

Sigma Delta Quantization for Compressive Sensing

Petros Boufounos and Richard G. Baraniuk

Rice University, ECE Department

ABSTRACT

Compressive sensing is a new data acquisition technique that aims to measure sparse and compressible signals at close to their intrinsic information rate rather than their Nyquist rate. Recent results in compressive sensing show that a sparse or compressible signal can be reconstructed from very few measurements with an incoherent, and even randomly generated, dictionary. To date the hardware implementation of compressive sensing analog-to-digital systems has not been straightforward. This paper explores the use of Sigma-Delta quantizer architecture to implement such a system. After examining the challenges of using Sigma-Delta with a randomly generated compressive sensing dictionary, we present efficient algorithms to compute the coefficients of the feedback loop. The experimental results demonstrate that Sigma-Delta relaxes the required analog filter order and quantizer precision. We further demonstrate that restrictions on the feedback coefficient values and stability constraints impose a small penalty on the performance of the Sigma-Delta loop, while they make hardware implementations significantly simpler.

Keywords: Sigma Delta, Quantization, Compressive Sensing

1. INTRODUCTION

Compressive sensing is a new low-rate signal acquisition method for signals that are sparse or compressible.¹⁻⁵ The fundamental premise is that certain classes of signals, such as natural images or some communications signals, have a representation in terms of a sparsity inducing basis (or sparsity basis for short) where most of the coefficients are zero or small and only a few are large. For example, smooth signals and piecewise smooth signals are sparse in a Fourier and wavelet basis, respectively.

Recent results^{1,5} demonstrate that sparse or compressible signals can be directly acquired at a rate significantly lower than the Nyquist rate. The low-rate acquisition process projects the signal onto a small set of vectors, a dictionary, that are incoherent with the sparsity basis. The signals can subsequently be recovered using a greedy algorithm or a linear program that determines the sparsest representation consistent with the acquired samples. The quality of the reconstruction depends on the compressibility of the signal, the choice of the reconstruction algorithm, and the incoherence of the sampling dictionary with the sparsity basis. One of the most useful results is that randomly generated dictionaries are universal in the sense that, with very high probability, they are incoherent with any fixed sparsity basis. This property makes such dictionaries very desirable for compressive sensing applications.

To date, implementing random projections in an analog-to-digital data acquisition device is not straightforward. The most common architectures use an analog high-rate front end implementing the random projections, followed by a low-rate precision analog-to-digital converter decoupled from the analog projection system.⁶⁻⁹ Due to the limitations of analog hardware, the resulting devices are severely limited in the types of projections they can implement. Some systems are based on randomly modulating or randomly sampling the input.⁶⁻⁸ These systems lack universality since they specifically require that the input be sparse in the Fourier or a similar basis. Systems that randomly filter and subsequently subsample the input⁹ require a significant number of precision analog multipliers or a high-order switched capacitor filter. In this paper we develop a Sigma-Delta based architecture to efficiently implement a broad class of random projections.

Sigma-Delta quantization of oversampled bandlimited or bandpass signals has been successfully used in the design of Analog-to-Digital (A/D) and Digital-to-Analog converters.¹⁰ The usual architecture of classical A/D converters is shown in Fig. 1(a). The input signal is filtered using an analog anti-aliasing low-pass filter (LPF), and

Rice University, ECE Department MS 380, P.O. Box 1892, Houston, TX. {petrosb, richb}@rice.edu; dsp.rice.edu/cs.



(a) Traditional A/D architecture: Nyquist-rate sampling, followed by fine quantization. The input signal is filtered using an analog anti-aliasing low-pass filter (LPF), and then sampled (C/D) and quantized ($Q(\cdot)$) at the Nyquist rate.



(b) Sigma-Delta A/D architecture: Oversampling, followed by coarse Sigma-Delta quantization. The downsampling and the low-pass filter are implemented in the digital domain.

Figure 1. Typical high-level architecture of Analog-to-Digital converters.



(a) Standard CS architecture: Random projections, implemented in the analog domain, followed by fine quantization.



(b) Our proposed Sigma-Delta CS architecture: High-rate sampling, followed by coarse Sigma-Delta quantization and random projections. The Sigma-Delta block architecture is very similar to the classical Sigma-Delta block in Fig. 1(b), and the random projections are implemented in the digital domain.

Figure 2. High-level architecture of Compressive Analog-to-Digital converters.

then sampled (C/D) and quantized ($Q(\cdot)$) at the Nyquist rate. The main trade-off in Sigma-Delta quantization, demonstrated in Fig. 1(b), is between the quantization accuracy and sampling rate. The anti-aliasing low-pass filter (LPF) and the precision quantizer in Fig 1(a) are difficult to implement due to their tight specifications. On the other hand, the Sigma-Delta architecture of Fig. 1(b) uses an inexpensive coarse quantizer in the Sigma-Delta loop and combines the downsampling and the low-pass filter in the digital domain, essentially eliminating the need for an anti-aliasing filter in the input.¹¹ The inexpensive oversampled Sigma-Delta architecture provides the same performance as very accurate fine quantizers operating at the Nyquist rate. Although classical Sigma-Delta conversion operates on uniformly oversampled signals, recent work demonstrates that Sigma-Delta quantization algorithms are also useful in reducing the error in the quantization of arbitrary frame representations.^{12–14}

The goal of this paper is to use the dictionary generated by the random projections and design principles for Sigma-Delta quantizers¹⁴ to implement a compressive sensing system. The fundamental architectural modifications, demonstrated in Fig. 2 parallel those demonstrated in Fig. 1(a) and (b). As described above, the analog random-projection component and the precision quantizer in Fig. 2(a) can be implemented easily only using random sampling or random modulation of the input signal.^{6–8} Such devices are useful only for the acquisition of signals that are sparse in the Fourier basis. An implementation that can compressively acquire a wide range of signals requires that the random projections component either uses a very long switched capacitor filter with randomly and rapidly changing coefficients or computes a discrete Fourier transform in the analog domain.⁹

In a Sigma-Delta based compressive sensing system these components are replaced by a coarse quantizer in a Sigma-Delta loop, followed by a digital-domain random projection component, as shown in Fig. 2(b). Although the Sigma-Delta loop requires a long switched capacitor filter, this is significantly shorter than the

filter implementing the random projections in Fig. 2(a). Unfortunately, implementation of Sigma-Delta loops for compressive sensing dictionaries poses challenges not present in frames commonly used in sampling applications.

In the next section a brief background on compressive sensing and on Sigma-Delta quantization is presented. It is thus demonstrated in Sec. 3 that the incoherence properties desired in compressive sensing dictionaries are mismatched to the properties desired in dictionaries suitable for Sigma-Delta converters. One way to overcome this mismatch is to use high-order, but potentially unstable, feedback loops. Section 3.3 discusses the stability of the feedback loops, and provides a sufficient, albeit severe, stability condition. Although in practice this condition is often relaxed, it demonstrates the importance of the magnitude of the compensation coefficients.

In classical Sigma-Delta quantizer design, the frame is usually predetermined. Therefore, the compensation coefficients in the feedback loop can be precomputed off-line. This is not the case in some compressive sensing systems, in which the sampling dictionary is generated randomly during the acquisition process. In such systems, the coefficients should also be computed efficiently during the acquisition process, as the dictionary is generated. Section 3 discusses these challenges in detail and describes the design goals of the system. Section 4 presents an algorithm to compute the feedback coefficients efficiently. This algorithm is particularly useful for systems that generate the dictionary at run-time.

2. BACKGROUND

This section presents a brief background on compressive sensing, followed by a brief overview of Sigma-Delta noise shaping for frame representations. The intent is to establish the notation and serve as a reference for the remainder of the paper.

2.1 Compressive Sensing

Compressive sensing is a new sampling and reconstruction method for signals that are known to be sparse or compressible in some basis.^{1,5} Without loss of generality, we assume a signal \mathbf{x} in an N dimensional vector space, such that x_n represents the signal in some sampling basis: $\mathbf{x} = \sum_n x_n \mathbf{b}_n$. The signal is K -sparse in a sparsity-inducing basis $\{\mathbf{s}_k\}$ if there are at most K non-zero coefficients $\{a_k\}$ in the basis expansion $\mathbf{x} = \sum_k a_k \mathbf{s}_k$. The signal is K -compressible if it is well approximated by the K most significant coefficients in the expansion.

Sparse and compressible signals can be sampled using inner products with a set of measurement vectors $\{\mathbf{u}_k\}$:

$$y_k = \langle \mathbf{x}, \mathbf{u}_k \rangle = \sum_n x_n u_{k,n}, \quad (1)$$

in which $u_{k,n}$ is the expansion coefficient of each measurement vector \mathbf{u}_k in the sampling basis \mathbf{b}_n . In the same space, Eq. (1) can also be viewed as a synthesis equation of the vector \mathbf{y} , which contains all the measurements, from the coefficients x_n in the dictionary $\mathbf{f}_n = \sum_k u_{k,n} \mathbf{b}_n$:

$$\mathbf{y} = \sum_n x_n \mathbf{f}_n. \quad (2)$$

The vector \mathbf{y} is sufficient to recover the signal, as long as the dictionary satisfies the restricted isometry property (RIP) of a certain order.^{1,5} Assuming that the sampling basis is the same as the sparsity basis and that the dictionary elements have unit norm, the dictionary satisfies RIP of order K with RIP constant δ_K if:

$$(1 - \delta_K) \sum_{i \in S_K} a_i^2 \leq \left\| \sum_{i \in S_K} a_i \mathbf{f}_i \right\|_2 \leq (1 + \delta_K) \sum_{i \in S_K} a_i^2, \quad (3)$$

for all coefficients a_i and sets S_K , in which S_K denotes a set of K indices of dictionary elements and corresponding coefficients, and $\|\cdot\|_2$ indicates the ℓ_2 norm in the sampling basis. Qualitatively, the RIP property of order K with a small RIP constant δ_K dictates that any subset of K dictionary elements should contain nearly orthogonal elements. Robust invertibility of the measurement operation for K -sparse signals requires an RIP of at least $2K$. The requirements are even stricter to guarantee that the measurement operation can be efficiently inverted using

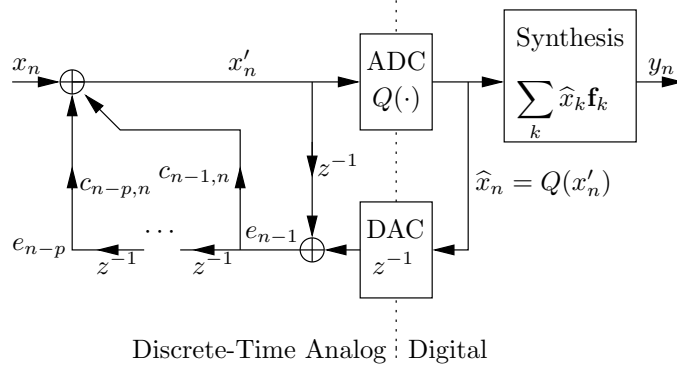


Figure 3. Block diagram of a typical Sigma-Delta Analog-to-Digital quantizer architecture. The ADC component is a high-rate sampler followed by a low-precision quantizer. The DAC component is a low precision discrete-time digital-to-analog converter. Except through the DAC and the ADC components, the digital components cannot access to the signal values at the analog side.

known algorithms such as basis pursuit⁵ or orthogonal matching pursuit.¹⁵ The RIP property can be similarly expressed for signals that are sparse in a basis different than the sampling basis.

Although it is not straightforward to design deterministic dictionaries that satisfy the RIP property with sufficiently high order and sufficiently low RIP constants, recent results demonstrate that generating a random dictionary is one of the most efficient ways to achieve the desired RIP characteristics with very high probability.²

2.2 Sigma-Delta Noise Shaping

Sigma-Delta quantization exploits the redundancy of the synthesis dictionary to provide robustness to quantization of the synthesis coefficients. It is an iterative algorithm that sequentially quantizes each coefficient in the redundant expansion and feeds back the quantization error to linearly modify the subsequent, yet unquantized, coefficients. Specifically, a vector represented using Eq. (2) is to be quantized. At every iteration n , the quantizer quantizes x_n to $\hat{x}_n = Q(x_n) = x_n - e_n$, in which $Q(\cdot)$ denotes the non-linear quantization function, and e_n is the quantization error. The subsequent coefficients are then updated to $x'_{n+i} = x_{n+i} - c_{n,n+i}e_n$, for $i = 1, \dots, p$, where p denotes the order of the quantizer loop. The next iteration quantizes x_{n+1} as it has been modified from all the previous iterations. In other words, the quantizer at each step quantizes

$$x'_n = x_n - \sum_{i=1}^p c_{n-i,n} e_{n-i}, \quad (4)$$

in which x_n is the original coefficient from Eq. (2). A typical Sigma-Delta quantizer block diagram, implementing an ADC, is shown in Fig. 3. The remainder of this paper assumes this specific implementation. We further assume that $Q(\cdot)$ represents a uniform midrise scalar quantizer with quantization interval Δ and quantization levels in the range $\pm q_{max}$.

The total Sigma-Delta quantization error is

$$\mathcal{E} = \sum_n e_n \left(\mathbf{f}_n - \sum_{i=1}^p c_{n,n+i} \mathbf{f}_{n+i} \right), \quad (5)$$

in which e_n denotes the error due to the quantization of each coefficient after all the previous coefficients have been quantized and the updates have been applied.

The Sigma-Delta design problem involves selecting the feedback coefficients $c_{n,n+i}$ to reduce the total error, subject to hardware and stability constraints. Unfortunately, the non-linearity of the quantization function $Q(\cdot)$ in the feedback loop makes Eq. (5) difficult to optimize for analytically. A common, but often inaccurate,

approach is to model the error e_n as a white random process with variance σ_e^2 and minimize the total error power of the linearized system:

$$E\{\mathcal{E}^2\} = \sigma_e^2 \sum_n \left\| \mathbf{f}_n - \sum_{i=1}^p c_{n,n+i} \mathbf{f}_{n+i} \right\|_2^2. \quad (6)$$

Alternatively, an upper bound on the total error in Eq. (5) can be derived using the triangle inequality, and subsequently minimized:

$$\|\mathcal{E}\| \leq \frac{\Delta}{2} \sum_n \left\| \mathbf{f}_n - \sum_{i=1}^p c_{n,n+i} \mathbf{f}_{n+i} \right\|_2. \quad (7)$$

A third approach is to greedily select the coefficients such that the incremental error in each iteration is minimized. All three approaches lead to the same optimization problem, namely to select coefficients that reduce or minimize the magnitude of the residual \tilde{c}_n :

$$\tilde{c}_n = \left\| \mathbf{f}_n - \sum_{i=1}^p c_{n,n+i} \mathbf{f}_{n+i} \right\|_2, \quad (8)$$

for all n . The residual should be small compared to the norm of \mathbf{f}_n , which, without loss of generality, is assumed to be equal to 1 for the remainder of this paper. The optimal coefficient selection computes the projection of \mathbf{f}_n onto the span of $\{\mathbf{f}_{n+1}, \dots, \mathbf{f}_{n+p}\}$.¹⁴ Still, hardware and stability considerations, among others, often necessitate a different design. The remainder of this paper discusses the properties of Eq. (5) and (8) in the context of compressive sensing, presents the difficulties in minimizing Eq. (8), and provides alternative approaches to select the coefficients.

3. SIGMA-DELTA DESIGN FOR COMPRESSIVE SENSING

This section explores the use of the hardware architecture in Fig. 3 to implement a compressive sensing device and discusses the challenges of this design.

3.1 Design Overview

The basic premise is that sampling the signal at the Nyquist rate oversamples the signal compared to the rate required using the random projections. This oversampling can be exploited using a Sigma-Delta component. Specifically, we use the same architecture presented in Fig. 3 with two modifications compared to a classical Sigma-Delta system:

- Instead of low-pass filtering, the synthesis component digitally computes the random projections using the samples x_n and the random measurement vectors, as defined in Eq. (2).
- Instead of using constant feedback coefficient values $c_{n,n+i}$ in the analog feedback loop, the coefficients dynamically change in accordance to the random dictionary. This is necessary because, in contrast to classical Sigma-Delta systems, compressive sensing systems rarely use shift-invariant dictionaries. Thus, the optimal coefficient values in Eq. (8) are different at every time step n .

In the terms of the high-level architecture presented in Fig. 2(b), the design in Fig. 3 implements the last two system components. Specifically, the Sigma-Delta feedback loop corresponds to the Sigma-Delta/Coarse $Q(\cdot)$ component, while the digital domain synthesis component corresponds to the random projections component.

The proposed design can implement a large variety of compressive sensing dictionaries, but avoids using the high-order analog random projection filters described in Sec. 1. These are implemented instead in the digital-domain synthesis component. The synthesis vectors used in this component are the \mathbf{f}_k implied by the compressed sensing measurement vectors, as described in Sec. 2.1. The coefficients in the feedback component

should, therefore, be computed accordingly, as described in Sec. 2.2. Still, using a Sigma-Delta architecture to implement a compressive sampling device poses challenges not encountered in classical systems.

The first challenge, described in detail in Sec. 3.2, is that a compressive sensing dictionary requires a high order Sigma-Delta feedback loop compared to a classical Sigma-Delta system. Specifically, in classical Sigma-Delta quantization, the order p of the feedback loop is usually low, rarely exceeding 4 or 5. Section 3.2 describes how RIP requirements on the dictionary essentially prohibit the design of low-order Sigma-Delta quantizers. Unfortunately, high order loops increase the potential for instabilities and make the design significantly harder. Section 3.3 considers sufficient constraints in the feedback coefficients to ensure the stability of Sigma-Delta designs. We should note that even though the analog feedback loop has high order compared to classical Sigma-Delta architectures, it has significantly lower order compared to the analog random projection filter required in Fig. 2(i).

The second challenge, discussed in Sec. 3.4, is that compressive sampling systems are often implemented using dictionaries that are randomly generated at the run-time of the system. In this case, the feedback coefficients cannot be pre-computed at the design stage of the system; they should be computed and change at run-time instead. Section 4 describes efficient algorithms to determine the coefficients for such systems. If the system uses pre-generated dictionaries, the optimal coefficient values can be computed at the design stage using simple constrained optimization methods.

We should emphasize that contrary to classical Sigma-Delta architectures, the signal is not oversampled beyond the Nyquist rate. In principle we could further oversample the signal beyond the Nyquist rate and improve the benefits of the Sigma-Delta architecture. This option is not explored in this paper; we aim to isolate the effect of the random projection dictionary.

3.2 Quantizer Order and the RIP

The RIP property of order K , as defined in Eq. (3), with a small RIP constant δ guarantees that Sigma-Delta compensation of order up to $p \leq K - 1$ is not effective. To show that, consider a set of $p \leq K - 1$ subsequent dictionary vectors $\{\mathbf{f}_n, \dots, \mathbf{f}_{n+p}\}$, for which the residual in Eq. (8) is equal to a small constant $\tilde{c}_n = \epsilon$. Substituting into the left-hand side of Eq. (3) and rearranging yields:

$$\epsilon \geq (1 - \delta_K) \left(1 + \sum_{i=1}^p c_{n,n+i}^2 \right), \quad (9)$$

$$\geq (1 - \delta_K) \quad (10)$$

which implies that if the RIP constant is small, then Sigma-Delta algorithms of order up to $p \leq K - 1$ have residual \tilde{c}_n close to 1. Therefore they cannot be effective in reducing the quantization error.

For comparison, consider the most common application of Sigma-Delta noise shaping, using a uniform oversampling frame of redundancy r . For this frame, the optimal choice for the feedback coefficient is $c_{n,n+1} = \text{sinc}(1/r)$ and the corresponding residual is equal to $\tilde{c}_n = \sqrt{1 - \text{sinc}^2(1/r)}$, where $\text{sinc}(x) = \sin \pi x / \pi x$. Even for an oversampling ratio $r = 4$, the lower bounds in Eq. (10) and Eq. (9) are $\delta_2 \geq 0.56$ and $\delta_2 \geq 0.76$ respectively, making the frame inappropriate for compressive sensing.

3.3 Sigma-Delta Stability

One of the potential issues in the implementation of Sigma-Delta noise shaping systems is their stability, especially as the order of the feedback loop increases. Although in most systems stability can be analyzed as part of the off-line design of the feedback loop,¹⁶ in many compressive sensing systems this is not the case; if the dictionary and the feedback coefficients are randomly generated on-line, the stability of the system should be a constraint in the on-line computation of the coefficients. If the dictionary is determined at the design stage, then the stability of the system becomes part of the constraints in optimizing Eq. (8).

The scalar quantizer $Q(\cdot)$ in Fig. 3 saturates when the input magnitude exceeds $(q_{\max} + \Delta/2)$. The system is considered to be stable if the input to the quantizer always has magnitude less than the saturation level. If

this is the case then the quantization error e_n is always bounded in magnitude by $\Delta/2$. Assuming the original representation coefficients in Eq. (2) are bounded in magnitude by x_{\max} , the triangle inequality can be used to bound Eq. (4):

$$x'_n \leq x_{\max} + \frac{\Delta}{2} \sum_{i=1}^p |c_{n-i,n}|. \quad (11)$$

It follows that each x'_n can be guaranteed not to saturate the scalar quantizer if:

$$\sum_{i=1}^p |c_{n-i,n}| \leq \frac{2}{\Delta} (q_{\max} - x_{\max}) + 1. \quad (12)$$

It should be noted that this is a worst-case stability condition. In practical designs, it is often violated to provide more flexibility in the design of the feedback loop at the expense of a slight probability of overflow of the quantizer. If that probability is small, the overflow can be ignored with only minimal error in the acquired signal. Study of this behavior is beyond the scope of this work. Nevertheless, the condition in Eq. (12) highlights the importance of the sum $\sum_{i=1}^p |c_{n-i,n}|$ in the stability of the system. For the remainder of this paper we use s_n to denote this sum:

$$s_n \equiv \sum_{i=1}^p |c_{n-i,n}|. \quad (13)$$

We also use s_{\max} to denote the maximum value of s_n over all n .

The condition of Eq. (12) can also be viewed as a requirement on the dynamic range of the quantizer. Specifically, as s_{\max} increases, q_{\max} , the dynamic range of the quantizer, should also increase to guarantee the stability of the system for the same input dynamic range x_{\max} .

3.4 Randomized Dictionaries

Usually, the frames used in Sigma-Delta quantizers have significant structural properties, such as shift-invariance, which are used to simplify the optimization of Eq. (8). This structure is preserved in the feedback coefficients, making their hardware implementation easier. This is not the case with the randomized dictionary used in compressive sensing. The coefficients computed have no structure and may take values in a large range.

Without any constraint or structure, the implementation of the feedback loop in Fig. 3 becomes a significant challenge. The left hand side of the figure, denoted using “Discrete-Time Analog,” is operating at a very high rate. Without any restriction on the coefficients, this implies that every analog gain element $c_{n-p,n}$ in the loop modifies its gain at the same high rate. Furthermore, a continuous range of feedback coefficients requires the use of very precise tunable elements or multipliers with a highly linear response and very short settling time. Present circuit technology makes such a design infeasible. A practical alternative is to restrict the coefficients to the set $\{-c, 0, +c\}$, or, more generally, to $\{-Lc, \dots, -c, 0, +c, \dots, +Lc\}$, in which L is a positive integer and c a positive real value. Note that we use the term *restriction* as opposed to the term *constraint* which refers to the stability constraint of the previous section. This convention is followed for the remainder of the paper.

Apart from their structure, Sigma-Delta quantizers for pre-determined frames have the further advantage that their design is performed once, off-line, before the quantizer is implemented. Thus, the complexity of computing appropriate feedback coefficients in Eq. (8) is usually not of significant interest. However, in some compressive sensing systems the dictionary is randomly generated at run time. Therefore, the feedback coefficients should also be computed at runtime, according to the generated dictionary. In this case the computation of the coefficients is part of the system design, and the complexity of this computation is an important factor.

4. COMPUTATION OF THE FEEDBACK COEFFICIENTS

In minimizing the error in Eq. (5) it is necessary to solve a global constrained optimization problem of Eq. (6) or (7), subject to the stability constraint of Eq. (12) and, possibly, the coefficient restrictions of Sec. 3.4. If the compressed sensing application allows for a pre-generated dictionary, then the feedback coefficients can be optimally computed off-line using a variety of constrained optimization algorithms. This case is not examined in this paper.

If the dictionary is generated on-line, the feedback coefficients should also be computed on-line. Section 4.1 proposes a greedy approach assuming unrestricted feedback coefficient values. A simple—and significantly more efficient—extension of this approach is presented in Sec. 4.2 for the case of coefficients restricted as described in Sec. 3.4. Note that these two algorithms make no assumptions on how the dictionary is generated. Thus, in addition to compressive sensing, they can also be used in other applications in which the dictionary is not known in advance.

4.1 Greedy Optimization

The greedy algorithm we develop here is an iterative algorithm that uses the set of coefficients $c_{n-i,m}$ for all $i = 1, \dots, p$ and $i < m \leq n$ to compute the set of coefficients $c_{n-i+1,n+1}$ for all $i = 1, \dots, p$. In a dictionary that is randomly generated at runtime, each iteration of the algorithm would normally be executed right after the generation of frame vector \mathbf{f}_{n+1} .

Specifically, the accumulated error at time n can be expressed as the sum of all the terms in Eq. (5) that include all the frame vectors up to and including \mathbf{f}_n . Using \mathcal{E}_n to denote the accumulated error, and rearranging Eq. (5), it follows that:

$$\mathcal{E}_n = \sum_{k < n-p} e_k \left(\mathbf{f}_k - \sum_{i=1}^p c_{k,k+i} \mathbf{f}_{k+i} \right) + \sum_{l=n-p}^n e_l \left(\mathbf{f}_l - \sum_{i=1}^{n-l} c_{l,l+i} \mathbf{f}_{l+i} \right) \quad (14)$$

$$= \sum_{k < n-p} e_k \mathbf{r}_{k,k+p} + \sum_{l=n-p}^n e_l \mathbf{r}_{l,n}, \quad (15)$$

in which $\mathbf{r}_{l,n} = \mathbf{f}_l - \sum_{i=1}^{n-l} c_{l,l+i} \mathbf{f}_{l+i}$ is the residual error vector after using $\{\mathbf{f}_{l+1}, \dots, \mathbf{f}_n\}$ to compensate for the quantization error of coefficient x_n , corresponding to the frame vector \mathbf{f}_l . It should be noted that $\|\mathbf{r}_{l,l+p}\| = \tilde{c}_l$, as defined in Eq. (8).

Given \mathcal{E}_n , the incremental error $\mathcal{E}_{n+1} - \mathcal{E}_n$ is equal to:

$$\mathcal{E}_{n+1} - \mathcal{E}_n = \sum_{l=n-p+1}^{n+1} e_l (\mathbf{r}_{l,n} - c_{l,n+1} \mathbf{f}_{n+1}) \quad (16)$$

$$= \sum_{i=0}^{p-1} e_{n-i} (\mathbf{r}_{n-i,n} - c_{n-i,n+1} \mathbf{f}_{n+1}) \quad (17)$$

$$= \sum_{i=0}^{p-1} e_{n-i} \mathbf{r}_{n-i,n+1}, \quad (18)$$

in which $\mathbf{r}_{n,n} = \mathbf{f}_n$. The greedy algorithm either uses the additive noise model of quantization to minimize the expected magnitude of the incremental error or the triangle inequality to minimize the upper bound:

$$E\{\|\mathcal{E}_{n+1} - \mathcal{E}_n\|^2\} = \sigma_e^2 \sum_{i=0}^{p-1} \|\mathbf{r}_{n-i,n} - c_{n-i,n+1} \mathbf{f}_{n+1}\|^2, \text{ or} \quad (19)$$

$$\|\mathcal{E}_{n+1} - \mathcal{E}_n\| \leq \frac{\Delta}{2} \sum_{i=0}^{p-1} \|\mathbf{r}_{n-i,n} - c_{n-i,n+1} \mathbf{f}_{n+1}\|. \quad (20)$$

The unconstrained minimization is straightforward by setting $c_{n-i,n+1} = \langle \mathbf{r}_{n-i,n}, \mathbf{f}_{n+1} \rangle / \|\mathbf{f}_{n+1}\|^2$, which projects $\mathbf{r}_{n-i,n}$ onto \mathbf{f}_{n+1} . Unfortunately this solution provides no guarantee that the coefficients are such that the system is stable. Instead, Eq. (19) or (20) should be optimized subject to the stability constraint of Eq. (12). Still, the unconstrained optimization problem can be used to provide lower bounds on the performance of the solution.

The greedy algorithm to compute the Sigma-Delta parameters can be summarized by the following iteration:

1. Randomly generate \mathbf{f}_{n+1} , according to the dictionary and the application requirements. We make no assumptions on how the dictionary is generated.
2. Compute the corresponding feedback coefficients by minimizing Eq. (19) or (20) subject to the stability constraint of Eq. (12).
3. Increase n to $n + 1$ and iterate from Step 1.

Despite the significant reduction of the computational cost, the constrained optimization problem required at every iteration of this algorithm is often too expensive to compute on-line at the rate the frame vectors are generated. Furthermore, the resulting coefficients are arbitrary making their implementation difficult, as described in Sec. 3.4. Fortunately, restricting the coefficient values such that it is possible to implement this system in hardware also makes the optimization problem easier to solve.

4.2 Constrained Coefficient Values

To make the system implementable in hardware, we restrict the feedback coefficients to the set $\{-c, 0, +c\}$, in which c is a positive real number. This makes implementation easier since each of the taps in Fig. 3 can be implemented using an inverter, an open circuit, and a unit-gain buffer, respectively, followed by a gain of c , which can be placed after the adder.

Using restricted coefficients, each of the residuals $\mathbf{r}_{n-i,n+1}$ has smaller magnitude than $\mathbf{r}_{n-i,n}$ only if:

$$\|\mathbf{r}_{n-i,n} - c_{n-i,n+1}\mathbf{f}_{n+1}\| < \|\mathbf{r}_{n-i,n}\|, \quad (21)$$

for either $c_{n-i,n+1} = +c$ or $c_{n-i,n+1} = -c$. This is equivalent to:

$$c < \frac{2|\langle \mathbf{r}_{n-i,n}, \mathbf{f}_{n+1} \rangle|}{\|\mathbf{f}_{n+1}\|^2}, \quad \text{and} \quad (22)$$

$$\text{sign}(c_{n-i,n+1}) = \text{sign}(\langle \mathbf{r}_{n-i,n}, \mathbf{f}_{n+1} \rangle). \quad (23)$$

When this is the case, the feedback tap corresponding to $c_{n-i,n+1}$ is a candidate for a closed circuit or an inverter, depending on the sign of $c_{n-i,n+1}$. Without the stability constraint, each candidate coefficient would be set to $\pm c$ and the algorithm would be finished. Unfortunately, the number of coefficients that can be non-zero is limited by the constraint to:

$$Z = \left\lfloor \frac{s_{\max}}{c} \right\rfloor = \left\lfloor \frac{2(q_{\max} - x_{\max}) + \Delta}{c\Delta} \right\rfloor, \quad (24)$$

in which $\lfloor \cdot \rfloor$ denotes rounding towards zero.

The coefficients finally selected from the candidates are the ones contributing the largest improvement in the magnitude or the power of the residual vectors:

$$\|\mathbf{r}_{n-i,n} - c_{n-i,n+1}\mathbf{f}_{n+1}\|^2 - \|\mathbf{r}_{n-i,n}\|^2, \quad \text{or} \quad (25)$$

$$\|\mathbf{r}_{n-i,n} - c_{n-i,n+1}\mathbf{f}_{n+1}\| - \|\mathbf{r}_{n-i,n}\|, \quad (26)$$

correspondingly, depending on whether the additive noise model or the upper bound is being optimized.

The extension of this optimization if coefficients can take values in the set $\{-Lc, \dots, -c, 0, +c, \dots, +Lc\}$ is straightforward using another greedy algorithm:

1. Initialize all the coefficients to $c_{n-i,n+1} = 0$

2. For all the coefficients $c_{n-i,n+1} \neq \pm Lc$, compute the improvement of changing each coefficient by $\pm c$ using:

$$\|\mathbf{r}_{n-i,n} - (c_{n-i,n+1} \pm c)\mathbf{f}_{n+1}\|^2 - \|\mathbf{r}_{n-i,n} - c_{n-i,n+1}\mathbf{f}_{n+1}\|^2, \text{ or} \quad (27)$$

$$\|\mathbf{r}_{n-i,n} - (c_{n-i,n+1} \pm c)\mathbf{f}_{n+1}\| - \|\mathbf{r}_{n-i,n} - c_{n-i,n+1}\mathbf{f}_{n+1}\|, \quad (28)$$

depending on the model chosen for the optimization.

3. Choose the coefficient $c_{n-i,n+1}$ that contributes the greatest strictly positive improvement and update it to $c_{n-i,n+1} \leftarrow c_{n-i,n+1} \pm c$.

4. Iterate from Step 2, for a total of Z iterations in which Z is the same as Eq. (24), or until no coefficient change contributes a positive improvement in the cost.

Each iteration of the algorithm strictly increases the chosen coefficient in magnitude by c . If it was optimal to decrease a coefficient magnitude by c at some iteration, then the algorithm would have never taken the step of increasing that coefficient magnitude to its present value at a previous iteration. Therefore, if it stops after Q iterations, the algorithm makes maximal use of the stability constraint.

It is straightforward to demonstrate that this greedy algorithm minimizes Eqs. (19) and (20), subject to the restriction that the coefficients take discrete values, as described. It can, therefore, be used in in Step 2 of the algorithm presented in Sec. (4.1). Furthermore, by setting $L = Q$ and taking the limit as $c \rightarrow 0$, this algorithm solves the unrestricted optimization problem, although it is not the most efficient approach to solve the unrestricted problem.

Note that Eq. (22) implies that if the magnitude of the residual error vector $\mathbf{r}_{n-i,n}$ is less than $c\|\mathbf{f}_{n+1}\|/2$ then the residual error \tilde{c}_{n-i} due the quantization of x_{n-i} cannot be reduced further.

5. EXPERIMENTAL PERFORMANCE

This section presents simulation results to evaluate the performance of these algorithms. The random frame used in the simulations is generated as follows. For each frame vector \mathbf{f}_n we compute a finite-support envelope by sampling a continuous-time window of length M , centered at nM/r , in which r is the redundancy of the frame. The envelope is normalized to have unit ℓ_2 norm and modulated by a random i.i.d. ± 1 sequence, with equal probability for the positive and the negative sign. Although a Tukey (cosine tapered) window is used in the results presented, the trends demonstrated in the figures are robust to the choice of window.

5.1 Unrestricted Coefficients

The unconstrained and unrestricted optimization problem of Eq. (19) and (20) establishes a performance baseline for the algorithm. In this case, as described in Sec. 4.1, the optimal choice of coefficients is $c_{n-i,n+1} = \langle \mathbf{r}_{n-i,n}, \mathbf{f}_{n+1} \rangle / \|\mathbf{f}_{n+1}\|^2$.

The results of these unrestricted experiments are plotted in Fig. 4. Specifically, Fig. 4(a) plots the performance of the algorithm described in Sec. 4.1 as a function of the frame redundancy r and the support M of the frame. The figure demonstrates that the performance of the algorithm improves as the redundancy r increases. Modifying the support of the frame, however, does not affect the performance. Figure 4(b) plots the average value of s_n , the sum of the magnitudes of the coefficients in the feedback loop. As the support and the redundancy increases, the system requires a quantizer with higher dynamic range to be stable. Figures 4(c) and (d) plot the same results as (a) and (b), respectively, but for a fixed frame support $M = 64$ and a varying order of the feedback loop p as a function of the support M : $p = 2M, 4M$, and $8M$. The results are plotted against the redundancy r of the frame. The figures demonstrate that the order of compensation has a significant effect on the performance of the Sigma Delta quantizer.

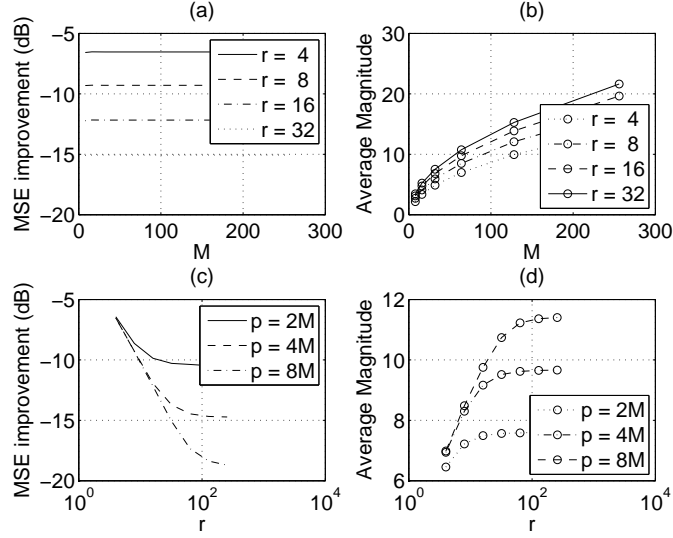


Figure 4. Simulation results for the unconstrained system: (a) Mean squared error improvement for Sigma-Delta of order $p = 8M$, and varying frame redundancy; (b) average sum of the coefficient magnitudes in the Sigma-Delta loop of order $p = 8M$, and varying frame redundancy; (c) mean squared error improvement for a frame of support $M = 64$, and varying Sigma-Delta order p ; (d) average sum of the coefficient magnitudes in the Sigma-Delta loop for a random frame of support $M = 64$, and varying Sigma-Delta order p .

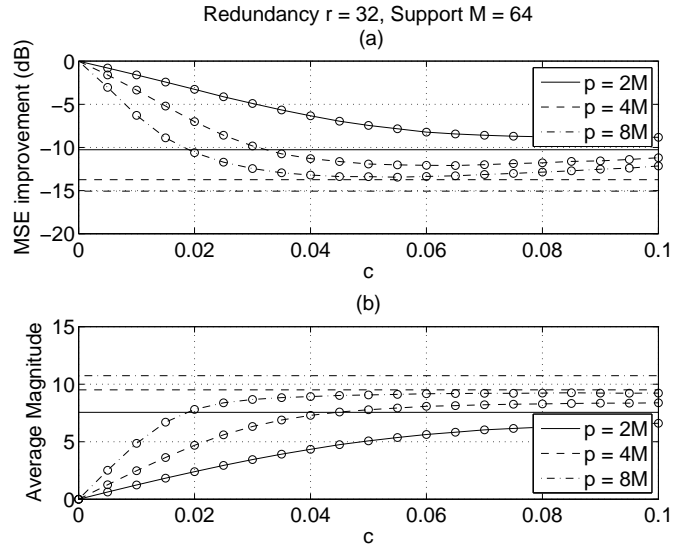


Figure 5. Simulation results assuming feedback coefficients in the set $\{-c, 0, +c\}$. For comparison, the unmarked horizontal lines plot the corresponding performance assuming unrestricted coefficient values. (a) Mean squared error improvement as a function of c , for frame redundancy $r = 32$ and varying Sigma-Delta order p ; (b) average sum of the coefficient magnitudes for the same experimental conditions.

5.2 Restricted Coefficient Values

As described in Sec. 4.2, implementing a system with unrestricted coefficient values is very difficult in practice. The results presented in this section assume the feedback coefficients can only take values in the set $\{-c, 0, +c\}$. Such a system is significantly easier to implement in practice. Under this assumption, two cases are examined. In the first case, the stability condition of Eq. (12) is not imposed on the system. In the second case, the stability

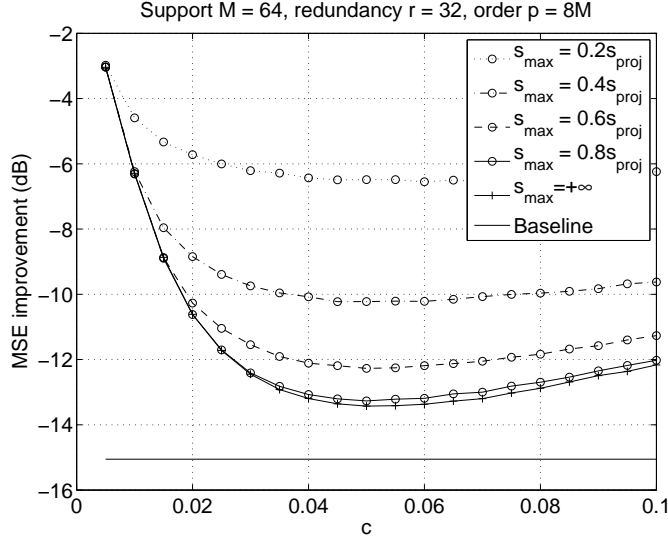


Figure 6. Simulation results assuming the feedback coefficients are restricted in the set $\{-c, 0, +c\}$ and the sum of their magnitude s_n is constrained by s_{\max} . As a baseline, we use the unrestricted and unconstrained problem described in Sec. 4.1. The average sum of coefficient magnitudes from the baseline problem, denoted using s_{proj} is also used as a reference to compute s_{\max} .

condition becomes a design constraint.

Figure 5 demonstrates the performance of the system as a function of the value of the constant c and the compensation order of the feedback loop. The simulation results are plotted—using the lines marked with a circle—for frames with support $M = 64$ and redundancy $r = 32$, although the results are similar for other parameter values. The stability condition of Eq. (12) is not imposed on the system. For comparison, the unmarked horizontal lines in the figures plot the performance of the corresponding unrestricted system. Figure 5(a) shows that with proper choice of c the performance of the system can be roughly within 2dB of the unrestricted system performance. Furthermore, Fig. 5(b) shows that imposing a restriction on the coefficient values significantly reduces the average s_n of the system. Thus, a quantizer with a smaller dynamic range is required to build the system.

Figure 6 demonstrates the effect of enforcing the stability constraint, in addition to restricting the value of feedback coefficients in the set $\{-c, 0, +c\}$. The benchmark to generate this figure is the average s_n of the unrestricted and unconstrained problem, as presented in Fig. 4 and denoted using s_{proj} . The performance for this case is plotted using the straight solid line. The figure presents the performance of the system when s_{\max} is constrained to be within a fraction of s_{proj} , as noted in the legend. For comparison the figure also plots the corresponding values from Fig. 5, i.e., assuming the stability constraint is not enforced.

The plot demonstrates how the stability constraint becomes dominant only as the value of c increases. This is expected since the coefficient magnitude sum cannot be greater than $p \cdot c$. Therefore, for small c the stability constraint is implicitly enforced by the order p of the feedback loop. As the stability constraint dominates, however, the performance of the system decreases. Nevertheless, the results in Fig. 6 demonstrate that a system that enforces a stability constraint can achieve essentially the same performance as a system with constrained coefficient values but without guaranteed stability; see the curve corresponding to $s_{\max} = 0.8s_{\text{proj}}$, for example.

6. CONCLUSIONS

The results in this paper demonstrate the potential of using Sigma-Delta quantization to implement compressive sampling systems, despite the challenges that implementations of such systems pose. We demonstrate that by adapting the coefficients in the Sigma-Delta feedback loop to the dynamically and randomly generated compressive sensing dictionary it is possible to significantly reduce the hardware complexity of the analog part of

the system. Although we do require a high-order Sigma-Delta feedback loop due to the RIP properties of the compressive sensing dictionaries, the order of this loop is significantly lower than the order of a feed-forward analog random projection filter followed by a precision quantizer. The efficient algorithms provided guarantee that the feedback loop is stable and that the quantizer does not exceed its dynamic range.

The goal of this paper was to explore the properties of the compressive sensing dictionary with respect to Sigma-Delta modulation, and therefore the results we demonstrate assume no oversampling of the signal compared to its Nyquist rate. Nevertheless, such oversampling has the potential of further simplifying the system at the expense of a slightly higher sampling rate. Specifically, oversampling increases the coherency of the sampling dictionary and, therefore, can reduce the order of the feedback loop, increase the stability of the system, and reduce the quantization bits required of the quantizer.

Most of the results presented on the trade-offs for the system performance are experimental. Further theoretical study of the trade-offs is necessary. Some of the results presented in this paper also apply in the design of classical Sigma-Delta systems; improved understanding of the trade-offs is helpful even in the design of Sigma-Delta systems for other applications.

ACKNOWLEDGMENTS

This work was supported by the grants DARPA/ONR N66001-06-1-2011 and N00014-06-1-0610, NSF CCF-0431150, NSF DMS-0603606, ONR N00014-06-1-0769 and N00014-06-1-0829, and AFOSR FA9550-04-1-0148.

REFERENCES

1. D. Donoho, "Compressed Sensing," *IEEE Trans. Info. Theory* **52**, pp. 1289–1306, Sept. 2006.
2. E. J. Candès and T. Tao, "Near optimal signal recovery from random projections: Universal encoding strategies?," *IEEE Trans. Info. Theory* **52**, pp. 5406–5425, Dec. 2006.
3. E. J. Candès, J. Romberg, and T. Tao, "Robust Uncertainty Principles: Exact Signal Reconstruction from Highly Incomplete Frequency Information," *IEEE Trans. Info. Theory* **52**, pp. 489–509, Feb. 2006.
4. E. Candès, J. Romberg, and T. Tao, "Stable Signal Recovery from Incomplete and Inaccurate Measurements," *Comm. Pure and Applied Math.* **59**, pp. 1207–1223, Aug. 2006.
5. E. J. Candès, "Compressive sampling," in *Proc. International Congress of Mathematicians*, **3**, pp. 1433–1452, (Madrid, Spain), 2006.
6. S. Kirolos, J. Laska, M. Wakin, M. Duarte, D. Baron, T. Ragheb, Y. Massoud, and R. Baraniuk, "Analog-to-information conversion via random demodulation," in *Proc. IEEE Dallas Circuits and Systems Workshop (DCAS)*, (Dallas, TX), Oct. 2006.
7. J. N. Laska, S. Kirolos, Y. Massoud, R. G. Baraniuk, A. C. Gilbert, M. Iwen, and M. J. Strauss, "Random sampling for analog-to-information conversion of wideband signals," in *Proc. IEEE Dallas Circuits and Systems Workshop (DCAS)*, (Dallas, TX), Oct. 2006.
8. J. N. Laska, S. Kirolos, M. F. Duarte, T. Ragheb, R. G. Baraniuk, and Y. Massoud, "Theory and implementation of an analog-to-information conversion using random demodulation," in *Proc. IEEE Int. Symposium on Circuits and Systems (ISCAS)*, (New Orleans, LA), May 2007.
9. J. Tropp, M. B. Wakin, M. F. Duarte, D. Baron, and R. G. Baraniuk, "Random filters for compressive sampling and reconstruction," in *IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, **III**, pp. 872–875, (Toulouse, France), May 2006.
10. R. Gray, "Oversampled Sigma-Delta modulation," *IEEE Trans. Comm.* **35**, pp. 481–489, May 1987.
11. P. Boufounos, *Quantization and Erasures in Frame Representations*. D.Sc. Thesis, MIT EECS, Cambridge, MA, Jan. 2006.
12. H. Bolcskei and F. Hlawatsch, "Noise reduction in oversampled filter banks using predictive quantization," *IEEE Trans. Info. Theory* **47**, pp. 155–172, Jan. 2001.
13. J. J. Benedetto, A. M. Powell, and O. Yilmaz, "Sigma-delta quantization and finite frames," *IEEE Trans. Info. Theory* **52**, pp. 1990–2005, May 2006.
14. P. Boufounos and A. V. Oppenheim, "Quantization noise shaping on arbitrary frame expansions," *EURASIP Journal on Applied Signal Processing, Special issue on Frames and Overcomplete Representations in Signal Processing, Communications, and Information Theory* **2006**, pp. Article ID 53807, 12 pages, DOI:10.1155/ASP/2006/53807, 2006.
15. J. Tropp and A. C. Gilbert, "Signal recovery from partial information via orthogonal matching pursuit," Apr. 2005. Preprint.
16. J. J. Benedetto, O. Yilmaz, and A. M. Powell, "Sigma-Delta quantization and finite frames," in *Proceedings of IEEE ICASSP 2004*, IEEE, (Montreal, Canada), May 2004.