Solving inverse problems via auto-encoders

Pei Peng, Shirin Jalali and Xin Yuan

***Abstract*—Compressed sensing (CS) is about recovering a structured signal from its under-determined linear measurements. Starting from sparsity, recovery methods have steadily moved towards more complex structures. Emerging machine learning tools such as generative functions that are based on neural networks are able to learn general complex structures from training data. This makes them potentially powerful tools for designing CS algorithms. Consider a desired class of signals $\mathcal{Q}$, $\mathcal{Q} \subset \mathbb{R}^n$, and a corresponding generative function $g : \mathcal{U}^k \to \mathbb{R}^n$, $\mathcal{U} \subset \mathbb{R}$, such that $\sup_{\mathbf{x}\in\mathcal{Q}} \min_{\mathbf{u}\in\mathcal{U}^k} \frac{1}{\sqrt{n}}\|g(\mathbf{u}) - \mathbf{x}\| \leq \delta$. A recovery method based on g seeks $g(\mathbf{u})$ with minimum measurement error. In this paper, the performance of such a recovery method is studied, under both noisy and noiseless measurements. In the noiseless case, roughly speaking, it is proven that, as k and n grow without bound and δ converges to zero, if the number of measurements (m) is larger than the input dimension of the generative model (k), then asymptotically, almost lossless recovery is possible. Furthermore, the performance of an efficient iterative algorithm based on projected gradient descent is studied. In this case, an auto-encoder is used to define and enforce the source structure at the projection step. The auto-encoder is defined by encoder and decoder (generative) functions $f : \mathbb{R}^n \to \mathcal{U}^k$ and $g : \mathcal{U}^k \to \mathbb{R}^n$, respectively. We theoretically prove that, roughly, given $m > 40k \log \frac{1}{\delta}$ measurements, such an algorithm converges to the vicinity of the desired result, even in the presence of additive white Gaussian noise. Numerical results exploring the effectiveness of the proposed method are presented.**

***Index Terms*—Compressed sensing, generative models, inverse problems, auto-encoders, deep learning.**

I. Introduction

A. Problem statement

Solving inverse problems is at the core of many data acquisition systems, such as magnetic resonance imaging (MRI) and optical coherence tomography [1]. In many of such systems, through proper quantization in time or space, the measurement system can be modeled as a system of linear equations as follows. The unknown signal to be measured is a high-dimensional signal $\mathbf{x} \in \mathcal{Q}$, where $\mathcal{Q}$ represents a compact subset of $\mathbb{R}^n$. The measured signal can be represented as $\mathbf{y} = A\mathbf{x} + \mathbf{z}$. Here $A \in \mathbb{R}^{m\times n}$, $\mathbf{y} \in \mathbb{R}^m$ and $\mathbf{z} \in \mathbb{R}^m$ denote the sensing matrix, the measurement vector, and the measurement noise, respectively. Typically the main goal is to design an efficient algorithm that recovers $\mathbf{x}$ from the measurements $\mathbf{y}$. In addition to computational complexity, the efficiency of such an algorithm is measured in terms of its required number of measurements, its reconstruction quality, and its robustness to noise. While classic recovery methods were designed assuming that m is larger than n, i.e., the number of unknown parameters, during the last decade, researchers have shown that, since signals of interest are typically highly-structured, efficient recovery is possible, even if $m \ll n$.

The main focus in compressed sensing (CS), i.e., solving the described ill-posed linear inverse problem, has been on structures, such as sparsity. Many signals of interest are indeed sparse or approximately sparse in some transform domain, which makes sparsity a fundamental structure, both from a theoretical and from a practical perspective. However, most of such signals of interest, in addition to being sparse, follow other more complex structures as well. Enabling recovery algorithms to take advantage of the full structure of a class of signals could considerably improve the performance. This has motivated researchers in CS to explore algorithms that go beyond simple models such as sparsity.

Developing a CS recovery method involves two major steps: i) studying the desired class of signals (e.g., natural images, or MRI images) and discovering the structures that are shared among them, and ii) devising an efficient algorithm that given $\mathbf{y} = A\mathbf{x} + \mathbf{z}$, finds a signal that is consistent with the measurements $\mathbf{y}$ and also the discovered structures. For instance, the well-known iterative hard thresholding algorithm [2] is an algorithm that is developed for the case where the discovered structure is sparsity.

One approach to address the described first step is to design a method that automatically learns complex signal models from training data. In other words, instead of requiring domain experts to closely study a class of signals, we build an algorithm that discovers complex source models from training data. While designing such learning mechanisms is in general very complicated, generative functions (GFs) defined by trained neural networks (NNs) present a successful modern tool in this

This paper was presented in part at IEEE International Symposium on Information Theory, Paris, France 2019 and at NeurIPS 2019 Workshop on Solving Inverse Problems with Deep Networks.

P. Peng is a Ph.D. student at Department of Electrical & Computer Engineering, Rutgers University, pp566@scarletmail.rutgers.edu

S. Jalali is with Nokia Bell Labs, 600 Mountain Avenue, Murray Hill, NJ, 07974, USA, shirin.jalali@nokia-bell-labs.com

X. Yuan is with Nokia Bell Labs, 600 Mountain Avenue, Murray Hill, NJ, 07974, USA, xyuan@bell-labs.com

area. The well-known universal approximation theory (UAT) states that with proper weights, NNs can approximate any regular function with arbitrary precision [3]–[6]. This suggests that trained NNs operating as GFs are potentially capable of capturing complex unknown structures.

In recent years, availability of i) large training datasets on one hand, and ii) computational tools such as GPUs on the other hand, has led to considerable progress in training effective NNs with state-of-art performance. While initially such networks were mainly trained to solve classification problems, soon researchers realized that there is no fundamental reason to restrict our attention to such problems. And indeed researchers have explored application of NNs in a wide range of applications including designing effective GFs. The role of a GF is to learn the distribution of a class of signals, such that it is able to generate samples from that class. (Refer to Chapters 4 and 12 in [7] to learn more about using GFs in classification.) Modern GFs achieve this goal typically through employing trained neural networks. Variational auto-encoders (VAEs) [8] and generative adversarial nets (GANs) [9] are examples of methods used to train complex GFs. The success of such approaches in solving machine learning problems heavily relies on their ability to learn distributions of various complex signals, such as image and audio files. This success has encouraged researchers from other areas, such as compression, denoising and CS, to look into application of such methods as tools to capture signals' structure. Specifically, in CS, this idea was originally proposed in [10] and further pursued in [11].

Given a class of signals, $\mathcal{Q} \subset \mathbb{R}^n$, consider a corresponding trained GF $g : \mathcal{U}^k \to \mathbb{R}^n$, $\mathcal{U} \subset \mathbb{R}$. Assume that g is trained by enough samples from $\mathcal{Q}$, such that it is able to represent signals from $\mathcal{Q}$, possibly with some bounded loss. In this paper, we study the performance of an optimization-based CS recovery method that employs g as a mechanism to capture the structure of signals in $\mathcal{Q}$. We derive sharp bounds connecting the properties of function g (its dimensions, its error in representing the signals in $\mathcal{Q}$, and its smoothness level) to the performance of the resulting recovery method. We also study, both theoretically and empirically, the performance of an iterative CS recovery method based on projected gradient descent (PGD) that employs g to capture and enforce the source model (structure). We connect the number of measurements required by such a recovery method with the properties of function g.

B. Notations

Vectors are denoted by bold letters, such as $\mathbf{x}$ and $\mathbf{y}$. Sets are denoted by calligraphic letters, such as $\mathcal{A}$ and $\mathcal{B}$. For a set $\mathcal{A}$, $|\mathcal{A}|$ denotes its cardinality. For $x \in \mathbb{R}$ and $b \in \mathbb{N}^+$, $[x]_b$ denotes the b bit quantized version of x is defined as $[x]_b = 2^{-b}\lfloor 2^b x \rfloor$. For a set $\mathcal{A} \subset \mathbb{R}$ and $b \in \mathbb{N}^+$, let $\mathcal{A}_b$ denote the set where every member in $\mathcal{A}$ is quantized in b bits, i.e., $\mathcal{A}_b \triangleq \{[x]_b : x \in \mathcal{A}\}$.

C. Paper organization

Section II describes the problem of CS using GFs and states our main result on the performance of an optimization that employs a GF to capture the source structure. Section III describes an efficient algorithm based on PGD to approximate the solution of the mentioned optimization which is based on exhaustive search. Section IV reviews some related work in the literature. Section V presents our simulation results on the performance of the algorithm based on PGD. Section VI presents the proofs of the main results and Section VII concludes the paper.

II. Recovery using GFs

Consider a class of signals represented by a compact set $\mathcal{Q} \subset \mathbb{R}^n$. (For example, $\mathcal{Q}$ can be the set of images of human faces, or the set of MRI images of human brains.) Let function $g : \mathcal{U}^k \to \mathbb{R}^n$ denote a GF trained to represent signals in set $\mathcal{Q}$. (Throughout the paper, we assume that $\mathcal{U}$ is a bounded subset of $\mathbb{R}$.)

Definition 1. *Function $g : \mathcal{U}^k \to \mathbb{R}^n$ is said to cover set $\mathcal{Q}$ with distortion δ, if*

$$\sup_{\mathbf{x}\in\mathcal{Q}} \min_{\mathbf{u}\in\mathcal{U}^k} \frac{1}{\sqrt{n}}\|g(\mathbf{u}) - \mathbf{x}\| \leq \delta. \tag{1}$$

In other words, when function g covers set $\mathcal{Q}$ with distortion δ, it is able to represent all signals in $\mathcal{Q}$ with a mean squared error less than δ^2.

Consider the standard problem of CS, where instead of explicitly knowing the structure of signals in $\mathcal{Q}$, we have access to function g, which is known to well-represent signals in $\mathcal{Q}$. In this setup, signal $\mathbf{x} \in \mathcal{Q}$ is measured as $\mathbf{y} = A\mathbf{x} + \mathbf{z}$, where $A \in \mathbb{R}^{m\times n}$, $\mathbf{y} \in \mathbb{R}^m$ and $\mathbf{z} \in \mathbb{R}^m$ denote the sensing matrix, the measurement vector, and the measurement noise, respectively. The goal is to recover $\mathbf{x}$ from $\mathbf{y}$, typically with $m \ll n$, via using the function g to define the structure of signals in $\mathcal{Q}$.

To solve this problem, ideally, we need to find a signal that is i) compatible with the measurements $\mathbf{y}$, and ii) representable with function g. Hence, ignoring the computational complexity issues, we would like to solve the following optimization problem:

$$\hat{\mathbf{u}} = \operatorname{argmin}_{\mathbf{u}\in\mathcal{U}^k} \|Ag(\mathbf{u}) - \mathbf{y}\|, \tag{2}$$

After finding $\hat{\mathbf{u}}$, signal $\mathbf{x}$ can be estimated as

$$\hat{\mathbf{x}} = g(\hat{\mathbf{u}}). \tag{3}$$

The main goal of this section is to theoretically study the performance of this optimization-based recovery method. We derive bounds that establish a connection between the ambient dimension of the signal n, the parameters of the function g, and the number of measurements m.

To prove such theoretical results, we put some constraints on function g. More precisely, consider $\mathbf{x} \in \mathcal{Q}$ and let and $\mathbf{y} = A\mathbf{x}+\mathbf{z}$, where $A \in \mathbb{R}^{m\times n}$ and $\mathbf{z} \in \mathbb{R}^m$. Assume that

1) g covers $\mathcal{Q}$ with distortion δ, where $\delta \in (0, 1)$,
2) g is L-Lipschitz,
3) $\mathcal{U}$ is a bounded subset of $\mathbb{R}$.

Define $\hat{\mathbf{u}}$ and $\hat{\mathbf{x}}$ as in (2) and (3), respectively. The following theorem characterizes the connection between the properties of function g (input dimension m and Lipschitz constant L), the number of measurements (m) and the reconstruction distortion ($\|\hat{\mathbf{x}} - \mathbf{x}\|$).

Theorem 1. *Consider compact set $\mathcal{Q} \subset \mathbb{R}^n$ and GF $g : \mathcal{U}^k \to \mathbb{R}^n$ that covers $\mathcal{Q}$ with distortion δ. (Here, $\mathcal{U}$ is a compact subset of $\mathbb{R}$.) Consider $\mathbf{x} \in \mathcal{Q}$ and let $\mathbf{y} = A\mathbf{x}+\mathbf{z}$, where $A \in \mathbb{R}^{m\times n}$ and $\mathbf{z} \in \mathbb{R}^m$. Assume that the entries of A and $\mathbf{z}$ are i.i.d. $\mathcal{N}(0, \frac{1}{n})$ and i.i.d. $\mathcal{N}(0, \sigma^2)$, respectively. Define $\hat{\mathbf{u}}$ and $\hat{\mathbf{x}}$ as* (2) *and* (3)*. Set free parameters $\eta > 2$ and $\upsilon \in (0, 1)$, such that $\frac{1}{2} - \frac{\upsilon}{2} - \frac{1}{\eta} > 0$. Assume that $m \le n$, and*

$$m \ge \eta k. \tag{4}$$

Then, $\frac{1}{\sqrt{n}}\|\hat{\mathbf{x}} - \mathbf{x}\|$ is smaller than

$$\sqrt{6L\sigma}(\tfrac{2k}{m})^{\frac{1}{4}}\delta^{\frac{1}{2}-\frac{\upsilon}{2}-\frac{1}{\eta}} + 4\sigma\delta^{-\frac{2}{\eta}}\sqrt{\tfrac{k\ln\frac{1}{\delta}}{m}} + \alpha, \tag{5}$$

where $\alpha \triangleq 2\delta^{1-\frac{1}{\eta}} + \delta^{\frac{1}{2}-\frac{1}{\eta}}\sqrt{2\sigma} + 3L\delta^{1-\upsilon-\frac{1}{\eta}}\sqrt{\frac{k}{m}} + L\delta^{1-\upsilon}\sqrt{\frac{k}{n}} = o(\delta^{\frac{1}{2}-\frac{\upsilon}{2}-\frac{1}{\eta}})$, with a probability larger than

$$1 - \mathrm{e}^{-(\upsilon-\zeta)k\ln\frac{1}{\delta}} - \mathrm{e}^{-k\ln\frac{1}{\delta}} - 3\mathrm{e}^{-0.8m}, \tag{6}$$

where $\zeta = O(\frac{1}{\ln\frac{1}{\delta}})$.

The proof of Theorem 1 is presented in Section VI-A.

To better understand the implications of Theorem 1, the following corollary considers the case of noiseless measurements (i.e. $\sigma = 0$).

Corollary 1. *Consider the same setup as Theorem 1, where $\sigma = 0$, i.e., the measurements are noise-free. Set free parameters $\eta > 1$ and $\upsilon \in (0, 1)$, such that $1 - \upsilon - \frac{1}{\eta} > 0$. If $m \ge \eta k$, then with a probability larger than $1 - \mathrm{e}^{-(\upsilon-\zeta)k\ln\frac{1}{\delta}} - \mathrm{e}^{-0.8m}$,*

$$\tfrac{1}{\sqrt{n}}\|\hat{\mathbf{x}} - \mathbf{x}\| \le \tfrac{3L}{\sqrt{\eta}}\delta^{1-\upsilon-\frac{1}{\eta}} + \alpha, \tag{7}$$

where $\alpha = o(\delta^{1-\upsilon-\frac{1}{\eta}})$.

Proof. The proof is a straightforward application of the proof of Theorem 1. Note that since there is no measurement noise in this case, we will not get error terms that are $O(\delta^{\frac{1}{2}-\frac{\upsilon}{2}-\frac{1}{\eta}})$. Therefore, The condition on η and υ here has changed to $\eta > 1$ and $1-\upsilon-\frac{1}{\eta} > 0$. □

Remark 1. Consider a lossless GF for a given class of signals described by $\mathcal{Q}$, a compact subset of $\mathbb{R}^n$. That is, $\sup_{\mathbf{x}\in\mathcal{Q}} \min_{\mathbf{u}\in\mathcal{U}^k} \|\mathbf{x} - g(\mathbf{u})\| = 0$. In this case, $\delta = 0$. In such a scenario, Corollary 1 states that, essentially, $m > k$ measurements are sufficient for almost lossless recovery.

The optimization described in (2) was first proposed and analyzed in [10]. It was shown in [10] that $O(k \log L)$ measurements are sufficient for accurate recovery. However, in our results (Theorem 1 and Corollary 1), the number of measurements does not scale with L (Lipschitz constant) or δ and instead is proportional to k. This is consistent with our expectations as the Minkowski dimension of a class of signals that are generated by a GF with input dimension k is also k, and therefore, in the noiseless setting, we expect to be able to recover the signal from k noise-free measurements [12].

Remark 2. In the presence of Gaussian noise, first note that, unlike prior work, the error terms in Theorem 1 scale with the noise power (σ^2), rather than $\|\mathbf{z}\|$. Moreover, the dominant error term that does not disappear as δ converges to zero is $4\sigma\delta^{-\frac{2}{\eta}}\sqrt{\frac{k\ln\frac{1}{\delta}}{m}}$. To understand the role of this term, first note that, in the presence of Gaussian noise, due to the trade-off between bias and variance, it is not optimal to choose a model with δ close to zero. Instead, the optimal choice of δ would depend on the power of noise (σ), and as the noise power increases, models with larger values of δ will result in more accurate estimates. (Refer to Section V for numerical validation of this point.) Second, note that the mentioned error term scales with m as $O(\frac{1}{\sqrt{m}})$. This implies that, for any noise power σ^2 and any representation error δ, as the number of measurements m grows, the effect of this term vanishes as $O(\frac{1}{\sqrt{m}})$.

III. AE-PGD ALGORITHM

The optimization described in (2) is a challenging non-convex optimization. The GF g defined using an NN is a differentiable function. Therefore, one approach to solving $\min_{\mathbf{u}\in\mathcal{U}^k} \|Ag(\mathbf{u}) - \mathbf{y}\|$ is to apply the standard gradient descent (GD) algorithm [10]. However, since the problem is non-convex, there is no guarantee that the solution derived based on this approach is close to the optimal solution. Another approach is to note that

$\min_{\mathbf{u}\in\mathcal{U}^k}\|Ag(\mathbf{u})-\mathbf{y}\| \equiv \min_{\hat{\mathbf{x}}\in\{g(\mathbf{u}):\,\mathbf{u}\in\mathcal{U}^k\}}\|A\hat{\mathbf{x}}-\mathbf{y}\|$ and apply PGD as follows: For $t=0,1,\ldots$, let

$$\mathbf{s}^{t+1} = \hat{\mathbf{x}}^t + \mu A^T(\mathbf{y}-A\hat{\mathbf{x}}^t)$$
$$\mathbf{u}^{t+1} = \operatorname{argmin}_{\mathbf{u}\in\mathcal{U}^k}\|\mathbf{s}^{t+1}-g(\mathbf{u})\| \quad (8)$$
$$\hat{\mathbf{x}}^{t+1} = g(\mathbf{u}^{t+1}). \quad (9)$$

Still the described optimization is non-convex and therefore there is no guarantee that the algorithm will converge to the desired solution. The following theorem establishes this result and connects the number of measurements m, the representation error of the GF δ, the input dimension of the GF k, with the convergence performance of the PGD-based algorithm. Furthermore, it shows the robustness of this approach to additive white Gaussian noise.

Theorem 2. *Consider $\mathbf{x}\in\mathcal{Q}$, and $\mathbf{y}=A\mathbf{x}+\mathbf{z}$, where $\mathcal{Q}$ denotes a compact subset of $\mathbb{R}^n$ and $A\in\mathbb{R}^{m\times n}$. Here, $z_1,\ldots,z_m$ are i.i.d. $\mathcal{N}(0,\sigma^2)$. Assume that function $g:[0,1]^k\to\mathbb{R}^n$ is L-Lipschitz and satisfies (1), for some $\delta>0$. Define $\tilde{\mathbf{u}}$ and $\tilde{\mathbf{x}}$ as $\operatorname{argmin}_{\mathbf{u}\in\mathcal{U}^k}\|\mathbf{x}-g(\mathbf{u})\|$ and $\tilde{\mathbf{x}}=g(\tilde{\mathbf{u}})$, respectively. Choose free parameters $\alpha,\upsilon\in\mathbb{R}^+$ and define η, γ_1 and γ_2 as*

$$\eta \triangleq \tfrac{k}{n}(1+(\sqrt{\tfrac{n}{m}}+2)^2)L^2\delta^{2\alpha}, \quad (10)$$

$$\gamma_1 \triangleq (2+\sqrt{\tfrac{n}{m}})^2(L\delta^{\alpha}\sqrt{\tfrac{k}{n}}+1) \quad (11)$$

and

$$\gamma_2 \triangleq \sqrt{\tfrac{2k}{n}}(2+\sqrt{\tfrac{n}{m}}), \quad (12)$$

respectively. Assume that

$$m \geq 40(1+\alpha+\upsilon)k\log\frac{1}{\delta}. \quad (13)$$

Let $\mu=\frac{1}{m}$. For $t=0,1,\ldots$, define $(\mathbf{s}^{t+1},\mathbf{u}^{t+1},\hat{\mathbf{x}}^{t+1})$ as (9). Then, for every t, if $\frac{1}{\sqrt{n}}\|\hat{\mathbf{x}}^t-\tilde{\mathbf{x}}\|\geq\delta$, then, either $\frac{1}{\sqrt{n}}\|\hat{\mathbf{x}}^{t+1}-\tilde{\mathbf{x}}\|\leq\delta$, or

$$\tfrac{1}{\sqrt{n}}\|\tilde{\mathbf{x}}-\hat{\mathbf{x}}^{t+1}\| \leq \tfrac{0.9+\eta}{\sqrt{n}}\|\tilde{\mathbf{x}}-\hat{\mathbf{x}}^t\| + \left(\sqrt{\tfrac{6(1+\alpha)\left(\log\frac{1}{\delta}\right)k}{m}}+\gamma_2 L\delta^{\alpha}\right)\tfrac{\sigma}{\sqrt{n}}+\gamma_1\delta,$$

with a probability larger than $1-2^{-2k\upsilon\log\frac{1}{\delta}}-\mathrm{e}^{-\frac{m}{2}}-\mathrm{e}^{-0.1(1+\alpha)\left(\log\frac{1}{\delta}\right)k+2(\ln 2)k}-\mathrm{e}^{-0.15m}$.

Theorem 2 states that although the original optimization is not convex, having roughly more than $40k\log\frac{1}{\delta}$ measurements, the described PGD algorithm converges, even in the presence of additive white Gaussian noise.

In order to implement the proposed iterative method described in (9), the step that might seem challenging is the projection step, i.e., $\mathbf{u}^{t+1}=\operatorname{argmin}_{\mathbf{u}\in\mathcal{U}^k}\|\mathbf{s}^{t+1}-g(\mathbf{u})\|$. Since the cost function is differentiable, one can use GD to solve it [11]. However, since the cost is not convex, there is no guarantee that the solution will be close to the optimal. Moreover, using GD to solve this optimization, adds to the computational complexity of the problem. Therefore, instead, we consider training a separate neural network that approximates the solution to this optimization. Concatenating this neural network with the neural network that define g essentially yields an "auto-encoder" (AE) that maps a high-dimensional signal into low-dimensions, and then back to its original dimension. Using this perspective, the last two steps of the algorithm, basically pass $\mathbf{s}^{t+1}$ through an AE. (See Fig. 1.) We refer to the PGD-based algorithm where the projection is achieved by an AE as the AE-PGD method.

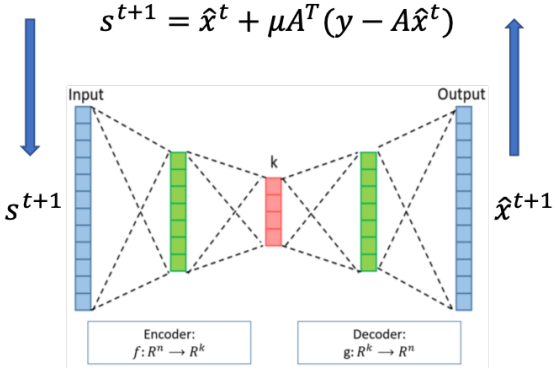

Fig. 1: AE-PGD CS recovery. The top equation is the gradient descent and the bottom plot shows the AE employed to perform the projection of the signal.

IV. Related work

Using NNs for CS has been an active area of research in recent years. (See [10], [11], [13]–[20] for a non-comprehensive list of such results.) Closely studying the literature in this area reveals that, interestingly, the role imagined for the NN to play is not shared among different approaches. While in some methods, NNs are directly trained to solve the inverse problem, in others, they are trained, independent of the CS recovery problem, as GFs that capture the source model. Our focus in this paper is on the latter type of methods where the role of the NN is to build a powerful GF that captures the source complex structure. Application of NN-based GFs to solve CS problems was first proposed in [10], which proved that roughly $O(kd\log n)$ measurements are enough for recovering the signal using the optimization described in (2). (d denotes the number of hidden layers.) The authors of [21] demonstrate that under strong assumptions on the generator function, the optimization problem described in (2) has a descent direction. In [11], an iterative algorithm based on PGD (similar to the one studied here) was proposed and

studied. Here, we derive sharp theoretical guarantees for both i) the exhaustive search method described in Section II and ii) the PGD-based algorithm. Our bounds directly connect the number of measurements with the properties of the GF, such as its input dimension and its representation quality. In both cases, we study the performance under additive white Gaussian noise as well.

Another related line of work is on using compression codes in designing efficient compression-based recovery methods [22]. The goal of such methods it to elevate the scope of structures used by CS algorithms to those used by compression codes. Such an optimization is similar to (3). However, the difference between these two approaches is that while a lossy compression code can be represented by a *discrete* set of codewords, a GF g has a *continuous* input $\mathcal{U}^k$.

V. Simulation results

To further study the performance of the AE-PGD recovery method, we examine its performance on three different datasets: i) the MNIST hand-written digits [23], ii) the chest X-ray images provided by NIH [24], and iii) facial images from the CelebA dataset [25]. The AE structure (2-layer encoder, and 2-layer decoder) (except the one reported in Section V-C) and the PGD algorithm are shown in Fig. 1. The implementations are performed in PyTorch using Nvidia 1080 Ti GPU. We use the average peak signal-to-noise ratio (PSNR) to evaluate the quality of the reconstructed images. All the codes can be found at [26].

Remark 3. While Theorem 2 proves the convergence of the AE-PGD method for $\mu = \frac{1}{m}$, in our simulations, we observed that changing the step size could in fact improve the performance. Therefore, using cross validation, in each setup, we optimize the value of μ. Potentially, one could further improve the performance by optimizing the step size at each iteration. However, this comes at a great computational complexity.

A. MNIST

In our first set of experiments, we study the MNIST dataset. Each image in this dataset consists of 28×28 pixels. We use $35,000$ images for training and 300 images for testing. We consider an AE with fully-connected layers with sigmoid activation functions, such that the hidden layers of the encoder and the decoder each consists of $1,500$ hidden nodes. We set the size of the output layer of the encoder and the input layer of the GF (k) to 100. The step size μ is set to 0.7.

Fig. 2 compares the performance of the AE-PGD recovery with Lasso [27] and BM3D-AMP [28] under different sampling rates m/n, in both noise-free and noisy settings (middle plot corresponds to signal-to-noise-ratio (SNR) of 10 dB). It can be seen that in the *noise free* case, when the sampling rate is low (e.g. 0.1 and 0.05), the AE-PGD method outperforms the other methods. When the sampling rate is higher (e.g. 0.2 and 0.3), BM3D-AMP achieves the best performance. In the *noisy* case, although BM3D-AMP still has the highest PSNR at high sample rates, its performance drops significantly. Some reconstructed images by the three algorithms (under noise free case) compared with the ground truth are shown in the right plot in Fig. 2.

B. X-ray Images

We next explore the performance of the AE-PGD method on chest X-ray images [24]. In this dataset, each image is of size 128×128. We use $35,000$ training images and 100 testing images. We compare the performance of the AE-PGD method with BM3D-AMP and Lasso-DCT. This time, we consider two different NNs calling the results NN1-PGD and NN2-PGD. In both cases, μ is set to 0.7. Both NNs are structured as before with different number of nodes as follows:

i) NN1, $k = 2000$ and there are 5000 hidden nodes in the first layer of the encoder and the second layer of the decoder;
ii) NN2, $k = 3000$, there are hidden 8000 hidden nodes in the first layer of the encoder and the second layer of the decoder.

In this case, all the activation functions, except those at the final layer of the decoder, are set to rectified linear unit (ReLU) function. The activation functions of the final layer are set as the sigmoid function.

Fig. 3 shows the average PSNR on test images in both noiseless (left) and noisy (middle) settings, again at SNR = 10 dB. The capacity of each NN refers to the average representation error corresponding to each NN. Clearly the performance of the AE-PGD cannot exceed the capacity of the NN it employs. It can be observed that for both NNs, the AE-PGD method in fact achieves the capacity. This implies that to improve the performance of the AE-PGD method in high SNR regimes, one needs to design a NN with higher capacity, i.e., lower representation error.

Recall that Theorem 2 proves that the AE-PGD method converges, given enough measurement samples. To better understand the convergence behavior of the algorithm, Fig. 4 shows the average number of iterations of the NN1-PGD and NN2-PGD methods, as a function of sampling rate. The step size is fixed to 0.7 as before. It can be seen that in both cases typically no more than 20 iterations are required.

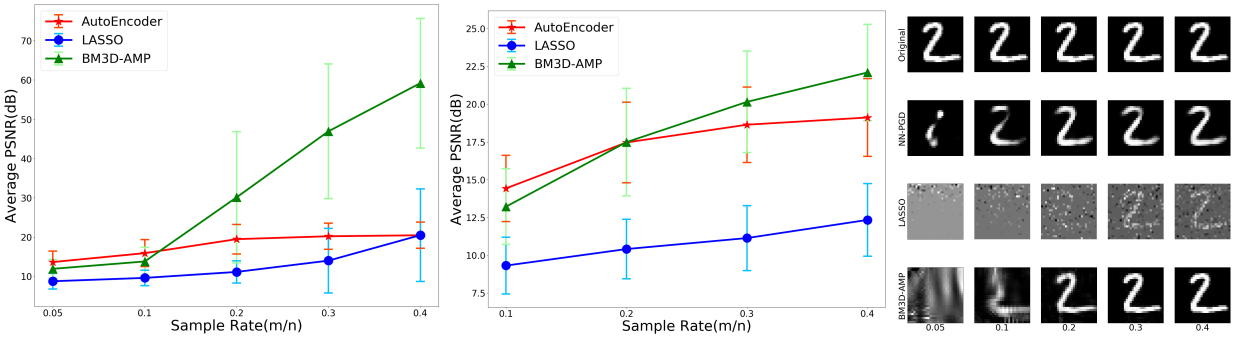

Fig. 2: Comparing Lasso, BM3D-AMP and the proposed auto-encoder based inversion in the *noise free* case (left) and *noisy* case (middle with SNR = 10dB). Right: reconstructed images at different sampling rate in the noise free case.

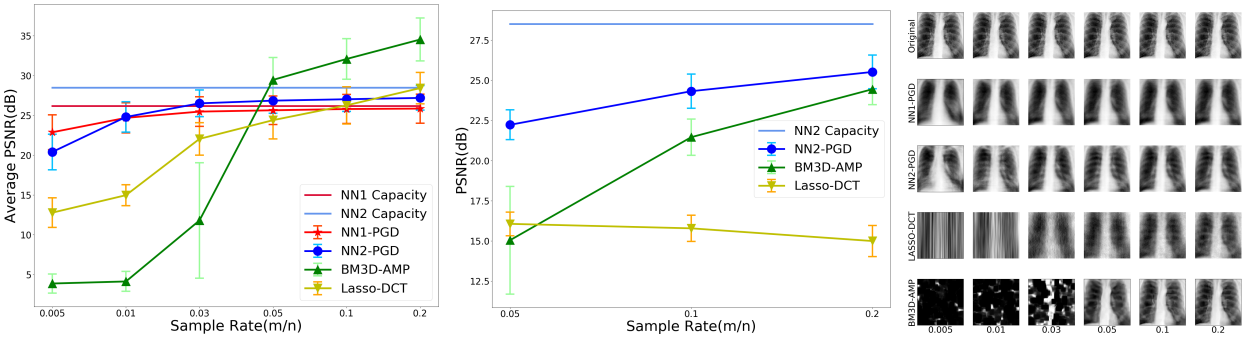

Fig. 3: PSNR of the reconstructed X-ray images under noise free (left) and noisy (middle with SNR = 10dB) cases and some example images (right).

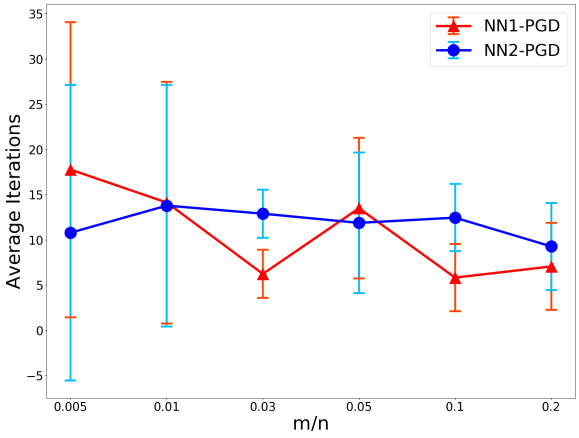

Fig. 4: Average number of iterations versus sampling rate for NN1-PGD and NN2-PGD

C. Parallel Block-wise Neural Networks

As shown in the previous section, the bottleneck in achieving high performance at higher sampling rates seem to be the accuracy of the representation error of the AE. In other words, to improve the performance of the AE-PGD method at higher sampling rates, we need to train AEs with representation error. On the other hand, the size of chest X-ray images suggests that to achieve this goal one needs to train larger AEs. Given our computational limitations, for example due to our GPU memory, we next design a block-wise AE neural network, which breaks images into smaller blocks as follows. Again, the goal is to train a NN with higher capacity. We crop each image into four smaller 74×74 images. Then we train a separate AE consisting of fours parts working in parallel. Each part is an AE with the same structure as the one shown in Fig. 1, with $k = 3000$, and 8000 hidden nodes for the other hidden layers. We allow some overlap between the image segments to avoid any blocking effects. For the pixels that are represented by more than one block, we take the average. As before, the step size is set to 0.7.

In Fig. 5, we compare the performance of the block-wise AE (referred to as 4NNs-PGD) with that of NN2-PGD (described in the previous section). It can be observed that, when the sampling rate is larger than 0.03, 4NNs-PGD achieves a better performance than NN2-PGD. On the other hand, at lower sampling rates, the network with a lower capacity (i.e., NN2) outperforms the 4NN network. We also saw earlier that at lower

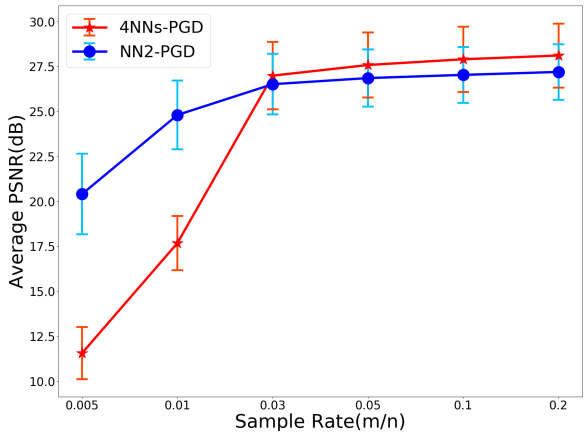

Fig. 5: Average PSNR of block-wise NN (4NNS-PGD) and NN2-PGD.

sampling rates, the NN2 network outperforms BM3D-AMP. All these results show the power of AEs, as they can be designed to operate at different accuracies. In summary, the simulation results suggest that as the CS sampling rate grows, to achieve the best performance, one needs to adjust the accuracy of the employed AE accordingly.

D. U-net Refinement

As shown in the previous section, training high-accuracy AEs is key to improving the performance of the AE-PGD algorithm at higher sampling rates. Instead of directly improving the performance of the AE, in this section we explore a detour strategy as follows: We train a U-Net [29] as a refinement function to improve the reconstructed image quality. To train the U-Net, we first train an AE (As described earlier) and then pass the original training dataset through the trained AE to generate a new training dataset for U-Net. Then we use the original images and their reconstructions of the AE to train the U-Net such that the output is close to the original images. In other words, the U-NET receives the output of the AE and is expected to regenerate the input of the AE, as much as possible. After training the U-Net, we first use the PGD-AE method to find $\hat{\mathbf{x}}$ as before, and then refine it by passing it through the trained U-Net. (The U-Net structure is shown in Fig. 6.)

Fig. 7 compares the performance of the AE-PGD with and without U-NET refinement. Here, the AE is the blocked AE referred to as 4NN earlier. It can be observed that this refinement step improves the performance of the 4NNs-PGD method significantly at higher sampling rates, e.g., almost 3 dB at sampling rate 0.2. However, the achieved performance is still below that of BM3D-AMP when sampling rate is high.

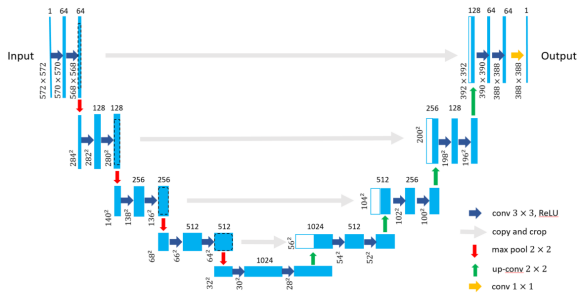

Fig. 6: U-Net to refine the reconstruction results of the proposed PGD algorithm.

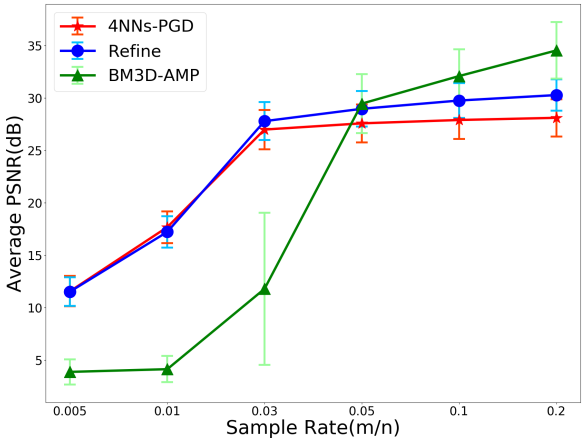

Fig. 7: Average PSNR of the two stage NN (AE + U-Net).

It is worth noting that since the images in this dataset are rather noisy, and the U-Net seems to perform some denoising of the original images. On the other hand, the PSNR is calculated by comparing a reconstructed image with the original one. Therefore, PSNR might not be an optimal measure to compare the performance of the algorithms. Inspecting the recovered images shown in Fig. 8 reveals that at high sampling rates, BM3D-AMP reconstructs images that are very close to the original ones, by even recovering the noise. But the AE-PGD method with refinement reconstructs less-noisy images, which arguably include almost all the details of the original images.

E. Facial Images

The small size of the digit images and the noisy nature of the X-ray images potentially pose as some limitations into the performance of any recovery method. As the final example, we test the AE-PGD method on some clean facial images from the CelebA dataset [25]. This time, each image consists of three 64×64 frames. We use $50,000$ images for training, and 100 images for test. We compare the performance of the NN-PGD method (with

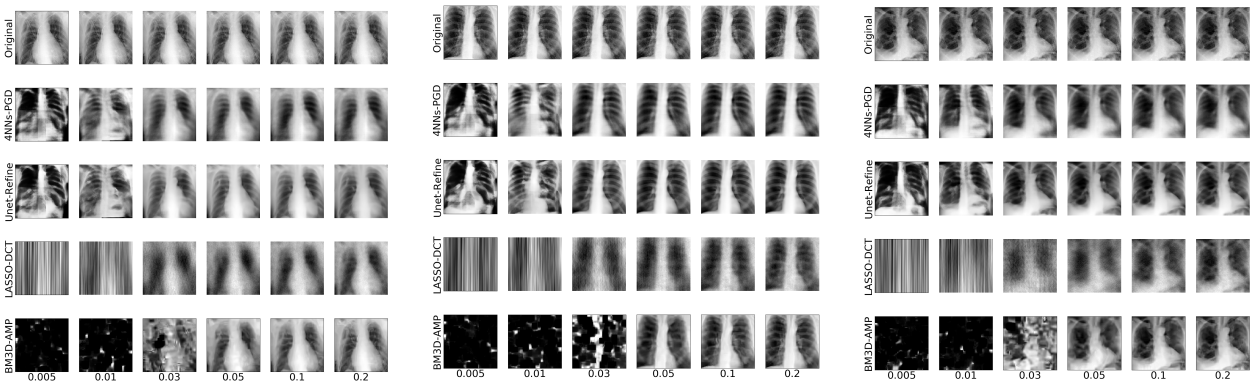

Fig. 8: Exemplar reconstructed X-ray images by different algorithms compared with the original noisy images.

and without U-Net refinement) with that of BM3D-AMP. For the AE, the input and output dimensions are set to $3 \times 64 \times 64$ and $k = 3000$. The number of hidden nodes in the encoder and the decoder are set to $12,000$. All nodes in the AE are set to use the sigmoid activation function. As before, the U-Net is trained by using the images that are passed through the AE.

Fig. 9 show the performance of i) the AE-PGD method without refinement, and ii) the BM3D-AMP method. For the AE-PGD, the step size μ is set to $(0.2, 0.5, 0.7, 0.9)$ at sampling rates $(0.01, 0.05, 0.1, 0.2)$, respectively. As before, the BM3D-AMP outperforms the AE-PGD method at higher sampling rates. The figure does not show the performance of the U-Net refinement as it has negligible effect in terms of PSNR. However, while the refinement algorithm does not improve the performance much in terms of PSNR, as shown in Fig. 10, it makes a considerable visual impact on the quality of the recovered images.

Carefully inspecting the figures shown in Fig. 10 simultaneously reveals some the strengths and some of the weaknesses of the AE-PGD method. The AE is essentially trained on human faces and therefore is capable of representing figures. On the other hand, its ability to capture the other details such as the background or accessories is limited. Therefore, comparing the images recovered by the AE-PGD method with those recovered by the BM3D-AMP reveals that while the former ones have better visual qualities in terms of the faces themselves, still the overall PSNR of the latter group is better as they have a more uniform performance across the whole figure.

VI. Proofs

The following lemma from [30] on the concentration of Chi-squared random variables is used in the proof.

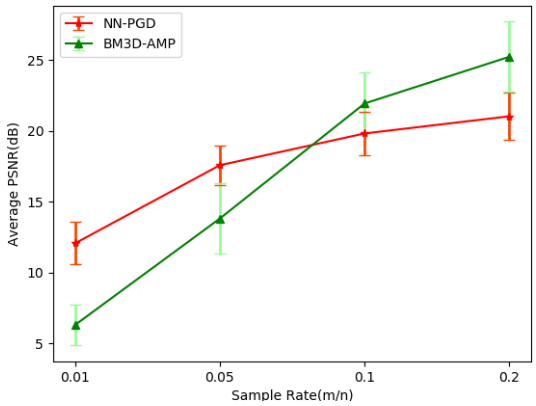

Fig. 9: Average PSNR of 100 testing images in the CelebA dataset using different algorithms.

Lemma 1 (Chi-squared concentration)**.** *Assume that $U_1, \ldots, U_n$ are i.i.d. $\mathcal{N}(0,1)$. For any $\tau \geq 0$ we have*

$$\mathrm{P}\left(\sum_{i=1}^m U_i^2 > m(1+\tau)\right) \leq \mathrm{e}^{-\frac{m}{2}(\tau - \ln(1+\tau))}, \quad (14)$$

and for $\tau \in (0,1)$,

$$\mathrm{P}\left(\sum_{i=1}^m U_i^2 < m(1-\tau)\right) \leq \mathrm{e}^{\frac{m}{2}(\tau + \ln(1-\tau))}. \quad (15)$$

Also, the following lemma from [31] are used in the proof of Theorem 2.

Lemma 2. *Consider $\mathbf{u} \in \mathbb{R}^n$ and $\mathbf{v} \in \mathbb{R}^n$ such that $\|\mathbf{u}\| = \|\mathbf{v}\| = 1$. Let $\alpha \triangleq \langle \mathbf{u}, \mathbf{v} \rangle$. Consider matrix $A \in \mathbb{R}^{m \times n}$ with i.i.d. standard normal entries. Then, for any $\tau > 0$, $\mathrm{P}\left(\langle \mathbf{u}, \mathbf{v} \rangle - \frac{1}{m}\langle A\mathbf{u}, A\mathbf{v} \rangle \geq \tau\right) \leq \mathrm{e}^{m((\alpha-\tau)s) - \frac{m}{2}\ln((1+s\alpha)^2 - s^2)}$, where $s > 0$ is a free parameter smaller than $\frac{1}{1-\alpha}$. Specifically, for $\tau = 0.45$,*

$$\mathrm{P}\left(\langle \mathbf{u}, \mathbf{v} \rangle - \frac{1}{m}\langle A\mathbf{u}, A\mathbf{v} \rangle \geq 0.45\right) \leq 2^{-0.05m}. \quad (16)$$

Lemma 3. *Consider $\mathbf{u}$ and $\mathbf{v}$, where $u_1, \ldots, u_n, v_1, \ldots, v_n$ are i.i.d. $\mathcal{N}(0,1)$. Then the distribution of $\langle \mathbf{u}, \mathbf{v} \rangle = \sum_{i=1}^n u_i v_i$ is the same as*

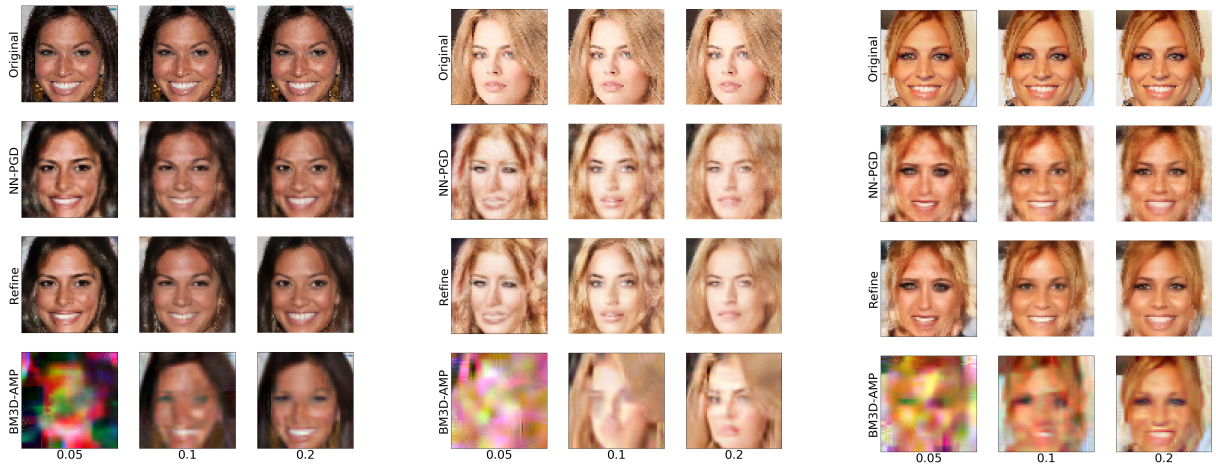

Fig. 10: Reconstructed facial images by different algorithms at various sampling rates.

the distribution of $\|\mathbf{u}\|G$*, where* $G \sim \mathcal{N}(0,1)$ *is independent of* $\|\mathbf{u}\|$.

A. Proof of Theorem 1

Define $\tilde{\mathbf{u}}$ and $\tilde{\mathbf{x}}$ as (2) and (3), respectively. Since $\hat{\mathbf{u}}$ is the minimizer of $\|Ag(\mathbf{u})-\mathbf{y}\|$, over all $\mathbf{u}\in\mathcal{U}^k$, we have $\|Ag(\hat{\mathbf{u}})-\mathbf{y}\| \le \|Ag(\tilde{\mathbf{u}})-\mathbf{y}\|$. Moreover, by the triangle inequality,

$$\|Ag([\hat{\mathbf{u}}]_b)-\mathbf{y}\| \le \|Ag(\hat{\mathbf{u}})-\mathbf{y}\| + \|Ag(\hat{\mathbf{u}})-Ag([\hat{\mathbf{u}}]_b)\|. \tag{17}$$

Recall that $\mathbf{y}=A\mathbf{x}+\mathbf{z}$. Therefore,

$$\begin{aligned}
&\|A(g([\hat{\mathbf{u}}]_b)-\mathbf{x})-\mathbf{z}\| \\
&\le \|A(g(\tilde{\mathbf{u}})-\mathbf{x})-\mathbf{z}\| + \|A(g(\hat{\mathbf{u}})-g([\hat{\mathbf{u}}]_b))\| \\
&\le \|A(g(\tilde{\mathbf{u}})-\mathbf{x})-\mathbf{z}\| + \sigma_{\max}(A)L\|\hat{\mathbf{u}}-[\hat{\mathbf{u}}]_b\| \\
&\le \|A(g(\tilde{\mathbf{u}})-\mathbf{x})-\mathbf{z}\| + 2^{-b}\sqrt{k}\sigma_{\max}(A)L. \qquad (18)
\end{aligned}$$

Define $\mathbf{e}_1$ and $\mathbf{e}_2$ as $\mathbf{e}_1=g(\tilde{\mathbf{u}})-\mathbf{x}$ and $\mathbf{e}_2=g([\hat{\mathbf{u}}]_b)-\mathbf{x}$, respectively. Define their normalized versions as $\bar{\mathbf{e}}_i=\mathbf{e}_i/\|\mathbf{e}_i\|$, $i=1,2$. Given $\tau_1>0$ and $\tau_2\in(0,1)$, define events $\mathcal{E}_1$ and $\mathcal{E}_2$ as $\mathcal{E}_1 \triangleq \{\|A\bar{\mathbf{e}}_1\| \le \sqrt{\frac{m}{n}}(1+\tau_1)\}$, and $\mathcal{E}_2 \triangleq= \{\|A(g(\mathbf{u})-\mathbf{x})\| \ge \sqrt{\frac{m}{n}}(1-\tau_2)\|g(\mathbf{u})-\mathbf{x}\| : \forall \mathbf{u}\in\mathcal{U}_b^k\}$, respectively. Furthermore, given $\tau_z>0$, $\tau_3>0$ and $\tau_4>0$, define events $\mathcal{E}_a$, $\mathcal{E}_z$, $\mathcal{E}_3$ and $\mathcal{E}_4$ as $\mathcal{E}_a \triangleq \{\sigma_{\max}(A)\le 1+2\sqrt{\frac{m}{n}}\}$, $\mathcal{E}_z \triangleq \{\|\mathbf{z}\| \le \sigma\sqrt{m(1+\tau_z)}\}$, $\mathcal{E}_3 \triangleq \{|\langle A\bar{\mathbf{e}}_1,\mathbf{z}\rangle| \le \sigma\tau_3\sqrt{\frac{m}{n}}\}$, and $\mathcal{E}_4 \triangleq \{|\langle A\bar{\mathbf{e}},\mathbf{z}\rangle| \le \sigma\tau_4\sqrt{\frac{m}{n}} : \bar{\mathbf{e}}=\frac{g(\mathbf{u})-\mathbf{x}}{\|g(\mathbf{u})-\mathbf{x}\|}, \mathbf{u}\in\mathcal{U}_b^k\}$, respectively. From Lemma 1, $\mathrm{P}(\mathcal{E}_1^c) \le \mathrm{e}^{-\frac{m}{2}(\tau_1-\ln(1+\tau_1))}$, and, for a fixed $\mathbf{u}\in\mathcal{U}_b^n$, with a probability larger than $\mathrm{e}^{\frac{m}{2}(\tau_2+\ln(1-\tau_2))}$,

$$\|A(g(\mathbf{u})-\mathbf{x})\| \ge m(1-\tau_2)\|g(\mathbf{u})-\mathbf{x}\|. \tag{19}$$

Therefore, applying the union bound, it follows that

$$\mathrm{P}(\mathcal{E}_2^c) \le |\mathcal{U}_b|^k \mathrm{e}^{\frac{m}{2}(\tau_2+\ln(1-\tau_2))}.$$

Given a unit-norm vector $\bar{\mathbf{e}}\in\mathbb{R}^n$, $\sqrt{n}A\bar{\mathbf{e}}$ is a random vector in $\mathbb{R}^m$ with i.i.d. $\mathcal{N}(0,1)$ entries. Therefore, according to Lemma 3, $\sqrt{n}\langle A\bar{\mathbf{e}},\mathbf{z}\rangle$ has the same distribution as $\|\mathbf{z}\|G$, where $G\sim\mathcal{N}(0,1)$ is independent of $\|\mathbf{z}\|$. On the other hand, using the law of total probability, $\mathrm{P}(\mathcal{E}_3\cap\mathcal{E}_4) \ge 1-\mathrm{P}(\mathcal{E}_z^c)-\mathrm{P}(\mathcal{E}_3,\mathcal{E}_z)-\mathrm{P}(\mathcal{E}_4,\mathcal{E}_z)$, $i\in\{3,4\}$. Also

$$\mathrm{P}(\mathcal{E}_z^c) \le \mathrm{e}^{-\frac{m}{2}(\tau_z-\ln(1+\tau_z))},$$

and

$$\begin{aligned}
\mathrm{P}(\mathcal{E}_3^c,\mathcal{E}_z) &= \mathrm{P}(|G|\|\mathbf{z}\| > \tau_3\sigma\sqrt{m}, \|\mathbf{z}\| \le \sigma\sqrt{m(1+\tau_z)}) \\
&\le \mathrm{P}\left(|G| > \tfrac{\tau_3}{\sqrt{1+\tau_z}}\right) \le 2\mathrm{e}^{-\frac{\tau_3^2}{1+\tau_z}}.
\end{aligned}$$

Similarly, using the union bound, $\mathrm{P}(\mathcal{E}_4^c,\mathcal{E}_z) \le 2|\mathcal{U}_b|^k\mathrm{e}^{-\frac{\tau_4^2}{1+\tau_z}}$.

Conditioned on $\mathcal{E}_a$, $2^{-b}\sqrt{k}\sigma_{\max}(A)L\le\Delta$, where

$$\Delta \triangleq 2^{-b}\sqrt{k}L\left(1+2\sqrt{\frac{m}{n}}\right), \tag{20}$$

Therefore, conditioned on $\mathcal{E}_a$, raising both sides of (18) to power two and cancelling the common $\|\mathbf{z}\|^2$ term, it follows that $\|A\mathbf{e}_2\|^2-2\langle A\mathbf{e}_2,\mathbf{z}\rangle \le \|A\mathbf{e}_1\|^2-2\langle A\mathbf{e}_1,\mathbf{z}\rangle+2\Delta\|A\mathbf{e}_1+\mathbf{z}\|+\Delta^2 \le \|A\mathbf{e}_1\|^2-2\langle A\mathbf{e}_1,\mathbf{z}\rangle+2\Delta(\|A\mathbf{e}_1\|+\|\mathbf{z}\|)+\Delta^2$, where the last step follows from the triangle inequality. Therefore,

$$\begin{aligned}
\|\mathbf{e}_2\|^2\|A\bar{\mathbf{e}}_2\|^2 &\le \|\mathbf{e}_1\|^2\|A\bar{\mathbf{e}}_1\|^2 + 2\|\mathbf{e}_1\||\langle A\bar{\mathbf{e}}_1,\mathbf{z}\rangle| \\
&+ 2\|\mathbf{e}_2\||\langle A\bar{\mathbf{e}}_2,\mathbf{z}\rangle| + 2\Delta(\|\mathbf{e}_1\|\|A\bar{\mathbf{e}}_1\|+\|\mathbf{z}\|)+\Delta^2. \qquad (21)
\end{aligned}$$

Conditioned on $\mathcal{E}_1\cap\mathcal{E}_2\cap\mathcal{E}_3\cap\mathcal{E}_4$, noting that $\frac{1}{\sqrt{n}}\|\mathbf{e}_1\|\le\delta$, it follows from (21) that

$$\|\mathbf{e}_2\|^2 m(1-\tau_2) - 2\|\mathbf{e}_2\|\sigma\tau_4\sqrt{mn} - n(\gamma_1+\gamma_2) \le 0, \tag{22}$$

where

$$\gamma_1 \triangleq (1+\tau_1)\delta^2 m + 2\sigma\tau_3\delta\sqrt{m}, \tag{23}$$

and

$$\gamma_2 \triangleq \Delta^2 + 2\Delta\sqrt{m}\Big(\sigma\sqrt{1+\tau_z} + \delta\sqrt{1+\tau_1}\Big). \tag{24}$$

Since (22) is a second order equation, (30) implies that $\|\mathbf{e}_2\|$ should be smaller than the largest root of this equation. Noting that $\sqrt{a^2+b^2+c^2} \leq |a|+|b|+|c|$, for all a, b, and c in $\mathbb{R}$, it follows that

$$\tfrac{1}{\sqrt{n}}\|\hat{\mathbf{x}}_b - \mathbf{x}\| \leq \tfrac{1}{\sqrt{m}}\Big(\tfrac{2\sigma\tau_4}{1-\tau_2} + \sqrt{\tfrac{\gamma_1}{1-\tau_2}} + \sqrt{\tfrac{\gamma_2}{1-\tau_2}}\Big). \tag{25}$$

To finish the proof, we need to set the free parameters appropriately, such that the error probabilities converge to zero.

- Set $b = \lceil (1-\upsilon)\log\frac{1}{\delta}\rceil$.
- Let $\tau_1 = 3$. Therefore, $\mathrm{P}(\mathcal{E}_1^c) \leq \mathrm{e}^{-\frac{m}{2}(3-\ln 4)} \leq \mathrm{e}^{-0.8m}$.
- Let $\tau_2 = 1-\delta^{\frac{2}{\eta}}$. Since $\mathcal{U}$ is a compact set, there exist integer numbers a_1 and a_2, such that $\mathcal{U} \subseteq [a_1, a_2]$. Therefore, $|\mathcal{U}_b| \leq a 2^{-b}$, where $a \triangleq a_2 - a_1$. Therefore, since, for $\tau_2 \in (0,1)$, $(\tau_2 + \ln(1-\tau_2)) \leq 0$ and $m \geq \eta k$ by assumption, it follows that

$$\begin{aligned} &k\ln|\mathcal{U}_b| + \frac{m}{2}(\tau_2 + \ln(1-\tau_2)) \\ &\leq k(\ln a + b\ln 2) + \frac{\eta k}{2}(\tau_2 + \ln(1-\tau_2)) \\ &\leq k(\ln a - (1-\upsilon)\ln\delta) + \frac{\eta k}{2}(\tau_2 + \ln(1-\tau_2)), \end{aligned} \tag{26}$$

  where the last line follows because $b = \lceil (1-\upsilon)\log\frac{1}{\delta}\rceil$ and hence $b\ln 2 \leq (1-\upsilon)\log\frac{1}{\delta}\ln 2 = -(1-\upsilon)\ln\delta$. Therefore, from (26), inserting the value of τ_2, we have

$$\begin{aligned} &k\ln|\mathcal{U}_b| + \frac{m}{2}(\tau_2 + \ln(1-\tau_2)) \\ &\leq k(\ln a - (1-\upsilon)\ln\delta) + \frac{\eta k}{2}(1-\delta^{\frac{2}{\eta}} + \frac{2}{\eta}\ln\delta) \\ &= -k(\upsilon-\zeta)\ln\frac{1}{\delta}, \end{aligned} \tag{27}$$

  where

$$\zeta = \tfrac{\ln a + \frac{\eta}{2}(1-\delta^{\frac{2}{\eta}})}{\ln\frac{1}{\delta}}. \tag{28}$$

  Note that ζ only depend on a, η and δ and $\zeta = O(1/\ln\frac{1}{\delta})$. Therefore, $\mathrm{P}(\mathcal{E}_2) \leq \mathrm{e}^{-(\upsilon-\zeta)k\ln\frac{1}{\delta}}$.
- Set $\tau_z = 1$. Then, $\mathrm{P}(\mathcal{E}_z^c) \leq \mathrm{e}^{-\frac{m}{2}(1-\ln 2)} \leq \mathrm{e}^{-0.15m}$.
- Set $\tau_3 = \sqrt{m}$. As proved earlier, $\mathrm{P}(\mathcal{E}_3, \mathcal{E}_z^c) \leq 2\mathrm{e}^{-\frac{\tau_3^2}{1+\tau_z}}$. Hence, for $\tau_3 = \sqrt{m}$ and $\tau_z = 1$, $\mathrm{P}(\mathcal{E}_3, \mathcal{E}_z^c) \leq \mathrm{e}^{-\frac{m}{2}}$.
- Set $\tau_4 = 2\sqrt{k\ln\frac{1}{\delta}}$. We need to set τ_4 such that $|\mathcal{U}_b|^k \mathrm{e}^{-\frac{\tau_4^2}{1+\tau_z}}$ converges to zero, as the dimensions of the problems grow. Note that $\ln(|\mathcal{U}_b|^k \mathrm{e}^{-\frac{\tau_4^2}{1+\tau_z}}) = kb\ln 2 - \frac{1}{2}\tau_4^2 = k\lceil(1-\upsilon)\log\frac{1}{\delta}\rceil\ln 2 - \frac{1}{2}\tau_4^2 \leq k\ln\frac{1}{\delta} - \frac{1}{2}\tau_4^2$. Setting $\tau_4 = 2\sqrt{k\ln\frac{1}{\delta}}$, it follows that $\ln(|\mathcal{U}_b|^k \mathrm{e}^{-\frac{\tau_4^2}{1+\tau_z}}) \leq -k\ln\frac{1}{\delta}$.

For the selected values of the parameters, we have

$$\gamma_1 = 2(2\delta+\sigma)\delta m, \tag{29}$$

and

$$\gamma_2 = \Delta^2 + 2\Delta\sqrt{m}(\sigma\sqrt{2} + 2\delta). \tag{30}$$

But, since by assumption, $m \leq n$, $\Delta \leq 32^{-b}\sqrt{k}L \leq 3\sqrt{k}L\delta^{1-\upsilon}$. Therefore,

$$\gamma_2 \leq 9kL^2\delta^{2-2\upsilon} + 6\sqrt{km}L\delta^{1-\upsilon}(\sigma\sqrt{2}+2\delta).$$

In summary, combining the bounds on γ_1 and γ_2 with (25), it follows that

$$\begin{aligned} &\tfrac{1}{\sqrt{n}}\|\hat{\mathbf{x}}_b - \mathbf{x}\| \leq 4\sigma\delta^{-\frac{2}{\eta}}\sqrt{\tfrac{k\ln\frac{1}{\delta}}{m}} + \delta^{\frac{1}{2}-\frac{1}{\eta}}\sqrt{2(2\delta+\sigma)} \\ &+\delta^{\frac{1}{2}-\frac{\upsilon}{2}-\frac{1}{\eta}}\sqrt{9(\tfrac{k}{m})L^2\delta^{1-\upsilon} + 6\sqrt{\tfrac{k}{m}}L(\sigma\sqrt{2}+2\delta)}. \end{aligned} \tag{31}$$

Finally, to finish the proof, note that $\|\hat{\mathbf{x}} - \mathbf{x}\| \leq \|\hat{\mathbf{x}} - \hat{\mathbf{x}}_b\| + \|\hat{\mathbf{x}}_b - \mathbf{x}\| \leq L\|\mathbf{u} - [\mathbf{u}]_b\| + \|\hat{\mathbf{x}}_b - \mathbf{x}\| \leq L2^{-b}\sqrt{k} + \|\hat{\mathbf{x}}_b - \mathbf{x}\|$. Therefore, using $\sqrt{a^2+b^2+c^2} \leq |a|+|b|+|c|$, it follows that $\frac{1}{\sqrt{n}}\|\hat{\mathbf{x}} - \mathbf{x}\|$ is smaller than

$$\begin{aligned} &\tfrac{1}{\sqrt{n}}\|\hat{\mathbf{x}} - \mathbf{x}\| 4\sigma\delta^{-\frac{2}{\eta}}\sqrt{\tfrac{k\ln\frac{1}{\delta}}{m}} + 2\delta^{1-\frac{1}{\eta}} + \delta^{\frac{1}{2}-\frac{1}{\eta}}\sqrt{2\sigma} \\ &+3L\delta^{1-\upsilon-\frac{1}{\eta}}\sqrt{\tfrac{k}{m}} + \sqrt{6L\sigma}(\tfrac{2k}{m})^{\frac{1}{4}}\delta^{\frac{1}{2}-\frac{\upsilon}{2}-\frac{1}{\eta}} \\ &+2\sqrt{3}(\tfrac{k}{m})^{\frac{1}{4}}\delta^{1-\frac{\upsilon}{2}-\frac{1}{\eta}} + L\delta^{1-\upsilon}\sqrt{\tfrac{k}{n}}, \end{aligned} \tag{32}$$

which, defining $\alpha \triangleq 2\delta^{1-\frac{1}{\eta}} + \delta^{\frac{1}{2}-\frac{1}{\eta}}\sqrt{2\sigma} + 3L\delta^{1-\upsilon-\frac{1}{\eta}}\sqrt{\frac{k}{m}} + L\delta^{1-\upsilon}\sqrt{\frac{k}{n}} = o(\delta^{\frac{1}{2}-\frac{\upsilon}{2}-\frac{1}{\eta}})$, concludes the proof.

B. Proof of Theorem 2

Recall that $\hat{\mathbf{u}} = \mathrm{argmin}_{\mathbf{u}\in\mathcal{U}^k}\|g(\mathbf{u}) - \mathbf{x}\|$ and $\hat{\mathbf{x}} = g(\hat{\mathbf{u}})$. Since $\hat{\mathbf{x}}^{t+1} = \mathrm{argmin}_{u^k\in\mathcal{U}^k}\|\mathbf{s}^{t+1} - g(\mathbf{u})\|$,

$$\|\mathbf{s}^{t+1} - \hat{\mathbf{x}}^{t+1}\| \leq \|\mathbf{s}^{t+1} - \hat{\mathbf{x}}\|.$$

But $\|\mathbf{s}^{t+1} - \hat{\mathbf{x}}^{t+1}\|^2 = \|\mathbf{s}^{t+1} - \hat{\mathbf{x}} + \hat{\mathbf{x}} - \hat{\mathbf{x}}^{t+1}\|^2 = \|\mathbf{s}^{t+1} - \hat{\mathbf{x}}\|^2 + \|\hat{\mathbf{x}} - \hat{\mathbf{x}}^{t+1}\|^2 + 2\langle\mathbf{s}^{t+1} - \hat{\mathbf{x}}, \hat{\mathbf{x}} - \hat{\mathbf{x}}^{t+1}\rangle$. Therefore,

$$\begin{aligned} &\|\hat{\mathbf{x}} - \hat{\mathbf{x}}^{t+1}\|^2 \leq 2\langle\hat{\mathbf{x}} - \mathbf{s}^{t+1}, \hat{\mathbf{x}} - \hat{\mathbf{x}}^{t+1}\rangle \\ &= 2\langle\hat{\mathbf{x}} - \hat{\mathbf{x}}^t, \hat{\mathbf{x}} - \hat{\mathbf{x}}^{t+1}\rangle - 2\mu\langle A(\hat{\mathbf{x}} - \hat{\mathbf{x}}^t), A(\hat{\mathbf{x}} - \hat{\mathbf{x}}^{t+1})\rangle \\ &\quad -2\mu\langle A(\mathbf{x} - \tilde{\mathbf{x}}), A(\hat{\mathbf{x}} - \hat{\mathbf{x}}^{t+1})\rangle - \mu\langle A^T\mathbf{z}, \hat{\mathbf{x}} - \hat{\mathbf{x}}^{t+1}\rangle. \end{aligned} \tag{33}$$

For $t = 1, 2, \ldots$, define a normalized error vector as follows

$$\mathbf{e}^t = \frac{\hat{\mathbf{x}} - \hat{\mathbf{x}}^t}{\|\hat{\mathbf{x}} - \hat{\mathbf{x}}^t\|}. \tag{34}$$

Using this definition, the triangle inequality and the Cauchy-Schwartz inequality, we rewrite (33) as follows

$$\begin{aligned}\|\hat{\mathbf{x}} - \hat{\mathbf{x}}^{t+1}\| &\leq 2\left(\langle \mathbf{e}^{t+1}, \mathbf{e}^t\rangle - \mu\langle A\mathbf{e}^{t+1}, A\mathbf{e}^t\rangle\right)\|\hat{\mathbf{x}} - \hat{\mathbf{x}}^t\| \\ &+ 2\mu\|A(\mathbf{x} - \hat{\mathbf{x}})\|\|A\mathbf{e}^{t+1}\| + 2\mu\left|\langle A^T\mathbf{z}, \mathbf{e}^{t+1}\rangle\right| \\ &\leq 2\left(\langle \mathbf{e}^{t+1}, \mathbf{e}^t\rangle - \mu\langle A\mathbf{e}^{t+1}, A\mathbf{e}^t\rangle\right)\|\hat{\mathbf{x}} - \hat{\mathbf{x}}^t\| \\ &+ 2\mu(\sigma_{\max}(A))^2\|\mathbf{x} - \hat{\mathbf{x}}\| + 2\mu\left|\langle A^T\mathbf{z}, \mathbf{e}^{t+1}\rangle\right|. \end{aligned} \tag{35}$$

To prove the desired result that connects the error at iteration $t+1$, $\|\hat{\mathbf{x}} - \hat{\mathbf{x}}^{t+1}\|$, to the error at iteration t, $\|\hat{\mathbf{x}} - \hat{\mathbf{x}}^t\|$, we first define the quantized versions of the error and the reconstruction vectors. The reason for this discretization becomes clear later when we use them to prove our concentration results.

For $t = 1, 2, \ldots$, define $\mathbf{u}_b^t \triangleq [\mathbf{u}^t]_b$ and

$$\hat{\mathbf{x}}_b^t \triangleq g(\mathbf{u}_b^t).$$

Also, let

$$\eta^t \triangleq \hat{\mathbf{x}}^t - \hat{\mathbf{x}}_b^t.$$

Assume that the quantization level b is selected as follows

$$b = \left\lceil (1+\alpha)\log\tfrac{1}{\delta}\right\rceil. \tag{36}$$

Since by assumption g is a Lipschitz function, we have

$$\begin{aligned}\|\eta^t\| = \|g(\mathbf{u}^t) - g(\mathbf{u}_b^t)\| &\leq L\|\mathbf{u}^t - \mathbf{u}_b^t\| \leq L2^{-b}\sqrt{k} \\ &\leq L\delta^{1+\alpha}\sqrt{k} \end{aligned} \tag{37}$$

where the last line follow from (36). Let

$$\mathbf{e}_b^t \triangleq \frac{\hat{\mathbf{x}} - \hat{\mathbf{x}}_b^t}{\|\hat{\mathbf{x}} - \hat{\mathbf{x}}_b^t\|}.$$

We next bound $\|\mathbf{e}^t - \mathbf{e}_b^t\|$, the distance between $\mathbf{e}_b^t$ and $\mathbf{e}^t$, where $\mathbf{e}^t$ is defined in (34). Note that

$$\begin{aligned}\mathbf{e}^t = \frac{\hat{\mathbf{x}} - \hat{\mathbf{x}}^t}{\|\hat{\mathbf{x}} - \hat{\mathbf{x}}^t\|} &= \frac{\hat{\mathbf{x}} - \hat{\mathbf{x}}_b^t - \eta^t}{\|\hat{\mathbf{x}} - \hat{\mathbf{x}}_b^t - \eta^t\|} \\ &= \mathbf{e}_b^t - \frac{\hat{\mathbf{x}} - \hat{\mathbf{x}}_b^t}{\|\hat{\mathbf{x}} - \hat{\mathbf{x}}_b^t\|} + \frac{\hat{\mathbf{x}} - \hat{\mathbf{x}}_b^t - \eta^t}{\|\hat{\mathbf{x}} - \hat{\mathbf{x}}_b^t - \eta^t\|}. \end{aligned} \tag{38}$$

Therefore, by the triangle inequality, it follows that

$$\begin{aligned}\|\mathbf{e}^t - \mathbf{e}_b^t\| &\leq \tfrac{\left|\|\hat{\mathbf{x}} - \hat{\mathbf{x}}_b^t\| - \|\hat{\mathbf{x}} - \hat{\mathbf{x}}_b^t - \eta^t\|\right|}{\|\hat{\mathbf{x}} - \hat{\mathbf{x}}_b^t - \eta^t\|} + \tfrac{\|\eta^t\|}{\|\hat{\mathbf{x}} - \hat{\mathbf{x}}_b^t - \eta^t\|} \\ &\leq \tfrac{2\|\eta^t\|}{\|\hat{\mathbf{x}} - \hat{\mathbf{x}}_b^t - \eta^t\|} = \tfrac{2\|\eta^t\|}{\|\hat{\mathbf{x}} - \hat{\mathbf{x}}^t\|} \\ &\overset{(a)}{\leq} \tfrac{2L2^{-b}\sqrt{k}}{\|\hat{\mathbf{x}} - \hat{\mathbf{x}}^t\|} \\ &\overset{(b)}{\leq} \tfrac{2L\delta^{1+\alpha}\sqrt{k}}{\sqrt{n}\delta} = 2L\delta^{\alpha}\sqrt{\tfrac{k}{n}}, \end{aligned} \tag{39}$$

where (a) and (b) follow from (37) and our assumption that $\|\hat{\mathbf{x}} - \hat{\mathbf{x}}^t\| \geq \sqrt{n}\delta$.

Using the introduced quantizations, in the following, we bound the three terms on the RHS of (35).

Step 1: Bound $(\langle \mathbf{e}^{t+1}, \mathbf{e}^t\rangle - \mu\langle A\mathbf{e}^{t+1}, A\mathbf{e}^t\rangle)\|\hat{\mathbf{x}} - \hat{\mathbf{x}}^t\|$. First, note that $\langle \mathbf{e}^{t+1}, \mathbf{e}^t\rangle - \mu\langle A\mathbf{e}^{t+1}, A\mathbf{e}^t\rangle$ is equal to

$$\begin{aligned}&\langle \mathbf{e}^{t+1} - \mathbf{e}_b^{t+1} + \mathbf{e}_b^{t+1}, \mathbf{e}^t - \mathbf{e}_b^t + \mathbf{e}_b^t\rangle \\ &\quad - \mu\langle A(\mathbf{e}^{t+1} - \mathbf{e}_b^{t+1} + \mathbf{e}_b^{t+1}), A(\mathbf{e}^t - \mathbf{e}_b^t + \mathbf{e}_b^t)\rangle \\ &= \langle \mathbf{e}_b^{t+1}, \mathbf{e}_b^t\rangle - \mu\langle A\mathbf{e}_b^{t+1}, A\mathbf{e}_b^t\rangle \\ &\quad + \langle \mathbf{e}^{t+1} - \mathbf{e}_b^{t+1}, \mathbf{e}^t - \mathbf{e}_b^t\rangle - \mu\langle A(\mathbf{e}^{t+1} - \mathbf{e}_b^{t+1}), \\ &\qquad A(\mathbf{e}^t - \mathbf{e}_b^t)\rangle. \end{aligned} \tag{40}$$

Therefore, applying the Cauchy-Schwarz inequality and the triangle inequality, it follows that

$$\begin{aligned}&\left|\left(\langle \mathbf{e}^{t+1}, \mathbf{e}^t\rangle - \mu\langle A\mathbf{e}^{t+1}, A\mathbf{e}^t\rangle\right)\right. \\ &\quad \left. - \left(\langle \mathbf{e}_b^{t+1}, \mathbf{e}_b^t\rangle - \mu\langle A\mathbf{e}_b^{t+1}, A\mathbf{e}_b^t\rangle\right)\right| \\ &\leq |\langle \mathbf{e}^{t+1} - \mathbf{e}_b^{t+1}, \mathbf{e}^t - \mathbf{e}_b^t\rangle| \\ &\quad + \mu|\langle A(\mathbf{e}^{t+1} - \mathbf{e}_b^{t+1}), A(\mathbf{e}^t - \mathbf{e}_b^t)\rangle| \\ &\leq (1 + \mu(\sigma_{\max}(A))^2)\|\mathbf{e}^{t+1} - \mathbf{e}_b^{t+1}\|\|\mathbf{e}^t - \mathbf{e}_b^t\|. \end{aligned} \tag{41}$$

Define event $\mathcal{E}_1$ as

$$\mathcal{E}_1 \triangleq \{\sigma_{\max}(A) \leq 2\sqrt{m} + \sqrt{n}\}.$$

As mentioned earlier,

$$P(\mathcal{E}_1^c) \leq \mathrm{e}^{-\frac{m}{2}}.$$

Hence, conditioned on $\mathcal{E}_1$, $|(\langle \mathbf{e}^{t+1}, \mathbf{e}^t\rangle - \mu\langle A\mathbf{e}^{t+1}, A\mathbf{e}^t\rangle) - (\langle \mathbf{e}_b^{t+1}, \mathbf{e}_b^t\rangle - \mu\langle A\mathbf{e}_b^{t+1}, A\mathbf{e}_b^t\rangle)|$ can be bounded as

$$\begin{aligned}&\left(1 + \mu m\left(\sqrt{\tfrac{n}{m}} + 2\right)^2\right)\|\mathbf{e}^{t+1} - \mathbf{e}_b^{t+1}\|\|\mathbf{e}^t - \mathbf{e}_b^t\| \\ &\leq \tfrac{4k}{n}\left(1 + \left(\sqrt{\tfrac{n}{m}} + 2\right)^2\right)L^2\delta^{2\alpha}, \end{aligned} \tag{42}$$

where the last line follows from (39) and because $\mu = \frac{1}{m}$.

Next, we bound the quantized term $\langle \mathbf{e}_b^{t+1}, \mathbf{e}_b^t\rangle - \mu\langle A\mathbf{e}_b^{t+1}, A\mathbf{e}_b^t\rangle$. To do this, define the set of normalized error vectors as

$$\mathcal{F}_b \triangleq \left\{\tfrac{\hat{\mathbf{x}} - g(\mathbf{u})}{\|\hat{\mathbf{x}} - g(\mathbf{u})\|} : \mathbf{u} \in \mathcal{U}_b^k\right\}. \tag{43}$$

Clearly, $|\mathcal{F}_b| \leq |\mathcal{U}_b|^k$. Define $\mathcal{E}_2 \triangleq \{\langle \mathbf{e}_b, \mathbf{e}_b'\rangle - \frac{1}{m}\langle A\mathbf{e}_b, A\mathbf{e}_b'\rangle \leq 0.45 : \forall\ (\mathbf{e}_b, \mathbf{e}_b') \in \mathcal{F}_b^2\}$. Applying Lemma 2, and the union bound, it follows that

$$\begin{aligned}\mathrm{P}(\mathcal{E}_2^c) &\leq |\mathcal{U}_b|^{2k}2^{-0.05m} \leq 2^{2bk - 0.05m} \\ &\overset{(a)}{\leq} 2^{2k(1+\alpha)\log\frac{1}{\delta} - 0.05m} \\ &\overset{(b)}{\leq} 2^{-2k\upsilon\log\frac{1}{\delta}}, \end{aligned} \tag{44}$$

where (a) and (b) hold because b, defined in (36), is smaller than $\alpha\log\frac{1}{\delta} + 1$ and m is greater than

$k40(1+\alpha+\upsilon)\log\frac{1}{\delta}$ by assumption, respectively. Finally, conditioned on $\mathcal{E}_1\cap\mathcal{E}_2$, combining (40) and (42), it follows that

$$\begin{aligned}&2\left(\langle\mathbf{e}^{t+1},\mathbf{e}^t\rangle-\mu\langle A\mathbf{e}^{t+1},A\mathbf{e}^t\rangle\right)\|\hat{\mathbf{x}}-\hat{\mathbf{x}}^t\|\\&\le(0.9+\eta)\|\hat{\mathbf{x}}-\hat{\mathbf{x}}^t\|,\end{aligned}\tag{45}$$

where η is defined in (10).

Step 2: Bound $2\mu(\sigma_{\max}(A))^2\|\mathbf{x}-\hat{\mathbf{x}}\|$: Note that

$$\begin{aligned}\|\hat{\mathbf{x}}-\mathbf{x}\|&=\min_{\mathbf{u}\in\mathcal{U}_b^k}\|g(\mathbf{u})-\mathbf{x}\|\le\|g([\tilde{\mathbf{u}}]_b)-\mathbf{x}\|\\&=\|g([\tilde{\mathbf{u}}]_b)-g(\tilde{\mathbf{u}})+g(\tilde{\mathbf{u}})-\mathbf{x}\|\\&\le\|g([\tilde{\mathbf{u}}]_b)-g(\tilde{\mathbf{u}})\|+\|g(\tilde{\mathbf{u}})-\mathbf{x}\|\\&\le L\|[\tilde{\mathbf{u}}]_b-\tilde{\mathbf{u}}\|+\sqrt{n}\delta\\&\le L\sqrt{k}2^{-b}+\sqrt{n}\delta.\end{aligned}\tag{46}$$

Therefore, using (46), conditioned on $\mathcal{E}_2$, we have

$$\begin{aligned}&2\mu(\sigma_{\max}(A))^2\|\mathbf{x}-\hat{\mathbf{x}}\|\\&\le(2+\sqrt{\tfrac{n}{m}}\,)^2(L\sqrt{k}2^{-b}+\sqrt{n}\delta)\\&\le(2+\sqrt{\tfrac{n}{m}}\,)^2(L\delta^\alpha\sqrt{\tfrac{k}{n}}+1)\sqrt{n}\delta=\gamma_1\delta\sqrt{n},\end{aligned}\tag{47}$$

where γ_1 is defined (11). *Step 3: Bound* $2\mu\left|\langle A^T\mathbf{z},\mathbf{e}^{t+1}\rangle\right|$: First, note that $\langle A^T\mathbf{z},\mathbf{e}^{t+1}\rangle=\langle\mathbf{z},A\mathbf{e}^{t+1}\rangle$, and

$$\begin{aligned}&|\langle A^T\mathbf{z},\mathbf{e}^{t+1}\rangle|\\&=|\langle\mathbf{z},A\mathbf{e}^{t+1}\rangle|=|\langle\mathbf{z},A(\mathbf{e}^{t+1}-\mathbf{e}_b^{t+1}+\mathbf{e}_b^{t+1})\rangle|\\&\overset{(a)}{\le}|\langle\mathbf{z},A\mathbf{e}_b^{t+1}\rangle|+|\langle\mathbf{z},A(\mathbf{e}^{t+1}-\mathbf{e}_b^{t+1})\rangle|\\&\overset{(b)}{\le}|\langle\mathbf{z},A\mathbf{e}_b^{t+1}\rangle|+\sigma_{\max}(A)\|\mathbf{z}\|\|\mathbf{e}^{t+1}-\mathbf{e}_b^{t+1}\|\\&\overset{(c)}{\le}|\langle\mathbf{z},A\mathbf{e}_b^{t+1}\rangle|+\sigma_{\max}(A)\|\mathbf{z}\|L\delta^\alpha\sqrt{\frac{k}{n}},\end{aligned}\tag{48}$$

where (a), (b) and (c) follow from the triangle inequality, the Cauchy-Schwarz inequality and (39), respectively. Next, to bound $|\langle\mathbf{z},A\mathbf{e}_b^{t+1}\rangle|$, we employ Lemma 3. For $\tau>0$ and $\tau_z>0$, define events $\mathcal{E}_3$ and $\mathcal{E}_4$ as

$$\mathcal{E}_3\triangleq\{|\langle\mathbf{z},A\mathbf{e}_b\rangle|\le\sigma\sqrt{(1+\tau)m}:\mathbf{e}_b\in\mathcal{F}_b\},\tag{49}$$

and

$$\mathcal{E}_4\triangleq\{\|\mathbf{z}\|\le\sigma\sqrt{m(1+\tau_z)}\,\},\tag{50}$$

respectively. By the law of total probability,

$$\begin{aligned}\mathrm{P}(\mathcal{E}_3^c)&=\mathrm{P}(\mathcal{E}_3^c\cap\mathcal{E}_4)+\mathrm{P}(\mathcal{E}_3^c\cap\mathcal{E}_4^c)\\&\le\mathrm{P}(\mathcal{E}_3^c\cap\mathcal{E}_4)+\mathrm{P}(\mathcal{E}_4^c).\end{aligned}\tag{51}$$

For a fixed $\mathbf{e}_b\in\mathcal{F}_b$, $A\mathbf{e}_b$ is i.i.d. $\mathcal{N}(0,1)$ and independent of $\mathbf{z}$. Therefore, by Lemma 3, $\langle\mathbf{z},A\mathbf{e}_b\rangle$ has the same distribution as $\|\mathbf{z}\|G_{\mathbf{e}_b}$, where $G_{\mathbf{e}_b}$ is independent of $\mathbf{z}$ and is distributed as $\mathcal{N}(0,1)$. Hence, for a fixed $\mathbf{e}_b$,

$$\begin{aligned}&\mathrm{P}(\langle\mathbf{z},A\mathbf{e}_b^{t+1}\rangle\ge\sigma\sqrt{(1+\tau)m},\mathcal{E}_4)\\&=\mathrm{P}\left(G_{\mathbf{e}_b}\|\mathbf{z}\|\ge\sigma\sqrt{(1+\tau)m},\mathcal{E}_4\right)\\&\le\mathrm{P}\left(G_{\mathbf{e}_b}\ge\sqrt{\tfrac{1+\tau}{1+\tau_z}},\mathcal{E}_4^c\right)\\&\le\mathrm{P}\left(G_{\mathbf{e}_b}\ge\sqrt{\tfrac{1+\tau}{1+\tau_z}}\right)\\&\le\mathrm{e}^{-\frac{1+\tau}{2(1+\tau_z)}},\end{aligned}\tag{52}$$

where the last line holds because for $G\sim\mathcal{N}(0,1)$ and $\tau>0$, $\mathrm{P}(G>\tau)\le\mathrm{e}^{-\tau^2/2}$. Therefore, applying the union bound, it follows that

$$\begin{aligned}\mathrm{P}(\mathcal{E}_3^c\cap\mathcal{E}_4)&\le2^{2kb}\mathrm{e}^{-\frac{1+\tau}{2(1+\tau_z)}}\\&\le2^{2k(1+(1+\alpha)\log\frac{1}{\delta})}\mathrm{e}^{-\frac{1+\tau}{2(1+\tau_z)}}.\end{aligned}\tag{53}$$

Also, by Lemma 1,

$$\mathrm{P}(\mathcal{E}_4^c)\le\mathrm{e}^{-\frac{m}{2}(\tau_z-\ln(1+\tau_z))}.$$

Let $\tau_z=1$. Then, $\tau_z-\ln(1+\tau_z)>0.3$ and

$$\mathrm{P}(\mathcal{E}_4^c)\le\mathrm{e}^{-0.15m}.\tag{54}$$

Choosing

$$\tau=-1+6(1+\alpha)\Big(\log\tfrac{1}{\delta}\Big)k,$$

the exponent of the RHS of (53) can be bounded as $2(\ln2)k(1+(1+\alpha)\log\frac{1}{\delta})-\frac{1+\tau}{2(1+\tau_z)}=2(\ln2)k(1+(1+\alpha)\log\frac{1}{\delta})-1.5(1+\alpha)\left(\log\frac{1}{\delta}\right)k\le-0.1(1+\alpha)\left(\log\frac{1}{\delta}\right)k+2(\ln2)k$. Therefore,

$$\mathrm{P}(\mathcal{E}_3^c\cap\mathcal{E}_4)\le\mathrm{e}^{-0.1(1+\alpha)\left(\log\frac{1}{\delta}\right)k+2(\ln2)k}.\tag{55}$$

Moreover, for this choice of τ, conditioned on $\mathcal{E}_3$,

$$\mu\langle\mathbf{z},A\mathbf{e}_b^{t+1}\rangle\le\sigma\sqrt{\tfrac{1+\tau}{m}}=\sigma\sqrt{\tfrac{6(1+\alpha)\left(\log\frac{1}{\delta}\right)k}{m}}.\tag{56}$$

Also, conditioned on $\mathcal{E}_1\cap\mathcal{E}_3\cap\mathcal{E}_3$,

$$\begin{aligned}&\mu\sigma_{\max}(A)\|\mathbf{z}\|L\delta^\alpha\sqrt{\tfrac{k}{n}}\le\tfrac{2\sqrt{m}+\sqrt{n}}{m}\sigma\sqrt{2m}L\delta^\alpha\sqrt{\tfrac{k}{n}}\\&=\sigma\sqrt{\tfrac{2k}{n}}\left(2+\sqrt{\tfrac{n}{m}}\,\right)L\delta^\alpha\\&=\gamma_2\sigma L\delta^\alpha,\end{aligned}\tag{57}$$

where γ_2 is defined in (24). Hence, in summary, conditioned on $\mathcal{E}_1\cap\mathcal{E}_3\cap\mathcal{E}_3$,

$$\tfrac{2\mu}{\sigma}\left|\langle A^T\mathbf{z},\mathbf{e}^{t+1}\rangle\right|\le\sqrt{\tfrac{6(1+\alpha)\left(\log\frac{1}{\delta}\right)k}{m}}+\gamma_2L\delta^\alpha\tag{58}$$

Having the bounds on the three terms, combining (45), (47) and (58), conditioned on $\mathcal{E}_1\cap\mathcal{E}_3\cap\mathcal{E}_3\cap\mathcal{E}_4$, the desired result follows from dividing both sides of (35) by $\frac{1}{\sqrt{n}}$.

VII. Conclusions

In this paper, we have theoretically studied the performance of an idealized CS recovery method that em-

ploys exhaustive search over all the outputs of a GF corresponding to our desired class of signals $\mathcal{Q}$. In the asymptotic regime, where n (the ambient dimension of set $\mathcal{Q}$) grows without bound, having a family of GFs with input dimension $k = k_n$ and representation error $\delta = \delta_n$ converging to zero, we have shown that, roughly, k measurements are sufficient for almost lossless recovery. We have also studied the performance of an efficient algorithm based on PGD that employs an AE at each iteration to project the updated signal onto the set of desired signals. We refer to this method as AE-PGD and prove that given enough measurements, the algorithm converges to the vicinity of the optimal solution even in the presence of additive white Gaussian noise. We have provided simulation results that highlight both the power and the potential weaknesses of such recovery methods based on GFs.

REFERENCES

[1] D. Huang, E. A. Swanson, C. P. Lin, J. S. Schuman, W. G. Stinson, W. Chang, M. R. Hee, T. Flotte, K. Gregory, C. A. Puliafito, and al. et. Optical coherence tomography. *Science*, 254(5035):1178–1181, 1991.

[2] T. Blumensath and M. E. Davies. Iterative hard thresholding for compressed sensing. *Appl. Comp. Harmonic Anal. (ACHA)*, 27(3):265–274, 2009.

[3] G. Cybenko. Approximations by superpositions of a sigmoidal function. *Math. of Cont., Sig. and Sys.*, 2:183–192, 1989.

[4] K.I. Funahashi. On the approximate realization of continuous mappings by neural networks. *Neu. net.*, 2(3):183–192, 1989.

[5] K. Hornik, M. Stinchcombe, and H. White. Multilayer feedforward networks are universal approximators. *Neu. Net.*, 2(5):359–366, 1989.

[6] A. R. Barron. Approximation and estimation bounds for artificial neural networks. *Mach. lear.*, 14(1):115–133, 1994.

[7] J. Friedman, T. Hastie, and R. Tibshirani. *The elements of statistical learning*, volume 1. Springer series in statistics New York, NY, USA:, 2001.

[8] D. P. Kingma and M. Welling. Auto-encoding variational Bayes. In *Int. Conf. Learn. Rep. (ICLR)*, June 2013.

[9] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, 2016. http://www.deeplearningbook.org.

[10] A. Bora, A. Jalal, E. Price, and A. G. Dimakis. Compressed sensing using generative models. In *Int. Conf. Mach. Learn.*, pages 537–546, 2017.

[11] V. Shah and C. Hegde. Solving linear inverse problems using gan priors: An algorithm with provable guarantees. In *Int. Conf. on Aco., Speech, and Sig. Proc. (ICASSP)*, Apr. 2018.

[12] Y. Wu and S. Verdú. Rényi information dimension: Fundamental limits of almost lossless analog compression. *IEEE Trans. Inform. Theory*, 56(8):3721–3748, Aug. 2010.

[13] A. Mousavi, A. B. Patel, and R. G. Baraniuk. A deep learning approach to structured signal recovery. In *Ann. Allerton Conf. on Commun., Cont., and Comp.*, pages 1336–1343. IEEE, 2015.

[14] K. Kulkarni, S. Lohit, P. Turaga, R. Kerviche, and A. Ashok. ReconNet: Non-iterative reconstruction of images from compressively sensed measurements. In *Proc. of the IEEE Conf. on Comp. Vis. and Pat. Rec. (CVPR)*, pages 449–458, 2016.

[15] J.H. Rick Chang, C.-L. Li, B. Poczos, B.V.K. Vijaya Kumar, and A. C. Sankaranarayanan. One network to solve them all- solving linear inverse problems using deep projection models. In *Proc. of the IEEE Int. Conf. on Comp. Vis.*, pages 5888–5897, 2017.

[16] M. Borgerding, P. Schniter, and S. Rangan. AMP-inspired deep networks for sparse linear inverse problems. *IEEE Trans. Sig. Proc.*, 65(16):4293–4308, 2017.

[17] C. Metzler, A. Mousavi, and R. Baraniuk. Learned D-AMP: Principled neural network based compressive image recovery. In *Adv. in Neu. Inform. Process. Sys.*, pages 1772–1783, 2017.

[18] X. Yuan and Y. Pu. Parallel lensless compressive imaging via deep convolutional neural networks. *Optics Express*, 26(2):1962–1977, Jan 2018.

[19] K. H. Jin, M. T. McCann, E. Froustey, and M. Unser. Deep convolutional neural network for inverse problems in imaging. *Trans. Image Proc.*, 26(9):4509–4522, Sep. 2017.

[20] D. Van Veen, A. Jalal, M. Soltanolkotabi, E. Price, S. Vishwanath, and A. G. Dimakis. Compressed sensing with deep image prior and learned regularization. *arXiv preprint arXiv:1806.06438*, 2018.

[21] P. Hand and V. Voroninski. Global guarantees for enforcing deep generative priors by empirical risk. *IEEE Trans. Inform. Theory*, 66(1):401–418, 2019.

[22] S. Jalali and A. Maleki. From compression to compressed sensing. *Appl. Comp. Harmonic Anal. (ACHA)*, 40(2):352–385, 2016.

[23] Y LeCun, C. Cortes, and C. J.C. Burges. The mnist database of handwritten digits. In *http://yann.lecun.com/exdb/mnist/index.html*.

[24] National Institutes of Health Chest X-Ray Dataset. In *https://www.kaggle.com/nih-chest-xrays*.

[25] Z. Liu, P. Luo, X. Wang, and X. Tang. Deep learning face attributes in the wild. In *Proc. of Int. Conf. on Comp. Vis. (ICCV)*, Dec. 2015.

[26] https://github.com/Qihuan1988/Solving-inverse-problems-via-auto-encoders.

[27] R. Tibshirani. Regression shrinkage and selection via the lasso. *J. of the Roy. Stat. Soc.: Ser. B (Meth.)*, 58(1):267–288, 1996.

[28] C. A. Metzler, A. Maleki, and R. G. Baraniuk. BM3D-AMP: A new image recovery algorithm based on BM3D denoising. In *2015 IEEE Int. Conf. on Image Proc. (ICIP)*, pages 3116–3120. IEEE, 2015.

[29] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *Med. Image Comp. and Comp. Assis. Int. (MICCAI)*, volume 9351 of *LNCS*, pages 234–41. Springer, 2015.

[30] S. Jalali, A. Maleki, and R. G. Baraniuk. Minimum complexity pursuit for universal compressed sensing. *IEEE Trans. Inform. Theory*, 60(4):2253–2268, Apr. 2014.

[31] S. Jalali and A. Maleki. New approach to bayesian high-dimensional linear regression. *Inf. and Inf.: A J. of the IMA*, 7(4):605–655, 2018.