AN ANALOG SUB-LINEAR TIME SPARSE SIGNAL ACQUISITION FRAMEWORK BASED ON STRUCTURED MATRICES

Juhwan Yoo, Amin Khajehnejad, Babak Hassibi, and Azita Emami-Neyestanak

Department of Electrical Engineering, California Institute of Technology, Pasadena, CA 91125

ABSTRACT

Advances in compressed-sensing (CS) have sparked interest in designing information acquisition systems that process data at close to the *information* rate. Initial proposals for CS signal acquisition systems utilized random matrix ensembles in conjunction with convex relaxation based signal reconstruction algorithms. While providing universal performance bounds, random matrix based formulations present several practical problems due to: the difficulty in physically implementing key mathematical operations, and their dense representation. In this paper, we present a CS architecture which is based on a sub-linear time recovery algorithm (with minimum memory requirement) that exploits a novel structured matrix. This formulation allows the use of a reconstruction algorithm based on relatively simple computational primitives making it more amenable to implementation in a fully-integrated form. Theoretical recovery guarantees are discussed and a hypothetical physical CS decoder is described.

Index Terms— Compressed-Sensing, Structured-Matrices, Sub-linear Recovery Algorithms

1. INTRODUCTION

Compressed sensing (CS) [1, 2] is a relatively new signal processing technique which enables the recovery of sparse vectors from what was previously thought to be an insufficient quantity of information. The problem of CS is stated mathematically as follows: let k, N be positive integers such that $k << N$. There exists a matrix $\Phi \in \mathbb{R}^{M\times N}$, with $M = \mathcal{O}(k\log(N/k))$, such that (almost) every k-sparse vector $\mathbf{x} \in \mathbb{R}^{N\times 1}$ can be reconstructed from linear non-adaptive measurements $y = \Phi x$.

Although the assumption of sparsity seems prohibitively restrictive, many real-world signals can be well-approximated by sparse-ones. This observation is exploited in many (lossy) compression algorithms such as JPEG and MP3. A further benefit of sampling in this manner is that the samples are obtained at closer to the *information rate* of the signal and come in a compressed form.

Achieving sub-Nyquist rate signal acquisition is particularly appealing in light of the fact that current limitations on signal acquisition system bandwidth is limited by currently available digitizer performance [3,4]. This appeal is reflected in the copious attention these efforts have received in the literature. Research in this area has resulted in a number of CS Sampling (encoder) architectures, [5–8] as well as physical implementations [9, 10] (see [11] for more detail).

In contrast to the progress that has been made in developing CS encoders, to the best of the authors' knowledge, there appears to have been few if any attempts at realizing a hardware platform which implements a recovery algorithm for CS type samples (decoder). The relative paucity of effort at realizing a hardware CS decoder may stem from the fact that computations used in basis pursuit[1] are difficult to physically realize. For example, Fixed-Point Continuation (1.1), see [13]:

$$\begin{aligned} &\min_{\mathbf{x}\in\mathbb{R}^N} ||\mathbf{x}||_1 + \frac{\mu}{2} f(\mathbf{x}), \\ &\text{where } f(\mathbf{x}) = ||\Phi\mathbf{x} - \mathbf{y}||_2^2 \text{ and } \mu > 0 \end{aligned} \tag{1.1}$$

is the basis-pursuit method which uses the least complex functions. The two essential steps (during every iteration) involve performing a gradient descent based on $f(\mathbf{x})$[2] followed by a soft-thresholding operation $S_\nu(\cdot)$ to account for the ℓ_1 term.

$$\nabla f(\mathbf{x}) = \Phi^\intercal(\Phi\mathbf{x} - \mathbf{y}) \tag{1.2}$$

$$S_\nu(\cdot) := sgn(\cdot) \odot \max\{|\cdot| - \nu,\ 0\},\ \text{where } \nu > 0 \tag{1.3}$$

$$\text{with } (\mathbf{x}\odot\mathbf{y})_i = x_i y_i \text{ and } sgn(\cdot) \text{ is the signum function}$$

The computations in (1.2) and (1.3) requires: matrix multiplication and addition, as well as the implementation of several other exotic functions.

In this paper, we present the results of an ongoing effort to realize a hardware decoder. We introduce a sub-linear recovery algorithm that exploits a novel sampling matrix that is: deterministic, structured and highly scalable. The construction is based on labeling the ambient state-space (indexes of entries of the input vector $\mathbf{x}$) with binary sequences of length $n = \log_2 N$, and summing up entries of $\mathbf{x}$ that share the same values (up to a fixed length) for a subset of the binary digits of the labelling sequence. It is also notable, that the presented matrix is RIP-less but compatible with basis-pursuit methods [14]

2. DEFINITIONS AND NOTATIONS

The following definitions will be useful in describing the proposed measurement matrix:

Definition 1 *Let m,n and d be integers. A (n, d) summary is a pair $X = (S, \mathbf{c})$, where S is a subset of $\{1, 2, \cdots, n\}$ of size d, and $\mathbf{c}$ is a binary sequence of length d. A (m, n, d) summary*

[1] The standard recovery technique suggested by CS theory [12]

[2] The second term in (1.1) $\frac{\mu}{2} f(\mathbf{x})$ is used to to account for noise; the purpose of this term is to penalize any residual error.

codebook is a collection $\mathcal{C} = \{(S_i, \mathbf{c}_j) \mid 1 \leq i \leq m,\ 0 \leq j \leq 2^d - 1\}$ *of* (n, d) *summaries, where* S_i*'s are distinct subsets, and* $\mathbf{c}_j$ *is the length* d *binary representation of the integer* j*. If* $m = \binom{n}{d}$*,* $\mathcal{C}$ *is called the complete* (n, d) *summary codebook.*

We also set the following conventions. We say that a binary label $\mathbf{b}$ of length n "conforms" to a (n, d) summary $(S, \mathbf{c})$, or interchangeably that the summary $(S, \mathbf{c})$ "appears" in $\mathbf{b}$ iff $\mathbf{b}(S) = \mathbf{c}$. In addition, two (n, d) summaries $(S, \mathbf{c})$ and $(S', \mathbf{c}')$ are said to conform, if there is a binary label $\mathbf{b}$ of length n that conforms to both of them. We use the operators $\|$ and $\nparallel$ to denote conformity and its complement respectively. For a binary sequence $\mathbf{b}$, we denote its decimal numerical value by $\Delta(\mathbf{b})$. The following definition will also be useful later in the technical discussions.

Definition 2 *For a set* S *and a number* x*,* $\mathcal{I}(S, x)$ *indicates the number of elements of* S *that are less or equal than* x*. Consequently, if the elements of* S *are listed in increasing order as* $s_1 < s_2 < \cdots < s_d$*, then* $\mathcal{I}(S, s_i) = i\ \forall 1 \leq i \leq d$*.*

To a given (m, n, d) summary codebook $\mathcal{C}$ (Definition 1), we associate a binary matrix Φ of size $M \times N$ where $M = 2^d \times m$, and $N = 2^n$, in the following way. For every $(S, \mathbf{c}) \in \mathcal{C}$, there is a row $\phi = (\phi_1, \ldots, \phi_N)$ in Φ that satisfies:

$$\phi_j = \begin{cases} 1 & \mathbf{b}_j(S) = \mathbf{c} \\ 0 & \text{else} \end{cases} \qquad 1 \leq j \leq N, \tag{2.1}$$

where $\mathbf{b}_j$ is the n-bit binary representation of j, and $\mathbf{b}(S)$ is the subsequence of the binary sequence $\mathbf{b}$, indexed by the elements of the set S in increasing order. In other words, $\mathbf{a}$ has a 1 in the j'th coordinate, only if the binary representation of $j - 1$ conforms to $(S, \mathbf{c})$. Note that every column of A has exactly m ones, and each row has exactly 2^{n-d} ones.

To clarify the above construction protocol, we consider the following examples illustrated in Figure 1, with $n = 4$ and $d = 2$. Suppose that a summary $(S, \mathbf{c})$ is given with $S = \{1, 2\}$ and $\mathbf{c} = 10$. All possible binary sequences of length 4 that conform to $(S, \mathbf{c})$ are listed in Figure 1. If these sequences are converted to decimal values and increased by 1, they give the indices of the columns where there is a 1 in the considered row, namely $9, 10, 11$ and 12. The full row $\mathbf{a}$ of length 16 is displayed in Figure 1 as well.

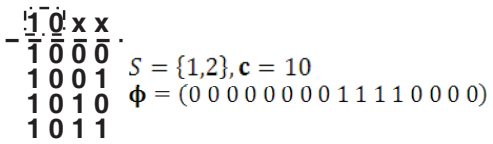

Fig. 1: An example $(4, 2)$ summary and the corresponding row of the structured measurement matrix.

An example binary 12×32 matrix is illustrated in Figure 2 which corresponds to a $(5, 2)$ summary codebook with 12 summaries. The rows of the displayed matrix are labeled with corresponding summaries and the columns are labeled by distinct binary sequences of length 5. Entries colored in black represent 1's, and white is 0.

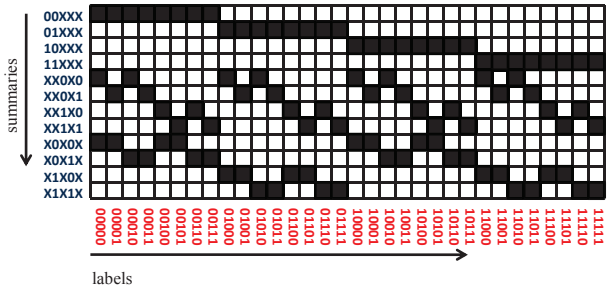

Fig. 2: An example of a measurement matrix constructed based on a $(5, 2)$ summary codebook. Black is 1 and white is 0.

2.1. Reconstruction Algorithm

The proposed sublinear time reconstruction algorithm is called Summarized Support Index Inference (SSII) [14, 15], as it attempts at finding (inferring) the labels of the support set indices of the unknown vector. The algorithm is based on iteratively inferring the nonzero entries of the signal based on one of the distinct values of $\mathbf{y}$ and its various occurrences. The detailed procedure is described in Algorithm 1. In every iteration, one support index label is identified. The main loop thus contains the following subroutines: 1) The *value classification* subroutine scans through the entries of $\mathbf{y}$ and groups together the nonzero coefficients with (almost) equal values, 2) The *coarse index identification* subroutine cycles through all the occurrences of a particular coefficient of the vector $\mathbf{y}$ grouped together by the value classification subroutine. For each group, the corresponding summaries are identified, and the subroutine attempts to identify a binary label that conforms to all of them. In the presence of noise, this subroutine settles for finding a label that conforms to as many summaries as possible. 3) In the case that the enforced label is not unique and contains a few undetermined bits, the *fine index identification* subroutine tries to estimate the remaining bits by a series of simple value counting rules. In the presence of noise, this turns into a set of hypothesis tests on particular subsets of coefficients of $\mathbf{y}$. 4) Finally, the rejection step verifies whether the estimated label is actually in the support set or not. If a certain statistical criteria for the measurements containing the estimated index is not satisfied, the label is rejected and the search continues. A block diagram describing these fundamental steps and there interconnections are depicted in Figure 3.

3. RECOVERY BOUNDS

We provide a theoretical guarantee for the performance of the proposed algorithm in the noiseless case under certain conditions on the sparse signal. Specifically, we consider the class of signals $\mathbf{x}$, for which no two disjoint subsets of the nonzero entries sum up to the exact same number. For simplicity we refer to the signals with this property as *distinguishable*. Note that for nonzero coefficients drawn randomly from a continuous distribution, this conditions hold with probability 1. The proof of the following theorem is given in the reference [14].

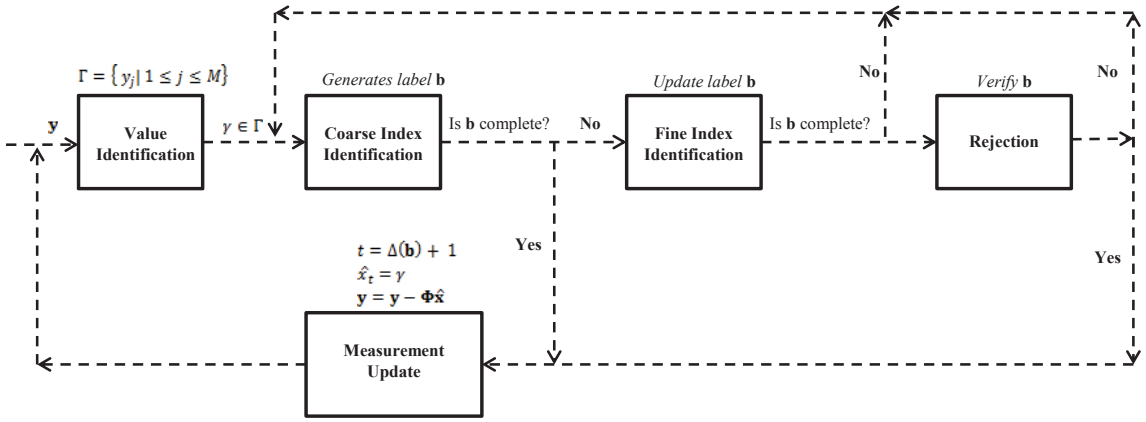

Fig. 3: Block diagram describing the subroutines of Algorithm 1.

Algorithm 1 Summarized Support Index Inference

Input: Vector $\mathbf{y} = (y_1, y_2, \ldots, y_M)^T$ and corresponding summaries $(S_j, \mathbf{c}_j),\ 1 \le j \le M$.
Output: Estimate $\hat{\mathbf{x}}$ of the sparse vector $\mathbf{x}$.
Initialize: Set $\hat{\mathbf{x}} := 0$.
Set $\mathcal{S} := \{S_j \,|1 \le j \le M\}$
Identify $\Gamma := \{y_j \,|1 \le j \le M \ y_i \ne 0\}$
for $\gamma \in \Gamma$ **do**
  Set $J := \{j | y_j = \gamma\}$
  if $\nexists j', j'' \in J$ s.t. $(S_{j'}, \mathbf{c}_{j'}) \nparallel (S_{j''}, \mathbf{c}_{j''})$ **then**
    Set $\mathbf{b} := -\mathbf{1}_{n\times 1}$.
    for $j \in J$ **do** $\mathbf{b}(S_j) := \mathbf{c}_j$ **end for**
  end if
  Set $S_1 := \{1 \le i \le n \mid \mathbf{b}(i) \ne -1\}$ and $S_2 := \{1, 2, \ldots, n\} \setminus S_1$.
  if $S_2 = \emptyset$ **then**
    Set $t := \Delta(\mathbf{b}) + 1$, $\hat{\mathbf{x}}_t := \gamma$, $\mathbf{y} := \mathbf{y} - \Phi\hat{\mathbf{x}}$, **goto** 5.
  else
    for $S \in P(S)$ and $l \in S_2 \cap S$ **do**
      $R_0 := \{y_j | S_j = S, \mathbf{b} \| (S_j, \mathbf{c}_j), \mathbf{c}_j(\mathcal{I}(S, l)) = 0\}$
      $R_1 := \{y_j | S_j = S, \mathbf{b} \| (S_j, \mathbf{c}_j), \mathbf{c}_j(\mathcal{I}(S, l)) = 1\}$
      if $R_0 = \{0\}$ **then** $\mathbf{b}(l) := 1$, **goto** 12. **end if**
      if $R_1 = \{0\}$ **then** $\mathbf{b}(l) := 0$, **goto** 12. **end if**
    end for
  end if
end for

Theorem 1 *Let Φ be a measurement matrix that corresponds to a random (m, n, d) summary codebook, and let $0 < \lambda < 1$ and $\alpha > 0$ be constants. Then, for $k = \lambda 2^{-d\log_2(\sqrt{\alpha/2}+1/2)}$ a random k-sparse distinguishable vector $\mathbf{x}$ can be recovered by Algorithm 1 with probability at least $k^3 n e^{-\alpha n} + kn\left(1 - (1-\lambda)d/n\right)^m$.*

Substituting the parameters d, m, λ and α with appropriate values in terms of M, k and N we can conclude from the above theorem that the required number of measurements for almost surely reconstructing k-sparse signals is:

$$M = (k \log N \log\log N) \tag{3.1}$$

4. HYPOTHETICAL HARDWARE IMPLEMENTATION

In current analog signal processing architectures which implement CS-based sampling, correlation between the input analog signal and a row of the sampling matrix Φ is accomplished by multiplying the input with the output of an LFSR. Since the primary advantage of the proposed sparse signal acquisition framework is in the decoder architecture, we simply state that an encoder architecture compatible with our proposed matrix Φ could be realized with the use of simple binary counters and comparison logic. Consequently, one possible advantage of this approach is that unlike an LFSR approach, only the logic connected to the least significant bit of the binary counter would need to operate at the maximum speed of the system.

A hypothetical decoder architecture is shown in Figure 3. The required operations are: 1) Sorting in the value identification block, 2) The coarse and fine index identification steps which are primarily based on a majority-selection operation, 3) a verification step which involves a matrix-vector multiply compactly and efficiently implementable due to the fact that each row of Φ has a concise encoding.

The most complex computation of our decoding architecture is contained in the value identification subroutine: thresholding, sorting, and pair-wise comparison of the entries of the observation vector $\mathbf{y}$. However, the complexity of these operations as a function of the number of observations M is unlikely to become a performance limiting factor when compared to the pragmatic issues in implementing an encoder of corresponding size. In addition, due to the fact that our matrix has a highly structured and concise description, memory storage requirements scale well with size and compares favorably with other classes of measurement matrices. We emphasize that even when compared to its other sub-linear counterparts (e.g. [16–19], our algorithm has minimum storage requirement and relies on very rudimentary signal-processing techniques that do not rely on higher-order interaction of elements of $\mathbf{y}$ and Φ.

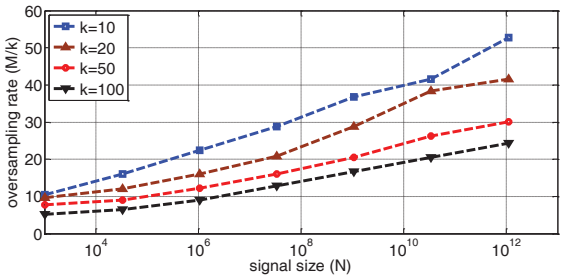

Fig. 4: Required oversampling rate for successful recovery of Algorithm 1 on proposed constructions versus signal dimension for various sparsity levels.

5. SIMULATIONS

The empirical performance of Algorithm 1 in the absence of noise is shown in Figure 4. For a comprehensive study of the algorithm including robustness to noise, we refer the reader [14]. Due to the efficiency of the method, it is possible to perform simulations for very large values of N. In Figure 4, the empirical required over-sampling rate for Algorithm 1 is plotted against the signal dimension N, for various sparsity levels k. The required criteria for the sufficient number of measurements here is the probability of successful recovery being larger than 90%. Note that when N is increased by 3 orders of magnitude, the required number of measurements increases by a factor of 3, which is an indication of the logarithmic dependence of M on N. Furthermore, as the signal becomes less sparse (i.e. k increases), the required oversampling factor decreases. For $k = 100$, this ratio is only about 3 for $N = 1024$, and about 8 for $N = 3.3 \times 10^7$. This is significantly better than existing sublinear recovery algorithms.

6. DISCUSSION

We have analyzed the SSRI algorithm with respect to many different practical performance metrics which took into consideration various physical implementation issues. Primary considerations in the design were: scalability of the ambient dimension and sparsity, performance bounds, recovery time, robustness to noise, and amenability to hardware implementation of the decoder: with the end-goal being real-time signal recovery. The theoretical aspects of the SSRI algorithm discussed in [14] along with a detailed comparison to other state-of-the-art sublinear algorithms. We emphasize there is no single sublinear recovery algorithm which delivers comparable performance in all the metrics considered.

While the SSRI algorithm is composed of simpler mathematical operations that make it more amenable to physical implementation, it has the same drawbacks typically accompanied by other sublinear algorithms reported in the literature. The most notable drawback being that the ability to reconstruct signals, at present, from a wide variety of structured dictionaries comes at the price of extensive memory requirements. However this is not a definitive restriction and is the subject of ongoing investigation.

7. ACKNOWLEDGEMENTS

The authors would like to acknowledge Mathew Till and Dr. Stephen Becker for insightful technical disucssions.

8. REFERENCES

[1] E. J. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. Inform. Theory*, vol. 52, no. 2, pp. 489–509, 2006.

[2] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inform. Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[3] B. Murmann, "Trends in low-power, digitally assisted A/D conversion," *IEICE Trans. Electron.*, vol. E93-C, no. 6, pp. 718–727, 2010.

[4] H. Walden, "Analog-to-digital converter survey and analysis," *IEEE J. Sel. Area Comm.*, vol. 17, no. 4, pp. 539–550, 1999.

[5] J. N. Laska, S. Kirolos, M. F. Duarte, T. Ragheb, R. G. Baraniuk, and Y. Massoud, "Theory and implementation of an analog-to-information converter using random demodulation," in *Proc. IEEE ISCAS, New Orleans*, May 2007, pp. 1959–1962.

[6] S. Kirolos, J. N. Laska, M. Wakin, M. F. Duarte, D. Baron, T. Ragheb, Y. Massoud, and R. G. Baraniuk, "Analog-to-information conversion via random demodulation," in *Proc. IEEE DCAS*, Sept. 2006.

[7] J. A. Tropp, J. N. Laska, M. F. Duarte, J. Romberg, and R. G. Baraniuk, "Beyond Nyquist: Efficient sampling of sparse bandlimited signals," *IEEE Trans. Inform. Theory*, vol. 56, no. 1, pp. 520–544, 2010.

[8] J. Romberg, "Compressive sensing by random convolution," *SIAM J. Imag. Sci.*, vol. 2, no. 4, pp. 1098–1128, 2009.

[9] J. Yoo, S. Becker, M. Loh, M. Monge, E. Candès, and A. Emami-Neyestanak, "A 100MHz-2GHz 12.5x sub-Nyquist rate compressed-sensing signal acquisition system," ISSCC, *submitted*, 2012.

[10] M. Mishali and Y. C. Eldar, "From theory to practice: Sub-Nyquist sampling of sparse wideband analog signals," *IEEE J. Sel. Top. Signal Process.*, vol. 4, no. 2, pp. 375–391, 2010.

[11] S. Becker, *Practical Compressed Sensing: modern data acquisition and signal processing*, Ph.D. thesis, California Institute of Technology, 2011.

[12] E. J. Candès and T. Tao, "Decoding by linear programming," *IEEE Trans. Inform. Theory*, vol. 51, no. 12, pp. 4203–4215, December 2005.

[13] E. Hale, W. Yin, and Y. Zhang, "Fixed-point continuation for ℓ_1-minimization:methodology and convergence," *SIAM J. Optim*, vol. 19, no. 3, pp. 1107–1130, 2008.

[14] A. Khajehnejad, Juhwan Yoo, and Babak Hassibi, "Practical sublinear algorithms for compressed sensing," *Priprint*, 2011.

[15] A. Khajehnejad, Juhwan Yoo, , and Animashree Anandkumar Babak Hassibi, "Summary based structrues with improved sublinear recovery for compressed sensing," *ISIT*, 2011.

[16] S. Sarvotham, D. Baron, and R. Baraniuk, "Fast measurement and reconstruction of sparse signals," in *IEEE Int. Symp. Info. Theory*, Seattle, July 2006.

[17] A. C. Gilbert, M. J. Strauss, J. A. Tropp, and R. Vershynin, "Algorithmic linear dimension reduction in the l1 norm for sparse vectors," in *Proc. 44th Allerton Conf. Communication, Control, and Computing*, Allerton, IL, 2006, Invited paper, special session: Compressed Sensing.

[18] O. Milenkovic W. Dai and H.Pham, "Structured sublinear compressive sensing via dense belief propagation," Preprint, arXiv, 2011.

[19] S. D. Howard, A. R. Calderbank, and S. J. Searle, "A fast reconstruction algorithm for deterministic compressive sensing using second order reed-muller codes," in *Proc. Conf. Inform. Sciences & Systems*, March 2008, pp. 11–15.