A Fast SNR Estimation Method for Sigma-Delta Modulator Design*

Jiandong Cheng, Guoyong Shi, and Ailin Zhang
School of Microelectronics, Shanghai Jiao Tong University, Shanghai 200240, China
shiguoyong@ic.sjtu.edu.cn

Abstract—This paper presents a fast symbolic method for computation of the signal-to-noise ratio (SNR) of switched-capacitor sigma-delta modulators. The key idea is to use a Taylor expansion polynomial to approximate the rational expression of a noise transfer function (NTF). This new method can be used in automatic optimization tools developed for switched-capacitor sigma-delta modulator design.

Index Terms—Sigma-Delta Modulator, Switched-Capacitor, Symbolic Analysis.

I. INTRODUCTION

Switched-capacitor (SC) ΣΔ Modulators (SDMs) are widely used for high performance analog signal processing [1]–[4]. Despite the popularity of SDMs, automatic design tools ranging from circuit-level to system-level are still lacking. Due to the mixed-signal nature, circuit-level simulation by a SPICE simulator is extremely slow for design verification. To quickly evaluate the performance of an SDM design, behavior-level simulations are now popular [5], [6].

Recently, there have appeared quite a number of publications addressing automatic synthesis methods of SDMs, which include automatic design of loop-transfer function coefficients [7], design space exploration [8], a synthesis method considering SNR (signal-noise ratio) variation [9], topology optimization [10], and a method to synthesize from architecture to netlist [11], etc. Other research on system-level synthesis includes [12]–[14], etc.

Despite the variety of efforts, the desire for automatic SDM synthesis is still far from reality. There are a number of intermediate steps in the synthesis cycle requiring time-consuming evaluations, such as proper selection of the loop-filter coefficients to guarantee stability, speed and resolution tradeoff, and SNR evaluations, among others. Design optimization usually requires running a synthesis cycle for many times. If each run of the synthesis cycle is too time-consuming, the design space exploration capability is damaged. Therefore, satisfactory synthesis greatly depends on the efficiency in every intermediate synthesis step.

In this work we specifically consider the issue of SNR computation and propose a symbolic method for efficiency. Establishing SNR as a function of the loop-transfer design parameters is also fundamental to optimizing an SDM. Although the proposed method is an approximate technique, it runs extremely fast with satisfactory accuracy.

In the following sections, we first make a simple review on the SDM design in Section II, then point out that the loop-transfer design of a switched-capacitor SDM can be analyzed by a symbolic analysis tool in Section III. An efficient approximate calculation method for SNR is introduced in Section IV. Test results for justifying the accuracy of SNR approximation are reported in Section V. Section VI concludes the paper.

*This research was supported in part by the National Natural Science Foundation of China (NSFC Grant No. 61176129) and by the SITU-Synopsys Joint Research Project (2010-2013) sponsored by Synopsys, Inc.

II. REVIEW OF SIGMA-DELTA MODULATOR

The idea of SDM is to use a low-resolution internal ADC to achieve high resolution by oversampling and feedback (noise-shaping). The internal ADC could be single-bit, i.e. a two-level comparator [1]. Besides choosing a proper oversampling ratio (OSR), the design of the loop-filter’s transfer function influences the SDM performance greatly.

Fig. 1 shows the general structure of a single-stage SDM [4], where $u$ is the analog input and $v$ is the digital output, and $e$ stands for the additive quantization noise. Overall, the output can be formulated as

$$V(z) = STF(z)U(z) + NTF(z)E(z)$$

(1)

where the signal transfer function (STF) and the noise transfer function (NTF) are

$$STF(z) = \frac{L_0(z)}{1 + L_1(z)} \quad \text{and} \quad NTF(z) = \frac{1}{1 + L_1(z)}$$

(2)

Here, the plus in the denominator $1 + L_1(z)$ is due to a negative feedback used in the DAC (digital-to-analog conversion) path.

![Fig. 1. A single-stage ΣΔ modulator.](image)

Shown in Fig. 2 is an example of second-order SDM [10]. The STF and NTF of this modulator can be derived easily. In general, the STF and NTF of an SDM have a rational form as follows

$$H(z) = \frac{b_0 + b_1z + b_2z^2 + \cdots + b_mz^m}{a_0 + a_1z + a_2z^2 + \cdots + a_nz^n}$$

(3)

The coefficients can be derived as functions of the circuit parameters in the loop-transfer architecture, such as the parameters shown in Fig. 2.

One important design metric is the SNR of an SDM. For a single-stage SDM, by using the linear model, SNR can be calculated by the following equations:

$$SNR = 10 \log \frac{P_s |STF(e^{j\omega})|^2}{P_{e,in}}$$

(4)

and

$$P_{e,in} = \frac{P_s}{\pi} \int_0^{\pi/R} |NTF(e^{j\omega})|^2 d\omega$$

(5)
where \( \omega_0 = 2\pi f_s \), \( R = \frac{1}{2BW} \) = OSR (the oversampling rate), \( P_s = \frac{1}{f_s} \) and \( P_b = \frac{1}{2f_s} \). \( f_n \) and \( A \) are the frequency and amplitude of test sine wave at the input, \( BW \) is the bandwidth of input signal, and \( f_s \) is the sampling frequency.

The SNR formula as given above requires the integration of NTF over the signal band. The SNR can be calculated directly by numerical integration, but it requires the construction of the loop-transfer functions, which is not always available. Hence, in practice the SNR is more commonly computed by time-domain simulation in a behavioral simulation environment. The time-domain output waveforms are sampled and analyzed by fast Fourier transform (FFT) to obtain the SNR [4].

In certain design scenario, we need to consider altering the loop-filter path coefficients. In that case, we have to run a substantial amount of Monte-Carlo simulations to estimate the varying SNR. It would be extremely time-consuming even using a behavioral simulator in Simulink\(^1\) [5].

We did an experiment to measure the correlation between the number of samples needed versus the accuracy of SNR computation. The 2nd-order SDM shown in Fig. 2 was used for examination. By choosing a set of numerical values for the SDM loop-filter parameters, Fig. 3 shows the relations between the results of SNR accuracy and behavioral simulation time versus the number of sampling points\(^2\). We see that achieving a better SNR accuracy would require exponentially increasing number of samples.

---

\(^1\)Simulink is a product of the Mathworks, Inc.

\(^2\)The test was performed on a desktop computer with a 2.3GHz dual-core processor and 2GB memory, running the Ubuntu 11.10 and the Matlab/Simulink version 2010a.

III. SYMBOLIC ANALYSIS OF SWITCHED-CAPACITOR CIRCUITS

A. Switched-Capacitor Circuit Analysis

Oversampling Sigma-Delta Modulators are mainly implemented in switched-capacitor circuits. Shown in Fig. 4 is a non-inverting integrator which has two input branches. The switched capacitors are controlled by two non-overlapping clocks \( \phi_1 \) and \( \phi_2 \). By applying charge conservation and \( z \)-transform, we can derive the relationship between the output and inputs as follows:

\[
Y(z) = \frac{X_1(z)C_a + X_2(z)C_b}{C_i} z^{-1} \frac{1}{1 - z^{-1}} = [\alpha X_1(z) + \beta X_2(z)] z^{-1} \frac{1}{1 - z^{-1}}
\]

where \( \alpha = \frac{C_a}{C_i} \) and \( \beta = \frac{C_b}{C_i} \). This equation shows that the sum and integration are performed together, with the term \( \frac{1}{1 - z^{-1}} \) representing integration. Extending it to more inputs is straightforward. Switched-capacitor circuits can be mapped to signal-flow graph (SFG) representations. Shown in Fig. 4(b) is such as SFG model.

With such basic SC building blocks, we can build up a SC circuit corresponding to the diagram in Fig. 2. Examples of such circuits can be found in many publications such as in [15].

B. Compact Symbolic Transfer Functions

Due to the charge-storage effect, a switched capacitor can be modeled by mutually controlled current sources depending on the two clock phases, see Fig. 5. This modeling method was used in the work [16] for a matrix-based nodal symbolic analysis. For using a nodal analysis formulation, the charge conservation is treated as current conservation in this method. Therefore, the charge quantities \( z^{-1/2}C'V'_{ij} \) and \( z^{-1/2}C''V''_{ij} \) are described as “currents”, where \( z^{-1/2} \) represents the half clock period delay. By a current-based formulation, the two capacitors \( C' \) and \( C'' \) are virtually treated as “conductors” in the equivalent network. In the notation we use, the single prime stands for clock phase-1 (\( \phi_1 \)) and the double primes for phase-2 (\( \phi_2 \)).

---

Fig. 2. A second-order SDM.

Fig. 3. SNR accuracy and simulation time versus the number of sampling points.

Fig. 4. Non-inverting integrator and its SFG model.

Fig. 5. Switched-capacitor equivalent circuit: (a) Capacitor symbol in two phases. (b) Equivalent circuit representation. It is assumed that \( C' = C'' = C \), \( V_{ij} := V_i - V_j \), and \( V'_{ij} := V_i' - V_j' \).
IV. APPROXIMATE SNR CALCULATION METHOD

As shown in (3), NTF would always be a rational function. The realizability condition requires that \( H(\infty) = 1 \), i.e., the highest order terms of the numerator and denominator have equal power \((m = n)\) and equal coefficients \((b_n = a_n)\) [4].

With NTF\((z)\) given as a rational expression, the integration in (5) cannot directly be derived analytically in terms of the polynomial coefficients. However, if we use a truncated Taylor polynomial to approximate the rational expression, by choosing the truncation appropriately, we may find good approximations. The coefficients of the Taylor polynomial are determined by matching the polynomial appropriately, we may find good approximations. The coefficients of the Taylor polynomial are determined by matching the polynomial coefficients. This procedure is explained in the rest of this section, which forms the main contribution of this paper.

Let the NTF be given as a rational function and it is expanded in a Taylor series centered at \( z = 1 \) (the dc point) up to the \( r \)th order,

\[
NTF(z) = \frac{b_0 + b_1 z + b_2 z^2 + \cdots + b_n z^n}{a_0 + a_1 z + a_2 z^2 + \cdots + a_n z^n} \\
\approx \frac{m_0 + m_1 (z-1) + m_2 (z-1)^2 + \cdots + m_r (z-1)^r}{\hat{p}'} = \hat{NTF}(z),
\]

where \( \hat{NTF}(z) \) denotes the approximation. Since the integration (see equation (5)) is from \( \omega = 0 \) to \( \pi/\hat{f} \), it is convenient to write the Taylor expansion at \( z = 1 \) because \( z = e^{i\omega} \). To facilitate coefficient matching, we convert both numerator and denominator to polynomials of \((z - 1)^k\); i.e.,

\[
NTF(z) = \frac{c_0 + c_1 (z-1) + \cdots + c_n (z-1)^n}{d_0 + d_1 (z-1) + \cdots + d_n (z-1)^n}.
\]

Comparing the coefficients, the coefficients of the Taylor polynomials can be calculated recursively as follows:

\[
m_0 = \frac{c_0}{d_0} \\
m_i = \frac{c_i - \sum_{k=0}^{i-1} m_k d_{i-k}}{d_0},
\]

for \( i = 1, \ldots, r \), where \( c_i = d_i = 0 \) if \( i > n \).

Returning to the integration of (5), for the convenience of integration, we change the integration variable \( \omega \) to \( z \). Knowing that \( z = e^{i\omega} \), we substitute \( d\omega \) by \( \frac{dz}{jz} \). Then, we define the indefinite integral

\[
I(z) := \int |NTF(e^{i\omega})|^2 d\omega \\
= \frac{1}{j} \int NTF(z) \frac{NTF(z^{-1})}{z} dz \\
\approx -j \hat{G}(z-1) \hat{g}_2(z-1) dz \\
= -j \hat{G}(z-1),
\]

where \( \hat{g}_1(z-1) \) is a polynomial in \((z - 1)^k\) approximating \( NTF(z) \) and \( \hat{g}_2(z - 1) \) is another polynomial in \((z - 1)^k\) approximating \( z^{-1} NTF(z^{-1}) \); \( \hat{G}(z - 1) \) is the polynomial in \((z - 1)^k\) after integration. The two rational functions \( NTF(z) \) and \( NTF(z^{-1})/z \) are separately approximated by two polynomials, \( \hat{g}_1(z-1) \) and \( \hat{g}_2(z-1) \), respectively; their coefficients can be computed similarly using the recursive formulas in (9).

Because \( I(z) \) is a real function of \( \omega \), it is also a real function of \( z \) even after approximation. Hence, the approximate integration result \( -j \hat{G}(z-1) \) must be a real function as well. It follows that

\[
P_{e,in} \approx P_{e,\pi} \text{Im} \left\{ \hat{G}(q-1) - \hat{G}(0) \right\},
\]

where \( q = e^{j\pi} \) is the integration upper bound for \( z \).

We make a comment on the numerical stability in the Taylor approximation. Since the NTF is conjugate-squared in the integrand, there are two ways to approximate: one is to Taylor-approximate the squared \( \hat{NTF}(z) \) and another polynomial in \((z - 1)^k\) approximating \( z^{-1} NTF(z^{-1}) \); \( \hat{G}(z - 1) \) is the polynomial in \((z - 1)^k\) after integration. The two rational functions \( NTF(z) \) and \( NTF(z^{-1})/z \) are separately approximated by two polynomials, \( \hat{g}_1(z-1) \) and \( \hat{g}_2(z-1) \), respectively; their coefficients can be computed similarly using the recursive formulas in (9).

Remark 1: Since an approximate SNR can be computed symbolically, it becomes easy to compute the sensitivity of SNR with respect to the loop-filter parameters. Due to space limit, we shall address this issue in another work.

V. TEST RESULTS

To test the approximation accuracy of SNR by Taylor approximation, we used the NTF of a fifth-order SCMOD given in [4]. The approximation accuracy of SNR depends on the chosen Taylor expansion order. As an example, we show in Fig. 7 that adequate approximation has been reached when the Taylor expansion order...
TABLE I

<table>
<thead>
<tr>
<th>Ckt</th>
<th>Structure</th>
<th>OSR</th>
<th>GPDD Size</th>
<th>Taylor Order</th>
<th>SNR (dB)</th>
<th>Time (ms)</th>
<th>Samples</th>
<th>SNR (dB)</th>
<th>Time (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SDM2</td>
<td>CIFF</td>
<td>32</td>
<td>125</td>
<td>5</td>
<td>46.4252</td>
<td>43.432</td>
<td>$2^{20}$</td>
<td>46.5446</td>
<td>8040.1</td>
</tr>
<tr>
<td>SDM3</td>
<td>CIFF</td>
<td>64</td>
<td>172</td>
<td>6</td>
<td>78.8137</td>
<td>51.075</td>
<td>$2^{18}$</td>
<td>78.4695</td>
<td>2597.72</td>
</tr>
<tr>
<td>SDM5</td>
<td>CRFF</td>
<td>80</td>
<td>479</td>
<td>8</td>
<td>118.144</td>
<td>157.61</td>
<td>$2^{16}$</td>
<td>118.694</td>
<td>939.438</td>
</tr>
</tbody>
</table>

Fig. 7. SNR accuracy for the 5th-order NTF using different Taylor approximation orders.

exceeds six. The numerical integration result used for reference was calculated using the method given in [18], [19].

We have integrated the approximate SNR computation method in our symbolic GPDD program which can be used for constructing a symbolic rational loop-transfer function of a switch-capacitor SDM circuit. Then, the program can generate a symbolic SNR approximation given a Taylor expansion order. Given in Table I are some experimental data that demonstrates the efficiency of the proposed method. We used three SDM circuits in the test: SDM2 (2nd order), SDM3 (3rd order), and SDM5 (5th order). The circuits SDM2 and SDM5 have been mentioned earlier in this paper. The SDM3 circuit is borrowed from a recent paper [20]. Their architectures like CIFF (cascade-of-integrators in feedforward form) and CRFF (cascade-of-resonators in feedforward form) are defined in [4]. Most of the columns have clear meaning, except for the following: GPDD Size stands for the symbolic construction complexity, i.e., the number of BDD nodes needed to represent a symbolic NTF [17]. The SNR computed by a Simulink Simulator uses FFT by sampling the time responses [4]. The numbers of sample points are given in Table I.

VI. CONCLUSION

We have presented an approximate symbolic method for fast calculation of SNR in the design of switched-capacitor Sigma-Delta modulators. We have demonstrated the significant speedup and accuracy preservation by using the proposed method. The purpose of creating a symbolic SNR representation is for optimizing the SDM circuits with SNR chosen as a main optimization target. Due to space limit, the detailed optimization procedures and experimental results will be reported elsewhere.

REFERENCES
