Sampling Sparse Multitone Signals with a Random Demodulator

Michael Lexa, Mike Davies and John Thompson
{michael.lexa,john.thompson,mike.davies}@ed.ac.uk

University of Edinburgh
Institute of Digital Communications
Technical Report

May 2010, Revised October 2010

Abstract

The random demodulator is a compressed sensing based, uniform sub-Nyquist sampling strategy for acquiring spectrally sparse continuous-time signals. This report presents a concise mathematical description of the random demodulator and serves as a reference to the accompanying MATLAB software that simulates the sampling and signal recovery.

1 Signal Model and System Description

Sparse multitone signals. *Multitone signals* are bandlimited, continuous-time, periodic signals. As such, every multitone signal $x(t)$ has a Fourier series (FS) representation

$$x(t) = \sum_{n=-N/2}^{N/2-1} X(n) e^{j\frac{2\pi}{T_x}nt}, \tag{1}$$

where $\{X(n)\}$ denotes the FS coefficients of $x(t)$, T_x denotes the period of $x(t)$ in seconds, and N is assumed to be a positive even integer. The summation limits are finite because the harmonics of the fundamental tone $1/T_x$ are assumed to be bounded, i.e. $-\pi W \leq \frac{2\pi n}{T_x} < \pi W$ for $n \in \mathbb{Z}$. It is in this sense that multitone signals are bandlimited. Letting $N = T_x W$, the summation limits in (1) result. We say a multitone signal is *sparse* if a small number of the FS coefficients (out of the $N+1$ possible) are nonzero. More precisely, let K denote the number of nonzero coefficients (or equivalently the number of nonzero frequencies), then a sparse multitone signal is one that satisfies $K \ll N$.

Random demodulator. The RD is a uniform sub-Nyquist sampling strategy for acquiring sparse multitone signals [1,2]. For this system, a sparse multitone signal $x(t)$ is first multiplied by a periodic signal $p(t)$ and then the product $x(t)p(t)$ is filtered and sampled at a sub-Nyquist rate (see Figure 1). The signal $p(t)$ is a periodic extension of a finite duration random square wave taking values $\{\pm 1\}$. Let T_p denote the period of $p(t)$ and let its chipping rate[1] equal W Hz. Here, we assume T_p is

[1] The chipping rate is defined as the fastest rate at which the signals can switch from $+1$ to -1 or vice versa.

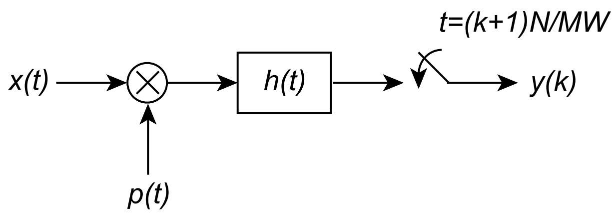

Figure 1: Schematic diagram of the random demodulator.

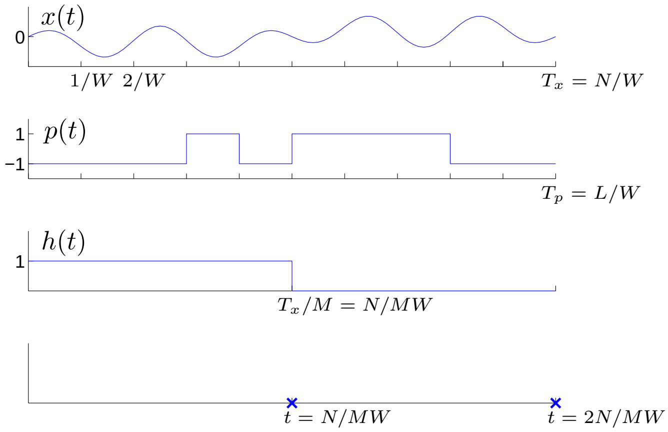

Figure 2: Signals associated with random demodulator. The top graph shows one period of a typical sparse multitone signal ($W = 10$ Hz, $N = 10, T_x = 1$ sec). The middle graphs show one period of a possible realization of $p(t)$ ($L = 10, T_p = 1$) and the impulse response of the ideal integrator ($M = 2$), respectively. The bottom graph indicates the sampling times within the observation interval. Here, the sampling rate is $M/N = 1/5$ of the Nyquist rate or 2 Hz.

some integer multiple of the Nyquist period ($T_p = L/W, L > 1$). The signal $p(t)$ has a Fourier series expansion

$$p(t) = \sum_{n=-\infty}^{\infty} P(n) e^{j\frac{2\pi}{T_p}nt},$$

where $\{P(n)\}$ is the set of Fourier series coefficients. The filter $h(t)$ is an ideal integrator with impulse response $h(t) = \mathrm{rect}\left(\frac{2M}{T_x}t - 1\right)$, where

$$\mathrm{rect}(x) = \begin{cases} 1 & \text{for } -1 \leq x \leq 1 \\ 0 & \text{otherwise} \end{cases}$$

and T_x is the period of $x(t)$ (c.f. (1)). The sampling period T_s is taken to be $M \in \mathbb{Z}^+$ times shorter than the period of $x(t)$ ($T_s = T_x/M$), or equivalently, M times shorter than the observation interval. The system therefore samples at the rate of MW/N Hz, $N > M$.

In the following analysis, we assume M evenly divides N and that the period of $x(t)$ (observation interval) equals the period of $p(t)$ ($T_x = T_p$). The first assumption is made for ease of exposition but can be removed with some modifications.

For this problem, the specific number and location of the nonzero FS coefficients are unknown. The goal therefore is to recover the support (i.e. the frequencies) and amplitudes of the nonzero FS coefficients and reconstruct the original input signal from the sub-Nyquist samples.

2 System Analysis and Signal Reconstruction

By inspection of Figure 1,

$$g(t) = x(t)p(t) * h(t) = \int_{-\infty}^{\infty} x(\tau)p(\tau)h(t-\tau)\, d\tau \tag{2}$$

$$= \int_{t-\frac{T_x}{M}}^{t} x(\tau)p(\tau)\, d\tau \tag{3}$$

$$= \sum_{n=-N/2}^{N/2-1} X(n) \int_{t-\frac{T_x}{M}}^{t} p(\tau) e^{j\frac{2\pi}{T_x}n\tau}\, d\tau, \tag{4}$$

where $*$ denotes convolution. Sampling at $t = (k+1)\frac{T_x}{M}$ for $k = 0, 1, \dots$ yields

$$\begin{aligned}
y(k) = g\big((k+1)\tfrac{T_x}{M}\big) &= \sum_{n=-N/2}^{N/2-1} X(n) \int_{k\frac{T_x}{M}}^{(k+1)\frac{T_x}{M}} p(\tau) e^{j\frac{2\pi}{T_x}n\tau}\, d\tau \\
&= \sum_{n=-N/2}^{N/2-1} X(n) \sum_{m=0}^{N/M-1} \int_{k\frac{T_x}{M}+\frac{m}{W}}^{k\frac{T_x}{M}+\frac{m+1}{W}} p(\tau) e^{j\frac{2\pi}{T_x}n\tau}\, d\tau \\
&= \sum_{n=-N/2}^{N/2-1} \sum_{m=0}^{N/M-1} p_{k\frac{N}{M}+m} X(n) \int_{k\frac{T_x}{M}+\frac{m}{W}}^{k\frac{T_x}{M}+\frac{m+1}{W}} e^{j\frac{2\pi}{T_x}n\tau}\, d\tau \\
&= \begin{cases} T_x \sum_{n=-N/2}^{N/2-1} \sum_{m=0}^{N/M-1} p_{k\frac{N}{M}+m} X(n) \frac{e^{j\frac{2\pi}{N}n}-1}{j2\pi n} e^{j\frac{2\pi}{N}n\left(k\frac{N}{M}+m\right)}, & n \neq 0 \\ \frac{1}{W} \sum_{n=-N/2}^{N/2-1} \sum_{m=0}^{N/M-1} p_{k\frac{N}{M}+m} X(n), & n = 0 \end{cases}
\end{aligned} \tag{5}$$

where the first three steps follow from the additivity of the integral and the specific nature of $p_i(t)$. Here, $p_{k\frac{N}{M}+m} = p(k\frac{T_x}{M} + \frac{m}{W})$. By letting $l = k\frac{N}{M} + m$, (5) may be rewritten as

$$y(k) = \frac{N}{W} \sum_{n=-N/2}^{N/2-1} \sum_{l=k\frac{N}{M}}^{(k+1)\frac{N}{M}-1} p_l \frac{e^{j\frac{2\pi}{N}n} - 1}{j2\pi n} e^{j\frac{2\pi}{N}nl} X(n), \quad \text{for } k = 0, 1, \dots, \tag{6}$$

where this expression should be understood to be consistent with (5) for $n = 0$ above. For practical systems, only a finite number of samples will be collected for further processing. Mathematically, this operation is equilvalent to windowing (multiplying) the sequence $y(k)$ by the discrete rectangular function

$$w(k) = \begin{cases} 1, & 0 \le k \le M-1 \\ 0, & \text{otherwise} \end{cases}. \tag{7}$$

In (6), multiplication by $w(k)$ has no effect other than to restrict the range of k. Typically, windowing is an important operation to consider because it significantly affects frequency domain information on which most digital sampling schemes rely. Interestingly, however, windowing does not play an important role in the operation of the random demodulator in this respect because signal reconstruction has nothing to do with preserving or manipulating frequency domain information. Reconstruction instead relies on compressed sensing recovery algorithms. To see this, consider the matrix form of (6) for $k = 0, \dots, M-1$,

$$\mathbf{y} = \mathbf{\Phi\Psi s} \tag{8}$$

where $\mathbf{y} = [y(0), \dots, y(M-1)]'$, $\mathbf{\Psi}$ is a $N \times N$ sparsifying matrix whose $(n,l)^{th}$ entry equals $e^{j\frac{2\pi}{N}nl}, n = -N/2, \dots, N/2-1,\ l = 0, \dots, N-1$, $\mathbf{\Phi}$ is a $M \times N$ measurement matrix of the form

$$\mathbf{\Phi} = \begin{bmatrix} p_0 & \cdots & p_{\frac{N}{M}-1} & & & & & & \\ & & & p_{\frac{N}{M}} & \cdots & p_{\frac{2N}{M}-1} & & & \\ & & & & & & \ddots & & \\ & & & & & & p_{(M-1)\frac{N}{M}} & \cdots & p_{N-1} \end{bmatrix}$$

and

$$\mathbf{s} = \begin{bmatrix} \alpha_{-N/2} X(-\frac{N}{2}) \\ \vdots \\ \alpha_{N/2-1} X(\frac{N}{2}-1), \end{bmatrix}, \qquad \alpha_n = \frac{T_x}{j2\pi n}\big(e^{j\frac{2\pi}{N}n} - 1\big), \quad \alpha_0 = \frac{1}{W}.$$

This is the matrix equation derived by Tropp et al. in [2] that relates the discrete time-domain output samples $y(k)$ to the FS coefficients of the input signal $X(n)$.

We can reconstruct $x(t)$ from the samples $\mathbf{y}$ because (8) can be solved (with high probability) given the sparse nature of the problem and provided that the matrix $\mathbf{\Phi\Psi}$ satisfies certain conditions (see [2] for more details). Normally, we would need to collect $M = N$ samples to make (8) invertable, but the sparsity of x enbles recovery from only $M \ll N$ samples. Ideally, we want to find the sparsest solution (i.e. the smallest number of nonzero Fourier series coefficients) that agrees with the data,

$$\widehat{\mathbf{s}} = \arg\min \|\mathbf{v}\|_0 \quad \text{subject to} \quad \mathbf{A}\mathbf{v} = \mathbf{y}, \tag{9}$$

where $\mathbf{A} = \mathbf{\Phi\Psi}$ and $\|\cdot\|_0$ denotes the ℓ_0 norm that simply counts the number of nonzero elements of its argument. Because of its combinatorial nature, (9) is often very computationally burdensome to solve. Fortunately, any of a number of existing CS algorithms can be used to recover (estimate) $\mathbf{s}$. For example, ℓ_1-minimization [3], orthogonal matching pursuit (OMP) [4] or iterative hard thresholding [5] all apply. By obtaining the estimate $\widehat{\mathbf{s}}$, one obtains estimates for the magnitudes of the nonzero FS coefficients, as well as their location. That is, one obtains a set of FS coefficients over a set of indices Ω that represent the spectral support of $x(t)$, $\{\widehat{X}(n), n \in \Omega \subset [-N/2, N/2-1]\}$. Knowing these indices is tantmount to knowing the nonzero frequencies comprising $x(t)$. Thus once (8) is solved, $x(t)$ can be reconstructed (approximately) by plugging the recovered magnitudes and frequencies into (11),

$$\widehat{x}(t) = \sum_{n\in\Omega} \widehat{X}(n) e^{j\frac{2\pi}{T_x}nt} \tag{10}$$

where $\widehat{X}(n) = \widehat{\mathbf{s}}/\alpha_n, n \in \Omega$ are the nonzero FS coefficients.

For more information about the random demodulator's performance guarantees and the conditions of successful signal recovery, see [2].

3 Computer Simulations

Computer simulations cannot replicate continuous-time analog processing, rather they must replace the signals of interest and the processing by their digital equivalents. In the case of the random demodulator, the signals $x(t)$ and $p(t)$ are replaced by finite length vectors and the analog filter $h(t)$ is replaced by a digital filter. The elements of the input vector $\mathbf{x} = [x(0), \ldots, x(N-1)]'$ are considered samples of $x(t)$ obtained at or above the Nyqusit rate. Thus a computer simulated random demodulator *subsamples* a Nyquist rate sequence and then reconstructs $\mathbf{x}$ from the samples $y(k), k = 0, \ldots, M-1$.

For simulations, system analysis simplifies because the input $\mathbf{x}$ now has a discrete time Fourier (DFT) representation,

$$x(m) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} X(n) e^{j\frac{2\pi}{N}nm}. \tag{11}$$

$$\begin{aligned}
g(m) = x(m)p(m) * h(m) &= \sum_{\max(0,m+1-N)}^{\min(m,N-1)} x(l)p(l)h(m-l) && (12)\\
&= \sum_{l=\max(0,m+1-\frac{N}{M})}^{\min(m,N-1)} x(l)p(l) && (13)\\
&= \frac{1}{N} \sum_{n=-N/2}^{N/2-1} X(n) \sum_{l=\max(0,m+1-\frac{N}{M})}^{\min(m,N-1)} p(l) e^{j\frac{2\pi}{N}nl}. && (14)
\end{aligned}$$

Sub-sampling at $m = (k+1)\frac{N}{M} - 1$ for $k = 0, \ldots, M-1$, yields

$$\begin{aligned}
y(k) = g((k+1)\tfrac{N}{M}) &= \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \sum_{l=\max(0,k\frac{N}{M}+1-\frac{N}{M})}^{\min((k+1)\frac{N}{M}-1,N-1)} p(l)X(n)e^{j\frac{2\pi}{N}nl} && (15)\\
&= \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \sum_{l=k\frac{N}{M}}^{(k+1)\frac{N}{M}-1} p(l)X(n)e^{j\frac{2\pi}{N}nl}, \quad k = 0, \ldots, M-1. && (16)
\end{aligned}$$

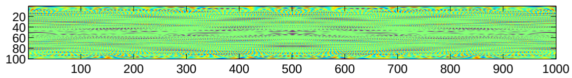

Figure 3: Graphical representation of the matrix $\mathbf{A} = \mathbf{\Phi\Psi}$. $T_x = 1$ sec, $N = 1000, M = 100$.

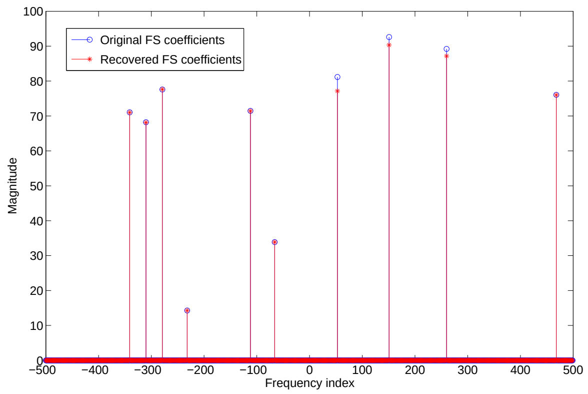

Figure 4: This figure shows the support and amplitide of the original nonzero FS coefficients (blue) compared to the recovered support and amplitudes (red).

A comparison of (16) with (6) reveals the absence of the complex factor $\alpha_n = \frac{T_x}{j2\pi n}\left(e^{j\frac{2\pi}{N}n} - 1\right)$ and the addition of the factor $1/N$ in the digital formulation. The MATLAB scripts accompanying this report reflect these changes. Similar to the analysis in Section 2, (16) may be written in matrix form

$$\mathbf{y} = \mathbf{\Phi\Psi x}, \tag{17}$$

where all quantities are as before. Note that in this case the vector $\mathbf{x}$ replaces $\mathbf{s}$ because of the absence of α_n.

3.1 An Example

Figures 3 to 6 display the results of one proto-typical example where a multitone signal bandlimited to 500 Hz with 10 randomly chosen frequencies is sampled at a rate that is an order of magnitude below the Nyquist rate (W = 1000 Hz, sampling rate = 100 Hz). The MATLAB scripts used to simulate the sampling and reconstruction of the random demodulator are available at [6]. The signal is reconstructed using the OMP algorithm provided as part of the *Sparsify 0.4* software package written by Thomas Blumensath [7]. For this example, the figures clearly indicate exact recovery of the signal's frequency support and recovery of the nonzero amplitudes with a mean-squared error of 0.00135.

References

[1] J. Laska, S. Kirolos, M. Duarte, R. Baraniuk, and Y. Massoud, "Theory and implementation of an analog-to-information converter using random demodulation," *Proc. IEEE Int. Symp. Circuits and Systems (ISCAS)*, pp. 1959–1962, May 2007.

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

[3] E. Candes, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. Info. Th.*, vol. 52, no. 2, pp. 489–509, Feb 2006.

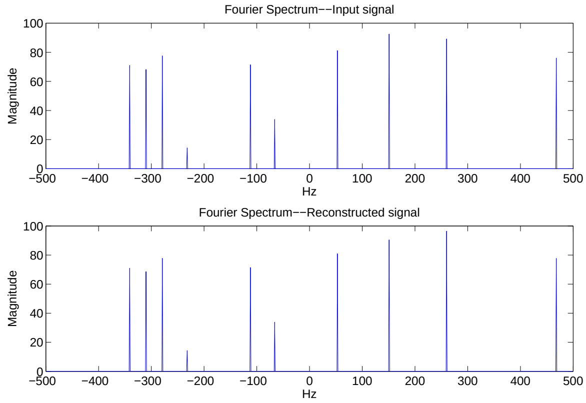

Figure 5: The plots show the frequency spectra of $x(t)$ and $\hat{x}(t)$ respectively.

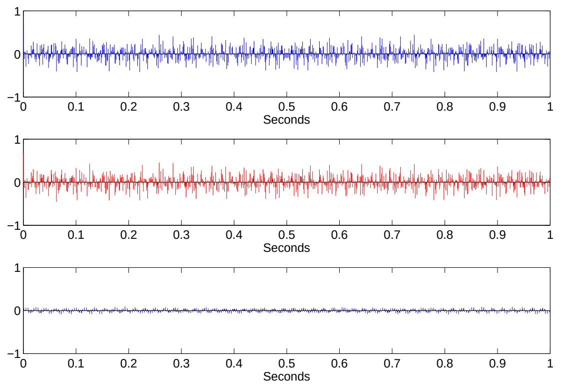

Figure 6: The top and middle panels show the original (blue) and reconstructed (red) signals, respectively. The bottom panel displays the difference signal.

[4] J. Tropp, "Signal recovery from random measurements via orthogonal matching persuit," *IEEE Trans. Info. Th.*, vol. 53, no. 12, pp. 4655–4666, Dec 2007.

[5] T. Blumensath and M. Davies, "Interative thresholding for sparse approximations," *Jour. of Fourier Anal. and App.*, 2007.

[6] Michael Lexa, "Random demodulator MATLAB scripts," [Online]. Available: `www.see.ed.ac.uk/~mlexa`, 2010.

[7] Thomas Blumensath, "Sparsify 0.4 software package," [Online]. Available: `www.personal.soton.ac.uk/tb1m08/sparsify/`, 2010.