

# RANDOM FILTERS FOR COMPRESSIVE SAMPLING AND RECONSTRUCTION

Joel A. Tropp,<sup>m</sup> Michael B. Wakin,<sup>r</sup> Marco F. Duarte,<sup>r</sup> Dror Baron,<sup>r</sup> and Richard G. Baraniuk<sup>r</sup>

<sup>m</sup> Department of Mathematics, The University of Michigan at Ann Arbor

<sup>r</sup> Department of Electrical and Computer Engineering, Rice University

## ABSTRACT

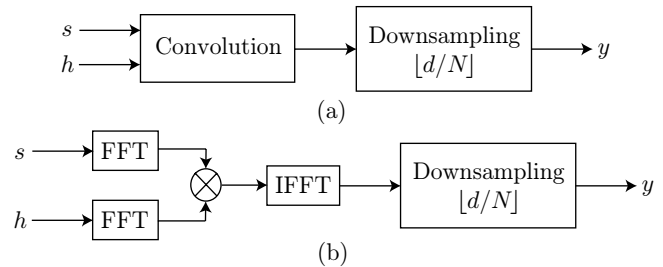
We propose and study a new technique for efficiently acquiring and reconstructing signals based on convolution with a fixed FIR filter having random taps. The method is designed for sparse and compressible signals, i.e., ones that are well approximated by a short linear combination of vectors from an orthonormal basis. Signal reconstruction involves a non-linear Orthogonal Matching Pursuit algorithm that we implement efficiently by exploiting the nonadaptive, time-invariant structure of the measurement process. While simpler and more efficient than other random acquisition techniques like Compressed Sensing, random filtering is sufficiently generic to summarize many types of compressible signals and generalizes to streaming and continuous-time signals. Extensive numerical experiments demonstrate its efficacy for acquiring and reconstructing signals sparse in the time, frequency, and wavelet domains, as well as piecewise smooth signals and Poisson processes.

## 1. INTRODUCTION

Natural signals often contain some type of structure that makes them *compressible*. That is, a compressible signal of length  $d$  can be well approximated using  $m$  real numbers, with  $m \ll d$ . Natural images, for example, are compressed by a factor of 10 or more when expressed in terms of their largest wavelet coefficients. The usual approach to acquiring a compressible (digital) signal is to take measurements in the Dirac basis and then use a nonlinear algorithm, such as a JPEG coder, to obtain a more efficient approximation. But this approach is not practicable if the signal is presented at a high rate or if the measurement device has limited computational resources (as in a sensor network).

As a result, Candès, Romberg and Tao [1] and Donoho [2] have proposed an approach, known as *Compressed Sensing* (CS), in which a *random* linear projection is used to acquire efficient representations of compressible signals directly. This method is effective for *sparse* signals, i.e., signals that are well approximated by a short linear combination of vectors

JAT was supported by NSF DMS grant 0503299. MBW, MFD, DB, and RGB were supported by NSF-CCF, NSF-NeTS, ONR, and AFOSR. E-mail: jtropp@umich.edu, {wakin, duarte, drorb, richb}@rice.edu. Web: dsp.rice.edu/cs.



**Fig. 1.** Block diagrams for signal acquisition through random filtering: (a) using convolution; (b) using FFT/IFFT. The FIR filter  $h$  has random taps, which must be known in order to recover the signal  $s$  from the compressed data  $y$ .

from an orthonormal basis. Signals are reconstructed by solving a costly linear program [1, 2] or using a more efficient greedy pursuit [3]. At present, CS is not suited to real-time applications or large data sets because the measurement process requires access to the entire signal at once; the measurements are not time-invariant; and the reconstruction algorithms are very expensive.

In this work, we propose *random filters* as a new paradigm for compressive signal acquisition. Our approach captures a signal  $s$  by convolving it with a random-tap FIR filter  $h$  and then downsampling the filtered signal to obtain a compressed representation  $y$ . Figure 1 illustrates the measurement process. Reconstruction of  $s$  involves a nonlinear algorithm.

At first glance, one might think this method would convert a signal into garbage. In fact, the random filter is *generic* enough to summarize many types of compressible signals. At the same time, the random filter has enough *structure* to accelerate measurement and reconstruction algorithms. Our method has several benefits:

- measurements are time-invariant and nonadaptive;
- measurement operator is stored and applied efficiently;
- we can trade longer filters for fewer measurements;
- it is easily implementable in software or hardware; and
- it generalizes to streaming or continuous-time signals.

This paper describes how random filters can be used to capture finite-length, discrete-time signals. In Section 3, we discuss two different methods for signal acquisition that take full advantage of the structure of the random filter. In Section 4, we present a reconstruction algorithm, based on Orthogonal Matching Pursuit (OMP), that uses the structure of

the random filter to accelerate computations. Section 5 reports on extensive numerical experiments, which confirm that random filters are effective tools for signal acquisition and recovery, and Section 6 describes conclusions and future work.

## 2. BACKGROUND

This work concentrates on signals that can be expressed efficiently in an orthonormal basis  $\Psi$ , called the *sparsity basis*. We say that a signal  $s$  of length  $d$  is *m-sparse* with respect to the basis if there is a vector  $\theta$  with  $m \ll d$  nonzero entries for which  $s \approx \Psi\theta$ . This model covers many signal classes, including (i) Poisson processes and spike trains, (ii) piecewise polynomials and splines, and (iii) signals in weak  $\ell_p$  balls.

Results from the CS literature provide a benchmark for studying the performance of random filters. Taking  $N$  linear measurements of the signal  $s$  can be viewed as multiplication  $y = \Phi s$  by a  $N \times d$  measurement matrix  $\Phi$ . Several distributions for  $\Phi$  have been proposed. If  $\Phi$  is fully i.i.d. Gaussian, then several different algorithms can recover *m-sparse* signals from  $N = O(m \log d)$  measurements. If  $\Phi$  is a Rademacher (i.e.,  $\pm 1$ ) matrix or a random row submatrix of the DFT matrix, then a similar number of measurements suffice in practice.

The costs for CS encoding and decoding depend significantly on the type of measurement matrix. Gaussian and Rademacher matrices require storage and computation  $O(dN)$  for encoding. Fourier measurement matrices improve storage to  $O(d)$  and encoding times to  $O(d \log d)$ . Two different algorithms,  $\ell_1$  minimization [1, 2] and OMP [3], are commonly used for signal reconstruction. The  $\ell_1$  minimization approach uses linear programming to solve the problem  $\hat{\theta} = \arg \min_{\theta} \|\theta\|_1$  subject to  $\Phi\Psi\theta = y$ . Reconstruction costs via  $\ell_1$  minimization have not been reported, but one expects them to be  $O(d^{3.5})$  in general. Greedy pursuit methods such as OMP attempt to build up an approximation to  $\theta$  based on correlations between  $y$  and the columns of the matrix  $\Phi\Psi$ . OMP requires  $O(mNd)$  arithmetic operations in general, but it can be improved to  $O(md \log d)$  with Fourier measurements.

## 3. COMPRESSIVE SAMPLING

This section defines more precisely what we mean by a random filter, and it compares two different methods for determining the compressed signal. A *random filter* of length  $B$  is an FIR filter whose  $B$  taps are i.i.d. random variables. In particular, we are interested in the cases where the taps are drawn (i) from the  $\mathcal{N}(0, 1)$  distribution or (ii) from the Bernoulli/Rademacher distribution of  $\{\pm 1\}'s$ .

### 3.1. Encoding methods

Draw a random filter  $h$  of length  $B$ . Note that the filter requires just  $O(B)$  storage. To take  $N$  measurements of a signal

$s$  of length  $d$ , we must calculate

$$y = \mathcal{D}_{\downarrow}(h * s),$$

where  $\mathcal{D}_{\downarrow}$  downsamples by a factor of  $\lfloor d/N \rfloor$ . Note that, because this process is linear, the map from the signal  $s$  to the summary  $y$  can be viewed as  $y = \Phi s$ , where  $\Phi$  is an  $N \times d$  matrix. This matrix is *banded* and *quasi-Toeplitz*: each row has  $B$  nonzero entries, and each row of  $\Phi$  is a copy of the row above, shifted right by  $\lfloor d/N \rfloor$  places.

**Method 1:** The first method for calculating the measurements, illustrated in Figure 1(a), performs linear convolution and downsampling simultaneously. For  $n = 0, 1, \dots, N - 1$ , the  $n$ -th measurement is calculated as

$$y(n) = \sum_{j=0}^{B-1} s(n \lfloor d/N \rfloor + j) h(B - j).$$

Computing  $N$  measurements requires  $O(BN)$  arithmetic operations. This method can be applied in systems where the input  $s$  is streaming, since the measurements are localized in time and also time-invariant.

**Method 2:** The second method, illustrated in Figure 1(b), uses FFTs to calculate the convolution. In this case, we compute

$$y = \mathcal{D}_{\downarrow} \mathcal{F}^{-1} \{H(\omega) S(\omega)\},$$

which is equivalent to using a circulant  $\Phi$  matrix. The cost of computing the measurements is  $O(d \log d)$ , independent of the filter length or the number of measurements. Compared to Method 1, this calculation may be faster if the filter has many taps. Note, however, that the entire signal must be presented at once.

It appears that these two encoding methods are at least as efficient as anything described in the CS literature. We also note that filtering can be performed with other standard methods, such as overlap-add, but we omit this discussion.

### 3.2. Number of measurements

The number of measurements required to capture a class of signals depends on several different factors:

- The sparsity level  $m$  and length  $d$  of the signals.
- The length  $B$  of the random filter.
- The *coherence* between the measurement matrix and the sparsity basis. When  $\Phi\Psi$  has small entries, signal acquisition requires fewer measurements.

Explaining these tradeoffs theoretically is a major project.

One expects that signals sparse in the time domain, i.e.,  $\Psi = \mathbf{I}$ , are the most difficult to acquire with random filters because of high coherence. Yet we present empirical evidence in Section 5 that random filters are effective for recovering time-sparse signals: a random filter of length  $d$  performs as well as a fully Gaussian matrix. When the filter length decreases, the number of measurements increases somewhat. For signals sparse in the frequency domain, the number of measurements depends weakly on the filter length; a four-tap filter already yields good reconstruction probability.

ALGORITHM 1: RANDOM FILTER RECONSTRUCTION

**Initialize** residual  $\mathbf{r}_0 = \mathbf{y}$ . **For**  $t = 1, 2, \dots, N$  **do**

**A.** Find the column  $i_t$  of  $\Phi\Psi$  such that  

$$i_t = \arg \max_i |\langle \mathbf{r}_{t-1}, (\Phi\Psi)_i \rangle|.$$

**B.** Compute the new residual  

$$\mathbf{r}_t = \mathbf{y} - \mathbf{P}_t \mathbf{y}$$
 where  $\mathbf{P}_t$  is the orthogonal projector onto the span of the  $t$  columns chosen from  $\Phi\Psi$ .

**Output:** Columns  $\{i_t\}$  and coefficients  $\{\hat{\theta}_{i_t}\}$  such that  

$$\mathbf{P}_N \mathbf{y} = \sum_{t=1}^N \hat{\theta}_{i_t} (\Phi\Psi)_{i_t}.$$

4. SIGNAL RECONSTRUCTION

Reconstructing a signal from the compressed data requires a nonlinear algorithm. The CS literature advocates greedy algorithms, such as Orthogonal Matching Pursuit [3] and Tree-based Matching Pursuit [4], or optimization-based algorithms involving  $\ell_1$  minimization [1, 2] for this purpose. Both of these approaches can be accelerated using the properties of random filters, and we believe that random filters will also lead to novel reconstruction algorithms that take full advantage of the localization and time-invariance of the measurements. In this paper, we adapt OMP to reconstruct signals.

Algorithm 1 is a high-level description of our OMP reconstruction procedure. The inputs to the algorithm are the random filter  $\mathbf{h}$  (which determines the matrix  $\Phi$ ), the sparsity basis  $\Psi$ , and the compressed data  $\mathbf{y} = \Phi\mathbf{s}$ . The algorithm returns  $N$  locations and estimated values for the nonzero entries of  $\theta$ .

Using the structure of the measurement process, we can implement Algorithm 1 very efficiently. For simplicity, suppose that the sparsity basis  $\Psi = \mathbf{I}$ , so the input signal is well approximated by a small number of spikes. The primary cost in Algorithm 1 is the computation of the inner products in Step A, which nominally requires  $O(dN)$  operations. This cost can be reduced substantially by taking into account the fact that  $\Phi$  performs filtering and downsampling. First, up-sample  $\mathbf{r}_t$  so it has length  $d$  and then convolve it with the time-reversed filter  $\mathcal{R}\mathbf{h}$ . Using direct convolution, this step costs  $O(dB)$  arithmetic operations. Using the FFT to implement the convolution, the cost is  $O(d \log d)$ .

We can also apply the orthogonal projector  $\mathbf{P}_t$  efficiently. Precompute the inner products between columns of  $\Phi$  using the fact that the Gram matrix  $\Phi^T \Phi$  is Toeplitz and banded. Then the marginal cost of calculating  $\mathbf{P}_t \mathbf{y}$  drops from  $O(tN)$  to  $O(N)$ , which is a significant reduction unless the sparsity level  $m$  is very small.

We believe that similar gains can be achieved in algorithms for  $\ell_1$  minimization by exploiting the structure of the random filter. This is an interesting area for future research.

5. NUMERICAL RESULTS

We present results from extensive numerical work, which offer compelling evidence that random filters are a powerful approach to compressive signal acquisition and reconstruction.

Three related experiments establish that random filters can capture sparse signals in three different sparsity bases. Let us describe the experimental setup. Each data point involves (i) a signal length  $d$ , (ii) a sparsity level  $m$  and a sparsity basis  $\Psi$ , (iii) a number  $N$  of measurements, and (iv) a filter length  $B$ . For each data point, we fix a random filter with  $\mathcal{N}(0, 1)$  taps. For each of 1000 trials, we generate a signal  $\mathbf{s}$  whose  $m$  nonzero coefficients are  $\mathcal{N}(0, 1)$ . We take  $N$  measurements and use Algorithm 1 to reconstruct the signal. If they match perfectly, then we record a *success*. The reconstruction probability is the fraction of the 1000 trials that result in success. As a control, we perform the same experiment using a fully random i.i.d. matrix  $\Phi$  with  $\mathcal{N}(0, 1)$  entries; Algorithm 1 (without enhancements) is used for reconstruction.

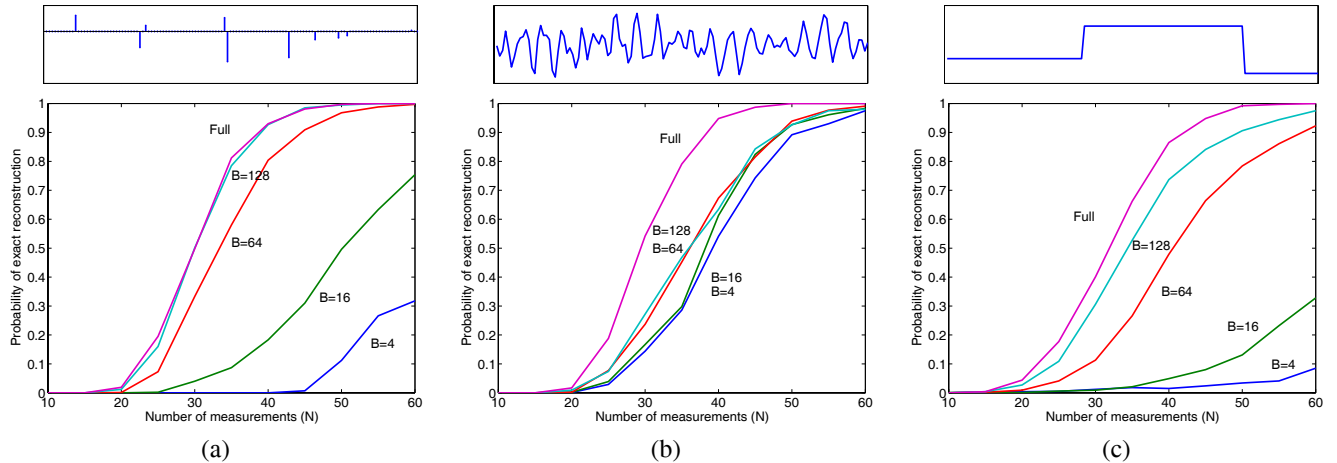
We begin with signals that are **sparse in the time domain**, i.e.,  $\Psi = \mathbf{I}$ . Recall that this case is challenging due to high coherence. We choose the signal length  $d = 128$  and sparsity  $m = 10$ . Figure 2(a) displays the probability of success for several filter lengths, in comparison with fully random measurements. Observe that the two longest filters ( $B = 64$  and  $128$ ) succeed almost as well as the fully Gaussian matrix, despite having far fewer degrees of freedom.

We now consider signals that are **sparse in the Fourier domain**, i.e.,  $\Psi = \mathcal{F}$ . As above, the signal length  $d = 128$  and sparsity  $m = 10$ . Figure 2(b) displays the probability of success for several filter lengths. Note that all four filters yield similar probabilities of success, which are slightly worse than the Gaussian matrix. The filter length has a limited impact since the Fourier basis is incoherent with the random filter.

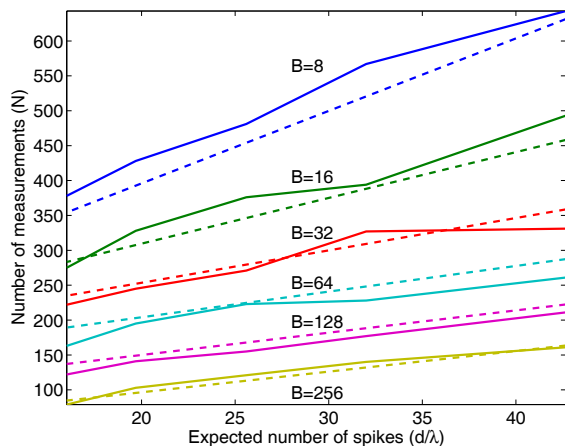
We performed the same experiment for signals **sparse in the Haar wavelet domain**. The results were slightly superior to the first experiment and somewhat worse than the second experiment because the Haar basis is localized in time—but not as much as the Dirac basis. We omit the figure.

Two additional experiments examine the performance of random filters for signals with few degrees of freedom per unit time, cf. [5]. First, we attempt to acquire and reconstruct **piecewise constant signals**. In each trial, the signal has two uniformly random breakpoints and  $\mathcal{N}(0, 1)$  coefficients. Reconstruction is performed with respect to the Haar wavelet basis. Otherwise, the experimental setup is identical. Figure 2(c) displays the results for four filter lengths. Since the wavelet coefficients of piecewise polynomials are localized in time, the longer filters convey a significant advantage.

Finally, we attempt to acquire and reconstruct (discrete) **Poisson processes** using random filters. These signals contain spike trains with geometrically distributed interarrival times. The symbol  $\lambda$  denotes the expected interarrival time, so the expected number of spikes is  $d/\lambda$ . In this experiment,



**Fig. 2.** Probability of exact reconstruction versus the number of measurements  $N$  for four filter lengths  $B$ . Signal length  $d = 128$ . (a) Time-sparse signals. (b) Fourier-sparse signals. (c) Piecewise polynomial signals. Typical signals appear at top.



**Fig. 3.** Number of measurements  $N$  to achieve 90% probability of exact reconstruction of a discrete Poisson process as a function of interarrival rate  $\lambda$  for different filter lengths  $B$ . The dashed lines mark the trend described by (\*). Signal length  $d = 1024$ .

the signal length  $d = 1024$ , and we examine five interarrival times. For each filter length, we determine the least number  $N$  of measurements required to achieve 90% recovery probability. The results appear in Figure 3. The following trend matches the data well:

$$N = \frac{d}{\lambda} \left( \frac{61.4}{B} + 2.72 \right) + 198 \cdot 0.9935^B. \quad (*)$$

In words, the number of measurements is a linear function of the sparsity level  $m = d/\lambda$ . The intercept can be interpreted as a “startup cost” for using a filter of length  $B$ , which decreases for longer filters. The slope can be viewed as the number of measurements required to increase the number of spikes by one; there is a minimal cost of 2.72 plus another term inversely proportional to the filter length. More research is necessary to understand this phenomenon.

## 6. CONCLUSIONS AND FUTURE WORK

This paper has proposed a new method, *random filtering*, for compressive sampling of sparse signals. We show that convolution with a random filter followed by downsampling yields an efficient summary of a compressible signal. This approach has some features in common with proposed methods for Compressed Sensing. In particular, random filters are *universal*, because they can capture a wide variety of sparse signals using a small number of random linear measurements. In contrast with CS, random filters require much *less storage and computation* for measurement and for reconstruction. Moreover, random filters are *time-invariant and time-localized*, so they can be used to acquire streaming signals, and they have potential for real-time applications.

Since this paper is exploratory, it offers a cornucopia of possibilities for future work. First, the theory of random filtering requires further development. Second, it would be valuable to understand the tradeoffs between measurement and computation costs. Third, we hope to extend this work to settings such as compressive sampling of images and video.

## 7. REFERENCES — Available at [dsp.rice.edu/cs](http://dsp.rice.edu/cs)

- [1] E. J. Candès and T. Tao, “Near optimal signal recovery from random projections: Universal encoding strategies?,” Preprint, Oct. 2004.
- [2] D. L. Donoho, “Compressed sensing,” Preprint, Sep. 2004.
- [3] J. A. Tropp and A. C. Gilbert, “Signal recovery from partial information via Orthogonal Matching Pursuit,” Submitted to *IEEE Trans. Inform. Theory*, April 2005.
- [4] M. F. Duarte, M. B. Wakin, and R. G. Baraniuk, “Fast reconstruction of piecewise smooth signals from random projections,” in *Online Proc. SPARS05*, Rennes, France, Nov. 2005.
- [5] M. Vetterli, P. Marziliano, and T. Blu, “Sampling signals with finite rate of innovation,” *IEEE Trans. Signal Proc.*, vol. 50, no. 6, June 2002.