ON FINITE ALPHABET COMPRESSIVE SENSING

Abhik Kumar Das and Sriram Vishwanath

Dept. of E.C.E., The University of Texas at Austin, USA

ABSTRACT

This paper considers the problem of compressive sensing over a finite alphabet, where the finite alphabet may be inherent to the nature of the data or a result of quantization. We show that there are significant benefits to analyzing the problem while incorporating its finite alphabet nature, versus ignoring it and employing a conventional real alphabet based toolbox. Specifically, when the alphabet is finite, our techniques have a lower sample complexity compared to real-valued compressive sensing for low levels of sparsity, facilitate constructive designs of sensing matrices based on coding-theoretic techniques, and allow for lesser amount of data storage.

***Index Terms*—** compressive sensing, finite alphabet.

1. INTRODUCTION

Compressive sensing has witnessed an explosion of research and literature in recent years, finding useful applications in diverse fields [1, 2, 3, 4]. The theory behind compressive sensing permits the sensing and recovery of sparse high-dimensional signals using a small number of linear measurements [5, 6, 7]. There are multiple practical algorithms for accurate recovery of real-valued sparse signals from their linear measurements, in the presence or absence of noise [8, 9, 10, 11, 12, 13]. From an analytical perspective, there is a large body of literature on necessary and sufficient conditions for accurate reconstruction of sparse signals.

In practice, signals are not always real-valued. For example, opinion polls, ranking information, commodity sales numbers, and counting data sets including arrivals at a queue/server are inherently discrete-valued. Moreover, some of what might otherwise be regarded as continuous-valued data sets are conventionally "binned" into finite alphabet sets; examples include rainfall data, sensor data and power generation data. In such cases, knowledge of the nature of alphabet can prove to be useful, which together with the underlying sparsity property can lead to alternate and potentially efficient algorithms for finite alphabet compressive sensing.

In this paper, we consider a setup where the sensed information belongs to a known finite alphabet. We treat this alphabet as a subset of a suitable finite field; this allows us to use tools from algebraic coding theory to construct sensing matrices and design efficient sparse signal recovery algorithms. In this process, we establish a deeper connection between the areas of algebraic coding theory and compressive sensing than what is currently understood in literature [14, 15].

Motivation: It is known that the recovery of sparse signals from their linear measurements in compressive sensing reduces to solving a ℓ_0-minimization problem. An advantage of analyzing compressive sensing in the finite field domain is that ℓ_0-minimization is solvable in polynomial time for certain families of sensing matrices. In contrast, ℓ_0-minimization is non-convex and NP-hard in the real-valued domain; therefore, its convex relaxed version, the ℓ_1-minimization problem, is solved in its place, that gives the correct answer if the sensing matrix satisfies some incoherence property like RIP. Another important reason for finite-valued analysis of compressive sensing is storage space. Though real values are useful analytical abstract artifacts, in practice, values must be stored and processed in form of discrete alphabet. The amount of storage space needed is an important area of concern for applications of compressive sensing. Our methodology not only affords a lower sample complexity under certain settings, but also requires lesser amount of storage space in terms of bits of information needed for exact signal reconstruction.

We wish to emphasize that the tools we use are well-known to the coding community; what makes it interesting is their connection and relevance to compressive sensing. Our main application domain is in tracking discrete-valued time-series data; we show that our approach does not suffer from error accumulation like real-valued compressive sensing.

Relation to Prior Work: The fact that real-valued compressive sensing allows for recovery of sparse signals based on linear measurements is reminiscent of error correction in linear channel codes and compression by lossless source codes over finite alphabet or fields [16, 17]. Such similarities have been identified in existing literature to serve varied goals. For example, the use of bipartite expander graphs to design sensing matrices over reals is investigated in [18]. The connection between real-valued compressive sensing and linear channel codes is explored in [15], by viewing sparse signal compression as syndrome-based source coding over real numbers and making use of linear codes over large finite fields. The design of real-valued sensing matrices based on LDPC codes is examined in [19] and [20]. For real-valued compressive sensing over finite alphabet, the approaches that have been examined include approximate message passing

[21], recovery based on sphere decoding and semi-definite relaxation [22]. However, an algebraic understanding of compressive sensing, particularly over finite fields, is yet limited, which is the main contribution of this paper.

Due to limitation of space, the proofs of the results are not presented in this paper. The audience can refer to [23] for a detailed version of the paper, along with the proofs.

2. SYSTEM MODEL

Notation: We use $\mathbb{F}_q$ to represent the finite field with q elements, where q is a prime or power of a prime. For any field $\mathbb{F}$, we use $\mathbb{F}[x]$ to denote the polynomial ring in x with coefficients from $\mathbb{F}$. For $n \in \mathbb{N}$ and $\mathbf{x} \in \mathbb{F}^n$, we use $\mathrm{wt}(\mathbf{x})$ to denote the number of non-zero elements in $\mathbf{x}$. For $x \in \mathbb{R}$, we use $\lfloor x \rfloor$ and $\lceil x \rceil$ to represent its floor and ceiling values.

Given $b, n, q \in \mathbb{N}$ with $b < n$, and a finite alphabet $\mathcal{A} \subseteq \mathbb{R}$ with $0 \in \mathcal{A}$ and $|\mathcal{A}| = q$, we consider the following ensemble:

$$\mathcal{S} = \{\mathbf{x} = (x_1, x_2, \ldots, x_n) \in \mathcal{A}^n : \mathrm{wt}(\mathbf{x}) \leq b\}.$$

This ensemble represents the space of n-dimensional signals that are at most b-sparse with entries from $\mathcal{A}$. We assume that q is a prime number or its power, and consider a bijective mapping $\phi : \mathcal{A} \to \mathbb{F}_q$ with the restriction $\phi(0) = 0$, i.e., $0 \in \mathbb{R}$ gets mapped as the zero of $\mathbb{F}_q$. This allows us to interpret $\mathcal{A}$ as $\mathbb{F}_q$ and we define the following set of vectors:

$$\mathcal{S}_q = \{\mathbf{x} = (\phi(z_1), \ldots, \phi(z_n)) : (z_1, \ldots, z_n) \in \mathcal{S}\}.$$

By construction, the vectors in $\mathcal{S}_q$ are at most b-sparse.

We develop a framework for efficient compression of any $\mathbf{x} \in \mathcal{S}_q$ using linear measurements generated by the process

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{n}, \tag{1}$$

where $\mathbf{A} \in \mathbb{F}_q^{m \times n}$ is the sensing matrix, $\mathbf{n} \in \mathbb{F}_q^m$ is the noise vector and $\mathbf{y} \in \mathbb{F}_q^m$ is the vector of measurements. Note that $\mathbf{y}$ can be thought of as a noisy compressed version of $\mathbf{x}$. The overall goal of the problem setting is to design $\mathbf{A}$ such that $\mathbf{x}$ can be recovered accurately and efficiently from $\mathbf{y}$.

Given a uniform distribution over the vectors in $\mathcal{S}_q$, source coding theorem [24] states that the number of measurements required to characterize $\mathbf{x} \in \mathcal{S}_q$ is at least $\log_2 |\mathcal{S}_q| = \Omega(b \log(n/b))$ for $b < n/2$. Here, we provide schemes for designing $m = \Theta(b\lceil \log_q n \rceil)$ measurements to recover $\mathbf{x}$. This matches the lower bound on number of measurements, in order sense, for the case $b = O(n^\alpha)$, $\alpha \in [0, 1)$.

Field Lifting: A critical algebraic tool that we utilize for designing the sensing matrices is field lifting. The use of this concept allows one to embed the existing system setup in a high-dimensional space that offer more degrees of freedom. The concept is briefly described as follows; its detailed description as well as that of the relevant algebraic aspects is given in [23]. Given $s \in \mathbb{N}$, we consider a primitive polynomial $p(x)$ of degree s over $\mathbb{F}_q$ and its root α, that is a primitive element of $\mathbb{F}_{q^s}$. We assume that the number of measurements satisfies $m = m's$, and define matrix mapping $\phi_s : \mathbb{F}_q^{m \times n} \to \mathbb{F}_{q^s}^{m' \times n}$ and vector mapping $\psi_s : \mathbb{F}_q^m \to \mathbb{F}_{q^s}^{m'}$. ϕ_s maps $\mathbf{C} = [c_{ij}] \in \mathbb{F}_q^{m \times n}$ to $\phi_s(\mathbf{C}) = [c'_{kl}] \in \mathbb{F}_{q^s}^{m' \times n}$, where $c'_{kl} = \sum_{t=0}^{s-1} c_{(k-1)s+t+1,l}\alpha^t$. Likewise, ψ_s maps $\mathbf{c} = [c_i] \in \mathbb{F}_q^m$ to $\psi_s(\mathbf{c}) = [c'_k] \in \mathbb{F}_{q^s}^{m'}$, where $c'_k = \sum_{t=0}^{s-1} c_{(k-1)s+t+1}\alpha^t$. Note that fixing $p(x)$ and α make these mappings bijective.

3. NOISELESS MEASUREMENTS

In this section, we analyze the problem of recovering $\mathbf{x} \in \mathcal{S}_q$ in absence of noise, i.e., $\mathbf{n} = \mathbf{0}$. Note that this situation resembles the process of syndrome decoding in linear codes, where $\mathbf{x}, \mathbf{y}$ and $\mathbf{A}$ play the roles of error vector, syndrome vector and parity check matrix of the linear code respectively [25]. We exploit this idea for designing $\mathbf{A}$ and algorithms for recovering $\mathbf{x}$ from $\mathbf{y}$. We refer to a linear code $\mathcal{C}$ as an $[N, K, D]_q$ code ($N, K, D \in \mathbb{N}$ and q is a prime or its power) if the code alphabet is $\mathbb{F}_q$, codeword length is N, number of codewords is q^K and Hamming distance between any two codewords is at least D. Then the following theorem holds:

Theorem 3.1. *Given $m = m's$, it is possible to exactly recover $\mathbf{x} \in \mathcal{S}_q$ from $\mathbf{y}$ if $\phi_s(\mathbf{A})$ is the parity check matrix of a $[n, n-m', d]_{q^s}$ linear code with $n > m', d > 2b$.*

Designing $\phi_s(\mathbf{A})$ as the parity check matrix of a suitably structured Reed-Solomon code gives the following corollary:

Corollary 3.2. *Given $m = 2bs$ and $s \geq \lceil \log_q n \rceil$, it is possible to exactly recover $\mathbf{x} \in \mathcal{S}_q$ from $\mathbf{y}$ using $O(nbs^2)$ field operations in $\mathbb{F}_q$ if $n > 2b$ and $\phi_s(\mathbf{A})$ is the parity check matrix of a $[n, n-2b, 2b+1]_{q^s}$ Reed-Solomon code.*

In general, any family of linear codes that admits a polynomial time syndrome decoding algorithm can be used for constructing sensing matrices, in place of Reed-Solomon codes. Examples include BCH codes, LDGM codes and special classes of LDPC codes based on expanders.

Number of measurements: Corollary 3.2 suggests that one can design $m = 2b\lceil \log_q n \rceil$ measurements with the recovery algorithm requiring $O(nb(\log n)^2)$ field operations in $\mathbb{F}_q$. This scaling of m is order-wise optimal for $b = O(n^\alpha)$, $\alpha \in [0, 1)$, with respect to the lower bound of $\Omega(b \log(n/b))$. A sufficient condition that ensures ℓ_1-minimization leads to accurate recovery in real-valued compressive sensing is that the sensing matrix satisfies RIP of order $2b$ with parameter $\delta_{2b} < (\sqrt{2} - 1)$ [10]. A convenient way of generating RIP matrices is by choosing its entries from a sub-Gaussian distribution in an i.i.d. fashion. Given $\delta \in (\sqrt{2} - 1, 1)$ and arbitrary $\kappa_1 > 0$, if the number of measurements satisfies $m \geq 2\kappa_1 b \log_e(n/2b)$, then exact recovery is possible using ℓ_1-minimization with probability $\geq 1 - 2\exp(-\kappa_2 m)$, where κ_2 depends on δ, κ_1. Therefore, for $b \leq 0.5q^{-(\kappa_1 \log_e q)^{-1}} n^{1-(\kappa_1 \log_e q)^{-1}}$ (i.e., sparsity level is below some threshold) the number of measurements required for the finite field (or finite alphabet) framework is smaller compared to real-valued compressive sensing.

Storage space: The storage space needed for the measurements (i.e., $\mathbf{y}$) is at most $2b\lceil\log_q n\rceil \log_2 q$ bits, that lies between $2b\log_2 n$ and $2b\log_2 n + 2b\log_2 q$ bits. The storage space taken by real-values is in theory, infinite, and in practice, with j-bit quantization, is linear in j. For real-valued compressive sensing and same number of measurements, this amounts to $2jb\lceil\log_q n\rceil$ bits of storage space. Note that we have $j > \log_2 q$, as at least $\log_2 q$ bits are needed to resolve among the elements in a finite alphabet of size q. This gives storage space of at least $2b\lceil\log_q n\rceil \log_2 q$ bits. Therefore, the storage requirement for the finite field framework is smaller compared to the real-valued framework if $j > \log_2 q$.

Thus, the algebraic approach offers benefits in terms of number of measurements as well as storage space (in bits), provided the sparsity levels are below some threshold.

4. NOISY MEASUREMENTS

In this section, we analyze the problem of recovering $\mathbf{x} \in \mathcal{S}_q$ in presence of measurement noise. Here, we recover $\mathbf{x}$ from $\mathbf{y}$ in two steps. First, we eliminate the effect of errors introduced by $\mathbf{n}$, using error correction capability of linear codes. Next, we retrieve $\mathbf{x}$ as described in Section 3. Due to limited space, we consider only the probabilistic noise model in this paper, the analysis for worst-case noise is presented in [23].

The probabilistic noise model is widely used for modeling errors resulting from transmissions across communication channels or networks. For the sake of simplicity, we assume that $\mathbf{n}$ is generated by m independent uses of a q-ary symmetric channel with crossover probability $\lambda \in (0, 1 - q^{-1})$; similar analysis can be performed for noise generated by general probability distributions. In other words, if $\mathbf{n} = [n_1\ n_2\ \cdots\ n_m]^T$, n_i has the probability distribution $P(n_i = a) = \lambda/(q-1)$ for $a \in \mathbb{F}_q \backslash \{0\}$ and $1-\lambda$ for $a = 0$.

We say that a linear code $\mathcal{C}$ achieves probability of error of at most P_e over a channel if $\max_{\mathbf{c}\in\mathcal{C}} P_e(\mathbf{c}) \leq P_e$, where $P_e(\mathbf{c})$ refers to the probability that codeword $\mathbf{c}$ is decoded erroneously by a nearest neighbor codeword decoder, conditioned on the fact that $\mathbf{c}$ was originally sent across the channel. We also define $H_q(x) \triangleq -x\log_q x - (1-x)\log_q(1-x) + x\log_q(q-1)$, $x \in (0,1)$. Then the following theorem holds:

Theorem 4.1. *Given $\lambda \in (0, 1-q^{-1})$, $m = cm'$, $m' \geq m''s$ and $c > 1/(1-H_q(\lambda))$, it is possible to exactly recover $\mathbf{x} \in \mathcal{S}_q$ from $\mathbf{y}$ with probability $\geq (1-P_e)$ if $\mathbf{A} = \mathbf{G}\mathbf{A}'$, where $\mathbf{G}$ is the generator matrix of a $[m, m', d]_q$ linear code that achieves probability of error of at most P_e over a symmetric channel with crossover probability λ, and some set of $m''s$ rows of $\mathbf{A}'$ forms $\mathbf{A}''$ such that $\phi_s(\mathbf{A}'')$ is the parity check matrix of a $[n, n-m'', d']_{q^s}$ linear code with $d' > 2b$.*

$\mathbf{A}'$ can be constructed to have $\mathbf{A}''$ as its sub-matrix, with $\phi_s(\mathbf{A}'')$ as the parity check matrix of a suitable Reed-Solomon code. One family of linear codes that achieve small error probabilities over q-ary symmetric channel is concatenated codes [25, 26]. For example, given $\lambda \in (0, 1-q^{-1})$, $\epsilon \in (0, 1-H_q(\lambda))$, $\rho \in (0,1)$ and large enough $t \in \mathbb{N}$, it is possible to design a concatenated $[N, K, D]_q$ linear code with $N = tq^{\lfloor\epsilon t\rfloor}$ and $K = \lfloor\epsilon t\rfloor\lceil\rho q^{\lfloor\epsilon t\rfloor}\rceil$, whose decoding algorithm requires $O(N^2\log N)$ operations in $\mathbb{F}_q$ and reals [25]. Furthermore, the code achieves probability of error of at most $q^{-c(\epsilon,\rho)N}$ over the symmetric channel with crossover probability λ, where $c(\epsilon, \rho)$ is a positive constant that depends on q, ϵ, ρ. We refer to this linear code as $\mathcal{C}_{con}(t, \epsilon, \rho)$. Designing $\mathbf{G}$ to be its generator matrix results in the following corollary:

Corollary 4.2. *Given $\lambda \in (0, 1-q^{-1})$, $\rho \in (0,1)$, $\epsilon \in (0, 1-H_q(\lambda))$ and $s \geq \lceil\log_q n\rceil$, it is possible to exactly recover $\mathbf{x} \in \mathcal{S}_q$ from $\mathbf{y}$ with probability $\geq (1-q^{-cbs})$ ($c > 0$ depends only on q, ϵ, ρ) for large enough values of b, n using $O((n + b\log(bs))bs^2)$ operations in $\mathbb{F}_q, \mathbb{R}$ if $n > 2b$ and $\mathbf{A} = \mathbf{G}\mathbf{A}'$, where $\mathbf{G}$ is the generator matrix of code $\mathcal{C}_{con}(\lceil t^*\rceil, \epsilon, \rho)$, $t^* \in \mathbb{R}$ being the solution to $\epsilon\rho x q^{\epsilon x} = 4qbs$, and some set of $2bs$ rows of $\mathbf{A}'$ forms $\mathbf{A}''$ such that $\phi_s(\mathbf{A}'')$ is the parity check matrix of a $[n, n-2b, 2b+1]_{q^s}$ Reed-Solomon code.*

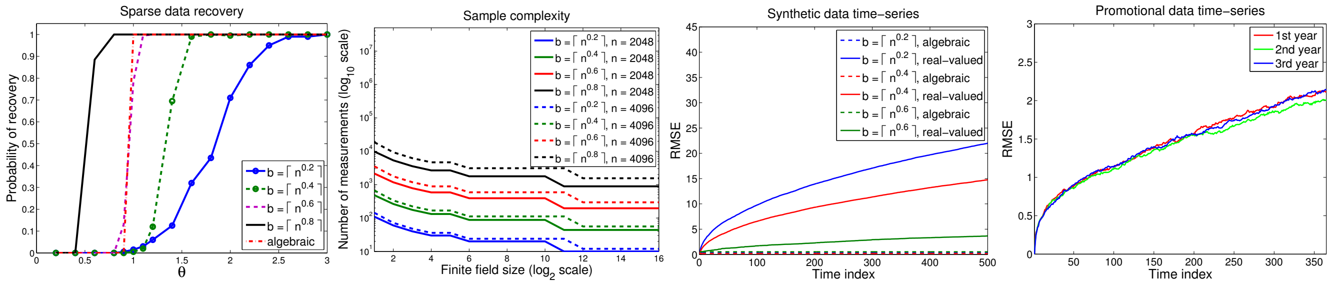
Corollary 4.2 suggests one can design $m = \Theta(b\lceil\log_q n\rceil)$ measurements and the recovery process requires $O((n + b\log b + b\log\log n)b(\log n)^2)$ field operations in $\mathbb{F}_q, \mathbb{R}$. This scaling of m is order-wise optimal for $b = O(n^\alpha)$, $\alpha \in [0,1)$, with respect to the lower bound of $\Omega(b\log(n/b))$. Note that there are no theoretical guarantees for real-valued compressing sensing with this noise model; most of the guarantees are designed for Gaussian noise or ℓ_2-norm bounded noise.

5. SIMULATION RESULTS

In this section, we present simulation results showing the utility of our approach (detailed description given in [23]).

Synthetic sparse data: We show the effect of sparsity levels on our approach vs. real-valued compressive sensing using synthetically generated sparse data. For this, we choose $n = 1024$ and sparsity levels $b = \lceil n^r\rceil$, $r = 0.2, 0.4, 0.6, 0.8$. We set $m = 2\lfloor\theta b\rfloor\lceil\log_q n\rceil$, where $q = 256$ and θ is varied from 0.2 to 3. The details of constructing $\mathbf{x}$, sensing matrices for algebraic and real-valued approaches, definitions of error-free events and probability of recovery are given in [23].

Figure 1a shows the plot of θ vs. probability of recovery. The fact that the number of measurements required for the algebraic approach is lesser compared to that of real-valued compressive sensing for sparsity levels $b = \lceil n^r\rceil$, $r = 0.2, 0.4, 0.6$, corroborates the remark made about sample complexity in Section 3. This also implies lesser storage space for the measurement vector and sensing matrix for these situations with low sparsity levels, since every symbol in the field $\mathbb{F}_q$ can be represented in $\log_2 q = 8$ bits whereas



(a) (b) (c) (d)

Fig. 1: Simulation results for (a),(b) synthetic sparse data and (c),(d) discrete-valued time series.

reals are potentially assigned more bits (assigning lesser bits to reals would give quantization error as overhead).

Since $m = 2b\lceil \log_q n \rceil$ for the algebraic approach, this implies that the number of measurements required decreases as the field size increases. We demonstrate this relationship in Figure 1b, where we consider $q = 2^i, i = 1, 2, \ldots, 16$ with $n = 2048, 4096$ and $b = \lceil n^r \rceil$, $r = 0.2, 0.4, 0.6, 0.8$. One can observe that the number of measurements saturates to $2b$ for large enough q, the lower bound on the number of samples for differentiating between any two b-sparse vectors.

Tracking discrete-valued time series: The problem of the tracking time series is an important one and has been well-studied in literature [27, 28]. In many situations, the time-series is discrete-valued, for example, time-series data that corresponds to the backlog in queues, and the changes between two successive time instances have a sparse structure, such as sequence of video frames and time series from human motion recognition. As such, the concept of compressive sensing can be used - the idea is to compress the sparse changes so as to minimize the storage requirement.

As an example, consider a discrete-valued time series $(\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_t)$, where $\mathbf{z}_i \in \mathcal{A}^n$ and $\mathcal{A} \subset \mathbb{R}$ is the discrete alphabet, with the property $\mathrm{wt}(\mathbf{e}_i) \leq b$, $\mathbf{e}_i \triangleq \mathbf{z}_{i+1} - \mathbf{z}_i$, $i = 1, 2, \ldots, t-1$, and $b << n$. Then one approach is to use real-valued compressive sensing – consider a sensing matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$, satisfying some incoherence property and compress/track the discrete-valued time series as $(\mathbf{z}_1, \mathbf{A}\mathbf{e}_1, \ldots, \mathbf{A}\mathbf{e}_{t-1})$. The decompression algorithm comprises of recovering $\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_{t-1}$ using ℓ_1-minimization and getting the estimate of the discrete time series. Another approach is the algebraic one – interpret $\mathcal{A}$ as finite field $\mathbb{F}_q$ or its subset and perform compression using the methodology described in Section 3. Next, we provide simulation results for the tracking error of a synthetic quantized time series and a promotion data based discrete-valued time series.

We consider $n = 1024$, $t = 500$, $q = 256$ and sparsity levels $b = \lceil n^r \rceil$, $r = 0.2, 0.4, 0.6$. We choose $m = 2b\lceil \log_q n \rceil$, where $q = 256$. The details of constructing the synthetic real-valued time series and quantized time-series from it, sensing matrices for algebraic and real-valued approaches, and definitions of tracking error are given in [23]. Figure 1c shows the plot for tracking error vs. time index for different sparsity levels. Note that this increasing nature of tracking error for compressive sensing over reals is due to error propagation in the estimates of the time series; this includes both the quantization error as well as error in even determining the sparsity patterns of the changes in the time-series vector variable. Also, the tracking error reduces with increasing b, as then m becomes larger and closer to the optimal number of measurements required for real-valued compressive sensing for error-free recovery of the sparse changes in the time-series. In contrast, we have exact recovery for the sparse changes in the algebraic approach, so the tracking error at any time only comprises of the quantization error.

The promotional data time-series comes from [29]; we use the 'promotions.dat' file that contains a time-series with $n = 1000, t = 1000$ over three years. The entries in the time series come from $\{0, 1\}$. We perform the same manner of tracking as for the synthetic time series for all three years, the only difference being that the promotional time series is already discrete-valued, so there is no quantization error. We set $m = 2b\lceil \log_q n \rceil$, $q = 1024$. Then the tracking error is always zero for the algebraic approach. Figure 1d shows the tracking errors with increasing time index. Note that for compressive sensing over reals it increases with time due to error propagation. Also, the algebraic approach requires lesser storage space if a real is assigned $\geq \log_2 q = 10$ bits.

6. CONCLUSION

In this paper, we develop an algebraic framework for compressive sensing over finite alphabet; we provide constructive approaches for designing sensing matrices and polynomial-time-complexity algorithms for sparse source recovery, all while maintaining optimality in terms of sample complexity. Furthermore, we demonstrate that our approach outperforms real-valued compressive sensing in terms of sample complexity and storage. In terms of utility, compressive sensing over finite alphabet proves to be a natural fit for the purpose of compressing/tracking discrete-valued time-series data.

7. REFERENCES

[1] R. Baranuik, "Compressive sensing," *IEEE Signal Process. Magazine*, vol. 24, pp. 118–121, Jul 2007.

[2] F. Parvaresh, H. Vikalo, S. Misra, and B. Hassibi, "Recovering sparse signals using sparse measurement matrices in compressed DNA microarrays," *IEEE Sel. Topics in Signal Proc.*, vol. 2, 2008.

[3] J. Romberg, "Imaging via compressive sampling," *IEEE Signal Process. Magazine*, vol. 25, pp. 14–20, Mar 2008.

[4] M. Duarte, M. Davenport, D. Takhar, J. Laska, T. Sun, K. Kelly, and R. G. Baraniuk, "Single-pixel imaging via compressive sampling," *IEEE Signal Process. Magazine*, vol. 25, pp. 83–91, Mar 2008.

[5] E. Candes and T. Tao, "Decoding by linear programming," *IEEE Trans. on Information Theory*, vol. 51, pp. 4203–4215, Dec 2005.

[6] E. Candes and T. Tao, "Near optimal signal recovery from random projections: Universal encoding strategies?," *IEEE Trans. on Information Theory*, vol. 52, pp. 5406–5425, Dec 2006.

[7] D. Donoho, "Compressed sensing," *IEEE Trans. on Information Theory*, vol. 52, pp. 1289–1306, Apr 2006.

[8] R. Tibshirani, "Regression shrinkage and selection via the Lasso," *Royal Statistical Society*, vol. 58, pp. 267–288, 1996.

[9] S. Chen, D. Donoho, and M. Saunders, "Atomic decomposition by basis pursuit," *SIAM Journal of Scien. Comp.*, vol. 20, pp. 33–61, 1998.

[10] E. Candes and T. Tao, "The Dantzig selector: Statistical estimation when p is much larger than n," *Annals of Statistics*, vol. 35, pp. 2392–2404.

[11] S. Sarvotham, D. Baron, and R. Baraniuk, "Sudocodes - fast measurement and reconstruction of sparse signals," *IEEE ISIT*, Jul 2006.

[12] D. Donoho and Y. Tsaig, "Fast solution of l1-norm minimization problems when the solution may be sparse," *Stanford University Department of Statistics Technical Report*, 2006.

[13] J. Tropp and A. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Trans. on Information Theory*, vol. 53, pp. 4655–4666, Dec 2007.

[14] S. Draper and S. Malekpour, "Compressed sensing over finite fields," *IEEE ISIT)*, pp. 669–673, Jul 2009.

[15] F. Zhang and H. Pfister, "Compressed sensing and linear codes over real numbers," *IEEE ITA Workshop*, pp. 558–561, Jan 2008.

[16] W. Ryan and S. Lin, "Channel codes: Classical and modern," *Cambridge University Press*, 2009.

[17] I. Csiszar, "Linear codes for sources and source networks: error exponents, universal coding," *IEEE Trans. on Information Theory*, vol. 28, pp. 585–592, Jul 1982.

[18] W. Xu and B. Hassibi, "Efficient compressive sensing with deterministic guarantees using expander graphs," *IEEE ITW*, Sept 2007.

[19] S. Sarvotham, D. Baron, and R. Baranuik, "Compressed sensing reconstruction via belief propagation," *Rice ECE Dept. Tech. Report*, 2006.

[20] A. Dimakis and P. Vontobel, "LP decoding meets LP decoding: a connection between channel coding and compressed sensing," *IEEE Allerton*, 2009.

[21] D. Sejdinovic A. Muller and R. Piechocki, "Approximate message passing under finite alphabet constraints," *IEEE ICASSP*, 2012.

[22] G. Leus Z. Tian and V. Lottici, "Detection of sparse signals under finite-alphabet constraints," *IEEE ICASSP*, 2009.

[23] A. Das and S. Vishwanath, "On finite alphabet compressive sensing," http://uts.cc.utexas.edu/∼akdas/finitecs.pdf.

[24] T. Cover and J. Thomas, "Elements of information theory," *Wiley-Interscience 2nd Edition*, 2006.

[25] R. Roth, "Introduction to coding theory," *Cambridge Univ. Press*, 2006.

[26] I. Dumer, "Concatenated codes and their multilevel generalizations," *Handbook of Coding Theory*, 1998.

[27] J. Li H. Chen and P. Mohapatra, "RACE: Time series compression with rate adaptivity and error bound for sensor networks," *IEEE MASS*, 2004.

[28] R. Xu O. Concha and M. Piccardi, "Compressive sensing of time series for human action recognition," *ACM DICTA*, 2010.

[29] Causality Workbench Team, "PROMO: Simple causal effects in time series," Aug. 2008.