This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TCSII.2016.2572219

Lee's Rule Extended

Marko Neitola

***Abstract*—The initial step on designing a Delta-Sigma data converter is typically generating a noise transfer function based on a rule-of-thumb such as Lee's famous rule for single-bit converters. As known, rules of such nature are merely suggestive and should be treated as an educated guess. This brief proposes a novel design rule based on optimizing noise transfer functions to a fixed pair of norm-based metrics. Through a vast amount of behavioral simulations, a simple rule was created and it was found quite accurate. The new rule is applicable for both one- and multi-bit converters.**

***Index Terms*—delta-sigma, transfer function, optimization.**

I. Introduction

THE prior art in rule-of-thumb -based optimization for a baseband Delta-Sigma Modulator (DSM) noise transfer function (NTF) contains classics that are well-known for most researchers in the field such as [1]–[4].

The subject of NTF optimization via ad-hoc rules is still relevant and the above classics (especially [1]) are cited in thousands of publications and patents. There are, of course, classics on non-linear e.g. [5], [6] and quasi-linear [7], [8] DSM analysis. Such analyses yield more realistic results, but the true dynamics for high-order (≥ 3) DSMs are still intractable [9].

Classic ad-hoc rules for single-bit systems were thoroughly tested by Schreier in [4] and they were found quite unreliable. He tested single-bit DSMs mainly on how the simulated dc-stability agrees with common rules-of-thumb by Lee (1) [1], Agrawal and Shenoi (2) [2] and Kenney and Carley (3) [10]:

$$||H||_\infty = max(|H(e^{j\omega}|) < 2 \tag{1}$$

$$||H||_2^2 < 3, \text{ where } ||H||_2 = \sqrt{\int_0^\pi |H(e^{j\omega})|^2 d\omega} \tag{2}$$

$$||h||_1 \leq 3 - u_{max}, \text{ where } ||h||_1 = \sum_{n=1}^{\infty} |h(n)| \tag{3}$$

The norms in (1) and (2) are the infinity-norm and the 2-norm based on NTF's frequency domain expression $H(z)$, respectively. The norm $||h||_1$ used in (3) is the 1-norm for $h(n)$, which is the impulse response of $H(z)$. Term u_{max} in (3) is the maximum stable dc-input level. In [4], Schreier concluded that simulations are the most reliable method for verifying stability as none of the existing ad-hoc rules for stability are adequate for the design of high-order single-bit delta-sigma modulators.

For the increasing number of quantization levels, the quantization noise behavior in multibit DSMs is less correlated with the stimulus: the uniform quantizer noise distribution model becomes more accurate. A $||h||_1$-based condition for stable maximum level u_{max} that guarantees a non-overloading quantizer with N_{LEV} levels is [9], [10]:

$$u_{max} \leq N_{LEV} - hQ, \text{ where} \tag{4}$$

$$hQ = ||h||_1 - 1 \tag{5}$$

In (4), the quantizer step is assumed as 2 and u_{max} is an arbitrary stimulus whose full-scale ranges from $-N_{LEV}+1$ to $N_{LEV}-1$. The parameter hQ is the maximum accumulation of quantization errors and it should be less than or equal to the total number of quantization levels.

As mentioned, the ad-hoc rules are widely used as a helpful tool. Recent publications dedicated to NTF optimization via a norm-based rules are however quite scarce. Roverato *et al.* (2015) [11] used Lee's rule in their proposal for a fixed-angle pole-placement scheme for NTFs used in tunable-center-frequency bandpass DSMs. Løkken *et al.* (2006) [12] proposed a norm-based approach to predict the non-overload quantizer input range.

This brief proposes a method of extracting a general design rule for single- and multi-bit systems. The method is based on optimizing a NTF that complies with two norms: $||H||_\infty$ and $||h||_1$. Here, the NTF pole optimization is a customization of optimization procedure CLANS (Closed-Loop Analysis of Noise-Shaping Coders) for multi-bit discrete-time systems. CLANS was published by Kenney and Carley in [3], [10], further documented in [13] and published as MATLAB-code in [10] and later in [14]. The $||h||_1$ -norm used in CLANS is the parameter hQ described in (5).

Section II presents the modifications made for the CLANS-routine in [14]. The added norm, $||H||_\infty$, is easy to accommodate and additional changes were made to improve the controllability of poles' quality factors.

In Sect. III, it will be presented how a large amount of NTFs was optimized for swept hQ and $||H||_\infty$. Section IV explains how the new design rule was extracted from behavioral DSM simulations using the optimized NTFs. The design rule is verified in Sect. V.

The novel design rule finds the best-suited parameters $\{hQ, ||H||_\infty\}$ for maximum input level and the number of quantization levels. The parameters are then used to optimize the NTF programmatically. After this, the NTF is ready to be verified by behavioral simulations.

II. CLANS – Closed-Loop Analysis of Noise-Shaping Coders

This section explains the changes made for the original CLANS-routine, which optimizes discrete-time NTF poles for

The author is with the CAS research unit, Faculty of Information Technology and Electrical Engineering, University of Oulu, Finland, e-mail: marko.neitola@oulu.fi.

baseband multi-bit DSMs. The original code is enclosed in Schreier's Delta-Sigma toolbox [14].

A. The Original Version

As explained in [13], the CLANS-routine optimizes the NTF poles using bilinear transformation (Tustin's method). The optimized s-domain parameters are the corner frequency ω_n for real poles and complex pole pairs and specifically for the complex pole pair: the damping ratio ζ. The optimization contains parameter r_{max} to control the maximum radius of the optimized pole:

$$z = r_{max} \cdot \frac{1+s}{1-s}. \tag{6}$$

The objective for optimization is to minimize the NTF in-band rms level. CLANS [14] uses the built-in function 'fmincon' from Matlab's Optimization toolbox. The function supports user-defined constraint functions that may contain one inequality $c(x)$ and one equation $c_{eq}(x)$. Here x contains the optimized s-domain pole parameters $\{\omega_n, \zeta\}$. In CLANS [14], $c_{eq}(x)$ is omitted and the inequality $c(x)$ is:

$$c(x) = hQ - hQ_{GOAL} \leq 0, \tag{7}$$

where hQ_{GOAL} is intended value for hQ. This results in hQ-values less than or equal to the intended value. The current version (2016) of CLANS [14] has $c(x)$ squared, which makes the converged hQ-value is more likely to converge to the goal value. The squaring turns the inequality to an equation, as the error squared cannot be less than zero.

B. Modified version – Two Norms and Q-Factor Boundaries

According to [10], [13], the reason for the s-domain pole optimization in CLANS is that it improves the convergence. The optimization can be expedited by omitting the transformation, but the transformation enables setting boundaries in the optimized corner frequencies and, most importantly, the quality factors Q.

In the original version, the user-provided maximum pole radius r_{max} appears to be yet another parameter to be optimized. By setting optimization boundaries to the damping ratio ζ (and therefore the quality factor Q) seems as a more obvious way to limit non-stable or high-Q pole results. Moreover, upper and lower parameter limits are inherent in the 'fmincon' optimization function used by CLANS. The r_{max} parameter can therefore be omitted.

The new version of CLANS uses built-in Matlab-functions for the bilinear approximation (found in the Control System Toolbox): 'd2c' (discrete-time to continuous-time) and 'c2d' (continuous to discrete). The aforementioned functions enable experimenting with conversion methods.

As mentioned, squaring the constraint function $c(x)$ in (7) changes the inequality into an equation. This can also be done by taking the magnitude of $c(x)$, which was done here. The infinity norm $||H||_\infty$ will be forced to its goal value by the constraint function equation $c_{eq}(x)$:

$$c_{eq}(x) = ||H||_\infty - ||H||_{\infty GOAL} = 0. \tag{8}$$

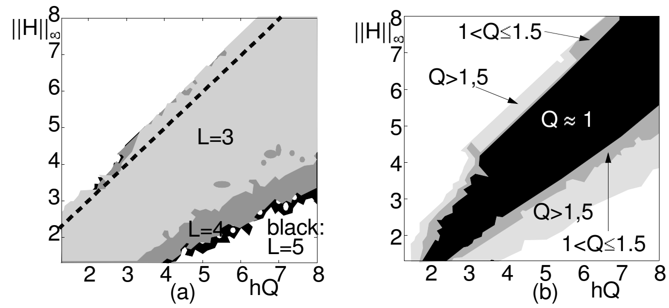

Fig. 1. NTF optimization with $Q \in [0.5, 4]$: a) Attainable hQ and $||H||_\infty$ for a group of NTF orders L $\in [3, 5]$ and b) Quality factors for L=4.

The objective here is to find accurate hQ and $||H||_\infty$ for the NTF. The initial task in creating the novel design rule is to create a large set of NTFs with specific pair of NTF metrics. This will be presented next.

III. NTF Pole Optimization

Here, thousands of NTFs were optimized for swept NTF metrics $\{hQ, ||H||_\infty\}$. The oversampling ratio was set to 32. The obtained $\{hQ, ||H||_\infty\}$ were not always in-line with the goal values, so the maximal absolute deviation was set to ± 0.3 for both NTF metrics. The matching results for three NTF orders are presented as areas in Fig. 1a.

In Fig. 1a, the area for fifth-order NTF (black) is the largest and it is beneath all other areas. The achievable $\{hQ, ||H||_\infty\}$ areas increase as a function of NTF order L: more poles results in more degrees of freedom. The dashed line in Fig. 1a depicts the upper limit for $||H||_\infty$. Above this limit, accurate NTF metrics cannot be attained. At the limit, the complex poles tend to converge at the in-band edge. The limitation (10) can be found by brute-force simulations, but it is mathematically justified by comparing the norms via z-transform of the impulse response $h(n)$ in (9).

$$max(|\sum_{n=0}^{\infty}(h(n) \cdot e^{-j\omega n})|) \leq \sum_{n=0}^{\infty}(|h(n)|) \tag{9}$$

$$\Rightarrow ||H||_\infty \leq ||h||_1 = hQ - 1 \tag{10}$$

The areas in Fig. 1a contain a few visible fragments where the optimization converged to inaccurate hQ or $||H||_\infty$. These were found to be more severe in the case of the non-bounded quality factor. These fragments are naturally shown in the simulation results presented in the next Section.

From NTFs, it is possible to observe the contours of maximal quality factor Q versus hQ and $||H||_\infty$, see Fig. 1b. The example in Fig. 1b is only for order of 4, but the middle-point-line of near-unity Q is quite the same from orders 3 to 7. This information was used as a secondary guideline to create the resulting new design rule in Sect. IV-B.

IV. Simulations and the New Design Rule

A. Acid Test – Find Minimum N_{LEV} Through Simulations

In the simulations, the objective is to find the stable minimum for the number of quantization levels required for given input level. Naturally, the simulation setup (a.k.a. acid test) needs to be quite extensive. Full certainty can not be promised, but the following steps provide at least a moderate proof of stability. For a certain maximum input level or amplitude A:

1) Perform four single-tone simulations with amplitude A. The tones are from mid in-band to the band edge. The obtained performance {SNDR,SFDR} values are not allowed to vary more than {10,20} dB. The maximum total in-band noise n_{max} is stored.
2) Re-perform the above with tiny amplitude to ensure modest SNDR curve monotonicity. The maximum in-band noise should not exceed n_{max} by no more than 10 dB.
3) Perform two-tone simulation with maximum input level of $A/2$. The max. in-band noise should not exceed n_{max} by no more than 20 dB.
4) Perform dc-simulations with levels A/2 to $A/\sqrt{2}$. The maximum in-band noise should not exceed n_{max} by no more than 20 dB.

Here, SNDR is the signal to noise and distortion ratio and SFDR is the spurious-free dynamic range. In step 4, dc-stability is tested and (to be on the safe side) the multiplicand $1/\sqrt{2}$ was added. The maximum dc-input level can be smaller than the maximum 1-tone amplitude, especially in the case of two-level quantization.

For all NTFs swept by $\{hQ, ||H||_\infty\}$, the above consecutive simulation rules were repeated with the decreasing number of quantization levels. The minimum number of quantization levels were found if any of the steps failed. The test criteria may seem merciful, but they allow aggressive NTFs i.e. NTFs at the verge of instability. Here, the maximum number of quantization levels was set to 8.

Beginning with the NTF order L of 5, Fig. 2 shows the number of minimum number of stable quantization levels for four different amplitudes A (full-scale is normalized to unity). In Fig. 2, the quality factor of poles is bounded to $Q \in [0.5, 4]$. The case of non-bounding Q seen in Fig. 3. The non-bounded version is more fragmented and the fragments are due to failed NTF optimizations. As mentioned, the allowed maximum absolute deviation from the goal values was set to 0.3 for both hQ and $||H||_\infty$.

Figures 4 and 5 show the corresponding results for NTF order of 3. There are fragments due to unbounded Q in Fig. 5, but not as severe as for NTF order of 5 (Fig. 3).

In Fig. 4, the area of two quantization levels contains a gap in which two quantization levels fail the acid test. The gap also exists in the case of non-bounding Q (Fig. 5), but it seems slightly smaller. Apparently, near the gap the stable NTF for 1-bit DSM requires high-Q poles for L=3. Such gap did not appear for orders 4 to 7.

Figures 2 to 5 provide a tangible depiction of NTF's level of aggression: the minimum number of quantization levels for given input level A. The input level (or amplitude) A is the maximum input level u_{max} **only** at the verge of step-increment in figures 2 to 5.

Bounding the quality factor reduces the probability of fragments in optimization results. For bounded Q, the N_{LEV}

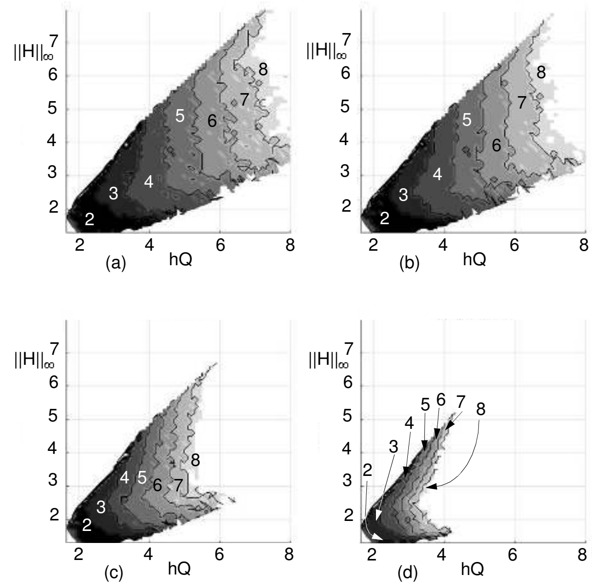

Fig. 2. Minimum amount of quantization levels for L = 5 and $Q \in [0.5, 4]$: a) A = 0.3, b) A = 0.5, c) A = 0.7 and d) A = 0.9.

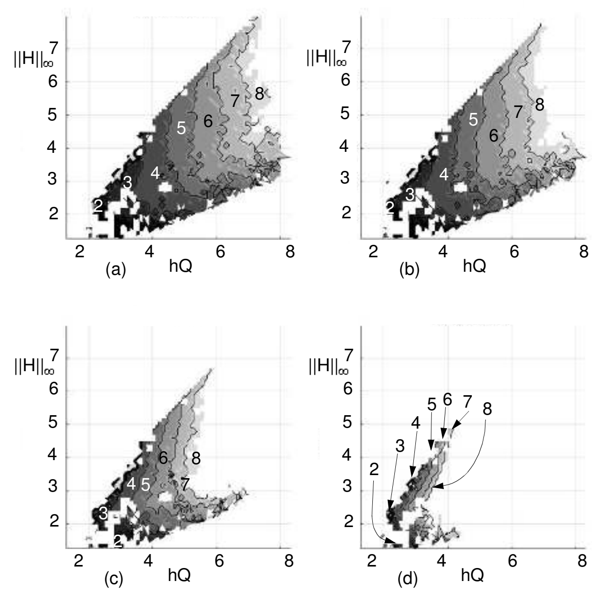

Fig. 3. Minimum amount of quantization levels for L = 5 and no bounds in Q: a) A = 0.3, b) A = 0.5, c) A = 0.7 and d) A = 0.9.

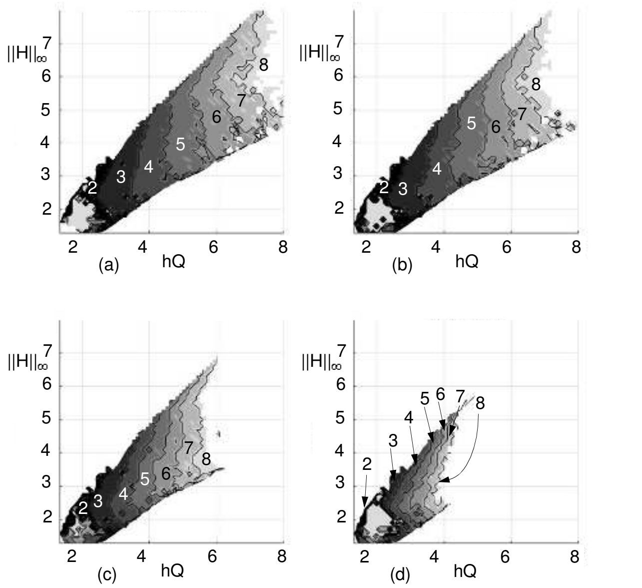

Fig. 4. Minimum amount of quantization levels for L = 3 and $Q \in [0.5, 4]$: a) A = 0.3, b) A = 0.5, c) A = 0.7 and d) A = 0.9.

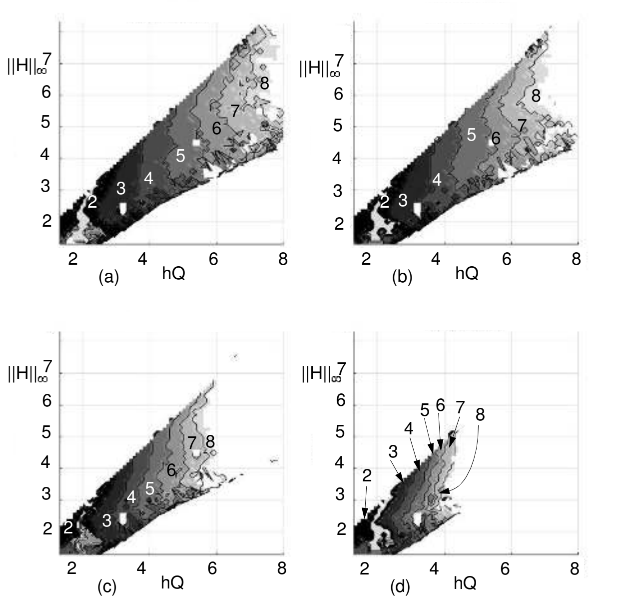

Fig. 5. Minimum amount of quantization levels for L = 3 and no bounds in Q: a) A = 0.3, b) A = 0.5, c) A = 0.7 and d) A = 0.9.

areas were created for orders 3 to 7 from which the novel rule was created.

B. The New Design Rule

An aggressive NTF refers to a NTF that enables high performance but has often limited stable input level range. Perhaps counter-intuitively, aggressive does not necessarily mean high-Q poles. From the acid-test simulations for orders 3 to 7, the best performance for a certain number of quantization levels was always found at the verge of the next quantization level, regardless of quality factor Q.

A linear dependency between hQ and $||H||_\infty$ was chosen in the novel rule, see the lower equation of (11). This dependency was chosen so that

- the NTF order has negligible effect on how the quantization level thresholds are placed for different values of u_{max} and
- the NTF quality factors are near unity thus avoiding high-Q poles (see Fig. 1b).

The required number of levels N_{LEV} and maximum stable amplitude $A = u_{max}$ define the value of hQ, see the upper equation of (11). This equation was found by a polynomial fit: the values u_{max} and N_{LEV} versus hQ were tabulated using the linear $||H||_\infty = 3/4 \cdot hQ$ dependency. The fit order is 2 and the maximum-absolute and rms fit errors (error between obtained and fitted $hQ(N_{lev}, u_{max})$) were about 16 % and 7 %, respectively.

$$\left.\begin{aligned} hQ =& 1.23 \cdot N_{LEV} + 9.62 \cdot u_{max} - 1.06 \cdot u_{max} \cdot N_{LEV} \\ & - 6.25 \cdot u_{max}^2 - 2.53, u_{max} \in [0.3, 0.9] \\ ||H||_\infty =& 3/4 \cdot hQ \end{aligned}\right\} \quad (11)$$

The above dependencies were approximated from NTF orders of 3 to 7. The group of equations in (11) provide a simple procedure for NTF design: first, find hQ for the given number of quantization levels N_{LEV} and maximum input level u_{max}. Then, find the needed infinity-norm. Generating the NTF requires the modified CLANS-procedure. The modified CLANS-routine in [15] enables generating the NTF using the new rule (11) directly, i.e. the function calculates the NTF from u_{max} and N_{LEV} provided by the user.

V. Feasibility Test

The novel design rule in (11) is based on simulations with the oversampling ratio (OSR) of 32. The oversampling ratio does affect where the poles converge, but it should not have major influence on the graphs N_{LEV} versus $\{hQ, ||H||_\infty\}$. Reducing the OSR reduces the number of viable pole-convergence points and makes the NTF more susceptible to instability (especially for higher orders).

Whether the NTF zeros are optimized has little significance on where the poles converge and thus does not affect the rule. Of course, the NTF zeros have a major impact on the modulator's performance, but not on its stability.

The new rule was tested by setting maximum stable amplitude level at $u_{max} = 0.6$ (full-scale is normalized to unity)

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TCSII.2016.2572219

TABLE I
ACHIEVED NUMBER OF QUANTIZATION LEVELS FOR MAXIMUM STABLE INPUT-LEVEL OF 0.6

		Required number of quantization levels						
OSR	L	2	3	4	5	6	7	8
16	3	2	3	4	5	***5***	***6***	***7***
16	4	2	3	4	5	6	7	***7***
16	5	**4**	3	4	5	6	7	***7***
16	6	2	3	4	5	6	***6***	***7***
16	7	2	3	4	5	***5***	***6***	***7***
32	3	**6**	3	4	5	6	7	***7***
32	4	2	3	***3***	5	6	7	8
32	5	2	3	4	5	6	7	8
32	6	N/A	3	4	5	6	7	8
32	7	**5**	3	4	5	6	7	8
64	3	N/A	N/A	N/A	5	6	N/A	8
64	4	2	3	**5**	5	6	7	8
64	5	2	3	**5**	5	6	7	7
64	6	2	3	**5**	5	6	7	8
64	7	**3**	3	**5**	5	6	7	8

using the acid-test described at the beginning of Sect. IV-A. The required values for N_{LEV} ranged from 2 to 8 for five NTF orders L and three oversampling ratios.

The results are shown in Table I. From the achieved number of quantization levels, the results with bold font are the cases where the achieved N_{LEV} was larger than the goal. Those are interpreted as bad results, as well as the instable cases notated as 'N/A'. The total number of bad results was 13 out of 105. There are also 13 cases where the achieved N_{LEV} was smaller than the goal by one level. These are not as severe results and these are emphasized using the bold-italic font.

From Table I, the percentage of instable results or too high values of N_{LEV} is about 12%. The percentage of perfect matches for N_{LEV} is about 75%. These are impressive results for such a simple rule.

As can be seen from Table I, the new rule does not guarantee stability, but tends to find NTF properties that match the user-provided maximum input level and the number of quantization levels. So how does the condition for multi-bit design (4) [10] presented in the introduction agree with the novel rule? Reflecting the condition (4) with attained N_{LEV} versus u_{max}, the condition (4) is more conservative than (11) as it predicts smaller stable input levels.

Here and in Sect. IV, the original Lee's rule (1) was not compromised for NTFs suited for 1-bit systems. Nevertheless, the conservative version of Lee's rule, $||H||_\infty < 1.5$ [4] was occasionally compromised.

VI. CONCLUSION

To generate a valid noise transfer function (NTF) targeted to a certain performance level and number of quantization levels (N_{LEV}) can be a complex task. This brief suggested a rule that helps to generate NTFs that match the designer's need.

The rule is based on a large group of optimized NTFs followed by simulations. NTFs comply strictly with two norm-based metrics. These are the frequency-domain infinity-norm and the time-domain 1-norm. Simulated N_{LEV} versus two norms results were used to derive a design rule that has impressive accuracy.

The novel rule is based on user-defined N_{LEV} and maximum stable input level to generate norms that the NTF has to comply with. A noise transfer function that realizes such norms can be generated by the Matlab-software available in [15], which is based on the code for the CLANS-procedure available in [14].

REFERENCES

[1] W. Lee, "A Novel Higher Order Interpolative Modulator Topology for High Resolution Oversampling A/D Converters," Master's thesis, MIT, Cambridge, MA, 1987.
[2] B. Agrawal and K. Shenoi, "Design Methodology for ΣΔM," *IEEE Trans. Commun.*, vol. 31, pp. 360 – 370, 1983.
[3] J. Kenney and L. Carley, "Clans: a High-Level Synthesis Tool for High Resolution Data Converters," in *IEEE International Conference on Computer-Aided Design*, 1988, pp. 496–499.
[4] R. Schreier, "An Empirical Study of High-Order Single-Bit Delta-Sigma Modulators," *IEEE Trans. Circuits Syst. II*, vol. 40, pp. 461–466, 1993.
[5] R. Gray, "Quantization Noise Spectra," *IEEE Trans. Inf. Theory*, vol. 36, pp. 1220–1244, 1990.
[6] N. He, N. Kuhlman, and A. Buzo, "Double-Loop Sigma-Delta Modulation with dc Input," *IEEE Trans. Commun.*, vol. 38, pp. 487–495, 1990.
[7] S. Ardalan and J. Paulos, "An Analysis of Nonlinear Behavior in Delta-Sigma Modulators," *IEEE Trans. Circuits Syst.*, vol. 34, pp. 593–603, 1987.
[8] T. Ritoniemi, T. Karema, and H. Tenhunen, "Design of Stable High Order 1-bit Sigma-Delta Modulators," in *IEEE International Symposium on Circuits and Systems*, vol. 4, May 1990, pp. 3267–3270.
[9] R. Schreier and G. C. Temes, *Understanding Delta-Sigma Data Converters*. Hoboken, NJ: John Wiley and Sons, 2005.
[10] J. Kenney and L. Carley, "Design of Multibit Noise-Shaping Data Converters," *Analog Integrated Circuits Signal Processign Journal*, vol. 3, pp. 259–272, 1993.
[11] E. Roverato, M. Kosunen, and J. Ryynänen, "The Synthesis of Noise Transfer Functions for Bandpass Delta-Sigma Modulators with Tunable Center Frequency," in *European Conference on Circuit Theory and Design*, August 2015, pp. 1–4.
[12] I. Løkken, A. Vinje, T. Sather, and B. Hernes, "Quantizer Nonoverload Criteria in Sigma Delta modulators," *IEEE Trans. Circuits Syst. II*, vol. 53, pp. 1383–1387, 2006.
[13] S. R. Norsworthy, R. Schreier, and G. Temes, *Delta-Sigma Data Converters*. New York: IEEE Press, 1997.
[14] R. Schreier. (2000) Delta-Sigma Toolbox. [Online]. Available: http://www.mathworks.com/matlabcentral/fileexchange/19-delta-sigma-toolbox
[15] M. Neitola. (2016) Two-Fom CLANS. [Online]. Available: http://www.mathworks.com/matlabcentral/profile/authors/870073-marko-neitola