

A Game Theoretic Approach to Expander-based Compressive Sensing

Sina Jafarpour
Department of Computer Science
Princeton University
sina@cs.princeton.edu

Volkan Cevher
Idiap Research Institute
Ecole Polytechnique Federale de Lausanne
volkan.cevher@epfl.ch

Robert E. Schapire
Department of Computer Science
Princeton University
schapire@cs.princeton.edu

Abstract—We consider the following expander-based compressive sensing (e-CS) problem: Given $\Phi \in \mathbb{R}^{M \times N}$ ($M < N$), which is the adjacency matrix of an expander graph, and a vector $y \in \mathbb{R}^M$, we seek to find a vector x^* with at most k -nonzero entries such that $x^* = \arg \min_{\|x\|_0 \leq k} \|y - \Phi x\|_1$, whenever it exists ($k \ll N$). Such problems are not only non-smooth, barring naive convexified sparse recovery approaches, but also are NP-Hard in general. To handle the non-smoothness, we provide a saddle-point reformulation of the e-CS problem, and propose a novel approximation scheme, called the *game-theoretic approximate matching estimator* (GAME) algorithm. We then show that the restricted isometry property of expander matrices in the ℓ_1 -norm circumvents the intractability of e-CS in the worst case. GAME therefore finds a sparse approximation \hat{x} to optimal solution such that $\|x^* - \hat{x}\|_1 = \mathcal{O}(\|y - \Phi x^*\|_1)$. We also propose a convex optimization approach to e-CS based on Nesterov smoothing, and discuss its (dis)advantages.

I. INTRODUCTION

Compressive sensing (CS) [1], [2] provides a rigorous foundation for underdetermined linear regression problems by integrating three central tenets: signal sparsity in a known basis, a measurement matrix that stably embeds sparse signals in the ℓ_2 -norm, and polynomial sparse recovery algorithms with recovery guarantees. The stable embedding feature in the ℓ_2 -norm, also known as the restricted isometry property, dictates the recovery guarantees, the computational as well as the space complexity of the existing recovery algorithms since only dense matrices satisfy it. As regression problems in high-dimensions are currently the *modus operandi* in image compression, data streaming, medical signal processing, and digital communications, there is a great interest for alternative approaches to reduce storage requirements and computational costs without sacrificing robustness guarantees (c.f., [3]).

Expander-based compressive sensing (e-CS) is an emerging alternative, in which the adjacency matrix of an expander graph is used as the measurement matrix. An expander graph is a regular bipartite graph for which every sufficiently small subset of variable nodes has a small number of colliding edges and a significant number of unique neighbors. As the resulting matrices are sparse, e-CS requires less storage, and provides salient computational advantages in recovery. These matrices also satisfy a restricted isometry property for sparse signals in the canonical sparsity basis, albeit in the weaker ℓ_1 -norm [3].

In the e-CS context, the sparse recovery approaches can be split into two distinct camps with one based on message

passing and the other based on convex optimization. *Message-passing algorithms* [4], [5] exploit the combinatorial structure of the expander graphs. While these algorithms are efficient and rather easy to implement, their approximation guarantees are meaningful only in extremely high-dimensions, as they feature large constants that are not suitable for practical applications. *Optimization-based methods* exploit the geometry of the expander graphs, and perform significantly better than the message passing approaches in practical applications [6]. Unfortunately, the computational cost of solving sparse recovery methods using the interior point method is typically cubic in the data dimension N , and computationally scalable gradient-only based alternatives have difficulties when the objective is not differentiable as in e-CS.

In this paper, we reformulate the e-CS problem as a zero-sum game in the spirit of the game theoretic approach in [7]. Based on this new perspective, we propose a computationally efficient scheme, called the *game-theoretic approximate matching estimator* (GAME) algorithm, to obtain a sparse approximation for the optimal game solution. In our game-theoretic approach, instead of smoothing the matrix game objective uniformly as in Nesterov’s accelerated gradient algebra [8], we approximate it by a factorized objective, which features computational advantages. The computational complexity of GAME is $\tilde{O}(MN)$, where N is the data domain dimension and M denotes the number of measurements. For comparison, we also derive a Nesterov-based convex optimization approach to solve the e-CS problem.

At the core of GAME, we employ a Bregman scheme [9] based on the ℓ_2 -norm, as compared to [7], in which we use an entropy-based Bregman function. Thanks to the mechanics of our Bregman function, the most costly operation per iteration of GAME is the application of Φ^\top (Φ is used only once). In comparison, the Nesterov-based approach requires solving three smooth optimization problems per iteration. Experimental results indicate that the proposed algorithm also has salient advantages over the Nesterov approach. We establish the theoretical convergence rate of the algorithm and show that $O(1/\varepsilon^2)$ iterations are needed to get ε additive approximation error. Nevertheless, the algorithm empirically exhibits $O(1/\varepsilon)$ convergence, matching the best known rates [8].

II. PRELIMINARIES

Throughout the paper, vectors are denoted by bold symbols. For every integer N , define $[N] \doteq \{1, \dots, N\}$. For each $i \in [N]$, let e_i denote the i -th canonical vector with one at its i -th entry, and zero everywhere else. Let k be an integer smaller than N . A vector $\mathbf{x} \in \mathbb{R}^N$ is k -sparse if it has at most k non-zero entries. The ℓ_p norm of a vector $\mathbf{a} \in \mathbb{R}^N$ is defined as $\|\mathbf{a}\|_p \doteq \left(\sum_{i=1}^N |a_i|^p\right)^{1/p}$. We also use the notation $\mathbf{a}^\top \mathbf{b}$ to denote the inner product between two vectors \mathbf{a} and \mathbf{b} .

By $[-1, 1]$, we mean the interval between -1 and 1 , whereas $\{-1, 1\}$ is the discrete set with the elements -1 and 1 . For $\theta \in \mathbb{R}_+$, we define the soft thresholding function $\mathcal{S}(\alpha, \theta)$ as

$$\mathcal{S}(\alpha, \theta) = \begin{cases} \theta & \text{if } \alpha > \theta \\ -\theta & \text{if } \alpha < -\theta \\ \alpha & \text{otherwise.} \end{cases} \quad (1)$$

For every positive τ , let $\Delta(\tau)$ denote the set of all vectors whose ℓ_1 norm is bounded by τ , and for every integer k let $\Delta(\tau, k)$ denote the set of all k -sparse vectors in $\Delta(\tau)$:

$$\Delta(\tau, k) \doteq \{\mathbf{x} : \|\mathbf{x}\|_1 < \tau \text{ and } \mathbf{x} \text{ is } k\text{-sparse}\}. \quad (2)$$

We also define the *restricted ℓ_1 -norm* of a matrix Φ as

$$\|\Phi\|_{k,1} \doteq \max_{\mathbf{x}: k\text{-sparse}} \frac{\|\Phi\mathbf{x}\|_1}{\|\mathbf{x}\|_1}. \quad (3)$$

The best k -term approximation of any vector $\mathbf{a} \in \mathbb{R}^N$, denoted $\mathcal{H}_k(\mathbf{a})$, is the k -sparse vector obtained by selecting the k -largest (in magnitude) elements of \mathbf{a} .

With respect to a strongly convex function $\mathcal{R} : \mathcal{X} \rightarrow \mathbb{R}$ in some p -norm, the Bregman divergence is defined as

$$\mathcal{B}^{\mathcal{R}}(P, Q) = \mathcal{R}(P) - \mathcal{R}(Q) - (P - Q)^\top \nabla \mathcal{R}(Q), \quad (4)$$

for $(P, Q) \in \mathcal{X} \times \mathcal{X}$, where ∇ is the gradient operator. Examples include the Euclidean norm $\mathcal{B}^{\mathcal{R}}(P, Q) = \|P - Q\|_2^2$, (with $\mathcal{X} = \mathbb{R}^M$ and $\mathcal{R}(P) = \|P\|_2^2$, which is strongly convex in ℓ_2 -norm), and the relative entropy function

$$\mathcal{B}^{\mathcal{R}}(P, Q) = \sum_{i=1}^M Q(i) - \sum_{i=1}^M P(i) + \sum_{i=1}^M P(i) \ln \frac{Q(i)}{P(i)},$$

(with $\mathcal{X} = \mathbb{R}_+^M$ and $\mathcal{R}(P) = -\sum_{i=1}^M P(i) \ln P(i)$, which is strongly convex in ℓ_1 -norm).

III. SPARSE APPROXIMATION IN ℓ_1 -NORM

Let Φ be any $M \times N$ matrix with $M \ll N$, let \mathbf{x}^* be a sparse vector in $\Delta(\tau, k)$, and let $\boldsymbol{\mu}$ be any vector in \mathbb{R}^M . Let $\mathbf{y} \doteq \Phi\mathbf{x}^* + \boldsymbol{\mu}$ denote the measurement vector. Sparse approximation in the ℓ_1 -norm refers to the following problem

$$\min_{\mathbf{x} \in \Delta(\tau, k)} \|\Phi\mathbf{x} - \mathbf{y}\|_1. \quad (5)$$

Unfortunately, since $\Delta(\tau, k)$ is not sparse, solving the problem of Equation (5) is intractable. However, in this section we will show how one can approximately solve this problem efficiently by reformulating Equation (5) as a bilinear min-max game.

Algorithm 1 The GAME algorithm

Inputs: \mathbf{y} , Φ , and parameters T , τ and $\eta > 0$.

Output: A T -sparse approximation $\hat{\mathbf{x}}$ to \mathbf{x}^* .

- 1: Set $\mathbf{P}^1 = [\mathbf{0}]_{1 \times 2M}$.
 - 2: **for** $t = 1, \dots, T$ **do**
 - 3: Calculate $\boldsymbol{\xi}^t \doteq -\Phi^\top \mathbf{P}^t$.
 - 4: Find the index $i^t \doteq \arg \max_{i \in [M]} |\xi_i^t|$.
 - 5: Set $\mathbf{x}^t = \tau \cdot \text{sign}(\xi_{i^t}^t) \cdot e_{i^t}$. (Note that $\mathbf{x}^t \in \Delta(\tau, 1)$.)
 - 6: Update $\mathbf{Q}^{t+1} = \mathbf{P}^t + \eta(\Phi\mathbf{x}^t - \mathbf{y})$.
 - 7: For each $j \in [M]$, let $\mathbf{P}_j^{t+1} = \mathcal{S}(\mathbf{Q}_j^{t+1}, 1)$.
 - 8: **end for**
 - 9: Output $\hat{\mathbf{x}} \doteq \frac{1}{T} \sum_{t=1}^T \mathbf{x}^t$.
-

Note that throughout this section we assume that an upper-bound on the ℓ_1 -norm of \mathbf{x}^* is known *a priori*. While this assumption is directly valid in many applications, we will still provide a way to efficiently compute an estimate in the e-CS problem in Section V.

To cast (5) as a min-max game, we rewrite the ℓ_1 -objective:

$$\begin{aligned} \|\Phi\mathbf{x} - \mathbf{y}\|_1 &= \sum_{j=1}^M |(\Phi\mathbf{x} - \mathbf{y})_j| = \\ &= \max_{\mathbf{E} \in \{-1, 1\}^M} \mathbf{E}^\top (\Phi\mathbf{x} - \mathbf{y}) = \max_{\mathbf{P} \in [-1, 1]^M} \mathbf{P}^\top (\Phi\mathbf{x} - \mathbf{y}). \end{aligned} \quad (6)$$

The last equality follows from the fact that the maximum of a linear program occurs at a boundary point. For every $\mathbf{x} \in \Delta(\tau)$ and $\mathbf{P} \in [-1, 1]^M$ define

$$\mathcal{L}(\mathbf{x}, \mathbf{P}) \doteq \mathbf{P}^\top (\Phi\mathbf{x} - \mathbf{y}). \quad (7)$$

The sparse approximation in the ℓ_1 -norm is then equivalent to the problem of finding the min-max optimal solution of \mathcal{L} :

$$\min_{\mathbf{x} \in \Delta(\tau, k)} \|\Phi\mathbf{x} - \mathbf{y}\|_1 = \min_{\mathbf{x} \in \Delta(\tau, k)} \max_{\mathbf{P} \in [-1, 1]^M} \mathcal{L}(\mathbf{x}, \mathbf{P}). \quad (8)$$

Although solving the optimal solution of Equation (8) is not tractable, we now introduce the *game-theoretic approximate matching estimator* (GAME) algorithm which finds a sparse approximation to this min-max optimal solution. The GAME algorithm relies on the general primal-dual approach which was originally used for learning to play repeated games [10] (see also [11] and [12]).

Remark III.1. *The pseudocode, given in Algorithm 1, describes a special GAME Algorithm, which exploit the Euclidean distance as its Bregman divergence for concreteness. Our results generalize directly for other Bregman divergences.*

The general GAME algorithm starts with finding a \mathbf{P}^1 such that $\nabla \mathcal{R}(\mathbf{P}^1) = 0$. Then at every iteration, in step 6, the algorithm finds a \mathbf{Q}^{t+1} with

$$\nabla \left(\mathcal{B}^{\mathcal{R}}(\mathbf{Q}^{t+1}, \mathbf{P}^t) - \eta \mathbf{Q}^{t+1 \top} (\Phi\mathbf{x}^t - \mathbf{y}) \right) = 0,$$

and then updates \mathbf{P}^{t+1} via the Bregman projection

$$\mathbf{P}^{t+1} \doteq \arg \min_{\mathbf{P} \in [-1, 1]^M} \mathcal{B}^{\mathcal{R}}(\mathbf{P}, \mathbf{Q}^{t+1}).$$

In general, different choices for the Bregman function may lead to different convergence bounds with different running times to perform the new projections and updates. For instance, a multiplicative update version of the algorithm (MU-GAME) can be derived by using the Bregman divergence based on the relative entropy function. Surprisingly, the derived guarantees for GAME can be shown to also hold for MU-GAME in a straightforward manner.

The lemma below is based on Theorem 3.3 in [10], and bounds the loss of a primal-dual strategy in a repeated game.

Lemma III.1. *Let T be any positive integer. Let G be a uniform upper-bound on $\|\Phi \mathbf{x}^t - f\|_2$ for all t , and define $\eta = \frac{\sqrt{M}}{\sqrt{T}G}$. Suppose $\langle (\mathbf{P}^1, \mathbf{x}^1), \dots, (\mathbf{P}^T, \mathbf{x}^T) \rangle$ is a sequence generated by the GAME algorithm after T iterations. Then,*

$$\sum_{t=1}^T \mathcal{L}(\mathbf{P}, \mathbf{x}^t) \leq \sum_{t=1}^T \mathcal{L}(\mathbf{P}^t, \mathbf{x}^t) + \frac{G\sqrt{M}}{2\sqrt{T}}.$$

Here, we invoke Lemma III.1 to show that for every positive ε , as long as $T = O\left(\frac{M\tau^2}{(\varepsilon\|\boldsymbol{\mu}\|_1)^2}\right)$, the GAME algorithm after T iterations finds a T -sparse vector $\hat{\mathbf{x}}$ with $\varepsilon\|\boldsymbol{\mu}\|_2$ multiplicative approximation error in the measurement domain.

Theorem III.1. *Let ε be any number in $(0, 1]$, and let $\hat{\mathbf{x}}$ be the output of the GAME algorithm after*

$$T = M \left(\frac{(1 + \|\Phi\|_{k+1,1})\tau}{2\varepsilon\|\boldsymbol{\mu}\|_1} \right)^2 \quad (9)$$

iterations. Then $\hat{\mathbf{x}}$ is a vector in $\Delta(\tau, T)$ with

$$\|\Phi \hat{\mathbf{x}} - \mathbf{y}\|_1 \leq (1 + \varepsilon)\|\boldsymbol{\mu}\|_1. \quad (10)$$

Proof: At every iteration t , \mathbf{x}^t is a 1-sparse solution of the minimization problem $\min \mathcal{L}(\mathbf{P}^t, \mathbf{x})$. Moreover, $\|\Phi \mathbf{x}^t - \mathbf{y}\|_2 \leq \|\Phi \mathbf{x}^t - \mathbf{y}\|_1 \leq$

$$(\|\boldsymbol{\mu}\|_1 + \|\Phi\|_{k+1,1}\|\mathbf{x}^t - \mathbf{x}^*\|_1) \leq (1 + 2\|\Phi\|_{k+1,1})\tau.$$

Let $\hat{\mathbf{P}} \doteq \frac{1}{T} \sum_{t=1}^T \mathbf{P}^t$. We have

$$\begin{aligned} \min_{\mathbf{x}} \max_{\mathbf{P}} \mathcal{L}(\mathbf{P}, \mathbf{x}) &=^a \max_{\mathbf{P}} \min_{\mathbf{x}} \mathcal{L}(\mathbf{P}, \mathbf{x}) \geq^b \min_{\mathbf{x}} \mathcal{L}(\hat{\mathbf{P}}, \mathbf{x}) \\ &\geq^c \frac{1}{T} \sum_{t=1}^T \min_{\mathbf{x}} \mathcal{L}(\mathbf{P}^t, \mathbf{x}) =^d \frac{1}{T} \sum_{t=1}^T \mathcal{L}(\mathbf{P}^t, \mathbf{x}^t) \\ &\geq^e \max_{\mathbf{P}} \mathcal{L}\left(\mathbf{P}, \frac{1}{T} \sum_{t=1}^T \mathbf{x}^t\right) - \frac{G\sqrt{M}}{2\sqrt{T}}. \end{aligned} \quad (11)$$

Equality (a) is the min-max theorem. Inequality (b) follows from the definition of max. Inequality (c) is a consequence of the linearity of \mathcal{L} and concavity of min. Equality (d) is valid by the definition of \mathbf{x}^t , and Inequality (e) follows from Lemma III.1 and linearity of \mathcal{L} . As a result,

$$\max_{\mathbf{P}} \mathcal{L}(\mathbf{P}, \hat{\mathbf{x}}) \leq \min_{\mathbf{x}} \max_{\mathbf{P}} \mathcal{L}(\mathbf{P}, \mathbf{x}) + \frac{G\sqrt{M}}{2\sqrt{T}}. \quad (12)$$

Finally, it follows from the definition of \mathcal{L} that $\max_{\mathbf{P}} \mathcal{L}(\mathbf{P}, \hat{\mathbf{x}}) = \|\Phi \hat{\mathbf{x}} - \mathbf{y}\|_1$, and

$$\min_{\mathbf{x}} \max_{\mathbf{P}} \mathcal{L}(\mathbf{P}, \mathbf{x}) = \min_{\mathbf{x} \in \Delta(\tau, T)} \|\Phi \mathbf{x} - \mathbf{y}\|_1 \leq \|\boldsymbol{\mu}\|_1. \quad \blacksquare$$

IV. EXPANDER-BASED COMPRESSED SENSING

In expander-based compressed sensing (e-CS), measurements are obtained via a normalized adjacency matrix of an expander graph.

Definition IV.1 (Expander graph). *A bipartite graph \mathcal{G} is a (k, ε, d) -expander graph if (i) \mathcal{G} is a left-regular graph with degree d , and (ii) any subset Λ of variable (left) nodes of size less than k has at least $(1 - \varepsilon)d|\Lambda|$ neighbors.*

Using a probabilistic method, one can show that for all (k, ε, N) , a random bipartite graph with $M = O\left(\frac{k \log N/k}{\varepsilon^2}\right)$, and $d = O\left(\frac{\log N/k}{\varepsilon}\right)$ is an expander graph with high probability. Explicit constructions for expander graphs also exist (c.f., [3]). Let A be the adjacency of an expander graph and let $\Phi \doteq \frac{A}{d}$. The following theorems are proved by Berinde *et al.* [6].

Theorem IV.1. *Let Φ be the normalized adjacency matrix of a (k, ε, d) -expander graph. Then, the following holds:*

$$(1 - 2\varepsilon)\|\mathbf{x}\|_1 \leq \|\Phi \mathbf{x}\|_1 \leq \|\mathbf{x}\|_1, \quad (13)$$

for any k -sparse vector \mathbf{x} and $\|\Phi\|_{k,1} \leq 1$.

Theorem IV.2. *Let Φ be the normalized adjacency matrix of a $(2k, \varepsilon, d)$ -expander graph. Let $\alpha(\varepsilon) \doteq \frac{2\varepsilon}{1-2\varepsilon}$. Let \mathbf{x}^* be any k -sparse vector in \mathbb{R}^N . Then for any vector $\mathbf{x} \in \mathbb{R}^N$*

$$\|\mathbf{x}^* - \mathbf{x}\|_1 \leq \frac{\|\mathbf{x}\|_1 - \|\mathbf{x}^*\|_1}{1 - 2\alpha(\varepsilon)} + \frac{2\|\Phi(\mathbf{x}^* - \mathbf{x})\|_1}{(1 - 2\varepsilon)(1 - 2\alpha(\varepsilon))}. \quad (14)$$

In e-CS, Φ is used as the sensing matrix, and the goal is to find a vector $\hat{\mathbf{x}}$ with $\|\mathbf{x}^* - \hat{\mathbf{x}}\|_1 \approx \|\boldsymbol{\mu}\|_1$. There are two different recovery approaches for e-CS in the literature:

Combinatorial approach [4], [5]: In this approach, message-passing algorithms are proposed for approximating \mathbf{x}^* . The recovery algorithms rely on the combinatorial properties of these graphs, and have lower computational complexity (e.g., $\mathcal{O}(N \log \frac{N}{k})$). Nevertheless, the big \mathcal{O} notation hides large constants, and the message-passing algorithms are only suitable for extremely high-dimensions.

Geometric approach [6]: This approach is based on Theorem IV.2, and uses ℓ_1 -minimization methods to find a vector $\hat{\mathbf{x}}$ with $\|\hat{\mathbf{x}}\|_1 \leq \|\mathbf{x}^*\|_1$, and $\|\Phi \hat{\mathbf{x}} - \mathbf{y}\|_1 \leq \|\boldsymbol{\mu}\|_1$. Even though this approach has the best practical performance, it is computationally inefficient ($\mathcal{O}(N^{1.5}M^2)$ running time). Moreover, since $\|\Phi \mathbf{x} - \mathbf{y}\|_1$ is not even differentiable, most gradient-based optimization methods are not directly applicable.

In Section V we propose an alternative that *approximately* solves the objective of the geometric approach.

V. EXPANDER-BASED GAME ALGORITHM

In this section, we propose an efficient algorithm, called e-GAME, that finds an estimate $\hat{\mathbf{x}}$ with $\|\hat{\mathbf{x}} - \mathbf{x}^*\|_1 = O(\|\boldsymbol{\mu}\|_1)$. A pseudocode is shown in Algorithm 2. The following lemma is key in establishing the guarantees of e-GAME.

Lemma V.1. *Let $\langle \hat{\mathbf{x}}^1, \dots, \hat{\mathbf{x}}^\Theta \rangle$ be the vectors generated by the e-GAME algorithm. Then at least one of the following two conditions holds. That is, either*

- (C1). *there exists an index t with $\|\hat{\mathbf{x}}^t\|_1 \leq \|\mathbf{x}^*\|_1$ and $\|\Phi(\hat{\mathbf{x}}^t - \mathbf{x}^*)\|_1 \leq (2 + \varepsilon)\|\boldsymbol{\mu}\|_1$; or*
(C2). *for every iteration t , $\text{Lo}^t \leq \|\mathbf{x}^*\|_1 \leq \text{Up}^t$.*

Proof: We prove Lemma V.1 by induction. First consider $t = 0$ with Theorem IV.1 and the triangle inequality:

$$(1 - 2\varepsilon)\|\mathbf{x}^*\|_1 \leq \|\Phi\mathbf{x}^*\|_1 \leq \|\mathbf{y}\|_1 + \|\boldsymbol{\mu}\|_1.$$

Assume that Condition (C2) holds for $t - 1$. We now show that it is also valid for index t via two different cases:

Case 1: $\|\Phi\hat{\mathbf{x}}^t - \mathbf{y}\|_1 > (1 + \varepsilon)\|\boldsymbol{\mu}\|_1$. If $\|\mathbf{x}^*\|_1 \leq \tau^t$ then

$$\min_{\|\mathbf{x}\|_1 \leq \tau^t} \|\Phi\mathbf{x} - \mathbf{y}\|_1 \leq \|\Phi\mathbf{x}^* - \mathbf{y}\|_1 = \|\boldsymbol{\mu}\|_1 < \frac{\|\Phi\hat{\mathbf{x}}^t - \mathbf{y}\|_1}{(1 + \varepsilon)},$$

which contradicts the $(1 + \varepsilon)$ approximation guarantee of the GAME algorithm. Therefore we must have $\|\mathbf{x}^*\|_1 \geq \tau^t = \text{Lo}^t$. It also follows from the induction hypothesis that $\|\mathbf{x}^*\|_1 \leq \text{Up}^{t-1} = \text{Up}^t$.

Case 2: $\|\Phi\hat{\mathbf{x}}^t - \mathbf{y}\|_1 \leq (1 + \varepsilon)\|\boldsymbol{\mu}\|_1$. In this case, if $\tau^t \leq \|\mathbf{x}^*\|_1$, then we have $\|\hat{\mathbf{x}}^t\|_1 \leq \|\mathbf{x}^*\|_1$ and

$$\|\Phi(\hat{\mathbf{x}}^t - \mathbf{x}^*)\|_1 \leq \|\Phi\hat{\mathbf{x}}^t - \mathbf{y}\|_1 + \|\Phi\mathbf{x}^* - \mathbf{y}\|_1 \leq (2 + \varepsilon)\|\boldsymbol{\mu}\|_1,$$

which is Condition (C1). Therefore, if (C1) is not valid then we must have $\text{Up}^t = \tau^t > \|\mathbf{x}^*\|_1$. Also again from the induction hypothesis we get $\|\mathbf{x}^*\|_1 \geq \text{Lo}^{t-1} = \text{Lo}^t$. ■

The following theorem proves that at least one estimate $\mathcal{H}_k(\hat{\mathbf{x}}^t)$ is sufficiently close to \mathbf{x}^* .

Theorem V.1. *Let ε and δ be any two positive numbers, and let $\langle \hat{\mathbf{x}}^1, \dots, \hat{\mathbf{x}}^\Theta \rangle$ be the vectors generated by the e-GAME algorithm. Then, at least one of the following two conditions holds:*

- (C1). *There exists an index t with*

$$\|\mathbf{y} - \Phi\mathcal{H}_k(\hat{\mathbf{x}}^t)\|_1 \leq \left(1 + \frac{4(2 + \varepsilon)}{(1 - 2\varepsilon)(1 - 2\alpha(\varepsilon))}\right) \|\boldsymbol{\mu}\|_1.$$

- (C2).

$$\|\mathbf{y} - \mathcal{H}_k(\hat{\mathbf{x}}^\Theta)\|_1 \leq \frac{2\delta\|\mathbf{x}^*\|_1}{1 - 2\alpha(\varepsilon)} + \left(1 + \frac{4(2 + \varepsilon)}{(1 - 2\varepsilon)(1 - 2\alpha(\varepsilon))}\right) \|\boldsymbol{\mu}\|_1$$

Proof: The proof of Theorem V.1 relies on Lemma V.1. If Condition (C1) is satisfied for some index t , then Theorem IV.2 implies that

$$\|\mathbf{x}^* - \hat{\mathbf{x}}^t\|_1 \leq \frac{2(2 + \varepsilon)\|\boldsymbol{\mu}\|_1}{(1 - 2\varepsilon)(1 - 2\alpha(\varepsilon))},$$

whereas if Condition (C2) is satisfied, then we have

$$\|\hat{\mathbf{x}}^t\|_1 - \|\mathbf{x}^*\|_1 \leq \text{Up}^t - \|\mathbf{x}^*\|_1 \leq \text{Up}^t - \text{Lo}^t \leq \frac{\text{Up}^0 - \text{Lo}^0}{2^t},$$

Algorithm 2 The e-GAME Algorithm

Inputs: \mathbf{y} , Φ , and parameters ε , and δ .

Output: An approximation $\hat{\mathbf{x}}$ for the vector \mathbf{x}^* .

- 1: Set $\text{Lo}^0 = 0$, $\text{Up}^0 = \frac{\|\mathbf{y}\|_1 + \|\boldsymbol{\mu}\|_1}{(1 - 2\varepsilon)}$, and $\Theta = \log_2\left(\frac{2}{\delta}\right)$.
 - 2: **for** $t = 1, \dots, \Theta$ **do**
 - 3: Let $\hat{\mathbf{x}}^t$ be the solution of the GAME algorithm with $\tau^t = \frac{\text{Lo}^{t-1} + \text{Up}^{t-1}}{2}$ and T^t of Equation (9).
 - 4: **if** $\|\Phi\hat{\mathbf{x}}^t - \mathbf{y}\|_1 \leq (1 + \varepsilon)\|\boldsymbol{\mu}\|_1$ **then**
 - 5: Set $\text{Up}^t = \tau^t$, and $\text{Lo}^t = \text{Lo}^{t-1}$.
 - 6: **else**
 - 7: Set $\text{Lo}^t = \tau^t$, and $\text{Up}^t = \text{Up}^{t-1}$.
 - 8: **end if**
 - 9: **end for**
 - 10: Output $\hat{\mathbf{x}} \doteq \arg \min_{\mathcal{H}_k(\hat{\mathbf{x}}^t)} \|\Phi\mathcal{H}_k(\hat{\mathbf{x}}^t) - \mathbf{y}\|_1$.
-

at every iteration t . In this case, if $\Theta = \log_2\left(\frac{2}{\delta}\right)$, then Theorem IV.1 implies that $\|\hat{\mathbf{x}}^\Theta\|_1 - \|\mathbf{x}^*\|_1 \leq$

$$\frac{(\|\mathbf{y}\|_1 + \|\boldsymbol{\mu}\|_1)\delta}{2} \leq (\|\mathbf{y}\|_1 - \|\boldsymbol{\mu}\|_1)\delta \leq \|\Phi\mathbf{x}^*\|_1\delta \leq \|\mathbf{x}^*\|_1\delta.$$

Furthermore, since both $\|\mathbf{x}^*\|_1$ and $\|\hat{\mathbf{x}}\|_1$ are smaller than τ^Θ , Theorem III.1 guarantees that

$$\|\Phi\hat{\mathbf{x}}^\Theta - \mathbf{y}\|_1 \leq (1 + \varepsilon) \min_{\|\mathbf{x}\|_1 \leq \tau^\Theta} \|\Phi\mathbf{x} - \mathbf{y}\|_1 \leq (1 + \varepsilon)\|\boldsymbol{\mu}\|_1.$$

Therefore, Theorem IV.2 implies that

$$\|\mathbf{x}^* - \hat{\mathbf{x}}^\Theta\|_1 \leq \frac{\delta\|\mathbf{x}^*\|_1}{1 - 2\alpha(\varepsilon)} + \frac{2(2 + \varepsilon)\|\boldsymbol{\mu}\|_1}{(1 - 2\varepsilon)(1 - 2\alpha(\varepsilon))}.$$

Finally, let \mathbf{b} be any vector in \mathbb{R}^N . Since \mathbf{x}^* is k -sparse, from the triangle inequality and the definition of the best k -term approximation we have

$$\|\mathbf{x}^* - \mathcal{H}_k(\mathbf{b})\|_1 \leq \|\mathbf{x}^* - \mathbf{b}\|_1 + \|\mathbf{b} - \mathcal{H}_k(\mathbf{b})\|_1 \leq 2\|\mathbf{x}^* - \mathbf{b}\|_1.$$

Now, observe that $\mathbf{x}^* - \mathcal{H}_k(\hat{\mathbf{x}}^t)$ is always $2k$ -sparse. As a result, Theorem IV.1 yields that at every iteration t

$$\begin{aligned} \|\mathbf{y} - \Phi\mathcal{H}_k(\hat{\mathbf{x}}^t)\|_1 &\leq \|\boldsymbol{\mu}\|_1 + \|\Phi(\mathbf{x}^* - \mathcal{H}_k(\hat{\mathbf{x}}^t))\|_1 \\ &\leq \|\boldsymbol{\mu}\|_1 + \|(\mathbf{x}^* - \mathcal{H}_k(\hat{\mathbf{x}}^t))\|_1 \leq \|\boldsymbol{\mu}\|_1 + 2\|(\mathbf{x}^* - \hat{\mathbf{x}}^t)\|_1. \end{aligned} \quad (15)$$

Combining Theorem V.1 and Theorem IV.1 provides the following corollary.

Corollary V.1. *Let \mathbf{x}^* be any k -sparse vector in \mathbb{R}^N , and let $\boldsymbol{\mu}$ be any noise vector in \mathbb{R}^M . Let $\text{SNR}_1 \doteq \frac{\|\mathbf{x}^*\|_1}{\|\boldsymbol{\mu}\|_1}$. Then the e-GAME algorithm recovers a k -sparse vector $\hat{\mathbf{x}}$ with $\|\mathbf{x}^* - \hat{\mathbf{x}}\|_1 = \mathcal{O}(\|\boldsymbol{\mu}\|_1)$. Moreover, the overall recovery time is $\mathcal{O}\left(MN \log\left(\frac{N}{k}\right) \text{SNR}_1^2 \log \text{SNR}_1\right)$.*

Proof: Since $\mathbf{x}^* - \hat{\mathbf{x}}$ is $2k$ -sparse, Theorem IV.1 and Theorem V.1 with $\delta = \frac{1}{\text{SNR}_1}$ guarantee that

$$(1 - 2\varepsilon)\|\mathbf{x}^* - \hat{\mathbf{x}}\|_1 \leq \|\mathbf{y} - \Phi\hat{\mathbf{x}}\|_1 + \|\boldsymbol{\mu}\|_1 = \mathcal{O}(\|\boldsymbol{\mu}\|_1).$$

To calculate the overall running time of the algorithm note that the e-GAME algorithm requires $\Theta = \mathcal{O}(\log \text{SNR}_1)$ iterations.

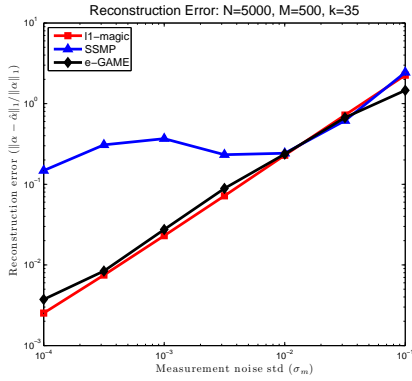


Fig. 1: Approximate recovery experiments with ℓ_1 -magic (ℓ_1 -minimization), SSMP (message passing), and e-GAME (Game-theoretic) algorithms. The measurement noise standard deviation ranges from 10^{-5} to 10^{-1} , and the approximation error is measured as $\|\mathbf{x}^* - \hat{\mathbf{x}}\|_1 / \|\mathbf{x}^*\|_1$.

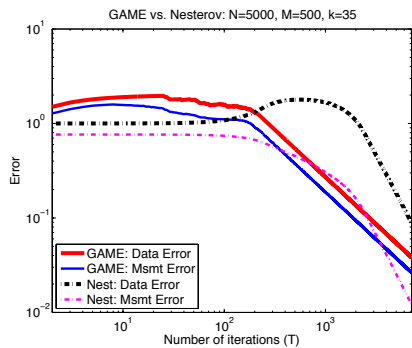


Fig. 2: The dependency between the approximation error and the number of iterations of the GAME Algorithm.

At each such iteration, the GAME algorithm requires $T = \mathcal{O}(M\text{SNR}_1^2)$ iterations (Equation (9)) in which the bottleneck is one adjoint operation (i.e., calculating $\Phi^\top \mathbf{P}^t$), which can be done efficiently in $\mathcal{O}(N \log \frac{N}{k})$. ■

Remark V.1. *An alternative approach is to use the Nesterov’s smoothing method for approximately solving non-smooth objective functions [8]. We omit the details of this implementation due to lack of space. With the Nesterov method, we still need $\mathcal{O}(\log \text{SNR}_1)$ outer iterations, while the number of inner iterations can be reduced to $T = \mathcal{O}(M\text{SNR}_1)$. However, each inner iteration of the Nesterov method requires solving three smooth convex optimization problems, and is much more complicated than calculating one matrix-vector multiplication.*

VI. EXPERIMENTAL RESULTS

In this section, we provide experimental results supporting the fidelity of the e-GAME algorithm. Figure 1 compares the performance of the e-GAME algorithm with ℓ_1 -magic [13] which directly solves the ℓ_1 minimization problem, and with SSMP [5] which is a combinatorial message-passing algorithm. Throughout these experiments we use $N = 5000$,

$M = 500$ and $k = 35$ for illustration to demonstrate the typical behavior of the algorithms for other N, M and k .

Figure 1 plots the average reconstruction error $\frac{\|\mathbf{x}^* - \hat{\mathbf{x}}\|_1}{\|\mathbf{x}^*\|_1}$ as a function of the noise level. Interestingly, the approximation error of the e-GAME algorithm is very close to the approximation error of the ℓ_1 -magic algorithm, and significantly lower than the error of the SSMP algorithm, while our implementation of the e-GAME algorithm only requires a fraction of the running time of the ℓ_1 -magic algorithm.

Figure 2 plots the data-domain and the measurement-domain errors of GAME and Nesterov approximation algorithms as a function of the number of iterations. It shows that when GAME is used, both the data-domain and the measurement-domain losses consistently decrease. We also observed that the GAME algorithm always has the tendency of selecting a sparse candidate. Moreover, after some initial burn-in, the rates of convergence are approximately $\frac{1}{T}$ (as opposed to the slower rate of $\frac{1}{\sqrt{T}}$, which was expected from theory). In contrast, the Nesterov method has the initial tendency of selecting a dense solution, and it is only after 1000 iterations that a decrease in the data-domain loss is observed.

VII. CONCLUSIONS

We proposed a scalable algorithm to solve the sparse-approximation problem in the ℓ_1 -norm by reformulating the problem as a min-max game. We proved that the algorithm requires $\tilde{\mathcal{O}}(1/\varepsilon^2)$ iterations to obtain ε multiplicative approximation error. However, the algorithm empirically needs only $\tilde{\mathcal{O}}(1/\varepsilon)$ iterations. Future work will focus on closing the gap between the theoretical and empirical convergence rates.

REFERENCES

- [1] E. Candès, J. Romberg, and T. Tao, “Stable signal recovery from incomplete and inaccurate measurements,” *Comm. Pure Appl. Math.*, vol. 59, no. 8, pp. 1207–1223, 2006.
- [2] D. Donoho, “Compressed Sensing,” *IEEE transactions on Information Theory*, Vol. 52 (4), pp. 1289–1306, April 2006.
- [3] A. Gilbert and P. Indyk, “Sparse recovery using sparse matrices,” *Proceedings of the IEEE*, vol. 98, pp. 937–947, 2010.
- [4] S. Jafarpour, W. Xu, B. Hassibi, and R. Calderbank, “Efficient and robust compressed sensing using optimized expander graphs,” *IEEE Transactions on Information Theory*, vol. 55, pp. 4299–4308, 2009.
- [5] R. Berinde and P. Indyk, “Sequential sparse matching pursuit,” in *Proc. 46th Allerton Conf.*, 2009, pp. 36–43.
- [6] R. Berinde, A. Gilbert, P. Indyk, H. Karloff, and M. Strauss, “Combining geometry and combinatorics: a unified approach to sparse signal recovery,” in *Proc. 46th Allerton Conf.*, 2008, pp. 798–805.
- [7] S. Jafarpour, V. Cevher, and R. Schapire, “Game Theory meets Compressed Sensing,” in *Proc. 36th ICASSP Conf.*, 2011.
- [8] Y. Nesterov, *Introductory lectures on convex optimization: A basic course*. Springer, 2004.
- [9] Y. Censor and S. Zenios, *Parallel Optimization: Theory, Algorithms and Applications*. Oxford University Press, 1997.
- [10] E. Hazan, “A survey: The convex optimization approach to regret minimization,” *Preprint available at http://ie.technion.ac.il/~Eehazan/papers/OCO-survey.pdf*, 2011.
- [11] Y. Freund and R. Schapire, “Adaptive game playing using multiplicative weights,” *Games. Econ. Behav.*, 2011.
- [12] A. J. Grove, N. Littlestone, and D. Schuurmans, “General convergence results for linear discriminant updates,” *Machine Learning*, vol. 43, pp. 173–210, 2001.
- [13] E. Candès and J. Romberg, “ ℓ_1 -MAGIC: Recovery of Sparse Signals via Convex Programming,” *Available at http://www.acm.caltech.edu/1lmagic*, 2005.