Inexact Gradient Projection and Fast Data Driven Compressed Sensing

Inexact Gradient Projection and Fast Data Driven Compressed Sensing

Mohammad Golbabaee, Mike E. Davies, *Fellow, IEEE,*

Abstract

We study convergence of the iterative projected gradient (IPG) algorithm for arbitrary (possibly non-convex) sets and when both the gradient and projection oracles are computed approximately. We consider different notions of approximation of which we show that the Progressive Fixed Precision (PFP) and the $(1+\varepsilon)$-optimal oracles can achieve the same accuracy as for the exact IPG algorithm. We show that the former scheme is also able to maintain the (linear) rate of convergence of the exact algorithm, under the same embedding assumption. In contrast, the $(1+\varepsilon)$-approximate oracle requires a stronger embedding condition, moderate compression ratios and it typically slows down the convergence.

We apply our results to accelerate solving a class of data driven compressed sensing problems, where we replace iterative exhaustive searches over large datasets by fast approximate nearest neighbour search strategies based on the cover tree data structure. For datasets with low intrinsic dimensions our proposed algorithm achieves a complexity logarithmic in terms of the dataset population as opposed to the linear complexity of a brute force search. By running several numerical experiments we conclude similar observations as predicted by our theoretical analysis.

Index Terms

Iterative projected gradient, approximate updates, linear convergence, compressed sensing, constrained least squares, data driven models, cover trees, approximate nearest neighbour search.

I. Introduction

Signal inference under limited and noisy observations is a major line of research in signal processing, machine learning and statistics and it has a wide application ranging from biomedical imaging, astrophysics, remote sensing to data mining. Incorporating the structure of signals is proven to significantly help with an accurate inference since natural datasets often have limited degrees of freedom as compared to their original ambient dimensionality. This fact has been invoked in Compressed sensing (CS) literature by adopting efficient signal models to achieve accurate signal reconstruction given near-minimal number of measurements i.e. much smaller than the signal ambient dimension (see [1, 2, 3, 4, 5, 6] and e.g. [7] for an overview on different CS models). CS consists of a linear sampling protocol:

$$y \approx Ax^*, \tag{1}$$

where a linear mapping A samples an m-dimensional vector y of noisy measurements from a ground truth signal x^* which typically lives in a high ambient dimension $n \gg m$. Natural signals often have efficient compact representations using nonlinear models such as low dimensional smooth manifolds, low-rank matrices or the Union of Subspaces (UoS) that itself includes sparse (unstructured) or structured sparse (e.g. group, tree or analysis sparsity) representations in properly chosen orthobases or redundant

MG and MED are with the Institute for Digital Communications (IDCOM), School of Engineering, University of Edinburgh, EH9 3JL, United Kingdom. E-mail: {m.golbabaee, mike.davies}@ed.ac.uk. This work is partly funded by the EPSRC grant EP/M019802/1 and the ERC C-SENSE project (ERC-ADG-2015-694888). MED is also supported by the Royal Society Wolfson Research Merit Award.

dictionaries [7]. CS reconstruction algorithms for estimating x^* from y are in general more computationally complex (as opposed to the simple linear acquisition of CS) as they typically require solving a nonlinear optimization problem based around a prior signal model. A proper model should be carefully chosen in order to efficiently promote the low-dimensional structure of signal while not bringing a huge computational burden to the reconstruction algorithm.

Consider the following constrained least square problem for CS reconstruction:

$$\min_{x\in\mathcal{C}}\{f(x) := \frac{1}{2}\|y - Ax\|^2\}, \tag{2}$$

where, the constraint set $\mathcal{C} \in \mathbb{R}^n$ represents the signal model. First order algorithms in the form of projected Landweber iteration a.k.a. iterative projected gradient (IPG) descent or Forward-Backward are very popular for solving (2). Interesting features of IPG include flexibility of handling various and often complicated signal models, e.g. $\mathcal{C}$ might be convex, nonconvex or/and semi-algebraic such as sparsity or rank constraints (these last models result in challenging combinatorial optimization problems but with tractable projection operators). Also IPG (and more generally the proximal-gradient methods) has been considered to be particularly useful for big data applications [8]. It is memory efficient due to using only first order local oracles e.g., the gradient and the projection onto $\mathcal{C}$, it can be implemented in a distributed/parallel fashion, and it is also robust to using cheap statistical estimates e.g. in stochastic descend methods [9] to shortcut heavy gradient computations.

In this regard a major challenge that IPG may encounter is the computational burden of performing an *exact* projection step onto certain complex models (or equivalently, performing an exact but complicated gradient step). In many interesting inverse problems the model projection amounts to solving another optimization within each iteration of IPG. This includes important cases in practice such as the total variation penalized least squares [10, 11], low-rank matrix completion [12] or tree sparse model-based CS [6]. Another example is the convex inclusion constraints $\mathcal{C} = \bigcap_i \mathcal{C}_i$, appearing in multi constrained problems e.g. [13, 14], where one might be required to perform a Dykstra type feasibility algorithm at each iteration [15, 16]. Also, for data driven signal models the projection will typically involve some form of search through potentially large datasets. In all these cases accessing an exact oracle could be either computationally inefficient or even not possible (e.g. in analysis sparse recovery [17] or tensor rank minimization [18] where the exact projection is NP hard), and therefore a natural line of thought is to carry those steps with cheaper *approximate* oracles.

A. Contributions

In this paper we feature an important property of the IPG algorithm; that *it is robust against bounded adversarial (worst-case) errors in calculation of the projection and gradient steps.* We cover different types of oracles: i) A *fixed precision* (FP) oracle which compared to the exact one has an additive bounded approximation error. ii) A *progressive fixed precision* (PFP) oracle which allows for larger (additive) approximations in the earlier iterations and refines the precision as the algorithm progresses. iii) A $(1+\varepsilon)$-approximate oracle which introduces a notion of relatively optimal approximation with a multiplicative error (as compared to the exact oracle).

Our analysis uses a notion of model-restricted bi-Lipschitz *embedding* similar to e.g. [19], however in a more local form and with an improved conditioning (we discuss this in more details in Section IV). With that respect, our analysis differs from the previous related works in the convex settings as the embedding enables us for instance to prove a globally optimal recovery result for nonconvex models, as well as establishing linear rate of convergences for the inexact IPG applied for solving CS problems (e.g. results of [20] on linear convergence of the inexact IPG assumes strong convexity which does not hold in solving underdetermined least squares such as CS).

In summary, we show that the FP type oracles restrict the final accuracy of the main reconstruction problem. This limitation can be overcome by increasing the precision at an appropriate rate using the PFP

type oracles where one could achieve the same solution accuracy as for the exact IPG algorithm under the same embedding assumptions (and even with the convergence rate). We show that the $(1+\varepsilon)$-approximate projection can also achieve the accuracy of exact IPG however under a stronger embedding assumption, moderate compression ratios and using possibly more iterations (since using this type of oracle typically decreases the rate of convergence). In all the cases above we study conditions that provide us with linear convergence results.

Finally we apply this theory to a stylized data driven compressed sensing application that requires a nearest neighbour search order to calculate the model projection. We shortcut the computations involved, (iteratively) performing exhaustive searches over large datasets, by using approximate nearest neighbour search strategies corresponding to the aforementioned oracles and motivated by the cover tree structure introduced in [21]. Our proposed algorithm achieves a complexity logarithmic in terms of the dataset population (as opposed to the linear complexity of a brute force search). By running several numerical experiments on different datasets we conclude similar observations as predicted by our theoretical results.

B. Paper organization

The rest of this paper is organized as follows: In Section II we review and compare our results to the previous related works on inexact IPG. In Section III we define the inexact IPG algorithm for three types of approximate oracles. Section IV includes our main theoretical results on robustness and linear convergence of the inexact IPG for solving CS reconstruction problem. In Section VI we discuss an application of the proposed inexact algorithms to accelerate solving data driven CS problems. We also briefly discuss the cover tree data structure and the associated exact/approximate search strategies. Section VII is dedicated to the numerical experiments on using inexact IPG for data driven CS. And finally we discuss and conclude our results in Section VIII.

II. Related works

Inexact proximal-gradient methods (in particular IPG) and their Nesterov accelerated variants have been the subject of a substantial amount of work in convex optimization [20, 22, 23, 24, 25]. Here we review some highlights and refer the reader for a comprehensive literature review to [20]. Fixed precision approximates have been studied for the gradient step e.g. for using the smoothing techniques where the gradient is not explicit and requires solving an auxiliary optimization, see for more details [22] and [23] for a semi-definite programming example in solving sparse PCA. A similar approach has been extensively applied for carrying out the projection (or the proximity operator) approximately in cases where it does not have an analytical expression and requires solving another optimization within each iteration e.g. in total variation constrained inverse problems [10, 11] or the overlapping group Lasso problem [26]. Fortunately in convex settings one can stop the inner optimization when its duality gap falls below a certain threshold and achieve a fixed precision approximate projection. In this case the solution accuracy of the main problem is proportional to the approximation level introduced within each iteration. Recently [20] studied the progressive fixed precision (PFP) type approximations for solving convex problems, e.g. a sparse CUR type factorization, via gradient-proximal (and its accelerated variant) methods. The authors show that IPG can afford larger approximation errors (in both gradient and projection/proximal steps) in the earlier stages of the algorithm and by increasing the approximation accuracy at an appropriate rate one can attain a similar convergence rate and accuracy as for the exact IPG however with significantly less computation.

A part of our results draws a similar conclusion for solving nonconvex constrained least squares that appears in CS problems by using inexact IPG. Note that an FP type approximation has been previously considered for the nonconvex IPG e.g. for the UoS [19] or the manifold [27] CS models, however these works only assume inexact projections whereas our result covers an approximate gradient step as well. Of more importance and to the best of our knowledge, our results on incorporating the PFP type

approximation in nonconvex IPGs and analysing the associated rate of (global) convergence is the first result of its kind. We also note that the results in [20] are mainly about minimizing convex cost functions (i.e. not necessarily recovery guarantees) and that the corresponding linear convergence results only hold for uniformly strong convex objectives. We instead cover cases with local (and not uniform) strong convexity and establish the linear convergence of IPG for solving underdetermined inverse problems such as CS.

Using relative $(1+\varepsilon)$-approximate projections (described in Section IV-D) for CS recovery has been subject of more recent research activities (and mainly for nonconvex models). In [17] the authors studied this type of approximation for an IPG sparse recovery algorithm under the UoS model in redundant dictionaries. Our result encompasses this as a particular choice of model and additionally allows for inexactness in the gradient step. The work of [28] studied similar inexact oracles for a stochastic gradient-projection type algorithm customized for sparse UoS and low-rank models (see also [29] for low-rank CS using an accelerated variant of IPG). We share a similar conclusion with those works; for a large ε more measurements are required for CS recovery and the convergence becomes slow. Hegde et al. [30] proposed such projection oracle for tree-sparse signals and use it for the related model-based CS problem using a CoSamp type algorithm (see also [31, 32] for related works on inexact CoSamp type algorithms). In a later work [33] the authors consider a modified variant of IPG with $(1+\varepsilon)$-approximate projections for application to structured sparse reconstruction problems (specifically tree-sparse and earth-mover's distance CS models). For this scenario they are able to introduce a modified gradient estimate (called the Head approximation oracle) to strengthen the recovery guarantees by removing the dependency on ε, albeit with a more conservative Restricted Isometry Property. Unfortunately, this technique does not immediately generalize to arbitrary signal models and we therefore do not pursue this line of research here.

III. Preliminaries

Iterative projected gradient iterates between calculating the gradient and projection onto the model i.e. for positive integers k the *exact* form of IPG follows:

$$x^k = \mathcal{P}_{\mathcal{C}}\left(x^{k-1} - \mu \nabla f(x^{k-1})\right) \tag{3}$$

where, μ is the step size, $\nabla f(x) = A^T(Ax-y)$ and $\mathcal{P}_{\mathcal{C}}$ denote the exact gradient and the Euclidean projection oracles, respectively. The exact IPG requires the constraint set $\mathcal{C}$ to have a well defined, not necessarily unique but computationally tractable Euclidean projection $\mathcal{P}_{\mathcal{C}} : \mathbb{R}^n \to \mathcal{C}$

$$\mathcal{P}_{\mathcal{C}}(x) \in \operatorname{argmin}_{u\in\mathcal{C}} \|u - x\|.$$

Throughout we use $\|.\|$ as a shorthand for the Euclidean norm $\|.\|_{\ell_2}$.

In the following we define three types of approximate oracles which frequently appear in the literature and could be incorporated within the IPG iterations. We also briefly discuss their applications. Each of these approximations are applicable to the data driven problem we will consider in Section VI.

A. Fixed Precision (FP) approximate oracles

We first consider approximate oracles with *additive* bounded-norm errors, namely the fixed precision gradient oracle $\widetilde{\nabla}^{\nu_g} f(x) : \mathbb{R}^n \to \mathbb{R}^n$ where:

$$\|\widetilde{\nabla}^{\nu_g} f(x) - \nabla f(x)\| \leqslant \nu_g, \tag{4}$$

and the fixed precision projection oracle $\mathcal{P}_{\mathcal{C}}^{\nu_p} : \mathbb{R}^n \to \mathcal{C}$ where:

$$\mathcal{P}_{\mathcal{C}}^{\nu_p}(x) \in \left\{u \in \mathcal{C} : \|u-x\|^2 \leqslant \inf_{u'\in\mathcal{C}} \|u'-x\|^2 + \nu_p^2\right\}. \tag{5}$$

The values of ν_g, ν_p denote the levels of inaccuracy in calculating the gradient and projection steps respectively.[1] The corresponding *inexact* IPG iterates as follows:

$$x^k = \mathcal{P}_{\mathcal{C}}^{\nu_p^k}\left(x^{k-1} - \mu\widetilde{\nabla}^{\nu_g^k} f(x^{k-1})\right). \tag{6}$$

Note that, unlike [19, 27] in this formulation we allow for variations in the inexactness levels at different stages of IPG. The case where the accuracy levels are bounded by a constant threshold $\nu_p^k = \nu_p$ and $\nu_g^k = \nu_g$, $\forall k$, refers to an inexact IPG algorithm with *fixed precision* (FP) approximate oracles.

Examples: Such errors may occur for instance in distributed network optimizations where the gradient calculations could be noisy during the communication on the network, or in CS under certain UoS models with infinite subspaces [19] where an exact projection might not exist by definition (e.g. when $\mathcal{C}$ is an open set) however an FP type approximation could be achievable. It also has application in finite (discrete) super resolution [34], source localization and separation [35, 36] and data driven CS problems e.g., in Magnetic Resonance Fingerprinting [37, 38], where typically a continuous manifold is discretized and approximated by a large dictionary for e.g. sparse recovery tasks.

B. Progressive Fixed Precision (PFP) approximate oracles

One obtains a *Progressive Fixed Precision* (PFP) approximate IPG by refining the FP type precisions thorough the course of iterations. Therefore any FP gradient or projection oracle which has control on tuning the accuracy parameter could be used in this setting and follows (6) with decaying sequences ν_p^k, ν_g^k.

Examples: For instance this case includes projection schemes that require iteratively solving an auxiliary optimization (e.g. the total variation ball [11], sparse CUR factorization [20] or the multi-constraint inclusions [15, 16], etc) and can progress (at an appropriate rate) on the accuracy of their solutions by adding more subiterations. We also discuss in Section VI-C another example of this form which is customized for fast approximate nearest neighbour searches with application to the data driven CS framework.

C. $(1+\varepsilon)$-approximate projection

Obtaining a fixed precision (and thus PFP) accuracy in projections onto certain constraints might be still computationally exhaustive, whereas a notion of relative optimality could be more efficient to implement. The $(1+\varepsilon)$-approximate projection is defined as follows: for a given $\varepsilon \geqslant 0$,

$$\mathcal{P}_{\mathcal{C}}^{\varepsilon}(x) \in \Big\{u \in \mathcal{C} : \|u - x\| \leqslant (1+\varepsilon) \inf_{u' \in \mathcal{C}} \|u' - x\|\Big\}. \tag{7}$$

We note that $\mathcal{P}_{\mathcal{C}}^{\varepsilon}(x)$ might not be unique. In this regard, the inexact IPG algorithm with a $(1+\varepsilon)$-approximate projection takes the following form:

$$x^k = \mathcal{P}_{\mathcal{C}}^{\varepsilon}\left(x^{k-1} - \mu\widetilde{\nabla}^{\nu_g^k} f(x^{k-1})\right). \tag{8}$$

Note that we still assume a fixed precision gradient oracle with flexible accuracies ν_g^k. One could also consider a $(1+\varepsilon)$-approximate gradient oracle and in combination with those aforementioned inexact projections however for brevity and since the analysis would be quite similar to the relative approximate projection we decide to skip more details on this case.

[1]Note that our fixed precision projection (5) is defined on the squared norms in a same way as in e.g. [19, 27].

Examples: The tree s-sparse projection in $\mathbb{R}^n$ and in the exact form requires solving a dynamical programming problem with $O(ns)$ running time [39] whereas solving this problem approximately with $1+\varepsilon$ accuracy requires the time complexity $O(n\log(n))$ [30] which better suits imaging problems in practice with a typical Wavelet sparsity level $s=\Omega(\log(n))$. Also [28, 29] show that one can reduce the cost of low-rank matrix completion problem by using randomized linear algebra methods, e.g. see [40, 41], and carry out fast low-rank factorizations with a $1+\varepsilon$ type approximation. In a recent work on low-rank tensor CS [18] authors consider using a $1+\varepsilon$ approximation (within an IPG algorithm) for low-rank tensor factorization since the exact problem is generally NP hard [42]. Also in Section VI-C we discuss a data driven CS recovery algorithm which uses $1+\varepsilon$ approximate nearest neighbour searches in order to break down the complexity of the projections from $O(d)$ (using an exhaustive search) to $O(\log(d))$, for d denoting a large number of data points with low intrinsic dimensionality.

IV. Main results

A. Uniform linear embeddings

The success of CS paradigm heavily relies on the embedding property of certain random sampling matrices which preserves signal information for low dimensional but often complicated/combinatorial models. It has been shown that IPG can stably predict the true signal x^* from noisy CS measurements provided that A satisfies the so called Restricted Isometry Property (RIP):

$$(1-\theta)\|x-x'\|^2 \leqslant \|A(x-x')\|^2 \leqslant (1+\theta)\|x-x'\|^2, \quad \forall x,x'\in\mathcal{C} \tag{9}$$

for a small constant $0<\theta<1$. This has been shown for models such as sparse, low-rank and low-dimensional smooth manifold signals and by using IPG type reconstruction algorithms which in the nonconvex settings are also known as Iterative Hard Thresholding [6, 12, 27, 43, 44]. Interestingly these results indicate that under the RIP condition (and without any assumption on the initialization) the first order IPG algorithms with cheap local oracles can globally solve nonconvex optimization problems.

For instance random orthoprojectors and i.i.d. subgaussian matrices A satisfy RIP when the number of measurements m is proportional to the intrinsic dimension of the model (i.e. signal sparsity level, rank of a data matrix or the dimension of a smooth signal manifold, see e.g. [7] for a review on comparing different CS models and their measurement complexities) and sublinearly scales with the ambient dimension n.

A more recent work generalizes the theory of IPG to arbitrary *bi-Lipschitz embeddable* models [19], that is for given $\mathcal{C}$ and A it holds

$$\alpha\|x-x'\|^2 \leqslant \|A(x-x')\|^2 \leqslant \beta\|x-x'\|^2 \quad \forall x,x'\in\mathcal{C}.$$

for some constants $\alpha,\beta>0$. Similar to the RIP these constants are defined *uniformly* over the constraint set i.e. $\forall x,x'\in\mathcal{C}$. There Blumensath shows that if

$$\beta<1.5\alpha,$$

then IPG robustly solves the corresponding noisy CS reconstruction problem *for all* $x^*\in\mathcal{C}$. This result also relaxes the RIP requirement to a nonsymmetric and unnormalised notion of linear embedding whose implication in deriving sharper recovery bounds is previously studied by [45].

B. Hybrid (local-uniform) linear embeddings

Similarly the notion of restricted embedding plays a key role in our analysis. However we adopt a more local form of embedding and show that it is still able to guarantee stable CS reconstruction.

Main assumption. *Given* $(x_0\in\mathcal{C},\mathcal{C},A)$ *there exist constants* $\beta,\alpha_{x_0}>0$ *for which the following inequalities hold:*

- *Uniform Upper Lipschitz Embedding (ULE)*

$$\|A(x-x')\|^2 \leqslant \beta\|x-x'\|^2 \quad \forall x, x' \in \mathcal{C}$$

- *Local Lower Lipschitz Embedding (LLE)*

$$\|A(x-x_0)\|^2 \geqslant \alpha_{x_0}\|x-x_0\|^2 \quad \forall x \in \mathcal{C}$$

Upon existence, β and α_{x_0} denote respectively the smallest and largest constants for which the inequalities above hold.

This is a weaker assumption compared to RIP or the uniform bi-Lipschitz embedding. Note that for any $x_0 \in \mathcal{C}$ we have:

$$\alpha \leqslant \alpha_{x_0} \leqslant \beta \leqslant |||A|||^2$$

(where $|||.|||$ denotes the matrix spectral norm i.e. the largest singular value). However with such an assumption one has to sacrifice the *universality* of the RIP-dependent results for a signal x^* dependent analysis. Depending on the study, local analysis could be very useful to avoid e.g. worst-case scenarios that might unnecessarily restrict the recovery analysis [46]. Similar local assumptions in the convex settings are shown to improve the measurement bound and the speed of convergence up to very sharp constants [47, 48].

Unfortunately we are currently unable to make the analysis fully local as we require the uniform ULE constraint. Nonetheless, one can always plug the stronger bi-Liptchitz assumption into our results throughout (i.e. replacing α_{x^*} with α) and regain the universality.

C. Linear convergence of (P)FP inexact IPG for CS recovery

In this section we show that IPG is robust against deterministic (worst case) errors. Moreover, we show that for certain decaying approximation errors, the IPG solution maintains the same accuracy as for the approximation-free algorithm.

Theorem 1. *Assume $(x^* \in \mathcal{C}, \mathcal{C}, A)$ satisfy the main Lipschitz assumption with constants $\beta < 2\alpha_{x^*}$. Set the step size $(2\alpha_{x^*})^{-1} < \mu \leqslant \beta^{-1}$. The sequence generated by Algorithm* (6) *obeys the following bound:*

$$\|x^k - x^*\| \leqslant \rho^k \left(\|x^*\| + \sum_{i=1}^{k} \rho^{-i} \mathbf{e}^i \right) + \frac{2\sqrt{\beta}}{\alpha_{x^*}(1-\rho)} w \tag{10}$$

where

$$\rho = \sqrt{\frac{1}{\mu\alpha_{x^*}} - 1} \quad \textit{and} \quad \mathbf{e}^i = \frac{2\nu_g^i}{\alpha_{x^*}} + \frac{\nu_p^i}{\sqrt{\mu\alpha_{x^*}}},$$

and $w = \|y - Ax^*\|$.

Remark 1. Theorem 1 implications for the exact IPG (i.e. $\nu_p^k = \nu_g^k = 0$) and inexact FP approximate IPG (i.e. $\nu_p^k = \nu_p, \nu_g^k = \nu_g, \forall k$) improve [19, Theorem 2] in three ways: first by relaxing the uniform lower Lipschitz constant α to a local form $\alpha_{x^*} \geqslant \alpha$ with the possibility of conducting a local recovery/convergence analysis. Second, by improving the embedding condition for CS stable recovery to

$$\beta < 2\alpha_{x^*}, \tag{11}$$

or $\beta < 2\alpha$ for a uniform recovery $\forall x^* \in \mathcal{C}$ (c.f. $\beta < 1.5\alpha$ in [19]). And third, by improving the rate ρ of convergence.

The following corollary is an immediate consequence of the linear convergence result established in Theorem 1 for which we do not provide a proof:

Corollary 1. *With assumptions as in Theorem 1 the IPG algorithm with FP approximate oracles achieves the solution accuracy*

$$\|x^K - x^*\| \leqslant \frac{1}{1-\rho}\left(\frac{2\nu_g}{\alpha_{x^*}} + \frac{\nu_p}{\sqrt{\mu\alpha_{x^*}}} + \frac{2\sqrt{\beta}}{\alpha_{x^*}}w\right) + \tau$$

for any $\tau > 0$ and in a finite number of iterations

$$K = \left\lceil \frac{1}{\log(\rho^{-1})}\log\left(\frac{\|x^*\|}{\tau}\right)\right\rceil$$

As it turns out in our experiments and aligned with the result of Corollary 1, the solution accuracy of IPG can not exceed the precision level introduced by a FP oracle. In this sense Corollary 1 is tight as a trivial converse example would be that IPG starts from the optimal solution x^* but an adversarial FP scheme projects it to another point within a fixed distance.

Interestingly one can deduce another implication from Theorem 1 and overcome such limitation by using a PFP type oracle. Remarkably one achieves a linear convergence to a solution with the same accuracy as for the exact IPG, as long as $\mathbf{e}^k$ geometrically decays. The following corollary makes this statement explicit:

Corollary 2. *Assume $\mathbf{e}^k \leqslant Cr^k$ for some error decay rate $0 < r < 1$ and a constant C. Under the assumptions of Theorem 1 the solution updates $\|x^k - x^*\|$ of the IPG algorithm with PFP approximate oracles is bounded above by:*

$$\begin{cases} \max(\rho,r)^k\left(\|x^*\| + \dfrac{C}{1-\frac{\min(\rho,r)}{\max(\rho,r)}}\right) + \dfrac{2\sqrt{\beta}}{\alpha_{x^*}(1-\rho)}w, & r \neq \rho \\ \rho^k\Big(\|x^*\| + Ck\Big) + \dfrac{2\sqrt{\beta}}{\alpha_{x^*}(1-\rho)}w, & r = \rho \end{cases}$$

Which implies a linear convergence at rate

$$\bar{\rho} = \begin{cases} \max(\rho,r) & r \neq \rho \\ \rho + \xi & r = \rho \end{cases}$$

for an arbitrary small $\xi > 0$.

Remark 2. Similar to Corollary 1 one can increase the final solution precision of the PFP type IPG with logarithmically more iterations i.e. in a finite number $K = O(\log(\tau^{-1}))$ of iterations one achieves $\|x^K - x^*\| \leqslant O(w) + \tau$. Therefore in contrast with the FP oracles one achieves an accuracy within the noise level $O(w)$ that is the precision of an approximation-free IPG.

Remark 3. Using the PFP type oracles can also maintain the rate of linear convergence identical as for the exact IPG. For this the approximation errors suffice to follow a geometric decaying rate of $r < \rho$.

Remark 4. The embedding condition (11) sufficient to guarantee our stability results is invariant to the precisions of the FP/PFP oracles and it is the same as for an exact IPG.

D. Linear convergence of inexact IPG with $(1+\varepsilon)$-approximate projection for CS recovery

In this part we focus on the inexact algorithm (8) with a $(1+\varepsilon)$-approximate projection. As it turns out by the following theorem we require a stronger embedding condition to guarantee the CS stability compared to the previous algorithms.

Theorem 2. *Assume $(x^* \in \mathcal{C}, \mathcal{C}, A)$ satisfy the main Lipschitz assumption and that*

$$\sqrt{2\varepsilon + \varepsilon^2} \leqslant \delta\frac{\sqrt{\alpha_{x^*}}}{|||A|||} \quad \textit{and} \quad \beta < (2 - 2\delta + \delta^2)\alpha_{x^*}$$

for $\varepsilon \geqslant 0$ and some constant $\delta \in [0,1)$. Set the step size $\left((2-2\delta+\delta^2)\alpha_{x^*}\right)^{-1} < \mu \leqslant \beta^{-1}$. The sequence generated by Algorithm (8) *obeys the following bound:*

$$\|x^k - x^*\| \leqslant \rho^k \left(\|x^*\| + \kappa_g \sum_{i=1}^{k} \rho^{-i} \nu_g^i \right) + \frac{\kappa_w}{1-\rho} w$$

where

$$\rho = \sqrt{\frac{1}{\mu\alpha_{x^*}} - 1} + \delta, \quad \kappa_g = \frac{2}{\alpha_{x^*}} + \frac{\sqrt{\mu}}{|||A|||}\delta,$$
$$\kappa_w = 2\frac{\sqrt{\beta}}{\alpha_{x^*}} + \sqrt{\mu}\delta, \quad \textit{and} \quad w = \|y - Ax^*\|.$$

Remark 5. Similar conclusions follow as in Corollaries 1 and 2 on the linear convergence, logarithmic number of iterations vs. final level of accuracy (depending whether the gradient oracle is exact or FP/PFP) however with a stronger requirement than (11) on the embedding; increasing ε i.e. consequently δ, limits the recovery guarantee and slows down the convergence (compare ρ in Theorems 1 and 2). Also approximations of this type result in amplifying distortions (i.e. constants κ_w, κ_g) due to the measurement noise and gradient errors. For example for an exact or a geometrically decaying PFP gradient updates and for a fixed ε chosen according to the Theorem 2 assumptions, Algorithm (8) achieves a full precision accuracy $\|x^K - x^*\| \leqslant O(w) + \tau$ (similar to the exact IPG) in a finite number $K = O(\log(\tau^{-1}))$ of iterations.

Remark 6. The assumptions of Theorem 2 impose a stringent requirement on the scaling of the approximation parameter i.e. $\varepsilon = O\left(\sqrt{\frac{\alpha_{x^*}}{|||A|||}}\right)$ which is not purely dependent on the model-restricted embedding condition but also on the spectral norm $|||A|||$. In this sense since $|||A|||$ ignores the structure $\mathcal{C}$ of the problem it might scale very differently than the corresponding embedding constants α_{x^*}, α and β. For instance an $m \times n$ i.i.d. Gaussian matrix has w.h.p. $|||A||| = \Theta(n)$ (when $m \ll n$) whereas, e.g. for sparse signals, the embedding constants α, β w.h.p. scale as $O(m)$. A similar gap exists for other low dimensional signal models and for other compressed sensing matrices e.g. random orthoprojectors. This indicates that the $1+\varepsilon$ oracles may be sensitive to the CS sampling ratio i.e. for $m \ll n$ we may be limited to use very small approximations $\varepsilon = O(\sqrt{\frac{m}{n}})$.

In the following we show by a deterministic example that this requirement is indeed tight. We also empirically observe in Section VII that such a limitation indeed holds in randomized settings (e.g. i.i.d. Gaussian A) and on average. Although it would be desirable to modify the IPG algorithm to avoid such restriction, as was done in [33] for specific structured sparse models, we note that this is the same term that appears due to 'noise folding' when the signal model is not exact or when there is noise in the signal domain (see the discussion in Section IV-E). As such most practical CS systems will inevitably have to avoid regimes of extreme undersampling.

A converse example: Consider a noiseless CS recovery problem where $n = 2$, $m = 1$ and the sampling matrix (i.e. here a row vector) is

$$A = [\cos(\gamma) \quad -\sin(\gamma)]$$

for some parameter $0 \leqslant \gamma < \pi/2$. Consider the following one-dimensional signal model along the first coordinate:

$$\mathcal{C} = \{x \in \mathbb{R}^2 : x(1) \in \mathbb{R},\, x(2) = 0\}.$$

We have indeed $|||A||| = 1$. It is easy to verify that both of the embedding constants w.r.t. to $\mathcal{C}$ are

$$\alpha_{x^*} = \beta = \cos(\gamma)^2.$$

Therefore one can tune $\gamma \to \pi/2$ to obtain arbitrary small ratios for $\sqrt{\frac{\alpha_{x^*}}{|||A|||}} = \cos(\gamma)$.

Assume the true signal, the corresponding CS measurement, and the initialization point are

$$x^* = [1 \quad 0]^T, \quad y = Ax^* = \cos(\gamma) \quad \text{and} \quad x^0 = [0 \quad 0]^T.$$

Consider an adversarial $(1+\varepsilon)$-approximate projection oracle which performs the following step for any given $x \in \mathbb{R}^2$:

$$\mathcal{P}_{\mathcal{C}}^{\varepsilon}(x) := [x(1) + \varepsilon x(2) \quad 0]^T.$$

For simplicity we assume no errors on the gradient step. By setting $\mu = 1/\cos(\gamma)^2$, the corresponding inexact IPG updates as

$$x^k(1) = 1 + \varepsilon \tan(\gamma) \left(x^{k-1}(1) - 1 \right)$$

and only along the first dimension (we note that due to the choice of oracle $x^k(2) = 0, \forall k$). Therefore we have

$$x^k(1) = 1 - (\varepsilon \tan(\gamma))^k .$$

which requires $\varepsilon < 1/\tan(\gamma) = O(\cos(\gamma))$ for convergence, and it diverges otherwise. As we can see for $\gamma \to \pi/2$ (i.e. where A becomes extremely unstable w.r.t. sampling the first dimension) the range of admissible ε shrinks, regardless of the fact that the restricted embedding $\alpha_{x^*} = \beta$ exhibits a perfect isometry; which is an ideal situation for solving a noiseless CS (i.e. in this case an exact IPG takes only one iteration to converge).

E. When the projection is not onto the signal model

One can also make a distinction between the projection set $\mathcal{C}$ and the signal model here denoted as $\mathcal{C}'$ (i.e. $x^* \in \mathcal{C}'$) by modifying our earlier definitions (5) and (7) in the following ways: an approximate FP projection reads,

$$\mathcal{P}_{\mathcal{C}}^{\nu_p}(x) \in \left\{ u \in \mathcal{C} : \|u - x\|^2 \leqslant \inf_{u' \in \mathcal{C}'} \|u' - x\|^2 + \nu_p^2 \right\}, \tag{12}$$

and a $(1+\varepsilon)$-approximate projection reads

$$\mathcal{P}_{\mathcal{C}}^{\varepsilon}(x) \in \left\{ u \in \mathcal{C} : \|u - x\| \leqslant (1+\varepsilon) \inf_{u' \in \mathcal{C}'} \|u' - x\| \right\}. \tag{13}$$

This enables us to project onto relaxed sets that are explicitly larger than the signal model and may therefore be achieved with less computation but with slightly weaker recovery and increased distortion (c.f. the notion of 'algorithmic weakening' in [49]). Alternatively one could project onto a smaller set that may not contain x^*. Such a strategy may well improve the embedding conditions but introduce an intrinsic bias into the reconstruction estimate.

With respect to such a distinction, Theorems 1 and 2 still hold (with the same embedding assumption/constants on the projection set $\mathcal{C}$), conditioned that $x^* \in \mathcal{C}$. Indeed this can be verified by following identical steps as in the proof of both theorems. This allows our analysis to cover throughout an approximate projection onto a possibly larger set $\mathcal{C}$ including the original signal model $\mathcal{C}'$ i.e. $x^* \in \mathcal{C}' \subseteq \mathcal{C}$, which for instance finds application in fast tree-sparse signal or low-rank matrix CS recovery [29, 30]. Such an inclusion is also important to derive a uniform recovery result *for all* $x^* \in \mathcal{C}'$.

The case where $x^* \notin \mathcal{C}$ can also be bounded in a similar fashion as in [19]. We first consider a proximity point in $\mathcal{C}$ i.e. $x^o := \operatorname{argmin}_{u \in \mathcal{C}} \|x^* - u\|$ and update the noise term to $w := \|y - Ax^o\| \leqslant \|y - Ax^*\| + \|A(x^* - x^o)\|$. We then use Theorems 1 and 2 to derive an error bound, here on $\|x^k - x^o\|$. For this we assume the embedding condition *uniformly* holds over the projection set $\mathcal{C}$ (which includes x^o). As a result we get a bound on the error $\|x^k - x^*\| \leqslant \|x^* - x^o\| + \|x^k - x^o\|$ which includes a bias term with respect to the distance of x^* to $\mathcal{C}$. Note that since here w also includes a signal (and not only measurement) noise term introduced by $\|A(x^* - x^o)\|$, the results are subjected to *noise folding* i.e. a noise amplification with a similar unfavourable scaling (when $m \ll n$) to our discussion in Remark 6 (for more details on CS noise folding see e.g. [50, 51]).

V. PROOFS

A. Proof of Theorem 1

We start from a similar argument as in [19, proof of Therorem 2]. Set $g := 2\nabla f(x^{k-1}) = 2A^T(Ax^{k-1} - y)$ and $\widetilde{g} := 2\widetilde{\nabla}^{\nu_g} f(x^{k-1}) = g + 2e_g^k$ for some vector e_g^k which by definition (4) is bounded $\|e_g^k\| \leqslant \nu_g^k$. It follows that

$$\begin{aligned}\|y - Ax^k\|^2 - \|y - Ax^{k-1}\|^2 &= \langle x^k - x^{k-1}, g\rangle + \|A(x^k - x^{k-1})\|^2 \\ &\leqslant \langle x^k - x^{k-1}, g\rangle + \beta\|x^k - x^{k-1}\|^2,\end{aligned}$$

where the last inequality follows from the ULE property in Definition IV-B. Assuming $\beta \leqslant 1/\mu$, we have

$$\begin{aligned}\langle x^k - x^{k-1}, g\rangle + \beta\|x^k - x^{k-1}\|^2 &\leqslant \langle x^k - x^{k-1}, g\rangle + \frac{1}{\mu}\|x^k - x^{k-1}\|^2 \\ &= \langle x^k - x^{k-1}, \widetilde{g}\rangle + \frac{1}{\mu}\|x^k - x^{k-1}\|^2 - \langle x^k - x^{k-1}, 2e_g^k\rangle \\ &= \frac{1}{\mu}\|x^k - x^{k-1} + \frac{\mu}{2}\widetilde{g}\|^2 - \frac{\mu}{4}\|\widetilde{g}\|^2 - \langle x^k - x^{k-1}, 2e_g^k\rangle.\end{aligned}$$

Due to the update rule of Algorithm (6) and the inexact (fixed-precision) projection step, we have

$$\begin{aligned}\|x^k - x^{k-1} + \frac{\mu}{2}\widetilde{g}\|^2 &\leqslant \|\mathcal{P}_{\mathcal{C}}(x^{k-1} - \frac{\mu}{2}\widetilde{g}) - x^{k-1} + \frac{\mu}{2}\widetilde{g}\|^2 + (\nu_p^k)^2 \\ &\leqslant \|x^* - x^{k-1} + \frac{\mu}{2}\widetilde{g}\|^2 + (\nu_p^k)^2. \qquad (14)\end{aligned}$$

The last inequality holds for any member of $\mathcal{C}$ and thus here for x^*. Therefore we can write

$$\begin{aligned}\|y - Ax^k\|^2 - \|y - Ax^{k-1}\|^2 &\leqslant \frac{1}{\mu}\|x^* - x^{k-1} + \frac{\mu}{2}\widetilde{g}\|^2 - \frac{\mu}{4}\|\widetilde{g}\|^2 - \langle x^k - x^{k-1}, 2e_g^k\rangle + (\frac{\nu_p^k}{\sqrt{\mu}})^2 \\ &= \langle x^* - x^{k-1}, \widetilde{g}\rangle + \frac{1}{\mu}\|x^* - x^{k-1}\|^2 - \langle x^k - x^{k-1}, 2e_g^k\rangle + (\frac{\nu_p^k}{\sqrt{\mu}})^2 \\ &\leqslant \langle x^* - x^{k-1}, g\rangle + \frac{1}{\mu}\|x^* - x^{k-1}\|^2 + 2\nu_g^k\|x^k - x^*\| + (\frac{\nu_p^k}{\sqrt{\mu}})^2. \qquad (15)\end{aligned}$$

The last line replaces $\widetilde{g} = g + 2e_g^k$ and uses the Cauchy-Schwartz inequality.

Similarly we use the LLE property in Definition IV-B to obtain an upper bound on $\langle x^* - x^{k-1}, g\rangle$:

$$\begin{aligned}\langle x^* - x^{k-1}, g\rangle &= w^2 - \|y - Ax^{k-1}\|^2 - \|A(x^* - x^{k-1})\|^2 \\ &\leqslant w^2 - \|y - Ax^{k-1}\|^2 - \alpha_{x^*}\|x^* - x^{k-1}\|^2,\end{aligned}$$

where $w = \|y - Ax^*\|$. Replacing this bound in (15) and cancelling $-\|y - Ax^{k-1}\|^2$ from both sides of the inequality yields

$$\|y - Ax^k\|^2 - 2\nu_g^k\|x^k - x^*\| \leqslant \left(\frac{1}{\mu} - \alpha_{x^*}\right)\|x^{k-1} - x^*\|^2 + (\frac{\nu_p^k}{\sqrt{\mu}})^2 + w^2. \qquad (16)$$

We continue by lower-bounding the left-hand side of this inequality:

$$\begin{aligned}\|y - Ax^k\|^2 - 2\nu_g^k\|x^k - x^*\| &= \|A(x^k - x^*)\|^2 + w^2 - 2\langle y - Ax^*, A(x^k - x^*)\rangle - 2\nu_g^k\|x^k - x^*\| \\ &\geqslant \|A(x^k - x^*)\|^2 + w^2 - 2w\|A(x^k - x^*)\| - 2\nu_g^k\|x^k - x^*\| \\ &\geqslant \alpha_{x^*}\|x^k - x^*\|^2 + w^2 - 2(w\sqrt{\beta} + \nu_g^k)\|x^k - x^*\| \\ &= \left(\sqrt{\alpha_{x^*}}\|x^k - x^*\| - \frac{\nu_g^k}{\sqrt{\alpha_{x^*}}} - \sqrt{\frac{\beta}{\alpha_{x^*}}}w\right)^2 - \left(\frac{\nu_g^k}{\sqrt{\alpha_{x^*}}} + \sqrt{\frac{\beta}{\alpha_{x^*}}}w\right)^2 + w^2.\end{aligned}$$

The first inequality uses the Cauchy-Schwartz's and the second inequality follows from the ULE and LLE properties. Using this bound together with (16) we get

$$\left(\sqrt{\alpha_{x^*}}\|x^k-x^*\|-\frac{\nu_g^k}{\sqrt{\alpha_{x^*}}}-\sqrt{\frac{\beta}{\alpha_{x^*}}}w\right)^2 \leqslant \left(\frac{1}{\mu}-\alpha_{x^*}\right)\|x^{k-1}-x^*\|^2+(\frac{\nu_p^k}{\sqrt{\mu}})^2+\left(\frac{\nu_g^k}{\sqrt{\alpha_{x^*}}}+\sqrt{\frac{\beta}{\alpha_{x^*}}}w\right)^2$$
$$\leqslant \left(\sqrt{\frac{1}{\mu}-\alpha_{x^*}}\|x^{k-1}-x^*\|+\frac{\nu_g^k}{\sqrt{\alpha_{x^*}}}+\frac{\nu_p^k}{\sqrt{\mu}}+\sqrt{\frac{\beta}{\alpha_{x^*}}}w\right)^2.$$

The last inequality assumes $\mu \leqslant \alpha_{x^*}^{-1}$ which holds since we previously assumed $\mu \leqslant \beta^{-1}$ (and by definition $\alpha_{x^*} \leqslant \beta$). As a result we deduce that

$$\|x^k-x^*\| \leqslant \rho\|x^{k-1}-x^*\|+\mathbf{e}^k+2\frac{\sqrt{\beta}}{\alpha_{x^*}}w \tag{17}$$

for ρ and $\mathbf{e}^k$ defined in Theorem 1. Applying this bound recursively (and setting $x^0=0$) completes the proof:

$$\|x^k-x^*\| \leqslant \rho^k\|x^*\|+\sum_{i=1}^{k}\rho^{k-i}\mathbf{e}^i+\frac{2\sqrt{\beta}}{\alpha_{x^*}(1-\rho)}w.$$

Note that for convergence we require $\rho<1$ and therefore, a lower bound on the step size which is $\mu>(2\alpha_{x^*})^{-1}$.

B. Proof of Corollary 2

Following the error bound (10) derived in Theorem 1 and by setting $\mathbf{e}^k \leqslant Cr^k$ we obtain:

$$\|x^k-x^*\| \leqslant \rho^k\left(\|x^*\|+C\sum_{i=1}^{k}(r/\rho)^i\right)+\frac{2\sqrt{\beta}}{\alpha_{x^*}(1-\rho)}w,$$

which for $r<\rho$ implies

$$\|x^k-x^*\| \leqslant \rho^k\left(\|x^*\|+\frac{C}{1-r/\rho}\right)+\frac{2\sqrt{\beta}}{\alpha_{x^*}(1-\rho)}w,$$

and for $r>\rho$ implies

$$\|x^k-x^*\| \leqslant \rho^k\|x^*\|+Cr^k\sum_{i=1}^{k}(\rho/r)^{k-i}+\frac{2\sqrt{\beta}}{\alpha_{x^*}(1-\rho)}w$$
$$\leqslant r^k\left(\|x^*\|+\frac{C}{1-\rho/r}\right)+\frac{2\sqrt{\beta}}{\alpha_{x^*}(1-\rho)}w,$$

and for $r=\rho$ we immediately get

$$\|x^k-x^*\| \leqslant \rho^k\|x^*\|+Ck\rho^k+\frac{2\sqrt{\beta}}{\alpha_{x^*}(1-\rho)}w.$$

Note that there exists a constant c such that for an arbitrary small $\xi>0$ it holds $k\rho^k \leqslant c(\rho+\xi)^k$. Therefore we also achieve a linear convergence for the case $r=\rho$.

C. Proof of Theorem 2

As before set $g = 2A^T(Ax^{k-1} - y)$ and $\widetilde{g} = g + 2e_g^k$ for some bounded gradient error vector e_g^k i.e. $\|e_g^k\| \leqslant \nu_g^k$. Note that here the update rule of Algorithm (8) uses the $(1+\varepsilon)$-approximate projection which by definition (7) implies

$$\begin{aligned}\|x^k - x^{k-1} + \frac{\mu}{2}\widetilde{g}\|^2 &= \|\mathcal{P}_{\mathcal{C}}^{\varepsilon}(x^{k-1} - \frac{\mu}{2}\widetilde{g}) - x^{k-1} + \frac{\mu}{2}\widetilde{g}\|^2 \\ &\leqslant (1+\varepsilon)^2 \inf_{u\in\mathcal{C}} \|u - x^{k-1} + \frac{\mu}{2}\widetilde{g}\|^2 \\ &\leqslant \|x^* - x^{k-1} + \frac{\mu}{2}\widetilde{g}\|^2 + \varphi(\varepsilon)^2\frac{\mu^2}{4}\|\widetilde{g}\|^2\end{aligned}$$

where $\varphi(\varepsilon) := \sqrt{2\varepsilon + \varepsilon^2}$. For the last inequality we replace u with two feasible points $x^*, x^{k-1} \in \mathcal{C}$.

As a result by only replacing ν_g^k with $\mu\varphi(\varepsilon)\|\widetilde{g}\|/2$ in expression (14), we can follow identical steps as for the proof of Theorem 1 up to (17), revise the definition of $\mathbf{e}^k := 2\nu_g^k/\alpha_{x^*} + \sqrt{\mu}\varphi(\varepsilon)\|\widetilde{g}\|/(2\sqrt{\alpha_{x^*}})$ and write

$$\|x^k - x^*\| \leqslant \sqrt{\frac{1}{\mu\alpha_{x^*}} - 1}\|x^{k-1} - x^*\| + \frac{2\nu_g^k}{\alpha_{x^*}} + \frac{\varphi(\varepsilon)}{2}\sqrt{\frac{\mu}{\alpha_{x^*}}}\|\widetilde{g}\| + 2\frac{\sqrt{\beta}}{\alpha_{x^*}}w.$$

Note that so far we only assumed $\mu \leqslant \beta^{-1}$.

On the other hand by triangle inequality we have

$$\begin{aligned}\|\widetilde{g}\| &\leqslant \|g\| + 2\nu_g^k \\ &\leqslant 2\|A^TA(x^{k-1} - x^*)\| + 2\|A^T(y - Ax^*)\| + 2\nu_g^k \\ &\leqslant 2\sqrt{\beta}|||A|||\|(x^{k-1} - x^*)\| + 2|||A|||w + 2\nu_g^k \\ &\leqslant 2\sqrt{1/\mu}|||A|||\|(x^{k-1} - x^*)\| + 2|||A|||w + 2\nu_g^k.\end{aligned}$$

The third inequality uses the ULE property and the last one holds since $\mu \leqslant \beta^{-1}$. Therefore, we get

$$\begin{aligned}\|x^k - x^*\| \leqslant &\left(\sqrt{\frac{1}{\mu\alpha_{x^*}} - 1} + \varphi(\varepsilon)\frac{|||A|||}{\sqrt{\alpha_{x^*}}}\right)\|x^{k-1} - x^*\| \\ &+ \left(\frac{2}{\alpha_{x^*}} + \varphi(\varepsilon)\sqrt{\frac{\mu}{\alpha_{x^*}}}\right)\nu_g^k + \left(2\frac{\sqrt{\beta}}{\alpha_{x^*}} + \varphi(\varepsilon)\sqrt{\frac{\mu}{\alpha_{x^*}}}|||A|||\right)w.\end{aligned}$$

Based on assumption $\varphi(\varepsilon)\frac{|||A|||}{\sqrt{\alpha_{x^*}}} \leqslant \delta$ of the theorem we can deduce

$$\|x^k - x^*\| \leqslant \rho\|x^{k-1} - x^*\| + \left(\frac{2}{\alpha_{x^*}} + \frac{\sqrt{\mu}}{|||A|||}\delta\right)\nu_g^k + \left(2\frac{\sqrt{\beta}}{\alpha_{x^*}} + \sqrt{\mu}\delta\right)w$$

where $\rho = \sqrt{\frac{1}{\mu\alpha_{x^*}} - 1} + \delta$.

Applying this bound recursively (and setting $x^0 = 0$) completes the proof:

$$\|x^k - x^*\| \leqslant \rho^k\|x^*\| + \kappa_g\sum_{i=1}^{k}\rho^{k-i}\nu_g^i + \frac{\kappa_w}{1-\rho}w$$

for κ_g, κ_w defined in Theorem 2. The condition for convergence is $\rho < 1$ which implies $\delta < 1$ and a lower bound on the step size which is $\mu > (\alpha_{x^*} + (1-\delta)^2\alpha_{x^*})^{-1}$.

VI. Application in Data driven compressed sensing

Many CS reconstruction programs resort to signal models promoted by certain (semi) algebraic functions $h(x) : \mathbb{R}^n \to \mathbb{R}_+ \cup \{0\}$. For example we can have

$$\mathcal{C} := \big\{x \in \mathbb{R}^n : h(x) \leqslant \zeta\big\},$$

where $h(x)$ may be chosen as the ℓ_0 or ℓ_{0-2} semi-norms or as the $\mathrm{rank}(x)$ which promotes sparse, group-sparse or low-rank (for matrix spaces) solutions, respectively. One might also replace those penalties with their corresponding convex relaxations namely, the ℓ_1, ℓ_{1-2} norms or the nuclear norm.

Data driven compressed sensing however corresponds to cases where in the absence of an algebraic physical model one resorts to collect a large number of data samples in a dictionary and use it as a *point cloud* model for CS reconstruction [7]. Data driven CS finds numerous applications e.g. in Hyperspectral imagery [52], Mass spectroscopy (MALDI imaging) [53], Raman Imaging [54] and Magnetic Resonance Fingerprinting (MRF) [37, 38] just to name a few. For instance the USGS Hyperspectral library [2] contains the spectral signatures (reflectance) of thousands of substances measured across a few hundred frequency bands. This side information is shown to be useful for CS reconstruction and classification in both convex and nonconvex settings (see e.g. [52] for more details and relations to sparse approximation in redundant dictionaries). Data driven CS may also apply to algebraic models with non trivial projections. For example in the MRF reconstruction problem one first constructs a huge dictionary of fingerprints i.e. the magnetization responses (across the readout times) for many $T1, T2$ relaxation values (i.e. spin-spin and spin-echo) presented in normal tissues [37]. This corresponds to sampling a two-dimensional manifold associated with the solutions of the dynamic *Bloch equations* [38], which in the MRF settings neither the response nor the projection has an analytic closed-form solution.

A. A data driven CS in product space

To explore how our theoretical results can be used to accelerate CS reconstruction we consider a stylized data driven application for which we explain how one can obtain each of the aforementioned approximate projections. Consider a multi-dimensional image such as HSI, MALDI or MRF that can be represented by a $\widetilde{n} \times J$ matrix X, where $n = \widetilde{n}J$ is the total number of spatio-spectral pixels, J is the spatial resolution and $\widetilde{n}$ is the number of spectral bands e.g. $\widetilde{n} = 3$ for an RGB image, $\widetilde{n} \approx 400$ for an HSI acquired by NASA's AVIRIS spectrometer, $\widetilde{n} \approx 5000$ for a MALDI image [53]. In the simplest form we assume that each spatial pixel corresponds to a certain material with a specific signature, i.e.

$$X_j \in \widetilde{\mathcal{C}}, \quad \forall j = 1, \ldots, J,$$

where X_j denotes the jth column of X and

$$\widetilde{\mathcal{C}} := \bigcup_{i=1}^{d} \{\psi_i\} \in \mathbb{R}^{\widetilde{n}}$$

is the point cloud of a large number d of signatures ψ_i e.g. in a customized spectral library for HSI or MALDI data.

The CS sampling model follows (1) by setting $x^* := X_{\text{vec}}$, where by $X_{\text{vec}} \in \mathbb{R}^n$ we denote the vector-rearranged form of the matrix X. The CS reconstruction reads

$$\min_{\substack{X_j \in \widetilde{\mathcal{C}}, \\ \forall j = 1, \ldots, J}} \big\{f(X) := \frac{1}{2}\|y - AX_{\text{vec}}\|^2\big\}. \tag{18}$$

[2] http://speclab.cr.usgs.gov

or equivalently and similar to problem (2)

$$\min_{x\in\prod_{j=1}^{J}\widetilde{\mathcal{C}}}\big\{f(x):=\frac{1}{2}\|y-Ax\|^2\big\}.$$

The only update w.r.t. problem (2) is the fact that now the solution(s) x lives in a product space of the same model i.e.

$$\mathcal{C}:=\prod_{j=1}^{J}\widetilde{\mathcal{C}} \tag{19}$$

(see also [55] on product/Kronecker space CS however using a sparsity inducing semi-algebraic model). We note that solving directly this problem for a general A (e.g. sampling models which non trivially combine columns/spatial pixels of X) is *exponentially hard* $O(d^J)$ because of the combinatorial nature of the product space constraints. In this regard, a tractable scheme which has been frequently considered for this problem e.g. in [38] would be the application of an IPG type algorithm in order to break down the cost into the gradient and projection computations (here the projection requires $O(Jd)$ computations to search the closest signatures to the current solution) to locally solve the problem at each iteration.

B. Measurement bound

The classic Johnson-Lindenstrauss lemma says that one can use random linear transforms to stably embed point clouds into a lower dimension of size $O(\log(\#\widetilde{\mathcal{C}}))$ where $\#$ stands for the set cardinality [56]:

Theorem 3. *Let $\widetilde{\mathcal{C}}$ be a finite set of points in $\mathbb{R}^{\widetilde{n}}$. For A drawn at random from the i.i.d. normal distribution and a positive constant $\widetilde{\theta}$, with very high probability one has*

$$(1-\widetilde{\theta})\|(x-x')\|\leqslant\|A(x-x')\|\leqslant(1+\widetilde{\theta})\|(x-x')\|,\quad\forall x,x'\in\widetilde{\mathcal{C}}$$

provided $m=O(\log(\#\widetilde{\mathcal{C}})/\widetilde{\theta}^2)$.

Note that this definition implies an RIP embedding for the point cloud model according to (9) with a constant $\theta<3\widetilde{\theta}$ which in turn (and for small enough $\widetilde{\theta}$) implies the sufficient embedding condition for a stable CS recovery using the exact or approximate IPG. This bound considers an arbitrary point cloud and could be improved when data points $\widetilde{\mathcal{C}}\subseteq\mathcal{M}$ are derived from a low-dimensional structure, e.g. a smooth manifold $\mathcal{M}$ (such as the MR Fingerprints) for which one can alternatively use the corresponding RIP measurement complexity i.e. $m=O(\dim(\mathcal{M})/\widetilde{\theta}^2)$ where $\dim(\mathcal{M})\ll\widetilde{n}$ is the intrinsic topological dimension of the manifold [57].

We note that such a measurement bound for a product space model (19) without considering any structure between spaces turns into

$$m=O\left(J\min\left\{\log(d),\dim(\mathcal{M})\right\}\right).$$

C. Cover tree for fast nearest neighbour search

With the data driven CS formalism and discretization of the model the projection step of IPG reduces to searching for the nearest signature in each of the product spaces, however in a dictionary of potentially very large cardinality d. And thus search strategies with linear complexity in d e.g. an exhaustive search, can be a serious bottleneck for solving such problems. A very well-established approach to overcome the complexity of an exhaustive nearest neighbour (NN) search on a large dataset consists of hierarchically partitioning the solution space and forming a *tree* whose nodes represents those partitions, and then using branch-and-bound methods on the resulting tree for a fast Approximate NN (ANN) search with $o(d)$ complexity e.g. see [21, 58].

In this regard, we address the computational shortcoming of the projection step in the exact IPG by preprocessing $\widetilde{\mathcal{C}}$ and forming a *cover tree* structure suitable for fast ANN searches [21]. A cover tree is

a levelled tree whose nodes at different scales form covering nets for data points at multiple resolutions; if $\sigma := \max_{\psi\in\widetilde{\mathcal{C}}} \|\psi_{\text{root}} - \psi\|$ corresponds to the maximal coverage by the root, then nodes appearing at any finer scale $l > 0$ form a $(\sigma 2^{-l})$-covering net for their descendants i.e. as we descend down the tree the covering resolution refines in a dyadic coarse-to-fine fashion.

We consider three possible search strategies using such a tree structure:

- **Exact NN:** which is based on the branch-and-bound algorithm proposed in [21, Section 3.2]. Note that we should distinguish between this strategy and performing a brute force search. Although they both perform an exact NN search, the complexity of the proposed algorithm in [21] is shown to be way less in practical datasets.
- $(1+\varepsilon)$**-ANN:** this search is also based on the branch-and-bound algorithm proposed in [21, Section 3.2] which includes an early search termination criteria (depending on the accuracy level ε) for which one obtains an approximate oracle of the type defined by (7). Note that the case $\varepsilon = 0$ refers to the exact tree NN search described above.
- **FP-ANN:** that is traversing down the tree up to a scale $l = \lceil \log(\frac{\nu_p}{\sigma}) \rceil$ for which the covering resolution falls below a threshold ν_p on the search accuracy. This search results in a fixed precision type approximate oracle as described in Section III-A and in a sense it is similar to performing the former search with $\varepsilon = 0$, however on a truncated (low-resolution) cover tree.

All strategies could be applied to accelerate the projection step of an exact or inexact IPG (with variations discussed in Sections III-C and III-A) to tackle the data driven CS problem (18). In addition one can iteratively refine the accuracy of the FP-ANN search (e.g. $\nu_p^k = r^k$ for a certain decay rate $r < 1$ and IPG iteration number k) and obtain a PFP type approximate IPG discussed in Section III-B.

Note that while the cover tree construction is blind to the explicit structure of the data, several key growth properties such as the tree's explicit depth, the number of children per node, and importantly the overall search complexity are characterized by the intrinsic dimension of the model, called the *doubling dimension*, and defined as follows [59, 60]:

Definition 1. *Let $B(q, r)$ denotes a ball of radius r centred at a point q in some metric space. The doubling dimension $\dim_D(\mathcal{M})$ of a set $\mathcal{M}$ is the smallest integer such that every ball of $\mathcal{M}$ (i.e. $\forall r > 0$, $\forall q \in \mathcal{M}$, $B(q, 2r) \cap \mathcal{M}$) can be covered by $2^{\dim_D(\mathcal{M})}$ balls of half radius i.e. $B(q', r) \cap \mathcal{M}$, $q' \in \mathcal{M}$.*

The doubling dimension has several appealing properties e.g. $\dim_D(\mathbb{R}^n) = \Theta(n)$, $\dim_D(\mathcal{M}_1) \leqslant \dim_D(\mathcal{M}_2)$ when $\mathcal{M}_1$ is a subspace of $\mathcal{M}_2$, and $\dim(\cup_{i=1}^I \mathcal{M}_i) \leqslant \max_i \dim_D(\mathcal{M}_i) + \log(I)$ [58, 60]. Practical datasets are often assumed to have small doubling dimensions e.g. when $\widetilde{\mathcal{C}} \subseteq \mathcal{M}$ samples a low-dimensional manifold $\mathcal{M}$ with certain smoothness and regularity one has $\dim_D(\widetilde{\mathcal{C}}) \leqslant \dim_D(\mathcal{M}) = O(\dim(\mathcal{M}))$ [61] (where $\dim(\mathcal{M})$ denotes the topological manifold dimension).[3]

Equipped with such a notion of dimensionality, the following theorem bounds the complexity of a $(1+\varepsilon)$-ANN cover tree search [21, 58]:

Theorem 4. *Given a query which might not belong to $\widetilde{\mathcal{C}}$, the approximate $(1+\varepsilon)$-ANN search on a cover tree takes at most*

$$2^{O(\dim_D(\widetilde{\mathcal{C}}))} \log \Delta + (1/\varepsilon)^{O(\dim_D(\widetilde{\mathcal{C}}))} \tag{20}$$

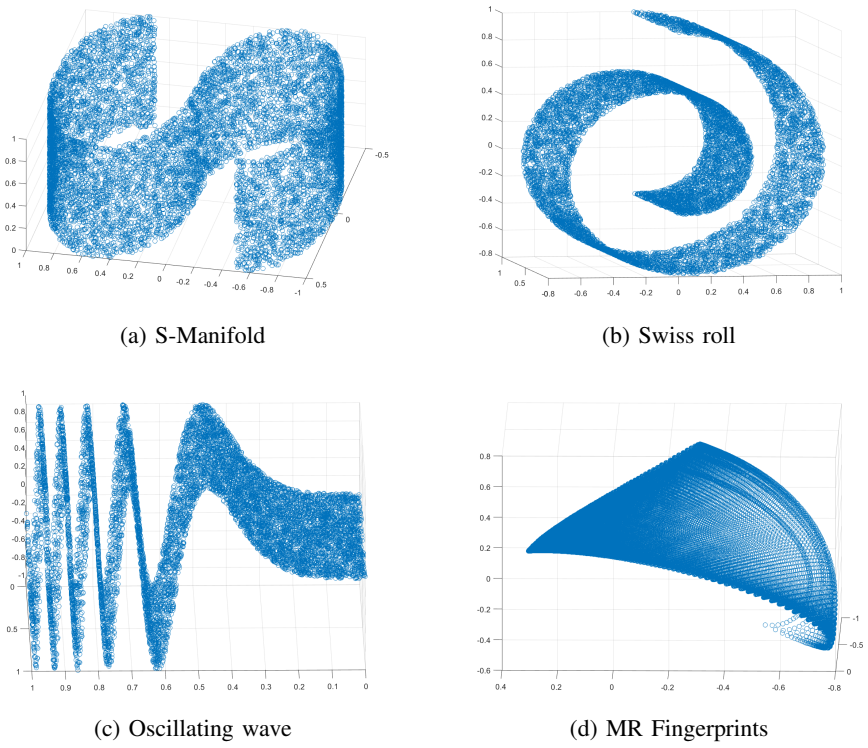
computations in time with $O(\#\widetilde{\mathcal{C}})$ memory requirement, where Δ is the aspect ratio of $\widetilde{\mathcal{C}}$.

For most applications $\log(\Delta) = O(\log(d))$ [58] and thus for datasets with low dimensional structures i.e. $\dim_D = O(1)$ and by using approximations one achieves a logarithmic search complexity in d, as opposed to the linear complexity of a brute force search.

[3] Although the doubling dimension $\dim_D$ scales similar to the RIP measurement complexity for certain sets e.g. linear subspaces, UoS, smooth manifolds..., this does not generally hold in $\mathbb{R}^n$ and one needs to distinguish between these notions, see for more discussions [62, 63].

Dataset	Population (d)	Ambient dim. ($\widetilde{n}$)	CT depth	CT res.
S-Manifold	5000	200	14	2.43E-4
Swiss roll	5000	200	14	1.70E-4
Oscillating wave	5000	200	14	1.86E-4
MR Fingerprints	29760	512	13	3.44E-4

TABLE I: Datasets for data-driven CS evaluations; a cover tree (CT) structure is formed for each dataset. The last two columns respectively report the number of scales and the finest covering resolution of each tree.



(a) S-Manifold

(b) Swiss roll

(c) Oscillating wave

(d) MR Fingerprints

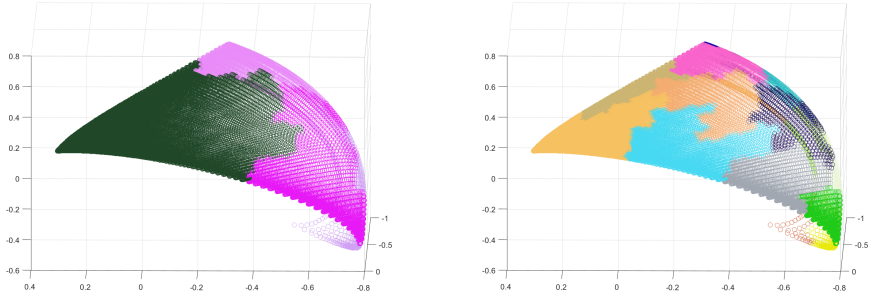
Fig. 1: Illustration of the low dimensional structures of datasets presented in Table I. Points are depicted along the first three principal components of each dataset.

Note that the complexity of an *exact* cover tree search could be arbitrarily high and thus the same applies to the FP and PFP type ANN searches since they are also based on performing exact NN (on a truncated tree). However in the next section we empirically observe that the complexity of an exact cover tree NN (and also the FP and PFP type ANN) is much lower than performing an exhaustive search.

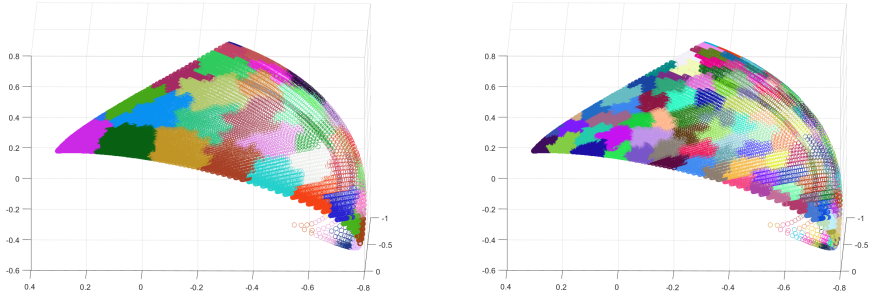
VII. Numerical experiments

We test the performance of the exact/inexact IPG algorithm for our product-space data driven CS reconstruction using the four datasets described in Table I. The datasets are uniformly sampled (populated) from 2-dimensional continuous manifolds embedded in a higher ambient dimension, see also Figure 1[4].

[4]The S-manifold, Swiss roll and Oscillating wave are synthetic machine learning datasets available e.g. in [64]. The Magnetic Resonance Fingerprints (MRF) is generated by solving the Bloch dynamic equation for a uniform grid of relaxation times $T1, T2$ and for an external magnetic excitation pattern, discussed and implemented in [37].



(a) Scale 2 (b) Scale 3



(c) Scale 4 (d) Scale 5

Fig. 2: A cover tree is built on MR Fingerprints dataset: (a-d) data partitions covered by distinct parent/ancestor nodes appearing at scales 2-5 are highlighted in different colours. The coverage resolution refines by increasing the scale.

To proceed with fast ANN searches within IPG, we separately build a cover tree structure per dataset i.e. a preprocessing step. As illustrated for the MRF manifold in Figure 2 the coverage levels decrease in a coarse-to-fine manner as we traverse down the tree i.e. increasing the scale.

Along with a brute-force exact search, three cover tree based ANN search strategies are investigated as described in the previous section:

- FP-ANN for precision parameters $\nu_p = \{0.1, 0.05, .01, 0.001\}$.
- PFP-ANN for varying precision errors $\nu_p^k = r^k$ decaying at rates $r = \{0.05, 0.1, 0.15, \ldots, 0.95\}$.
- $(1+\varepsilon)$-ANN for near optimality parameters $\varepsilon = \{0, 0.2, 0.4, \ldots, 4\}$. The case $\varepsilon = 0$ corresponds to an exact NN search, however by using the branch-and-bound algorithm on the cover tree proposed in [21].

Gaussian CS sampling: From each dataset we select $J = 50$ points at random and populate our signal matrix $X \in \mathbb{R}^{\widetilde{n} \times J}$. We then subsample the signal using the linear noiseless model discussed in (18), where the sampling matrix $A \in \mathbb{R}^{m \times \widetilde{n}J}$ is drawn at random from the i.i.d. normal distribution. We denote by $\frac{m}{n} \leqslant 1$ (where, $n = \widetilde{n}J$) as the subsampling ratio used in each experiment.

Throughout we set the maximum number of IPG iterations to 30. The step size is set to $\mu = 1/m \approx 1/\beta$ which is a theoretical value satisfying the restricted Lipschitz smoothness condition for the i.i.d. Normal sampling ensembles in our theorems and related works on iterative hard thresholding algorithms e.g. see [27, 43, 44].

Figure 3 shows the normalized solution MSE measured by $\frac{\|x^k - x^*\|}{\|x^*\|}$ at each iteration of the exact and inexact IPG algorithms, and for a given random realization of the sampling matrix A and selected signals X. For the FP-ANN IPG the convergence rate is unchanged from the exact IPG algorithm but the reconstruction accuracy depends on the chosen precision parameter and for lower precisions the algorithm

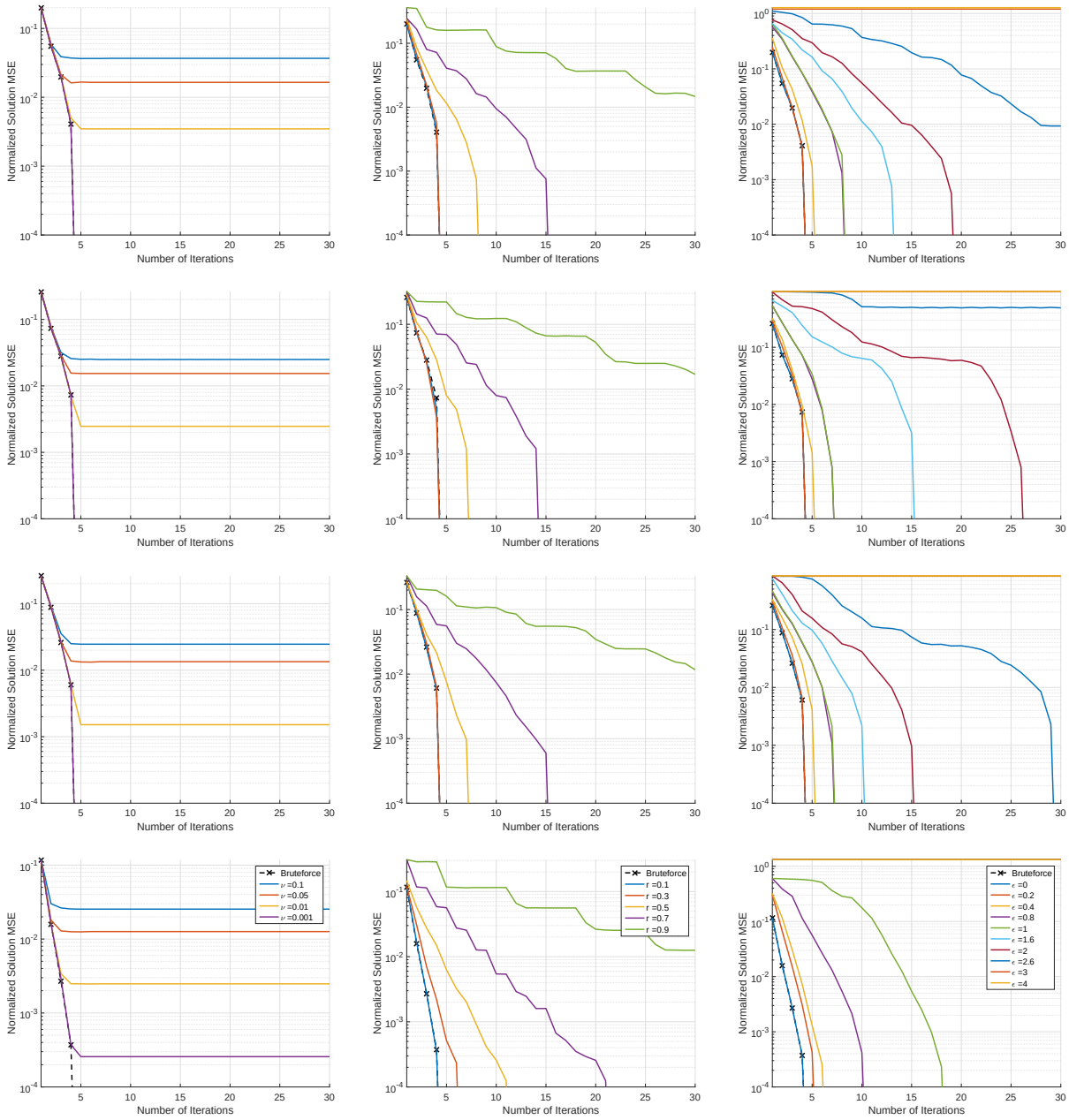


Fig. 3: Convergence of the exact/inexact IPG for subsampling ratio $\frac{m}{n} = 0.2$. Columns from left to right correspond to inexact algorithms with FP, PFP and $1+\varepsilon$ ANN searches, respectively (legends for the plots in each column are identical and included in the last row). Rows from top to bottom correspond to S-Manifold, Swiss roll, Oscillating wave and MR Fingerprints datasets, respectively.

stops at an earlier iteration with reduced accuracy, but with the benefit of requiring a smaller search tree.

The PFP-ANN IPG ultimately achieves the same accuracy of the exact algorithm. Refining the approximations at a slow rate slows down the convergence of the algorithm (i.e. the staircase effect visible in the plots correspond to $r = \{0.7, 0.9\}$), whereas choosing too fast error decays, e.g. $r = 0.1$, does not improve the convergence rate beyond the exact algorithm and thus potentially leads to computational inefficiency. The $(1+\varepsilon)$-ANN IPG algorithm can also achieve the precision of an exact recovery for moderately chosen approximation parameters. The case $\varepsilon = 0$ (unsurprisingly) iterates the same steps

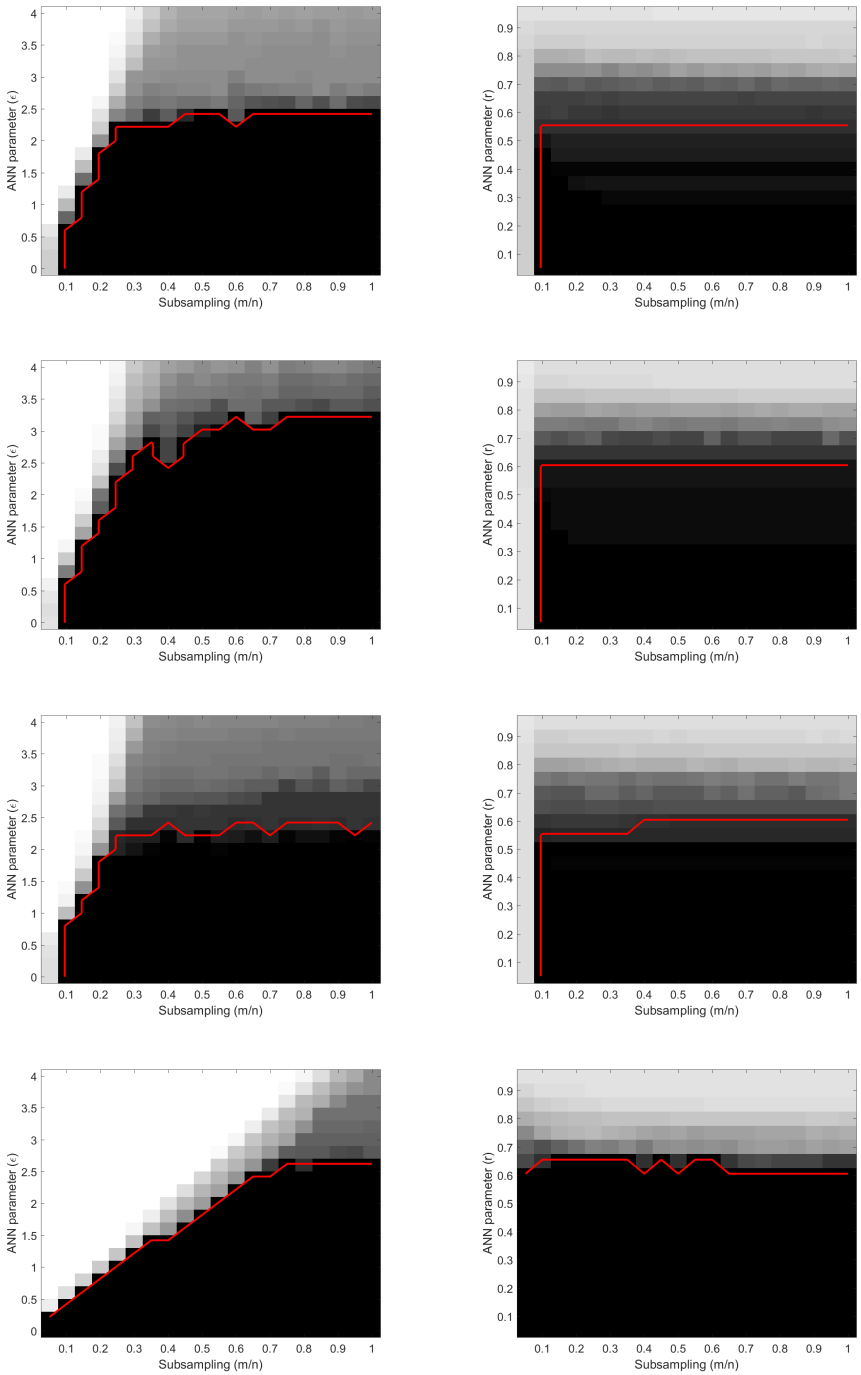


Fig. 4: Recovery phase transitions for IPG with approximate projection (i.e. ANN search). Image intensities correspond to the normalized solution MSE for a search parameter and a given subsampling ratio (ranging between 5-100$\%$). Intensities in all plots are identically set with a logarithmic scale: black pixels correspond to accurate points with MSE $\leqslant 10^{-6}$, white pixels represent points with MSE $\geqslant 1$, and the region below the red curve is defined as the exact recovery region with MSE $\leqslant 10^{-4}$. Rows from top to bottom correspond to the phase transitions of S-Manifold, Swiss roll, Oscillating wave and MR Fingerprints datasets. The left and the right columns correspond to two cover tree based ANN searches namely, the $(1+\varepsilon)$-ANN and the PFP-ANN with decay parameter r.

as for the IPG with brute-force search. Increasing ε slows down the convergence and for a very large parameter, e.g. $\varepsilon = \{3, 4\}$, the algorithm diverges.

Figure 4 illustrates the recovery phase transitions for the inexact IPG using the PFP-ANN and $(1+\varepsilon)$-ANN searches. The normalized MSE is averaged over 10 random realizations of the sampling matrix A and 20 randomly subselected signal matrices X for a given A. In each image the area below the red curve has the solution MSE less than 10^{-4} and is chosen as the recovery region. We can observe that the PFP-ANN oracle results in a recovery region which is almost invariant to the chosen decay parameter r (except for the slow converging case $r \gtrsim 0.6$, due to the limit on the maximum number of iterations).

In the case of the $1+\varepsilon$ oracle we see a different behaviour; smaller values of ε allow for a larger recovery region and larger approximations are restricted to work only in high sampling regimes. This observation is in agreement with our theoretical bounds on recovery and it shows that the $(1+\varepsilon)$-approximate oracles are sensitive to the compression ratio, even though an exact (or a better-chosen approximate) IPG might still report recovery in the same sampling regime.

Finally in Table II we report the total cost of projections for each iterative scheme. The cost is measured as the total number of pairwise distances calculated for performing the NN or ANN searches, and it is averaged over the same trials as previously described[5]. For a better evaluation we set the algorithm to terminate earlier (than 30 iterations) when the objective function does not progress more than a tolerance level $tol = 10^{-8}$. For each scheme the reported parameter achieves an average normalized solution MSE $\leqslant 10^{-4}$ in the smallest amount of computations. For comparison we also include the cost of exact IPG implemented with the brute-force and exact ($\varepsilon = 0$) cover tree NN searches. When using a brute-force NN search the cost per iteration is fixed and it is equal to the whole dataset population; as a result the corresponding exact IPG reports the highest computation. Replacing the brute-force search with a cover tree based exact NN search significantly reduces the computations. This is related to the low dimensionality of the manifolds in our experiments for which a cover tree search, even for performing an exact NN, turns out to require many fewer pairwise distances evaluations. Remarkably, the approximate algorithm $(1 + \varepsilon)$-ANN IPG consistently outperforms all other schemes by reporting 4-10 times acceleration compared to the exact algorithm with $\varepsilon = 0$, and about (or sometimes more than) 2 orders of magnitude acceleration compared to the IPG with an exact brute-force search; in fact for larger datasets the gap becomes wider as the $(1 + \varepsilon)$-ANN complexity stays relatively invariant to the population size. The FP-ANN IPG reports similar computations as for the exact tree search ($\varepsilon = 0$) algorithm because in order to achieve the desired accuracy the (exact) search is performed up to a very fine level of the tree. A gradual progress along the tree levels by the PFP-ANN IPG however improves the search time and reports a comparable computation cost to the $(1 + \varepsilon)$-ANN. Also it can be observed that by taking more samples the overall projection cost reduces which is related to the fast convergence (i.e. less iterations) of the algorithm once more measurements are available.

VIII. Conclusion and discussions

We studied the robustness of the iterative projected gradient algorithm against inexact gradient and projection oracles and for solving inverse problems in compressed sensing. We considered fixed precision, progressive fixed precision and $(1 + \varepsilon)$-approximate oracles. A notion of model information preserving under a hybrid local-uniform embedding assumption is at the heart of our analysis. We showed that under the same assumptions, the algorithm with PFP approximate oracles achieves the accuracy of the exact IPG. For a certain rate of decay of the approximation errors this scheme can also maintain the rate of linear

[5]In our evaluations, we exclude the computation costs of the gradient updates, i.e. the forward and backward operators, which can become dominant when datasets are not very large and the sampling matrix is dense e.g. a Gaussian matrix. For structured embedding matrices such as the fast Johnson-Lindenstrauss transform [65] or randomized orthoprojectors e.g. in MRI applications the cost of gradient updates becomes a tiny fraction of the search step, particularly when dealing with a large size dataset.

	Total NN/ANN cost ($\times 10^4$)											
Subsampling ratio ($\frac{m}{n}$)	10%				20%				30%			
Datasets	SM	SR	OW	MRF	SM	SR	OW	MRF	SM	SR	OW	MRF
Brute-force NN	194.23	193.67	215.10	923.34	130.80	127.19	140.89	744.23	113.55	109.34	123.06	699.48
CT's exact NN ($\varepsilon = 0$)	8.11	8.90	15.47	33.05	4.90	5.19	8.99	24.74	3.87	4.08	7.19	20.91
FP-ANN Parameter ν_p	8.11 1E-3	8.90 1E-3	15.47 1E-3	-	4.90 1E-3	5.19 1E-3	9.00 1E-3	-	3.88 1E-3	4.07 1E-3	7.21 1E-3	-
PFP-ANN Parameter r	2.94 4E-1	3.50 5E-1	7.10 5E-1	3.41 4E-1	1.96 3E-1	2.41 3E-1	3.94 4E-1	2.84 4E-1	1.78 4E-1	1.99 3E-1	3.38 4E-1	2.52 2E-1
$(1+\varepsilon)$-ANN Parameter ε	**2.36** 4E-1	**2.77** 4E-1	**4.54** 4E-1	**2.78** 4E-1	**1.54** 4E-1	**1.86** 4E-1	**2.91** 4E-1	**2.21** 4E-1	**1.31** 4E-1	**1.60** 4E-1	**2.46** 6E-1	**1.92** 4E-1

TABLE II: Average computational complexity of the exact/inexact IPG measured by the total number of pairwise distances (in the ambient dimension) calculated within the NN/ANN steps to achieve an average solution MSE $\leqslant 10^{-4}$ (algorithms with less accuracies are marked as '-'). For each ANN scheme the lowest cost and the associated parameter is reported. SM, SR, OW and MRF abbreviate S-Manifold, Swiss roll, Oscillating wave and the MR Fingerprints datasets, respectively.

convergence as for the exact algorithm. We also conclude that choosing too fast decays does not help the convergence rate beyond the exact algorithm and therefore can result in computational inefficiency. The $(1+\varepsilon)$-approximate IPG can also achieves the accuracy of the exact algorithm, however under a stronger embedding condition, slower rate of convergence and possibly more noise amplification compared to the exact algorithm. We show that this approximation is sensitive to the CS subsampling regime i.e. for high compression ratios one can not afford too large approximation. We applied our results to a class of data driven compressed sensing problems, where we replaced exhaustive NN searches over large datasets with fast and approximate alternatives introduced by the cover tree structure. Our experiments indicate that the inexact IPG with $(1+\varepsilon)$-ANN searches (and also comparably the PFP type search) can significantly accelerate the CS reconstruction.

Our results require a lower bound on the chosen step size which is a critical assumption for globally solving a nonconvex CS problem, see e.g. [19, 43, 44]. With no lower bound on the step size only convergence to a local critical point is guaranteed (for the exact algorithm) see e.g. [66, 67]. Recent studies [48, 68, 69] established *fast linear* convergence for solving non strongly convex problems such as CS with the exact IPG, and by assuming a notion of *local* (model restricted) strong convexity as well as choosing large enough step sizes. In our future work we would like to make more explicit connection between these results and our embedding assumptions and extend our approximation robustness study to the convex CS recovery settings with sharper bounds than e.g. [20].

A limitation of our current method is the dependency on ε for the CS recovery using $(1+\varepsilon)$-ANN searches. In future we plan to investigate whether the ideas from [33] can be generalized to include our data-driven signal models.

Finally we did not provide much discussion or experiments on the applications of the approximate gradient updates within the IPG. We think such approximations might be related to the sketching techniques for solving large size inverse problems, see [70, 71]. In our future work we would like to make explicit connections in this regard as well.

REFERENCES

[1] D. L. Donoho, "Compressed sensing," *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.

[2] E. J. Candes, J. Romberg, and T. Tao, "Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information," *IEEE Transactions on Information Theory*, vol. 52, no. 2, pp. 489–509, Feb 2006.

[3] J. A. Tropp, A. C. Gilbert, and M. J. Strauss, "Algorithms for simultaneous sparse approximation. part i: Greedy pursuit," *Signal Processing*, vol. 86, no. 3, pp. 572 – 588, 2006.

[4] E. Candès and B. Recht, "Exact matrix completion via convex optimization," *Commun. ACM*, vol. 55, no. 6, pp. 111–119, Jun. 2012.

[5] R. G. Baraniuk and M. B. Wakin, "Random projections of smooth manifolds," *Foundations of Computational Mathematics*, vol. 9, no. 1, pp. 51–77, 2009.

[6] R. G. Baraniuk, V. Cevher, M. F. Duarte, and C. Hegde, "Model-based compressive sensing," *IEEE Transactions on Information Theory*, vol. 56, no. 4, pp. 1982–2001, April 2010.

[7] R. G. Baraniuk, V. Cevher, and M. B. Wakin, "Low-dimensional models for dimensionality reduction and signal recovery: A geometric perspective," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 959–971, June 2010.

[8] V. Cevher, S. Becker, and M. Schmidt, "Convex optimization for big data: Scalable, randomized, and parallel algorithms for big data analytics," *IEEE Signal Processing Magazine*, vol. 31, no. 5, pp. 32–43, Sept 2014.

[9] L. Bottou, *Large-Scale Machine Learning with Stochastic Gradient Descent*. Heidelberg: Physica-Verlag HD, 2010, pp. 177–186.

[10] A. Chambolle and T. Pock, "A first-order primal-dual algorithm for convex problems with applications to imaging," *Journal of Mathematical Imaging and Vision*, vol. 40, no. 1, pp. 120–145, 2011.

[11] J. M. Fadili and G. Peyré, "Total variation projection with first order schemes," *IEEE Transactions on Image Processing*, vol. 20, no. 3, pp. 657–669, 2011.

[12] S. Ma, D. Goldfarb, and L. Chen, "Fixed point and bregman iterative methods for matrix rank minimization," *Mathematical Programming*, vol. 128, no. 1, pp. 321–353, 2011.

[13] A. d'Aspremont, L. E. Ghaoui, M. I. Jordan, and G. R. G. Lanckriet, "A direct formulation for sparse pca using semidefinite programming," *SIAM Review*, vol. 49, no. 3, pp. 434–448, 2007.

[14] M. Golbabaee and P. Vandergheynst, "Hyperspectral image compressed sensing via low-rank and joint-sparse matrix recovery," in *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2012, pp. 2741–2744.

[15] R. L. Dykstra, "An algorithm for restricted least squares regression," *Journal of the American Statistical Association*, vol. 78, no. 384, pp. 837–842, 1983.

[16] J. P. Boyle and R. L. Dykstra, *A Method for Finding Projections onto the Intersection of Convex Sets in Hilbert Spaces*. New York, NY: Springer New York, 1986, pp. 28–47.

[17] R. Giryes and M. Elad, "Iterative hard thresholding with near optimal projection for signal recovery," in *10th international conference on Sampling Theory and Applications (SampTA 2013)*, 2013, pp. 212–215.

[18] H. Rauhut, R. Schneider, and eljka Stojanac, "Low rank tensor recovery via iterative hard thresholding," *Linear Algebra and its Applications*, vol. 523, pp. 220 – 262, 2017.

[19] T. Blumensath, "Sampling and reconstructing signals from a union of linear subspaces," *IEEE Transactions on Information Theory*, vol. 57, no. 7, pp. 4660–4671, July 2011.

[20] M. Schmidt, N. L. Roux, and F. R. Bach, "Convergence rates of inexact proximal-gradient methods for convex optimization," in *Advances in Neural Information Processing Systems 24*, 2011, pp. 1458–1466.

[21] A. Beygelzimer, S. Kakade, and J. Langford, "Cover trees for nearest neighbor," in *Proceedings of the 23rd international conference on Machine learning*. ACM, 2006, pp. 97–104.

[22] O. Devolder, F. Glineur, and Y. Nesterov, "First-order methods of smooth convex optimization with inexact oracle," *Mathematical Programming*, vol. 146, no. 1, pp. 37–75, 2014.

[23] A. d'Aspremont, "Smooth optimization with approximate gradient," *SIAM Journal on Optimization*, vol. 19, no. 3, pp. 1171–1183, 2008.

[24] S. Villa, S. Salzo, L. Baldassarre, and A. Verri, "Accelerated and inexact forward-backward algorithms," *SIAM Journal on Optimization*, vol. 23, no. 3, pp. 1607–1633, 2013.

[25] J.-F. Aujol and C. Dossal, "Stability of over-relaxations for the forward-backward algorithm, application to fista," *SIAM Journal on Optimization*, vol. 25, no. 4, pp. 2408–2433, 2015.
[26] L. Yuan, J. Liu, and J. Ye, "Efficient methods for overlapping group lasso," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 9, pp. 2104–2116, Sept 2013.
[27] P. Shah and V. Chandrasekaran, "Iterative projections for signal identification on manifolds: Global recovery guarantees," in *2011 49th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, Sept 2011, pp. 760–767.
[28] N. Nguyen, D. Needell, and T. Woolf, "Linear convergence of stochastic iterative greedy algorithms with sparse constraints," *arXiv preprint arXiv:1407.0088*, 2014.
[29] A. Kyrillidis and V. Cevher, "Matrix recipes for hard thresholding methods," *Journal of Mathematical Imaging and Vision*, vol. 48, no. 2, pp. 235–265, 2014.
[30] C. Hegde, P. Indyk, and L. Schmidt, "A fast approximation algorithm for tree-sparse recovery," in *2014 IEEE International Symposium on Information Theory*, 2014, pp. 1842–1846.
[31] M. A. Davenport, D. Needell, and M. B. Wakin, "Signal space cosamp for sparse recovery with redundant dictionaries," *IEEE Transactions on Information Theory*, vol. 59, no. 10, pp. 6820–6829, Oct 2013.
[32] R. Giryes and D. Needell, "Greedy signal space methods for incoherence and beyond," *Applied and Computational Harmonic Analysis*, vol. 39, no. 1, pp. 1 – 20, 2015.
[33] C. Hegde, P. Indyk, and L. Schmidt, "Approximation algorithms for model-based compressive sensing," *IEEE Transactions on Information Theory*, vol. 61, no. 9, pp. 5129–5147, 2015.
[34] G. Tang, B. N. Bhaskar, and B. Recht, "Sparse recovery over continuous dictionaries-just discretize," in *2013 Asilomar Conference on Signals, Systems and Computers*, 2013, pp. 1043–1047.
[35] A. Asaei, M. Golbabaee, H. Bourlard, and V. Cevher, "Structured sparsity models for reverberant speech separation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, no. 3, pp. 620–633, March 2014.
[36] M. Golbabaee, A. Alahi, and P. Vandergheynst, "Scoop: A real-time sparsity driven people localization algorithm," *Journal of Mathematical Imaging and Vision*, vol. 48, no. 1, pp. 160–175, Jan. 2014.
[37] D. Ma, V. Gulani, N. Seiberlich, K. Liu, J. Sunshine, J. Durek, and M. Griswold, "Magnetic resonance fingerprinting," *Nature*, vol. 495, no. 7440, pp. 187–192, 2013.
[38] M. Davies, G. Puy, P. Vandergheynst, and Y. Wiaux, "A compressed sensing framework for magnetic resonance fingerprinting," *SIAM Journal on Imaging Sciences*, vol. 7, no. 4, pp. 2623–2656, 2014.
[39] C. Cartis and A. Thompson, "An exact tree projection algorithm for wavelets," *IEEE Signal Processing Letters*, vol. 20, no. 11, pp. 1026–1029, 2013.
[40] A. Deshpande and S. Vempala, *Adaptive Sampling and Fast Low-Rank Matrix Approximation*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 292–303.
[41] N. Halko, P. G. Martinsson, and J. A. Tropp, "Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions," *SIAM Review*, vol. 53, no. 2, pp. 217–288, 2011.
[42] J. Håstad, "Tensor rank is NP-complete," *Journal of Algorithms*, vol. 11, no. 4, pp. 644 – 654, 1990.
[43] T. Blumensath and M. E. Davies, "Iterative hard thresholding for compressed sensing," *Applied and Computational Harmonic Analysis*, vol. 27, no. 3, pp. 265 – 274, 2009.
[44] T. Blumensath, "Accelerated iterative hard thresholding," *Signal Processing*, vol. 92, no. 3, pp. 752 – 756, 2012.
[45] J. D. Blanchard, C. Cartis, and J. Tanner, "Compressed sensing: How sharp is the restricted isometry property?" *SIAM Review*, vol. 53, no. 1, pp. 105–125, 2011.
[46] S. Vaiter, M. Golbabaee, J. Fadili, and G. Peyré, "Model selection with low complexity priors," *Information and Inference: A Journal of the IMA*, vol. 4, no. 3, p. 230, 2015.

[47] V. Chandrasekaran, B. Recht, P. A. Parrilo, and A. S. Willsky, "The convex geometry of linear inverse problems," *Foundations of Computational Mathematics*, vol. 12, no. 6, pp. 805–849, 2012.
[48] S. Oymak, B. Recht, and M. Soltanolkotabi, "Sharp time-data tradeoffs for linear inverse problems," *arXiv preprint arXiv:1507.04793*, 2015.
[49] V. Chandrasekaran and M. I. Jordan, "Computational and statistical tradeoffs via convex relaxation," *Proceedings of the National Academy of Sciences*, vol. 110, no. 13, pp. E1181–E1190, 2013.
[50] E. Arias-Castro and Y. C. Eldar, "Noise folding in compressed sensing," *IEEE Signal Processing Letters*, vol. 18, no. 8, pp. 478–481, 2011.
[51] M. A. Davenport, J. N. Laska, J. R. Treichler, and R. G. Baraniuk, "The pros and cons of compressive sensing for wideband signal acquisition: Noise folding versus dynamic range," *IEEE Transactions on Signal Processing*, vol. 60, no. 9, pp. 4628–4642, 2012.
[52] M. Golbabaee, S. Arberet, and P. Vandergheynst, "Compressive source separation: Theory and methods for hyperspectral imaging," *IEEE Transactions on Image Processing*, vol. 22, no. 12, pp. 5096–5110, Dec 2013.
[53] J. H. Kobarg, P. Maass, J. Oetjen, O. Tropp, E. Hirsch, C. Sagiv, M. Golbabaee, and P. Vandergheynst, "Numerical experiments with maldi imaging data," *Advances in Computational Mathematics*, vol. 40, no. 3, pp. 667–682, 2014.
[54] B. M. Davis, A. J. Hemphill, D. Cebeci Malta, M. A. Zipper, P. Wang, and D. Ben-Amotz, "Multivariate hyperspectral raman imaging using compressive detection," *Analytical Chemistry*, vol. 83, no. 13, pp. 5086–5092, 2011.
[55] M. F. Duarte and R. G. Baraniuk, "Kronecker compressive sensing," *IEEE Transactions on Image Processing*, vol. 21, no. 2, pp. 494–504, 2012.
[56] W. B. Johnson and J. Lindenstrauss, "Extensions of lipschitz mappings into a hilbert space," *Contemporary mathematics*, vol. 26, no. 189-206, p. 1, 1984.
[57] K. L. Clarkson, "Tighter bounds for random projections of manifolds," in *Proceedings of the Twenty-fourth Annual Symposium on Computational Geometry*, ser. SCG '08. New York, NY, USA: ACM, 2008, pp. 39–48.
[58] R. Krauthgamer and J. R. Lee, "Navigating nets: Simple algorithms for proximity search," in *Proceedings of the Fifteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, ser. SODA '04, 2004.
[59] P. Assouad, "Étude d'une dimension métrique liée la possibilité de plongements dans r^n," *C. R. Acad. Sci. Paris Sér. A-B.*, vol. 15, no. 288, pp. A731–A734, 1979.
[60] J. Heinonen, *Lectures on analysis on metric spaces*. Springer Science & Business Media, 2012.
[61] S. Dasgupta and Y. Freund, "Random projection trees and low dimensional manifolds," in *Proceedings of the fortieth annual ACM symposium on Theory of computing*. ACM, 2008, pp. 537–546.
[62] P. Indyk and A. Naor, "Nearest-neighbor-preserving embeddings," *ACM Trans. Algorithms*, vol. 3, no. 3, Aug. 2007.
[63] S. Kpotufe and S. Dasgupta, "A tree-based regressor that adapts to intrinsic dimension," *Journal of Computer and System Sciences*, vol. 78, no. 5, pp. 1496 – 1515, 2012.
[64] W. K. Allard, G. Chen, and M. Maggioni, "Multi-scale geometric methods for data sets ii: Geometric multi-resolution analysis," *Applied and Computational Harmonic Analysis*, vol. 32, no. 3, pp. 435 – 462, 2012.
[65] N. Ailon and B. Chazelle, "Approximate nearest neighbors and the fast johnson-lindenstrauss transform," in *Proceedings of the thirty-eighth annual ACM symposium on Theory of computing*. ACM, 2006, pp. 557–563.
[66] Y. Nesterov, "Gradient methods for minimizing composite functions," *Mathematical Programming*, vol. 140, no. 1, pp. 125–161, 2013.

[67] H. Attouch, J. Bolte, and B. F. Svaiter, "Convergence of descent methods for semi-algebraic and tame problems: proximal algorithms, forward–backward splitting, and regularized gauss–seidel methods," *Mathematical Programming*, vol. 137, no. 1, pp. 91–129, 2013.

[68] A. Agarwal, S. Negahban, and M. J. Wainwright, "Fast global convergence rates of gradient methods for high-dimensional statistical recovery," in *Advances in Neural Information Processing Systems 23*. Curran Associates, Inc., 2010, pp. 37–45.

[69] J. Liang, J. Fadili, and G. Peyré, "Local linear convergence of forward–backward under partial smoothness," in *Advances in Neural Information Processing Systems 27*. Curran Associates, Inc., 2014, pp. 1970–1978.

[70] M. Pilanci and M. J. Wainwright, "Iterative hessian sketch: Fast and accurate solution approximation for constrained least-squares," *J. Mach. Learn. Res.*, vol. 17, no. 1, pp. 1842–1879, 2016.

[71] J. Tang, M. Golbabaee, and M. Davies, "Gradient projection iterative sketch for large scale constrained least-squares," *Proceedings of the 34th International Conference on Machine Learning (ICML)*, 2017.