\top\right\| = \left\|U^\top U\right\|$$

Now we have $(U^\top U)_{ii} = \langle a_i, a_i\rangle^3 = 1$, and by (a) with overwhelming probability $(U^\top U)_{ij} = \langle a_i, a_j\rangle^3 = \tilde{O}\left(\frac{1}{d^{3/2}}\right)$. Thus by Gershgorin circle theorem, we have

$$\|U^\top U\| \leqslant \max_{i\in[d^3]} \sum_{j\in[d^3]} \left|(U^\top U)_{ij}\right| = \tilde{O}\left(\frac{n}{d^{3/2}}\right)$$

■

**Corollary 48** *For $n \leqslant d^{3/2}/\operatorname{polylog} d$, let $a_1, \cdots, a_n$ be $n$ i.i.d. random unit vectors, and let $s_1, \ldots, s_n$ be independent random signs.*

*(a)*

$$\left\|\sum_{i=1}^{n} s_i \cdot a_i a_i^\top\right\| \overset{w.ov.p}{=} \tilde{O}\left(\sqrt{\frac{n}{d}}+1\right).$$

*(b)*

$$\left\|\sum_{i=1}^{n} s_i \cdot a_i^{\otimes 2}\left(a_i^{\otimes 2}\right)^\top\right\| \overset{w.ov.p}{=} \tilde{O}\left(\sqrt{\frac{n}{d}}\right).$$

*(c)*

$$\left\|\sum_{i=1}^{n} s_i \cdot a_i^{\otimes 3}\left(a_i^{\otimes 3}\right)^\top\right\| \overset{w.ov.p}{=} \tilde{O}(1).$$

**Lemma 49 (Lemma 5.9 in Hopkins et al. (2016))** *For $R = \sqrt{2}\left(\mathbb{E}_{a\sim N(0,\mathrm{Id}_d)}(aa^\top)^{\otimes 2}\right)^{+1/2}$, denote $\Phi = \sum_i e_i^{\otimes 2} \in \mathbb{R}^{d^2}$, (a) we have $\|R\| = 1$ and moreover*

$$R = \sqrt{2}\left(\Sigma^{+}\right)^{1/2} = \Pi_{\text{sym}} - \frac{1}{d}\left(1-\sqrt{\frac{2}{d+2}}\right)\Phi\Phi^\top$$

*(b) for any $v \in \mathbb{R}^d$,*

$$\|R(v\otimes v) - v\otimes v\|_2^2 = \left(\frac{1}{d+2}\right)\cdot\|v\|^4$$

**Proof** (a) has been proved in Lemma 5.9 of Hopkins et al. (2016). For (b), without loss of generality, we assume $\|v\| = 1$. Then we have

$$R(v\otimes v) - v\otimes v = -\frac{1}{d}\left(1-\sqrt{\frac{2}{d+2}}\right)\langle\Phi^\top, v\otimes v\rangle\Phi$$

Since $\|\Phi\| = \sqrt{d}$ and $\langle\Phi^\top, v\otimes v\rangle = \sum_{i=1}^{d}\langle v, e_i\rangle^2 = 1$, we have

$$\|R(v\otimes v) - v\otimes v\|^2 = \frac{1}{d+2}$$

which concludes the proof. ■

**Lemma 50** *[Similar to Lemma 5.11 in Hopkins et al. (2016)] Let $a_1, \ldots, a_n \in \mathbb{R}^d$ independently and uniformly sampled from the unit sphere. Let $R = \sqrt{2}\cdot\left(\mathbb{E}\left((aa^\top)^{\otimes 2}\right)\right)^{+1/2}$. Let $u_i = a_i \otimes a_i$. With overwhelming probability, every $j \in [n]$ satisfies (a) $\sum_{i\neq j}\left\langle u_j, R^2 u_i\right\rangle^2 = \tilde{O}\left(n/d^2\right)$ (b) $\|Ru_j - u_j\|^2 \leqslant \tilde{O}\left(\frac{1}{d}\right)$.*

**Proof** (a) We follow the same proof as in the lemma 5.11 of Hopkins et al. (2016) (which is for $a_1, \dots, a_n \sim N(0, \mathrm{Id}_d)$):

$$\begin{aligned}\sum_{i \neq j} \left\langle u_j, R^2 u_i \right\rangle^2 &= \sum_{i \neq j} \left\langle u_j, \left(\Pi_{\text{sym}} - \frac{1}{d+2} \Phi \Phi^\top\right) u_i \right\rangle^2 \\ &= \sum_{i \neq j} \left( \langle a_j, a_i \rangle^2 - \frac{1}{d+2} \|u_j\|^2 \|u_i\|^2 \right)^2 \\ &= \sum_{i \neq j} \tilde{O}(1/d)^2 \\ &= \tilde{O}\left(n/d^2\right) .\end{aligned}$$

(b) This follows directly from Lemma 5.9(b) by replacing $v$ with $a_i$. ∎

**Lemma 51 (Lemma 5.9 in Hopkins et al. (2016))** *For $n \leqslant \tilde{O}\left(d^{3/2}\right)$, let $R = \sqrt{2}\left(\mathbb{E}\, a^{\otimes 2}\left(a^{\otimes 2}\right)^\top\right)^{-1/2}$ where $a \sim N(0, \mathrm{Id})$, and $a_1, a_2, \dots, a_n \in \mathbb{R}^d$ be i.i.d random vectors sampled uniformly from the unit sphere. Then with probability at least $1 - \tilde{O}\left(\frac{n}{d^{3/2}}\right)$, we have*

$$\left\| \sum_{i=1}^n R a_i^{\otimes 2} \left(R a_i^{\otimes 2}\right)^\top - \Pi \right\| \leqslant \tilde{O}\left(\frac{n}{d^2}\right)$$

*where $\Pi$ is the projection matrix to the span of $\left\{R a_i^{\otimes 2}\right\}$.*

**Lemma 52** *For vectors $a_1, a_2, \dots, a_n \in \mathbb{R}^d$ sampled uniformly at random from unit sphere, and $R = \sqrt{2}\left(\mathbb{E}\, a^{\otimes 2}\left(a^{\otimes 2}\right)^\top\right)^{+1/2}$, we have*

$$\left\| \sum_{i=1}^n R a_i^{\otimes 2} \left((R a_i^{\otimes 2})^{\otimes 2}\right)^\top \right\| \leqslant 1 + \tilde{O}\left(\frac{n}{d^{3/2}}\right)$$

**Proof** Let $U \in \mathbb{R}^{n \times d^2}$ be a matrix with the $i$-th row vector given By $R a_i^{\otimes 2}$, and let $V \in \mathbb{R}^{n \times d^4}$ be a matrix with the $i$-th row vector given by $\left(R a_i^{\otimes 2}\right)^{\otimes 2}$. Then we have $\sum_{i=1}^n R a_i^{\otimes 2} \left((R a_i^{\otimes 2})^{\otimes 2}\right)^\top = U V^t op$. Our strategy is then to bound $\|U\|$ and $\|V\|$.

First with high probability we have

$$\|U\| = \sqrt{\|U U^\top\|} = \sqrt{\left\| \sum_{i=1}^n R a_i^{\otimes 2} \left(R a_i^{\otimes 2}\right)^\top \right\|} \leqslant 1 + \tilde{O}\left(\frac{n}{d^{3/2}}\right)$$

Second with high probability have

$$\|V\| = \sqrt{\|V V^\top\|} = \sqrt{\left\| \sum_{i=1}^n \left(R a_i^{\otimes 2} \left(R a_i^{\otimes 2}\right)^\top\right)^{\otimes 2} \right\|} \leqslant 1 + \tilde{O}\left(\frac{n}{d^{3/2}}\right)$$

It then follows that $\|U V^\top\| \leqslant \|U\| \|V\| \leqslant 1 + \tilde{O}\left(\frac{n}{d^{3/2}}\right)$ . ∎

**Lemma 53 (Concentration of random tensor contractions Ma et al. (2016))** *Let $g$ be a standard Gaussian vector in $\mathbb{R}^k, g \sim \mathcal{N}(0, \mathrm{Id}_k)$. Let $A$ be a tensor in $(\mathbb{R}^k) \otimes (\mathbb{R}^\ell) \otimes (\mathbb{R}^m)$, and call the three modes of $A \alpha, \beta, \gamma$ respectively. Let $A_i$ be a $\ell \times m$ slice of $A$ along mode $\alpha$. Then,*

$$\mathbb{P}\left[\left\|\sum_{i=1}^{k} g_i A_i\right\| \geqslant t \cdot \max\left\{\left\|A_{\{\alpha\beta\}\{\gamma\}}\right\|, \left\|A_{\{\alpha\gamma\}\{\beta\}}\right\|\right\}\right] \leqslant (m+\ell)\exp\left(-\frac{t^2}{2}\right)$$

### G.3. Rademacher bounds on general matrices

**Theorem 54** *(Follows directly from (Tropp, 2015, Theorem 4.6.1)) Let $A_1, \ldots, A_n$ be a sequence of symmetric matrices with dimension $d^{\Theta(1)}$ and let $s_1, \ldots, s_n$ be a sequence of i.i.d. Rademacher random variables. Let $Y = \sum_{i=1}^n s_i \cdot A_i$ and $v(Y) = \left\|\sum_{i=1}^n A_i^2\right\|_2$. Then with overwhelming probability*

$$\|Y\|_2 \leqslant \tilde{O}\left(\sqrt{v(Y)}\right).$$

**Lemma 55** *((Hopkins et al., 2016, Corollary 5.5)) Let $s_1, \ldots, s_n$ be independent random signs. Let $A_1, \ldots, A_n$ and $B_1, \ldots, B_n$ be Hermitian matrices. Then, w.ov.p.,*

$$\left\|\sum_{i\in[n]} s_i \cdot A_i \otimes B_i\right\| \leqslant \tilde{O}\left(\left(\max_{i\in[n]}\|B_i\|\right) \cdot \left\|\sum_{i\in[n]} A_i^2\right\|^{\frac{1}{2}}\right).$$

The next lemma doesn't contain any randomness but it's very similar to the one above and used in the same context, so we will also list it here.

**Lemma 56** *For $i = 1, \ldots, n$ let $A_i, B_i$ be symmetric matrices and suppose that for all $i$ we have that $A_i$ is psd. Then $\|\sum_{i=1}^n A_i \otimes B_i\| \leqslant \left(\max_{i\in[n]} \|B_i\|\right) \cdot \|\sum_{i=1}^n A_i\|$.*

**Proof** Let $b = \max_{i\in[n]} \|B_i\|$. For each $i$ we have that $A_i \otimes B_i \preceq A_i \otimes b \cdot \mathrm{Id}$ since $A_i$ and $b \cdot \mathrm{Id} - B_i$ are psd and the Kronecker product of two psd matrices is also psd. By summing over all $i$ we get that $\sum_{i=1}^n A_i \otimes B_i \preceq b \cdot (\sum_{i=1}^n A_i) \otimes \mathrm{Id}$ which implies the claim. ■

Finally we use a decoupling lemma from probability theory. A special version of this lemma has been used in Hopkins et al. (2016).

**Theorem 57 (Theorem 1 in de la Peña and Montgomery-Smith (1995))** *For any constant $k$, let $s, \left\{s^{(1)}\right\}, \left\{s^{(2)}\right\}, \ldots, s^{(\ell)} \in \{\pm 1\}^d$ be independent Rademacher vectors. Let $\{M_{i_1,i_2,\ldots,i_\ell} : i_1, i_2, \ldots, i_\ell \in [d]\}$ be a family of matrices. Then there is constant $C$ which depends only on $k$, so that for every $t > 0$,*

$$\mathbb{P}\left(\left\|\sum_{0\leqslant i_1\neq i_2\neq\ldots\neq i_\ell\leqslant d} s_{i_1}s_{i_2}\ldots s_{i_\ell} M_{i_1,i_2,\ldots,i_\ell}\right\|_{op} > t\right) \leqslant C\cdot\mathbb{P}\left(C\left\|\sum_{0\leqslant i_1\neq i_2\neq\ldots\neq i_\ell\leqslant d} s_{i_1}^{(1)}s_{i_2}^{(2)}\ldots s_{i_\ell}^{(\ell)} M_{i_1,i_2,\ldots,i_\ell}\right\|_{op} > t\right)$$

### G.4. Optimizer of tensor injective norm

**Lemma 58 (Lemma 5.20 in Hopkins et al. (2016))** *Let $T = \sum_{i\in[n]} a_i \otimes a_i \otimes a_i$ for normally distributed vectors $a_i \sim \mathcal{N}\left(0, \frac{1}{d}\mathrm{Id}_d\right)$. For all $0 < \gamma, \gamma' < 1$,*

- *With overwhelming probability, for every $v \in \mathbb{R}^d$ such that $\sum_{i\in[n]} \langle a_i, v\rangle^3 \geqslant 1-\gamma$*

$$\max_{i\in[n]} |\langle a_i, v\rangle| \geqslant 1 - O(\gamma) - \tilde{O}\left(n/d^{3/2}\right)$$

- *With overwhelming probability over $a_1, \ldots, a_n$, if $v \in \mathbb{R}^d$ with $\|v\| = 1$ satisfies $\langle v, a_j\rangle \geqslant 1-\gamma'$ for some $j$ then $\sum_i \langle a_i, v\rangle^3 \geqslant 1 - O(\gamma') - \tilde{O}\left(n/d^{3/2}\right)$*

## Appendix H. Linear algebra

In this section, we record some linear algebra facts and results used in the paper.

**Lemma 59** *For $n' = \tilde{O}(d^{3/2})$ and $d \leqslant n \leqslant n'$, suppose vectors $b_2, \ldots, b_n$ satisfy $\|M - \Pi\| \leqslant \tilde{O}\left(\frac{n'}{d^{3/2}}\right)$, where $M = \sum_{i=1}^n b_i b_i^\top$ and $\Pi$ is the projection matrix to the span of $\{b_i : i \in [2, n]\}$. Then we have $\left\|M^\top M - M\right\| \leqslant \tilde{O}\left(\frac{n'}{d^{3/2}}\right)$.*

**Proof** Since $\Pi^2 = \Pi$, we have $M^\top M - \Pi = M^\top M - M^\top \Pi + M^\top \Pi - \Pi = M^\top(M - \Pi) + (M - \Pi)^\top \Pi$. Since $\|\Pi\| = 1$ and $\|M\| \leqslant \|\Pi\| + \|\Pi - M\| \leqslant 1 + \tilde{O}\left(\frac{n}{d^{3/2}}\right)$, it follows that $\|M^\top(M - \Pi) + (M - \Pi)^\top \Pi\| \leqslant \|M - \Pi\| \left(\|M^\top\| + \|\Pi\|\right) \leqslant \tilde{O}\left(\frac{n}{d^{3/2}}\right)$ and we have the claim. ■

### H.1. Fast SVD algorithm

For implementation, we use the lazy SVD algorithm from Allen-Zhu and Li (2016).

**Lemma 60 (Implicit gapped eigendecomposition; Lemma 7 in Hopkins et al. (2019), Corollary 4.4 in Allen-Zhu a** *Suppose a symmetric matrix $M \in \mathbb{R}^{d\times d}$ has an eigendecomposition $M = \sum_j \lambda_j v_j v_j^\top$, and that $Mx$ may be computed within $t$ time steps for $x \in \mathbb{R}^d$. Then $v_1, \ldots, v_n$ and $\lambda_1, \ldots, \lambda_n$ may be computed in time $\tilde{O}\left(\min\left(n(t + nd)\delta^{-1/2}, d^3\right)\right)$, where $\delta = (\lambda_n - \lambda_{n+1})/\lambda_n$. The dependence on the desired precision is polylogarithmic.*

## Appendix I. Fast matrix multiplications and tensor contractions

To easily compute the running time of Theorem 1 under a specific set of parameters $n, d$, we include here a table (Fig. 7) from Gall and Urrutia (2018) with upper bounds on rectangular matrix multiplication constants. We remind the reader that basic result in algebraic complexity theory states that the algebraic complexities of the following three problems are the same:

- computing a $(d^k \times d) \times (d \times d)$ matrix multiplication,
- computing a $(d \times d^k) \times (d^k \times d)$ matrix multiplication,
- computing a $(d \times d) \times (d \times d^k)$ matrix multiplication.

| $k$ | upper bound on $\omega(k)$ |
|---|---|
| 0.31389 | 2 |
| 0.32 | 2.000064 |
| 0.33 | 2.000448 |
| 0.34 | 2.001118 |
| 0.35 | 2.001957 |
| 0.40 | 2.010314 |
| 0.45 | 2.024801 |
| 0.50 | 2.044183 |

| $k$ | upper bound on $\omega(k)$ |
|---|---|
| 0.5286 | 2.057085 |
| 0.55 | 2.067488 |
| 0.60 | 2.093981 |
| 0.65 | 2.123097 |
| 0.70 | 2.154399 |
| 0.75 | 2.187543 |
| 0.80 | 2.222256 |
| 0.85 | 2.258317 |
| 0.90 | 2.295544 |
| 0.95 | 2.333789 |
| 1.00 | 2.372927 |

| $k$ | upper bound on $\omega(k)$ |
|---|---|
| 1.10 | 2.453481 |
| 1.20 | 2.536550 |
| 1.30 | 2.621644 |
| 1.40 | 2.708400 |
| 1.50 | 2.796537 |
| 1.75 | 3.021591 |
| 2.00 | 3.251640 |
| 2.50 | 3.721503 |
| 3.00 | 4.199712 |
| 4.00 | 5.171210 |
| 5.00 | 6.157233 |

Figure 7: $\omega(k)$ denotes the exponent of the multiplication of an$(d \times d^k)$ by a $(d^k \times d)$ matrix, so that the running time is $O(d^{\omega(k)})$.

### I.1. Fast algorithms for low rank tensors

We state the running time for some common tensor operations given implicit representation. The proofs are very similar to the lemma 8 in Hopkins et al. (2019).

The first lemma is about computing tensor contraction.

**Lemma 61 (Time for computing tensor contraction)** *Let $\ell \in \tilde{O}(d^2)$. Suppose we are given $U, V \in \mathbb{R}^{d^3 \times n}$. Consider vectors $x_1, x_2, \dots x_\ell \in \mathbb{R}^{d^2}$ ,$g_1, g_2, \dots, g_\ell \in \mathbb{R}^{d^2}$, and tensor $\mathbf{T} \in \left(\mathbb{R}^d\right)^{\otimes 6}$ satisfying $T_{\{1,2,3\}\{4,5,6\}} = UV^\top$. Then there is an algorithm computing $\left(x_i^\top \otimes \mathrm{Id}_{d^2} \otimes g_i^\top\right) \mathbf{T}$ for all $i \leqslant \ell$, in $O\left(n \cdot d^4 + d^{2\left(\omega\left(\frac{1}{2}(1+\log_d n)\right)\right)}\right)$ time. When $n = O\left(d^{3/2}/\operatorname{polylog}(d)\right)$, this is bounded by $\tilde{O}(n \cdot d^4 + d^{5.25})$ time.*

**Proof** Since $\left(x_i^\top \otimes \mathrm{Id}_{d^2} \otimes g_i^\top\right) \mathbf{T} = \left(x_i \otimes \mathrm{Id}_d\right)^\top UV^\top (g_i \otimes \mathrm{Id}_d)$, we only need to obtain $Y_i = V^\top (g_i \otimes \mathrm{Id}_d)$ and $Z_i = U^\top\left(x_i \otimes \mathrm{Id}_d\right)$ for all $i \in [\ell]$, and then compute $Z_i^\top Y_i$ for all $i \in [\ell]$. Since $Y_i \in \mathbb{R}^{n \times d}$ and $Z_i \in \mathbb{R}^{n \times d}$, the last step takes time $n \cdot d^2 \cdot \ell = \tilde{O}(n \cdot d^4)$.

To obtain $Y_i$ for all $i \in [\ell]$, we construct a $d^2 \times \ell$ matrix $G$, whose $i$-th column is given by $g_i$. Then $Y_i$ can all be obtained as sub-matrix of $M_1 = V^\top (G \otimes \mathrm{Id}_d)$. We write $V$ as a block matrix: $V^\top = (V_1^\top, V_2^\top, \dots, V_d^\top)$ where $V_1, V_2, \dots, V_d \in \mathbb{R}^{n \times d^2}$. Then $M_1$ is equivalent to a reshaping of $(V')^\top G$ where $V' = (V_1, V_2, \dots, V_d)$. Since $V' \in \mathbb{R}^{d^2 \times nd}$, $G \in \mathbb{R}^{d^2 \times \ell}$, and $\ell = \tilde{O}(d^2)$, this matrix multiplication takes time at most $O\left(d^{2\left(\omega\left(\frac{1}{2}(1+\log_d n)\right)\right)}\right)$. By the same reasoning, it takes time at most $O\left(d^{2\left(\omega\left(\frac{1}{2}(1+\log_d n)\right)\right)}\right)$ to obtain $Z_i$ for all $i \in [\ell]$.

In conclusion, the running time of is bounded by $O\left(n \cdot d^4 + d^{2\left(\omega\left(\frac{1}{2}(1+\log_d n)\right)\right)}\right)$. Since $\omega(5/4) \leqslant 2.622$, this is bounded by $O(n \cdot d^4 + d^{5.25})$. ■

The second lemma is about computing singular value decomposition for rectangular flattening of a low rank order-6 tensor. The proof has already appeared in the proof of lemma 8 in Hopkins et al. (2019).

**Lemma 62 (Time for computing singular value decomposition)** *Suppose we are given matrices $U \in \mathbb{R}^{d^3 \times n}$ and $Z \in \mathbb{R}^{nd \times d^2}$. Then for matrix $M := Z^\top (UU^\top \otimes \mathrm{Id}_d) Z$ and $k = O(n)$, there is a $\tilde{O}(n^2 d^3 \delta^{-1})$ time algorithm obtaining $P \in \mathbb{R}^{d^3 \times k}$ and diagonal matrix $\Lambda \in \mathbb{R}^{k \times k}$ such that*

$$\left\| M^{1/2} - P\Lambda^{1/2} P^\top \right\| \leqslant (1+\delta)\rho_k$$

*where $\rho_k$ is the $k$-th largest eigenvalue of $M^{1/2}$.*

**Proof** We first claim that matrix-vector multiplication by $M$ can be implemented in $O\left(nd^3\right)$ time, with $O\left(n^2 d^3\right)$ preprocessing time for computing the product $U^\top U$. The matrix-vector multiplications by $Z$ and $Z^\top$ take time $O\left(nd^3\right)$, and then multiplying $Zy$ by $U^\top U \otimes \mathrm{Id}_d$ is reshaping-equivalent to multiplying $U^\top U$ into the $n \times d$ matrix reshaping of $Zy$, which takes $O\left(n^2 d\right)$ time with the precomputed $n \times n$ matrix $U^\top U$. Therefore, by Lemma 60, it takes time $\tilde{O}\left(n^2 d^3 \delta^{-1/2}\right)$ to yield a rank-$k$ eigendecomposition $P\Lambda P^\top$ such that $\left\| M^{1/2} - P\Lambda^{1/2} P^\top \right\| \leqslant (1+\delta)\rho_k$ ■

## Appendix J. Missing proofs

In this section we will give the proofs we omitted in the main body of the paper.

### J.1. Reducing to isotropic components

In this section, we prove that the components $a_i^{\otimes 2}$ are nearly isotropic in the sense of Frobenius norm. Concretely we prove the following theorem.

**Lemma 63 (Restatement of Lemma 17)** *For $n = O\left(d^{3/2}/\operatorname{polylog}(d)\right)$ and $n' \leqslant n$, let $a_1, a_2, \ldots, a_{n'} \in \mathbb{R}^d$ be $(n,d)$-nicely-separated. Let $R = \sqrt{2} \cdot \left(\mathbb{E}_{a \sim N(0, \mathrm{Id}_d)} a^{\otimes 2} \left(a^{\otimes 2}\right)^\top\right)^{+1/2}$, for any tensor $\hat{\mathbf{M}} = \sum_{i=1}^n a_i^{\otimes 6} + \mathbf{E}$ with $\|\mathbf{E}\|_F \leqslant \tilde{O}\left(\frac{n'}{d^{3/2}}\right) \cdot \sqrt{n}$, we have*

$$\left\| \hat{\mathbf{M}} - \sum_{i=1}^{n'} \left(R a_i^{\otimes 2}\right)^{\otimes 3} \right\|_F \leqslant \tilde{O}\left(\frac{n'}{d^{3/2}}\right) \cdot \sqrt{n}\,.$$

This will allow us to rewrite $\hat{\mathbf{M}} = \sum_{i=1}^{n'} \left(R a_i^{\otimes 2}\right)^{\otimes 3} + \mathbf{E}'$ where $\|\mathbf{E}'\| \leqslant \frac{1}{\operatorname{polylog}(d)}$. The advantage is that the component vectors $R a_i^{\otimes 2}$ now become isotropic, and the spectral norm of $\sum_{i=1}^{n'} R a_i^{\otimes 2} \left(R a_i^{\otimes 2}\right)^\top$ is tightly bounded.

The lemma follows as a corollary of the statement below:

**Lemma 64** *For $n = O\left(d^{3/2}/\operatorname{polylog}(d)\right)$ and $d \leqslant n' \leqslant n$, let $a_1, a_2, \ldots, a_{n'} \in \mathbb{R}^d$ be $(n,d)$-nicely-separated. Let $R = \sqrt{2} \cdot \left(\mathbb{E}_{a \sim N(0, \mathrm{Id}_d)} a^{\otimes 2} \left(a^{\otimes 2}\right)^\top\right)^{+1/2}$. Let vectors $b_i := R a_i^{\otimes 2}$ for $i \in [n']$. Then we have*

$$\left\| \sum_{i=1}^{n'} b_i^{\otimes 3} - a_i^{\otimes 6} \right\|_F \leqslant 10\delta\sqrt{n'}$$

*where $\delta = \tilde{O}\left(\frac{n}{d^{3/2}}\right)$.*

**Proof** We decompose the square of Frobenius norm into the sum of two parts:

$$\begin{aligned}\left\|\sum_{i=1}^{n'} b_i^{\otimes 3} - \sum_{i=1}^{n'} a_i^{\otimes 6}\right\|_F^2 &= \sum_{i,j\in[n']} \left\langle b_i^{\otimes 3} - a_i^{\otimes 6}, b_j^{\otimes 3} - a_j^{\otimes 6}\right\rangle \\ &= \sum_{i\in[n']} \left\|b_i^{\otimes 3} - a_i^{\otimes 6}\right\|^2 + \sum_{\substack{i,j\in[n'] \\ i\neq j}} \left\langle b_i^{\otimes 3} - a_i^{\otimes 6}, b_j^{\otimes 3} - a_j^{\otimes 6}\right\rangle\end{aligned}$$

For the first part, by $(n,d)$ nicely-separated assumption(Theorem 15), we have $\|b_i - a_i^{\otimes 2}\|^2 \leqslant \tilde{O}(1/d)$ and thus $\langle b_i, a_i^{\otimes 2}\rangle \leqslant 1 - \tilde{O}(1/d)$. It follows that $\langle b_i, a_i^{\otimes 2}\rangle^3 \geqslant 1 - \tilde{O}(1/d)$ and $\left\|b_i^{\otimes 3} - a_i^{\otimes 6}\right\|^2 \leqslant \tilde{O}(1/d)$. By summation, we have

$$\sum_{i=1}^{n'} \left\|b_i^{\otimes 3} - a_i^{\otimes 6}\right\|^2 \leqslant \tilde{O}(1/d)\cdot n \leqslant \delta^2 n$$

For the second part, we have

$$\sum_{\substack{i,j\in[n'] \\ i\neq j}} \left\langle b_i^{\otimes 3} - a_i^{\otimes 6}, b_j^{\otimes 3} - a_j^{\otimes 6}\right\rangle = \sum_{\substack{i,j\in[n'] \\ i\neq j}} \langle b_i, b_j\rangle^3 - 2\sum_{\substack{i,j\in[n'] \\ i\neq j}} \langle a_i^{\otimes 2}, b_j\rangle^3 + \sum_{\substack{i,j\in[n'] \\ i\neq j}} \langle a_i, a_j\rangle^6$$

For the first term, by assumption, for each $j \in [n]$

$$\left|\sum_{i\in[n']\setminus\{j\}} \langle b_i, b_j\rangle^3\right| \leqslant \sum_{i\in[n']\setminus\{j\}} |\langle b_i, b_j\rangle| \langle b_i, b_j\rangle^2 \leqslant \tilde{O}\left(\frac{n}{d^2}\right)$$

thus we have

$$\left|\sum_{\substack{i,j\in[n'] \\ i\neq j}} \langle b_i, b_j\rangle^3\right| \leqslant (1+o(1)) \sum_{\substack{i,j\in[n'] \\ i\neq j}} \langle b_i, b_j\rangle^2 \leqslant n'\cdot \tilde{O}\left(\frac{n}{d^2}\right) \leqslant \delta^2\cdot n'$$

For the second term, denote $c_i = a_i^{\otimes 2} - b_i$, using the $(n,d)$-nicely-separated assumption that $\|c_i\|^2 \leqslant \tilde{O}(1/d)$ and $\left\|\sum_{j\in[n']} b_j b_j^\top\right\| \leqslant 1 + o(1)$, we have

$$\begin{aligned}\left|\sum_{\substack{i,j\in[n'] \\ i\neq j}} \langle a_i^{\otimes 2}, b_j\rangle^3\right| &\leqslant (1+o(1)) \sum_{\substack{i,j\in[n'] \\ i\neq j}} \langle a_i^{\otimes 2}, b_j\rangle^2 \\ &= (1+o(1)) \sum_{\substack{i,j\in[n'] \\ i\neq j}} \langle c_i + b_i, b_j\rangle^2\end{aligned}$$

$$
\begin{aligned}
&\leqslant 2(1+o(1)) \sum_{\substack{i,j\in[n'] \\ i\neq j}} \langle c_i, b_j\rangle^2 + 2(1+o(1)) \sum_{\substack{i,j\in[n'] \\ i\neq j}} \langle b_i, b_j\rangle^2 \\
&\leqslant 2(1+o(1)) \sum_{i\in[n']} c_i^\top \left( \sum_{j\in[n]\setminus\{i\}} b_j b_j^\top \right) c_i + 2(1+o(1)) n' \cdot \tilde{O}(n/d^2) \\
&\leqslant 2(1+o(1)) \sum_{i\in[n']} \|c_i\|^2 + \tilde{O}(nn'/d^2) \\
&\leqslant \tilde{O}\left( nn'/d^2 + n/\sqrt{d} \right) \\
&\leqslant o(\delta^2 \cdot n')
\end{aligned}
$$

For the third term, by the $(n,d)$-nicely-separated property, we have $\langle a_i, a_j\rangle^6 \leqslant \frac{1}{d^{3/2}}$. Therefore we have

$$
\sum_{\substack{i,j\in[n'] \\ i\neq j}} \langle a_i, a_j\rangle^6 \leqslant \frac{1}{d^{3/2}} \cdot n'^2 \leqslant \delta^2 n'
$$

Therefore in all, we have $\left\| \sum_{i=1}^{n'} b_i^{\otimes 3} - \sum_{i=1}^{n'} a_i^{\otimes 6} \right\|_F^2 \leqslant 2\delta^2 n'$ and thus we have the claim. ■

**Proof** [Proof of Lemma 17] By Lemma 64, we have $\left\| \sum_{i=1}^{n'} b_i^{\otimes 3} - \sum_{i=1}^{n'} a_i^{\otimes 6} \right\|_F \leqslant \tilde{O}\left(\frac{n}{d^{3/2}}\right)\sqrt{n'}$. Since $\left\| M - \sum_{i=1}^{n} a_i^{\otimes 6} \right\|_F \leqslant \tilde{O}\left(\frac{n}{d^{3/2}}\right)\sqrt{n'}$, by triangle inequality, we have the claim. ■

### J.2. Satisfaction of nice-separation property by independent random vectors

In this section, we prove Lemma 16 using the concentration results from Appendix G.

**Proof** [Proof of Lemma 16] Property (i),(ii),(iii) follows from lemma Lemma 47. Property (iv) follows from the lemma Lemma 51. Property (5),(6) follows from the lemma Theorem 50. Property (7),(8) follows from the lemma Lemma 46. ■

### J.3. Gaussian rounding

#### J.3.1. Spectral gap from random contraction

In this section, we will prove the spectral gap of diagonal terms.

**Lemma 65 (Restatement of Lemma 22)** *Consider the setting of Lemma 14. Let* $R = \sqrt{2} \cdot \left( \mathbb{E}_{a\sim \mathrm{Id}_d} \left(aa^\top\right)^{\otimes 2} \right)^{+1/2}$. *Let* $S_0 \subseteq [n]$ *be of size* $n'$ *where* $d \leqslant n' \leqslant n$ *and assume that the set* $\{a_i \mid i \in S_0\}$ *is* $(n,d)$*-nicely separated. Further, let* $\hat{\mathbf{M}}$ *be such that*

$$
\|\mathbf{M}^{\leqslant 1} - \sum_{i\in S_0} (Ra_i^{\otimes 2})^{\otimes 3}\|_{\mathrm{F}} \leqslant \varepsilon\sqrt{n'} \quad \text{and} \quad \left\| M^{\leqslant 1}_{\{1,2,3,4\}\{5,6\}} \right\|, \left\| M^{\leqslant 1}_{\{1,2,5,6\}\{3,4\}} \right\| \leqslant 1.
$$

*Consider the matrix* $M_g = (g \otimes \mathrm{Id}_{d^2} \otimes \mathrm{Id}_{d^2})\, \mathbf{M}^{\leqslant 1}_{\{1,2\}\{3,4\}\{5,6\}}$ *in Algorithm 20. Then for every* $\alpha \geqslant 1 + 10\log\log n/\log n$, *there exists a subset* $S \subseteq S_0$ *of size* $m \geqslant 0.99n'$, *such that for each* $i \in S$, *and* $v = Ra_i^{\otimes 2}$, *with probability at least* $1/d^{2\alpha}$ *over* $g$, *we have* $M = cvv^\top + N$ *where*

– $\|cvv^\top\| \geqslant (1+\frac{1}{\log d})\|N\|$

– $\|Nv\|, \|vN\| \leqslant \varepsilon c\|v\|^2$

The proof of this lemma involves a simple fact from standard Gaussian tail bound:

**Lemma 66** *Given any unit norm vector $v \in \mathbb{R}^{d^2}$, for standard random Gaussian vector $g \sim N(0, \mathrm{Id}_{d^2})$, we have*

$$\mathbb{P}\left\{|\langle g, v\rangle| \geqslant \sqrt{2\alpha \log n}\right\} = \tilde{\Theta}(n^{-\alpha})$$

**Proof** Since the distribution of $\langle g, v\rangle$ is given by $N(0,1)$. By taking $t = \sqrt{2\alpha\log n}$ in the fact 44, we have the claim. ∎

We will also use the following simple fact(a similar fact appears in Schramm and Steurer (2017)):

**Fact 67** *If $P_1, \ldots, P_n \in \mathbb{R}^{d^2 \times s}$ s.t*

$$\left\|\sum_{i=1}^n P_i P_i^\top\right\| \leqslant 1 + o(1)$$

*and $E \in \mathbb{R}^{d^2\times d^2}$ s.t $\|E\|_F \leqslant \varepsilon\sqrt{n}$ , then for a $1-\delta$ fraction of $i \in [n]$*

$$\left\|P_i^\top E\right\|_F \leqslant \varepsilon/\delta$$

**Proof** This follows from the fact that

$$\begin{aligned}
\sum_{i=1}^n \left\|P_i^\top E\right\|_F^2 &= \sum_{i=1}^n \left\langle E, P_i P_i^\top E\right\rangle \\
&= \left\langle E, \left(\sum_{i=1}^n P_i P_i^\top\right) E\right\rangle \\
&\leqslant \|E\|_F \left\|\left(\sum_{i=1}^n P_i P_i^\top\right) E\right\|_F \\
&\leqslant \|E\|_F^2 \left\|\sum_{i=1}^n P_i P_i^\top\right\| \\
&\leqslant \|E\|_F^2
\end{aligned}$$

∎

Now we prove the Lemma 22:

**Proof** [Proof of Lemma 22] For notation simplicity, for $j \in [d^2]$ we denote matrices $T_j$ as the $j$-th slice in the first mode of $\mathbf{M}^{\leqslant 1}_{\{1,2\}\{3,4\}\{5,6\}} \in \mathbb{R}^{d^2\times d^2\times d^2}$. Further we denote $\mathbf{X} = \sum_{i\in S_0}(Ra_i^{\otimes 2})^{\otimes 3}$, $\mathbf{E} = \mathbf{M}^{\leqslant 1}_{\{1,2\}\{3,4\}\{5,6\}} - \mathbf{X}$ and $E_j$ as the $j$-th slice in the first mode.

W.l.o.g. assume that $S_0 = [n']$. For each $i \in [n']$, we denote $b_i = Ra_i^{\otimes 2}$. We first prove that for each $i \in [n']$ s.t $\left\|\left(b_i b_i^\top \otimes \mathrm{Id}_{d^2}\right) E\right\|_F \leqslant 100\varepsilon$ and $\|E\left(b_i \otimes \mathrm{Id}_{d^3}\right)\|_F \leqslant 100\varepsilon$, we have $i \in S$. By Lemma 51, we have $\left\|\sum_{i=1}^{n'} b_i b_i^\top\right\| \leqslant 1 + \tilde{O}\left(\frac{n}{d^{3/2}}\right)$, and

$$\left\|\sum_{i=1}^{n'} (b_i b_i^\top)(b_i b_i^\top)^\top\right\| = \left\|\sum_{i=1}^{n'} \|b_i\|^2 b_i b_i^\top\right\| \leqslant 1 + \tilde{O}\left(\frac{n}{d^{3/2}}\right)$$

. Thus by Fact 67, the assumptions $\left\|\left(b_i b_i^\top \otimes \mathrm{Id}_{d^2}\right) E\right\|_F \leqslant 100\varepsilon$ and $\|E\left(b_i \otimes \mathrm{Id}_{d^3}\right)\|_F \leqslant 100\varepsilon$ are satisfied for at least $0.99n'$ of the component vectors. The lemma thus follows.

Without loss of generality, we suppose $\|\left(b_i \otimes \mathrm{Id}_{d^2}\right) E\|_F \leqslant 100\varepsilon$ and $\left\|\left(b_i b_i^\top \otimes \mathrm{Id}_{d^2}\right) E\right\|_F \leqslant 100\varepsilon$. We denote $g^{\|} = \frac{1}{\|b_i\|^2} \langle g, b_i\rangle\, b_i$ and $g^{\perp} = g - g^{\|}$ .Then by the property of Gaussian distribution, $g^{\|}, g^{\perp}$ are independent. Then we have

$$M = \langle g, b_1\rangle b_1 b_1^\top + \sum_{j=1}^{d^2} \left(g^{\|} + g^{\perp}\right)_j \cdot \left(\mathbf{X} - b_1^{\otimes 3} + E\right)_j = \langle g, b_1\rangle b_1 b_1^\top + N$$

where $N = \sum_{j=1}^{d^2} \left(g^{\|}\right)_j \cdot \left(\mathbf{X} - b_1^{\otimes 3} + \mathbf{E}\right)_j + \sum_{j=1}^{d^2} g_j^{\perp} \cdot T_j$.

First by Lemma 66, with probability at least $\Theta(d^{-2\alpha})$, $\langle g, b_1\rangle = \|b_1\|\|g^{\|}\| \geqslant \sqrt{4\alpha \log d}$. We denote this event as $\mathcal{G}_1(\alpha)$. On the other hand, we denote

$$\mathcal{E}_{>1}(\rho) \overset{\text{def}}{=} \left\{\left\|\sum_{j=1}^{d^2} g_j^{\perp} \cdot T_j\right\| \leqslant \sqrt{4(1+\rho)\log d}\right\}$$

By Lemma 53 and the independence between $g^{\perp}$ and $g^{\|}$ , we have

$$\mathop{\mathbb{P}}_{g}\left[\mathcal{E}_{>1}(\rho) \mid \mathcal{G}_1(\alpha)\right] \geqslant 1 - d^{-\rho}$$

Next we bound $g \cdot \left(\mathbf{X} - b_1^{\otimes 3}\right)$ and $\sum_j g_j^{\|} E_j$ separately. For the first one, by the nicely-separated assumption, we have $\max_{i\geqslant 2}|\langle b_j, b_1\rangle| \leqslant \sqrt{n}/d$ and $\|\sum_{i=2}^{n} b_j b_j^\top\| \leqslant 2$. It follows that

$$\left\|\sum_j g_j^{\|}\left(\mathbf{X} - b_1^{\otimes 3}\right)\right\| = \frac{\|g^{\|}\|}{\|b_1\|} \cdot \left\|\sum_{i=2}^{n} \langle b_j, b_1\rangle b_j b_j^\top\right\| \leqslant 2\|g^{\|}\| \cdot \max_{i\geqslant 2}|\langle b_j, b_1\rangle| \leqslant \tilde{O}\left(\frac{\sqrt{n}}{d}\right)\|g^{\|}\|$$

For the second one we have

$$\sum_j g_j^{\|} E_j = \frac{\langle g, b_1\rangle}{\|b_1\|} \cdot \sum_j b_1(j) \cdot E_j = \frac{\langle g, b_1\rangle}{\|b_1\|} \cdot \left(b_1 \otimes \mathrm{Id}_{d^2}\right) E$$

Since by assumption, we have $\|\left(b_1 \otimes \mathrm{Id}_{d^2}\right) E\|_F \leqslant 100\varepsilon$. We have

$$\left\|\sum_j g_j^{\|} E_j\right\| \leqslant 100\varepsilon\|g^{\|}\|$$

Combining both parts, the event $\mathcal{G}_1(\alpha)$ and $\mathcal{E}_{>1}(\rho)$ implies

$$\|N\| \leqslant \left\|\sum_j g_j^{\|} E_j\right\| + \left\|\sum_j g_j^{\perp} T_j\right\| + \left\|\sum_j g_j^{\|} \cdot \left(\mathbf{X} - b_1^{\otimes 3}\right)\right\| \leqslant \left(100\varepsilon + \sqrt{\frac{1+\rho}{\alpha}}\right) \|g^{\|}\|$$

Finally, we consider the event

$$\mathcal{E}_{b_1,E}(\theta) \stackrel{\text{def}}{=} \left\{ \left\| \left(\sum_{j=1}^{d^2} g_j^{\perp} \cdot T_j\right) b_1 \right\|_2, \left\| \left(\sum_{j=1}^{d^2} g_j^{\perp} \cdot T_j\right)^{\top} b_1 \right\|_2 \leqslant 100\varepsilon \cdot \sqrt{2(1+\theta)} \right\}$$

First we consider the following decomposition

$$\left(\sum_{j=1}^{d^2} g_j^{\perp} \cdot T_j\right) b_1 = \sum_{j=1}^{d^2} g_j^{\perp} \cdot \left(\mathbf{X} - b_1^{\otimes 3}\right)_j b_1 + \sum_{j=1}^{d^2} g_j^{\perp} \cdot E_j b_1$$

For the first term, let $X^{\perp} = \sum_{i=1}^{n'} b_i b_i^{\top}$ and $X_g^{\perp} = \sum_{i=1}^{n'} \langle b_i, g^{\perp}\rangle b_i b_i^{\top}$. Then since $X_g^{\perp} \preceq \left(\max_{1\leqslant i\leqslant n} \left|\langle b_i, g^{\perp}\rangle\right|\right) \cdot X^{\perp}$, we have

$$\begin{aligned}
\left\|\sum_{j=1}^{d^2} g_j^{\perp} \cdot \left(\mathbf{X} - b_1^{\otimes 3}\right)_j b_1\right\|^2 &= \left\|\sum_{i=1}^{n'} \langle b_i, g^{\perp}\rangle b_i b_i^{\top} b_1\right\|^2 \\
&= \langle b_1, (X_g^{\perp})^2 b_1\rangle \\
&\leqslant \left(\max_{1\leqslant i\leqslant n} \left|\langle b_i, g^{\perp}\rangle\right|^2\right) \langle b_1, (X^{\perp})^2 b_1\rangle
\end{aligned}$$

By Lemma 59 and the $(n,d)$-nicely-separated property, with probability $1 - o(1)$ we have $\|(X^{\perp})^2 - X^{\perp}\| \leqslant \tilde{O}\left(\frac{n}{d^{3/2}}\right)$. It then follows that

$$\begin{aligned}
\langle b_1, (X^{\perp})^2 b_1\rangle &\leqslant \langle b_1, (X^{\perp}) b_1\rangle + \tilde{O}\left(\frac{n}{d^{3/2}}\right) \\
&\leqslant \sum_{i=2}^{n'} \langle b_i, b_1\rangle^2 + \tilde{O}\left(\frac{n}{d^{3/2}}\right) \\
&\leqslant \tilde{O}\left(\frac{n}{d^2}\right) + \tilde{O}\left(\frac{n}{d^{3/2}}\right) \\
&\leqslant \tilde{O}\left(\frac{n}{d^{3/2}}\right)
\end{aligned}$$

The last step follows from the fact that $\sum_{i=2}^{n'} \langle b_i, b_1\rangle^2 \geqslant \tilde{O}\left(\frac{n}{d^{3/2}}\right)$ .Since with probability $1 - n^{-\Omega(\log n)}$ over $g$,

$$\max_{1\leqslant i\leqslant n'} \left|\langle b_i, g^{\perp}\rangle\right| \leqslant \sqrt{2\log^2 n}$$

we have

$$\left\|\sum_{j=1}^{d^2} g_j^{\perp} \cdot \left(\mathbf{X} - b_1^{\otimes 3}\right)_j b_1\right\| \leqslant \tilde{O}\left(\frac{n}{d^{3/2}}\right)$$

For the second term, by assumption we have

$$\mathbb{P}\left[\left\|\sum_{j=1}^{d^2} g_j^{\perp} \cdot E_j a\right\|_2 \leqslant 100\varepsilon\sqrt{2(1+\theta)}\right] \geqslant 1 - d^{-\theta}$$

It follows that

$$\mathbb{P}\left[\mathcal{E}_{a_1,E}(\theta)\right] \geqslant 1 - 2d^{-\theta}$$

Now since

$$\mathbb{P}_g\left[\mathcal{E}_{>1}(\rho) \cap \mathcal{E}_{b_1,E}(\theta) \mid \mathcal{G}_1(\alpha)\right] \geqslant 1 - d^{-\rho} - 2d^{-\theta} - n^{-\Omega(\log n)}$$

by the independence between $\mathcal{E}_{b_1,E}(\theta)$ and $\mathcal{G}_1(\alpha)$, we have

$$\mathbb{P}_g\left[\mathcal{E}_{>1}(\rho) \cap \mathcal{E}_{b_1,E}(\theta) \cap \mathcal{G}_1(\alpha)\right] = \mathbb{P}_g\left[\mathcal{E}_{>1}(\rho) \cup \mathcal{E}_{a_1,E}(\theta) \mid \mathcal{G}_1(\alpha)\right] \mathbb{P}_g\left[\mathcal{G}_1(\alpha)\right] \geqslant (1-d^{-\rho}-2d^{-\theta})\Theta(n^{-\alpha})$$

Now we write $M_g = cb_1b_1^\top + N$. By setting $\rho, \theta = \frac{\log\log n}{\log n}$, and $\alpha = (1+2\rho) \geqslant (1+\frac{1}{\log n})^2(1+\rho)$, we have all three conditions are satisfied when $\mathcal{E}_{>1}(\rho) \cap \mathcal{E}_{b_1,E}(\theta) \cap \mathcal{G}_1(\alpha)$ holds. Indeed, by event $\mathcal{G}_1(\alpha)$ and $\mathcal{E}_{>1}(\rho)$, we have $c = \|b_1\|\|g^{\|}\| \geqslant \sqrt{4\alpha\log d}$ and $\|N\| \leqslant \left(100\varepsilon + \sqrt{\frac{1+\rho}{\alpha}}\right)\|g^{\|}\| \leqslant (1+\frac{1}{\log n})c$; By event $\mathcal{E}_{b_1,E}(\theta)$, $\|Nb_1\|, \|N^\top b_1\| \leqslant 100\varepsilon \cdot \sqrt{2(1+\theta)} + \left\|\sum_j g_j^{\|} E_j\right\| + \left\|\sum_j g_j^{\|} \cdot \left(\mathbf{X} - b_1^{\otimes 3}\right)\right\| \leqslant \frac{c}{\mathrm{polylog}(d)}$. ■

### J.3.2. RECOVERING CONSTANT FRACTION OF COMPONENTS

**Lemma 68 (Restatement of Lemma 23)** *Consider the setting of Lemma 14. Let $S_0 \subseteq [n]$ be of size $n' \leqslant n$ and assume that the set $\{a_i \mid i \in S_0\}$ is $(n,d)$-nicely separated. Consider the matrix $M_g$ and its top right singular vector $u_r \in \mathbb{R}^{d^2}$ obtained in one iteration of Algorithm 20. Then there exists a set $S \subseteq S_0$, such that for each $i \in S$, it holds with probability $\tilde{\Theta}(d^{-2})$ that*

- $\langle u, Ra_i^{\otimes 2}\rangle \geqslant 1 - \frac{1}{\mathrm{polylog}(d)}$.
- *the ratio between largest and second largest singular values of $M_g$ is larger than $1+\frac{1}{\mathrm{polylog}(d)}$*

To prove the lemma above we will use a lemma on getting estimation vector from the spectral gap, which already appears in the previous literature:

**Lemma 69 (Lemma 4.7 in Schramm and Steurer (2017))** *Let $M_g$ be a $\mathbb{R}^{n\times n}$ symmetric matrix s.t $M_g = cvv^\top + N$ where $v$ has unit norm, $c \geqslant (1+\delta)\|N\|$, and $\|Nv\|, \|vN\| \leqslant \gamma\|v\|^2$. Suppose $\gamma \leqslant \frac{\delta}{\mathrm{polylog}(d)}$, then the top eigenvector of $M_g$ denoted by $u$ satisfies:*

$$\langle u, v\rangle^2 \geqslant 1 - \frac{1}{\mathrm{polylog}(d)}$$

*Further the ratio between largest and second largest singular values of $M_g$ is larger than $1 + O\left(\frac{1}{\mathrm{polylog}(d)}\right)$.*

**Proof** [Proof of Lemma 23] W.l.o.g assume that $S_0 = [n']$. For $i \in [n']$, we denote $b_i := Ra_i^{\otimes 2}$. Combining Lemma 69, Lemma 22, for some $S \subseteq [n']$ with size at least $0.99n'$, for each $i \in S$, with probability $\tilde{\Theta}(d^{-2})$ over $g$, we have $M_g = cb_ib_i^\top + N$, where $\|b_i\| = 1 \pm \tilde{O}\left(\frac{1}{\sqrt{d}}\right)$ and $|c| \geqslant (1 + \frac{1}{\log(d)})\|N\|$ , and $\|Nv\|, \|vN\| \leqslant \frac{1}{\operatorname{polylog}(d)}$.

Now by Lemma 69, there exists unit norm vector $u \in \{u_L, u_R\}$ s.t $\langle u, Ra_i^{\otimes 2}\rangle^2 \geqslant 1 - \frac{1}{\operatorname{polylog}(d)}$. Since $\left\|Ra_i^{\otimes 2} - a_i^{\otimes 2}\right\| \leqslant \tilde{O}\left(\frac{1}{\sqrt{d}}\right)$ , it follows that $\left|\langle u, Ra_i^{\otimes 2}\rangle\right| \geqslant 1 - \frac{1}{\operatorname{polylog}(d)}$. ■

**Lemma 70 (Restatement of Lemma 24)** *Consider the setting of Lemma 14. Suppose for some unit norm vector $a \in \mathbb{R}^d$, and unit vector $u \in \mathbf{R}^{d^2}$, $\langle u, Ra^{\otimes 2}\rangle \geqslant 1 - \frac{1}{\operatorname{polylog}(d)}$. Then flattening $u$ into a $d \times d$ matrix $U$, the top left or right singular vector of $U$ denoted by $v$ will satisfy $\langle a, v\rangle^2 \geqslant 1 - \frac{1}{\operatorname{polylog}(d)}$.*

**Proof** Since $\|Ra_i^{\otimes 2} - a_i^{\otimes 2}\|^2 \leqslant \tilde{O}(\frac{1}{\sqrt{d}})$, we have $\langle u, a^{\otimes 2}\rangle \geqslant 1 - \frac{1}{\operatorname{polylog}(d)}$. Let the singular value decomposition of $U$ be $U = \sum_{i=1}^d \sigma_i w_i v_i^\top$, where $\sigma_i$ are singular vectors. Then by the best rank-1 approximation property of $\sigma_1 w_1 v_1^\top$, we have $\|\sigma_1 w_1 v_1^\top - U\|_F \leqslant \|aa^\top - U\|_F$. By triangle inequality, we have $\|\sigma_1 w_1 v_1^\top - aa^\top\|_{\mathrm{F}} \leqslant 2\|aa^\top - U\|_{\mathrm{F}}$. Since $\langle u, a^{\otimes 2}\rangle \geqslant 1 - \frac{1}{\operatorname{polylog}(n)}$, we have $\|U - aa^\top\|_{\mathrm{F}} \leqslant \frac{1}{\operatorname{polylog}(d)}$. It follows that $\|\sigma_1 w_1 v_1^\top - aa^\top\|_{\mathrm{F}} \leqslant \frac{1}{\operatorname{polylog}(d)}$. Since $\sigma_1 \leqslant 1$, we have $2\sigma_1\langle w_1, a\rangle\langle v_1, a\rangle \geqslant 1 + \sigma_1^2 - \frac{1}{\operatorname{polylog}(d)}$, which implies that $\langle w_1, a\rangle\langle v_1, a\rangle \geqslant 1 - \frac{1}{\operatorname{polylog}(d)}$. Now since $w_1, v_1, a$ has unit norm, we have $\langle w_1, a\rangle^2, \langle v_1, a\rangle^2 \geqslant 1 - \frac{1}{\operatorname{polylog}(d)}$. ■

**Lemma 71 (Restatement of Lemma 21)** *Let $S_0 \subseteq [n]$ be of size $d \leqslant n' \leqslant n$ and assume that the set $\{a_i \mid i \in S_0\}$ is $(n, d)$-nicely separated. Consider $v_l$ and $v_r$ in Algorithm 20, then there exists a set $S \subseteq S_0$ of size $m \geqslant 0.99n'$ such that for each $i \in S$ it holds with probability $\tilde{\Theta}(d^{-2})$ that $\max_{v \in \{\pm v_l, \pm v_r\}}\langle v, a_i\rangle \geqslant 1 - \frac{1}{\operatorname{polylog}(d)}$.*

**Proof** Combining Lemma 23 and Lemma 24, we have the claim. ■

### J.3.3. PRUNING LIST OF COMPONENTS

**Lemma 72 (Restatement of Theorem 25)** *Let $S$ be the set of vector computed in Step 3 of Algorithm 20 and $S'$ be the subset of components of Lemma 21 then for each $b \in S$ there exists a unique $i \in S'$ such that $\langle b, a_i\rangle \geqslant 1 - \frac{1}{\operatorname{polylog} d}$.*

In order to prove this we will use the following two facts.

**Fact 73** *Let $a, b_1, b_2 \in \mathbb{R}^d$ be unit norm vectors. If $\langle a, b_1\rangle \geqslant 1 - \delta$ and $\langle a, b_2\rangle \geqslant 1 - \delta$, then $\langle b_1, b_2\rangle \geqslant 1 - 2\delta$*

**Proof** Since we have $\|a - b_1\|^2 = 2 - 2\langle a, b_1\rangle \leqslant 2\delta$ and same for $\|a - b_1\|$, it follows that

$$\begin{aligned}\langle b_1, b_2\rangle &= \langle a, a\rangle + \langle b_1, b_2 - a\rangle + \langle b_1 - a, b_2\rangle \\ &\geqslant 1 - 2\sqrt{2\delta}\end{aligned}$$

■

**Fact 74** *Let $a_1, a_2, b_1, b_2 \in \mathbb{R}^d$ be unit norm vector such that $\langle a_1, b_1\rangle \geqslant 1-\delta_1$, $\langle a_2, b_2\rangle \geqslant 1-\delta_1$, and $|\langle a_1, a_2\rangle| \leqslant \delta_2$. Then $\langle b_1, b_2\rangle \leqslant \frac{\delta_2+8\delta_1}{2}$.*

**Proof** Since $\langle a_1, b_1\rangle = \frac{2-\|a_1-b_1\|^2}{2}$ and $\langle a_2, b_2\rangle = \frac{2-\|a_2-b_2\|^2}{2}$, we have $\|a_1-b_1\| \leqslant \sqrt{2\delta_1}$ and $\|a_2-b_2\| \leqslant \sqrt{2\delta_1}$. For the same reason, $\|a_1-a_2\|^2 = 2-2\langle a_1, a_2\rangle \geqslant 2-2\delta_2$ By triangle inequality, we then have $\|b_1-b_2\| \geqslant \sqrt{2-\delta_2} - 2\sqrt{2\delta_1}$. It then follows that

$$\langle b_1, b_2\rangle = \frac{2-\|b_1-b_2\|^2}{2} \geqslant \frac{\delta_2+8\delta_1}{2}$$

■

Now we are ready to prove Theorem 25.

**Proof** [Proof of Theorem 25] By the discussion above Theorem 25 we know that for $C$ computed in Step 1 of Algorithm 20 it holds that

$$\forall i \in S' \colon \max_{b\in\mathcal{C}} |\langle b, a_i\rangle| \geqslant 1 - \frac{1}{\operatorname{polylog}(n)}$$

and

$$\forall b \in C \colon \max_{i\in S} |\langle b, a_i\rangle| \geqslant 1 - \frac{1}{\operatorname{polylog}(n)}$$

To prove the lemma it is sufficient to show that

– for each $b_j \in S'$ there exists a unique $i \in S$ such that

$$\langle b_j, a_i\rangle \geqslant 1-\delta$$

– for each $i \in S$ there exists a unique $b_j \in S'$ such that

$$\langle b_j, a_i\rangle \geqslant 1-\delta$$

Regarding the first point: By the first condition in equation J.3.3, for each $j \in S'$, there exists $i \in S$ such that $\langle b_j, a_i\rangle \geqslant 1-\delta$. For the sake of contradiction assume that there exists $k \in S, k \neq i$ such that $\langle b_j, a_k\rangle \geqslant 1-\delta$. By our assumptions on the components (cf. Theorem 15) we have $|\langle a_i, a_k\rangle| \leqslant \delta$. Thus, invoking Fact 74 with $b_1, b_2 = b_i$, $a_1 = a_i$, and $a_2 = a_k$, we get that $1 = \langle b_j, b_j\rangle \leqslant \frac{9}{2}\cdot\delta < 1$. Hence, for each $b_j \in S'$, there is exactly one $i \in [n]$ such that $\langle b_j, a_i\rangle \geqslant 1-\delta$.

Regarding the second point: By Fact 73, for any two vectors $b_{j_1}, b_{j_2}$ s.t $\langle b_{j_1}, a\rangle \geqslant 1-\delta$ and $\langle b_{j_2}, a\rangle \geqslant 1-\delta$, we must have $\langle b_{j_1}, b_{j_2}\rangle \geqslant 1-2\delta \geqslant 0.99$. Thus by the construction of $S'$, for each $a_i$ there is at most one $b_j \in S'$, such that $\langle a_i, b_j\rangle \geqslant 1-\delta$. On the other hand suppose there exists $i \in S$ such that $\max_{j\in S'}\langle b_j, a_i\rangle \leqslant 1-\delta$. Then for each $b_j \in S'$, we have $\langle b_j, a_\ell\rangle \geqslant 1-\delta$ for some $\ell \neq i$. Further by the list recovery guarantee, there exist $k \in [L]$ s.t $\langle b_k, a_i\rangle \geqslant 1-\delta$. This means that by Fact 74, for any vector $b$ in $S'$, $\langle b_k, b\rangle \leqslant O(\delta)$. By construction, such vector $b_k$ should be contained in the set $S'$, which leads to contradiction.

■

### J.4. Full recovery

In this section, we prove a technical lemma used for the proof of Theorem 6.

**Lemma 75** *For $d \leqslant n \leqslant O\left(d^{3/2}/\operatorname{polylog}(d)\right)$ and $m \geqslant d$, suppose vectors $a_1, a_2, \ldots, a_m$ are $(n,d)$ nicely-separated, and $c_1, c_2, \ldots, c_m \in \mathbb{R}^d$ has norm bounded by $\tilde{O}\left(\frac{\sqrt{n}}{d}\right)$. Suppose for each $j \in [6]$, either for each $i \in [m]$, $g_i^{(j)} = a_i$, or for each $i \in [m]$, $g_i^{(j)} = c_i$. Further suppose that for at least one of $j \in \{1,2,3\}$ and at least one of $j \in \{4,5,6\}$ , $g_i^{(j)} = c_i$. Suppose $M \in \mathbb{R}^{m \times m}$ has entries.*

$$M_{i,j} = \langle g_i^{(1)}, g_j^{(4)} \rangle \langle g_i^{(2)}, g_j^{(5)} \rangle \langle g_i^{(3)}, g_j^{(6)} \rangle$$

*Then the frobenius norm of $M_{i,j}$ is bounded by $\tilde{O}\left(\sqrt{\frac{n}{d\sqrt{d}}}\right)$*

**Proof** We divide the choices of $g^{(1)}, g^{(2)}, \ldots, g^{(6)}$ into 4 different cases, according to the inner product in $\langle g_i^{(1)}, g_j^{(4)} \rangle$, $\langle g_i^{(2)}, g_j^{(5)} \rangle$, $\langle g_i^{(3)}, g_j^{(6)} \rangle$. Particularly if $g_i^{(t)} = a_i$ and $g_j^{(t+3)} = c_j$, or $g_i^{(t)} = c_i$ and $g_j^{(t+3)} = a_j$, then we call $\langle g_i^{(t)}, g_j^{(t+3)} \rangle$ a cross inner product pair.

(1). There are no cross inner product pairs, i.e

$$\left|\left\{k \in [3] : \{g_i^{(k)}, g_j^{(k+3)}\} \in \{a_i, a_j\}, \{c_i, c_j\}\right\}\right| = 3\,.$$

Since $a_i$ satisfies the $(n,d)$ nicely-separated assumption, $\langle a_i, a_j \rangle^2 \leqslant \tilde{O}\left(\frac{1}{d}\right)$. Since $\|c_i\| \leqslant \frac{\sqrt{n}}{d}$, $\langle c_i, c_j \rangle^2 \leqslant \tilde{O}\left(\frac{1}{d}\right)$. In this case we have

$$\|M\|_{\mathrm{F}}^2 = \sum_{\substack{i,j \in [n] \\ i \neq j}} \langle g_i^{(1)}, g_j^{(4)} \rangle^2 \langle g_i^{(2)}, g_j^{(5)} \rangle^2 \langle g_i^{(3)}, g_j^{(6)} \rangle^2 \leqslant n^2 \cdot (\frac{1}{d})^3 = \frac{n^2}{d^3}$$

(2). There is one cross inner product pair, i.e

$$\left|\left\{k \in [3] : \{g_i^{(k)}, g_j^{(k+3)}\} \in \{\{a_i, a_j\}, \{c_i, c_j\}\}\right\}\right| = 2\,.$$

Since $a_i$ satisfies $(n,d)$ nicely-separated assumption , we have $\langle a_i, a_j \rangle^2 \leqslant \tilde{O}\left(\frac{1}{d}\right)$, and

$$\left\|\sum_{j \in [n]} a_j a_j^\top\right\| \leqslant \frac{n}{d}$$

Further $\|c_i\| \leqslant \frac{\sqrt{n}}{d}$ and $\langle c_i, c_j \rangle^2 \leqslant \left(\frac{n}{d^2}\right)^2 \leqslant \tilde{O}\left(\frac{1}{d}\right)$. Thus we have

$$\begin{aligned}
\|M\|_{\mathrm{F}}^2 &= \sum_{\substack{i,j \in [n] \\ i \neq j}} \langle g_i^{(1)}, g_j^{(4)} \rangle^2 \langle g_i^{(2)}, g_j^{(5)} \rangle^2 \langle g_i^{(3)}, g_j^{(6)} \rangle^2 \\
&\leqslant \sum_{\substack{i,j \in [n] \\ i \neq j}} \frac{1}{d^2} \cdot \langle c_i, a_j \rangle^2
\end{aligned}$$

$$
\begin{aligned}
&= \frac{1}{d^2} \sum_{i \in [n]} c_i^\top \left( \sum_{\substack{j \in [n] \\ j \neq i}} a_j a_j^\top \right) c_i \\
&\leqslant \frac{1}{d^2} \sum_{i \in [n]} \|c_i\|^2 \left\| \sum_{\substack{j \in [n] \\ j \neq i}} a_j a_j^\top \right\| \\
&\leqslant \frac{1}{d^2} \cdot n \cdot \frac{n}{d^2} \cdot \frac{n}{d} \\
&\leqslant o\left(n^2/d^3\right)
\end{aligned}
$$

(3). There are 2 cross inner product pairs, i.e,

$$\left|\left\{k \in [3] : \{g_i^{(k)}, g_j^{(k+3)}\} \in \{\{a_i, a_j\}, \{c_i, c_j\}\}\right\}\right| = 1\,.$$

Since $a_i$ satisfies $(n,d)$ nicely-separated assumption , we have $\langle a_i, a_j\rangle^2 \leqslant \tilde{O}\left(\frac{1}{d}\right)$. Further $\|c_i\| \leqslant \frac{\sqrt{n}}{d}$ and $\langle c_i, c_j\rangle^2 \leqslant \left(\frac{n}{d^2}\right)^2 \leqslant \tilde{O}\left(\frac{1}{d}\right)$. We consider two different sub-cases:

- $M_{i,j} = \langle a_i, c_j\rangle^2 \langle a_j, c_i\rangle^2 \langle c_i, c_j\rangle^2$ or $M_{i,j} = \langle a_i, c_j\rangle^2 \langle a_j, c_i\rangle^2 \langle a_i, a_j\rangle^2$. By the $(n,d)$ nicely-separated assumption on $a_j$, we have

$$\left\| \sum_{j \in [n]} a_j a_j^\top \right\| \leqslant \frac{n}{d}$$

  Thus in this case we have

$$
\begin{aligned}
\|M\|_F^2 &\leqslant \frac{1}{d} \sum_{\substack{i,j \in [n] \\ i \neq j}} \langle a_i, c_j\rangle^2 \langle a_j, c_i\rangle^2 \\
&= \frac{1}{d} \cdot \frac{1}{\sqrt{d}} \sum_{\substack{i,j \in [n] \\ i \neq j}} c_j^\top a_i a_i^\top c_j \\
&\leqslant \frac{1}{d} \cdot \frac{1}{\sqrt{d}} \cdot \left( \sum_j \|c_j\|^2 \right) \cdot \left\| \sum_{\substack{i \in [n] \\ i \neq j}} a_i a_i^\top \right\| \\
&\leqslant \frac{1}{d} \cdot \frac{1}{\sqrt{d}} \cdot \frac{n}{\sqrt{d}} \cdot \frac{n}{d} \\
&\leqslant \tilde{O}\left(\frac{n^2}{d^3}\right)
\end{aligned}
$$

- $M_{i,j} = \langle a_i, c_j\rangle^4 \langle c_i, c_j\rangle^2$ or $M_{i,j} = \langle a_i, c_j\rangle^4 \langle a_i, a_j\rangle^2$. By the $(n,d)$ nicely-separated assumption on $a_j$, we have

$$\left\| \sum_{j \in [n]} (a_j a_j^\top)^{\otimes 2} \right\| \leqslant \frac{n}{d}$$

In this case we have

$$\begin{aligned}
\|M\|_F^2 &\leqslant \frac{1}{d}\sum_{i\neq j}\langle a_i, c_j\rangle^4 \\
&\leqslant \frac{1}{d}\cdot\left(\sum_{j=1}^{n}\|c_j\|^4\right)\cdot\left\|\sum_i a_i^{\otimes 2}\left(a_i^{\otimes 2}\right)^\top\right\| \\
&\leqslant \tilde{O}(\frac{1}{d}\cdot\frac{n}{d}\cdot\frac{n}{d}) \\
&\leqslant \tilde{O}\left(n^2/d^3\right)
\end{aligned}$$

(4). For the final case, we have three cross inner product pairs, i.e

$$\left|\left\{k\in[3]:\{g_i^{(k)}, g_j^{(k+3)}\}\in\{\{a_i,a_j\},\{c_i,c_j\}\}\right\}\right| = 0\,.$$

Then w.l.o.g let $M_{i,j} = \langle a_i, c_j\rangle^2\langle a_j, c_i\rangle$.

In this case, we use the fact that

$$\begin{aligned}
\|M\|_F^2 &= \sum_{\substack{i,j\in[n]\\ i\neq j}}\langle a_i, c_j\rangle^4\langle a_j, c_i\rangle^2 \\
&= \sum_{\substack{i,j\in[n]\\ i\neq j}}\langle a_i^{\otimes 2} - Ra_i^{\otimes 2} + Ra_i^{\otimes 2}, c_j^{\otimes 2}\rangle^2\langle a_j, c_i\rangle^2 \\
&\leqslant 2\sum_{\substack{i,j\in[n]\\ i\neq j}}\langle a_i^{\otimes 2} - Ra_i^{\otimes 2}, c_j^{\otimes 2}\rangle^2\langle a_j, c_i\rangle^2 + 2\sum_{\substack{i,j\in[n]\\ i\neq j}}\langle Ra_i^{\otimes 2}, c_j^{\otimes 2}\rangle^2\langle a_j, c_i\rangle^2
\end{aligned}$$

For the first term, by the $(n,d)$ nicely-separated property, we have $\left\|Ra_i^{\otimes 2} - a_i^{\otimes 2}\right\|^2 \leqslant \tilde{O}\left(\frac{1}{d}\right)$, Thus

$$\begin{aligned}
\sum_{\substack{i,j\in[n]\\ i\neq j}}\langle a_i^{\otimes 2} - Ra_i^{\otimes 2}, c_j^{\otimes 2}\rangle^2\langle a_j, c_i\rangle^2 &\leqslant \tilde{O}\left(\frac{1}{d^2}\right)\sum_{\substack{i,j\in[n]\\ i\neq j}}\langle a_j, c_i\rangle^2 \\
&= \tilde{O}\left(\frac{1}{d^2}\right)\sum_{\substack{i,j\in[n]\\ i\neq j}} c_i^\top a_j a_j^\top c_i \\
&\leqslant \tilde{O}\left(\frac{1}{d^2}\right)\cdot n\cdot\max_i\|c_i\|^2\cdot\left\|\sum_{j\in[n]\setminus\{i\}} a_j a_j^\top\right\| \\
&\leqslant \tilde{O}\left(\frac{1}{d^2}\right)\cdot n\cdot\tilde{O}\left(\frac{n}{d^2}\right)\cdot\tilde{O}\left(\frac{n}{d}\right) \\
&\leqslant \tilde{O}\left(\frac{n^3}{d^5}\right) = o(n^2/d^3)
\end{aligned}$$

For the second term, by the $(n,d)$ nicely-separated property of $a_i$, we have $\left\|\sum_i Ra_i^{\otimes 2}\left(Ra_i^{\otimes 2}\right)^\top\right\| \leqslant 2$. We then have

$$\begin{aligned}
\sum_{\substack{i,j\in[n]\\ i\neq j}} \langle Ra_i^{\otimes 2}, c_j^{\otimes 2}\rangle^2 \langle a_j, c_i\rangle^2 &\leqslant \frac{1}{\sqrt{d}} \sum_{\substack{i,j\in[n]\\ i\neq j}} \langle Ra_i^{\otimes 2}, c_j^{\otimes 2}\rangle^2 \\
&\leqslant \frac{1}{\sqrt{d}} \cdot \left(\sum_{j=1}^{n} \|c_j\|^4\right) \cdot \left\|\sum_i Ra_i^{\otimes 2}\left(Ra_i^{\otimes 2}\right)^\top\right\| \\
&\leqslant \tilde{O}\left(\frac{1}{\sqrt{d}} \cdot \frac{n}{d}\right) \cdot 2 \\
&\leqslant \tilde{O}\left(\frac{n}{d\sqrt{d}}\right)
\end{aligned}$$

Thus overall we can conclude that for each choice of $g$, $\|M\|_{\mathrm{F}} \leqslant \tilde{O}\left(\sqrt{\frac{n}{d\sqrt{d}}}\right)$. ■