DESIGN OF SIGMA-DELTA MODULATORS WITH ARBITRARY TRANSFER FUNCTIONS

Yongtao Wang, †Khurram Muhammad, Kaushik Roy

ECE, Purdue University, West Lafayette, IN 47907, USA
†Texas Instruments, Dallas, TX 75243, USA

ABSTRACT

This paper addresses the design of sigma-delta modulators with arbitrary signal and noise tranfer functions by presenting a genetic algorithm (GA) based search method. The objective function is defined to include the difference D between the magnitude of the frequency responses of the designed transfer functions and the ideal one, the quantizer gain $\lambda_{critical}$ for which the poles of the modulator start moving out of the unit circle, and the spread of the coefficients S. Stability can be improved by reducing $\lambda_{critical}$ while a smaller S reduces the implementation complexity. A genetic algorithm (GA) searches for poles/zeros of the transfer functions to minimize the objective function $D + w_1 * \lambda_{critical} + w_2 * S$, where w_1 and w_2 are two weighing factors. Numerical results demonstrate the effectiveness of the proposed method.

1. INTRODUCTION

Sigma-delta modulators are widely used in A/D and D/A conversion [2]. With the advent of deep-submicron CMOS processes having diminishing voltage supply, the dynamic range in the analog amplitude domain is diminishing with each process node. On the other hand, the switching speed of the transistors is increasing. Consequently, resolution in amplitude can be traded-off with resolution in time. This new paradigm has resently been exploited in the design of an all digital PLL for application to Bluetooth standard [1].

Sigma-delta modulators with arbitrary signal and noise transfer functions can be used for a multitude of applications [3]. For example, the signal transfer function can be shaped to be lowpass, so that close-in interferers can be rejected. This would relax the design of the preceding anlog filters. Similarly the noise transfer function can be modified such that the quantization noise is flat in a wider bandwidth, thereby, allowing for a high dynamic range in a wider modulation band in an ADC and easing the design of following analog filters in a D/A converter. In this paper, we address the design of such transfer functions using genetic algorithm based search. To the best of our knowledge, such work has not been presented before.

2. MODELING OF A SIGMA-DELTA MODULATOR

2.1. Linear model and transfer functions

Single-quantizer sigma-delta modulators are generally described by the architecture shown in Figure 1 [2]. In the linear model, the quantizer is modeled as a summing node with additive quatnization noise [2]. Simple derivation yields $V(z) = G(z)U(z) + H(z)E(z)$, where the noise transfer function (NTF) is $H(z) = \frac{1}{1+L_1(z)}$ and the signal transfer function (STF) is $G(z) = \frac{L_0(z)}{1+L_1(z)}$.

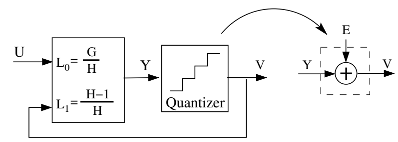

Figure 1: Linear model of a sigma-delta modulator.

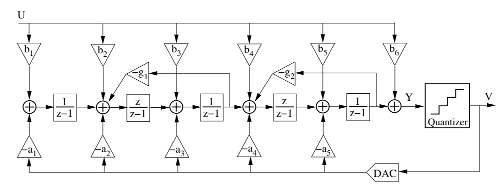

Figure 2: CRFB topology of a 5th-order sigma-delta modulator.

The loop filters $L_0(z)$ and $L_1(z)$ can be expressed as functions of the coefficients in a targeted modulator topology. For example, a cascade-of-resonators feedback form (CRFB) topology [6] of a fifth-order sigma-delta modulator is shown in Figure 2. The expressions of the tranfer functions in terms of the coefficients are usually very complex especially for high-order sigma-delta modulators. Therefore, instead of directly dealing with the coefficients, we express $G(z)$ and $H(z)$ using the zero/pole/gain representation (i.e. the zpk format) as

$$G(z) = k_G \frac{\prod_i (z - z_i^G)}{\prod_j (p - p_j^G)},\ H(z) = \frac{\prod_m (z - z_m^H)}{\prod_j (p - p_j^G)}. \quad (1)$$

Note that both $G(z)$ and $H(z)$ generally share the same poles to save hardware. To avoid delay-less loop in the modulator, the first coefficient of NTF impulse response must be unity. As a result, gain of $H(z)$ in its zpk format must be 1, and the numerator and the denominator of $H(z)$ must be of the same order to guarantee the causality of the system.

2.2. Variable gain model and stability analysis

Stability is a critical issue in designing any sigma-delta modulator. What complicates the stability of the sigma-delta modulators is the nonlinear quantizer. In the variable gain model as shown in Figure 3, the quantizer is modeled as a variable gain λ, which is defined as the ratio of the output and input voltage of the quantizer. As the input of quantizer varies, the gain λ varies ranging from zero to positive infinity. In practical implementations, λ does not reach zero since the input of the quantizer can not reach infinity. With the variable gain model, the input-output relation of the modulator can be expressed as

$$\frac{V(z)}{U(z)} = \frac{\lambda L_0(z)}{1 - \lambda L_1(z)}. \quad (2)$$

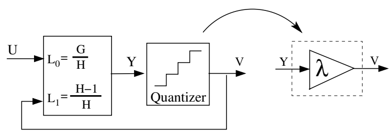

Figure 3: Variable gain model of a sigma-delta modulator.

The poles of the modulator are the roots of *the characteristic equation* $1 + \lambda R = 0$, where $R = -L_1$. Hence the trajectory of the poles of the modulator as λ varies is the root locus of the characteristic equation by varing λ from zero to postive infinity. Figure 4 shows the root locus of a fourth-order sigma-delta modulator. Traditional root locus analysis is readily applicable here. The root locus begins at the poles of R when $\lambda = 0$ and ends at the zeros R when $\lambda = +\infty$. Note that $R = \frac{1-H}{H} = \frac{H_{den}-H_{num}}{H_{num}}$, where H_{den} and H_{num} denote the denominator and the numerator of H. R has four poles since H has four zeros. Thus the root locus has four branches. Recall that in its zpk format, H has unity gain and H_{den} and H_{num} are of the same order. So the order of $(H_{den} - H_{num})$ is one less than that of H_{num}. As a result, the root locus has one asymptote at $-\infty$ and one branch of the root locus exits the unit circle along the negative real axis.

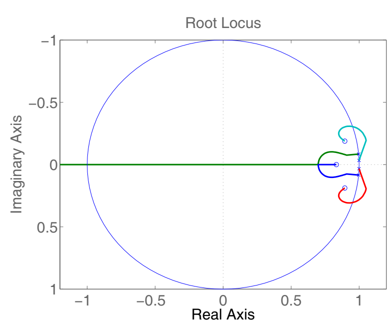

Figure 4: Root locus of a 4th-order sigma-delta modulator.

A necessary requirement for the stability of the modulator is that roots of the characteristic equation do not exit the unit circle for all occurring values of λ , except through the negative real axis [4]. Generally the roots stay inside the unit circle only for certain range(s) of λ. As λ decreases from positive infinity to zero, some branches of the root locus start moving out of the unit circle. Different branches may intersect the unit circle at different values of λ. Among these values of gain λ, we define the largest as $\lambda_{critical}$. Note that λ decreases as the input of the qauntizer increases. So the value of $\lambda_{critical}$ characterizes that smallest quantizer input at which some poles of the modulator start moving out of the unit circle for the first time. If the modulator operates at the quantizer gain smaller than $\lambda_{critical}$ for several consecutive clock cycles, the quantizer input would keep increasing and go out of bound, thereby causing system instability. Therefore, there are two ways to improve the stability of the system. One is to make sure that quantizer input is below a certain level so that λ is always (or almost always) smaller than $\lambda_{critical}$. For example, the quantizer input can be scaled or clipped. The other way is to design the modulator in a way such that $\lambda_{critical}$ is small and the modulator is more tolerant to large quantizer input. In this work, we use $\lambda_{critical}$ as the metric of the stability of the modulator and in the process of transfer function design, we incorporate the consideration of improving stability through reducing $\lambda_{critical}$.

3. PROBLEM FORMULATION

To design the transfer functons $G(z)$ and $H(z)$ is to determine k_G and their zeros and poles. There are three considerations in our problem formulation as will described in what follows.

First, the STF and NTF need to achieve certain signal processing and noise shaping capabilities. One important requirement on NTF is to achieve certain amount of quantization noise attenuation in the signal band. In addition, depending on applications, other requirements such as attenuations in other frequency bands of interest can be imposed on NTF and/or STF. To meet the requirments, k_G and zeros/poles of the STF and NTF need to be properly chosen so that the resulting transfer functions approximate the desired ones. Hence the task is to minimize a weighted sum of deviations between the responses and the desired ones since we may attach different importance to different frequency bands and/or transfer functions. we denote the sum of the weighted deviation as D.

Second, stability can be improved by reducing $\lambda_{critical}$. However, it is difficult to obtain an analytical expression for $\lambda_{critical}$, if any. In this work we propose to determine $\lambda_{critical}$ by a line search. We note that if, for a certain gain λ_1, all roots of the characteristic equation are inside the unit circle, then $\lambda_{critical} < \lambda_1$; otherwise $\lambda_{critical} \geq \lambda_1$. Thus traditional bisection line search method can be used to determine $\lambda_{critical}$. Since $\lambda_{critical}$ can be very close to zero, precision of the line search determines the smallest possible value of $\lambda_{critical}$. However, the range of $\lambda_{critical}$ is from zero to $+\infty$ while the line search needs an initial bracket with finite length. In this work we set the initial bracket to be $(0, 10)$. We first test the case when $\lambda = 10$. If all the roots of the characteristic equation are inside the unit circle, then $\lambda_{critical}$ is in interval $(0, 10)$ and we perform a bisection line search starting with the initial bracket $(0, 10)$. Otherwise, we set $\lambda_{critical} = 10$. We note that for good solutions, $\lambda_{critical}$ is generally much smaller than 10. The initial bracket $(0, 10)$ is large enough and this will not affect the proposed search algorithm since solutions with $\lambda_{critical} > 10$ rarely come into the search space after several iterations. Precision of the line search can be set to be 0.01, or smaller. A general guideline is that for better stability smaller $\lambda_{critical}$ is generally preferred, thereby requiring smaller precision in the line search.

The last consideration is the spread of the coefficients for a target modulator topology. The spread of the coefficients, denoted by S, is defined as the ratio of the maximum of the magnitude of the coefficients to the minimum of the magnitude of the coefficients. In analog design, S affects the maximum capacitance ratio. While in digital sigma-delta modualtors, it affects the wordlength for representing the coefficients. So S has an important impact on the hardware complexity and small S is desirable.

We formulate the problem as determining k_G, zeros/poles of G and H such that $D + w_1 * \lambda_{critical} + w_2 * S$ is minimized, where w_1 and w_2 are two weighing factors. Poles of H are constrained to be inside the unit circle. This is a complex nonlinear optimization problem. Inclusion of $\lambda_{critical}$ and S further complicates the problem. We resort to genetic algorithm for its capability of handling almost arbitrary kinds of constraints and objectives. When applied with proper care, it can be effective in finding good solutions for hard-to-solve practical problems.

4. GA-BASED SEARCH ENGINE

Genetic algorithm is a stochastic search method that mimics the metaphor of natural biological evolution. In the GA-based search engine, each of the variables we are to search is called a gene. In this work, genes are real-encoded. The set of genes are then cascaded to form a chromosome. Genetic algorithm operates on a population of N chromosomes applying the principle of survival of the fittest to produce improved solutions after generations of evolution. Each population undergoes three genetic operations–*selection, recombination, mutation*–and forms a new population. This process continues until certain stop criterion is met.

In this work, an initial population is generated randomly with constraints imposed by the problem. For example, poles of H are constrained to be inside the unit circle. The initial population can also be an existing set of candidate solutions.

Fitness of a chromosome is closely related to the objective function. In our case the objective function is $D + w_1 * \lambda_{critical} + w_2 * S$. Since our goal is to minimize the objective function, the smaller the objective function, the fitter the chromosome. Also note that the value of the objective function is always positive in our case. We define the fitness as the inverse of the objective function.

As mentioned above, there are three genetic operations: *selection*, *recombination* and *mutation*. In this work, *binary tournament selection* is applied, in which two chromosomes are randomly chosen and the fitter one is selected and is put into the mating pool. we perform *real-valued intermediate recombination* since the genes are real-valued. An offspring is produced according to the rule: $offspring = parent1 + \alpha * (parent2 - parent1)$, where α is a scaling factor chosen uniformly at random over an interval $(-d, 1+d)$. d is set to be 0.25 in this work. The recombination is performed according to certain probability, namely *recombination probability* p_r. An offspring is selected with probability p_m to undergo mutation. In this work, we use the mutation operator of the Breeder Genetic Algorithm [5], which is described as follows. Given a variable x, the mutated variable y is computed according to $y = x \pm range * \delta$. The plus or minus sign is chosen with probability 0.5. $range$ defines the mutation range, which is usually set to be 0.5 times the domain of definition of the variable x. Note that the domain of k_G is $(-\infty, +\infty)$.We set the mutation range of k_G to be 10, which is large enough for us to explore the design space. δ is computed as $\delta = \sum_{i=0}^{m-1} a_i 2^{-i}$, where m is called the *precision constant*. Before starting the mutation, a_i is set to be 0. Then each a_i is flipped to 1 with probability $\frac{1}{m}$.

We can stop the search after a certain number of generations, or when the current solution is good enough, or when the improvement over a certain number of generations is very small. The GA-based search engine is flexible enough to use different stop criteria.

5. NUMERICAL RESULTS

In this section we provide two design examples and numerical results to demonstrate the effectiveness of our proposed method. The choice of weighing factors w_1 and w_2 affects the shape of the search space and the solution the search algorithm will lead us to. Proper selection of weighing factors is required to strike a balance between the three design considerations and provide proper guidance to the search engine.

In the first example, we apply our method to design of a *regular* 4th-order sigma-delta modulator. By *regular* we mean that we put all four zeros in the signal band of NTF for attenuating quantization noise. The oversampling ratio (OSR) is 40 and the STF is designed to be unity over the whole frequency range.

The *delsig* toolbox in [6] produces a design which achieves a peak SNR of 87.2dB with 1-bit quantization. With CRFB topology, $S = 1403$. It is stable when the input amplitude is less than 0.7 and $\lambda_{critical} = 0.496$. First we focus on searching for solutions with large peak SNR and good stability (i.e., stable input range is about (0, 0.7) or (0, 0.8)). In addition, we want to avoid designs with overly large spread of coefficients. With these goals in mind we set $w_1 = 0.01$ and $w_2 = 10^{-6}$. The search engine first yields a satisfactory solution, which is called *solution A*. Continuing on *solution A*, the genetic algorith based search engine finds another good solution, which is called *solution B*. In comparison to the *delsig* solution, *solutions A* has a slightly smaller peak SNR, 86.3dB, but a larger stable input range (0, 0.8) with $\lambda_{critical} = 0.200$ and smaller coefficient spread $S = 1168$. *Solution B* has a peak SNR of 89.6dB, which is 2.4dB higher than that of *delsig* solution. Its stable input range is (0, 0.7) with $\lambda_{critical} = 0.234$ and a samller coefficient spread $S = 922$. We increase w_2 to 10^{-3} to search for designs with even smaller coefficient spread and we reach a solution, *solution C*, which has a peak SNR of 86.4dB, stable input range of (0, 0.7), $\lambda_{critical} = 0.260$. Compared to *solution B*, although the peak SNR is about 3dB lower, coefficient spread of *Solution C* is reduced by a factor of 1.95 to be just 473, which is about one-third of that of the *delsig* solution.

Then we challenge ourselves to search for solutions with a stable input range of (0, 1), i.e., with full-range stability. For improved stability we need to increase w_1 so that the search engine will be guided to search for solutions with much reduced $\lambda_{critical}$. With $w_1 = 5$ and $w_2 = 10^{-6}$, the search egnine yields a solution, *solution D*, with full-range stability. Notably $\lambda_{critical}$ is reduced to 0.053 and the stability is improved. Peak SNR of *solution D* is 75dB and $S = 791$. With $w_1 = 5$ and $w_2 = 10^{-3}$, our search engine

Table 1: Coefficients of solutions of the first example.

	$a_1(b_1)$	$a_2(b_2)$	$a_3(b_3)$	$a_4(b_4)$	g_1	g_2
A	0.0082	0.0442	0.4586	0.7109	0.0043	0.0009
B	0.0153	0.0519	0.5395	0.6931	0.0046	0.0011
C	0.0187	0.0566	0.5483	0.7477	0.0047	0.0021
D	0.0013	0.0117	0.4731	0.6727	0.0050	0.0013
E	0.0046	0.0257	0.5026	0.6936	0.0040	0.0040

Table 2: Coefficients of solutions of the second example.

	$a_1(b_1)$	$a_2(b_2)$	$a_3(b_3)$	$a_4(b_4)$	$a_5(b_5)$	g_1	g_2
F	0.0275	0.1760	0.4765	0.8611	1.0135	0.0038	0.3820
G	0.0312	0.1704	0.4878	0.8190	0.9489	0.0060	0.3820
H	0.0104	0.1470	0.3882	0.8179	1.0600	0.0036	0.3820
I	0.0084	0.0892	0.4037	0.7255	0.9557	0.0060	0.3820

finds a solution, *solution E* with a very small spread of coefficients $S = 250$. Its peak SNR is 73dB, stable input range is (0,0.9) with $\lambda_{critical} = 0.098$.

Coefficients of the solutions A, B, C, D and E are shown in Table 1. Coefficient b_5 of all five soltuons is 1. This example demonstrates that our proposed method can explore a much broader design space and find good solutions with different characteristics in terms of peak SNR, stable input range and spread of coefficients. This enables designers to make tradeoffs between different objectives and/or constraints, and to accomodate different design requirements.

In the second example, we design an *irregular* 5th-order sigma-delta modualtor with OSR being 40. In NTF, instead of putting all zeros into the signal band we put three zeros into the signal band and a pair of complex conjugate zeros at a out-of-band frequency $f = 0.1$(The sampling frequency f_s is normalized to 1). The STF is unity over the whole frequency range. For this type of *irregular* modulators, existing methods such as *delsig* may not be directly applicable since their problem formulation and stability criterion assume that all zeros are put into the signal band of the NTF. However, our problem formulation and stability analysis based on root-locus analysis, do not rely on that assumption.

As with the first design example, by choosing different weighing factors, we can seach for solutions with different characteristics. With $w_1 = 0.6$ and $w_2 = 10^{-6}$ the search engine finds a solution with peak SNR 68dB, stable input range (0, 0.7), $\lambda_{critical} = 0.1626$ and $S = 267$, which is called *solution F*. One may note that this peak SNR value is much lower than that of *solution B*. This is mainly due to the fact that in the second exmaple, only three, not five, zeros are put into the signal band of NTF for attenuating quantization noise. Setting $w_1 = 0.6$ and $w_2 = 10^{-2}$ leads to *solution G* with a much reduced coefficient spread 167 and peak SNR 63dB, stable input range (0, 0.7) and $\lambda_{critical} = 0.1817$. With $w_1 = 5$ and $w_2 = 10^{-6}$, the search engine finds *solution H* with full-range stability, peak SNR 63.7dB, $\lambda_{critical} = 0.0709$ and $S = 298$. By setting $w_1 = 5$ and $w_2 = 10^{-2}$, we find *solution I* with full-range stability, peak SNR 56dB, $\lambda_{critical} = 0.0724$ and a much smaller coefficient spread $S = 168$. Coefficients of solution F, G, H and I are shown in Table 2. Coefficient b_6 of all four solutions is 1.

Due to limited space, only the SNR plot of the four solutions of the second example is shown in Figure 5.

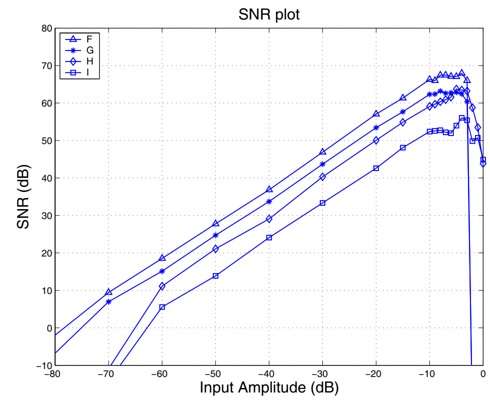

Figure 5: SNR plot of the solutions of the second example.

As can be seen from the above two examples, the genetic algorithm based search engine can effectively search for solutions with different characteristics and enables tradeoffs between different design considerations. Also we observe that reduced $\lambda_{critical}$ generally leads to improved stability, full-range stability usually comes at the expense of lower peak SNR, and, limiting spread of coefficients could hurt peak SNR and/or stability.

6. CONCLUSIONS

We propose a new method for design of sigma-delta modualtors with arbitrary transfer functions. A GA-based search engine is devloped to determine the poles and zeros of the transfer functions for minimizing an objective function defined as a weighted sum of deviation in frequency response, $\lambda_{critical}$ and the spread of the coefficients S. Desgin examples and numerical results demonstrate effectiveness of our proposed method.

7. REFERENCES

[1] R. B. Staszewski *et. al.*, "All Digital Phase-Domain TX Frequency Synthesizer for Bluetooth Radios in 0.13um CMOS", Digest of Technical Papers ISSCC, 15.3, 2004.

[2] S. R. Norsworthy, R. Schreier and G. C. Temes, Eds., *Delta-Sigma Data Converters: Theory, Design, and Simulation.* Piscataway, NJ: IEEE Press, 1997.

[3] V. da Fonte Dias, "Sigma-delta signal processing", *IEEE International Symposium on Circuits and Systmes*, vol. 5, pp. 421-424, May 1994.

[4] E. F. Stikvoort, "Some remarks on the stability and performance of the noise shaper or sigma-delta modulato", *IEEE Transactions on Communications*, vol. 36, no. 10, pp. 1157 - 1162, Oct. 1988.

[5] H. Mühlenbein, "The Breeder Genetic Algoirithm–a provable optimal search algorithm and its applications", Colloquium on Applications of Genetic Algorithms, IEE 94/067, London, 1994.

[6] R. Schreier, "The Delta-Sigma Toolbox 6.0", *www.mathworks.com/matlabcentral/fileexchange*, Jan. 2003.