Self-Calibration and Biconvex Compressive Sensing

Shuyang Ling and Thomas Strohmer
Department of Mathematics
University of California at Davis
Davis CA 95616
{syling,strohmer}@math.ucdavis.edu

January 26, 2015

Abstract

The design of high-precision sensing devises becomes ever more difficult and expensive. At the same time, the need for precise calibration of these devices (ranging from tiny sensors to space telescopes) manifests itself as a major roadblock in many scientific and technological endeavors. To achieve optimal performance of advanced high-performance sensors one must carefully calibrate them, which is often difficult or even impossible to do in practice. In this work we bring together three seemingly unrelated concepts, namely Self-Calibration, Compressive Sensing, and Biconvex Optimization. The idea behind self-calibration is to equip a hardware device with a smart algorithm that can compensate automatically for the lack of calibration. We show how several self-calibration problems can be treated efficiently within the framework of biconvex compressive sensing via a new method called SparseLift. More specifically, we consider a linear system of equations $y = DAx$, where both x and the diagonal matrix D (which models the calibration error) are unknown. By "lifting" this biconvex inverse problem we arrive at a convex optimization problem. By exploiting sparsity in the signal model, we derive explicit theoretical guarantees under which both x and D can be recovered exactly, robustly, and numerically efficiently via linear programming. Applications in array calibration and wireless communications are discussed and numerical simulations are presented, confirming and complementing our theoretical analysis.

1 Introduction

The design of high-precision sensing devises becomes ever more difficult and expensive. Meanwhile, the need for precise calibration of these devices (ranging from tiny sensors to space telescopes) manifests itself as a major roadblock in many scientific and technological endeavors. To achieve optimal performance of advanced high-performance sensors one must carefully calibrate them, which is often difficult or even impossible to do in practice. It is therefore necessary to build into the sensors and systems the capability of *self-calibration* – using the information collected by the system to simultaneously adjust the calibration parameters and perform the intended function of the system (image reconstruction, signal estimation, target detection, etc.). This is often a challenging problem requiring advanced mathematical solutions. Given the importance of this problem there has been a substantial amount of prior work in this area. Current self-calibration algorithms are based on joint estimation of the parameters of interest and the calibration

parameters using standard techniques such as maximum likelihood estimation. These algorithms tend to be very computationally expensive and can therefore be used only in relatively simple situations. This has prevented their widespread applications in advanced sensors.

In this paper we take a step toward developing a new framework for self-calibration by bringing together three seemingly unrelated concepts, namely Self-Calibration, Compressive Sensing, and Biconvex Optimization. By "lifting" the problem and exploiting sparsity in the signal model and/or the calibration model we express the biconvex problem of self-calibration as a convex optimization problem, which can be solved efficiently via linear programming. This new method, called *SparseLift*, ensures equivalence between the biconvex problem and the convex problem at the price of a modest increase in the number of measurements compared to the already perfectly calibrated problem. SparseLift is inspired by PhaseLift, an algorithm for phase retrieval [11, 8], and by the blind deconvolution framework via convex programming due to Ahmed, Recht, and Romberg [1].

Self-calibration is a field of research in its own right, and we do not attempt in this paper to develop a framework that can deal with any kind of self-calibration problems. Instead our goal is more modest—we will consider a special, but at the same time in practice very important, class of self-calibration problems. To that end we first note that many self-calibration problems can be expressed as (either directly or after some modification such as appropriate discretization)

$$\boldsymbol{y} = \boldsymbol{A}(\boldsymbol{h})\boldsymbol{x} + \boldsymbol{w}, \tag{1.1}$$

where $\boldsymbol{y}$ represents the given measurements, $\boldsymbol{A}(\boldsymbol{h})$ is the system matrix depending on some unknown calibration parameters $\boldsymbol{h}$ and $\boldsymbol{x}$ is the signal (function, image, ...) one aims to recover and $\boldsymbol{w}$ is additive noise. As pointed out, we do not know $\boldsymbol{A}(\boldsymbol{h})$ exactly due to the lack of calibration. Our goal is to recover $\boldsymbol{x}$ and $\boldsymbol{h}$ from $\boldsymbol{y}$. More precisely, we want to solve

$$\min_{\boldsymbol{h},\boldsymbol{x}} \|\boldsymbol{A}(\boldsymbol{h})\boldsymbol{x} - \boldsymbol{y}\|_2. \tag{1.2}$$

If $\boldsymbol{A}(\boldsymbol{h})$ depends linearly on $\boldsymbol{h}$, then (1.2) is a *bilinear* inverse problem. In any case, the optimization problem (1.2) is not convex, which makes its numerical solution rather challenging.

In many applications (1.1) represents an underdetermined system even if $\boldsymbol{A}(\boldsymbol{h})$ is perfectly known. Fortunately, we can often assume $\boldsymbol{x}$ to be sparse (perhaps after expressing it in a suitable basis, such as a wavelet basis). Moreover, in some applications $\boldsymbol{h}$ is also sparse or belongs to a low-dimensional subspace. Hence this suggests considering

$$\min_{\boldsymbol{h},\boldsymbol{x}} \|\boldsymbol{A}(\boldsymbol{h})\boldsymbol{x} - \boldsymbol{y}\|_2 + f(\boldsymbol{x}, \boldsymbol{h}), \tag{1.3}$$

where $f(\boldsymbol{x}, \boldsymbol{h})$ is a convex function that promotes sparsity in $\boldsymbol{x}$ and/or $\boldsymbol{h}$. Often it is reasonable to assume that (1.3) is *biconvex*, i.e., it is convex in $\boldsymbol{h}$ for fixed $\boldsymbol{x}$ and convex in $\boldsymbol{x}$ for fixed $\boldsymbol{h}$. In other cases, one can find a close convex relaxation to the problem. We can thus frequently interpret (1.3) as a *biconvex compressive sensing* problem.

However, useful as (1.3) may be, it is still too difficult and too general to develop a rigorous and efficient numerical framework for it. Instead we focus on the important special case

$$\boldsymbol{y} = \boldsymbol{D}(\boldsymbol{h})\boldsymbol{A}\boldsymbol{x}, \tag{1.4}$$

where $\boldsymbol{D}(\boldsymbol{h})$ is a diagonal matrix depending on the unknown parameters $\boldsymbol{h}$ and $\boldsymbol{A}$ is known. The reader may think that we finally reached a level of simplicity that is trivial to analyze mathematically and no

longer useful in practice. However, as is often the case in mathematics, a seemingly simple model can be deceptive. As a matter of fact, on the one hand a rigorous analysis of the diagonal calibration model above requires quite non-trivial mathematical tools, and on the other hand this "simple" model does arise in numerous important applications, including radar, wireless communications, geolocation, direction finding, X-ray crystallography, cryo-electron microscopy, astronomy, medical imaging, etc. We will briefly discuss two such applications in more detail below.

Before moving on to applications, we pause for a moment to discuss the issue of identifiability. It is clear that if $(\boldsymbol{D}, \boldsymbol{x})$ is a pair of solutions to (1.4), then so is $(\alpha \boldsymbol{D}, \frac{1}{\alpha}\boldsymbol{x})$ for any non-zero α. Therefore, it is only possible to uniquely reconstruct $\boldsymbol{D}$ and $\boldsymbol{x}$ up to a scalar. Fortunately for most applications this is not an issue, and we will henceforth ignore this scalar ambiguity when we talk about recovering $\boldsymbol{D}$ and $\boldsymbol{x}$.

1.1 Array self-calibration and direction-of-arrival estimation

Consider the problem of direction-of-arrival estimation using an array with L antennas. Assuming that n signals are impinging on the array, the output vector (using the standard narrowband model) is given by

$$\boldsymbol{y}(t) = \sum_{k=1}^{n} \boldsymbol{D}(\boldsymbol{h})\boldsymbol{a}(\theta_k, \psi_k)x_k(t) + \boldsymbol{w}(t). \tag{1.5}$$

By discretizing the array manifold $\boldsymbol{a}(\theta_k, \psi_k)$, where (θ, ψ) are the azimuth and elevation angles, respectively, we can arrange equation (1.5) as

$$\boldsymbol{y} = \boldsymbol{D}(\boldsymbol{h})\boldsymbol{A}\boldsymbol{x} + \boldsymbol{w}, \tag{1.6}$$

where $\boldsymbol{w}$ now also includes discretization error. Here $\boldsymbol{a}(\theta, \psi)$ is a completely known function of (θ, ψ) and does not depend on calibration parameters, so $\boldsymbol{A}$ is a known matrix. The calibration is captured by the matrix $\boldsymbol{D}$. An important special case is when $\boldsymbol{D}(\boldsymbol{h})$ is a diagonal matrix whose diagonal elements represent unknown complex gains associated with each of the antennas. The calibrating issue may come from position offsets of each antenna and/or from gain discrepancies caused by changes in the temperature and humidity of the environment. If the true signal directions $(\bar{\theta}_1, \bar{\psi}_1), \ldots, (\bar{\theta}_n, \bar{\psi}_n)$ lie on the discretized grid of angles $(\theta_1, \psi_1) \cdots, (\theta_N, \psi_N)$ corresponding to the columns of $\boldsymbol{A}$, then the $N \times 1$ vector $\boldsymbol{x}$ is n-sparse, otherwise $\boldsymbol{x}$ is only approximately n-sparse. In the direction finding problem we usually have multiple snapshots of the measurements $\boldsymbol{y}(t)$, in which case $\boldsymbol{y}$ and $\boldsymbol{x}$ in (1.6) become matrices, whose columns are associated with different snapshots [18].

1.2 The Internet of Things, random access, and sporadic traffic

The Internet of Things (IoT) will connect billions of wireless devices, which is far more than the current Fourth-Generation (4G) wireless system can technically and economically accommodate. One of the many challenges of 5G, the future and yet-to-be-designed wireless communication system, will be its ability to manage the massive number of *sporadic traffic* generating IoT devices which are most of the time inactive, but regularly access the network for minor updates with no human interaction [34]. Sporadic traffic will dramatically increase in the 5G market and it is a common understanding among communications engineers that this traffic cannot be handled within the current 4G random access procedures. Dimensioning the channel access according to classical information and communication theory results in a severe waste of

resources which does not scale towards the requirements of the IoT. This means among others that the overhead caused by the exchange of certain types of information between transmitter and receiver, such as channel estimation, assignment of data slots, etc, has to be avoided. In a nutshell, if there is a huge number of devices each with the purpose to sporadically transmit a few bits of data, we must avoid transmitting each time a signal where the part carrying the overhead information is much longer than the part carrying the actual data. The question is whether this is possible.

In mathematical terms we are dealing with the following problem (somewhat simplified, but it captures the essence). For the purpose of exposition we consider the single user/single antenna case. Let $\boldsymbol{h}$ denote the channel impulse response and $\boldsymbol{x}$ the signal to be transmitted, where $\boldsymbol{x}$ is a sparse signal, its non-zero coefficients containing the few data that need to be transmitted. We encode $\boldsymbol{x}$ by computing $\boldsymbol{z} = \boldsymbol{Ax}$, where $\boldsymbol{A}$ is a fixed precoding matrix, known to the transmitter and the receiver. However, the non-zero locations of $\boldsymbol{x}$ are not known to the receiver, and nor is $\boldsymbol{h}$, since this is the overhead information that we want to avoid transmitting, as it will change from transmission to transmission. The received signal is given by the convolution of $\boldsymbol{h}$ and $\boldsymbol{z}$, i.e., $\boldsymbol{y} = \boldsymbol{h} * \boldsymbol{z}$. The important point here is that the receiver needs to recover $\boldsymbol{x}$ from $\boldsymbol{y}$, but does not know $\boldsymbol{h}$ either. This type of problem is often referred to as *blind deconvolution*. After applying the Fourier transform on both sides and writing $\boldsymbol{D} = \mathrm{diag}(\hat{\boldsymbol{h}})$ (where $\hat{}$ denotes the Fourier transform), we obtain $\hat{\boldsymbol{y}} = \boldsymbol{D}\widehat{\boldsymbol{Ax}}$, which is of the form (1.4) modulo a change of notation.

1.3   Notation and outline

Matrices are written in uppercase boldface letters (e.g. $\boldsymbol{X}$), while vectors are written in lowercase boldface letters (e.g. $\boldsymbol{x}$), and scalars are written in regular font or denoted by Greek letters. For a matrix $\boldsymbol{X}$ with entries X_{ij} we denote $\|\boldsymbol{X}\|_1 := \sum_{ij} |X_{ij}|$, $\|\boldsymbol{X}\|_*$ is the nuclear norm of $\boldsymbol{X}$, i.e., the sum of its singular values; $\|\boldsymbol{X}\|_\infty = \max_{ij} |X_{ij}|$, $\|\boldsymbol{X}\|$ is the operator norm of $\boldsymbol{X}$ and $\|\boldsymbol{X}\|_F$ is its Frobenius norm. $\mathrm{Tr}(\boldsymbol{X})$ denotes the trace of $\boldsymbol{X}$. If $\boldsymbol{x}$ is a vector, $\|\boldsymbol{x}\|$ is its 2-norm and $\|\boldsymbol{x}\|_0$ denotes the number of non-zero entries in $\boldsymbol{x}$, and $\bar{\boldsymbol{x}}$ denotes the complex conjugate of $\boldsymbol{x}$.

The remainder of the paper is organized as follows. In Section 2 we briefly discuss our contributions in relation to the state of the art in self-calibration. Section 3 contains the derivation of our approach and the main theorems of this paper. Sections 4 and 5 are dedicated to the proofs of the main theorems. We present numerical simulations in Section 6 and conclude in Section 7 with some open problems.

2   Related work and our contributions

Convex optimization combined with sparsity has become a powerful tool; it provides solutions to many problems which were believed intractable in the fields of science, engineering and technology. Two remarkable examples are *compressive sensing* and *matrix completion*. Compressive sensing [16, 12, 17], an ingenious way to efficiently sample sparse signals, has made great contributions to signal processing and imaging science. Meanwhile, the study of how to recover a low-rank matrix from a seemingly incomplete set of linear observations has attracted a lot of attention, see e.g. [5, 7, 29].

Besides drawing from general ideas in compressive sensing and matrix completion, the research presented in this work is mostly influenced by papers related to PhaseLift [3, 11, 26] and in particular by the very

inspiring paper on blind deconvolution by Ahmed, Recht, and Romberg [1]. PhaseLift provides a strategy of how to reconstruct a signal $\boldsymbol{x}$ from its quadratic measurements $y_i = |\langle \boldsymbol{x}, \boldsymbol{z}_i \rangle|^2$ via "lifting" techniques. Here the key idea is to lift a vector-valued quadratic problem to a matrix-valued linear problem. Specifically, we need to find a rank-one positive semi-definite matrix $\boldsymbol{X} = \boldsymbol{x}\boldsymbol{x}^*$ which satisfies linear measurement equations of $\boldsymbol{X}$: $\boldsymbol{y}_i = \text{Tr}(\boldsymbol{X}\boldsymbol{Z}_i)$ where $\boldsymbol{Z}_i = \boldsymbol{z}_i\boldsymbol{z}_i^*$. Generally problems of this form are NP-hard. However, solving the following nuclear norm minimization yields exact recovery of $\boldsymbol{x}\boldsymbol{x}^*$:

$$\min \|\boldsymbol{X}\|_*, \quad \text{subject to} \quad y_i = \text{Tr}(\boldsymbol{X}\boldsymbol{Z}_i), \boldsymbol{X} \succeq 0. \tag{2.1}$$

This idea of "Lifting" also applies to blind deconvolution [1], which refers to the recovery of two unknown signals from their convolution. The model can be converted into the form (3.1) via applying the Fourier transform and under proper assumptions. $\boldsymbol{X}_0 = \boldsymbol{h}_0\boldsymbol{x}_0^T$ can be recovered exactly by solving a nuclear norm minimization program in the following form,

$$\min \|\boldsymbol{X}\|_*, \quad \text{subject to} \quad y_i = \text{Tr}(\boldsymbol{X}\boldsymbol{A}_i), \quad \boldsymbol{A}_i \text{ is a rank-one matrix}, \tag{2.2}$$

if the number of measurements is at least larger than the dimensions of both $\boldsymbol{h}$ and $\boldsymbol{x}$. Compared with PhaseLift (2.1), the difference is that $\boldsymbol{X}$ can be asymmetric and there is of course no guarantee for positivity. Ahmed, Recht, and Romberg derived a very appealing framework for blind deconvolution via convex programming, and we borrow many of their ideas for our proofs.

The idea of our work is motivated by the two examples mentioned above and our results are related to and inspired by those of Ahmed, Recht, and Romberg. However, there are important differences in our setting. First, we consider the case where one of the signals is sparse. It seems unnecessary to use as many measurements as the signal length and solve (2.2) in that case. Indeed, as our results show we can take much fewer measurements in such a situation. This means we end up with a system of equations that is underdetermined even in the perfectly calibrated case (unlike [1]), which forces us to exploit sparsity to make the problem solvable. Thus [1] deals with a bilinear optimization problem, while we end up with a biconvex optimization problem. Second, the matrix $\boldsymbol{A}$ in [1] is restricted to be a Gaussian random matrix, whereas in our case $\boldsymbol{A}$ can be a random Fourier matrix as well. This additional flexibility is relevant in real applications. For instance a practical realization of the application in 5G wireless communications as discussed in Section 1.2 and Section 6.3 would never involve a Gaussian random matrix as spreading/coding matrix, but it could definitely incorporate a random Fourier matrix.

Bilinear compressive sensing problems have also been invesstigated in e.g. [14, 32, 24]. These papers focus on the issue of injectivity and the principle of identifiability. Except for the general setup, there is little overlap with our work. Self-calibration algorithms have been around in the literature for quite some time, see e.g. [19, 30, 2, 27] for a small selection, with blind deconvolution being a prominent special case, cf. [25] and the references therein. While many of these algorithms are definitely useful and interesting, most of them do not provide any guarantees of recovery. Also the proposed methods are very different from ours and do not involve sparsity, convex optimization, or ideas from compressive sensing. To the best of our knowledge, our method is the first work to bring self-calibration problems into the framework of biconvex compressive sensing. Uncertainties in the sensing matrix have been analyzed for instance in [22, 13], but with the focus on quantifiying the uncertainty, while our current work can be seen as a means to mitigate the uncertainty in the sensing matrix.

3   Self-calibration, biconvex optimization, and SparseLift

As discussed in the introduction, we are concerned with $\boldsymbol{y} = \boldsymbol{D}(\boldsymbol{h})\boldsymbol{A}\boldsymbol{x} + \boldsymbol{w}$, where the diagonal matrix $\boldsymbol{D}$ and $\boldsymbol{x}$ are unknown. Clearly, without further information it is impossible to recover $\boldsymbol{x}$ (or $\boldsymbol{D}$) from $\boldsymbol{y}$ since the number of unknowns is larger than the number of measurements. What comes to our rescue is the fact that in many applications additional knowledge about $\boldsymbol{D}$ and $\boldsymbol{x}$ is available. Of particular interest is the case where $\boldsymbol{x}$ is sparse and the diagonal entries of $\boldsymbol{D}$ belong to some known subspace.

Hence, suppose we are given
$$\boldsymbol{y} = \boldsymbol{D}\boldsymbol{A}\boldsymbol{x} + \boldsymbol{w}, \quad \boldsymbol{D} = \mathrm{diag}(\boldsymbol{B}\boldsymbol{h}), \tag{3.1}$$
where $\boldsymbol{y} \in \mathbb{C}^{L \times 1}$, $\boldsymbol{D} : \mathbb{C}^{L \times L}$, $\boldsymbol{h} \in \mathbb{C}^{k \times 1}$, $\boldsymbol{A} \in \mathbb{C}^{k \times N}$, $\boldsymbol{B} \in \mathbb{C}^{L \times k}$ and $\boldsymbol{x} \in \mathbb{C}^{N \times 1}$ is n-sparse. $\boldsymbol{D} = \mathrm{diag}(\boldsymbol{B}\boldsymbol{h})$ can be interpreted as the unknown calibrating parameters lying in the range of a linear operator $\boldsymbol{B}$. E.g., $\boldsymbol{B}$ may be a matrix consisting of the first k columns of the DFT (Discrete Fourier Transform) matrix which represents a scenario where the diagonal entries in $\boldsymbol{D}$ are slowly varying from entry to entry. One may easily formulate the following optimization problem to recover $\boldsymbol{h}$ and $\boldsymbol{x}$,

$$\min_{\boldsymbol{h},\boldsymbol{x}} \|\mathrm{diag}(\boldsymbol{B}\boldsymbol{h})\boldsymbol{A}\boldsymbol{x} - \boldsymbol{y}\|_2. \tag{3.2}$$

We can even consider adding an ℓ_1-component in order to promote sparsity in $\boldsymbol{x}$,

$$\min_{\boldsymbol{h},\boldsymbol{x}} \|\mathrm{diag}(\boldsymbol{B}\boldsymbol{h})\boldsymbol{A}\boldsymbol{x} - \boldsymbol{y}\|_2 + \lambda\|\boldsymbol{x}\|_1, \quad \lambda > 0. \tag{3.3}$$

However, both programs above are nonconvex. For (3.2), it is an optimization with a bilinear objective function because it is linear with respect to one vector when the other is fixed. For (3.3), the objective function is not even bilinear but biconvex [20].

3.1   SparseLift

To combat the difficulties arising from the bilinearity of the objective function in (3.2), we apply a novel idea by "lifting" the problem to its feasible set [8, 11] . It leads to a problem of recovering a rank-one matrix $\boldsymbol{h}\boldsymbol{x}^T$ instead of finding two unknown vectors. Denote by $\boldsymbol{a}_i$ the i-th column of $\boldsymbol{A}^T$ and by $\boldsymbol{b}_i$ the i-th column of $\boldsymbol{B}^*$. A little linear algebra yields

$$y_i = (\boldsymbol{B}\boldsymbol{h})_i\boldsymbol{x}^T\boldsymbol{a}_i = \boldsymbol{b}_i^*\boldsymbol{h}\boldsymbol{x}^T\boldsymbol{a}_i. \tag{3.4}$$

Let $\boldsymbol{X} := \boldsymbol{h}\boldsymbol{x}^T$ and define the linear operator $\mathcal{A} : \mathbb{C}^{k \times N} \to \mathbb{C}^L$,

$$\boldsymbol{y} = \mathcal{A}(\boldsymbol{X}) := \{\boldsymbol{b}_i^*\boldsymbol{X}\boldsymbol{a}_i\}_{i=1}^L$$

Now the problem is to find a rank-one matrix $\boldsymbol{X}$ satisfying the *linear* constraint $\mathcal{A}(\boldsymbol{X}) = \mathcal{A}(\boldsymbol{X}_0)$ where $\boldsymbol{X}_0 = \boldsymbol{h}_0\boldsymbol{x}_0^T$ is the true matrix. A popular method is to take advantage of the rank-one property and use nuclear norm minimization.

$$\min \|\boldsymbol{X}\|_*, \quad \text{subject to} \quad \mathcal{A}(\boldsymbol{X}) = \mathcal{A}(\boldsymbol{X}_0)$$

where $\|\boldsymbol{X}\|_*$ denotes the nuclear norm of $\boldsymbol{X}$ which is the sum of singular values.

Before we proceed, we need a bit of preparations. By using the inner product defined on $\mathbb{C}^{k \times N}$ as $\langle \boldsymbol{X}, \boldsymbol{Z} \rangle :=$ $\mathrm{Tr}(\boldsymbol{X}\boldsymbol{Z}^*)$, the adjoint operator $\mathcal{A}^*(\boldsymbol{u}) : \mathbb{C}^L \to \mathbb{C}^{k \times N}$ of $\mathcal{A}(\boldsymbol{X}) = \{\boldsymbol{b}_l^* \boldsymbol{X} \boldsymbol{a}_l\}_{l=1}^L$ and $\mathcal{A}^*\mathcal{A}(\boldsymbol{X})$ have their forms as

$$\mathcal{A}^*(\boldsymbol{u}) := \sum_{l=1}^L u_l \boldsymbol{b}_l \boldsymbol{a}_l^*, \quad \mathcal{A}^*\mathcal{A}(\boldsymbol{X}) = \sum_{l=1}^L \boldsymbol{b}_l \boldsymbol{b}_l^* \boldsymbol{X} \boldsymbol{a}_l \boldsymbol{a}_l^*.$$

We also need a matrix representation $\boldsymbol{\Phi} : L \times kN$ of $\mathcal{A} : \mathcal{A}(\boldsymbol{X}) = \{\boldsymbol{b}_l^* \boldsymbol{X} \boldsymbol{a}_l\}_{l=1}^L$ such that

$$\boldsymbol{\Phi} \,\mathrm{vec}(\boldsymbol{X}) = \mathrm{vec}(\mathcal{A}(\boldsymbol{X})), \tag{3.5}$$

$\mathrm{vec}(\boldsymbol{X})$ reshapes matrix $\boldsymbol{X}$ into a long column vector in lexicographic order. For each l, $\boldsymbol{b}_l^* \boldsymbol{X} \boldsymbol{a}_l = (\boldsymbol{a}_l \otimes \bar{\boldsymbol{b}}_l)^T \mathrm{vec}(\boldsymbol{X})$ follows from the property of Kronecker product, where $\bar{\boldsymbol{b}}_l$ is the complex conjugate of $\boldsymbol{b}_l$. It naturally leads to the block form of $\boldsymbol{\Phi}^*$,

$$\boldsymbol{\Phi}^* = [\boldsymbol{\phi}_1, \cdots, \boldsymbol{\phi}_l, \cdots, \boldsymbol{\phi}_L]_{l=1}^L, \quad \boldsymbol{\phi}_l = \bar{\boldsymbol{a}}_l \otimes \boldsymbol{b}_l : kN \times 1. \tag{3.6}$$

With all those notations in place, the following formula holds

$$\mathrm{vec}(\mathcal{A}^*\mathcal{A}(\boldsymbol{X})) = \boldsymbol{\Phi}^* \boldsymbol{\Phi} \,\mathrm{vec}(\boldsymbol{X}) = \sum_{l=1}^L \boldsymbol{\phi}_l \boldsymbol{\phi}_l^* \mathrm{vec}(\boldsymbol{X}) = \sum_{l=1}^L [(\bar{\boldsymbol{a}}_l \bar{\boldsymbol{a}}_l^*) \otimes (\boldsymbol{b}_l \boldsymbol{b}_l^*)] \,\mathrm{vec}(\boldsymbol{X}).$$

$\boldsymbol{\Phi}$ also has an explicit representation of its columns. Denote

$$\widetilde{\boldsymbol{\phi}}_{i,j} := \mathrm{diag}(\widetilde{\boldsymbol{b}}_i)\widetilde{\boldsymbol{a}}_j, \quad i = 1, 2, \cdots, k \text{ and } j = 1, 2, \cdots, N. \tag{3.7}$$

$\widetilde{\boldsymbol{b}}_i$ and $\widetilde{\boldsymbol{a}}_j$ are the i-th and j-th column of $\boldsymbol{B}$ and $\boldsymbol{A}$ respectively. Both $\widetilde{\boldsymbol{b}}_i$ and $\widetilde{\boldsymbol{a}}_j$ are of the size $L \times 1$. Then we can write $\boldsymbol{\Phi}$ into

$$\boldsymbol{\Phi} = \left[\widetilde{\boldsymbol{\phi}}_{1,1}, \cdots, \widetilde{\boldsymbol{\phi}}_{k,1}, \cdots, \widetilde{\boldsymbol{\phi}}_{1,N}, \cdots, \widetilde{\boldsymbol{\phi}}_{k,N}\right]_{kN \times L}.$$

Assume we pick a subset $\Gamma_p \subset \{1, 2, \cdots, L\}$ with $|\Gamma_p| = Q$, we define $\mathcal{A}_p(\boldsymbol{X}) := \{\boldsymbol{b}_l^* \boldsymbol{X} \boldsymbol{a}_l\}_{l \in \Gamma_p}$ from $\mathbb{C}^{k \times N} \to \mathbb{C}^Q$ and $\mathcal{A}_p^* \mathcal{A}_p(\boldsymbol{X}) := \sum_{l \in \Gamma_p} \boldsymbol{b}_l \boldsymbol{b}_l^* \boldsymbol{X} \boldsymbol{a}_l \boldsymbol{a}_l^*$. $\mathcal{A}_p$ also has its matrix representation $\boldsymbol{\Phi}_p : Q \times kN$ as

$$\boldsymbol{\Phi}_p^* = [\boldsymbol{\phi}_1, \cdots, \boldsymbol{\phi}_l, \cdots, \boldsymbol{\phi}_L]_{l \in \Gamma_p}, \tag{3.8}$$

such that $\mathrm{vec}(\mathcal{A}_p(\boldsymbol{X})) = \boldsymbol{\Phi}_p \,\mathrm{vec}(\boldsymbol{X})$ and $\mathrm{vec}(\mathcal{A}_p^*\mathcal{A}_p(\boldsymbol{X})) = \boldsymbol{\Phi}_p^* \boldsymbol{\Phi}_p \,\mathrm{vec}(\boldsymbol{X})$.

Concerning the support of $\boldsymbol{x}_0$, $\boldsymbol{X}_0$ and $\mathrm{vec}(\boldsymbol{X}_0)$, we can assume without loss of generality that the support of $\boldsymbol{x}_0$ is its first n entries and thus $\boldsymbol{X}_0 := \boldsymbol{h}_0 \boldsymbol{x}_0^T$ have its first n columns nonzero. By a slight abuse of notation, we write Ω for the supports of $\boldsymbol{x}_0$, $\boldsymbol{X}_0$ and $\mathrm{vec}(\boldsymbol{X}_0)$. Let $\boldsymbol{x}_\Omega$ and $\boldsymbol{X}_\Omega$ denote the orthogonal projections of $\boldsymbol{x}$ and $\boldsymbol{X}$ on Ω, respectively. Define $\mathcal{A}_\Omega$ and $\mathcal{A}_\Omega^*$ as the restriction of $\mathcal{A}$ and $\mathcal{A}^*$ onto Ω,

$$\mathcal{A}_\Omega(\boldsymbol{X}) = \{\boldsymbol{b}_l^* \boldsymbol{X}_\Omega \boldsymbol{a}_l\}_{l=1}^L = \{\boldsymbol{b}_l^* \boldsymbol{X}_\Omega \boldsymbol{a}_{l,\Omega}\}_{l=1}^L, \quad \mathcal{A}_\Omega^*(\boldsymbol{u}) = \sum_{l=1}^L u_l \boldsymbol{b}_l \boldsymbol{a}_{l,\Omega}^* \tag{3.9}$$

One can naturally define $\boldsymbol{\Phi}_\Omega : L \times kN$ with non-zero columns $\{\widetilde{\boldsymbol{\phi}}_{i,j}\}_{1 \le i \le k, 1 \le j \le n}$ satisfying $\boldsymbol{\Phi}_\Omega \,\mathrm{vec}(\boldsymbol{X}) = \mathcal{A}_\Omega(\boldsymbol{X})$ and $\boldsymbol{\Phi}_{\Omega^\perp} : L \times kN$ with non-zero columns $\{\widetilde{\boldsymbol{\phi}}_{i,j}\}_{1 \le i \le k, j > n}$ such that $\boldsymbol{\Phi}_{\Omega^\perp}^* \boldsymbol{u} = \mathrm{vec}(\sum_{l=1}^L u_l \boldsymbol{b}_l \boldsymbol{a}_{l,\Omega^\perp}^*)$.

Even if we stipulate that the degrees of freedom in $\boldsymbol{D}$ is smaller than L by assuming that the diagonal entries of $\boldsymbol{D}$ belong to some subspace, without any further knowledge about $\boldsymbol{x}$ the self-calibration problem is not solvable, since the number of unknowns exceeds the number of measurements. Fortunately, in many cases we can assume that $\boldsymbol{x}$ is sparse. Therefore, the matrix $\boldsymbol{X} = \boldsymbol{h}\boldsymbol{x}^T$ is not only of rank one but also sparse. A common method to recover $\boldsymbol{X}$ from $\boldsymbol{y}$ is to use a linear combination of $\|\cdot\|_*$ and $\|\cdot\|_1$ as an objective function (See [26] for an example in phase retrieval) and solve the following convex program

$$\min \|\boldsymbol{X}\|_* + \lambda\|\boldsymbol{X}\|_1, \lambda > 0, \quad \text{subject to} \quad \mathcal{A}(\boldsymbol{X}) = \mathcal{A}(\boldsymbol{X}_0). \tag{3.10}$$

(3.10) tries to impose both sparsity and rank-one property of a matrix. While (3.10) seems a natural way to proceed, we will choose a somewhat different optimization problem for two reasons. First, it may not be straightforward to find a proper value for λ, see also Figures 2 and 3 in Section 6. Second, in [28] it is shown that if we perform convex optimization by combining the two norms as in (3.10), then we can do no better, order-wise, than an algorithm that exploits only one of the two structures. Since for any matrix $\boldsymbol{Z}$ there holds $\|\boldsymbol{Z}\|_1 \geq \|\boldsymbol{Z}\|_*$, this suggests that it may suffice to simply try to minimize $\|\boldsymbol{X}\|_1$, which in turn may already yield a solution with small nuclear norm. Indeed, as well will demonstrate, ℓ_1-minimization performs well enough to recover $\boldsymbol{X}_0$ exactly under certain conditions although it seemingly fails to take advantage of the rank-one property. Hence we consider

$$\min \|\boldsymbol{X}\|_1, \quad \text{subject to} \quad \mathcal{A}(\boldsymbol{X}) = \mathcal{A}(\boldsymbol{X}_0). \tag{3.11}$$

This idea first lifts the recovery problem of two unknown vectors to a matrix-valued problem and then exploits sparsity through ℓ_1-minimization[1]. We will refer to this method as *SparseLift*. It takes advantage of the "linearization" via lifting, while maintaining the efficiency of compressive sensing and linear programming.

If the observed data are noisy, i.e., $\boldsymbol{y} = \mathcal{A}(\boldsymbol{X}_0) + \boldsymbol{w}$ with $\|\boldsymbol{w}\| < \eta$, we take the following alternative convex program

$$\min \|\boldsymbol{X}\|_1, \quad \text{subject to} \quad \|\mathcal{A}(\boldsymbol{X}) - \mathcal{A}(\boldsymbol{X}_0)\| < \eta. \tag{3.12}$$

It is useful to rewrite (3.11) in matrix-vector form. Noticing that $\mathcal{A}(\boldsymbol{V}) = \boldsymbol{\Phi}\boldsymbol{v}$ for $\boldsymbol{v} = \text{vec}(\boldsymbol{V})$, $\boldsymbol{V} \in \mathbb{C}^{k \times N}$ and $\boldsymbol{v} \in \mathbb{C}^{kN \times 1}$, we have the following equivalent formulation of (3.11)

$$\min \|\boldsymbol{v}\|_1, \quad \text{subject to} \quad \boldsymbol{\Phi}\boldsymbol{v} = \boldsymbol{\Phi}\boldsymbol{v}_0, \tag{3.13}$$

where $\boldsymbol{v}_0 := \text{vec}(\boldsymbol{X}_0)$. The matrix form of (3.12) is obtained similarly,

$$\min \|\boldsymbol{v}\|_1, \quad \text{subject to} \quad \|\boldsymbol{\Phi}\boldsymbol{v} - \boldsymbol{y}\| \leq \eta. \tag{3.14}$$

This model attempts to recover $\boldsymbol{X}_0$ from noisy observation $\boldsymbol{y}$, where $\boldsymbol{y} = \boldsymbol{\Phi}\boldsymbol{v}_0 + \boldsymbol{w}$ and $\|\boldsymbol{w}\| \leq \eta$.

3.2 Main results

Our main finding is that SparseLift enables self-calibration if we are willing to take a few more additional measurements. Furthermore, SparseLift is robust in presence of additive noise. In our theory we consider two choices for the matrix $\boldsymbol{A}$: in the one case $\boldsymbol{A}$ is a Gaussian random matrix; in the other case $\boldsymbol{A}$ consists of rows chosen uniformly at random with replacement from the $N \times N$ Discrete Fourier Transform (DFT) matrix $\boldsymbol{F}$ with $\boldsymbol{F}^*\boldsymbol{F} = \boldsymbol{F}\boldsymbol{F}^* = N\boldsymbol{I}_N$. In the latter case, we simply refer to $\boldsymbol{A}$ as a random Fourier matrix.

---

[1]Alternatively we could use a mixed ℓ_2-ℓ_1-norm here to enforce column-sparsity in $\boldsymbol{X}$, i.e., replace $\|\boldsymbol{X}\|_1$ by $\|X\|_{2,1}$. We leave a theoretical analysis of this modification for future research

Theorem 3.1. *Consider the model* (3.1) *and assume that $\boldsymbol{B}$ is an $L \times k$ tall matrix with $\boldsymbol{B}^*\boldsymbol{B} = \boldsymbol{I}_k$, $\boldsymbol{h}$ is a $k \times 1$ dense vector and $\boldsymbol{x}$ is an n-sparse and $N \times 1$ vector. Then the solution $\hat{\boldsymbol{X}}$ to (3.11) is equal to $\boldsymbol{h}_0 \boldsymbol{x}_0^T$ with probability at least $1 - O(L^{-\alpha+1})$ if*

1. *$\boldsymbol{A} \in \mathbb{R}^{L \times N}$ is a real Gaussian random matrix with $L < N$ with each entry $A_{ij} \overset{iid}{\sim} \mathcal{N}(0,1)$ and*

$$\frac{L}{\log^2 L} \geq C_\alpha \mu_{\max}^2 kn \log(kN). \tag{3.15}$$

2. *$\boldsymbol{A}$ is an $L \times N$ random Fourier matrix with $L < N$ and*

$$L = PQ, \quad P \geq \frac{\log(4\sqrt{2kn}\gamma)}{\log 2}, \quad Q \geq C_\alpha \mu_{\max}^2 kn(\log(L) + \log(kN)) \tag{3.16}$$

 where $\gamma = \sqrt{2N(\log(2kN)+1)+1}$.

In both cases $C_\alpha > 0$ is growing linearly with α and $\mu_{\max}$ is defined as the largest absolute value in the matrix $\sqrt{L}\boldsymbol{B}$, i.e.,

$$\mu_{\max} := \max_{ij} \sqrt{L}|B_{ij}|.$$

Remarks: We note that $\max_{1 \leq l \leq L} \|\boldsymbol{b}_l\|^2 \leq \frac{k}{L}\mu_{\max}^2$, $|B_{ij}| \leq \frac{\mu_{\max}}{\sqrt{L}}$ and $\mu_{\max} \geq 1$. The definition of $\mu_{\max}$ seems strict but it contains a lot of useful and common examples. For example, a matrix $\boldsymbol{B}$ has $\mu_{\max} = 1$ if each column is picked from an $L \times L$ unitary DFT matrix and $\boldsymbol{B}^*\boldsymbol{B} = \boldsymbol{I}_k$. We expect the magnitude of each entry in $\boldsymbol{B}$ does not vary too much from entry to entry and thus the rows of $\boldsymbol{B}$ are incoherent.

Theorem 3.2. *If the observed $\boldsymbol{y}$ is contaminated by noise, namely, $\boldsymbol{y} = \mathcal{A}(\boldsymbol{X}_0) + \boldsymbol{w}$ with $\|\boldsymbol{w}\| \leq \eta$, the solution $\hat{\boldsymbol{X}}$ given by (3.12) has the following error bound.*

1. *If $\boldsymbol{A}$ is a Gaussian random matrix,*

$$\|\hat{\boldsymbol{X}} - \boldsymbol{X}_0\|_F \leq (C_0 + C_1\sqrt{kn})\eta$$

 with probability of success at least $1 - O(L^{-\alpha+1})$. L satisfies (3.15) and both C_0 and C_1 are both constants.

2. *If $\boldsymbol{A}$ is a random Fourier matrix,*

$$\|\hat{\boldsymbol{X}} - \boldsymbol{X}_0\|_F \leq (C_0 + C_1\sqrt{P}\sqrt{kn})\eta$$

 with probability of success at least $1 - O(L^{-\alpha+1})$. L and P satisfy (3.16) and both C_0 and C_1 are both constants

The solution $\hat{\boldsymbol{X}}$ obtained by SparseLift, see (3.11) or (3.12) for noisy data, is not necessarily rank-one. Therefore, we compute the singular value decomposition of $\hat{\boldsymbol{X}}$ and set $\hat{\boldsymbol{h}} = \sqrt{\hat{\sigma}_1}\hat{\boldsymbol{u}}_1$ and $\hat{\boldsymbol{x}} = \sqrt{\hat{\sigma}_1}\hat{\boldsymbol{v}}_1$, where $\hat{\sigma}_1$ is the leading singular value and $\hat{\boldsymbol{u}}_1$ and $\hat{\boldsymbol{v}}_1$ are the leading left and right singular vectors, respectively.

Corollary 3.3. *Let $\hat{\sigma}_1\hat{\boldsymbol{u}}_1\hat{\boldsymbol{v}}_1^T$ be the best Frobenius norm approximation to $\hat{\boldsymbol{X}}$, then there exists a scalar α_0 and a constant c_0 such that*

$$\|\boldsymbol{h}_0 - \alpha_0\hat{\boldsymbol{h}}\| \leq c_0 \min(\epsilon/\|\boldsymbol{h}_0\|, \|\boldsymbol{h}_0\|), \quad \|\boldsymbol{x}_0 - \alpha_0^{-1}\hat{\boldsymbol{x}}\| \leq c_0 \min(\epsilon/\|\boldsymbol{x}_0\|, \|\boldsymbol{x}_0\|)$$

where $\epsilon = \|\hat{\boldsymbol{X}} - \boldsymbol{X}_0\|_F$.

[1, 11, 9] also derived error bounds of recovering the signals from the recovered "lifted" matrix by using the leading eigenvector or leading left and right singular vectors. [11, 9] focus on PhaseLift and the recovered vector from $\hat{\boldsymbol{X}}$ is unique up to a unit-normed scalar. Our corollary is more similar to Corollary 1 in [1] because both of them deal with a bilinear system instead of a quadratic system like PhaseLift. There are two signals to recover from $\hat{\boldsymbol{X}}$ in our case and thus they are unique up to a scalar α_0. Without prior information, no theoretical estimate can be given on how large α_0 is.

The proof of Corollary 3.3 consists of considering the leading eigenvalue and eigenvector of the Hermitian matrices $\hat{\boldsymbol{X}}^* \hat{\boldsymbol{X}}$ and $\hat{\boldsymbol{X}} \hat{\boldsymbol{X}}^*$ and then using the sin-θ Theorem in [15]. The same idea is implemented in the proof of Theorem 1.2 from [11] where the detailed reasoning can be found.

4 Proof of Theorem 3.1

The outline of the proof of Theorem 3.1 follows a meanwhile well-established pattern in the literature of compressive sensing and low-rank matrix recovery, but as is so often the case, the difficulty lies in the technical details. We first derive sufficient conditions for exact recovery. The fulfillment of these conditions involves computing norms of certain random matrices as well as the construction of a dual certificate. The main part of the proof is dedicated to these two steps. While the overall path of the proof is similar for the case when $\boldsymbol{A}$ is a Gaussian random matrix or a random Fourier matrix, there are several individual steps that are quite different.

4.1 A sufficient condition for exact recovery

We will first give a sufficient condition for exact recovery by using SparseLift. Then we prove all the proposed assumptions are valid.

Proposition 4.1. *Let $\boldsymbol{X}_0 = \boldsymbol{h}_0 \boldsymbol{x}_0^T$ with support Ω is the unique minimizer to SparseLift (3.11) if there exists a $\boldsymbol{Y} \in Range(\mathcal{A}^*)$ satisfying*

$$\langle \operatorname{sgn}(\boldsymbol{X}_0) - \boldsymbol{Y}, \boldsymbol{H} \rangle + \|\boldsymbol{H}_{\Omega^\perp}\|_1 > 0 \tag{4.1}$$

for any $\boldsymbol{H} : \mathcal{A}(\boldsymbol{H}) = \boldsymbol{0}$.

This proposition can be regarded as a sufficient condition of exact recovery of a sparse vector via ℓ_1 minimization in compressive sensing (See Theorem 4.26 in [17]). Based on (4.1), we propose an alternative sufficient condition which is much easier to use.

Proposition 4.2. *$\boldsymbol{X}_0$ is the unique minimizer if there exists a $\boldsymbol{Y} \in Range(\mathcal{A}^*)$ satisfying the following conditions,*

$$\|\mathcal{A}(\boldsymbol{H}_\Omega)\|_F \geq \frac{1}{\sqrt{2}} \|\boldsymbol{H}_\Omega\|_F, \qquad \|\mathcal{A}(\boldsymbol{H}_{\Omega^\perp})\|_F \leq \gamma \|\boldsymbol{H}_{\Omega^\perp}\|_F \tag{4.2}$$

$$\| \operatorname{sgn}(\boldsymbol{X}_0) - \boldsymbol{Y}_\Omega\|_F \leq \frac{1}{4\sqrt{2}\gamma}, \qquad \|\boldsymbol{Y}_{\Omega^\perp}\|_\infty \leq \frac{1}{2} \tag{4.3}$$

for any $\boldsymbol{H} : \mathcal{A}(\boldsymbol{H}) = \boldsymbol{0}$.

Remark: We will later specify γ in (4.20) and (4.21).

Proof: The proof starts with (4.1). By splitting the inner product of $\langle \text{sgn}(\boldsymbol{X}_0) - \boldsymbol{Y}, \boldsymbol{H} \rangle$ on Ω and $\Omega^\perp$, the following expression can guarantee the exact recovery of l_1 minimization,

$$\langle \text{sgn}(\boldsymbol{X}_0) - \boldsymbol{Y}_\Omega, \boldsymbol{H}_\Omega \rangle - \langle \boldsymbol{Y}_{\Omega^\perp}, \boldsymbol{H}_{\Omega^\perp} \rangle + \|\boldsymbol{H}_{\Omega^\perp}\|_1 > 0.$$

By using the Hölder inequality, we arrive at the stronger condition

$$-\|\text{sgn}(\boldsymbol{X}_0) - \boldsymbol{Y}_\Omega\|_F \|\boldsymbol{H}_\Omega\|_F + (1 - \|\boldsymbol{Y}_{\Omega^\perp}\|_\infty)\|\boldsymbol{H}_{\Omega^\perp}\|_1 > 0. \tag{4.4}$$

Applying $\|\mathcal{A}(\boldsymbol{H}_\Omega)\|_F \geq \frac{1}{\sqrt{2}}\|\boldsymbol{H}_\Omega\|_F$ and $\|\mathcal{A}(\boldsymbol{H}_{\Omega^\perp})\|_F \leq \gamma\|\boldsymbol{H}_{\Omega^\perp}\|_F$, we have

$$\frac{1}{\sqrt{2}}\|\boldsymbol{H}_\Omega\|_F \leq \|\mathcal{A}(\boldsymbol{H}_\Omega)\|_F = \|\mathcal{A}(\boldsymbol{H}_{\Omega^\perp})\|_F \leq \gamma\|\boldsymbol{H}_{\Omega^\perp}\|_F \leq \gamma\|\boldsymbol{H}_{\Omega^\perp}\|_1. \tag{4.5}$$

The second equality follows from $\mathcal{A}(\boldsymbol{H}_\Omega) + \mathcal{A}(\boldsymbol{H}_{\Omega^\perp}) = \boldsymbol{0}$. Substitute (4.5) into (4.4) and we only need to prove the following expression in order to achieve exact recovery,

$$\left(1 - \|\boldsymbol{Y}_{\Omega^\perp}\|_\infty - \sqrt{2}\gamma\|\text{sgn}(\boldsymbol{X}_0) - \boldsymbol{Y}_\Omega\|_F\right)\|\boldsymbol{H}_{\Omega^\perp}\|_1 > 0.$$

If $\|\text{sgn}(\boldsymbol{X}_0) - \boldsymbol{Y}_\Omega\|_F \leq \frac{1}{4\sqrt{2}\gamma}$, $\|\boldsymbol{Y}_{\Omega^\perp}\|_\infty \leq \frac{1}{2}$ and $\boldsymbol{H}_{\Omega^\perp} \neq 0$, the above expression is strictly positive. On the other hand, $\boldsymbol{H}_{\Omega^\perp} = 0$ implies $\boldsymbol{H} = 0$ from (4.5). Therefore, the solution $\hat{\boldsymbol{X}}$ of SparseLift equals $\boldsymbol{X}_0$ if the proposed conditions (4.2) and (4.3) are satisfied.

∎

4.2 Weak restricted isometry property

In this section, we will prove Lemma 4.3, the weak version of the restricted isometry property, which achieves the first condition in (4.2). With Lemma 4.3, one can easily derive

$$\left|\langle \mathcal{A}(\boldsymbol{H}_\Omega), \mathcal{A}(\boldsymbol{H}_\Omega) \rangle - \|\boldsymbol{H}_\Omega\|_F^2\right| = \|\langle \mathcal{A}_\Omega^* \mathcal{A}_\Omega(\boldsymbol{H}_\Omega) - \boldsymbol{H}_\Omega, \boldsymbol{H}_\Omega \rangle\| \leq \delta\|\boldsymbol{H}_\Omega\|_F^2.$$

$\frac{1}{\sqrt{2}}\|\boldsymbol{H}_\Omega\|_F \leq \|\mathcal{A}(\boldsymbol{H}_\Omega)\|_F \leq \sqrt{\frac{3}{2}}\|\boldsymbol{H}_\Omega\|_F$ follows from the result above by choosing $\delta = \frac{1}{2}$.

Lemma 4.3. *For linear operators $\mathcal{A}$ and $\boldsymbol{\Phi}$ defined in (3.4) and (3.5) respectively,*

$$\|\boldsymbol{\Phi}_\Omega^* \boldsymbol{\Phi}_\Omega - \boldsymbol{I}_\Omega\| = \|\mathcal{A}_\Omega^* \mathcal{A}_\Omega - \boldsymbol{I}_\Omega\| \leq \delta$$

holds true with probability at least $1 - L^{-\alpha+1}$ where $\boldsymbol{I}_\Omega$ is the identity operator on Ω if

1. *$\boldsymbol{a}_l$ is the l-th row of an $L \times N$ Gaussian random matrix $\boldsymbol{A}$ and $L \geq C_\alpha \mu_{\max}^2 kn \max\{\log L/\delta^2, \log^2 L/\delta\}$.*

2. *$\boldsymbol{a}_l$ is the l-th row of an $L \times N$ random Fourier matrix $\boldsymbol{A}$ and $L \geq C_\alpha \mu_{\max}^2 kn \log L/\delta^2$.*

In both cases, C_α grows linearly with respect to α.

In fact, it suffices to prove Lemma 4.6 and Lemma 4.7 because Lemma 4.3 is a special case of them when $Q = L, P = 1$.

4.2.1 Two ingredients of the proof

Before we move to the proof of Lemma 4.6 and Lemma 4.7, we introduce two important ingredients which will be used later. One ingredient concerns the rows $\{b_l\}_{l=1}^{L}$ of B. There exists a disjoint partition of index set $\{1, 2, \cdots, L\}$ into $\{\Gamma_p\}_{p=1}^{P}$ with each $|\Gamma_p| = Q$ and $Q > C\mu_{\max}^2 k \log(L)$ such that

$$\max_{1 \leq p \leq P} \left\| T_p - \frac{Q}{L} I_k \right\| < \frac{Q}{4L}, \quad \text{where } T_p = \sum_{l \in \Gamma_p} b_l b_l^* \tag{4.6}$$

The existence of such a partition follows from the proof of Theorem 1.2 in [10] and is used in Section 3.2 of [1]. (4.6) also can give a simple bound for the operator norm of both T_p and T_p^{-1}.

$$\max_{1 \leq p \leq P} \|T_p\| \leq \frac{5Q}{4L}, \qquad \max_{1 \leq p \leq P} \|T_p^{-1}\| \leq \frac{4L}{3Q} \tag{4.7}$$

The other ingredient concerns the non-commutative Bernstein inequality. For Gaussian and Fourier cases, the inequalities are slightly different. This following theorem, mostly due to [31], is used for proving the Fourier case of (4.19).

Theorem 4.4. *Consider a finite sequence of $\mathcal{Z}_l$ of independent centered random matrices with dimension $M \times M$. Assume that $\|\mathcal{Z}_l\| \leq R$ and introduce the random matrix*

$$S = \sum_{l \in \Gamma_p} \mathcal{Z}_l. \tag{4.8}$$

Compute the variance parameter

$$\sigma^2 = \max \left\{ \left\| \sum_{l=1}^{Q} \mathrm{E}(\mathcal{Z}\mathcal{Z}^*) \right\|, \left\| \sum_{l=1}^{Q} \mathrm{E}(\mathcal{Z}^*\mathcal{Z}) \right\| \right\}, \tag{4.9}$$

then for all $t \geq 0$

$$\Pr(\|S\| \geq t) \leq 2M \exp\left(-\frac{t^2/2}{\sigma^2 + Rt/3} \right).$$

For the Gaussian case we replace $\|\mathcal{Z}_l\| \leq R$ by $\|\mathcal{Z}_l\|_{\psi_1} \leq R$ where the norm $\|\cdot\|_{\psi_1}$ of a matrix is defined as

$$\|Z\|_{\psi_1} := \inf_{u \geq 0} \{\mathrm{E}[\exp(\|Z\|/u)] \leq 2\}.$$

The concentration inequality is different if a_l is Gaussian because of the unboundedness of S. We are using the form in [23].

Theorem 4.5. *For a finite sequence of independent $M \times M$ random matrices $\mathcal{Z}_l$ with $R := \max_{1 \leq l \leq Q} \|\mathcal{Z}_l\|_{\psi_1}$ and S and σ^2 as defined in (4.8) and (4.9), we have the tail bound on the operator norm of S,*

$$\Pr(\|S\| \geq t) \leq 2M \exp\left(-\frac{1}{C_0} \frac{t^2}{\sigma^2 + \log\left(\frac{\sqrt{Q}R}{\sigma^2}\right) Rt} \right), \tag{4.10}$$

where C_0 is an absolute constant.

12

4.2.2 A key lemma and its proof

For each subset of indices Γ_p, we denote

$$\mathcal{A}_{p,\Omega}(\boldsymbol{X}) = \{\boldsymbol{b}_l^* \boldsymbol{X} \boldsymbol{a}_{l,\Omega}\}_{l \in \Gamma_p}, \quad \mathcal{A}_{p,\Omega}^*(\boldsymbol{u}) = \sum_{l \in \Gamma_p} u_l \boldsymbol{b}_l \boldsymbol{a}_{l,\Omega}^*, \quad \mathcal{A}_{p,\Omega}^* \mathcal{A}_{p,\Omega}(\boldsymbol{X}) = \sum_{l \in \Gamma_p} \boldsymbol{b}_l \boldsymbol{b}_l^* \boldsymbol{X} \boldsymbol{a}_{l,\Omega} \boldsymbol{a}_{l,\Omega}^* \tag{4.11}$$

Under certain conditions, it is proven that $\mathcal{A}_{p,\Omega}^* \mathcal{A}_{p,\Omega}$ does not deviate from the identity much for each p. We write this result into Lemma 4.6.

Lemma 4.6. *For any equal partition $\{\Gamma_p\}_{p=1}^P$ of $\{1, 2, \cdots, L\}$ with $|\Gamma_p| = Q$ such that (4.6) holds true, we have*

$$\max_{1 \le p \le P} \sup_{\|\boldsymbol{X}_\Omega\|_F = 1} \left\| \mathcal{A}_{p,\Omega}^* \mathcal{A}_{p,\Omega}(\boldsymbol{X}) - \frac{Q}{L} \boldsymbol{X}_\Omega \right\|_F \le \frac{Q}{2L}, \tag{4.12}$$

with probability at least $1 - PL^{-\alpha} \ge 1 - L^{-\alpha+1}$ if we choose $Q \ge C_\alpha kn \log(L)$ where C_α grows linearly with respect to α and

1. $\boldsymbol{a}_l$ *is the l-th row of an $L \times N$ Gaussian random matrix $\boldsymbol{A}$ and $Q \ge C_\alpha \mu_{\max}^2 kn \log^2 L$.*

2. $\boldsymbol{a}_l$ *is the l-th row of an $L \times N$ random Fourier matrix $\boldsymbol{A}$ and $Q \ge C_\alpha \mu_{\max}^2 kn \log L$.*

First, we prove the following key lemma. Lemma 4.6 is a direct result from this key lemma and from (4.6). We postpone the proof of Lemma 4.6 to Section 4.2.3.

Lemma 4.7. *For any fixed $0 < \delta \le 1$ and equal partition $\{\Gamma_p\}_{p=1}^P$ of $\{1, 2, \cdots, L\}$ with $|\Gamma_p| = Q$, we have*

$$\max_{1 \le p \le P} \sup_{\|\boldsymbol{X}_\Omega\|_F = 1} \left\| \mathcal{A}_{p,\Omega}^* \mathcal{A}_{p,\Omega}(\boldsymbol{X}) - \boldsymbol{T}_p \boldsymbol{X}_\Omega \right\|_F \le \frac{\delta Q}{L}, \quad \boldsymbol{T}_p = \sum_{l \in \Gamma_p} \boldsymbol{b}_l \boldsymbol{b}_l^*, \tag{4.13}$$

with probability at least $1 - PL^{-\alpha} \ge 1 - L^{-\alpha+1}$ if

1. $\boldsymbol{a}_l$ *is the l-th row of an $L \times N$ Gaussian random matrix $\boldsymbol{A}$ and $Q \ge C_\alpha \mu_{\max}^2 kn \max\{\log L/\delta^2, \log^2 L/\delta\}$.*

2. $\boldsymbol{a}_l$ *is the l-th row of an $L \times N$ random Fourier matrix $\boldsymbol{A}$ and $Q \ge C_\alpha \mu_{\max}^2 kn \log L/\delta^2$.*

In both cases C_α grows linearly with respect to α.

Proof: We prove the Gaussian case first; the proof of the Fourier case is similar. Let $\boldsymbol{a}_l$ be a standard Gaussian random vector. $\boldsymbol{a}_{l,\Omega}$ denotes the projected vector $\boldsymbol{a}_l$ on the support of $\boldsymbol{x}_0$ with $|\operatorname{supp}(\boldsymbol{x}_0)| = n$. We first rewrite the linear operator $\mathcal{A}_{p,\Omega}^* \mathcal{A}_{p,\Omega}(\boldsymbol{X}) - \boldsymbol{T}_p \boldsymbol{X}_\Omega$ into matrix form. Using the Kronecker product we define $\mathcal{Z}_l$ as follows:

$$\mathcal{Z}_l := (\boldsymbol{a}_{l,\Omega} \boldsymbol{a}_{l,\Omega}^* - \boldsymbol{I}_{N,\Omega}) \otimes (\boldsymbol{b}_l \boldsymbol{b}_l^*). \tag{4.14}$$

Every $\mathcal{Z}_l$ is a centered random matrix since $\operatorname{E}(\boldsymbol{a}_{l,\Omega} \boldsymbol{a}_{l,\Omega}^*) = \boldsymbol{I}_{N,\Omega}$, which follows from restricting $\boldsymbol{a}_l \boldsymbol{a}_l^*$ to the support Ω and taking its expectation. $\boldsymbol{I}_{N,\Omega}$ is an $N \times N$ matrix with $\boldsymbol{I}_n$ as its $n \times n$ leading submatrix. It is easy to check that

$$\operatorname{vec}(\mathcal{A}_{p,\Omega}^* \mathcal{A}_{p,\Omega}(\boldsymbol{X}) - \boldsymbol{T}_p \boldsymbol{X}_\Omega) = \sum_{l \in \Gamma_p} \mathcal{Z}_l \operatorname{vec}(\boldsymbol{X}_\Omega).$$

More precisely, $\boldsymbol{a}_{l,\Omega}$ in $\mathcal{Z}_l$ needs to be replaced by $\bar{\boldsymbol{a}}_{l,\Omega}$ according to the property of Kronecker product. But there is no problem here since each $\boldsymbol{a}_l$ is a real random vector. For simplicity, we denote $\mathcal{Z}_l(\boldsymbol{X}) := \boldsymbol{b}_l \boldsymbol{b}_l^* \boldsymbol{X} (\boldsymbol{a}_{l,\Omega} \boldsymbol{a}_{l,\Omega}^* - \boldsymbol{I}_{N,\Omega})$. In order to use Theorem 4.5 to estimate $\|\sum_{l=1}^L \mathcal{Z}_l\|$, we need to know $\|\mathcal{Z}_l\|_{\psi_1}$ and σ^2. We first give an estimate for the operator norm of $\mathcal{Z}_l$.

$$\|\mathcal{Z}_l\| = |\boldsymbol{b}_l^* \boldsymbol{b}_l| \|\boldsymbol{a}_{l,\Omega} \boldsymbol{a}_{l,\Omega}^* - \boldsymbol{I}_{N,\Omega}\| \le \frac{\mu_{\max}^2 k}{L} \|\boldsymbol{a}_{l,\Omega} \boldsymbol{a}_{l,\Omega}^* - \boldsymbol{I}_{N,\Omega}\| \le \frac{\mu_{\max}^2 k}{L} \max(\|\boldsymbol{a}_{l,\Omega}\|^2, 1). \tag{4.15}$$

Here $\|\boldsymbol{a}_{l,\Omega}\|^2$ is a χ^2 random variable with n degrees of freedom. This implies that $\|\mathcal{Z}_l\|_{\psi_1} \leq \frac{C\mu_{\max}^2 kn}{L}$, following from Lemma 6 in [1]. Therefore there exists an absolute constant C_b such that

$$R = \max_{1 \leq p \leq P} \|\mathcal{Z}_l\|_{\psi_1} \leq \frac{C_b \mu_{\max}^2 kn}{L}.$$

The second step is to estimate σ^2. It suffices to consider $\mathcal{Z}_l^* \mathcal{Z}_l(\boldsymbol{X})$ because $\mathcal{Z}_l$ is a Hermitian matrix.

$$\mathcal{Z}_l^* \mathcal{Z}_l(\boldsymbol{X}) = \|\boldsymbol{b}_l\|^2 \boldsymbol{b}_l \boldsymbol{b}_l^* \boldsymbol{X} (\boldsymbol{a}_{l,\Omega} \boldsymbol{a}_{l,\Omega}^* - \boldsymbol{I}_{N,\Omega})^2 \leq \frac{\mu_{\max}^2 k}{L} \boldsymbol{b}_l \boldsymbol{b}_l^* \boldsymbol{X} (\boldsymbol{a}_{l,\Omega} \boldsymbol{a}_{l,\Omega}^* - \boldsymbol{I}_{N,\Omega})^2.$$

The expectation of $\mathcal{Z}_l^* \mathcal{Z}_l(\boldsymbol{X})$ is

$$\mathrm{E}(\mathcal{Z}_l^* \mathcal{Z}_l(\boldsymbol{X})) \leq \frac{\mu_{\max}^2 k(n+1)}{L} \boldsymbol{b}_l \boldsymbol{b}_l^* \boldsymbol{X} \boldsymbol{I}_{N,\Omega}, \tag{4.16}$$

following from $\mathrm{E}(\boldsymbol{a}_{l,\Omega} \boldsymbol{a}_{l,\Omega}^* - \boldsymbol{I}_{N,\Omega})^2 = (n+1)\boldsymbol{I}_{N,\Omega}$.

$$\sigma^2 = \left\| \sum_{l \in \Gamma_p} \mathrm{E}(\mathcal{Z}_l^* \mathcal{Z}_l) \right\| \leq \frac{\mu_{\max}^2 k(n+1)}{L} \sup_{\|\boldsymbol{X}\|_F = 1} \left\| \sum_{l \in \Gamma_p} \boldsymbol{b}_l \boldsymbol{b}_l^* \boldsymbol{X} \boldsymbol{I}_{N,\Omega} \right\|_F \tag{4.17}$$

$$\leq \frac{\mu_{\max}^2 k(n+1)}{L} \|\boldsymbol{T}_p\| \leq \frac{5\mu_{\max}^2 Qk(n+1)}{4L^2} \leq \frac{5\mu_{\max}^2 Qkn}{2L^2}. \tag{4.18}$$

Now we have estimates of R and σ^2 and can apply the concentration inequality (4.10). $\mathcal{Z}_l$ can be simply treated as a $kn \times kn$ matrix for each l because its support is only $kn \times kn$. There holds

$$\log\left(\frac{\sqrt{Q}R}{\sigma^2}\right) \leq C_1 \log L$$

for some constant C_1 because Q, R and σ^2 are at most of order L. Hence,

$$\Pr\left(\| \sum_{l \in \Gamma_p} \mathcal{Z}_l \| \geq \frac{\delta Q}{L} \right) \leq 2kn \exp\left(-\frac{Q^2}{C_0 L^2} \cdot \frac{\delta^2}{\frac{5\mu_{\max}^2 Qkn}{2L^2} + C_b \frac{\mu_{\max}^2 kn}{L} \cdot \frac{\delta Q}{L} \log L} \right)$$

$$\leq 2kn \exp\left(-\frac{1}{C_0} \cdot \frac{Q\delta^2}{5\mu_{\max}^2 kn/2 + C_b \delta \mu_{\max}^2 kn \log L} \right).$$

If we let $Q \geq C_0 \mu_{\max}^2 kn (\frac{5}{2} + C_b \delta \log L)(\alpha \log L + \log 2kn)/\delta^2$ and also properly assume $L > kn$, then $Q \geq C_\alpha \mu_{\max}^2 kn \max\{\log L/\delta^2, \log^2 L/\delta\}$ and

$$\Pr\left(\| \sum_{l \in \Gamma_p} \mathcal{Z}_l \| \geq \frac{\delta Q}{L} \right) \leq L^{-\alpha},$$

where C_α is a function which grows linearly with respect to α. Now we take the union bound over all $p = 1, 2, \cdots, P$ and have

$$\Pr\left(\max_{1 \leq p \leq P} \| \sum_{l \in \Gamma_p} \mathcal{Z}_l \| \leq \frac{\delta Q}{L} \right) \geq 1 - PL^{-\alpha} \geq 1 - L^{-\alpha+1}.$$

This completes the proof for the Gaussian case.

We proceed to the Fourier case. The proof is very similar and the key is to use Theorem 4.4. Now we assume $\boldsymbol{a}_l$ is the l-th row of a random Fourier matrix $\boldsymbol{A}$ and thus $\mathrm{E}(\boldsymbol{a}_l\boldsymbol{a}_l^*) = \boldsymbol{I}_N$. Here $\mathcal{Z}_l = (\bar{\boldsymbol{a}}_{l,\Omega}\bar{\boldsymbol{a}}_{l,\Omega}^* - \boldsymbol{I}_{N,\Omega}) \otimes \boldsymbol{b}_l\boldsymbol{b}_l^*$ because $\boldsymbol{a}_l$ is a complex random vector. However, it does not change the proof at all except the notation. Like the proof of the Gaussian case, we need to know $R = \max_{l\in\Gamma_p} \|\mathcal{Z}_l\|$ and also σ^2. Following the same calculations that yielded (4.15), (4.16) and (4.18), we have

$$R \leq |\boldsymbol{b}_l^*\boldsymbol{b}_l| \max(\|\bar{\boldsymbol{a}}_{l,\Omega}\|^2, 1) \leq \frac{\mu_{\max}^2 kn}{L}$$

and

$$\sigma^2 \leq \frac{\mu_{\max}^2 k}{L}\|\sum_{l\in\Gamma_p} \boldsymbol{b}_l\boldsymbol{b}_l^*\boldsymbol{X}\, \mathrm{E}(\bar{\boldsymbol{a}}_{l,\Omega}\bar{\boldsymbol{a}}_{l,\Omega}^* - \boldsymbol{I}_{N,\Omega})^2\| \leq \frac{\mu_{\max}^2 kn}{L}\|\boldsymbol{T}_p\| \leq \frac{5\mu_{\max}^2 Qkn}{4L^2}.$$

Then we apply Theorem 4.4 and obtain

$$\Pr\left(\|\sum_{l\in\Gamma_p} \mathcal{Z}_l\| \geq \frac{\delta Q}{L}\right) \leq 2kn\exp\left(-\frac{Q\delta^2}{5\mu_{\max}^2 kn/2 + 2\delta\mu_{\max}^2 kn/3}\right) \leq L^{-\alpha}$$

if we pick $Q \geq C_\alpha \mu_{\max}^2 kn \log L/\delta^2$. This implies

$$\Pr\left(\max_{1\leq p\leq P}\|\sum_{l\in\Gamma_p} \mathcal{Z}_l\| \leq \frac{\delta Q}{L}\right) \geq 1 - PL^{-\alpha} \geq 1 - L^{-\alpha+1}.$$

∎

4.2.3 Proof of Lemma 4.6

Proof: From Lemma 4.7, we know that for any fixed equal partition $\{\Gamma_p\}_{p=1}^P$ of $\{1, 2, \cdots, L\}$ with $|\Gamma_p| = Q$, we have

$$\max_{1\leq p\leq P}\sup_{\|\boldsymbol{X}_\Omega\|_F=1}\left\|\sum_{l\in\Gamma_p} \boldsymbol{b}_l\boldsymbol{b}_l^*\boldsymbol{X}_\Omega\boldsymbol{a}_{l,\Omega}\boldsymbol{a}_{l,\Omega}^* - \boldsymbol{T}_p\boldsymbol{X}_\Omega\right\|_F \leq \frac{\delta Q}{L} \tag{4.19}$$

In particular, we choose the partition $\{\Gamma_p\}_{p=1}^P$ which satisfies (4.6) and set $\delta = \frac{1}{4}$. In this case,

$$
\begin{aligned}
\sup_{\|\boldsymbol{X}_\Omega\|_F=1}\left\|\sum_{l\in\Gamma_p} \boldsymbol{b}_l\boldsymbol{b}_l^*\boldsymbol{X}_\Omega\boldsymbol{a}_{l,\Omega}\boldsymbol{a}_{l,\Omega}^* - \boldsymbol{I}_p\boldsymbol{X}_\Omega\right\|_F &\leq \sup_{\|\boldsymbol{X}_\Omega\|_F=1}\left\|\sum_{l\in\Gamma_p} \boldsymbol{b}_l\boldsymbol{b}_l^*\boldsymbol{X}_\Omega\boldsymbol{a}_{l,\Omega}\boldsymbol{a}_{l,\Omega}^* - \boldsymbol{T}_p\boldsymbol{X}_\Omega\right\|_F \\
&\quad + \sup_{\|\boldsymbol{X}_\Omega\|_F=1}\|\boldsymbol{T}_p\boldsymbol{X}_\Omega - \boldsymbol{I}_p\boldsymbol{X}_\Omega\|_F \\
&\leq \frac{Q}{4L} + \frac{Q}{4L} = \frac{Q}{2L}.
\end{aligned}
$$

∎

4.2.4 Estimating the operator norm of $\mathcal{A}$

The operator norm γ of $\mathcal{A}$ with $\boldsymbol{A}$ will be given for both the Gaussian case and the Fourier case such that $\|\mathcal{A}(\boldsymbol{X})\|_F \leq \gamma \|\boldsymbol{X}\|_F$ for any $\boldsymbol{X}$ with high probability. For the Gaussian case, Lemma 1 in [1] gives the answer directly.

Lemma 4.8. *For $\mathcal{A}$ defined in (3.4) with $\boldsymbol{A}$ a Gaussian random matrix and fixed $\alpha \geq 1$,*

$$\|\mathcal{A}\| \leq \sqrt{N \log(NL/2) + \alpha \log L} \tag{4.20}$$

with probability at least $1 - L^{-\alpha}$.

For the Fourier case, a direct proof will be given by using the non-commutative Bernstein inequality in Theorem 4.4.

Lemma 4.9. *For $\mathcal{A}$ defined in (3.4) with $\boldsymbol{A}$ a random Fourier matrix and fixed $\alpha \geq 1$, its operator norm γ yields*

$$\gamma \leq \sqrt{2N(\log(2kN) + 1) + 1} \tag{4.21}$$

with probability at least $1 - L^{-\alpha}$ if we choose $L \geq \alpha k \log(L)$.

Proof: First note that $\|\mathcal{A}^*\mathcal{A}\| = \|\mathcal{A}\|^2$. Therefore it suffices to estimate the norm of $\mathcal{A}^*\mathcal{A}$,

$$\mathcal{A}^*\mathcal{A}(\boldsymbol{X}) = \sum_{l=1}^{L} \boldsymbol{b}_l \boldsymbol{b}_l^* \boldsymbol{X} \boldsymbol{a}_l \boldsymbol{a}_l^* = \sum_{l=1}^{L} \boldsymbol{b}_l \boldsymbol{b}_l^* \boldsymbol{X} (\boldsymbol{a}_l \boldsymbol{a}_l^* - \boldsymbol{I}_N) + \boldsymbol{X}.$$

The second equality follows from the fact that $\sum_{l=1}^{L} \boldsymbol{b}_l \boldsymbol{b}_l^* = \boldsymbol{I}_k$. For each centered random matrix $\mathcal{Z}_l := (\bar{\boldsymbol{a}}_l \bar{\boldsymbol{a}}_l^* - \boldsymbol{I}_N) \otimes \boldsymbol{b}_l \boldsymbol{b}_l^*$, we have $\mathrm{vec}(\mathcal{A}^*\mathcal{A}(\boldsymbol{X})) = \sum_{l=1}^{L} \mathcal{Z}_l \mathrm{vec}(\boldsymbol{X}) + \mathrm{vec}(\boldsymbol{X})$ and

$$\|\mathcal{Z}_l\| = \|(\bar{\boldsymbol{a}}_l \bar{\boldsymbol{a}}_l^* - \boldsymbol{I}_N) \otimes \boldsymbol{b}_l \boldsymbol{b}_l^*\| = \|\bar{\boldsymbol{a}}_l \bar{\boldsymbol{a}}_l^* - \boldsymbol{I}_N\| \cdot \|\boldsymbol{b}_l \boldsymbol{b}_l^*\| \leq \frac{kN}{L}.$$

That means $R = \max_{1 \leq l \leq L} \|(\bar{\boldsymbol{a}}_l \bar{\boldsymbol{a}}_l^* - \boldsymbol{I}_N) \otimes \boldsymbol{b}_l \boldsymbol{b}_l^*\| \leq \frac{\mu_{\max}^2 kN}{L}$. In order to estimate the variance of each $\mathcal{Z}_l$, we only have to compute $\mathrm{E}(\mathcal{Z}_l \mathcal{Z}_l^*)$ since $\mathcal{Z}_l$ is a Hermitian matrix.

$$\mathrm{E}(\mathcal{Z}_l \mathcal{Z}_l^*) \leq \frac{k}{L} \mathrm{E}\left[(\bar{\boldsymbol{a}}_l \bar{\boldsymbol{a}}_l^* - \boldsymbol{I}_N)^2 \otimes \boldsymbol{b}_l \boldsymbol{b}_l^*\right] = \frac{\mu_{\max}^2 k(N-1)}{L} \boldsymbol{I}_N \otimes \boldsymbol{b}_l \boldsymbol{b}_l^*.$$

Hence the variance of $\sum_{l=1}^{L} \mathcal{Z}_l$ is

$$\sigma^2 \leq \frac{\mu_{\max}^2 k(N-1)}{L} \left\|\sum_{l=1}^{L} \boldsymbol{I}_N \otimes \boldsymbol{b}_l \boldsymbol{b}_l^*\right\| = \frac{\mu_{\max}^2 k(N-1)}{L} \leq \frac{\mu_{\max}^2 kN}{L}.$$

Now we apply Bernstein's inequality and for any $t \geq 0$, we have

$$
\begin{aligned}
\Pr\left(\|\sum_{l=1}^{L} \mathcal{Z}_l\| \geq t\right) &\geq 2kN \cdot \exp\left(-\frac{t^2/2}{\sigma^2 + Rt/3}\right) \\
&= 2kN \cdot \exp\left(-\frac{t^2/2}{\mu_{\max}^2 kN/L + \mu_{\max}^2 kNt/3L}\right).
\end{aligned}
$$

16

Suppose $t \geq 3$, then

$$\Pr\left(\|\sum_{l=1}^{L} \mathcal{Z}_l\| \geq t\right) \leq 2kN \cdot \exp\left(-\frac{t^2/2}{2\mu_{\max}^2 kNt/3L}\right)$$

$$= 2kN \cdot \exp\left(-\frac{3Lt}{4\mu_{\max}^2 kN}\right).$$

If we pick $t = 2N(\log(2kN) + 1)$, then

$$\Pr\left(\|\sum_{l=1}^{L} \mathcal{Z}_l\| \geq t\right) \leq \exp\left(-\frac{L}{\mu_{\max}^2 k}\right).$$

If $L \geq \alpha\mu_{\max}^2 k \log(L)$, we have

$$\|\mathcal{A}^*\mathcal{A}\| \leq \|\sum_{l=1}^{L} \mathcal{Z}_l\| + 1 \leq 2N(\log(2kN) + 1) + 1 \implies \gamma := \|\mathcal{A}\| \leq \sqrt{2N(\log(2kN) + 1) + 1}$$

with probability at least $1 - L^{-\alpha}$. $\blacksquare$

4.3 Construction of the dual certificate

In the last section we have shown that (4.2) holds true. Now we will prove that a $\boldsymbol{Y} \in \mathrm{Range}(\mathcal{A}^*)$ satisfying (4.3) exists by explicitly constructing it for both, the Gaussian and the Fourier case.

4.3.1 $\boldsymbol{A}$ is a Gaussian random matrix

Exact dual certificate via least square method:
It follows from Lemma 4.3 and 4.6 that $\boldsymbol{\Phi}_\Omega^* \boldsymbol{\Phi}_\Omega$ is invertible on Ω with $\|(\boldsymbol{\Phi}_\Omega^* \boldsymbol{\Phi}_\Omega)^{-1}\| \leq 2$ if $L > C_0 kn \log^2 L$ for some scalar C_0. We construct the exact dual certificate via (see [6])

$$\boldsymbol{p} := \boldsymbol{\Phi}_\Omega(\boldsymbol{\Phi}_\Omega^* \boldsymbol{\Phi}_\Omega)^{-1} \mathrm{vec}(\mathrm{sgn}(\boldsymbol{X}_0)), \quad \boldsymbol{q} := \boldsymbol{\Phi}^* \boldsymbol{p} = \mathrm{vec}(\mathcal{A}^*(\boldsymbol{p})). \tag{4.22}$$

Let $\boldsymbol{Y} = \mathcal{A}^*(\boldsymbol{p})$; it suffices to consider $\boldsymbol{q}$ and verify whether $\boldsymbol{q}_\Omega = \mathrm{vec}(\boldsymbol{X}_0)$ and $\|\boldsymbol{q}_{\Omega^\perp}\|_\infty \leq \frac{1}{2}$.
Theorem 4.10. *Conditioned on the event $\{\|(\boldsymbol{\Phi}_\Omega^* \boldsymbol{\Phi}_\Omega)^{-1}\| \leq 2\}$, there holds*

$$\|\boldsymbol{q}_{\Omega^\perp}\|_\infty \leq \frac{1}{2}$$

with probability at least $1 - L^{-\alpha}$ if we pick $L > 16\alpha kn(\log(L) + \log(kN))$.

Proof: Note $\boldsymbol{\Phi}_{\Omega^\perp}^* \boldsymbol{p} = (\widetilde{\boldsymbol{\phi}}_{s,t}^* \boldsymbol{p})_{1 \leq s \leq k, t > n}$ and $\widetilde{\boldsymbol{\phi}}_{s,t} = \mathrm{diag}(\widetilde{\boldsymbol{b}}_s)\widetilde{\boldsymbol{a}}_t$ is independent of $\boldsymbol{p}$ for $t > n$ because $\boldsymbol{p}$ only uses $\widetilde{\boldsymbol{a}}_j : 1 \leq j \leq n$. In this case,

$$\widetilde{\boldsymbol{\phi}}_{s,t}^* \boldsymbol{p} = \widetilde{\boldsymbol{a}}_t^* \mathrm{diag}(\widetilde{\boldsymbol{b}}_s^*)\boldsymbol{p}$$

where $\widetilde{\boldsymbol{a}}_t$ is an $L \times 1$ standard Gaussian random vector. Due to the independence between $\mathrm{diag}(\widetilde{\boldsymbol{b}}_s^*)\boldsymbol{p}$ and $\boldsymbol{a}_t$,

$$\widetilde{\boldsymbol{\phi}}_{s,t}^* \boldsymbol{p} \sim \mathcal{N}(0, \|\mathrm{diag}(\widetilde{\boldsymbol{b}}_s^*)\boldsymbol{p}\|^2),$$

17

where

$$\| \operatorname{diag}(\widetilde{\boldsymbol{b}}_s^*)\boldsymbol{p}\|^2 \;\leq\; \frac{\mu_{\max}^2}{L}\|\boldsymbol{p}\|^2 \leq \frac{\mu_{\max}^2}{L}\operatorname{vec}(\operatorname{sgn}(\boldsymbol{X}_0))^*(\boldsymbol{\Phi}_\Omega^*\boldsymbol{\Phi}_\Omega)^{-1}\operatorname{vec}(\operatorname{sgn}(\boldsymbol{X}_0))$$

$$\leq\; \frac{2\mu_{\max}^2}{L}\|\operatorname{sgn}(\boldsymbol{X}_0)\|_F^2 = \frac{2\mu_{\max}^2 kn}{L}.$$

Therefore, $\widetilde{\boldsymbol{\phi}}_{s,t}^*\boldsymbol{p}$ is a centered Gaussian random variable with variance σ^2 at most $\frac{2\mu_{\max}^2 kn}{L}$. For any $x > 0$, we have

$$\Pr\left(|\widetilde{\boldsymbol{\phi}}_{s,t}^*\boldsymbol{p}| > x\right) \;=\; \frac{2}{\sqrt{2\pi\sigma^2}}\int_x^\infty e^{-\frac{u^2}{2\sigma^2}}\,\mathrm{d}u \leq \frac{2}{\sqrt{2\pi}\sigma x}\int_x^\infty u e^{-\frac{u^2}{2\sigma^2}}\,\mathrm{d}u$$

$$=\; -\frac{2\sigma}{\sqrt{2\pi}x}e^{-\frac{u^2}{2\sigma^2}}\Big|_x^\infty = \sqrt{\frac{2}{\pi}}\frac{\sigma}{x}e^{-\frac{x^2}{2\sigma^2}}.$$

Taking the union bound over all pairs (s,t) of indices $(\widetilde{\boldsymbol{\phi}}_{s,t}^*)$ satisfying $t > n$ gives

$$\Pr\left(\max_{1\leq s\leq k,t>n}|\widetilde{\boldsymbol{\phi}}_{s,t}^*\boldsymbol{p}| > x\right) \leq kN\sqrt{\frac{2}{\pi}}\frac{\sigma}{x}e^{-\frac{x^2}{2\sigma^2}} \leq \frac{kN\sigma}{x}e^{-\frac{x^2}{2\sigma^2}}.$$

If we pick $x = \frac{1}{2}$ and $\frac{\sigma}{x} \leq 2\sqrt{\frac{2\mu_{\max}^2 kn}{L}}$,

$$\Pr\left(\max_{1\leq s\leq k,t>n}|\widetilde{\boldsymbol{\phi}}_{s,t}^*\boldsymbol{p}| > \frac{1}{2}\right) \leq \frac{kN\sigma}{x}e^{-\frac{x^2}{2\sigma^2}} \leq \frac{2\sqrt{2\mu_{\max}^2 kn}kN}{\sqrt{L}}e^{-\frac{L}{16\mu_{\max}^2 kn}}.$$

Choosing $L > 16\alpha\mu_{\max}^2 nk(\log(L) + \log(kN))$ impies that the probability of $\{\max_{1\leq s\leq k,t>n}|\widetilde{\boldsymbol{\phi}}_{s,t}^*\boldsymbol{p}| > \frac{1}{2}\}$ will be less than $L^{-\alpha}$. $\blacksquare$

4.3.2 $\boldsymbol{A}$ is a random Fourier matrix

An inexact dual certificate $\boldsymbol{Y}$ will be constructed to satisfy the conditions given in (4.3) via the golfing scheme, which was first proposed by David Gross in [21] and has been widely used since then, see e.g. [1, 17, 26]. The idea is to use part of the measurements in each step and construct a sequence of random matrices $\boldsymbol{Y}_p$ recursively in $\operatorname{Range}(\mathcal{A}^*)$. The sequence converges to $\operatorname{sgn}(\boldsymbol{X}_0)$ exponentially fast and at the same time keeps each entry of $\boldsymbol{Y}_p$ small on $\Omega^\perp$. We initialize $\boldsymbol{Y}_0 := \boldsymbol{0}$ and set

$$\boldsymbol{Y}_p := \boldsymbol{Y}_{p-1} + \frac{L}{Q}\mathcal{A}_p^*\mathcal{A}_p(\operatorname{sgn}(\boldsymbol{X}_0) - \boldsymbol{Y}_{p-1,\Omega}). \tag{4.23}$$

We denote $\boldsymbol{Y} := \boldsymbol{Y}_P$. Let $\boldsymbol{W}_p := \boldsymbol{Y}_{p,\Omega} - \operatorname{sgn}(\boldsymbol{X}_0)$ be the residual between $\operatorname{sgn}(\boldsymbol{X}_0)$ and $\boldsymbol{Y}_p$ on Ω and $\boldsymbol{W}_0 := \boldsymbol{Y}_0 - \operatorname{sgn}(\boldsymbol{X}_0) = -\operatorname{sgn}(\boldsymbol{X}_0)$. $\boldsymbol{W}_p$ yields the following relations due to (4.23),

$$\boldsymbol{W}_p \;=\; \boldsymbol{W}_{p-1} - \frac{L}{Q}\mathcal{A}_{p,\Omega}^*\mathcal{A}_{p,\Omega}(\boldsymbol{W}_{p-1})$$

$$=\; \frac{L}{Q}\left(\frac{Q}{L} - \mathcal{A}_{p,\Omega}^*\mathcal{A}_{p,\Omega}\right)(\boldsymbol{W}_{p-1}).$$

where $\mathcal{A}_{p,\Omega}$ is defined in (4.11). Following from Lemma 4.6, $\|\boldsymbol{W}_p\|_F$ decreases exponentially fast with respect to p with probability at least $1 - L^{-\alpha+1}$,

$$\|\boldsymbol{W}_p\|_F \leq \frac{1}{2}\|\boldsymbol{W}_{p-1}\|_F, \quad \|\boldsymbol{W}_p\|_F \leq 2^{-p}\sqrt{kn}. \tag{4.24}$$

The value of P can be determined to make $\|\boldsymbol{W}_P\|_F = \|\boldsymbol{Y}_\Omega - \mathrm{sgn}(\boldsymbol{X}_0)\|_F \leq \frac{1}{4\sqrt{2}\gamma}$. Hence,

$$2^{-P}\sqrt{kn} \leq \frac{1}{4\sqrt{2}\gamma} \implies P \geq \frac{\log(4\sqrt{2kn}\gamma)}{\log 2}. \tag{4.25}$$

The next step is to bound the infinity norm of $\boldsymbol{Y}$ on $\Omega^\perp$. We first rewrite the recursive relation (4.23); $\boldsymbol{Y}$ has the following form,

$$\boldsymbol{Y} = -\frac{L}{Q}\sum_{p=1}^{P}\mathcal{A}_p^*\mathcal{A}_p\boldsymbol{W}_{p-1}. \tag{4.26}$$

Our aim is to achieve $\|\boldsymbol{Y}_{\Omega^\perp}\|_\infty \leq \frac{1}{2}$. It suffices to make $\|\mathcal{P}_{\Omega^\perp}\mathcal{A}_p^*\mathcal{A}_p(\boldsymbol{W}_{p-1})\|_\infty \leq 2^{-p-1}\frac{Q}{L}$ in order to make $\|\boldsymbol{Y}\|_\infty \leq \frac{1}{2}$. $\mathcal{P}_{\Omega^\perp}$ is the operator which orthogonally projects a matrix onto $\Omega^\perp$. For each p,

$$\mathrm{E}(\mathcal{A}_p^*\mathcal{A}_p(\boldsymbol{W}_{p-1})) = \boldsymbol{T}_p\boldsymbol{W}_{p-1},$$

since $\boldsymbol{W}_{p-1}$ is known at step p and $\mathcal{A}_p^*\mathcal{A}_p$ is independent of $\boldsymbol{W}_{p-1}$. $\boldsymbol{T}_p\boldsymbol{W}_{p-1}$ has the same support as $\boldsymbol{W}_{p-1}$ does and thus this gives $\mathrm{E}\left[\mathcal{P}_{\Omega^\perp}\mathcal{A}_p^*\mathcal{A}_p(\boldsymbol{W}_{p-1})\right] = \boldsymbol{0}$. The next theorem proves that $\|\mathcal{P}_{\Omega^\perp}\mathcal{A}_p^*\mathcal{A}_p(\boldsymbol{W}_{p-1})\|_\infty \leq 2^{-p-1}\frac{Q}{L}$ for all $1 \leq p \leq P$ with probability at least $1 - L^{-\alpha+1}$ if $L \geq PQ$ and $Q \geq C_\alpha\mu_{\max}^2 kn(\log(kN) + \log(L))$.

Theorem 4.11. *Conditioned on Lemma 4.6 and (4.24), for fixed p,*

$$\mathrm{Pr}\left(\|\mathcal{P}_{\Omega^\perp}\mathcal{A}_p^*\mathcal{A}_p(\boldsymbol{W}_{p-1})\|_\infty > \frac{Q}{2^{p+1}L}\right) \leq 2kN\exp\left(-\frac{3Q}{128kn}\right).$$

In particular this gives

$$\mathrm{Pr}\left(\|\mathcal{P}_{\Omega^\perp}\mathcal{A}_p^*\mathcal{A}_p(\boldsymbol{W}_{p-1})\|_\infty \leq \frac{Q}{2^{p+1}L}, \quad \forall\, 1 \leq p \leq P\right) \geq 1 - PL^{-\alpha} \geq 1 - L^{-\alpha+1},$$

by choosing $Q \geq \frac{128\mu_{\max}^2\alpha}{3}kn(\log(kN) + \log(L))$ and taking the union bound for all p.

Proof: For any $\boldsymbol{z} = \mathrm{vec}(\boldsymbol{Z})$ where $\boldsymbol{Z}: k \times N$ has support Ω, we have

$$\mathcal{A}_p^*\mathcal{A}_p(\boldsymbol{Z}) = \boldsymbol{\Phi}_p^*\boldsymbol{\Phi}_p\boldsymbol{z}$$

directly following from (3.4) and (3.5). Therefore,

$$\|\mathcal{P}_{\Omega^\perp}\mathcal{A}_p^*\mathcal{A}_p(\boldsymbol{Z})\|_\infty = \max_{s\in\Omega^\perp}|\langle\boldsymbol{e}_s, \boldsymbol{\Phi}_p^*\boldsymbol{\Phi}_p\boldsymbol{z}\rangle|,$$

where $\boldsymbol{e}_s$ is a $kN \times 1$ unit vector with its s-th entry equal to 1. Here $s > kn$ because only the first kn entries of $\boldsymbol{z}$ are nonzero. The main idea of the proof is to estimate $|\langle\boldsymbol{e}_s, \boldsymbol{\Phi}_p^*\boldsymbol{\Phi}_p\boldsymbol{z}\rangle|$ for each $s \in \Omega^\perp$ and take the union bound over all $s \in \Omega^\perp$. For any fixed $s \in \Omega^\perp$,

$$\langle\boldsymbol{e}_s, \boldsymbol{\Phi}_p^*\boldsymbol{\Phi}_p\boldsymbol{z}\rangle = \langle\boldsymbol{e}_s, \sum_{l\in\Gamma_p}\phi_l\phi_l^*\boldsymbol{z}\rangle = \sum_{l\in\Gamma_p}\langle\boldsymbol{e}_s, \phi_l\phi_l^*\boldsymbol{z}\rangle, \quad \phi_l = \bar{\boldsymbol{a}}_l \otimes \boldsymbol{b}_l.$$

19

Let $z_l := \langle \boldsymbol{e}_s, \boldsymbol{\phi}_l \boldsymbol{\phi}_l^* \boldsymbol{z} \rangle$. Notice $\mathrm{E}(z_l) = 0$ because

$$\mathrm{E}(\langle \boldsymbol{e}_s, \boldsymbol{\phi}_l \boldsymbol{\phi}_l^* \boldsymbol{z} \rangle) = \boldsymbol{e}_s^*(\boldsymbol{I}_k \otimes \boldsymbol{b}_l \boldsymbol{b}_l^*)\boldsymbol{z} = \boldsymbol{e}_s^* \operatorname{vec}(\boldsymbol{b}_l \boldsymbol{b}_l^* \boldsymbol{Z}).$$

$\boldsymbol{b}_l \boldsymbol{b}_l^* \boldsymbol{Z}$ and $\boldsymbol{Z}$ have the same support and thus any entry of $\boldsymbol{b}_l \boldsymbol{b}_l^* \boldsymbol{Z}$ on $\Omega^\perp$ equals to 0. Therefore each z_l is a bounded and centered random variable. We can use scalar version of Bernstein inequality (a special version of Theorem 4.4) to estimate $\sum_{l \in \Gamma_p} z_l$. First we estimate $|z_l|$,

$$|z_l| \le |\boldsymbol{e}_s^* \boldsymbol{\phi}_l \boldsymbol{\phi}_l^* \boldsymbol{z}| = |\boldsymbol{e}_s^* \boldsymbol{\phi}_l| \cdot |\boldsymbol{\phi}_l^* \boldsymbol{z}| \le \frac{\mu_{\max}}{\sqrt{L}} |\boldsymbol{\phi}_l^* \boldsymbol{z}|.$$

There holds $|\boldsymbol{e}_s^* \boldsymbol{\phi}_l| \le \|\bar{\boldsymbol{a}}_l \otimes \boldsymbol{b}_l\|_\infty = \|\bar{\boldsymbol{a}}_l\|_\infty \|\boldsymbol{b}_l\|_\infty \le \frac{\mu_{\max}}{\sqrt{L}}$, which follows from the fact that $\boldsymbol{a}_l$ is a row of DFT matrix and each entry of $\boldsymbol{a}_l$ is of unit norm. Moreover, $\boldsymbol{z}$ has support Ω,

$$|z_l| \le \frac{\mu_{\max}}{\sqrt{L}} |\boldsymbol{\phi}_l^* \boldsymbol{z}| = \frac{\mu_{\max}}{\sqrt{L}} |\boldsymbol{\phi}_{l,\Omega}^* \boldsymbol{z}| \le \frac{\mu_{\max}}{\sqrt{L}} \|\boldsymbol{\phi}_{l,\Omega}\| \cdot \|\boldsymbol{z}\| \le \frac{\mu_{\max}}{\sqrt{L}} \cdot \sqrt{\frac{kn}{L}} \|\boldsymbol{z}\| \le \frac{\mu_{\max} \sqrt{kn} \|\boldsymbol{z}\|}{L}.$$

Therefore, $R := \max_l |z_l| \le \frac{\mu_{\max} \sqrt{kn} \|\boldsymbol{z}\|}{L}$. The variance of $\sum_{l \in \Gamma_p} z_l$ is estimated as

$$\begin{aligned}
\mathrm{E}\,|z_l|^2 &=& \mathrm{E}\,|\boldsymbol{e}_s^* \boldsymbol{\phi}_l \boldsymbol{\phi}_l^* \boldsymbol{z}|^2 = \mathrm{E}\,|\boldsymbol{e}_s^* \boldsymbol{\phi}_l|^2 |\boldsymbol{\phi}_l^* \boldsymbol{z}|^2 \\
&\le& \frac{\mu_{\max}^2}{L} \mathrm{E}\,|\boldsymbol{\phi}_l^* \boldsymbol{z}|^2 = \frac{\mu_{\max}^2}{L} \boldsymbol{z}^*(\boldsymbol{I}_N \otimes \boldsymbol{b}_l \boldsymbol{b}_l^*)\boldsymbol{z}
\end{aligned}$$

and

$$\sigma^2 = \sum_{l \in \Gamma_p} \mathrm{E}\,|z_l|^2 \le \frac{\mu_{\max}^2}{L} \boldsymbol{z}^*(\boldsymbol{I}_N \otimes \boldsymbol{T}_p)\boldsymbol{z} \le \frac{5\mu_{\max}^2 Q \|\boldsymbol{z}\|^2}{4L^2}.$$

Now we apply the Bernstein inequality in Theorem 4.4 and have the estimate for $\sum_{l \in \Gamma_p} z_l$,

$$\Pr\left(|\sum_{l \in \Gamma_p} z_l| > t \right) \le 2 \exp\left(-\frac{t^2/2}{5\mu_{\max}^2 Q \|\boldsymbol{z}\|^2/4L^2 + \mu_{\max}\sqrt{kn}\|\boldsymbol{z}\| t/3L} \right).$$

In particular, if we choose $\boldsymbol{z} = \operatorname{vec}(\boldsymbol{W}_{p-1})$ with $\|\boldsymbol{z}\| \le 2^{-p+1}\sqrt{kn}$ and $t = 2^{-p-1}\frac{Q}{L}$, we have

$$\Pr\left(|\sum_{l \in \Gamma_p} z_l| > \frac{Q}{2^{p+1}L} \right) \le 2 \exp\left(-\frac{3Q}{128\mu_{\max}^2 kn} \right).$$

We take the union bound over all $s \in \Omega^\perp$ and obtain

$$\Pr\left(\|\mathcal{P}_{\Omega^\perp} \mathcal{A}_p^* \mathcal{A}_p(\boldsymbol{W}_{p-1})\|_\infty > \frac{Q}{2^{p+1}L} \right) \le 2kN \exp\left(-\frac{3Q}{128\mu_{\max}^2 kn} \right).$$

∎

5 Stability theory

Now we consider the noisy case by solving the convex program (3.14). Our goal is to prove Theorem 3.2. The main ingredient of our proof is Theorem 4.33 in [17]. A variant of that theorem, adapted to our setting, reads as follows.

Theorem 5.1. *Let* $\widetilde{\boldsymbol{\phi}}_{s,t}$ *be the columns of* $\boldsymbol{\Phi} \in \mathbb{C}^{L \times kN}$, $\boldsymbol{v}_0 = \mathrm{vec}(\boldsymbol{X}_0) \in \mathbb{C}^{kN \times 1}$ *with support on* Ω *and* $\boldsymbol{y} = \boldsymbol{\Phi}\boldsymbol{v}_0 + \boldsymbol{w}$ *with* $\|\boldsymbol{w}\| \leq \eta$. *For* $\delta, \beta, \theta, \eta, \tau \geq 0$ *and* $\delta < 1$,

$$\|\boldsymbol{\Phi}_\Omega^* \boldsymbol{\Phi}_\Omega - \boldsymbol{I}_\Omega\| \leq \delta, \quad \max_{1 \leq s \leq k, t > n} \|\boldsymbol{\Phi}_\Omega^* \widetilde{\boldsymbol{\phi}}_{s,t}\| \leq \beta, \tag{5.1}$$

and that there exists a vector $\boldsymbol{q} = \boldsymbol{\Phi}^* \boldsymbol{p} \in \mathbb{C}^{kN \times 1}$ *with* $\boldsymbol{p} \in \mathbb{C}^L$ *such that*

$$\|\boldsymbol{q}_\Omega - \mathrm{sgn}(\boldsymbol{v}_0)\| \leq \frac{1}{4\sqrt{2}\gamma}, \quad \|\boldsymbol{q}_{\Omega^\perp}\|_\infty \leq \theta, \quad \|\boldsymbol{p}\| \leq \tau\sqrt{kn}. \tag{5.2}$$

If $\rho : \theta + \frac{\beta}{4\sqrt{2}\gamma(1-\delta)} < 1$, *then the minimizer* $\hat{\boldsymbol{v}}$ *to* (3.14) *satisfies*

$$\|\hat{\boldsymbol{v}} - \boldsymbol{v}_0\| \leq (C_0 + C_1\tau\sqrt{kn})\eta \tag{5.3}$$

where C_0 *and* C_1 *are two constants.*

Actually, we have already proven in the previous section that we can choose $\delta = \frac{1}{2}, \theta = \frac{1}{2}$. The only two unknown components are τ and β. τ follows from computing the norm of $\boldsymbol{p}$. We need to know the coherence μ of $\boldsymbol{\Phi}$ to estimate β and claim that $\beta \leq 1$. Define the coherence as the largest correlation between two different columns of the $L \times kN$ matrix $\boldsymbol{\Phi}$,

$$\mu := \max_{i_1 \neq i_2 \text{ or } j_1 \neq j_2} |\langle \widetilde{\boldsymbol{\phi}}_{i_1,j_1}, \widetilde{\boldsymbol{\phi}}_{i_2,j_2} \rangle|, \quad \widetilde{\boldsymbol{\phi}}_{s,t} = \mathrm{diag}(\widetilde{\boldsymbol{b}}_s)\widetilde{\boldsymbol{a}}_t \tag{5.4}$$

where $\widetilde{\boldsymbol{\phi}}_{s,t}$ is defined in (3.7).

5.1 Estimation of μ and β.

It is easy to obtain a bound of $\|\boldsymbol{\Phi}_\Omega^* \widetilde{\boldsymbol{\phi}}_{s,t}\|$ for $1 \leq s \leq k$ and $t > n$ with μ,

$$\|\boldsymbol{\Phi}_\Omega^* \widetilde{\boldsymbol{\phi}}_{s,t}\| \leq \sqrt{kn}\mu.$$

It suffices to bound μ by $1/\sqrt{kn}$ in order to achieve $\beta \leq 1$. The following lemma (see Lemma A.9 in [17]) will be used in the proof.

Lemma 5.2. *For any matrix* $\boldsymbol{C}$, *any entry is bounded by its operator norm, i.e.,*

$$\max_{i,j} |\boldsymbol{C}_{ij}| \leq \|\boldsymbol{C}\|. \tag{5.5}$$

Theorem 5.3. *There holds*

$$\max_{t>n} \|\boldsymbol{\Phi}_\Omega^* \widetilde{\boldsymbol{\phi}}_{s,t}\| \leq 1$$

with probability at least $1 - L^{-\alpha}$ *if*

1. $\boldsymbol{A}$ is a Gaussian random matrix and $L \geq 4C_{\tilde{\alpha}}\mu_{\max}^2 \max\{kn \log(L), \sqrt{kn} \log^2 L\}$ and $\tilde{\alpha} = 2\log(kN) + \alpha$.

2. $\boldsymbol{A}$ is a random Fourier matrix and $L \geq 4C_{\tilde{\alpha}}\mu_{\max}^2 kn \log(L)$ and $\tilde{\alpha} = 2\log(kN) + \alpha$.

Here, in both cases $C_{\tilde{\alpha}}$ is a function growing at most linearly with respect to $\tilde{\alpha}$.

Proof: We only prove the Gaussian case by using Lemma 4.3. The proof of the Fourier case is exactly the same. First we pick any two different columns $\widetilde{\boldsymbol{\phi}}_{i_1,j_1}$ and $\widetilde{\boldsymbol{\phi}}_{i_2,j_2}$ from $\boldsymbol{\Phi}$, where $1 \leq i_1, i_2 \leq k$ and $1 \leq j_1, j_2 \leq N$. Let $\widetilde{\Omega}$ be

$$\widetilde{\Omega} = \{(i_1, j_1), (i_1, j_2), (i_2, j_1), (i_2, j_2)\}. \tag{5.6}$$

$|\widetilde{\Omega}| \leq 4$ and $\boldsymbol{\Phi}_{\widetilde{\Omega}}$ contains $\widetilde{\boldsymbol{\phi}}_{i_1,j_1}$ and $\widetilde{\boldsymbol{\phi}}_{i_2,j_2}$. By Lemma 4.3, $\|\boldsymbol{\Phi}_{\widetilde{\Omega}}^*\boldsymbol{\Phi}_{\widetilde{\Omega}} - \boldsymbol{I}_{\widetilde{\Omega}}\| \leq \delta$ with probability $1 - L^{-\tilde{\alpha}}$ if $L \geq 4C_{\tilde{\alpha}} \max\{\log L/\delta^2, \log^2 L/\delta\}$. Therefore $|\langle \widetilde{\boldsymbol{\phi}}_{i_1,j_1}, \widetilde{\boldsymbol{\phi}}_{i_2,j_2}\rangle| \leq \delta$ follows from Lemma 5.2. Then we take the union bound over $(kN)^2/2$ pairs of indices (i_1, j_1) and (i_2, j_2). Therefore, $\mu = \max_{i_1,i_2,j_1,j_2} |\langle \widetilde{\boldsymbol{\phi}}_{i_1,j_1}, \widetilde{\boldsymbol{\phi}}_{i_2,j_2}\rangle| \leq 1/\sqrt{kn}$ with $L \geq 4C_{\tilde{\alpha}} \max\{kn \log L, \sqrt{kn} \log^2 L\}$ and probability at least $1 - (kN)^2 L^{-\tilde{\alpha}} \geq 1 - L^{-\alpha}$ if $\tilde{\alpha} = 2\log(kN) + \alpha$. ∎

5.2 Proof of Theorem 3.2 when $\boldsymbol{A}$ is a Gaussian random matrix

Proof: Let $\boldsymbol{p} = \boldsymbol{\Phi}_\Omega(\boldsymbol{\Phi}_\Omega^*\boldsymbol{\Phi}_\Omega)^{-1} \operatorname{sgn}(\boldsymbol{v}_0)$ and $\boldsymbol{q} = \boldsymbol{\Phi}^*\boldsymbol{p}$ as defined in (4.22). From Lemma 4.3, Theorem 4.10 and Theorem 5.3 we have

$$\|\boldsymbol{\Phi}_\Omega^*\boldsymbol{\Phi}_\Omega - \boldsymbol{I}_\Omega\| \leq \delta = \frac{1}{2}, \quad \theta = \|\boldsymbol{q}_{\Omega^\perp}\|_\infty < \frac{1}{2}, \quad \beta \leq 1$$

with probability at least $1 - O(L^{-\alpha+1})$.

$$\|\boldsymbol{p}\|^2 = \operatorname{sgn}(\boldsymbol{v}_0)^T(\boldsymbol{\Phi}_\Omega^*\boldsymbol{\Phi}_\Omega)^{-1}\operatorname{sgn}(\boldsymbol{v}_0) \leq 2kn \implies \|\boldsymbol{p}\| \leq \sqrt{2kn}, \quad \tau = \sqrt{2}$$

and also

$$\rho = \theta + \frac{\beta}{4\sqrt{2}\gamma(1-\delta)} \leq \frac{1}{2} + \frac{1}{2\sqrt{2}\gamma} < 1,$$

where γ is the operator norm of $\mathcal{A}$ and is certainly greater than 1. Applying Theorem 5.1 gives the desired result. ∎

5.3 Proof of Theorem 3.2 when $\boldsymbol{A}$ is a random Fourier matrix

Proof: We have an explicit form of $\boldsymbol{p}$ from (4.26)

$$\boldsymbol{p} = -\frac{L}{Q}(\mathcal{A}_1(\boldsymbol{W}_0), \cdots, \mathcal{A}_P(\boldsymbol{W}_{P-1}))^T,$$

such that $\boldsymbol{q} = \operatorname{vec}(\boldsymbol{Y})$, $\boldsymbol{Y} = \mathcal{A}^*(\boldsymbol{p}) = -\frac{L}{Q}\sum_{p=1}^P \mathcal{A}_p^*\mathcal{A}_p(\boldsymbol{W}_{p-1})$ and $\boldsymbol{Y}$ is an inexact dual certificate. The norm of $\boldsymbol{p}$ is

$$\|\boldsymbol{p}\|^2 = \frac{L^2}{Q^2}\sum_{i=1}^P \|\mathcal{A}_p(\boldsymbol{W}_{p-1})\|_F^2.$$

For each $\mathcal{A}_p(\boldsymbol{W}_{p-1})$,

$$\|\mathcal{A}_p(\boldsymbol{W}_{p-1})\|^2 = \langle \mathcal{A}_{p,\Omega}^* \mathcal{A}_{p,\Omega} \boldsymbol{W}_{p-1}, \boldsymbol{W}_{p-1} \rangle \leq \frac{3Q}{2L}\|\boldsymbol{W}_{p-1}\|_F^2 < \frac{2Q}{L} 4^{-p+1} kn,$$

which follows from (4.24) and $\|\mathcal{A}_{p,\Omega}^* \mathcal{A}_{p,\Omega}\| \leq \frac{3Q}{2L}$ in Lemma 4.6. Therefore,

$$\|\boldsymbol{p}\| \leq \sqrt{\frac{2L}{Q}} \sum_{p=1}^{P} 2^{-p+1}\sqrt{kn} \leq 2\sqrt{2P} \cdot \sqrt{kn}.$$

This implies $\tau = 2\sqrt{2P}$ where P is chosen in (4.25). We also have $\delta = \frac{1}{2}$, $\theta = \frac{1}{2}$ and $\beta \leq 1$ which are the same as Gaussian case and $\rho = \theta + \frac{\beta}{4\sqrt{2}\gamma(1-\delta)} < 1$. Now we have everything needed to apply Theorem 5.1. The solution $\hat{\boldsymbol{X}}$ to (3.14) satisfies

$$\|\hat{\boldsymbol{v}} - \boldsymbol{v}_0\|_F = \|\hat{\boldsymbol{X}} - \boldsymbol{X}_0\|_F \leq (C_0 + C_1\sqrt{P}\sqrt{kn})\eta,$$

with $P = O(\log(N))$, and C_0, C_1 are two constants. ∎

6 Numerical simulations

6.1 Recovery performance of SparseLift

We investigate the empirical probability of success of SparseLift over different pairs of (k, n) for fixed L and N. We fix $L = 128$, $N = 256$ and make both k and n range from 1 to 15. We choose $\boldsymbol{B}$ as the first k columns of an $L \times L$ unitary DFT matrix and $\boldsymbol{A}$ is sampled independently for each experiment. $\boldsymbol{x}$ has random supports with cardinality n and the nonzero entries of $\boldsymbol{h}$ and $\boldsymbol{x}$ yield $\mathcal{N}(0,1)$. For each (k, n), we run (3.11) 10 times and compute the probability of success. We classify a recovery as a success if the relative error is less than 1%,

$$\frac{\|\hat{\boldsymbol{X}} - \boldsymbol{X}_0\|_F}{\|\boldsymbol{X}_0\|_F} \leq 0.01, \quad \boldsymbol{X}_0 = \boldsymbol{h}_0 \boldsymbol{x}_0^T.$$

The results are depicted in Figure 1. The phase transition plots are similar for the case when $\boldsymbol{A}$ is a Gaussian random matrix or a random Fourier matrix. In this example choosing $kn < 70$ gives satisfactory recovery performance via solving SparseLift.

For comparison, we illustrate the performance when we solve the convex optimization problem that combines the nuclear norm and sparsity as in (3.10), see Figures 2 and 3. The values for λ are $\lambda = 0.1$ (the best value for λ for this experiment) and $\lambda = 10$. The settings of other parameters are exactly the same as the scenario of solving SparseLift. Figure 2 shows that (3.10) with $\lambda = 0.1$ performs similar to solving (3.11). However, if $\lambda = 10$ the results are clearly worse, in particular for the Gaussian case, see Figure 3. It supports our hypothesis that the performance of recovery is sensitive to the choice of λ and SparseLift is good enough to recover calibration parameters and the signal. This is in line with the findings in [28] which imply that convex optimization with respect to a combination of the norms $\|\cdot\|_1$ and $\|\cdot\|_*$ will not perform significantly better than using only one norm, the ℓ_1-norm in this case.

We also show the performance when using a mixed ℓ_2-ℓ_1-norm to enforce column-sparsity in $\boldsymbol{X}$ instead of just ℓ_1-norm. In this case we note a slight improvement over (3.11). We leave it as future research agenda to derive analogous results to Theorem 3.1 and Theorem 3.2 for the mixed ℓ_2-ℓ_1-norm in place of the ℓ_1-norm.
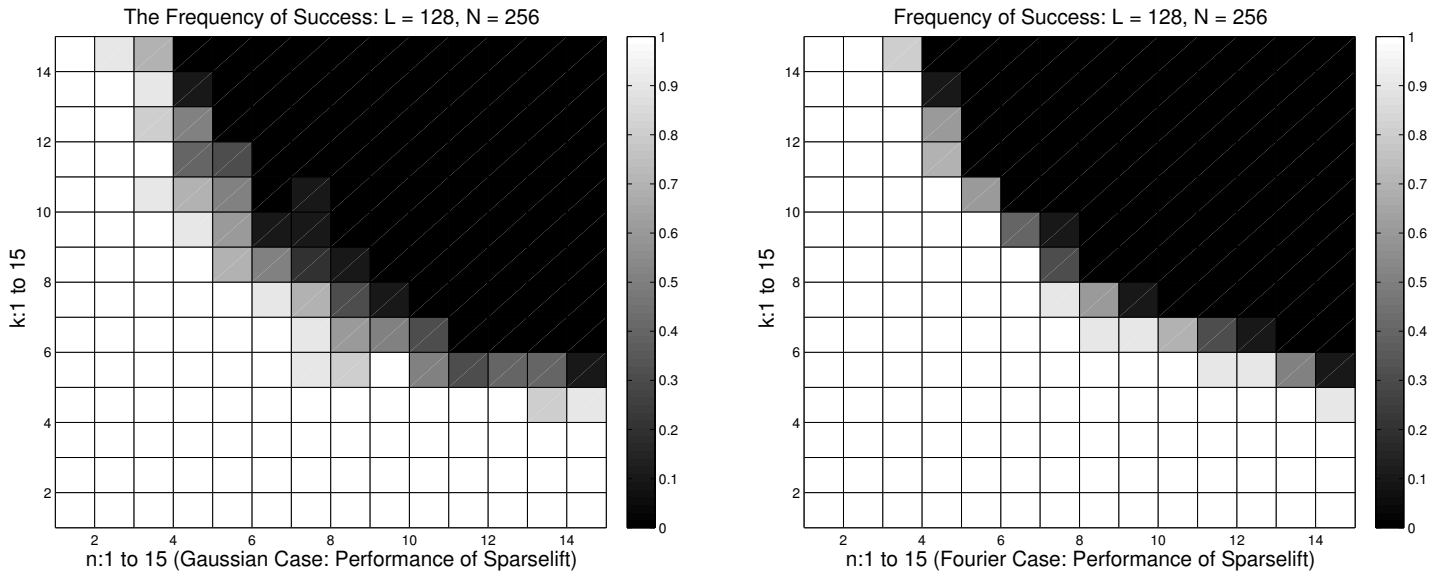
Figure 1: Phase transition plot of performance by solving SparseLift (3.11) directly.

6.2   Self-calibration and direction-of-arrival estimation

We examine the array self-calibration problem discussed in Section 1.1. We consider a scenario of three fully correlated equal-power signals. Note that the case of fully correlated signals is *much harder* than the case of uncorrelated (or weakly correlated) signals. Standard methods like MUSIC [33] or methods utilizing a mixed ℓ_1-ℓ_2 sparsity model as in [35] fail completely for fully correlated signals even in the absence of any calibration error. The reason is that these methods rely on the additional information obtained by taking multiple snapshots. Yet, for fully correlated signals all snapshots are essentially identical (neglecting noise) and therefore no new information is contained in multiple snapshots.

We use a circular array consisting of $L = 64$ omni-directional sensors. We assume that the calibration error is changing slowly across sensors, as for example in the case of sensors affected by changing temperature or humidity. There are of course many possibilities to model such a smooth change in calibration across sensors. In this simulation we assume that the variation can be modeled by a low-degree trigonometric polynomial (the periodicity inherent in trigonometric polynomials is fitting in this case, due the geometry of the sensors); in this example we have chosen a polynomial of degree 3. To be precise, the calibration error, i.e., the complex gain vector $\boldsymbol{d}$ is given by $\boldsymbol{d} = \boldsymbol{Bh}$, where $\boldsymbol{B}$ is a 64×4 matrix, whose columns are the first four columns of the 64×64 DFT matrix and $\boldsymbol{h}$ is a vector of length 4, whose entries are determined by taking a sample of a multivariate complex Gaussian random variable with zero mean and unit covariance matrix. The stepsize for the discretization of the grid of angles for the matrix $\boldsymbol{A}$ is 1 degree. The directions of arrival are set to $-10°, 5°$, and $20°$, and the SNR is 25dB. Since the signals are fully correlated we take only one snapshot.

We apply SparseLift, cf. (3.14), to recover the angles of arrival. Since the sensing matrix $\boldsymbol{A}$ does not fulfill the conditions required by Theorem 3.2 and the measurements are noisy, we cannot expect SparseLift to return a rank-one matrix. We therefore extract the vector with the angles of arrival by computing the right singular vector associated with the largest singular value of $\hat{\boldsymbol{X}}$. As can be seen in Figure 6.2 the estimate obtained via self-calibration provides a quite accurate approximation to the true solution.

24

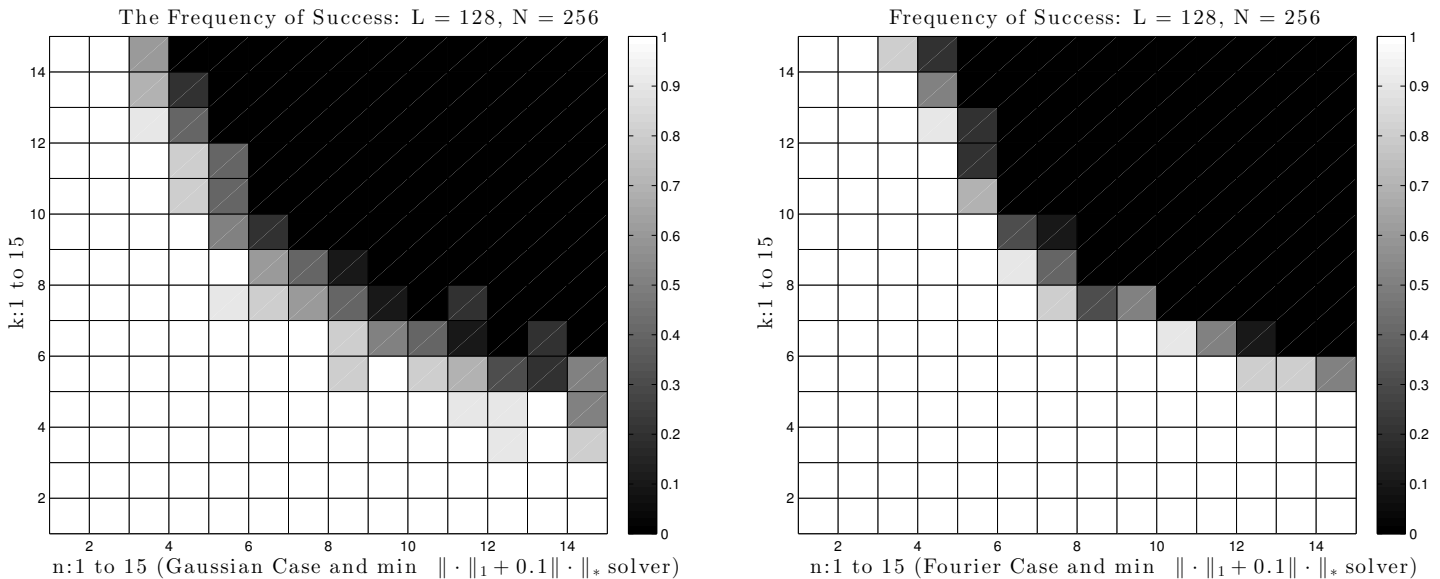Figure 2: Phase transition plot of performance via $\| \cdot \|_1 + \lambda \| \cdot \|_*$ minimization for $\lambda = 0.1$.

6.3   5G, the Internet of Things, and SparseLift

We briefly discussed in Section 1.2 how self-calibration problems of the form analyzed in this paper are anticipated to play a role in 5G wireless communication systems in connection with the Internet of Things, sporadic traffic and random access. Let us assume we deal with a CDMA communication scheme. To accommodate as many sporadic users as possible, we consider an "overloaded" CDMA system. This means that the matrix $\boldsymbol{A}$ containing the so-called spreading sequences (aptly called spreading matrix) has many more columns than rows. We use a random Fourier matrix for that purpose. Furthermore, we assume that the transmission channel is frequency-selective and time-invariant for the time of transmission of one sequence, but it may change from transmission to transmission. The channel is unknown at the receiver, it only knows an upper bound on the delay spread (the length of the interval containing the non-zero entries of $\boldsymbol{h}$). Moreover, the transmitted signal experiences additive Gaussian noise. The mathematician who is not familiar with all this communications jargon may rest assured that this is a standard scenario in wireless communication. In essence it translates to a linear system of equations of the form $\boldsymbol{y} = \boldsymbol{D}\boldsymbol{A}\boldsymbol{x} + \boldsymbol{w}$, where $\boldsymbol{A}$ is a flat random Fourier matrix, $\boldsymbol{D}$ is of the form $\boldsymbol{D} = \mathrm{diag}(\boldsymbol{B}\boldsymbol{h})$ and $\boldsymbol{w}$ is AWGN. To be specific, we choose the transmitted signal $\boldsymbol{x}$ to be of length $N = 256$ with $n = 5$ data-bearing coefficients (thus, $\|\boldsymbol{x}\|_0 = 5$), the locations of which are chosen at random. The spreading sequences are of length $L = 128$ (thus the system is overloaded by a factor of 2). The delay spread is assumed to be 5 and the SNR ranges from 0dB to 80dB (the latter SNR level is admittedly unrealistic in practical scenarios).

We attempt to recover $\boldsymbol{x}$ by solving (3.12). Here we assume $\boldsymbol{w}_i \sim \mathcal{N}(0, \sigma^2)$ and $\|\boldsymbol{w}\|^2/\sigma^2 \sim \chi_L^2$. We choose $\eta = (L + \sqrt{4L})^{1/2}\sigma$ as [1] in order to make $\boldsymbol{X}_0$ inside the feasible set.

The result is depicted in Figure 6.3; for comparison we also show the results when using a Gaussian random matrix as spreading matrix. We plot the relative error (on a dB scale) for different levels of SNR (also on a dB scale). The horizontal axis represents $\mathrm{SNR} = 10\log_{10}\left(\frac{\|\boldsymbol{X}_0\|_F^2}{\|\boldsymbol{w}\|^2}\right)$ and the vertical axis represents $10\log_{10}(\mathrm{Avg.Err.})$. We run 10 independent experiments for each SNR level and take the average relative error. The error curves show clearly the desirable linear behavior between SNR and mean square error with respect to the log-log scale.
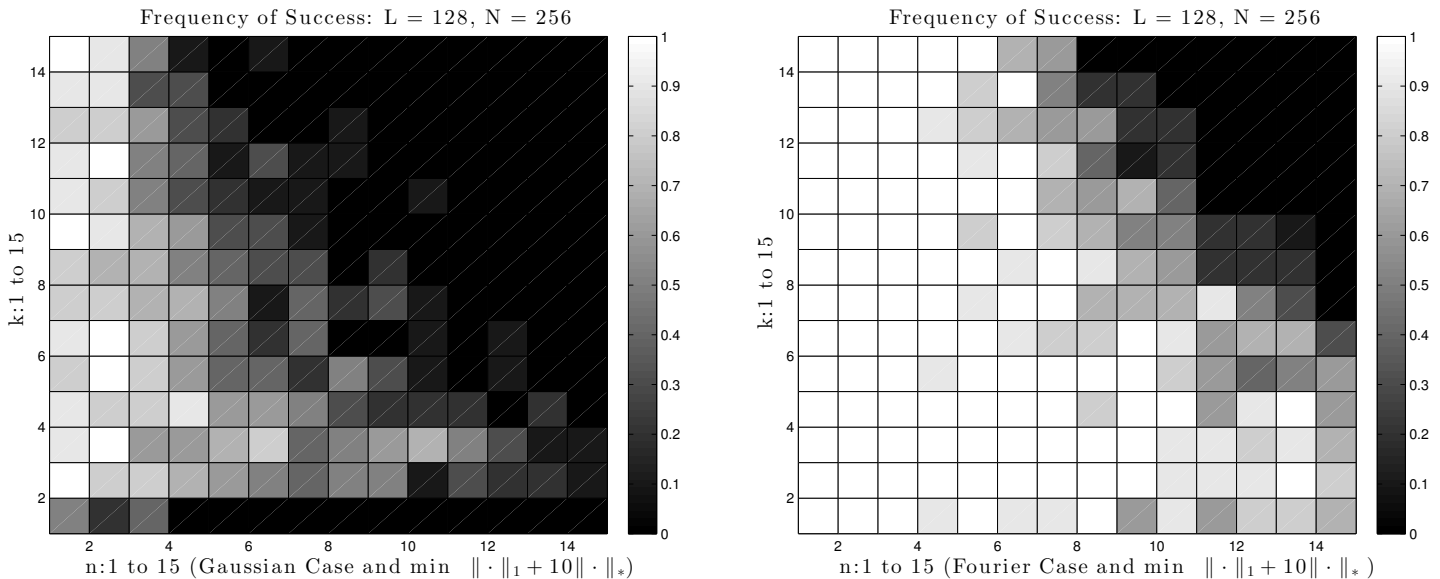
Figure 3: Phase transition plot of performance via $\|\cdot\|_1 + \lambda\|\cdot\|_*$ minimization for $\lambda = 10$.

7 Conclusion and outlook

Self-calibration is an emerging concept that will get increasingly important as hardware and software will become more and more intertwined. Several instances of self-calibration, such as blind deconvolution, have already been around for some time. In this work we have taken a step toward building a systematic mathematical framework for self-calibration. There remain more open questions than answers at this stage. Some obvious extensions of the results in this paper will be to consider the case where both $\boldsymbol{x}$ and $\boldsymbol{h}$ are sparse. Another useful generalization is the scenario where $\boldsymbol{x}$ is replaced by a matrix, like in the direction-of-arrival problem with multiple snapshots. Lifting the self-calibration problem in this case would lead to a tensor-valued (instead of a matrix-valued) underdetermined linear system [18]. The computational complexity for solving the associated relaxed convex problem will likely be too expensive for practical purposes and other methods will be needed. Extensions of the Wirtinger-Flow approach in [4] seem to be a promising avenue to derive computationally efficient algorithms for this and other self-calibration scenarios. Another important generalization is the case when there is mutual coupling between sensors, in which case $\boldsymbol{D}$ is no longer a diagonal matrix. Yet another very relevant setting consists of $\boldsymbol{D}$ depending non-linearly on $\boldsymbol{h}$, as is often the case when the sensors are displaced relative to their assumed locations. In conclusion, we hope that it has become evident that self-calibration is on the one hand an inevitable development in sensor design in order to overcome some serious current limitations, and on the other hand it is a rich source of challenging mathematical problems.
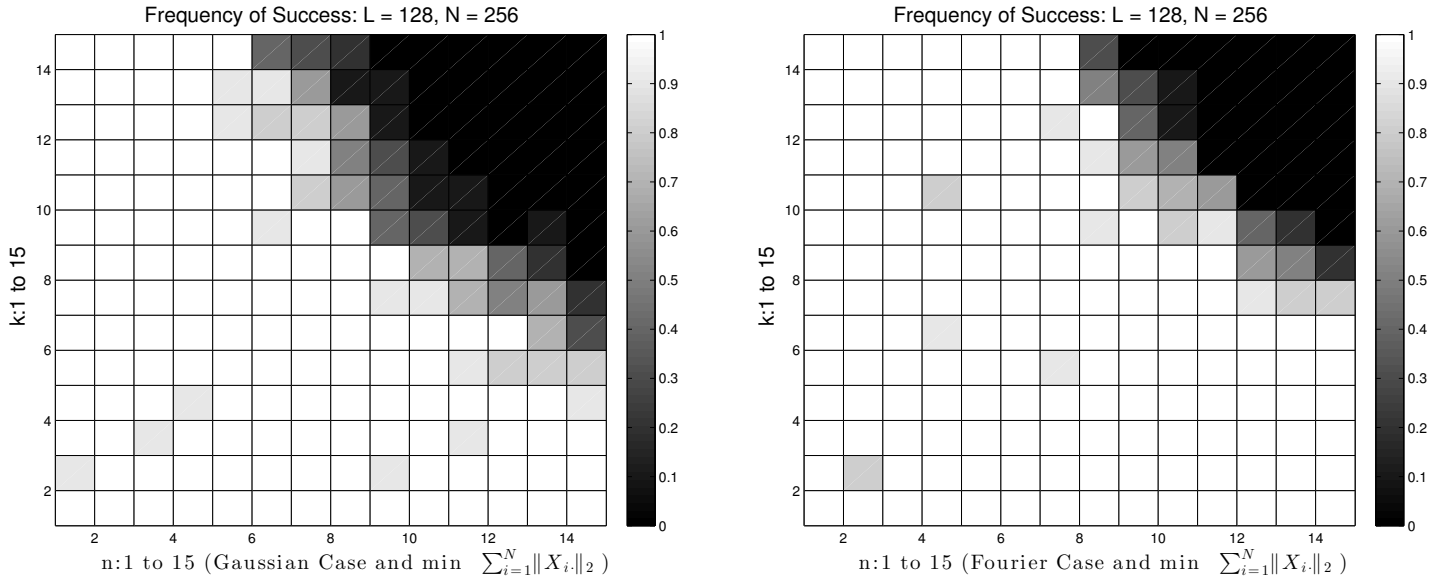
Acknowledgement

Figure 4: Phase transition plot of performance via $\| \cdot \|_{1,2}$ minimization.

References

[1] A. Ahmed, B. Recht, and J. Romberg. Blind deconvolution using convex programming. accepted in *IEEE Transactions on Information Theory*, arXiv:1211.5608, 2012.

[2] L. Balzano, L. and R. Nowak. Blind Calibration of Sensor Networks. *6th International Symposium on Information Processing in Sensor Networks, 2007.* IPSN 2007.

[3] E. Candès, Y. Eldar, T. Strohmer, and V. Voroninski. Phase retrieval via matrix completion. *SIAM J. Imag. Sci.*, 6(1):199–225, 2013.

[4] E. Candès, X. Li, and M. Soltanolkotabi. Phase retrieval via Wirtinger flow: Theory and algorithms. *Preprint, http://arxiv.org/abs/1407.1065*, 2014.

[5] E. Candès and B. Recht. Exact matrix completion via convex optimization. *Foundations of Computational Mathematics*, 9(6):717–772, 2009.

[6] E. Candès and B. Recht. Simple bounds for low-complexity model reconstruction. *Mathematical Programming*, 141(1-2): 577–589, 2013.

[7] E. Candès and T. Tao. The Power of Convex Relaxation: Near-Optimal Matrix Completion. *IEEE Trans. Inform. Theory*, 56(5):2053–2080, 2010.

[8] E. J. Candès, Y. C. Eldar, T. Strohmer, and V. Voroninski. Phase retrieval via matrix completion. *SIAM Journal on Imaging Sciences*, 6(1):199–225, 2013.

[9] E. J. Candes and X. Li. Solving quadratic equations via phaselift when there are about as many equations as unknowns. *Foundations of Computational Mathematics*, pages 1–10, 2012.

[10] E. J. Candès and J. Romberg. Sparsity and incoherence in compressive sampling. *Inverse problems*, 23(3):969, 2007.
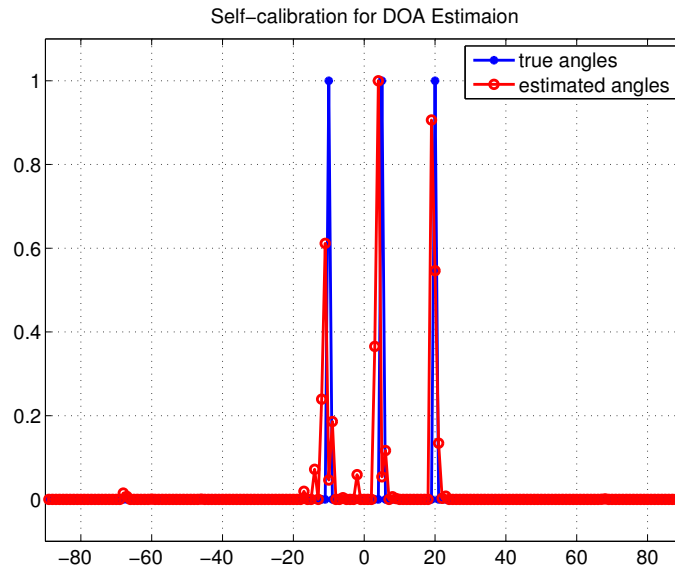
Figure 5: Estimated angles of arrival using self-calibration via SparseLift.

[11] E. J. Candès, T. Strohmer, and V. Voroninski. Phaselift: Exact and stable signal recovery from magnitude measurements via convex programming. *Communications on Pure and Applied Mathematics*, 66(8):1241–1274, 2013.

[12] E. J. Candès and T. Tao. Near-optimal signal recovery from random projections: Universal encoding strategies. *IEEE Trans. on Information Theory*, 52:5406–5425, 2006.

[13] Y. Chi, L.L. Scharf, A. Pezeshki, and A.R. Calderbank. Sensitivity to basis mismatch in compressed sensing. *IEEE Trans. Signal Processing*, 59(5):2182–2195, 2011.

[14] S. Choudhary and U. Mitra. Identifiability scaling laws in bilinear inverse problems. *Preprint*, arXiv:1402.2637, 2014.

[15] C. Davis and W. M. Kahan. The rotation of eigenvectors by a perturbation. III. *SIAM Journal on Numerical Analysis*, 7(1):1–46, 1970.

[16] D. L. Donoho. Compressed sensing. *IEEE Trans. on Information Theory*, 52(4):1289–1306, 2006.

[17] S. Foucart and H. Rauhut. *A mathematical introduction to compressive sensing*. Springer, 2013.

[18] B. Friedlander and T. Strohmer. Bilinear compressed sensing for array self-calibration. In *Asilomar Conference Signals, Systems, Computers*, Asilomar, 2014.

[19] B. Friedlander and A. J. Weiss. Self-calibration for high-resolution array processing. In S. Haykin, editor, *Advances in Spectrum Analysis and Array Processing, Vol. II*, chapter 10, pages 349–413. Prentice-Hall, 1991.

[20] J. Gorski, F. Pfeuffer, and K. Klamroth. Biconvex sets and optimization with biconvex functions: a survey and extensions. *Mathematical Methods of Operations Research*, 66(3):373–407, 2007.

[21] D. Gross. Recovering low-rank matrices from few coefficients in any basis. *Information Theory, IEEE Transactions on*, 57(3):1548–1566, 2011.

[22] M. Herman and T. Strohmer. General Deviants: An Analysis of Perturbations in Compressed Sensing. *IEEE J. Sel. Topics Sig. Proc.*, vol.4(2): 342–349, 2010.
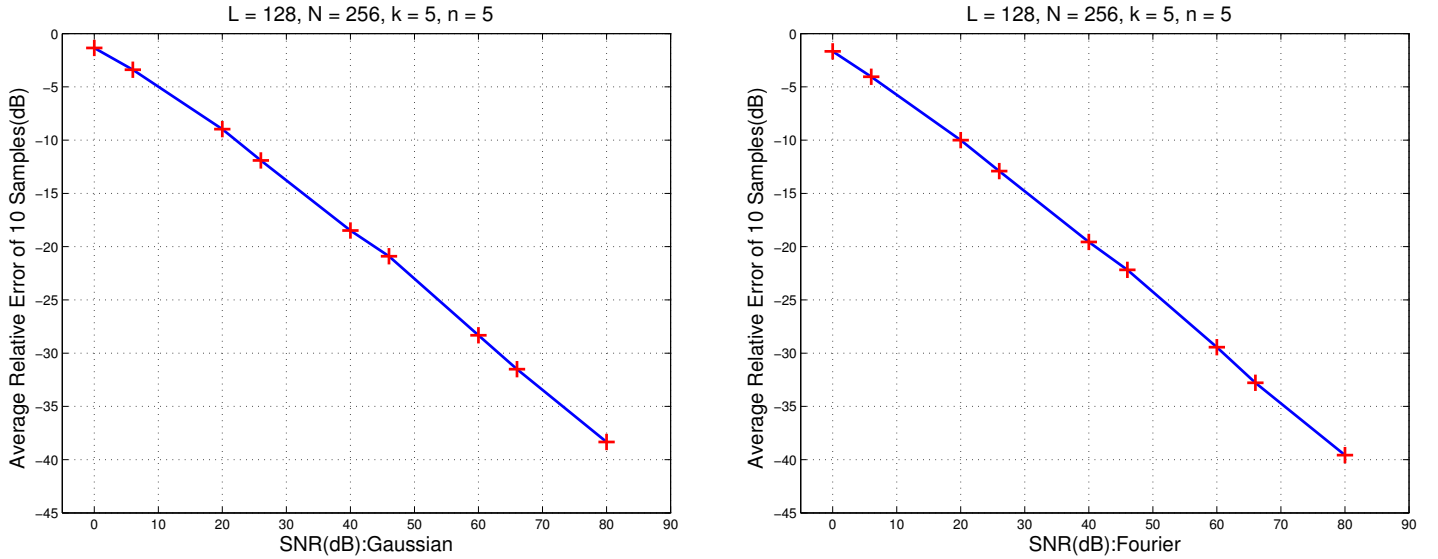
Figure 6: Sporadic traffic in 5G wireless communications: Performance of SparseLift in the presence of noise for different types of $\boldsymbol{A}$.

[23] V. Koltchinskii et al. Von Neumann entropy penalization and low-rank matrix estimation. *The Annals of Statistics*, 39(6):2936–2973, 2011.

[24] K. Lee, Y. Wu, and Y. Bresler. Near optimal compressed sensing of sparse rank-one matrices via sparse power factorization. *Preprint, arXiv:1312.0525*, 2013.

[25] A. Levin, Y. Weiss, F. Durand, and W. T. Freeman, Understanding blind deconvolution algorithms, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 33, no. 12, pp. 2354–2367, 2011.

[26] X. Li and V. Voroninski. Sparse signal recovery from quadratic measurements via convex programming. *SIAM Journal on Mathematical Analysis*, 45(5):3019–3033, 2013.

[27] A. Liu, G. Liao, C. Zeng, Z. Yang, and Q. Xu. An eigenstructure method for estimating DOA and sensor gain-phase errors. *IEEE Trans. Signal Processing*, 59(12):5944–5956, 2011.

[28] S. Oymak, A. Jalali, M. Fazel, Y. Eldar, and B. Hassibi. Simultaneously structured models with application to sparse and low-rank matrices. *ArXiv preprint arXiv:1212.3753*, 2012.

[29] B. Recht, M. Fazel, and P. Parrilo. Guaranteed minimum rank solutions of matrix equations via nuclear norm minimization. *SIAM Review*, 52(3):471–501, 2010.

[30] C. See. Sensor array calibration in the presence of mutual coupling and unknown sensor gains and phases. *Electronics Letters*, 30(5):373–374, 1994.

[31] J. A. Tropp. User-friendly tail bounds for sums of random matrices. *Foundations of Computational Mathematics*, 12(4):389–434, 2012.

[32] P. Walk and P. Jung. A stability result for sparse convolutions. *Preprint*, arXiv.org/abs/1312.2222, 2013.

[33] A. Weiss and B. Friedlander. Eigenstructure methods for direction finding with sensor gain and phase uncertainties. *J. Circuits, Systems and Signal Processing*, 9(3):271–300, 1989.

[34] G. Wunder, H. Boche, T. Strohmer, and P. Jung. Sparse signal processing concepts for efficient 5G system design. *IEEE Access*, accepted. Also available at *arXiv.org/abs/1411.0435*, 2014.

[35] Z. Yang and L. Xie. Exact joint sparse frequency recovery via optimization methods. *Preprint, arxiv.org/abs/1405.6585*, 2014.