On Incremental Sigma-Delta Modulation with Optimal Filtering

Sam Kavusi, Hossein Kakavand, and Abbas El Gamal*
Department of Electrical Engineering, Stanford University, Stanford, CA 94305
{skavusi, hossein, abbas}@stanford.edu

March 26, 2005

Abstract

The paper presents a quantization-theoretic framework for studying incremental $\Sigma\Delta$ quantization systems. The framework makes it possible to efficiently compute the quantization intervals and hence the transfer function of the quantizer, and to determine the mean square error (MSE) and maximum error for the optimal and conventional linear filters for first and second order incremental $\Sigma\Delta$ modulators. The results show that the optimal filter can significantly outperfom conventional linear filters in terms of both MSE and maximum error. The performance of conventional $\Sigma\Delta$ quantizers is then compared to that of incremental $\Sigma\Delta$ with optimal filtering for bandlimited signals. It is shown that incremental $\Sigma\Delta$ can outperform the conventional approach in terms of signal to noise+distortion ratio (SNDR) and the characteristics of the power spectral density (PSD). The framework is also used to provide a simpler and more intuitive derivation of the Zoomer algorithm.

Index Terms: Sigma-Delta ($\Sigma\Delta$), incremental A/D converter, optimal filter, $\Sigma\Delta$ transfer function, time-domain analysis.

1 Introduction

$\Sigma\Delta$ quantization systems, also known as scalar predictive quantizers or oversampling Analog-to-Digital converters, are widely used in audio systems [1, 2] and communication systems [3, 4, 5]. These systems can achieve very large dynamic range without the need for precise matching of circuit components. This is especially attractive for implementation in scaled technologies where transistors are fast but not very accurate. They are also among the most power efficient ADC architectures [4].

Exact analysis of $\Sigma\Delta$ quantizers is difficult due to their highly nonlinear structure. As a result, optimal decoding (filtering and decimation) algorithms are generally not known, although there has been much work on decoding schemes [6, 7, 8]. Conventionally, linear models with quantization noise modeled as additive white noise have been used [9]. This noise model, however, was shown to be quite inaccurate for coarse quantization. In [11], Candy derived a more accurate approximation of the noise power spectral density showing that it is not white. Subsequently, Gray [6] derived the exact spectrum and determined the optimal MSE linear filter for constant inputs. Gray's framework was later used in multiple studies (e.g., [12]). This type of exact analysis, however, is too complex for most practical $\Sigma\Delta$ systems and the conventional linear method is still used and then verified and tweaked using time-domain simulations [9, 10].

Recently, $\Sigma\Delta$ quantization systems have become popular in sensing applications such as imaging [13, 14, 15] and accurate temperature sensing [16]. In such applications, the input signal is very slowly varying with time, and as a result can be viewed as constant over the conversion period. Of particular interest is the special case of *incremental* $\Sigma\Delta$ [17], where the integrator is reset prior to each input signal conversion. In this case, the optimal filter with respect to the mean-squared error criterion denoted by Zoomer was found by Hein et al. [18]. The approach used in [18], however, does not naturally lead to full characterization of the quantizer

*This work was partially supported under DARPA Microsystems Technology Office Award No. N66001-02-1-8940.

transfer function or the determination of the MSE or maximum error, especially for second and higher order $\Sigma\Delta$ modulators. Knowledge of the exact transfer function of the $\Sigma\Delta$ quantization system can enable the system designer to achieve the system requirements with a lower oversampling ratio. Moreover, as noted by Markus et al. [17], data converters with high absolute accuracy are often required in instrumentation and measurement. In [17], bounds on the maximum error of incremental $\Sigma\Delta$ using some conventional linear filters was derived, but there is still a need to derive the maximum error of the optimal filter.

In this paper, we introduce a time-domain, quantization-theoretic framework for studying incremental $\Sigma\Delta$ quantization systems. The framework allows us to determine the quantization intervals and hence the transfer function of the quantizer, and to precisely determine the MSE and maximum error for the optimal filter. For constant inputs, we demonstrate significant improvements in both MSE and maximum error over conventional linear filters [17]. These results imply that an incremental $\Sigma\Delta$ with optimal filtering can achieve the same performance as that of conventional $\Sigma\Delta$ systems at lower oversampling ratio and hence lower power consumption. We show that incremental $\Sigma\Delta$ can outperform the conventional approach in terms of SNDR in addition to not having the idle tones artifacts of conventional $\Sigma\Delta$. Using our framework we are also able to compare the performance of conventional $\Sigma\Delta$ quantizers to that of incremental $\Sigma\Delta$ with optimal filtering for bandlimited input signals. We also use our framework to provide a simpler and more intuitive proof of the optimality of the Zoomer algorithm [18].

In the following section, we introduce our framework and use it to study first-order $\Sigma\Delta$ quantization systems. In Section 3, we extend our results to incremental second-order $\Sigma\Delta$ quantization systems. In each case, we compare the performance of the optimal filter to that of linear filters. In Section 4, we use our results to compare the performance of the incremental $\Sigma\Delta$ modulator using optimal filtering to conventional $\Sigma\Delta$ systems using linear filtering for sinusoidal inputs. We also show that incremental $\Sigma\Delta$ does not suffer from the problem of idle tones.

2 Incremental first-order $\Sigma\Delta$

A block diagram of a $\Sigma\Delta$ quantization system with constant input is depicted in Figure 1. The figure also provides the correspondance between the terminology used in the circuit design literature and the quantization-theoretic literature [20]. The $\Sigma\Delta$ modulator itself corresponds to the *encoder* in the quantization-theoretic setting. It outputs a binary sequence of length m that corresponds to the quantization interval that the input x belongs to. The set of such binary sequences is the *index set* in the quantization-theoretic setting. The filter, which corresponds to the *decoder*, provides an estimate $\hat{x}$ of the input signal x. Given an index, the optimal decoder outputs the *centroid*, i.e., the midpoint, of the quantization interval corresponding to the index.

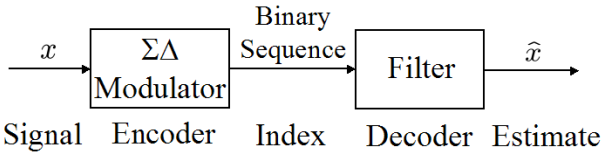

Figure 1: Block diagram of a $\Sigma\Delta$ quantization system with constant input. The terms under the figure provide the corresponding object names in quantization theory.

In this section we study the discrete time first-order $\Sigma\Delta$ modulator (see Figure 2). For simplicity we focus on the single-bit case. However, our results can be easily extended to multi-bit $\Sigma\Delta$ modulators. Without loss of generality, we assume that the comparator threshold value is 1 and the input signal $x \in [0, 1]$. The input to the modulator is integrated, thus creating a ramp with a slope proportional to the constant input. At time $n = 1, 2, \ldots, m$, the ramp sample $u(n)$ is compared to the threshold value of 1 and the output $b(n) = 1$ if $u(n) > 1$, otherwise $b(n) = 0$. If $b(n) = 1$, a one is subtracted from the ramp value. Thus in effect the

modulator is predicting the ramp value by counting the number of ones and subtracting off its prediction via the feedback loop. This operation is illustrated in Figure 3, where the output of the integrator $u(n)$ is plotted versus time for a fixed input value.

The output of the integrator is thus given by

$$u(n) = x - b(n-1) + u(n-1) = nx - \sum_{0}^{n-1} b(i). \tag{1}$$

We shall assume that the integrator is reset to zero at the beginning of conversion, i.e. $u(0) = 0$, which is the case in incremental $\Sigma\Delta$.

In conventional $\Sigma\Delta$ quantization systems, the binary sequence generated by the modulator is decoded using a linear filter, such as a counter or a triangular filter, to produce an estimate of the input signal [17]. Such linear filters, however, are not optimal and result in significantly higher distortion compared to the optimal filter. The optimal filter in the form of "Zoomer" implementation was derived in [18]. In [19] McIlrath developed a nonlinear iterative filter that achieves significantly better performance compared to linear filters.

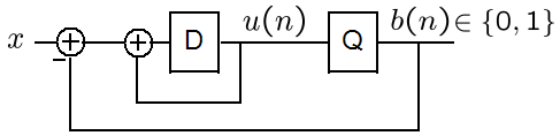

Figure 2: Block diagram for the first-order $\Sigma\Delta$ modulator. D refers to the delay element and Q refers to the one-bit quantizer.

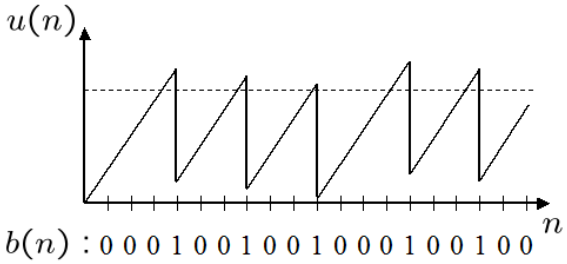

Figure 3: First-order sample integrator output and the corresponding sequence.

We are now ready to introduce our framework for studying incremental $\Sigma\Delta$ quantization systems. Note that to completely characterize a quantization system, one needs to determine the quantization intervals induced by its encoder. Equivalently, one needs to determine the *transition points* between consecutive quantization intervals.

The $\Sigma\Delta$ modulator can be viewed to be performing a sequence of *effective comparisons*, $u(n) \gtrless 1$ which are equivalent to $x \gtrless (1 + \sum_0^{n-1} b(i))/n$. Clearly any encoder that produces bits corresponding to the same effective comparisons is equivalent to the $\Sigma\Delta$ modulator. To find the transition points of the first-order $\Sigma\Delta$ modulator, we define the following equivalent encoder. Referring to (1), it is clear that the sequence of comparisons $u(n) \gtrless 1$, $1 \leq n \leq m$, is the same as the sequence of comparisons of the *predictor* $1 + \sum_0^{n-1} b(i)$ to the *equivalent ramp* nx, which is the line segment from the origin of slope x. Therefore, if the modulator is replaced by an encoder that performs the comparisons to the equivalent ramp, its output sequence would

be identical to that of the modulator for all input values. Note that this equivalence holds for all inputs $x \in [0, 1]$. Figure 4 depicts this equivalence graphically; the dotted line is the integrator output, the solid line is the equivalent ramp, and the dashed line is the predictor.

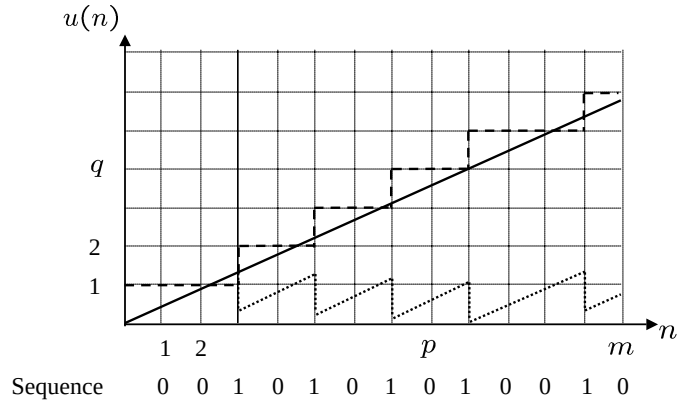

Figure 4: The equivalent ramp and it's corresponding predictor.

To find the transition points using the above equivalence, consider the square lattice shown in Figure 5

$$L_m = \left\{(p, q) : p, q \in \mathbb{Z}^+, 1 \leq q \leq p \leq m\right\}.$$

We now show that the set of transition points is simply the set of slopes of the equivalent ramps that pass through points of the square lattice, which we denote as

$$S_m = \left\{ \frac{q}{p} : (p, q) \in L_m \right\}.$$

The elements of S_m form the Farey sequence [21] of order m.

To show that S_m is the set of transition points, we first show that any element of S_m is a transition point. Consider the equivalent ramp of slope q/p. It is not difficult to see that any equivalent ramp of higher slope, i.e., any signal value $x > q/p$, generates a different sequence from q/p itself. Therefore, any $q/p \in S_m$ is a transition point.

To show that any transition point y belongs to S_m, we need to show that $y = q/p$ for some integers $1 \leq q \leq p \leq m$. Since y is a transition point, the sequence corresponding to y and $y + \delta$, for any $\delta > 0$, must differ in at least one sample time. Let such sample time be $1 \leq p \leq m$, then there must be an integer $q \leq p$ such that the predictor value is q at time p, which implies that $y = q/p \in S_m$.

The transition points can be computed simply by computing the elements of S_m or by parsing the Farey or Stern-Brocot tree [21]. The worst case running time is $\mathcal{O}(m^2)$.

The optimal filter produces the centroid of the quantization region corresponding to the sequence generated by the modulator. For example for the sequence 00101, which corresponds to $x \in [2/5, 1/2]$, the optimal filter produces the estimate $\hat{x} = 9/20$.

The equivalent encoder framework provides a simple and intuitive way to prove the optimality of the $\mathcal{O}(m)$ Zoomer algorithm [18]. To prove the optimality of this implementation, note that a filter is optimal if given any index sequence produced by the modulator, it outputs the centroid of the quantization interval to which the input signal belongs. So, for any input $x \in [0, 1]$, we need to show that the resulting UB and LB are the transition points for the quantization interval of x. We show this by induction. Clearly, this is the case for $m = 1$. Assume that this is also the case up to $m - 1$, i.e., for $m - 1$ samples, UB and LB are the transition points of the quantization interval of x. Since any transition point is a point on the square lattice, $UB - LB \leq 1/(m - 1) < 2/m$. Therefore, at the mth sampling time, there can be only one new transition

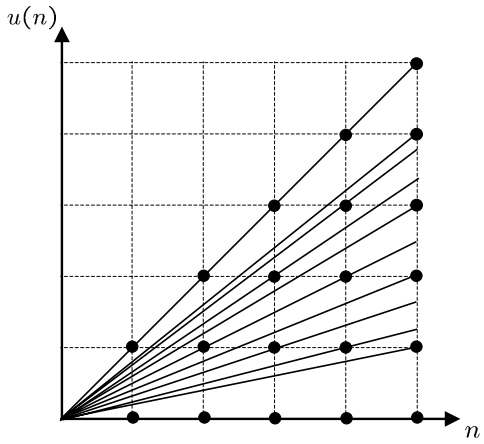

Figure 5: Lattice for first-order ΣΔ.

Algorithm 1 Optimal Filtering Algorithm(Zoomer Implementation)

```
begin
SEQ ← Modulator generated sequence
UB ← 1, LB ← 0
Predictor ← 1
for p = 1 : m do
  if SEQ(p) = 1 then
    LB ← max{LB, Predictor/p}
    Predictor ← Predictor + 1
  else
    UB ← min{UB, Predictor/p}
  end if
end for
return (UB + LB)/2
end
```

point between UB and LB. Let p be the predictor value at time m. Note that if $p/m \notin (LB, UB)$, then the mth bit does not change the quantization interval for x from that at time $m-1$ and hence (LB, UB) remains as the quantization interval for x. On the other hand, if $p/m \in (LB, UB)$, then it is a new transition point and the quantization interval specified by (LB, UB) is split into two new quantization intervals with x belonging to one of them. Specifically, if $SEQ(m) = 1$, then $LB \leftarrow p/m$ and the new quantization interval for x becomes $(p/m, UB)$, and if $SEQ(m) = 0$, $UB \leftarrow p/m$ and the new quantization interval for x becomes $(LB, p/m)$. In both cases the end points of the interval are transition points. This proves that algorithm 1 is indeed the optimal filtering algorithm.

The above procedures can be readily used to quantify the MSE of the optimal filter. In order to compare the MSE of the optimal filter to conventional linear filters, we need to find the correspondence of estimates produced by each filter to the quantization intervals. Figure 6 illustrates this correspondence for the counter (i.e., rectangular), triangular and MSE optimal linear [6] filters. The figure depicts the quantization intervals and their centroids, which are produced by the optimal filter, for $m = 4$, and the estimates for each filter and their correspondence to the quantization intervals. The figure clearly illustrates the source of suboptimality of linear filters as an interval is not necessarily mapped into its centroid and may in fact be mapped into an estimate that is outside the interval itself.

The Mean Square Error (MSE) is a measure of average noise power, and implies the overall accuracy of

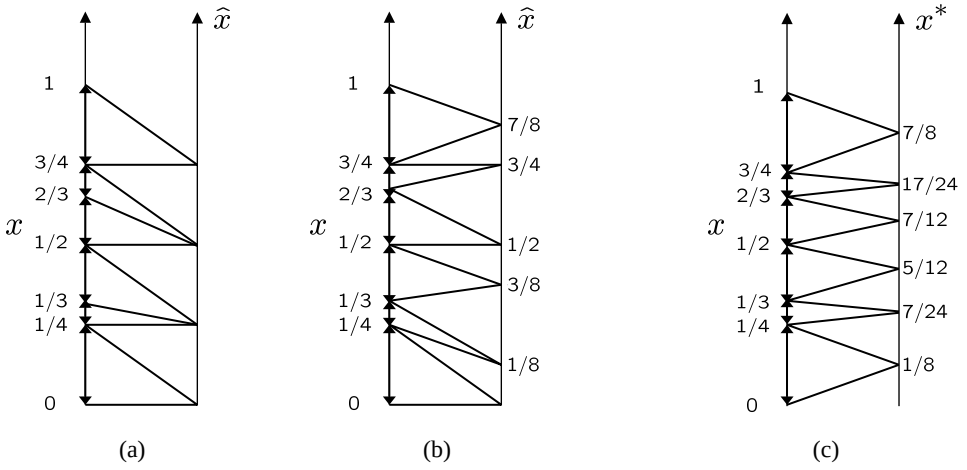

Figure 6: Quantization intervals and corresponding estimations for $m = 4$ using (a) counter filter (b) triangular filter and (c) optimal filter.

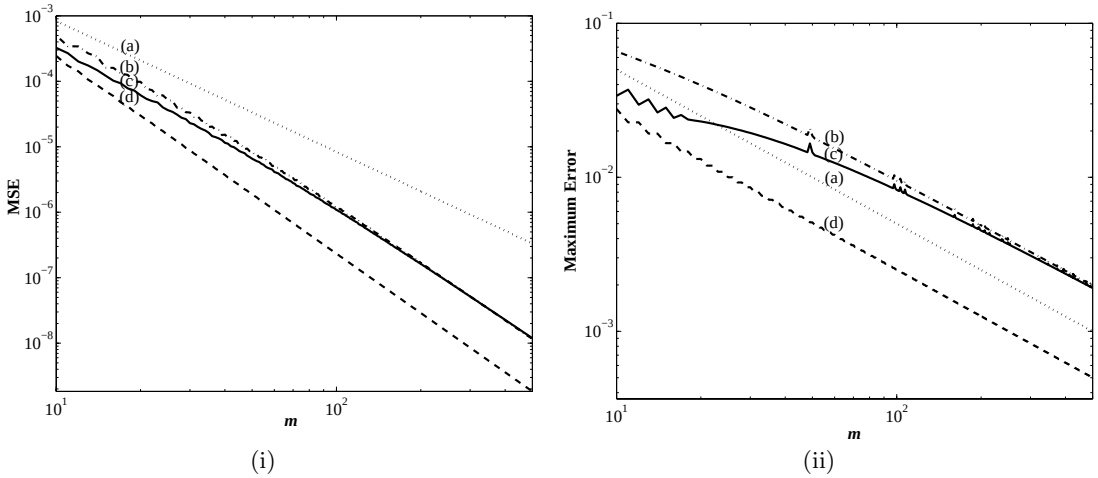

Figure 7: (i) MSE and (ii) Maximum error behavior as a function of oversampling ratio m, with gain/offset correction using (a) counter filter (b) triangular filter (c) optimal linear filter and (d) optimal filter.

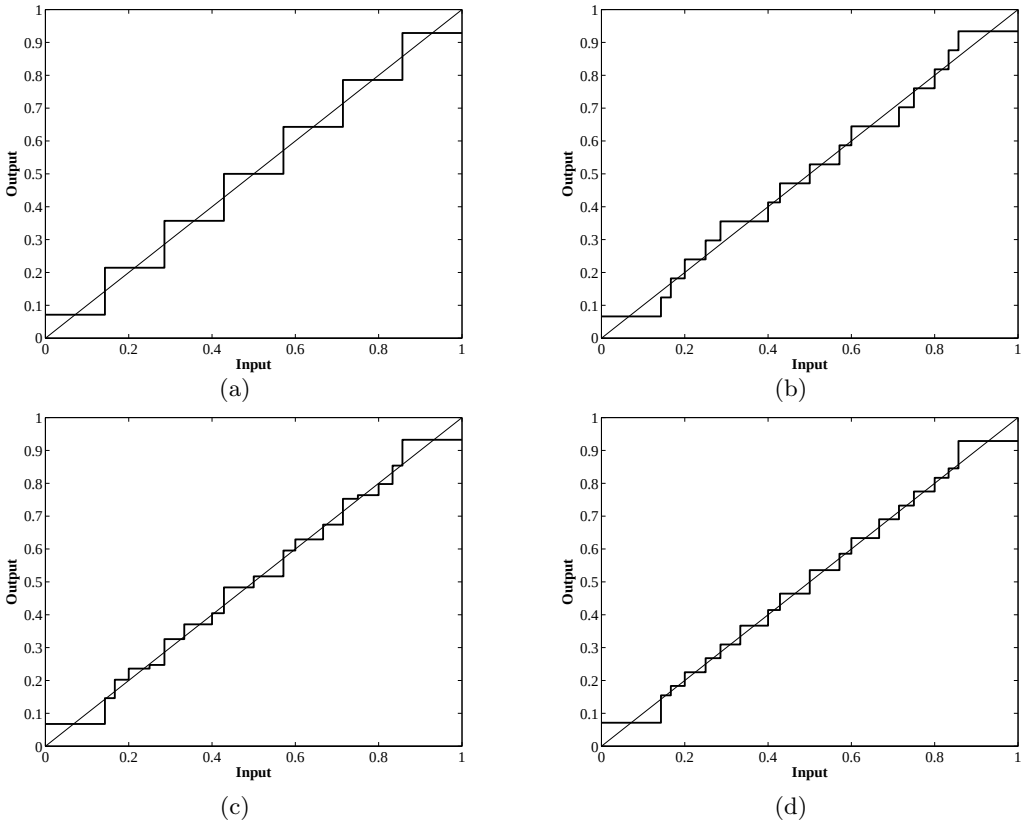

Figure 8: Transfer functions using (a) counter filter after gain/offset correction (b) triangular filter after gain/offset correction (c) optimal linear filter after gain/offset correction and (d) optimal filter, for $m = 7$.

the quantization system. Figure 7(i) compares the MSE for the optimal filter to the MSEs for the counter, triangular and optimal linear [6] filters. In plotting the MSE for each linear filter we corrected for its systematic offset and gain biases using the minimum MSE affine transformation found by computing the coefficients a and b that minimize

$$D = E\big[(x - \hat{x})^2\big] = \sum_{i=1}^{n} (x_i^* - a\hat{x}_i - b)^2 \Delta_i,$$

where $\{x_i^*\}_{i=1}^n$ and $\{\Delta_i\}_{i=1}^n$ are the centroids and lengths of the quantization intervals, respectively, and $\{\hat{x}_i\}_{i=1}^n$ are the estimates corresponding to the quantization intervals provided by the linear filter. Note that the optimal filter does not suffer from any systematic offset or gain biases. Also note that the outputs of all the filters considered here converge to the input value. Their convergence rates, however, differ. As expected, the counter has the worst convergence rate. For small m, the triangular filter is worse than the optimal linear filter, but as m increases, their MSEs become very close. Note that for example, in order to achieve a performance similar to a 10-bit uniform ADC one can use the optimal filter and operate the $\Sigma\Delta$ quantization system with half the speed of the case when a conventional filter is used; this implies at least a factor of 2 power saving in the modulator of the first-order $\Sigma\Delta$ quantization system.

Our framework can also be used to completely characterize the transfer function for first-order $\Sigma\Delta$ quantization systems, which can provide further insight into the behavior of various filters. As an example, in Figure 8 we plot the transfer functions for first-order $\Sigma\Delta$ quantizers using the optimal filter, the optimal linear filter and the rectangular filter for $m = 7$.

Note that the large distortion at the two ends of the $\Sigma\Delta$ quantization system can be intolerable in some

applications. For example, in imaging applications [13], it is very important to achieve low distortion at low input signal values (corresponding to low light).

Maximum error is another measure of distortion, which is very important when absolute accuracy is needed. If the application allows the designer to adjust the input range to $x \in [1/m, 1-1/m]$, instead of $[0,1]$, the large distortion at the two ends of the range (see Figure 8) can be avoided, resulting in a factor of two reduction in absolute error with very small decrease in the input range of $2/m$. Figure 7(ii) plots the maximum error in the range of $x \in [1/m, 1-1/m]$ for different filters versus the oversampling ratio m. As can be seen from the plots, the optimal filter can achieve significantly lower absolute error than linear filters for the same oversampling ratio. Alternatively, it can achieve the same absolute error at a much lower oversampling ratio. For example, to achieve an absolute error equal to that of a uniform 10-bit quantizer, a $\Sigma\Delta$ quantizer with optimal filtering requires half the oversampling ratio of a counter filter. It is also interesting to note that the counter outperforms the optimal MSE linear filter in terms of absolute accuracy.

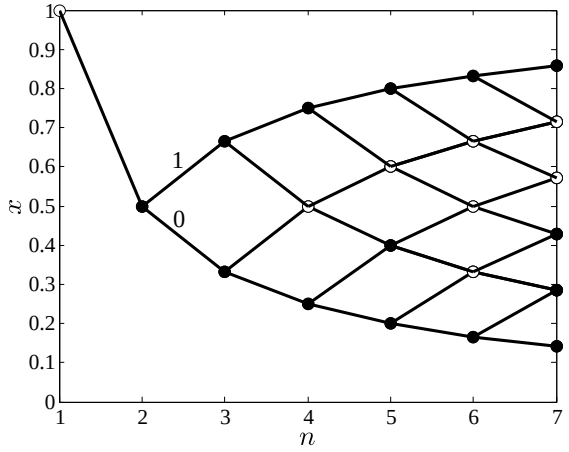

Figure 9: Directed acyclic graph showing quantization in first-order $\Sigma\Delta$ quantizers.

Another consequence of our framework is the ability to characterize the Directed Acyclic Graph (DAG) of the first-order $\Sigma\Delta$ encoder, i.e., the graph representing the sequence of "larger/smaller" comparisons effectively performed by the encoder. This graph may help in studying the redundancy in sequences produced by the encoder. It is useful for finding transition points for higher order $\Sigma\Delta$ quantizers, as we shall see in the next section. The DAG can be found using $\mathcal{O}(m^2)$ algorithm 2.

Figure 9 plots the DAG for $m = 7$. The horizontal axis corresponds to the sampling times and the vertical axis corresponds to the comparison threshold. The nodes correspond to the effective comparisons while the edges correspond to the outcomes of the comparisons (an upper edge corresponds to a 1, while a lower edge corresponds to 0). For example, the path defined by 011101 corresponds to all $3/4 < x < 4/5$, and provides the following information about the input: $x < 1,\ x > 1/2,\ x > 2/3,\ x > 3/4,\ x < 4/5,\ x > 2/3$. Note that there are two types of nodes, a solid node corresponding to a comparison where the outcome is unknown, and a hollow node corresponding to a comparison with a known outcome given the previous bits. To explain this difference, note that a path leading to a node defines an upper and a lower bound, UB and LB, on the range of input signals that correspond to this path. If the predictor corresponding to the node lies within the (LB, UB) interval, then the comparison has an unknown outcome. On the other hand, if the comparison threshold lies outside the (LB, UB) interval, then the outcome of the comparison is already known. For example, consider hollow node $(4, 0.5)$. The paths leading to the node correspond to the input ranges $(0.5, 0.66)$ and $(0.33, 0.5)$. In either case, the outcome of the comparison is known.

As explained earlier, given m comparisons, the first-order $\Sigma\Delta$ encoder generates $\mathcal{O}(m^2)$ quantization intervals, which is rather small compared to the possible 2^m quantization intervals. Study of the DAG

Algorithm 2 Generating the first-order $\Sigma\Delta$ Directed Acyclic Graph

```
begin
SEQ ← [0]
UB ← 1, LB ← 0
q ← 1, s ← 1
mark (q, s) as a hollow node on the graph
q ← 2
CALL DAG(SEQ, LB, UB, q)
end

FUNCTION DAG(SEQ, LB, UB, q)
if q ≤ m then
  Predictor ← Σ_{i=1}^{q-1} SEQ(i) + 1
  s ← Predictor/q
  if LB < s < UB then
    mark (q, s) as a solid node on the graph
    CALL DAG([SEQ; 1], s, UB, q + 1)
    CALL DAG([SEQ; 0], LB, s, q + 1)
  else if s ≥ UB then
    mark (q, s) as a hollow node on the graph
    CALL DAG([SEQ; 0], LB, UB, q + 1)
  else
    mark (q, s) as a hollow node on the graph
    CALL DAG([SEQ; 1], LB, UB, q + 1)
  end if
end if
```

clarifies this difference. The hollow nodes in the DAG do not create transition points and therefore do not add to the number of quantization intervals. However, it is expected that this redundancy in the effective comparisons improves the robustness of the system to errors. Higher order and multi-bit $\Sigma\Delta$ modulators are usually used to increase the number of quantization intervals and therefore achieve improved error. In the next section we extend the framework to second-order incremental $\Sigma\Delta$.

3 Second-order Incremental $\Sigma\Delta$

In this section we study the discrete time second-order $\Sigma\Delta$ quantizer. We focus on the second-order modulator depicted in Figure 10. Our results, however, can be easily extended to a much broader set of architectures. It consists of two integrators followed by a comparator, the output of which is fed back to the input of both integrators. We assume that the comparator threshold is 0 and the input signal $x \in [-1, 1]$. Note that unlike the first-order case the output of the comparator $b(n) = \text{sgn}(u(n)) \in \{-1, 1\}$. We also assume zero initial states, i.e., $u(0) = v(0) = 0$, corresponding to an incremental second-order modulator. Note that the first two bits generated by the modulator are 1 and -1, respectively, independent of the input. To achieve good performance, the modulator gain parameters a_1 and a_2 are chosen so that the output of the second integrator $v(n)$ is as close to 0 as possible for all x and $1 \leq n \leq m$. In the following discussion we assume a conventional choice of $a_1 = 1/2,\ a_2 = 2$.

The outputs of the two integrators (the states) are given by

$$\begin{aligned} u(n) &= \frac{1}{2}\left(x - b(n)\right) + u(n-1), \\ v(n) &= 2\left(u(n-1) - b(n-1)\right) + v(n-1). \end{aligned} \tag{2}$$

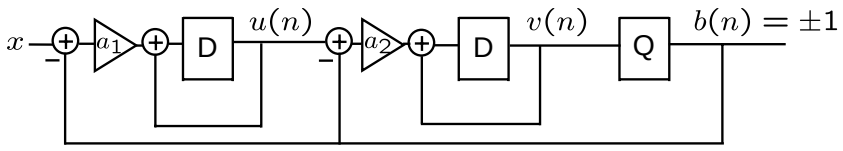

Figure 10: Block diagram for the second-order ΣΔ modulator. D refers to the delay element and Q refers to the one-bit quantizer.

which results in

$$v(n) = xn(n-1)/2 - \sum_{1}^{n-1}(n-i+1)b(i). \tag{3}$$

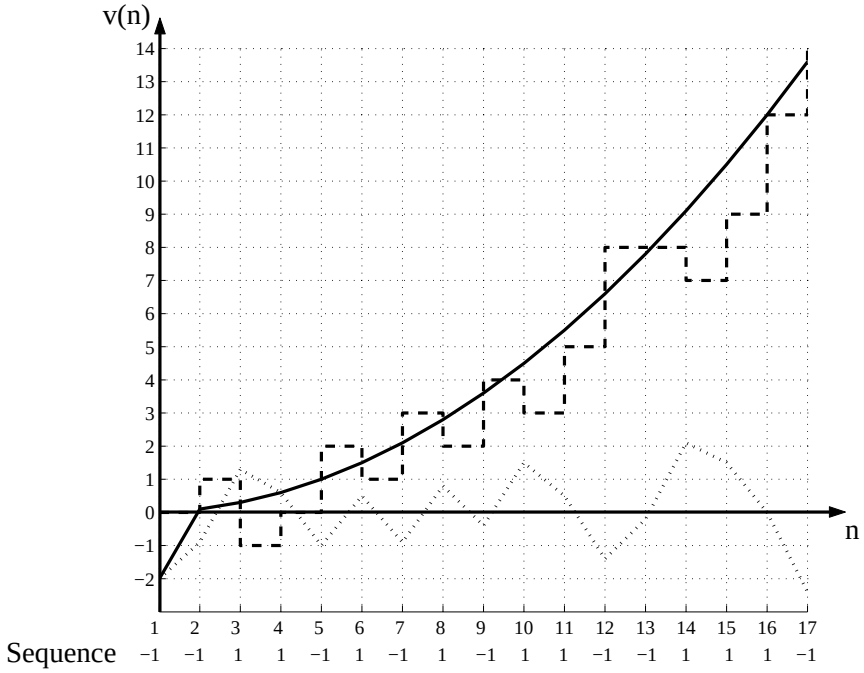

Figure 11: The equivalent ramp and its corresponding predictor.

Similar to the first-order case we can characterize the second-order ΣΔ quantization system by determining the quantization intervals induced by the encoder. This can be done by finding the transition points using the following equivalent encoder. Referring to (3) the sequence of comparisons $v(n) \gtrless 0$, $1 \le n \le m$ is the same as the sequence of comparisons of the predictor $\sum_{i=1}^{n-1}(n-i+1)b(i)$ to the equivalent ramp $xn(n-1)/2$, which is quadratic in n. They both represent the same effective comparisons, $x \gtrless 2(\sum_{i=1}^{n-1}(n-i+1)b(i))/(n^2-n)$. Therefore the output of the modulator is identical to the output of an encoder that performs comparisons between the predictor and the equivalent ramp. Figure 11 depicts this equivalence graphically; the dotted line is the integrator waveform, the solid line is the equivalent ramp, and the dashed line is the predictor.

The optimal filtering algorithm for the second-order modulator is essentially any filter that produces the centroid corresponding to the quantization interval. Zoomer implementation is the same as Algorithm 1 with the updating of the predictor replaced with that for the second-order predictor.

At any sample time, n, the set of possible values the second-order predictor can assume consists of every

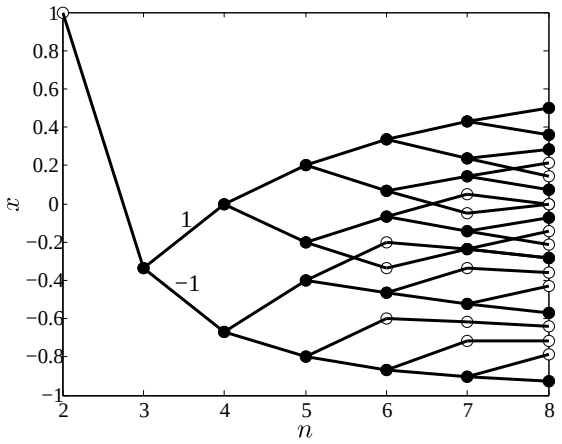

Figure 12: Directed graph showing quantization in the second-order $\Sigma\Delta$ quantizer.

other integer in the interval $[-(n-1)(n-2)/2+2, (n-1)(n-2)/2]$ on the square lattice as depicted in Figure 11. However, unlike the first-order case, not all such integers correspond to transition points, and therefore, to compute the set of transition points, we find and parse the directed acyclic graph of the second-order $\Sigma\Delta$ modulator.

To find the DAG for the second-order modulator, we use Algorithm 3, which is similar to Algorithm 2. We replace the predictor and the effective comparison for the first-order modulator in Algorithm 2 by that for the second-order, taking into consideration the fact that the second-order modulator produces $+1$s and -1s instead of 0s and 1s. Note that in Algorithm 3 q starts from 2 because of the two delay elements before the quantizer. Figure 12 plots the second-order DAG for $m = 8$. For example, the path defined by $1, -1, -1, -1, 1, -1, -1, -1$ corresponds to all $-8/10 < x < -2/3$, and provides the following information about the input: $x \geq 0,\ x < 1,\ x < -1/3,\ x < -2/3,\ x \geq -4/5,\ x < -3/5,\ x < -13/21,\ x < -9/14$. Again there are two types of nodes, a solid node corresponding to a comparison where the outcome is unknown and a hollow node corresponding to a comparison with a known outcome given the previous bits. The solid nodes correspond to transition points. Figure 13 plots all the transition points and their corresponding equivalent ramps for $m = 7$.

As in the first-order case, the derivation of the transition points can be used to determine the transfer function of the second-order quantizer. To compare the performance of filters such as the Kaiser or the natural linear filter [17] to that of the optimal filter, in Figure 14 we plot the transfer functions using these different filters. Note that the linear filters considered are again biased and we have corrected for the bias as before. Although the outputs of the linear filters converge to the correct input, they perform poorly compared to the optimal filter. In fact, as can be seen in Figure 14(b), the transfer function of the Kaiser filter is not even monotonic in the input. Also note that there is a significant systematic gain error in the natural filter shown in 14(a). Also in Figure 14 there is a large quantization interval at the high end for all the filters considered. This large interval is due to the fact that the first two bits generated by the modulator are 1 and -1 independent of the input and results in an asymmetric transfer function. Since this is a systematic problem that causes large distortions, in what follows we shall eliminate this large interval.

The transition points can be used to quantify the MSE and maximum error of the optimal filter and to compare it to that of conventional filters. Figure 15 plots the MSE and maximum error of the Kaiser low pass filter with $\beta = 2.5$, the natural filter and the optimal filter as a function of m. Note that the performance gain of the optimal filter over the non-optimal filters for second-order $\Sigma\Delta$ quantizer is more than that of the first-order. For example, in order to achieve performance comparable to a 10-bit uniform quantizer, a second-order $\Sigma\Delta$ quantizer with optimal filtering can be operated at 1/5th the speed of that using a conventional filter, which implies over a factor of 5 reduction in power consumption.

Algorithm 3 Generating the second-order ΣΔ Directed Acyclic Graph

```
begin
SEQ ← [1; −1]
UB ← 1, LB ← 0
q ← 2, s ← 1
mark (q, s) as a hollow node on the graph
q ← 3
CALL DAG(SEQ, LB, UB, q)
end

FUNCTION DAG(SEQ, LB, UB, q)
if q ≤ m then
   Predictor ← Σ_{i=1}^{q−1} (q − i + 1)SEQ(i)
   s ← Predictor/q(q − 1)
   if LB < s < UB then
      mark (q, s) as a solid node on the graph
      CALL DAG([SEQ; 1], s, UB, q + 1)
      CALL DAG([SEQ; −1], LB, s, q + 1)
   else if s ≥ UB then
      mark (q, s) as a hollow node on the graph
      CALL DAG([SEQ; −1], LB, UB, q + 1)
   else
      mark (q, s) as a hollow node on the graph
      CALL DAG([SEQ; 1], LB, UB, q + 1)
   end if
end if
```

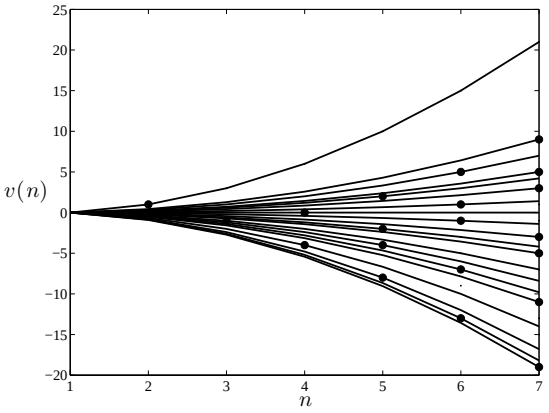

Figure 13: second-order transition points and their corresponding ramps

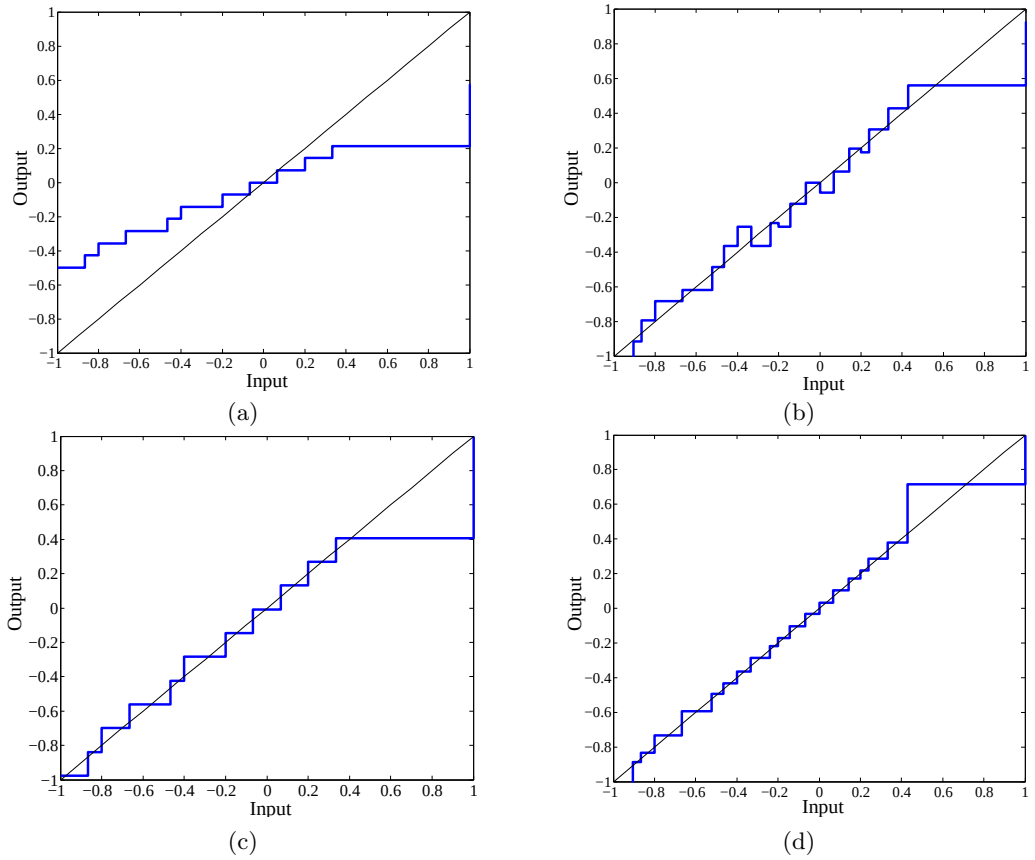

Figure 14: Transfer functions using (a) natural linear filter without gain/offset correction (b) Kaiser filter with $\beta = 2.5$ with gain/offset correction (c) natural linear filter with gain/offset correction and (d) optimal filter, for $m = 7$.

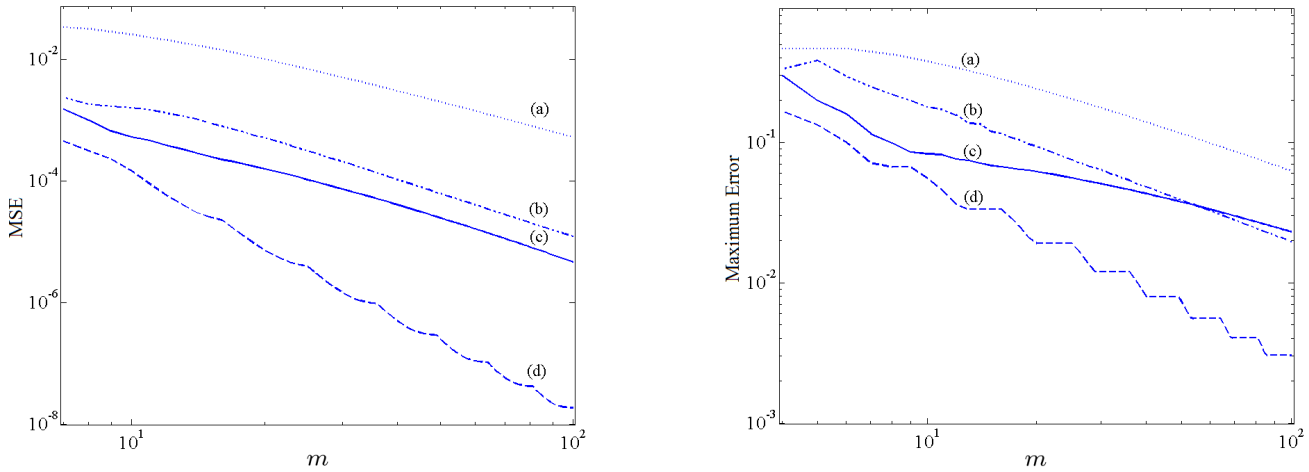

Figure 15: (i) MSE and (ii) Maximum error behavior as a function of oversampling ratio m, using (a) natural linear filter without gain/offset correction (b) Kaiser filter with $\beta = 2.5$ with gain/offset correction (c) natural linear filter with gain/offset correction and (d) optimal filter.

4 Application

In this section we use the results of the previous section to compare the performance of a conventional $\Sigma\Delta$ system with that of an incremental system with optimal filter by looking at their output spectrum. We perform the comparison for a sinusoid input and also for a small constant input.

Figure 16 shows the two different systems. f_1 and f_2 are the frequencies of the clock signals CLK_1 and CLK_2. The input signal in the conventional system is sampled at rate f_1, which is much faster than the Nyquist rate of the input signal and fed into a $\Sigma\Delta$ modulator operating at the same rate. The output sequence is decimated at rate f_2, which is slightly higher than the Nyquist rate of the input signal. Thus, the oversampling ratio is f_1/f_2. In the incremental $\Sigma\Delta$ system, the signal is sampled at rate f_2 and each sample is fed into the $\Sigma\Delta$ modulator, which operates at rate f_1. The modulator is reset after each conversion, i.e., at rate f_2. The output sequence is filtered using the optimal filter.

To compare the two systems, we first apply a sinusoidal signal to each system and generate the Power Spectral Densities (PSD) of their outputs and compute the SNDR for each system. In Figure 17 we plot the PSDs for a conventional second-order $\Sigma\Delta$ quantizer with 2500-tap Remez low pass filter with a cut-off frequency of 40KHz and an incremental second-order $\Sigma\Delta$ quantizer using optimal filtering. The results are for a sinusoidal input signal with amplitude $a = 0.5$ and frequency close to $f = 3000$Hz, modulator frequency sampling rate $f_1 = 20$MHz and Nyquist rate $f_2 = 200$KHz. Note that the actual frequency was chosen to be irrational but very close to $f = 3000$Hz. A Nuttall window is used. The SNDR for the incremental $\Sigma\Delta$ system is -97dB versus -90dB for the conventional system. Another performance measure of interest is the maximum tone power, which is the power of the largest tone created by each system. The maximum tone power of the incremental system is 5dB below that of the conventional system. This example demonstrates that by employing more sophisticated digital signal processing it is possible to improve the performance of the quantizer.

To study the idle tone difference, we apply a small dc input to the two systems. Figure 18 shows the PSD of the output of the two systems. Note that as expected (see Figure 18(b)) the incremental system inherently does not have any idle tones. This is because the same conversion is performed for all samples and, therefore, all filtered values are the same. However, in the case of the conventional $\Sigma\Delta$ the residue of the integrators vary for different samples, creating periodic distortion.

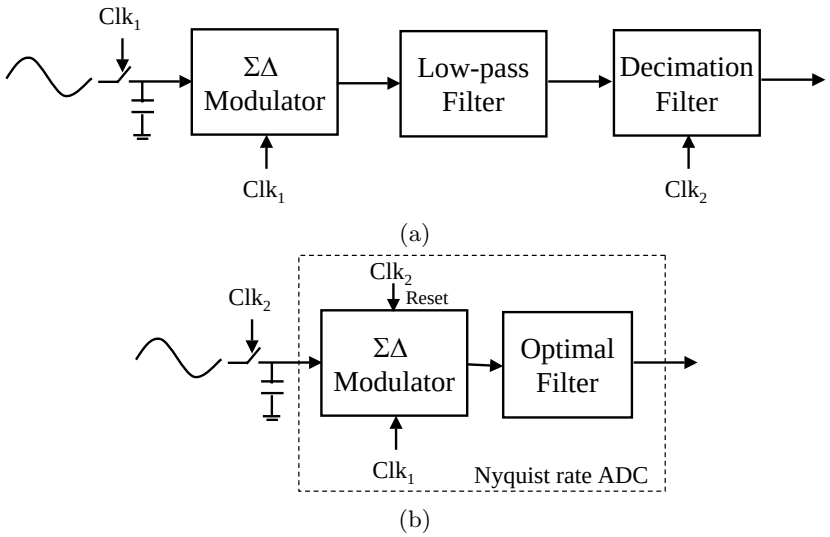

Figure 16: (a) Conventional scheme (b) Incremental with optimal filter.

In summary, the incremental ΣΔ system with optimal filtering can significantly outperform conventional ΣΔ even at moderate oversampling rates. As the oversampling rate is increased, the advantage of the incremental system becomes even more pronounced.

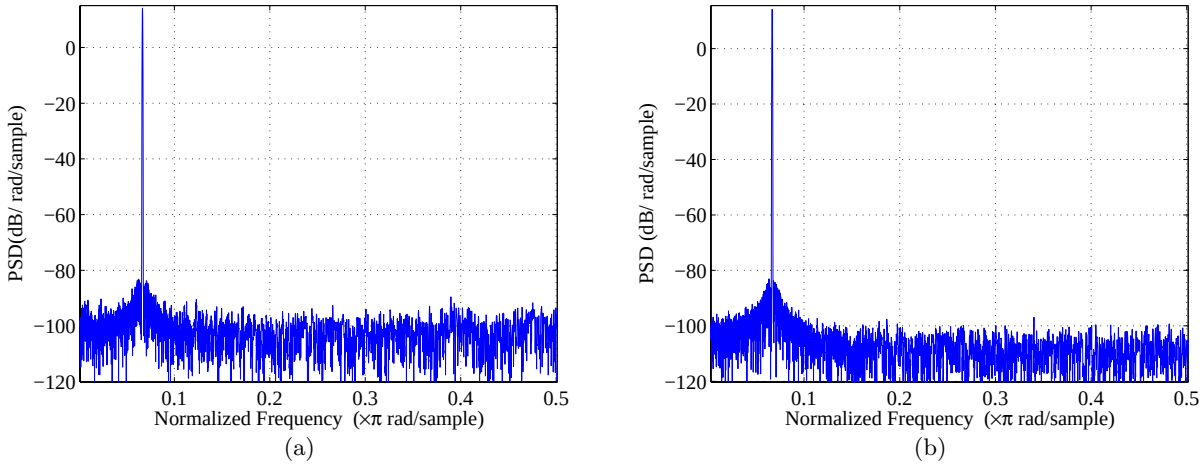

Figure 17: Example of power spectral density for a second-order ΣΔ modulator with sinusoid input used in two different ways: (a) Conventional scheme (b) Incremental with optimal filter.

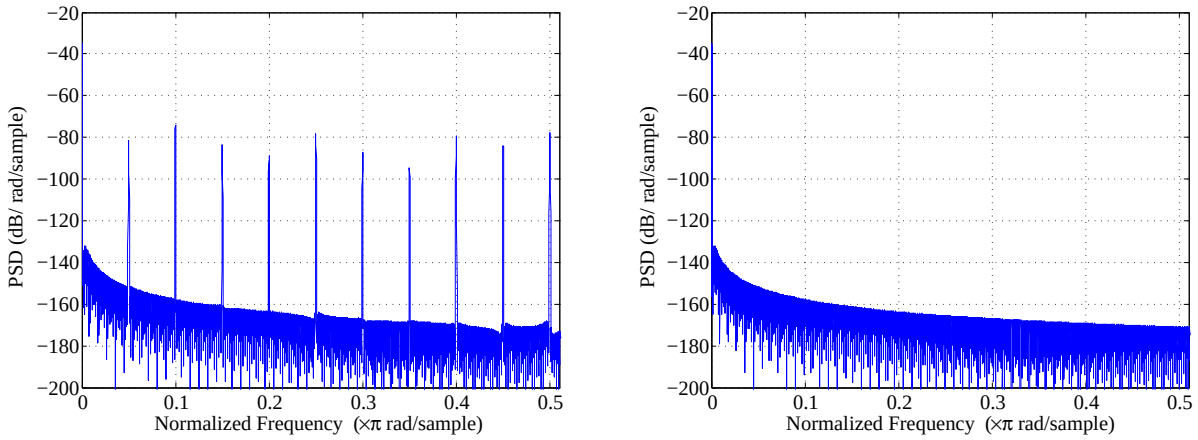

Figure 18: Example of power spectral density for a second-order $\Sigma\Delta$ modulator with small DC input of .001 used in two different ways: (a) Conventional scheme (b) Incremental with optimal filter.

5 Conclusion

We introduced a quantization-theoretic framework for analyzing incremental $\Sigma\Delta$ quantizers and used it to characterize the quantization intervals and hence transfer functions for first and second-order incremental $\Sigma\Delta$ modulators. We then used the computed quantization intervals and their corresponding sequences to characterize the performance of the optimal filter and conventional linear filters. We also showed that incremental $\Sigma\Delta$ with optimal filtering can outperform conventional $\Sigma\Delta$ systems for bandlimited input signals in terms of SNDR. Further, we showed that incremental $\Sigma\Delta$ systems do not suffer from idle tones. Even though we considered only single-bit first and second modulators in this paper, our framework can be readily extended to higher order and multi-bit modulators.

In practice, $\Sigma\Delta$ quantization system designers improve performance by increasing the oversampling ratio and/or using more complex modulators. However, high oversampling ratio results in high system power consumption. Our results show that the oversampling requirements can be relaxed by using optimal filtering and that the gain is more pronounced for higher order incremental $\Sigma\Delta$. The optimal filter requires complex signal processing than linear filters; however, the additional power consumed in the digital processing should be negligible compared to the savings in the analog front end, especially in scaled CMOS implementations.

Our analysis has assumed ideal circuit components. In practice, nonidealities such as temporal noise, offsets, and nonlinearities of the analog components limit the attainable performance. Temporal noise and offset can be lowered by increasing the oversampling ratio. However, nonlinearities and gain-bandwidth ultimately limit the performance for a given filter [22]. To achieve higher performance at lower power consumption, better filters would be needed. The time-domain analysis carried out in this paper may prove useful in deriving such filters.

ACKNOWLEDGMENTS

We wish to thank Professors R.M. Gray, B. Murmann, and B.A. Wooley, Dr. D. Su, Dr. K. Vleugels, Dr. A. Tabatabaei, K. Nam, J. Mammen, R. Navid, A. Agah, H. Eltoukhy, D. O'Brien, and K. Salama for helpful discussions.

References

[1] B.E. Boser and B.A. Wooley, "The design of sigma-delta modulation analog-to-digital converters, "IEEE Journal of Solid-State Circuits, volume 23, number 6, pp. 1298-1308 , December 1988.

[2] D. Reefman and E. Janssen, "Enhanced Sigma Delta Structures for Super Audio CD applications," 112th AES Convention, May 2002.

[3] K. Vleugels, S. Rabii and B.A. Wooley, "A 2.5-V sigma-delta modulator for broadband communications applications," IEEE Journal of Solid-State Circuits, volume 36, number 12, pp. 1887-1899, December 2001.

[4] R.H.M. van Veldhoven, "A triple-mode continuous-time $\Sigma\Delta$ modulator with switched-capacitor feedback DAC for a GSM-EDGE/CDMA2000/UMTSreceiver," IEEE Journal of Solid-State Circuits, volume 38, issue 12, pp. 2069 - 2076, December 2003.

[5] A. Tabatabaei, K. Onodera, M. Zargari, H. Samavati and D.K. Su, "A dual channel $\Sigma\Delta$ ADC with 40MHz aggregate signal bandwidth," IEEE International Solid-State Circuits Conference, pp. 66-67, February 2002.

[6] R. M. Gray, "Oversampled Sigma-Delta modulation,"IEEE Transactions on Information Theory, volume 34, number 4,pp. 826-834, May 1987.

[7] N. T. Thao and M. Vetterli, "Deterministic analysis of oversampled A/D conversion and decoding improvement based on consistent estimates," IEEE Transactions on Signal Processing, Volume 42, Issue 3, pp.519 - 531, March 1994.

[8] S. Hein and A. Zakhor, "Reconstruction of OversampledBandlimited Signals from Sigma Delta Encoded Binary Sequences," IEEE Transactions on Signal Processing, volume 42, number 4, pp. 799-811, March 1994.

[9] Richard Schreier and Gabor C. Temes, *Understanding Delta-Sigma Data Converters*, Wiley-IEEE Press, October 2004.

[10] L.A. Williams and B.A. Wooley, "MIDAS-a functional simulator for mixed digital and analog sampled data systems Williams," IEEE Proceedings of the International Symposium on Circuits and Systems, 1992.

[11] J. Candy and O. Benjamin, "The structure of quantization noise from Sigma-Delta modulation,"IEEE Transactions on Communications, volume 29, issue 9, pp. 1316-1323, September 1981.

[12] I. Galton, "Granular quantization noise in a class of delta-sigma modulators," IEEE Transactions on Information Theory, vol. 40, no. 3, pp. 848-859, 1994.

[13] S. Kavusi, H. Kakavand and A. El Gamal, "Quantitative Study of High Dynamic Range $\Sigma\Delta$-based Image Sensor Architectures,"Proceedings of the SPIE Infrared Technology and Applications XXX, volume 5406, pp. 341-350, April 2004.

[14] L.G. McIlrath, "A low-power low-noise ultrawide-dynamic-range CMOS imager with pixel-parallel A/D conversion," IEEE Journal of Solid-State Circuits,volume 36, number 5,pp. 846-853, May 2001.

[15] J. Nakamura, B. Pain, T. Nomoto, T. Nakamura and E.R. Fossum, "On-focal-plane signal processing for current-mode active pixel sensors," IEEE Transactions on Electron Devices, volume 44,Issue 10, pp. 1747-1758, October 1997.

[16] M. Pertijs, A. Niederkornand, Ma Xu, B.McKillop, A. Bakkerand and J. Huijsing, "A CMOS temperature sensor with a 3σ inaccuracy of $\pm 0.5^{o}$C from -50^{o}C to 120^{o}C,"IEEE International Solid-State Circuits Conference, pp. 200-201, February 2003.

[17] J. Markus, J. Silva and G.C. Temes, "Theory and applications of incremental $\Delta\Sigma$ converters,"IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications, volume 51, number 4, pp. 678-690 , April 2004.

[18] S. Hein and A. Zakhor, "New properties of Sigma Delta modulators with DC inputs," IEEE Transactions on Communications, volume 40, number 8, pp. 1375-1387, August 1992.

[19] L.G. McIlrath, "Algorithm for Optimal Decoding of First-order $\Sigma - \Delta$ Sequences,"IEEE Transactions on Signal Processing, volume 50, number 8, pp.1942-1950, August 2002.

[20] Allen Gersho and Robert M. Gray, *Vector Quantization and Signal Compression*, Kluwer Academic Publishers, January 1992.

[21] R. Graham, D. Knuth, and O. Patashnik, *Concrete Mathematics - A foundation for Computer Science*, Addison-Wesley Publishing Company, 1988.

[22] M. W. Hauser and R. W. Brodersen, "Circuit and technology considerations for MOS Delta-Sigma A/D converters,"IEEE Proceedings of the International Symposium on Circuits and Systems, 1986.