Blind Unwrapping of Modulo Reduced Gaussian Vectors: Recovering MSBs from LSBs

Elad Romanov
School of CSE
Hebrew University of Jerusalem, Israel
elad.romanov@mail.huji.ac.il

Or Ordentlich
School of CSE
Hebrew University of Jerusalem, Israel
or.ordentlich@mail.huji.ac.il

***Abstract*—We consider the problem of recovering n i.i.d samples from a zero mean multivariate Gaussian distribution with an unknown covariance matrix, from their modulo wrapped measurements, i.e., measurement where each coordinate is reduced modulo Δ, for some $\Delta > 0$. For this setup, which is motivated by quantization and analog-to-digital conversion, we develop a low-complexity iterative decoding algorithm. We show that if an informed decoder that knows the covariance matrix can recover each sample with small error probability, and n is large enough, the performance of the proposed blind recovery algorithm closely follows that of the informed one. We complement the analysis with numeric results that show that the algorithm performs well even in non-asymptotic conditions.**

I. Introduction

Let $\mathbf{X}_1, \ldots, \mathbf{X}_n \overset{i.i.d.}{\sim} \mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$ be n i.i.d. realizations of a zero-mean K-dimensional Gaussian random vector with covariance matrix $\mathbf{\Sigma} \in \mathbb{R}^{K \times K}$. Let $\mathbf{X}_i^*$ be the K-dimensional vector obtained by reducing each coordinate of $\mathbf{X}_i$ modulo Δ, for some $\Delta > 0$. This paper studies the problem of *blindly* reconstructing the original n samples $\{\mathbf{X}_1, \ldots, \mathbf{X}_n\}$ from their wrapped counterparts $\{\mathbf{X}_1^*, \ldots, \mathbf{X}_n^*\}$, where the term *blind* refers to *without knowing the covariance matrix* $\mathbf{\Sigma}$. As the modulo operation can be thought of as discarding the most significant bits (MSBs) in the binary representation of each coordinate and keeping only the least significant bits (LSBs), this problem can be alternatively thought of as that of blindly recovering the MSBs of the coordinates from their LSBs.

Since the modulo operation is not invertible, it should be clear that even in the *informed* case, where $\mathbf{\Sigma}$ is known, reconstruction of $\mathbf{X}$ from $\mathbf{X}^*$ can never be guaranteed to succeed, and there is always a finite error probability associated with the reconstruction process. The error probability in the informed case depends on the interplay between the covariance matrix $\mathbf{\Sigma}$ and the modulo size Δ. As blind recovery algorithms cannot do better than informed ones, our goal is to develop a blind algorithm that successfully recovers $\{\mathbf{X}_1, \ldots, \mathbf{X}_n\}$ from $\{\mathbf{X}_1^*, \ldots, \mathbf{X}_n^*\}$ with high probability, whenever Δ and (the unknown) $\mathbf{\Sigma}$ are such that an informed algorithm can achieve a high successful recovery probability.

The main motivation for studying the blind unwrapping problem comes from recent trends in the study of analog-to-digital converters (ADCs). In many emerging applications in communication and signal processing one needs to digitize highly correlated analog processes, where each process is observed by a separate ADC. As a representative, but by no means exclusive, example, consider the front-end of a massive multiple-input multiple-output (MIMO) receiver, where the number of antennas can be of the order of tens and even hundreds, whereas the number of users it serves is moderate, making the signals observed by the various antennas highly correlated. It was recently proposed to use the so-called *modulo ADCs* in such scenarios [1], [2]. A modulo ADC is a device that first reduces its input modulo Δ, and only then quantizes it. The modulo reduction limits the dynamic range of the resulting signal to the interval $\left[-\frac{\Delta}{2}, \frac{\Delta}{2}\right)$, which means that when Δ is small, one can quantize the wrapped signal to within a good precision using only a few bits. As high correlation between the signals observed by various modulo ADCs allows for correct reconstruction even with small Δ, this architecture successfully exploits the correlation between the signals for reducing the burden from the ADCs. Previous works [1], [2], [3] studied the performance of modulo ADCs for correlated signals, under the assumption that the decoder is informed of $\mathbf{\Sigma}$. In practice, one has no access to $\mathbf{\Sigma}$ and the blind setup is more appropriate for analog-to-digital conversion. Another motivation for the

This work was supported, in part, by ISF under Grant 1791/17, and by the GENESIS consortium and Heron Consortium via the Israel Ministry of Economy and Industry.

blind unwrapping problem comes from the problem of communication in the presence of additive noise with unknown covariance. See [4].

The main contribution of this work is an iterative algorithm, with complexity $\mathcal{O}\left(n^2 \log K + n\text{poly}K\right)$ for the blind unwrapping problem. We derive analytic results that demonstrate that if the error probability of the benchmark informed unwrapping algorithm is small enough, and n is large enough, then our blind algorithm attains roughly the same performance as those of the informed benchmark algorithm. We complement the analysis with some numerical experiments which show that our algorithm performs surprisingly well even when n and the informed error probability are quite moderate. In particular, the experiments show that the number of measurements required for successful blind unwrapping is dictated by the number of strong eigenvalues in $\mathbf{\Sigma}$, rather than the dimension of $\mathbf{\Sigma}$. In particular, for problem of a "sparse" nature, such as massive MIMO for example, the algorithm performs well with a relatively small number of measurements.

II. Proposed Algorithm, and Main Results

Define the modulo operation

$$x^* = [x] \bmod \Delta \triangleq x - \Delta \left\lceil \frac{x}{\Delta} \right\rfloor \in \left[-\frac{\Delta}{2}, \frac{\Delta}{2}\right),$$

where $\lceil t \rfloor \triangleq \text{argmin}_{b\in\mathbb{Z}} |t-b|$ is the "round" operation, which returns the closest integer to t, with the convention that $\lceil a + \frac{1}{2} \rfloor = a$ for $a \in \mathbb{Z}$. For a vector $\mathbf{x} \in \mathbb{R}^K$, we write $\mathbf{x}^*$ for the vector obtained by applying the modulo operation on each coordinate of $\mathbf{x}$, i.e., $\mathbf{x}^* \triangleq [x_1^* \ \cdots \ x_K^*]^T$.

Let $\mathbf{\Sigma} \in \mathbb{R}^{K\times K}$ be a positive definite covariance matrix, and let $\{\mathbf{X}_1, \ldots, \mathbf{X}_n\} \overset{i.i.d.}{\sim} \mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$. In the *Blind Unwrapping Problem* we are given only the modulo reduced random vectors $\{\mathbf{X}_1^*, \ldots, \mathbf{X}_n^*\}$ and our goal is to recover their unfolded versions $\{\mathbf{X}_1, \ldots, \mathbf{X}_n\}$, *without prior knowledge on the covariance matrix* $\mathbf{\Sigma}$. In particular, we are interested in devising an algorithm whose input is $\{\mathbf{X}_1^*, \ldots, \mathbf{X}_n^*\}$ and whose output is a set of n estimates $\{\hat{\mathbf{X}}_1, \ldots, \hat{\mathbf{X}}_n\}$. The performance of an algorithm is measured by

$$P_e \triangleq \Pr\left(\{\hat{\mathbf{X}}_1, \ldots, \hat{\mathbf{X}}_n\} \neq \{\mathbf{X}_1, \ldots, \mathbf{X}_n\}\right).$$

For a K-dimensional random vector $\mathbf{X} \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$ and a set $\mathcal{S} \subset \mathbb{R}^K$, we define the truncated random vector $\mathbf{Y} \sim [\mathbf{X}|\mathbf{X} \in \mathcal{S}]$, whose probability density function (pdf) is $f_{\mathbf{Y}}(\mathbf{y}) = \frac{f_{\mathbf{X}}(\mathbf{y})\mathbb{1}_{\{\mathbf{y}\in\mathcal{S}\}}}{\Pr(\mathbf{X}\in\mathcal{S})}$ and whose covariance matrix is $\mathbb{E}[\mathbf{Y}\mathbf{Y}^T] = \mathbb{E}[\mathbf{X}\mathbf{X}^T|\mathbf{X} \in \mathcal{S}]$. We propose a low-complexity algorithm for the blind unfolding problem. The algorithm is based on the following simple observation, which holds due to the fact the the modulo operation is invariant with respect to translation by integer copies of Δ.

Proposition 1: For any integer vector $\mathbf{a} \in \mathbb{Z}^K$ and and $\mathbf{x} \in \mathbb{R}^K$ it holds that

$$[\mathbf{a}^T\mathbf{x}^*] \bmod \Delta = [\mathbf{a}^T\mathbf{x}] \bmod \Delta.$$

The implication of Proposition 1 is that for $\mathbf{A} \in \mathbb{Z}^{K\times K}$, we have that $[\mathbf{A}\mathbf{x}^*]^* = [\mathbf{A}\mathbf{x}]^*$, and consequently, if $\mathbf{A}$ is further full-rank, and $\mathbf{A}\mathbf{X} \in \text{CUBE}$, where $\text{CUBE} \triangleq \left[-\frac{\Delta}{2}, \frac{\Delta}{2}\right)^K$, then $\mathbf{A}^{-1}\left([\mathbf{A}\mathbf{x}^*]^*\right) = \mathbf{x}$.

The algorithm iterates between two basic procedures: 1)Estimating the covariance matrix $\overset{\vee}{\mathbf{\Sigma}}$ of the Gaussian random vector $\mathbf{X}$ truncated to the set $\mathcal{S} = \text{CUBE}$, from the measurements $\{\mathbf{X}_1^*, \ldots, \mathbf{X}_n^*\}$. 2)Finding a full-rank integer matrix $\mathbf{A}$, based on $\overset{\vee}{\mathbf{\Sigma}}$, such that $\Pr(\mathbf{A}\mathbf{X} \notin \text{CUBE}) < \Pr(\mathbf{X} \notin \text{CUBE})$, provided that $\Pr(\mathbf{X} \notin \text{CUBE})$ was not very small to begin with, and updating the measurements set to $\{[\mathbf{A}\mathbf{X}_1^*]^*, \ldots, [\mathbf{A}\mathbf{X}_n^*]^*\} = \{[\mathbf{A}\mathbf{X}_1]^*, \ldots, [\mathbf{A}\mathbf{X}_n]^*\}$. In order to facilitate the analysis of the algorithm in the sequel, we will restrict attention to a subset of the full-rank integer matrices, namely the group of *unimodular* matrices $\text{GL}_K(\mathbb{Z})$, consisting of all matrices in $\mathbb{Z}^{K\times K}$ with determinant 1 or -1. Note that $\mathbf{A} \in \text{GL}_K(\mathbb{Z})$ implies that $\mathbf{A}^{-1} \in \text{GL}_K(\mathbb{Z})$.

Below we give the precise algorithm. The algorithm has two parameters: $d \in \mathbb{R}_+$ and $M \in \mathbb{N}$, that are chosen by the designer. The main algorithm makes use of the procedure EstTruncatedCov which will be described immediately.

Inputs: $(\mathbf{X}_1^*, \ldots, \mathbf{X}_n^*)$, Δ, and two design parameters d and M.

Main Algorithm:

- Initialization: $\mathbf{A} = \mathbf{I}$, $\tilde{\mathbf{A}} = \mathbf{0}^{K\times K}$, $\text{ctr} = 0$, $(\mathbf{V}_1^*, \ldots, \mathbf{V}_n^*) = (\mathbf{X}_1^*, \ldots, \mathbf{X}_n^*)$
- While $\tilde{\mathbf{A}} \neq \mathbf{I}$ and $\text{ctr} < M$
  1) $\overset{\vee}{\mathbf{\Sigma}} = \text{EstTruncatedCov}\left((\mathbf{V}_1^*, \ldots, \mathbf{V}_n^*), d\right)$
  2) Compute

$$\tilde{\mathbf{A}} = [\tilde{\mathbf{a}}_1|\cdots|\tilde{\mathbf{a}}_K]^T = \underset{\bar{\mathbf{A}}\in\text{GL}_K(\mathbb{Z})}{\text{argmin}} \max_{k\in[K]} \bar{\mathbf{a}}_k^T \overset{\vee}{\mathbf{\Sigma}} \bar{\mathbf{a}}_k, \tag{1}$$

  3) Set $\mathbf{V}_j^* \leftarrow [\tilde{\mathbf{A}}\mathbf{V}_j^*]^* = [\tilde{\mathbf{A}}\mathbf{V}_j]^*$ for $j = 1, \ldots, n$, $\mathbf{A} \leftarrow \tilde{\mathbf{A}} \cdot \mathbf{A}$, and $\text{ctr} \leftarrow \text{ctr} + 1$

Outputs: $\mathbf{A}$ and the estimates $\hat{\mathbf{X}}_j = \mathbf{A}^{-1}\mathbf{V}_j^*$, for $j = 1, \ldots, n$.

The algorithm $\text{EstTruncatedCov}\left((\mathbf{V}_1^*, \ldots, \mathbf{V}_n^*), d\right)$, which is used within the main algorithm, is as follows.

Inputs: $(\mathbf{V}_1^*, \ldots, \mathbf{V}_n^*)$, and a design parameter d.

EstTruncatedCov Algorithm:

1) Set $\mathbf{V}_0^* = \mathbf{0}$

2) Construct a graph where each of the $n+1$ points $(\mathbf{V}_0^*, \mathbf{V}_1^*, \ldots, \mathbf{V}_n^*)$ is a vertex, and an edge between $\mathbf{V}_i^*$ and $\mathbf{V}_j^*$ exists iff $\|\mathbf{V}_i^* - \mathbf{V}_j^*\|_2 < d$
3) Find the connected component of $\mathbf{V}_0^* = \mathbf{0}$, and denote it by $\mathcal{T}$
4) If $|\mathcal{T}| < K+1$, set $d \leftarrow 1.1d$ and return to step 2; else

Output: set

$$\overset{\vee}{\mathbf{\Sigma}} = \frac{1}{|\mathcal{T}|} \sum_{\mathbf{t} \in \mathcal{T}} \mathbf{t}\mathbf{t}^T,$$

as the estimate of $\mathbb{E}\left[\mathbf{V}\mathbf{V}^T | \mathbf{V} \in \mathrm{CUBE}\right]$.

For fixed M and a fixed d whose value does not decrease with K, the complexity of the algorithm can be shown [4] to be $\mathcal{O}\left(n^2 \log K + n\mathrm{poly}(K) + \left(\frac{5}{4}\right)^{K^3/4}\right)$ if the shortest lattice basis problem in (1) is solved exactly, and $\mathcal{O}\left(n^2 \log K + n\mathrm{poly}(K)\right)$ if it is approximated using the LLL algorithm [5].

Our analysis requires several assumptions on the underlying covariance matrix, which we shall now specify. We start by introducing some notation. For a symmetric positive definite matrix $\mathbf{\Sigma} \in \mathbb{R}^{K \times K}$, $\mathbf{\Sigma} \succ \mathbf{0}$, denote the smallest eigenvalue by $\lambda_K(\mathbf{\Sigma})$, and let $\mathbf{\Sigma}^{-1/2}$ be such that $\mathbf{\Sigma}^{-1} = \mathbf{\Sigma}^{-1/2}\mathbf{\Sigma}^{-1/2}$. For a full-rank matrix $\mathbf{G} \in \mathbb{R}^{K \times K}$ we define the lattice $\Lambda(\mathbf{G}) = \mathbf{G} \cdot \mathbb{Z}^K$. The packing radius of a lattice $\Lambda(\mathbf{G})$ is defined as $r_0(\Lambda(\mathbf{G})) = \frac{1}{2}\min_{\mathbf{b} \in \mathbb{Z}^K \setminus \{\mathbf{0}\}} \|\mathbf{G}\mathbf{b}\|$ and the effective radius $r_{\text{eff}}(\Lambda(\mathbf{G}))$ is defined as the radius of a K-dimensional Euclidean ball whose volume is $|\mathbf{G}|$, i.e., $V_K r_{\text{eff}}^K(\Lambda(\mathbf{G})) = |\mathbf{G}|$, where V_K is the volume of a K-dimensional unit ball.

Now, suppose that $\mathbf{X} \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$, and let $0 < \epsilon < 1$, $\tau_{\text{min}} > 0$, $0 < P < 1$ and $0 < \rho_{\text{pack}} < 1$ be parameters; the guarantees we give on the performance of our algorithm will be given in terms of these. We say that $\mathbf{\Sigma}$ satisfies assumption Ai, $i = 1, \ldots, 4$, if

$$\begin{aligned}
A1 &: \min\left\{\Pr(\mathbf{A}\mathbf{X} \notin \mathrm{CUBE}) : \mathbf{A} \in \mathrm{GL}_K(\mathbb{Z})\right\} \le \epsilon \\
A2 &: \lambda_K(\mathbf{\Sigma}) \ge \tau_{\text{min}}^2 \\
A3 &: \Pr(\mathbf{X} \notin \mathrm{CUBE}) \le P \\
A4 &: \frac{r_0(\Lambda(\mathbf{\Sigma}^{-1/2}))}{r_{\text{eff}}(\Lambda(\mathbf{\Sigma}^{-1/2}))} \ge \rho_{\text{pack}}.
\end{aligned}$$

Assumption $A1$ simply means that the informed benchmark integer-forcing decoder [1], $\hat{\mathbf{X}}_{\text{IF}} = \mathbf{A}^{\text{opt},-1}[\mathbf{A}^{\text{opt}}\mathbf{X}^*]^*$, attains error probability $\Pr(\hat{\mathbf{X}}_{\text{IF}} \neq \mathbf{X}) = \Pr(\mathbf{A}^{\text{opt}}\mathbf{X} \notin \mathrm{CUBE}) \le \epsilon$, where $\mathbf{A}^{\text{opt}} \triangleq \mathrm{argmin}_{\mathbf{A} \in \mathrm{GL}_K(\mathbb{Z})} \Pr(\mathbf{A}\mathbf{X} \notin \mathrm{CUBE})$. Assumption $A2$ limits the "spikiness" of $\mathbf{\Sigma}$, assumption $A3$ requires that the initial conditions of the algorithm, corresponding to $\mathbf{A} = \mathbf{I}$ are not too bad, and assumption $A4$ is rather technical.

Let $Q(t) = \frac{1}{\sqrt{2\pi}} \int_t^\infty e^{-\frac{t^2}{2}} dt$ be the Q-function, and $Q^{-1}(t)$ be its inverse. We break the analysis of our algorithm into three parts: (1) We first show that if the procedure $\mathrm{EstTruncatedCov}$ is replaced with any method ("Genie") that always returns a sufficiently accurate estimate of $\mathbb{E}[\mathbf{V}\mathbf{V}^T | \mathbf{V} \in \mathrm{CUBE}]$, then the main algorithm converges to a near optimal $\mathbf{A} \in \mathrm{GL}_K(\mathbb{Z})$. This is the content of Theorem 1. (2) We then show that given enough samples $\mathbf{X}_1^*, \ldots, \mathbf{X}_n^*$, a single call to the specific procedure $\mathrm{EstTruncatedCov}$ we propose produces, with high probability, a good estimate for the truncated covariance matrix. This is done in Theorem 2. (3) Lastly, in Theorem 3, we combine the previous results to give a guarantee for (a slightly modified version of) the entire recovery algorithm.

Theorem 1 (Guarantee on Genie-Aided Algorithm): Let $\mathbf{X} \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$, where $\mathbf{\Sigma}$ satisfies assumptions $A1$ and $A3$, and ϵ satisfies $Q^{-1}\left(\frac{\epsilon}{2}\right) \ge 6\sqrt{K}$. Consider a genie-aided version of the main algorithm, where the procedure $\mathrm{EstTruncatedCov}((\mathbf{V}_1^*, \ldots, \mathbf{V}_n^*), d)$ is replaced with a genie that returns a matrix $\overset{\vee}{\mathbf{\Sigma}}$ that satisfies $(1-\beta)\mathbf{\Sigma}_{\mathbf{V},\text{trunc}} \preceq \overset{\vee}{\mathbf{\Sigma}} \preceq (1+\beta)\mathbf{\Sigma}_{\mathbf{V},\text{trunc}}$, where $\mathbf{\Sigma}_{\mathbf{V},\text{trunc}} \triangleq \mathbb{E}[\mathbf{V}\mathbf{V}^T | \mathbf{V} \in \mathrm{CUBE}]$, for some $0 \le \beta \le 0.1$. Then, after

$$M = \left\lceil \frac{\log(180)}{\log\left(\frac{4}{3+P}\right)} + 2 \right\rceil \tag{3}$$

iterations, the matrix $\mathbf{A}$ found by the algorithm must satisfy

$$\Pr(\mathbf{A}\mathbf{X} \notin \mathrm{CUBE}) \le K \cdot Q\left(0.99\sqrt{\frac{1-\beta}{1+\beta}} \cdot Q^{-1}\left(\frac{\epsilon}{2}\right)\right).$$

The proof can be found in [4]. It is based on showing, via the recently proved Gaussian Correlation Inequality, that if $\mathbf{V} \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$, then

$$f(\Pr(\mathbf{X} \notin \mathrm{CUBE})) \cdot \mathbf{\Sigma} \preceq \mathbf{\Sigma}_{\mathbf{V},\text{trunc}} \preceq \mathbf{\Sigma},$$

for some decreasing mapping $f : [0,1] \mapsto [0,1]$ which we explicitly specify. Consequently, by $A1$, we know that the matrix $\tilde{\mathbf{A}}$ calculated in step 2 of the algorithm will be such that $\max_{k \in [K]} \mathrm{diag}(\tilde{\mathbf{A}}^T \mathbf{\Sigma} \tilde{\mathbf{A}})$ is small. Leveraging this fact along with a Gaussian extremal inequality due to Latała and Oleszkiewicz [6], shows that the probability of missing CUBE decreases from iteration to iteration.

The next result shows that given enough samples and that assumption $A1$ holds with small ϵ, the algorithm $\mathrm{EstTruncatedCov}$ indeed produces an accurate estimate of $\mathbf{\Sigma}_{\mathbf{V},\text{trunc}}$ from the measurements $\{\mathbf{V}_1^*, \ldots, \mathbf{V}_n^*\}$. In [4] we give precise guarantees on the accuracy of the estimate. As the exact result is rather cumbersome, here we only state the asymptotic version.

Theorem 2 (Guarantee on truncated covariance es-

timation): Let $\mathbf{\Sigma} \in \mathbb{R}^{K\times K}$ satisfy assumptions $A1$, $A2$ and $A3$, where ϵ is such that $\kappa_\epsilon \triangleq \frac{Q^{-1}\left(\frac{\epsilon}{2}\right)}{\sqrt{K}} > 1$. Suppose that the procedure $\mathrm{EstTruncatedCov}((\mathbf{V}_1^*,\ldots,\mathbf{V}_n^*),d)$ is run with a distance parameter $d = 2\eta\sqrt{K}\cdot\tau_{\min}\cdot\kappa_\epsilon$, for some $\eta \in \left(0, 1-\frac{1}{\kappa_\epsilon}\right)$, and $\mathbf{V}_1^*,\ldots,\mathbf{V}_n^*$ are n independent wrapped samples from $\mathcal{N}(\mathbf{0},\mathbf{\Sigma})$. Then, with probability at least $1-p_{\text{est-err}}$, it holds that

$$0.999\mathbf{\Sigma}_{\mathbf{V},\text{trunc}} \preceq \overset{\vee}{\mathbf{\Sigma}} \preceq 1.001\mathbf{\Sigma}_{\mathbf{V},\text{trunc}}\,,$$

where for fixed K and η taking $\epsilon\to 0$ and $n=\epsilon^{-\zeta}$ for some $\zeta < (1-\eta)^2$ we have that $p_{\text{est-err}} < n\epsilon^{0.99(1-\eta)^2}$

The proof is given in [4]. Assume ϵ is small. The main observation is that for two points $\mathbf{V}_i^*, \mathbf{V}_j^*$ such that $\mathbf{V}_i^* = \mathbf{V}_i$ but $\mathbf{V}_j^* \neq \mathbf{V}_j$, it must hold that $\|\mathbf{V}_i^* - \mathbf{V}_j^*\| > d$ with high probability, for $d = 2\eta\sqrt{K}\cdot\tau_{\min}\cdot\kappa_\epsilon$, where the probability decreases with η. Thus, with high probability the connected component of $\mathbf{0}$ in the graph constructed by the procedure $\mathrm{EstTruncatedCov}$ contains only points that were not folded. If n is large enough, the points will be dense and almost all points that were not folded will land in the connected component.

At this point, one might naively think that Theorems 1 and 2 immediately imply that the main algorithm successfully recovers all the points, with high probability, provided that ϵ is small enough and n large enough. The situation, however, is more subtle: when we use, in each iteration, the entire sample to estimate the truncated covariance, we are inadvertently creating interdependencies between the samples as we are multiplying by $\tilde{\mathbf{A}}$. This means that in the following iteration the samples we use are no longer i.i.d, hence we cannot apply Theorem 2 as is. To mitigate this difficulty, we introduce a slight modification of the algorithm. The twist is as follows: given the design parameter M, we partition $[n]$ into M disjoint index sets $\mathcal{I}_1,\ldots,\mathcal{I}_M$, each of size n/M. In the main algorithm, in step 2 of the mth iteration, instead of applying $\mathrm{EstTruncatedCov}((\mathbf{V}_1^*,\ldots,\mathbf{V}_n^*),d)$ we apply $\mathrm{EstTruncatedCov}((\mathbf{V}_{i_{m,1}}^*,\ldots,\mathbf{V}_{i_{m,n/M}}^*),d)$, where $\mathcal{I}_m = \{i_{m,1},\ldots,i_{m,N/M}\}$. Thus, every sample only participates once (or never participates) in $\mathrm{EstTruncatedCov}$.

Theorem 3 (Success probability for the (modified) main algorithm: recovering $\mathbf{X}_1,\ldots,\mathbf{X}_n$): Let $\mathbf{X} \sim \mathcal{N}(\mathbf{0},\mathbf{\Sigma})$, where $\mathbf{\Sigma}$ satisfies assumptions $A1$, $A3$ and $A4$. Suppose that the modified main algorithm is run with parameter M satisfying (3), and distance parameter

$$d = 0.01\cdot 2\sqrt{K}\cdot\frac{\Delta}{\chi_1(\epsilon;P,K)}\,,$$

where

$$\chi_1(\epsilon;P,K) \triangleq \frac{K^{2K-\frac{1}{2}}\cdot V_K\cdot\left(Q^{-1}\left(\frac{\epsilon}{2K}\right)\right)^K}{2^K\cdot\rho_{\text{pack}}^K\cdot\left(Q^{-1}\left(\frac{P}{2}\right)\right)^{K-1}}\,.$$

Suppose that $n=\epsilon^{-\zeta}$ for some $\zeta < (0.99)^2$. Then there is some $\epsilon^* = \epsilon^*(K,P,\rho_{\text{pack}},\zeta) > 0$ such that if $\epsilon<\epsilon^*$, then $P_e \le n\epsilon^{0.96}$.

Remark 1: The absence of assumption $A2$ here seems odd at first sight. Instead, we need to use $A4$ to be able to control $\lambda_K(\mathbf{\Sigma}^{(t)})$ throughout the run of the algorithm.

III. Numerical Results

In all experiments below, we consider $K\times K$ covariance matrices of the form

$$\mathbf{\Sigma} = \mathbf{I}_K + \mathrm{snr}\mathbf{H}\mathbf{H}^T, \tag{4}$$

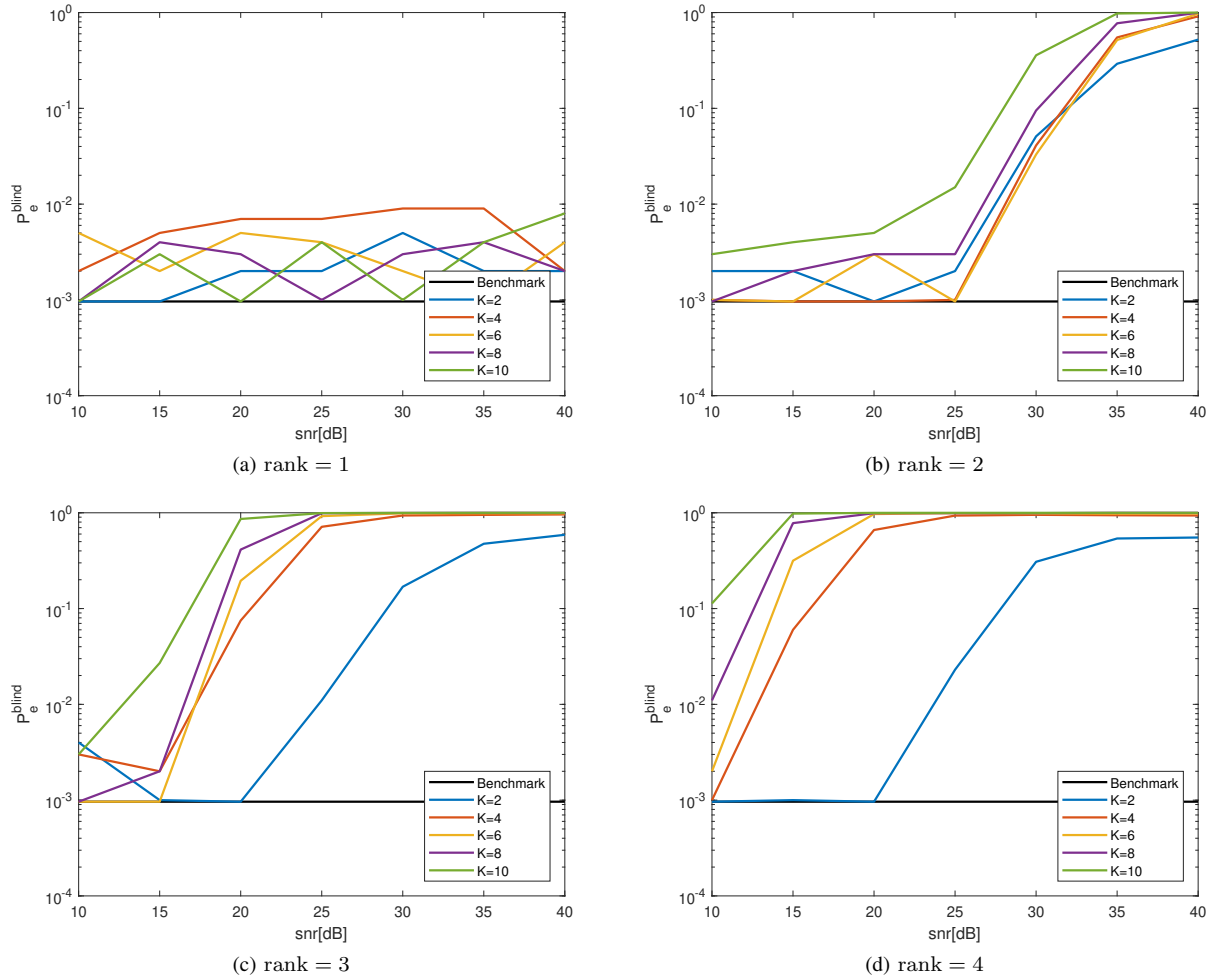
where $\mathbf{H}\in\mathbb{R}^{K\times\mathrm{rank}}$ for some integer rank, and $\mathrm{snr}>0$ is a parameter. Such covariance matrices correspond to the output of a narrowband MIMO channel $\mathbf{Y}_t = \mathbf{H}\mathbf{S}_t + \mathbf{Z}_t$, where $\mathbf{H}\in\mathbb{R}^{K\times\mathrm{rank}}$ is the channel matrix, $\mathbf{Z}_t$ is additive white Gaussian noise, and $\mathbf{S}_t$ is the vector of communication symbols transmitted at time t (which are assumed to be Gaussian i.i.d.). If one then applies a modulo ADC with modulo size Δ on the output of each receive antenna, the resulting K-dimensional vector would be of the form $\mathbf{X}_t^*$, where $\mathbf{X}_t = \mathbf{Y}_t + \mathbf{Q}_t$, and $\mathbf{Q}_t$ is the vector of quantization noises incurred by the modulo ADCs. Further making the simplifying assumption that $\mathbf{Q}_t$ is additive white Gaussian noise, we obtain that $\mathbf{X}_t\sim\mathcal{N}(\mathbf{0},\mathbf{\Sigma})$, where snr depends on the channel noise variance and the quantization noise variance.

We evaluate the performance of the algorithm for various values of K, rank, and snr. As our benchmark informed decoder, we take the integer-forcing decoder [1]. In the experiments, we find it convenient to set $\Delta = \Delta(\mathbf{\Sigma},\epsilon)$ such that the error probability of the integer-forcing decoder is fixed to some value, say ϵ. Thus, Δ may vary across different realizations of $\mathbf{\Sigma}$, but the (per-sample) error probability of the integer-forcing decoder remains fixed to ϵ throughout.

For covariance matrices of the form (4), we have that all eigenvalues of $\mathbf{\Sigma}$ are at least 1. Thus, we have that assumption A2 holds with $\tau_{\min}=1$, for all covariance matrices we consider here. Furthermore, by our choice of $\Delta(\mathbf{\Sigma},\epsilon)$ we have that assumption A1 holds. Thus, guided by our analysis we set $d = \eta\cdot\sqrt{K}\cdot\tau_{\min}\cdot Q^{-1}\left(\frac{\epsilon}{2}\right)$, with $\eta=0.5$, and $\tau_{\min}=1$ throughout all experiments. Furthermore, M was set to 30 in all experiments.

For fixed $\mathbf{H}$, snr, ϵ, and n, let $\mathbf{\Sigma}$ be as in (4) and define $P_e^{\text{blind}}(\mathbf{H},\mathrm{snr},\epsilon,n)$ as the probability that the estimates $\{\hat{\mathbf{X}}_1,\ldots,\hat{\mathbf{X}}_n\}$ produced by our algorithm are not all identical to $\{\mathbf{X}_1,\ldots,\mathbf{X}_n\}$, where the modulo size is $\Delta(\mathbf{\Sigma},\epsilon)$. Furthermore, for fixed K and rank we assume the entries of $\mathbf{H}$ are i.i.d. $\mathcal{N}(0,1)$, and define

$$P_e^{\text{blind}}(K,\mathrm{rank},\mathrm{snr},\epsilon,n) \triangleq \mathbb{E}\left[P_e^{\text{blind}}(\mathbf{H},\mathrm{snr},\epsilon,n)\right],$$



(a) rank $= 1$ (b) rank $= 2$

(c) rank $= 3$ (d) rank $= 4$

Fig. 1. Monte-Carlo evaluation of $P_e^{\text{blind}}(K, \text{rank}, \text{snr}, \epsilon, n)$ for $n = 1000$, $\epsilon \approx 9.6 \cdot 10^{-7}$, such that $1-(1-\epsilon)^n \approx 9.6 \cdot 10^{-4}$, and various values of K, rank and snr.

where the expectation is with respect to th randomness in $\mathbf{H}$. We estimate $P_e^{\text{blind}}(K, \text{rank}, \text{snr}, \epsilon, n)$ using Monte-Carlo simulations. In particular, for fixed $K, \text{rank}, \text{snr}, \epsilon$ and n we draw $r = 1000$ different realizations of $\mathbf{H}$, and for each one of those compute $\mathbf{\Sigma}$ according to (4). We then draw $\mathbf{X}_1, \ldots, \mathbf{X}_n \overset{i.i.d.}{\sim} \mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$, compute $\{\mathbf{X}_1^*, \ldots, \mathbf{X}_n^*\}$, where $\Delta = \Delta(\mathbf{\Sigma}, \epsilon)$, and apply the proposed algorithm that yield estimates $\{\hat{\mathbf{X}}_1, \ldots, \hat{\mathbf{X}}_n\}$. Our estimate for $P_e^{\text{blind}}(K, \text{rank}, \text{snr}, \epsilon, n)$ is taken as the fraction of realizations of $\mathbf{H}$ for which $\{\hat{\mathbf{X}}_1, \ldots, \hat{\mathbf{X}}_n\} \neq \{\mathbf{X}_1, \ldots, \mathbf{X}_n\}$

Figure 1 shows the Monte-Carlo estimate of $P_e^{\text{blind}}(K, \text{rank}, \text{snr}, \epsilon, n)$ for $n = 1000$, $\epsilon = 9.6 \cdot 10^{-7}$ and various values of K, rank and snr. In all experiments, whenever we had to solve an integer optimization problem of the form of (1), we have used the LLL algorithm [5] in order to get an approximate, possibly sub-optimal, solution. The results show that even for moderate values of n and ϵ, at moderate snr the algorithm performs close to the informed benchmark provided that rank, rather than K is small. The dependence on the rank rather than the problem's dimension is encouraging, because in many problems of practical interest where modulo ADCs are advantageous, the dimension is large, but the covariance matrix has a small number of significant eigenvalues.

REFERENCES

[1] O. Ordentlich and U. Erez, "Integer-forcing source coding," *IEEE Transactions on Information Theory*.

[2] O. Ordentlich, G. Tabak, P. K. Hanumolu, A. C. Singer, and G. W. Wornell, "A modulo-based architecture for analog-to-digital conversion," *IEEE Journal of Selected Topics in Signal Processing*.

[3] E. Domanovitz and U. Erez, "Outage probability bounds for integer-forcing source coding," in *ITW 2017*.

[4] E. Romanov and O. Ordentlich, "Blind unwrapping of modulo reduced Gaussian vectors: Recovering MSBs from LSBs," *arXiv preprint arXiv:1901.10396*, 2019.

[5] A. K. Lenstra, H. W. Lenstra, and L. Lovász, "Factoring polynomials with rational coefficients," *Mathematische Annalen*.

[6] R. Latala and K. Oleszkiewicz, "Gaussian measures of dilatations of convex symmetric sets," *Annals of probability*.