Algorithmic Aspects of Inverse Problems Using Generative Models

Chinmay Hegde[1]

Abstract— The traditional approach of hand-crafting priors (such as sparsity) for solving inverse problems is slowly being replaced by the use of richer learned priors (such as those modeled by generative adversarial networks, or GANs). In this work, we study the algorithmic aspects of such a learning-based approach from a theoretical perspective. For certain generative network architectures, we establish a simple non-convex algorithmic approach that (a) theoretically enjoys linear convergence guarantees for certain inverse problems, and (b) empirically improves upon conventional techniques such as back-propagation. We also propose an extension of our approach that can handle model mismatch (i.e., situations where the generative network prior is not exactly applicable.) Together, our contributions serve as building blocks towards a more complete algorithmic understanding of generative models in inverse problems.

I. INTRODUCTION

A. Motivation

Inverse problems arise in a diverse range of application domains including computational imaging, optics, astrophysics, and seismic geo-exploration. In each of these applications, there is a target signal or image (or some other quantity of interest) to be obtained; a device (or some other physical process) records measurements of the target; and the goal is to reconstruct an estimate of the signal from the observations.

Let us suppose that $x \in \mathbb{R}^n$ denotes the signal of interest and $y = \mathcal{A}(x) \in \mathbb{R}^m$ denotes the observed measurements. When $m < n$ the inverse problem is ill-posed, and some kind of prior (or regularizer) is necessary to obtain a meaningful solution. A common technique used to solve ill-posed inverse problems is to solve a constrained optimization problem:

$$\widehat{x} = \arg\min \ F(x), \qquad (1)$$
$$\text{s. t.} \quad x \in \mathcal{S},$$

where F is an objective function that typically depends on y and $\mathcal{A}$, and $\mathcal{S} \subseteq \mathbb{R}^n$ captures some sort of *structure* that x is assumed to obey.

A very common modeling assumption, particularly in signal and image processing applications, is *sparsity*, wherein $\mathcal{S}$ is the set of sparse vectors in some (known) basis representation. The now-popular framework of *compressive sensing* studies the special case where the forward measurement operator $\mathcal{A}$ can be modeled as a linear operator that satisfies certain (restricted) stability properties; when this is the case, accurate estimation of x^* can be performed provided the signal x is sufficiently sparse [1]. Parallel to the development of algorithms that leverage sparsity priors, the last decade

has witnessed analogous approaches for other families of structural constraints. These include structured sparsity [2], [3], union-of-subspaces [4], dictionary models [5], [6], total variation models [7], analytical transforms [8], among many others.

Lately, there has been renewed interest in prior models that are parametrically defined in terms of a *deep neural network*. We call these *generative network* models. Specifically, we define

$$\mathcal{S} = \{x \in \mathbb{R}^n \mid x = G(z), \ z \in \mathbb{R}^k\}$$

where z is a k-dimensional latent parameter vector and G is parameterized by the weights and biases of a d-layer neural network. One way to obtain such a model is to train a generative adversarial network (GAN) [9]. A well-trained GAN closely captures the notion of a signal (or image) being 'natural' [10], leading many to speculate that the range of such generative models can approximate a low-manifold containing naturally occurring images. Indeed, GAN-based neural network learning algorithms have been successfully employed to solve *linear* inverse problems such as image super-resolution and inpainting [11], [12]. However, most of these approaches are heuristic, and a general theoretical framework for analyzing the performance of such approaches is not available at the moment.

B. Contributions

Our focus in this paper is to take some initial steps into building such a theoretical framework. Specifically, we wish to understand the *algorithmic* costs involving in using generative network models for inverse problems: how computationally challenging they are, whether they provably succeed, and how to make such models robust.

The starting point of our work is the recent, seminal paper by [13], who study the benefits of using generative models in the context of compressive sensing. In this paper, the authors pose the estimated target as the solution to a non-convex optimization problem and establish upper bounds on the *statistical* complexity of obtaining a "good enough" solution. Specifically, they prove that if the generative network is a mapping $G : \mathbb{R}^k \to \mathbb{R}^n$ simulated by a d-layer neural network with width $\leq n$ and with activation functions obeying certain properties, then $m = O(kd \log n)$ random observations are sufficient to obtain a good enough reconstruction estimate. However, the authors do not study the *algorithmic* costs of solving the optimization problem, and standard results in non-convex optimization are sufficient to only obtain sublinear convergence rates. In earlier work [14], we established an algorithm with linear convergence rates for

---

[1]CH is with the Electrical and Computer Engineering Department at Iowa State University, Ames, IA USA. Email: chinmay@iastate.edu.

the same (compressive sensing) setup, and demonstrated its empirical benefits.

However, the earlier work [14] only provided an algorothm (and analysis) for linear inverse problems. In this work, we generalize this to a much wider range of *nonlinear* inverse problems. Using standard techniques, we propose a generic algorithm for solving (1), analyze its performance, and prove that it demonstrates linear convergence. This constitutes **Contribution I** of this paper.

A drawback of [13] (and our subsequent work [14]) is the inability to deal with targets that are outside the range of the generative network model. This is not merely an artifact of their analysis; generative networks are rigid in the sense that once they are learned, they are incapable of reproducing any target outside their range. (This is in contrast with other popular parametric models such as sparsity models; these exhibit a "graceful decay" property in the sense that if the sparsity parameter s is large enough, such models capture all possible points in the target space.) This issue is addressed, and empirically resolved, in the recent work of [15] who propose a hybrid model combining both generative networks and sparsity. This leads to a non-convex optimization framework (called *SparseGen*) which the authors theoretically analyze to obtain analogous statistical complexity results. However, here too, the theoretical contribution is primarily statistical and the algorithmic aspects of their setup are not discussed.

We address this gap, and propose an alternative algorithm for this framework. Our algorithm is new, and is a nonlinear extension of our previous work [16], [17]. Under (fairly) standard assumptions, this algorithm also can be shown to demonstrate linear convergence. This constitutes **Contribution II** of this paper.

In summary: we complement the work of [13] and [15] by providing algorithmic upper bounds for the corresponding problems that are studied in those works. Together, our contributions serve as further building blocks towards an algorithmic theory of generative models in inverse problems.

C. Techniques

At a high level, our algorithms are standard. The primary novelty is in their applications to generative network models, and some aspects of their theoretical analysis.

Suppose that $G : \mathbb{R}^k \to \mathbb{R}^n$ is the generative network model under consideration. The cornerstone of our analysis is the assumption of an ε-*approximate* (Euclidean) projection oracle onto the range of G. In words, we pre-suppose the availability of a computational routine P_G that, given any vector $x \in \mathbb{R}^n$, can return a vector $x' \in \text{Range}(G)$ that approximately minimizes $\|x - x\|_2^2$. The availability of this oracle, of course, depends on G and we comment on how to construct such oracles below in Section IV.

The first algorithm (for solving (1)) is a straightforward application of *projected gradient descent*, and is a direct nonlinear generalization of the approach proposed in [14]. The main difficulty is in analyzing the algorithm and proving linear convergence. To show this, we assume that the
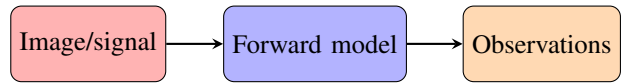

Fig. 1. Block diagram for a generic inverse problem. The goal is to reconstruct (an estimate of) the data/signal given knowledge of the observations or measurements.

objective function in (1) obeys the Restricted Strong Convexity/Smoothness assumptions [18]. With this assumption, proof of convergence follows from a straightforward modification of the proof given in [19].

The second algorithm (for handling model mismatch in the target) is new. The main idea (following the lead of [15]) is to pose the target x as the superposition of two components: $x = G(z) + \nu$, where ν can be viewed as an "innovation" term that is s-sparse in some fixed, known basis B. The goal is now to recover both $G(z)$ and ν. This is reminiscent of the problem of source separation or signal demixing [20], and in our previous work [17], [21] we proposed greedy iterative algorithms for solving such demixing problems. We extend this work by proving a nonlinear extension, together with a new analysis, of the algorithm proposed in [17].

II. BACKGROUND AND RELATED WORK

A. Inverse problems

The study of solving inverse problems has a long history. As discussed above, the general approach to solve an ill-posed inverse problem of the form depicted in Figure 1 is to assumes that the target signal/image obeys a *prior*. Older works mainly used hand-crafted signal priors; for example, [22]–[24] employ sparsity priors, and applied them in *linear* inverse problems such as super-resolution, denoising, compressive sensing, and missing entry interpolation.

B. Neural network models

The last few years have witnessed the emergence of trained *neural networks* for solving such problems. The main idea is to eschew hand-crafting any priors, and instead *learn* an end-to-end mapping from the measurement space to the image space. This mapping is simulated via a deep neural network, whose weights are learned from a large dataset of input-output training examples [25]. The works [26]–[32] have used this approach to solve several types of (linear) inverse problems, and has met with considerable success. However, the major limitations are that a new network has to be trained for each new linear inverse problem; moreover, most of these methods lack concrete theoretical guarantees. An exception of this line of work is the powerful framework of [33], which does *not* require retraining for each new problem; however, this too is not accompanied by theoretical analysis of statistical and computational costs.

C. Generative networks

A special class of neural networks that attempt to directly model the distribution of the input training samples are known as generative adversarial training networks, or GANs

[9]. GANs have been shown to provide visually striking results [10], [34]–[36]. The use of GANs to solve linear inverse problems was advocated in [13]. Specifically, given (noisy) linear observations $y = Ax + e$ of a signal $x \in \mathbb{R}^n$, assuming that x belongs to the range of a generative network $G : \mathbb{R}^n \to \mathbb{R}^k$, this approach constructs the reconstructed estimate $\hat{x}$ as follows:

$$\hat{z} = \arg\min_{z \in \mathbb{R}^k} \|y - AG(z)\|_2^2, \quad \hat{x} = G(\hat{z})$$

If the observation matrix $A \in \mathbb{R}^{m \times n}$ comprises $m = O(kd \log n)$ i.i.d. Gaussian measurements, then together with regularity assumptions on the generative network, they prove that the solution $\hat{x}$ satisfies:

$$\|x - \hat{x}\|_2 \le C\|e\|_2.$$

for some constant C that can be reliably upper-bounded. In particular, in the absence of noise the recovery of x is exact. However, there is no discussion of how computationally expensive this procedure is. Observe that the above minimization is highly non-convex (since for any reasonable neural network, G is a non-convex function) and possibly also non-smooth (if the neural network contains non-smooth activation functions, such as rectified linear units, or ReLUs). More recently, [37] improve upon the approach in [13] for solving more general nonlinear inverse problems (in particular, any inverse problem which has a computable derivative). Their approach involves simultaneously solving the inverse problem and training the network parameters; however, the treatment here is mostly empirical and a theoretical analysis is not provided.

Under similar statistical assumptions as [13], the work of [14] provably establishes a linear convergence rate, provided that a projection oracle (on to the range of G) is available, but only for the special case of compressive sensing. Our first main result (Contribution I) extends this algorithm (and analysis) to more general nonlinear inverse problems.

D. Model mismatch

A limitation of most generative network models is that they can only reproduce estimates that are within their range; adding more observations or tweaking algorithmic parameters are completely ineffective if a generative network model is presented with a target that is far away from the range of the model. To resolve this type of model mismatch, the authors of [15] propose to model the signal x as the superposition of two components: a "base" signal $u = G(z)$, and an "innovation" signal $v = B\nu$, where B is a known ortho-basis and ν is an l-sparse vector. In the context of compressive sensing, the authors of [15] solve a sparsity-regularized loss minimization problem:

$$(\hat{z}, \hat{v}) = \arg\min_{z,v} \|B^T v\|_1 + \lambda \|y - A(G(z) + v)\|_2^2.$$

and prove that the reconstructed estimate $\hat{x} = G(\hat{z}) + \hat{v}$ is close enough to x provided $m = O((k + l)d \log n)$

measurements are sufficient. However, as before, the algorithmic costs of solving the above problem are not discussed. Our second main result (Contribution II) proposes a new algorithm for dealing with model mismatches in generative network modeling, together with an analysis of its convergence and iteration complexity.

III. MAIN ALGORITHM AND ANALYSIS

Let us first establish some notational conventions. Below, $\|\cdot\|$ will denote the Euclidean norm unless explicitly specified. We use $O(\cdot)$-notation in several places in order to avoid duplication of constants.

We use $F(\cdot)$ to denote a (scalar) objective function. We assume that F has a continuous gradient $\nabla F = \left(\frac{\partial F}{\partial x_i}\right)_{i=1}^n$ which can be evaluated at any point $x \in \mathbb{R}^n$.

A. Definitions and assumptions

We now present certain definitions that will be useful for our algorithms and analysis.

Definition 1 (Approximate projection): A function $P_G : \mathbb{R}^n \to \text{Range}(G)$ is an ε-approximate projection oracle if for all $x \in \mathbb{R}^n$, $P_G(x)$ obeys:

$$\|x - P_G(x)\|_2^2 \le \min_{z \in \mathbb{R}^k} \|x - G(z)\|_2^2 + \varepsilon.$$

We will assume that for any given generative network G of interest, such a function P_G exists and is computationally tractable[1]. Here, $\varepsilon > 0$ is a parameter that is known *a priori*.

Definition 2 (Restricted Strong Convexity/Smoothness): Assume that F satisfies $\forall x, y \in S$:

$$\frac{\alpha}{2}\|x - y\|_2^2 \le F(y) - F(x) - \langle \nabla F(x), y - x \rangle \le \frac{\beta}{2}\|x - y\|_2^2.$$

for positive constants α, β.

This assumption is by now standard; see [18], [19] for in-depth discussions. This means that the objective function is strongly convex / strongly smooth along certain directions in the parameter space (in particular, those restricted to the set S of interest). The parameter $\alpha > 0$ is called the restricted strong convexity (RSC) constant, while the parameter $\beta > 0$ is called the restricted strong smoothness (RSS) constant. Clearly, $\beta \ge \alpha$. In fact, throughout the paper, we assume that $1 \le \frac{\beta}{\alpha} < 2$, which is a fairly stringent assumption but again, one that we do not know at the moment how to relax.

Definition 3 (Incoherence): A basis B and $\text{Range}(G)$ are called μ-incoherent if for all $u, u' \in \text{Range}(G)$ and all $v, v' \in \text{Span}(B)$, we have:

$$|\langle u - u', v - v' \rangle| \le \mu \|u - u'\|_2 \|v - v'\|_2.$$

for some parameter $0 < \mu < 1$.

Remark 1: In addition to the above, we will make the following assumptions in order to aid the analysis. Below, γ and Δ are positive constants.

---

[1]This may be a very strong assumption, but at the moment we do not know how to relax this. Indeed, the computational complexity of our proposed algorithms are directly proportional to the complexity of such a projection oracle.

Algorithm 1 ε-PGD

1: **Inputs:** y, T, ∇; **Output:** $\widehat{x}$
2: $x_0 \leftarrow \mathbf{0}$
3: **while** $t < T$ **do**
4: $\quad z_t \leftarrow x_t - \eta \nabla F(x_t)$
5: $\quad x_{t+1} \leftarrow P_G(z_t)$
6: $\quad t \leftarrow t + 1$
7: $\widehat{x} \leftarrow x_T$

- $\|\nabla F(x^*)\|_2 \leq \gamma$ (gradient at the minimizer is small).
- $\mathrm{diam}(\mathrm{Range}(G)) = \Delta$ (range of G is compact).
- $\gamma\Delta \leq O(\varepsilon)$.

B. Contribution I: An algorithm for nonlinear inverse problems using generative networks

We now present our first main result. Recall that we wish to solve the problem:

$$\widehat{x} = \arg\min \ F(x), \qquad (2)$$
$$\text{s. t.} \quad x \in \mathrm{Range}(G),$$

where G is a generative network. To do so, we employ *projected gradient descent* using the ε-approximate projection oracle for G. The algorithm is described in Alg. 1. We obtain the following theoretical result:

Theorem 1: If F satisfies RSC/RSS over $\mathrm{Range}(G)$ with constants α and β, then ε-PGD (Alg. 1) convergences linearly up to a ball of radius $O(\gamma\Delta) \approx O(\varepsilon)$.

$$F(x_{t+1}) - F(x^*) \leq \left(2 - \frac{\beta}{\alpha}\right)(F(x_t) - F(x^*)) + O(\varepsilon).$$

Proof: The proof is a minor modification of that in [19]. For simplicity we will assume that $\|\cdot\|$ refers to the Euclidean norm. Let us suppose that the step size $\eta = \frac{1}{\beta}$. Define

$$z_t = x_t - \eta \nabla F(x_t).$$

By invoking RSS, we get:

$$F(x_{t+1}) - F(x_t)$$
$$\leq \langle \nabla F(x_t), x_{t+1} - x_t \rangle + \frac{\beta}{2}\|x_{t+1} - x_t\|^2$$
$$= \frac{1}{\eta}\langle x_t - z_t, x_{t+1} - x_t \rangle + \frac{\beta}{2}\|x_{t+1} - x_t\|^2$$
$$= \frac{\beta}{2}\left(\|x_{t+1} - x_t\|^2 + 2\langle x_t - z_t, x_{t+1} - x_t \rangle + \|x_t - z_t\|^2\right)$$
$$\quad - \frac{\beta}{2}\|x_t - z_t\|^2$$
$$= \frac{\beta}{2}\left(\|x_{t+1} - z_t\|^2 - \|x_t - z_t\|^2\right),$$

where the last few steps are consequences of straightforward algebraic manipulation.

Now, since x_{t+1} is an ε-approximate projection of z_t onto $\mathrm{Range}(G)$ and $x^* \in \mathrm{Range}(G)$, we have:

$$\|x_{t+1} - z_t\|^2 \leq \|x^* - z_t\|^2 + \varepsilon.$$

Therefore, we get:

$$F(x_{t+1}) - F(x_t)$$
$$\leq \frac{\beta}{2}\left(\|x^* - z_t\|^2 - \|x_t - z_t\|^2\right) + \frac{\beta\varepsilon}{2}$$
$$= \frac{\beta}{2}\left(\|x^* - x_t + \eta\nabla F(x_t)\|^2 - \|\eta\nabla F(x_t)\|^2\right) + \frac{\beta\varepsilon}{2}$$
$$= \frac{\beta}{2}\left(\|x^* - x_t\|^2 + 2\eta\langle x^* - x_t, \nabla F(x_t)\rangle\right) + \frac{\beta\varepsilon}{2}$$
$$= \frac{\beta}{2}\|x^* - x_t\|^2 + \langle x^* - x_t, \nabla F(x_t)\rangle + \frac{\beta\varepsilon}{2}.$$

However, due to RSC, we have:

$$\frac{\alpha}{2}\|x^* - x_t\|^2 \leq F(x^*) - F(x_t) - \langle x^* - x_t, \nabla F(x_t)\rangle,$$

$$\langle x^* - x_t, \nabla F(x_t)\rangle \leq F(x^*) - F(x_t) - \frac{\alpha}{2}\|x^* - x_t\|^2.$$

Therefore,

$$F(x_{t+1}) - F(x_t)$$
$$\leq \frac{\beta - \alpha}{2}\|x^* - x_t\|^2 + F(x^*) - F(x_t) + \frac{\beta\varepsilon}{2}$$
$$\leq \frac{\beta - \alpha}{2} \cdot \frac{2}{\alpha}\left(F(x_t) - F(x^*) - \langle x_t - x^*, \nabla F(x^*)\rangle\right)$$
$$\quad + F(x^*) - F(x_t) + \frac{\beta\varepsilon}{2}$$
$$\leq \left(2 - \frac{\beta}{\alpha}\right)(F(x^*) - F(x_t)) + \frac{\beta - \alpha}{\alpha}\gamma\Delta + \frac{\beta\varepsilon}{2},$$

where the last inequality follows from Cauchy-Schwartz and the assumptions on $\|\nabla F(x^*)\|$ and the diameter of $\mathrm{Range}(G)$. Further, by assumption, $\gamma\Delta \leq O(\varepsilon)$. Rearranging terms, we get:

$$F(x_{t+1}) - F(x^*) \leq \left(\frac{\beta}{\alpha} - 1\right)(F(x_t) - F(x^*)) + C\varepsilon.$$

for some constant $C > 0$. ∎

This theorem asserts that the distance between the objective function at any iteration to the optimum *decreases by a constant factor* in every iteration. (The decay factor is $\frac{\beta}{\alpha} - 1$, which by assumption is a number between 0 and 1). Therefore, we immediately obtain linear convergence of ε-PGD up to a ball of radius $O(\varepsilon)$:

Corollary 1: After $T = O(\log \frac{F(x_0) - F(x^*)}{\varepsilon})$ iterations, $F(x_T) \leq F(x^*) + O(\varepsilon)$.

Therefore, the overall running time can be bounded as follows:

$$\text{Runtime} \leq (T_{\varepsilon-\text{PROJ}} + T_{\nabla}) \times \log(1/\varepsilon).$$

See [14] for empirical evaluations of PGD applied to a linear inverse problem (compressed sensing recovery).

C. Contribution II: Addressing signal model mismatch

We now generalize the ε-PGD algorithm to handle situations involving signal model mismatch. Assume that the target signal can be decomposed as:

$$x = G(z) + v,$$

where $\|B^T v\|_0 \leq l \ll n$ for some ortho-basis B.

Algorithm 2 MYOPIC ε-PGD

1: **Inputs:** y, T, ∇; **Output:** $\widehat{x}$
2: $x_0, u_0, v_0 \leftarrow \mathbf{0}$
3: **while** $t < T$ **do**
4:      $u_{t+1} = P_G(u_t - \eta \nabla_x F(x_t))$
5:      $v_{t+1} = \text{Thresh}_{B,l}(v_t - \eta \nabla_x F(x_t))$
6:      $x_{t+1} = u_{t+1} + \nu_{t+1}$
7:      $t \leftarrow t + 1$
8: $\widehat{x} \leftarrow x_T$

For this model, we attempt to solve a (slightly) different optimization problem:

$$\widehat{x} = \arg\min \ F(x), \tag{3}$$
$$\text{s. t.} \quad x = G(z) + v,,$$
$$\left\| B^T v \right\|_0 \le l. \tag{4}$$

We propose a new algorithm to solve this problem that we call *Myoptic ε-PGD*. This algorithm is given in Alg. 2[2].

Theorem 2: Let $\oplus$ denote the Minkowski sum. If F satisfies RSC/RSS over $\text{Range}(G) \oplus \text{Span}(B)$ with constants α and β, and if we assume μ-incoherence between B and $\text{Range}(G)$, we have:

$$F(x_{t+1}) - F(x^*) \le \left(\frac{2 - \frac{\beta}{\alpha} \frac{1 - 2.5\mu}{1 - \mu}}{1 - \frac{\beta}{2\alpha} \frac{\mu}{1 - \mu}} \right) (F(x_t) - F(x^*)) + O(\varepsilon).$$

Proof: We will generalize the proof technique of [17]. We first define some auxiliary variables that help us with the proof. Let:

$$z_t = x_t - \eta \nabla F(x_t),$$
$$z_t^u = u_t - \eta \nabla F(x_t),$$
$$z_t^v = v_t - \eta \nabla F(x_t).$$

and let $x^* = u^* + v^*$ be the minimizer that we seek. As above, by invoking RSS and with some algebra, we obtain:

$$F(x_{t+1}) - F(x^*) \le \frac{\beta}{2} \left(\|x_{t+1} - z_t\|^2 - \|x_t - z_t\|^2 \right), \tag{5}$$

However, by definition,

$$x_{t+1} = u_{t+1} + v_{t+1},$$
$$x_t = u_t + v_t.$$

Therefore,

$$\|x_{t+1} - z_t\|^2$$
$$= \|u_{t+1} - (u_t - \eta \nabla F(x_t)) +$$
$$\quad v_{t+1} - (v_t - \eta \nabla F(x_t)) + \eta \nabla F(x_t)\|^2$$
$$= \|u_{t+1} - (u_t - \eta \nabla F(x_t))\|^2 + \|\eta \nabla F(x_t)\|^2 +$$
$$\quad \|v_{t+1} - (v_t - \eta \nabla F(x_t))\|^2$$
$$\quad + 2\langle u_{t+1} - (u_t - \eta \nabla F(x_t)), \eta \nabla F(x_t) \rangle$$
$$\quad + 2\langle v_{t+1} - (v_t - \eta \nabla F(x_t)), \eta \nabla F(x_t) \rangle$$
$$\quad + 2\langle u_{t+1} - (u_t - \eta \nabla F(x_t)), v_{t+1} - (v_t - \eta \nabla F(x_t)) \rangle.$$

[2]The algorithm is a variant of block-coordinate descent, except that the block updates share the same gradient term.

But u_{t+1} is an ε-projection of z_t^u and u^* is in the range of G, we have:

$$\|u_{t+1} - z_t^u\|^2 \le \|u^* - z_t^u\|^2 + \varepsilon.$$

Similarly, since v_{t+1} is an l-sparse thresholded version of z_t^v, we have:

$$\|v_{t+1} - z_t^v\|^2 \le \|v^* - z_t^v\|^2.$$

Plugging in these two upper bounds, we get:

$$\|x_{t+1} - z_t\|^2$$
$$\le \|u^* - (u_t - \eta \nabla F(x_t))\|^2 + \varepsilon$$
$$\quad \|\eta \nabla F(x_t)\|^2 + \|v^* - (v_t - \eta \nabla F(x_t))\|^2$$
$$\quad + 2\langle u_{t+1} - (u_t - \eta \nabla F(x_t)), \eta \nabla F(x_t) \rangle$$
$$\quad + 2\langle v_{t+1} - (v_t - \eta \nabla F(x_t)), \eta \nabla F(x_t) \rangle$$
$$\quad + 2\langle u_{t+1} - (u_t - \eta \nabla F(x_t)), v_{t+1} - (v_t - \eta \nabla F(x_t)) \rangle.$$

Expanding squares and cancelling (several) terms, the right hand side of the above inequality can be simplified to obtain:

$$\|x_{t+1} - z_t\|^2$$
$$\le \|u^* + v^* - z_t\|^2 + \varepsilon$$
$$\quad + 2\langle u_{t+1} - u_t, v_{t+1} - v_t \rangle - 2\langle u^* - u_t, v^* - v_t \rangle$$
$$= \|x^* - z_t\|^2 + \varepsilon + 2\langle u_{t+1} - u_t, v_{t+1} - v_t \rangle$$
$$\quad - 2\langle u^* - u_t, v^* - v_t \rangle.$$

Plugging this into (5), we get:

$$F(x_{t+1}) - F(x^*)$$
$$\le \underbrace{\frac{\beta}{2} \left(\|x^* - z_t\|^2 - \|x_t - z_t\|^2 \right)}_{\mathbb{T}_1}$$
$$+ \underbrace{\beta \left(\langle u_{t+1} - u_t, v_{t+1} - v_t \rangle - 2\langle u^* - u_t, v^* - v_t \rangle \right)}_{\mathbb{T}_2}$$
$$+ \frac{\beta \varepsilon}{2}.$$

We already know how to bound the first term $\mathbb{T}_1$, using an identical argument as in the proof of Theorem 1. We get:

$$\mathbb{T}_1 \le \left(2 - \frac{\beta}{\alpha} \right) (F(x^*) - F(x_t)) + \frac{\beta - \alpha}{\alpha} \gamma \Delta.$$

The second term $\mathbb{T}_2$ can be bounded as follows. First, observe that

$$|\langle u_{t+1} - u_t, v_{t+1} \rangle|$$
$$\le \mu \|u_{t+1} - u_t\| \|v_{t+1} - v_t\|$$
$$\le \frac{\mu}{2} \left(\|u_{t+1} - u_t\|^2 + \|v_{t+1} - v_t\|^2 \right)$$
$$\le \frac{\mu}{2} \left(\|u_{t+1} + v_{t+1} - u_t - v_t\|^2 \right)$$
$$\quad + \mu |\langle u_{t+1} - u_t, v_{t+1} - v_t \rangle|.$$

This gives us the following inequalities:

$$|\langle u_{t+1} - u_t, v_{t+1} \rangle|$$
$$\leq \frac{\mu}{2(1-\mu)} \|x_{t+1} - x_t\|^2$$
$$= \frac{\mu}{2(1-\mu)} \left(\|x_{t+1} - x^*\|^2 + \|x_t - x^*\|^2 + \right.$$
$$\left. 2|\langle x_{t+1} - x^*, x_t - x^* \rangle| \right)$$
$$\leq \frac{\mu}{1-\mu} \left(\|x_{t+1} - x^*\|^2 + \|x_t - x^*\|^2 \right).$$

Similarly,

$$|\langle u^* - u_t, v^* - v_t \rangle| \leq \mu \|u^* - u_t\| \|v^* - v_t\|$$
$$\leq \frac{\mu}{2} \left(\|u^* - u_t\|^2 + \|v^* - v_t\| \right)$$
$$= \frac{\mu}{2} \left(\|u^* + v^* - u_t - v_t\|^2 \right)$$
$$+ \mu |\langle u^* - u_t, v^* - v_t \rangle|,$$

which gives:

$$|\langle u^* - u_t, v^* - v_t \rangle| \leq \frac{\mu}{2(1-\mu)} \|x^* - x_t\|^2.$$

Combining, we get:

$$\mathbb{T}_2 \leq \frac{\beta\mu}{2(1-\mu)} \left(3\|x^* - x_t\|^2 + \|x^* - x_{t+1}\|^2 \right).$$

Moreover, by invoking RSC and Cauchy-Schwartz (similar to the proof of Theorem 1), we have:

$$\|x^* - x_t\|^2 \leq \frac{1}{\alpha} \left(F(x_t) - F(x^*) \right) + O(\varepsilon),$$
$$\|x^* - x_{t+1}\|^2 \leq \frac{1}{\alpha} \left(F(x_{t+1}) - F(x^*) \right) + O(\varepsilon).$$

Therefore we obtain the upper bound on $\mathbb{T}_2$:

$$\mathbb{T}_2 \leq \frac{3\beta\mu}{2\alpha(1-\mu)} \left(F(x_t) - F(x^*) \right)$$
$$+ \frac{\beta\mu}{2\alpha(1-\mu)} \left(F(x_{t+1}) - F(x^*) \right) + C'\varepsilon.$$

Plugging in the upper bounds on $\mathbb{T}_1$ and $\mathbb{T}_2$ and re-arranging terms, we get:

$$\left(1 - \frac{\beta\mu}{2\alpha(1-\mu)} \right) \left(F(x_{t+1}) - F(x^*) \right)$$
$$\leq \left(2 - \frac{\beta}{\alpha} + \frac{3\beta\mu}{2\alpha(1-\mu)} \right) \left(F(x_t) - F(x^*) \right) + C'\varepsilon,$$

which leads to the desired result. ∎

IV. DISCUSSION

We provide some concluding remarks and potential directions for future work.

While our contributions in this paper are primarily theoretical, in recently published work [14] we have explored the practical benefits of our approach in the context of linear inverse problems such as compressive sensing. However, our algorithms proposed in this paper are generic, and can be used to solve a variety of *nonlinear* inverse problems. In future work, we will explore the empirical benefits for nonlinear settings, and also test the efficacy of our myopic PGD algorithm for handling model mismatch.

The main algorithmic message of this paper is to show that solving a variety of nonlinear inverse problems using a generative network model can be reduced to performing a set of ε-projections onto the range of the network model. This can be challenging in general; for most interesting generative networks, this itself is a nonconvex problem, and potentially hard. However, recent work by [38], [39] have studied special cases where this type of projection can be tractable; in particular, for certain neural networks satisfying certain randomness conditions, one can solve the projection problem using a variation of gradient descent (which is more or less what all approaches employ in practice). Studying the landscape of such projection problems is an interesting direction of future research.

We make several assumptions to enable our analysis. Some of them (for example, restricted strong convexity/smoothness; incoherence) are standard analysis tools and are common in the high-dimensional statistics and compressive sensing literature. However, in order to be applicable, they need to be verified for specific problems. A broader characterization of problems that *do* satisfy these assumptions will be of great interest.

REFERENCES

[1] E. Candès et al. Compressive sampling. In *Proc. of the intl. congress of math.*, volume 3, pages 1433–1452. Madrid, Spain, 2006.
[2] R. Baraniuk, V. Cevher, M. Duarte, and C. Hegde. Model-based compressive sensing. *IEEE Trans. Inform. Theory*, 56(4):1982–2001, Apr. 2010.
[3] C. Hegde, P. Indyk, and L. Schmidt. Fast algorithms for structured sparsity. *Bulletin of the EATCS*, 1(117):197–228, Oct. 2015.
[4] M. Duarte, C. Hegde, V. Cevher, and R. Baraniuk. Recovery of compressible signals from unions of subspaces. In *Proc. IEEE Conf. Inform. Science and Systems (CISS)*, March 2009.
[5] M. Elad and M. Aharon. Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Trans. Image Processing*, 15(12):3736–3745, 2006.
[6] M. Aharon, M. Elad, and A. Bruckstein. rmk-svd: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Trans. Signal Processing*, 54(11):4311–4322, 2006.
[7] T. Chan, J. Shen, and H. Zhou. Total variation wavelet inpainting. *Jour. of Math. imaging and Vision*, 25(1):107–125, 2006.
[8] S. Ravishankar and Y. Bresler. Learning sparsifying transforms. *IEEE Trans. Signal Processing*, 61(5):1072–1086, 2013.
[9] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Proc. Adv. in Neural Processing Systems (NIPS)*, pages 2672–2680, 2014.
[10] D. Berthelot, T. Schumm, and L. Metz. Began: Boundary equilibrium generative adversarial networks. *arXiv preprint arXiv:1703.10717*, 2017.
[11] R. Yeh, C. Chen, T. Lim, M. Hasegawa-Johnson, and M. Do. Semantic image inpainting with perceptual and contextual losses. *arXiv preprint arXiv:1607.07539*, 2016.

[12] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. *Proc. IEEE Conf. Comp. Vision and Pattern Recog. (CVPR)*, pages 105–114, 2017.

[13] A. Bora, A. Jalal, E. Price, and A. Dimakis. Compressed sensing using generative models. *Proc. Int. Conf. Machine Learning*, 2017.

[14] V. Shah and C. Hegde. Solving linear inverse problems using gan priors: An algorithm with provable guarantees. In *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Processing (ICASSP)*, Apr. 2018.

[15] M. Dhar, A. Grover, and S. Ermon. Modeling sparse deviations for compressed sensing using generative models. In *Proc. Int. Conf. Machine Learning*, 2018.

[16] C. Hegde and R. Baraniuk. SPIN: Iterative signal recovery on incoherent manifolds. In *Proc. IEEE Int. Symp. Inform. Theory (ISIT)*, July 2012.

[17] C. Hegde and R. Baraniuk. Signal recovery on incoherent manifolds. *IEEE Trans. Inform. Theory*, 58(12):7204–7214, Dec. 2012.

[18] G. Raskutti, M. J Wainwright, and B. Yu. Restricted eigenvalue properties for correlated gaussian designs. *J. Machine Learning Research*, 11(Aug):2241–2259, 2010.

[19] Prateek Jain and Purushottam Kar. Non-convex optimization for machine learning. *Foundations and Trends® in Machine Learning*, 10(3-4):142–336, 2017.

[20] M. McCoy and J. Tropp. Sharp recovery bounds for convex demixing, with applications. *Foundations of Comp. Math.*, 14(3):503–567, 2014.

[21] M. Soltani and C. Hegde. Fast algorithms for demixing signals from nonlinear observations. *IEEE Trans. Sig. Proc.*, 65(16):4209–4222, Aug. 2017.

[22] D. Donoho. De-noising by soft-thresholding. *IEEE Trans. Inform. Theory*, 41(3):613–627, 1995.

[23] Z. Xu and J. Sun. Image inpainting by patch propagation using patch sparsity. *IEEE Trans. Image Processing*, 19(5):1153–1165, 2010.

[24] W. Dong, L. Zhang, G. Shi, and X. Wu. Image deblurring and super-resolution by adaptive sparse domain selection and adaptive regularization. *IEEE Trans. Image Processing*, 20(7):1838–1857, 2011.

[25] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

[26] K. Kulkarni, S. Lohit, P. Turaga, R. Kerviche, and A. Ashok. Reconnet: Non-iterative reconstruction of images from compressively sensed measurements. In *Proc. IEEE Conf. Comp. Vision and Pattern Recog. (CVPR)*, pages 449–458, 2016.

[27] A. Mousavi, A. Patel, and R. Baraniuk. A deep learning approach to structured signal recovery. In *Proc. Allerton Conf. Communication, Control, and Computing*, pages 1336–1343, 2015.

[28] A. Mousavi and R. Baraniuk. Learning to invert: Signal recovery via deep convolutional networks. *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Processing (ICASSP)*, 2017.

[29] L. Xu, J. Ren, C. Liu, and J. Jia. Deep convolutional neural network for image deconvolution. In *Proc. Adv. in Neural Processing Systems (NIPS)*, pages 1790–1798, 2014.

[30] C. Dong, C. Loy, K. He, and X. Tang. Image super-resolution using deep convolutional networks. *IEEE Trans. Pattern Anal. Machine Intell.*, 38(2):295–307, 2016.

[31] J. Kim, J. Kwon Lee, and K. Mu Lee. Accurate image super-resolution using very deep convolutional networks. In *Proc. IEEE Conf. Comp. Vision and Pattern Recog. (CVPR)*, pages 1646–1654, 2016.

[32] R. Yeh, C. Chen, T.-Y. Lim, A. Schwing, M. Hasegawa-Johnson, and M. Do. Semantic image inpainting with deep generative models. In *Proc. IEEE Conf. Comp. Vision and Pattern Recog. (CVPR)*, volume 2, page 4, 2017.

[33] J. Rick Chang, C. Li, B. Poczos, B. Vijaya Kumar, and A. Sankara-narayanan. One network to solve them all–solving linear inverse problems using deep projection models. In *Proc. IEEE Conf. Comp. Vision and Pattern Recog. (CVPR)*, pages 5888–5897, 2017.

[34] M. Arjovsky, S. Chintala, and L. Bottou. Wasserstein gan. *arXiv preprint arXiv:1701.07875*, 2017.

[35] J.-Y. Zhu, T. Park, P. Isola, and A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proc. IEEE Conf. Comp. Vision and Pattern Recog. (CVPR)*, 2017.

[36] A. Brock, J. Donahue, and K. Simonyan. Large scale gan training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*, 2018.

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

[38] P. Hand and V. Voroninski. Global guarantees for enforcing deep generative priors by empirical risk. *arXiv preprint arXiv:1705.07576*, 2017.

[39] R. Heckel, W. Huang, P. Hand, and V. Voroninski. Deep denoising: Rate-optimal recovery of structured signals with a deep prior. *arXiv preprint arXiv:1805.08855*, 2018.