REUSABLE LOW-ERROR COMPRESSIVE SAMPLING SCHEMES THROUGH PRIVACY

*Anna C. Gilbert**, *Brett Hemenway, Martin J. Strauss, David P. Woodruff, Mary Wootters*

University of Michigan
Department of Mathematics
Ann Arbor, MI 48109
except for DPW,
IBM Almaden Labs
San José, CA

ABSTRACT

A compressive sampling algorithm recovers approximately a nearly sparse vector x from a much smaller "sketch" given by the matrix vector product Φx. Different settings in the literature make different assumptions to meet strong requirements on the accuracy of the recovered signal. Some are robust to noise (that is, the signal may be far from sparse), but the matrix Φ is only guaranteed to work on a single fixed x with high probability—it may not be re-used arbitrarily many times. Others require Φ to work on all x simultaneously, but are much less resilient to noise.

In this note, we examine the case of compressive sampling of a RADAR signal. Through a combination of mathematical theory and assumptions appropriate to our scenario, we show how a single matrix Φ can be used repeatedly on multiple input vectors x, and still give the best possible resilience to noise.

Index Terms— Privacy preserving, compressive sampling, Forall/Foreach

1. INTRODUCTION

The goal of compressive sampling is to design an $m \times N$ matrix Φ (or distribution on matrices) with $m \ll N$ so that one can efficiently recover a nearly k-sparse vector $x \in \mathbb{R}^N$ from the much smaller "sketch" Φx of x. The strongest guarantee is that the recovered vector $\widetilde{x}$ satisfy

$$\|\widetilde{x} - x\|_2 \leq (1+\epsilon)\|x_k - x\|_2, \tag{1}$$

where x_k is the largest k terms of x (in magnitude). Equation (1) is not always achievable with nontrivial m, and a variety of assumptions have been made to obtain it. We give two extremal examples. Fix a length N, a sparsity level k, and a distortion parameter ϵ.

Definition 1 (Forall/Malicious) *A compressive sampling algorithm in the* Forall *model consists of a matrix Φ a recovery algorithm $\mathcal{R}$, and a constant C such that, for any x with*

$$\|x_k - x\|_1 \leq C\sqrt{k}\|x_k - x\|_2, \tag{2}$$

the recovered vector $\widetilde{x} = \mathcal{R}(\Phi x)$ satisfies Equation (1)*.*

Algorithms in the Forall model were given by Donoho and by Candès, Romberg, Tao [1, 2] (among many others) with an optimal number of rows and recovery time polynomial in N, and by Gilbert, Strauss, Tropp, and Vershynin [3] with a slightly suboptimal number of rows but sublinear recovery time.

Definition 2 (Foreach/Oblivious) *A compressive sampling algorithm in the* Foreach *model consists of a distribution $\mathcal{D}_\Phi$ on matrices Φ and a recovery algorithm $\mathcal{R}$, such that, for any x, with high probability over $\Phi \sim \mathcal{D}_\Phi$, the recovered vector $\widetilde{x} = \mathcal{R}(\Phi, \Phi x)$ satisfies Equation* (1)*.*

There are many algorithms in the Foreach setting, including the algorithm of Hassanieh, et al. [4] which samples a slightly suboptimal number of positions of a spectrally sparse signal, and, in nearly optimal time, returns a strong approximation of the spectrum. This algorithm

*MJS and MW have been supported in part by NSF CCF 0743372 (CAREER) and MJS by DARPA ONR N66001-06-1-2011.

is a direct descendant of that of Mansour [5], which we will use as a representative algorithm in our result.

Both of these models make assumptions. The Forall model imposes a geometric condition on the tail of x, $x_k - x$. In the Foreach model, we assume that x is generated obliviously of Φ, and Φ cannot be "re-used" on a different x. Such assumptions are necessary. If the tail is allowed to be larger and Φ must work on all vectors simultaneously, then the number of measurements m must be $\Omega(N)$ [6], *i.e.*, there is no non-trivial scheme.

1.1. Noise resilience

The Foreach model is much more resilient to noise than the Forall model. With no restrictions on the tail, the Forall model guarantees only

$$\|\mathcal{R}(\Phi x) - x\|_2 \leq \frac{\epsilon}{\sqrt{k}}\|x_k - x\|_1, \tag{3}$$

which is generally much weaker than Equation (1) and corresponds to much less resilience to noise.

As an example, suppose $k \ll N$ and the largest k elements of x (the head) each have magnitude $1/k$ and the other $N - k \approx N$ elements of x (the tail) have the same magnitude, a noise floor a, to be determined. The Forall guarantee (3) becomes vacuous (*i.e.*, zero is an acceptable approximation to x) when $a \geq 1/N$. By contrast, the Foreach guarantee (1) becomes vacuous only when $a \geq 1/\sqrt{kN} \gg 1/N$.

1.2. Models of adversarial interaction

On the other hand, the Forall model assumes less about the interaction between parties that are constructing Φ and x. In the Forall model, a challenger Charlie presents a matrix Φ. An adversary Mallory may look at Φ and take unbounded time in constructing an input x satisfying Equation (2), and the recovery algorithm must still succeed. By contrast, in the Foreach model, Mallory is only allowed to see the distribution $\mathcal{D}_\Phi$ (*e.g.*, the word "Gaussian"), not the outcome Φ itself, when constructing x. In general, if Mallory sees Φ, she can find x (not satisfying Equation (2)) that breaks the algorithm unless the number of rows is nearly N. For example, Mallory may try to find an x whose tail transpose lies too close to many rows of Φ—in this case, Φ would blow up the noise, which may be problematic for a recovery algorithm.

Thus there are two ways in the current literature to ensure that Mallory does not find a signal x that defeats Charlie—ensure that Mallory does not see Φ, so that x cannot exploit weaknesses in Φ at all, or require that the tail of x be very small so that these weaknesses do not exist. However, there is a middle ground, if Mallory has partial information about Φ. There is more than one way for Mallory to get such information. In particular, after Charlie recovers an approximation $\widetilde{x}$ of x, Charlie may act on that information. In some situations, Charlie's action α can be observed by Mallory. In that way, Mallory gets a limited view of Φ, since α depends on $\widetilde{x}$ which depends on Φ, even conditioned on x. By analogy with differential power attacks and differential timing attacks in cryptography[1] , Mallory can use this information to recover Φ and/or to construct a new x' that breaks the algorithm; *i.e.*, x' is not recovered properly.

Consider a missile launched by Mallory at a target controlled by Charlie. The missile illuminates the target with RADAR. The target detects the RADAR using a compressive sampling algorithm with matrix Φ and takes appropriate defensive action, neutralizing the missile. The action is observed by the enemy, who sends a new missile with a new RADAR signal. The goal is to guarantee that the defensive player can reuse the same Φ and get the good noise resilience guarantee of Equation (1), even though the assumptions of the Foreach case no longer hold. We assume that the RADAR signal is a signal with sparse Fourier transform plus tail noise, appropriate for compressive sampling to detect. [2]

In this note, we give a formal definition for the *private* model of compressive sampling and we present the formal analysis of an algorithm (and assumptions) that support reusing a single random Φ with good noise resilience. While we do not address the efficiency of such algorithms directly in this note, we are motivated by the need for efficient algorithms and our comments apply to algorithms with runtimes polynomial in $k \log N$, as well as those algorithms that require time polynomial in N.

[1] In differential power or timing attacks, an adversary without a secret key observes the amount of time or electrical power used by a challenger in possession of the secret key to decrypt a secret message. Unless obfuscatory steps are taken, processing a 0 in a secret key can take less time or power than processing a 1. The adversary thereby gets some information about the secret key.

[2] Many of our comments below also hold when there is measurement noise, *i.e.*, the compressive receiver sees $\Phi x + \nu$ rather than Φx.

2. PRIVATE COMPRESSIVE SAMPLING

In this section, we give a formal definition of a private compressive sampling scheme. We also illustrate how existing schemes fail to retain private information (or leak information) so that a malicious adversary Mallory could learn enough information about the scheme through a sequence signals $x^{(i)}$ to foil the algorithm.

Definition 3 (Private) *A compressive sampling algorithm in the* private *model consists of a distribution $\mathcal{D}_\Phi$ on matrices Φ and a (possibly randomized) recovery algorithm $\mathcal{R}$ with state S (*e.g.*, the seed to generate Φ), such that, for any sequence $x^{(i)}$ in which $x^{(i)}$ may depend on the recoveries $\mathcal{R}\left(\Phi x^{(<i)}, S\right)$, with high probability over $\Phi \sim \mathcal{D}_\Phi$ and the coins of $\mathcal{R}$, we have*

$$\|\mathcal{R}(\Phi x^{(i)}, S) - x^{(i)}\|_2 \leq (1+\epsilon)\|x_k^{(i)} - x^{(i)}\|_2.$$

This definition is an extension of definitions in [7] and [8] to the dynamic setting, in which multiple queries are made based on previous responses.

The following two examples that illustrate how *components* of existing compressive sampling schemes are *not* private: (1) estimation of the Fourier coefficient $\widehat{x}(\omega)$ and (2) estimation of the power $\|x\|_2^2$. Charlie, the challenger, generates a single random vector r from a distribution $\mathcal{D}$ and then, upon receipt of a signal x designed by Mallory, he estimates

$$\widehat{x}(\omega) = \langle \mathrm{e}^{2\pi i\omega t}, r\rangle\langle r, x\rangle \quad \text{and} \quad \|x\|^2 = |\langle x, r\rangle\langle r, x\rangle|.$$

Mallory knows the distribution but not the specific vector r, and she sees Charlie's estimates. Mallory will generate a sequence of signals $x^{(i)}$ that are designed to leak information about the random vector r. Armed with enough information about r, she can generate a signal x' that Charlie will not be able to recover from a compressive sampling scheme. For example, Mallory sets $x^{(i)} = \delta_i$, the i canonical basis vector and learns $r_i = \langle r, x\rangle$ in the first (Fourier coefficient) estimate. For the second (power) estimate, she sets $x^{(3i)} = \delta_i$, $x^{(3i+1)} = \delta_i + \delta_{i+1}$, and $x^{(3i+2)} = \delta_i - \delta_{i+1}$ and learns r_i from an algebraic manipulation of the measurements.

Note, however, that although $\langle r, x\rangle$ can leak information about r, if Charlie uses $\langle r, x\rangle$ only in $\langle y, r\rangle\langle r, x\rangle$, then an overall sign or phase for the vector r disappears from the final result. Thus the overall result may be private even if intermediate results are leaky, but additional argument is needed.

3. MAIN RESULT

We assume that, through low-level hardware or otherwise, the receiver Charlie can approximate the total power $\|x\|_2^2$ in a way that depends only on x. That is, the receiver outputs a random variable X such that $X = (1 \pm \epsilon)\|x\|_2^2$ and that X comes from a distribution that is (computationally indistinguishable from a distribution) parametrized by x. We say that the power has been estimated privately. We also assume that Charlie can approximate specific Fourier coefficients of x quickly and privately; *i.e.*, in a way that depends only on x. We suppose that this can be done using a hardware matched filter.

Our goal is a private compressive sampling algorithm with a small number of rows, makes a small number of private Fourier coefficient estimates, runs in sublinear time, and returns a result satisfying Equation (1). We use the following result, which follows from [5].

Theorem 4 *Fix k, N as above and fix $\tau = c/k$ for some constant c. There is a distribution $\mathcal{D}_\Phi$ on matrices Φ and an intermediate algorithm $\mathcal{S}$ returning $S_\tau = \mathcal{S}(\Phi, \Phi x)$, that runs in time $k\log^{O(1)} N$ such that, for any x, with high probability over $\Phi \sim \mathcal{D}_\Phi$, we have, if a_ω is the ω'th Fourier coefficient of x,*

- *If $|a_\omega|^2 \geq \tau\|x\|_2^2$, then $\omega \in S_\tau$.*
- *$|S_\tau| = O(1/\tau)$.*

This gives a private compressive sampling scheme [7]:

Theorem 5 *Under the assumptions above, there is a compressive sampling algorithm in the private model.*

Proof. The idea is as follows. In the i'th iteration, write x instead of $x^{(i)}$ for simplicity.

1. By assumption, estimate the power $P = \|x\|_2^2$ privately, as $\widetilde{P}$.
2. Use a compressive sampling algorithm such as [5] to generate a candidate set S of frequencies that contains the frequencies with large amplitudes.
3. By assumption, privately estimate the amplitudes of the frequencies in S.
4. Compare each such amplitude squared $|a|^2$ with $\widetilde{P}/k$ and keep only the terms with $|a|^2 \geq \widetilde{P}/k$.

The correctness and efficiency of each step above follows either by assumption or from [5]. We now analyze the privacy. The algorithm [5] can produce a set S_τ containing all terms with amplitude at least τ (the τ-heavy terms) and none with amplitude less than $\tau/2$ (the $\tau/2$-light terms), with appropriate cost provided $\tau^2 \approx 1/k$. The trouble is that terms with amplitude in the range $[\tau/2, \tau]$ may or may not be in S_τ and their presence or absence leaks information to Mallory.

In the above algorithm, we can set $\tau = \sqrt{\widetilde{P}/k}$. Then the algorithm of [5] will return all terms with amplitude magnitude at least $\sqrt{\widetilde{P}/k}$. We then privately compute the amplitudes of these terms.

Mallory will see an action based on $\widetilde{x}$. We assume Mallory sees all of $\widetilde{x}$. To show that Mallory learns nothing about Φ, we now claim that Mallory effectively knew $\widetilde{x}$ in distribution already, without access to Φ. (This is known as a simulation proof in cryptography.)

Mallory knows x, so she can compute $\|x\|_2^2$ and learn $\widetilde{P}$. She also knows all the amplitudes in x, so she can replicate Charlie's exact computation of amplitudes. Any term Charlie compares with $\sqrt{\widetilde{P}/k}$ will be compared in the same way by Mallory, independent of Φ. If Charlie were to test all N terms with his hardware system alone, Mallory would not learn anything about Φ. (Indeed, Φ would not even be used.)

But Charlie will use Φ to cut down the list of candidates to S_τ, and the set S_τ depends on Φ in a non-private way. Since S_τ is guaranteed to contain all large terms and the comparison with $\sqrt{P/k}$ will be correct on *all* terms in S_τ, Mallory ends up with the same set of frequencies and amplitudes as Charlie. ∎

Note that Charlie is deliberately *discarding* useful information about $x^{(1)}$, namely, fairly reliable recovery of the medium-sized terms. This is the right approach, intuitively, because Charlie's acting on this information about $x^{(1)}$ gives information to Mallory.

4. CONCLUSION

In this note, we have defined a private model for compressive sampling, which allows for the best of both worlds. In the private model, the matrix Φ may be reused on an arbitrary number of input vectors x, and yet the guarantee of Equation (1) still applies. We present an efficient recovery algorithm for private compressive sampling. There's no free lunch, and we must make assumptions as well, but we hope that these assumptions are reasonable (more reasonable than those in the Forall/Foreach models) in many practical situations, as illustrated by our RADAR example.

5. REFERENCES

[1] D Donoho, "Compressed sensing," *Information Theory*, vol. 52, no. 4, pp. 1289–1306, Jan 2006.

[2] Candes, Romberg, and Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Transactions on Information Theory*, vol. 52, no. 2, 2006.

[3] A.C. Gilbert, M.J. Strauss, J.A. Tropp, and R. Vershynin, "One sketch for all: fast algorithms for compressed sensing," in *Proceedings of the thirty-ninth annual ACM symposium on Theory of computing*. ACM, 2007, pp. 237–246.

[4] Haitham Hassanieh, Piotr Indyk, Dina Katabi, and Eric Price, "Nearly optimal sparse Fourier transform," in *Proceedings of 44'th ACM Symposium on Theory of Computing*, 2012, To appear.

[5] Yishay Mansour, "Randomized interpolation and approximation of sparse polynomials," in *ICALP*, 1992, pp. 261–272.

[6] A Cohen, W Dahmen, and R DeVore, "Compressed sensing and best k-term approximation," *American Mathematical Society*, vol. 22, no. 1, pp. 211–231, 2009.

[7] Joe Kilian, André Madeira, Martin J. Strauss, and Xuan Zheng, "Fast private norm estimation and heavy hitters," in *TCC*, 2008, pp. 176–193.

[8] Joan Feigenbaum, Yuval Ishai, Tal Malkin, Kobbi Nissim, Martin J. Strauss, and Rebecca N. Wright, "Secure multiparty computation of approximations," *ACM Transactions on Algorithms*, vol. 2, no. 3, pp. 435–472, 2006.