Simultaneous SNR and SNR-Variation Optimization for Sigma-Delta Modulator Design*

Ailin Zhang[1], Guoyong Shi[1], Sheldon X.-D. Tan[2] and Jiandong Cheng[3]
[1]Dept. of Micro/Nano-Electronics, Shanghai Jiao Tong University, Shanghai 200240, China
[2]Dept. of Electrical and Computer Engineering, University of California, Riverside, CA 92521, USA
[3]ProPlus Electronics Co., Ltd. Shanghai, China
{zhangailin,shiguoyong}@sjtu.edu.cn, stan@ece.ucr.edu, chengjd@proplussolution.com

***Abstract*—Given a loop-filter topology, a switched-capacitor sigma-delta modulator (SC-SDM) can be optimized with respect to its loop transfer function (LTF) coefficients. However, if the goal is to co-optimize signal-to-noise ratio (SNR) and SNR-variation, fast computation of both measures becomes a critical issue. In this paper a simulated-annealing (SA) based algorithm is proposed which makes use of a recently proposed fast computation method for SNR and its variance. The SNR computation method is symbolic and the computation of variation is by sensitivity. Hence, every change of the LTF coefficients can be evaluated quickly to come up with new SNR and its variation. This paper has attempted different combinations of the optimization objectives by assigning different weights to the SNR and the SNR variation. The optimization results are compared.**

***Index Terms*—Coefficient optimization, simulated annealing (SA), switched-capacitor sigma-delta modulator (SC-SDM), symbolic SNR computation.**

I. Introduction

Switched-capacitor (SC) Sigma-Delta modulator (SDM) is an important class of analog-to-digital data converters still widely used in practice today [1]. Currently, the design methodology for SC-SDM has reached maturity. However, research on the design automation issues of SC-SDMs is still in progress [2]–[6]. A complete SC-SDM design process involves many steps from the system-level specification to the circuit-level implementation, full design automation of a general-purpose SC-SDM circuit has not been realized yet.

An integral part of SDM design is to find a good modulator topology and the associated coefficients of the modulator loop filter in order to achieve the specific design goals (such as SNR, resolution, and power, etc.). In addition to meeting the performance goals, the stability of the modulator loop must be guaranteed as well. Typically, a lot of trade-offs exist for the selection of a good topology. Almost all of the proposed SDM design automation techniques apply certain heuristic techniques [3], [5]–[9].

For SNR oriented design goals, a quick SNR assessment method is needed in heuristic search. Among the proposed approaches a common and most critical bottleneck is with the SNR computation. The most straightforward method is by behavioral simulation. However, in a synthesis loop running repeated behavioral simulations is extremely costly. Therefore, some papers proposed empirical SNR formulas to speed up the evaluation. But such formulas have the accuracy problem and in many cases would not produce satisfactory optimization results. This bottleneck has recently been resolved by the proposal of a fast symbolic SNR computation method [10], [11]. Following this research, in this work we propose to combine the SNR computation method with a simulated annealing (SA) search engine [12] to optimize the coefficients of a given modulator topology with the optimization goal set to the co-optimization of SNR and its variation, which has hardly been attempted in the literature.

The symbolic SNR computation method is reviewed in Section II and the SA optimization algorithm is formulated in Section III. In Section IV experimental results are presented. Conclusion is made in Section V.

II. Review on Symbolic SNR Calculation

The symbolic SNR computation method was proposed in [10] and applied to variational SNR analysis in [11]. This method is a symbolic and approximate method, fast with excellent accuracy. Hence, it is highly suitable for the repeated computation need in a synthesis loop. The basic principle is reviewed below.

For a single loop SC-SDM, let A and f_{in} be the amplitude and frequency of the sinusoidal input signal. Let f_s be the sampling frequency. The oversampling ratio (OSR) is denoted by $R = \mathrm{OSR} = f_s/2BW$. (BW stands for *bandwidth.*) Let $\omega_0 = 2\pi f_{in}/f_s$. Then the theoretical SNR can be defined by the following equation:

$$\mathrm{SNR}(\theta) := 10\log\frac{P_S|\,\mathrm{STF}(e^{j\omega_0};\theta)|^2}{P_{N,in}(\theta)}\quad (dB) \tag{1}$$

where $P_S := \frac{A^2}{2}$ denotes the signal power, STF stands for the *Signal Transfer Function* [11], and θ denotes the dependence on the design parameters. Let $P_{N,in}(\theta)$ denote the noise power in the signal band defined by

$$P_{N,in}(\theta) := \frac{P_N}{\pi}\int_0^{\pi/R}|\,\mathrm{NTF}(e^{j\omega};\theta)|^2 d\omega, \tag{2}$$

where $P_N := \frac{\Delta^2}{12}$ denotes the total quantization noise power and NTF stands for the *Noise Transfer Function* [11]. Note

*This research was supported in part by the National Natural Science Foundation of China (NSFC) under the grant No. 61176129 and by US NSF under the grant No. CCF-1116882.

that, due to oversampling, the integration band for the noise power has been reduced by R, i.e., the OSR.

Given a rational NTF

$$\mathrm{NTF}(z;\theta)=\frac{b_0+b_1z+\cdots+b_mz^m}{a_0+a_1z+\cdots+a_nz^n} \tag{3}$$

where the polynomial coefficients belong to the parameter set θ, and $z=e^{j\omega}$ is the Z-transform variable. In symbolic SNR computation, the key step is to expanded the NTF into a truncated Taylor series at a selected reference point, say, z_0. Typically, we choose $z_0=1$ [11]. Let

$$\begin{aligned}\mathrm{NTF}(z;\theta) &\approx m_0+m_1(z-1)+...+m_r(z-1)^r\\ &\equiv \widehat{\mathrm{NTF}}(z;\theta)\end{aligned} \tag{4}$$

where $\widehat{NTF}(z;\theta)$ denotes the truncated Taylor polynomial of the rth order. The polynomial coefficients m_k can be computed with explicit formulas successively, see [11] for the details.

Since $\widehat{NTF}(z;\theta)$ is a polynomial function of z, substituting $NTF(z)$ in the integral (2) directly yields an approximate symbolic SNR computation formula, but with a truncation order.

With the SNR calculated symbolically, its sensitivity can be calculated symbolically as well. Then an approximate SNR variance can be calculated by using the first-order sensitivity [11].

For example, the capacitors (or their ratios) are typically the design parameters for an SC-SDM. Let Θ_C denote the set of all capacitors C_i. Then each $\theta_C\in\Theta_C$ is sample of the capacitors. Let $\hat{\theta}_C$ denote a collection of the nominal capacitors. We denote the perturbation of a capacitors by a small variation $\Delta C_i=C_i-\hat{C}_i$, where $\hat{C}_i\in\hat{\theta}_C$ are nominal values. The approximate SNR variation is then calculated by the following first-order equation utilizing the SNR sensitivity:

$$\mathrm{SNR}(\theta_C)\approx\mathrm{SNR}(\hat{\theta}_C)+\sum_i\frac{\partial\,\mathrm{SNR}(\hat{\theta}_C)}{\partial C_i}\cdot\Delta C_i \tag{5}$$

where $\mathrm{SNR}(\hat{\theta}_C)$ is the nominal SNR and $\mathrm{SNR}(\theta_C)$ is the perturbed SNR. The SNR variance can be estimated by

$$\sigma^2_{SNR}\approx\sum_i\left\{\frac{\partial\,\mathrm{SNR}(\hat{\theta}_C)}{\partial C_i}\right\}^2\cdot\sigma^2_{C_i} \tag{6}$$

The above equations, although approximate, are all symbolic functions of the design parameters, which means that whenever the design parameters change, the SNR and SNR variance can be computed immediately with very low cost.

III. Optimization by Simulated Annealing

SA is a randomized search algorithm that has an embedded mechanism to escape from trapping at a local minimum point [12]. Hence, SA can offer a better opportunity of landing at a global optimal. Explained in this section are the construction details of the proposed SA algorithm. We only consider minimization problem in this work.

The parameter set θ is the search space of the SA algorithm. An initial θ can be selected at random or loaded from the result of another optimization procedure. The optimization objective function, denoted by $E(\theta)$, is treated as an energy function in SA. In each search step the current parameters θ are perturbed randomly, which results in a variation of the energy function ΔE. Whether or not the perturbed new parameters are accepted depends on a probability function that is temperature dependent. Let T_k be the annealing temperature at the kth step. The probability function is then defined by

$$p(E,T_k):=\exp\left(-\frac{\Delta E}{k_BT_k}\right),\quad \Delta E>0 \tag{7}$$

where k_B is the Boltzmann constant.

If the new solution θ' leads to a negative ΔE (i.e., the objective function decreases), it is always accepted. Otherwise, the deteriorated solution is accepted by the probability $p(E,T_k)$. By (7) we see that with the same amount of energy increase ΔE, the acceptance rate at a lower temperature T_k is always lower. Hence, if a gradually lowering temperature schedule (called a *cooling schedule*) is followed, the probability of accepting a worsened solution gradually vanishes. As a result, the SA search enters an equilibrium. This procedure is called the *Metropolis algorithm* of SA.

(i) Cooling schedule: In this work the *cooling schedule* is chosen as follows: The initial temperature is set to $T_0=1,000$ and the final temperature is $T_{final}=1$. In each step the temperature T_k is updated by

$$T_{k+1}=0.96T_k \tag{8}$$

At each temperature T_k we let the Metropolis algorithm run for a certain number of user-defined times (say, 1,024). If the energy function has never been decreased after 1,024 times of parameter update, we assume that an *equilibrium* has been reached and we should lower the current temperature. When the final temperature is reached, the SA search is terminated.

(ii) Coefficient update: The coefficients θ are real variables, each of them is updated by applying a random (positive/negative) increment in percentage

$$\hat{\theta}_i=\theta_i(1+\Delta_i) \tag{9}$$

where Δ_i is a random number in a given interval, say, $[-10\%,\ 10\%]$. The detailed value is user-specified. In practice placing an upper bound for all coefficients sounds reasonable because the coefficients are realized by capacitor ratios.

(iii) Definition of the energy function: Since our goal is to have as large as possible SNR and as small as possible SNR variation, we shall incorporate both the mean μ_{SNR} and standard deviation σ_{SNR} in the energy function. Assume that the variation of each coefficient is subject to a normal distribution. Then both μ_{SNR} and σ_{SNR} can be computed by the symbolic method previously reviewed. In this work we introduce two types of energy functions as follows and will compare their effects in optimization. The first energy function is a combination of the mean and standard deviation with the user-specified weights

$$E_1(\theta)=-\alpha\mu_{SNR}(\theta)+\beta\sigma_{SNR}(\theta) \tag{10}$$

where $\alpha \geq 0$ and $\beta \geq 0$ are the weights that control the importance of each part. The SA algorithm by using this energy function is called the $SA(-\mu{+}\sigma)$ program. The second energy function is chosen as

$$E_2(\theta) = -\mu_{SNR}(\theta) + 5\sigma_{SNR}(\theta) \tag{11}$$

which is obtained by choosing $\alpha = 1$ and $\beta = 5$ in (10). This energy function is equivalent to maximizing the lower bound of the 5σ deviation from the mean value of SNR. In principle, a high mean value and a narrower standard deviation would result in a larger value of the objective function $E_2(\theta)$. The SA algorithm by using this specific energy function is called the $SA(5\sigma)$ program.

(iv) Stability control: The works [13] provided some empirical conditions on the stability of SDM versus the loop filter coefficient. The following measures are defined

$$OBG = \max_{\omega \in [\frac{\pi}{R}, \pi]} \left| NTF(e^{j\omega}) \right| \tag{12}$$

$$NPG = \frac{1}{\pi} \int_0^{\pi} \left| NTF(e^{j\omega}) \right|^2 d\omega \tag{13}$$

where OBG stands for the *Out-of-Band Gain* and NPG stands for the *Noise Power Gain* [13]. The following empirical bounds for stability have been suggested [13],

$$NPG \leq C_1, \tag{14}$$

$$OBG \leq C_2. \tag{15}$$

The detailed values of C_1 and C_2 depend on the order and the topology of SDMs (single-loop or cascade). These conditions can be incorporated in the SA search either by adding to the energy function a penalty term or by imposing a stopping criterion when the conditions are violated.

(v) Stopping criterion: A variety of stopping conditions can be set, depending on the optimization goals. In our implementation we set the target values for SNR, SNR variation, and NPG value, etc. Whenever all the target values are met, the SA search is terminated. The detailed stop conditions will be stated in the experiment section.

IV. Experiment

A single-loop three-order SDM (denoted by SDM3) is used for validating the proposed optimization algorithm. The circuit is borrowed from [9] and the schematic is shown in Fig. 1.

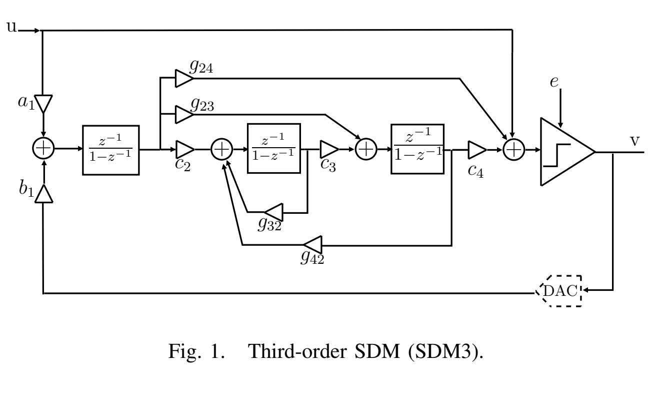


Fig. 1. Third-order SDM (SDM3).

The coefficients are labeled in the schematics as a_i, b_j, g_{ij}, $\cdots$; they are the signal path parameters to be optimized. The input signal is assumed to be a sinusoidal signal of amplitude $0.2V$ and frequency $11.714kHz$. The OSR is 32.

Initially, the mean value (μ_0), the standard deviation (σ_0) and the initial NPG (NPG_0) corresponding to the beginning coefficients are calculated. We tested both SA programs with different objective functions.

First we tested the SA program $SA(-\mu + \sigma)$ by using the following energy function:

$$E_1(\theta) = -0.01\ \mu_{SNR} + 12.0\ \sigma_{SNR} + 10^8\ (OBG - 1.6)^+. \tag{16}$$

where we place much higher weight to the standard deviation of SNR. Hence, the main goal is to have a lower SNR variation. x^+ denotes the retention of the positive part of x, i.e., $x^+ = 0$ whenever $x < 0$.

In this test the initial SDM was loaded by the result from reference [9], which wa already optimized by the method proposed there. The stopping conditions were set to the following: $\mu(\theta) \geq \mu_0$ (i.e., mean SNR gets no worse) and $\sigma(\theta) \leq 0.9\ \sigma_{SNR,0}$, and $NPG(\theta) \geq 1.1\ NPG_0$. Once these conditions are satisfied, the SA search is terminated.

The second SA program, $SA(5\sigma)$, is targeted at maximizing the 5σ lower bound of SNR by minimizing the energy function defined in (11). The stopping condition for this program was set to: the mean SNR reaches 64 dB, the standard deviation of SNR becomes 1 dB, and the NPG value has been increased from NPG_0 by 10%.

Table I summarizes the optimization results of the coefficients of SDM3, where the *Initial* coefficients are borrowed from [9], which have been optimized there.

In order to validate the optimization effect of the two SA algorithms, we have performed Monte Carlo behavioral simulation on the generated SDM. The optimized coefficients are statistically perturbed around the nominal values by assuming individual independent normal distribution. Each parameter is allowed to have a variation of 2% over its nominal value. The histograms of 10,000 Monte Carlo simulations are plotted in Fig. 2.

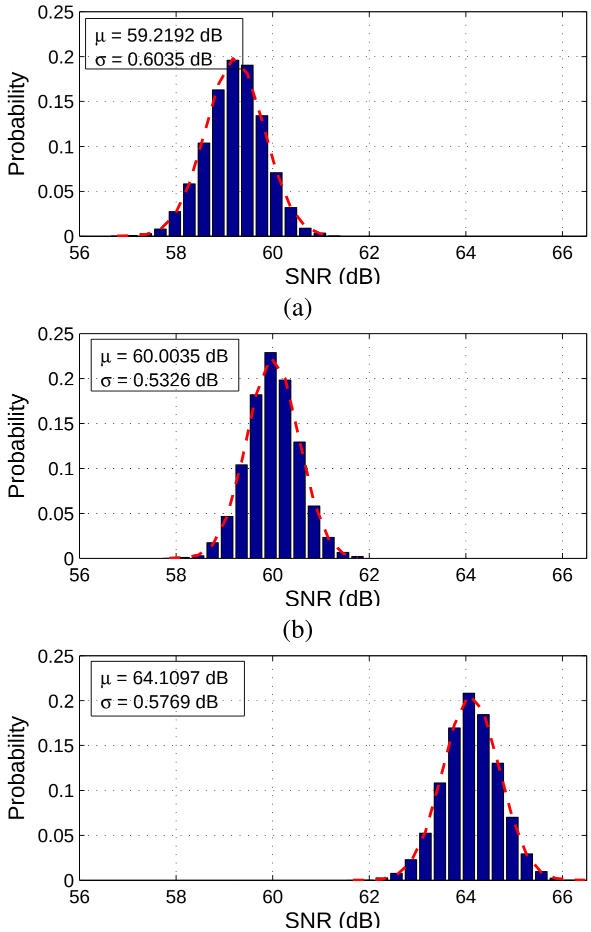
Comparing Fig. 2(b) to Fig. 2(a), we see that by the $SA(-\mu + \sigma)$ program the nominal SNR level has been increased from 59.2 dB to 60 dB while the standard deviation of SNR has been decreased from 0.60 dB to 0.53 dB. On the other hand, comparing Fig. 2(c) to Fig. 2(a), we see that by the $SA(5\sigma)$ program the nominal level of SNR has been increased from 59.2 dB to 64.1 dB (much higher) while the standard deviation is also reduced.

The following points can be concluded from the above experiment:

1) If a higher SNR level is the main target, using the $SA(5\sigma)$ program is suggested.
2) If the SNR variance is of more concern while a mild SNR is acceptable, we recommend to use the $SA(-\mu + \sigma)$ program with a proper choice of the weights α and β.

TABLE I
COEFFICIENT OPTIMIZATION RESULTS FOR SDM3.

SDM3	a_1	b_1	c_2	c_3	c_4	g_{23}	g_{24}	g_{32}	g_{42}	Time (s)
Initial [9]	0.870	0.870	0.264	0.285	0.584	0.550	0.916	-0.006	-0.02	–
$SA(-\mu+\sigma)$	0.5898	0.5898	0.3467	0.2483	0.8231	0.5574	1.3149	-0.0076	-0.0249	45.65
SA(5σ)	0.8720	0.8720	0.2241	0.5211	0.671	0.5227	0.8909	-0.0058	-0.0113	46.35



(c)

Fig. 2. Histograms of SDM3 by the three sets of coefficients: (a) Initial coefficients from [9]. (b) Coefficients optimized by $SA(-\mu+\sigma)$. (c) Coefficients optimized by $SA(5\sigma)$.

V. CONCLUSION

The main contribution of this work is the formulation of a simulated annealing algorithm for optimizing the coefficients of a Sigma-Delta modulator with a given topology. The optimization goal is set to be the co-optimization of SNR and SNR variation. In this work, the symbolic SNR computation engine has played the key role. Thanks to the fast computation engine of SNR and its variation, we are able to devise different optimization goals to investigate what kind of objective function is more appropriate for optimizing multiple goals related to SNR. As a heuristic strategy, the proposed algorithm can allow quite large degree of flexibility for a balanced optimization of SNR and its variance. This method as presented is still at a preliminary stage. Refinement and more extensive validation are needed to make this methodology practically useful.

REFERENCES

[1] R. Schreier, G. C. Temes, and J. Wiley, *Understanding Delta-Sigma Data Converters*. IEEE press Piscataway, NJ, 2005.

[2] G. G. Gielen and J. E. Franca, “CAD tools for data converter design: an overview,” *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 43, no. 2, pp. 77–89, Feb. 1996.

[3] T.-H. Kuo, K.-D. Chen, and J.-R. Chen, “Automatic coefficients design for high-order sigma-delta modulators,” *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 46, no. 1, pp. 6–15, Jan. 1999.

[4] K. Francken and G. G. Gielen, “A high-level simulation and synthesis environment for $\Delta\Sigma$ modulators,” *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 22, no. 8, pp. 1049–1061, Aug. 2003.

[5] O. Bajdechi, G. E. Gielen, and J. H. Huijsing, “Systematic design exploration of delta-sigma ADCs,” *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 51, no. 1, pp. 86–95, Jan. 2004.

[6] H. Tang and A. Doboli, “High-level synthesis of $\Delta\Sigma$ modulator topologies optimized for complexity, sensitivity, and power consumption,” *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 25, no. 3, pp. 597–607, Mar. 2006.

[7] A. Marques, V. Peluso, M. S. Steyaert, and W. M. Sansen, “Optimal parameters for $\Delta\Sigma$ modulator topologies,” *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 45, no. 9, pp. 1232–1241, Sep. 1998.

[8] S.-Y. Lee, C.-Y. Chen, J.-H. Hong, R.-G. Chang, and M. Po-Hung Lin, “Automated synthesis of discrete-time sigma-delta modulators from system architecture to circuit netlist,” *Microelectronics Journal*, vol. 42, no. 2, pp. 347–357, 2011.

[9] Y.-H. A. Ho, H.-K. Kwan, N. Wong, and K.-L. Ho, “Designing globally optimal delta–sigma modulator topologies via signomial programming,” *International Journal of Circuit Theory and Applications*, vol. 37, no. 3, pp. 453–472, April 2009.

[10] J. Cheng, G. Shi, and A. Zhang, “A fast SNR estimation method for sigma-delta modulator design,” in *Proc. IEEE Region 10 Conference (TENCON)*, Xi’an, China, Oct. 2013, pp. 1–4.

[11] J. Cheng and G. Shi, “Symbolic computation of SNR for variational analysis of sigma-delta modulator,” in *Proc. 19th Asia and South Pacific Design Automation Conference (ASPDAC)*, Singapore, Jan. 2014, pp. 443–448.

[12] S. Kirkpatrick, C. D. Gelatt, M. P. Vecchi *et al.*, “Optimization by simmulated annealing,” *Science*, vol. 220, no. 4598, pp. 671–680, May 1983.

[13] S. A. Skogstad and M. E. Høvin, “Using heuristic search to find stable high-order single-bit delta sigma modulators.” in *Proc. Symp. on Computational Intelligence in Robotics and Automation*, Jacksonville, FL, USA, June 2007, pp. 350–355.