Indian Institute of Technology (IIT)
Kharagpur

Summer Internship Report 2015

Quantization Noise in Advanced LIGO Digital Control Systems

Author:
Ayush Pandey
12IE32001
IIT Kharagpur

Supervisor:
Dr. Christopher Wipf
PostDoc Scholar
California Institute of Technology

Department of Electrical Engineering
2015

Preface

.

This document is a result of research and study done at California Institute of Technology as a part of the Laser Interferometer Gravitational Wave Observatory (LIGO), Summer Undergraduate Research Fellowship (SURF) Program in 2015. Quantization Noise and it's effects have been studied and presented in a way that would take the reader without much pre-requisites on the subject matter, from the basics in the first section describing the quantization process to all kinds of analysis and denoising techniques in the very last section and its application to the LIGO detector.
The project was focused on analyzing Quantization Noise in the Advanced LIGO Digital Control Systems. The theoretical background and a thorough literature review on the quantization noise subject (in general) has been presented in the first few sections of this report (sections 1 to 4). A brief description of the LIGO detector and it's working principle is given in section 5. Then, based on the theory developed, results on the quantization noise for Advanced LIGO digital filters, (in section 6) and for the Digital to Analog Converters used in Advanced LIGO (in section 8) have been given. Finally, a denoising technique called Noise Shaping is described to reduce the noise levels due to DACs in a desired frequency band. Simulations and testing results to support the arguments are also given in section 9.

1 What is Quantization

Quantization is a process of mapping continuous set of values into a finite or countably infinite set of values by approximations such as truncation, rounding etc. The function which performs this approximation operation is called a Quantizer. Hence, the input to a quantizer will be a continuous set of values and it will output the quantized values for each sample after performing Quantization. The efficiency of a Quantizer is easily understood to be the accuracy with which it can perform the quantization operation using its approximation technique. The accuracy in this case would simply be a measure of the loss or distortion caused by the Quantizer. This loss in data is known as Quantization Noise or Quantization Error. Intuitively, one can understand that the noise would be equal to the difference between the original input sample and the output of the Quantizer.

1.1 Examples

Though for this project and application we are concerned only by quantization noise in signal processing, but quantization is a well studied topic with respect to various other fields such as image processing, audio processing etc. Apart from these, quantization is something which is observable in daily life as well in the form of digital watches, weight measurements and so on.
In signal processing, quantization can occur when two signals are added, for eg:

$$(1.25) + (2.34500000199999) = 3.5950000012 \tag{1}$$

when the correct answer is 3.59500000199999, giving an error of approximately. The error in multiplication operation can be deduced similarly.

$$Error = 10^{-12} \tag{2}$$

Other than the mathematical operations, quantization noise also occurs on rounding operations such as :

$$round(12.34544567) = 12.3454457 \tag{3}$$

and for truncation of numbers to accommodate numbers in limited precision:

$$truncate(13.456) = 13 \tag{4}$$

Some other examples with respect to quantization in image processing are given in [**Examples**].

2 Modeling Quantization Noise

Let x be the input signal to the quantizer and x' be the output given by the quantizer. Let quantization noise be given by ρ which is the difference between the output (or quantized input) and the input. Quantization operation is a nonlinear operation, shown by the fact that the input-output waveform of any quantizer looks like a staircase (or some other non-linear function depending on the quantizer function). Modeling of such a non-linear ρ can be done in various ways but before that some basic concepts and theorems need to be kept in mind which are mentioned in brief in the following section.

2.1 Quantizing Theorem I and II

Widrow and Kollár in [**Kollar**] have described the statistical theory of Quantization. It has been explained that the quantization noise in a digital control system or for that matter any other digital signal processing application could be modeled by some mathematical algorithm. Two theorems on Quantization Noise from [**Kollar**] are summarized below:
The Quantizing Theorems I states that if the CF (Characteristic Function, $\phi(t)$) of the Probability Density Function (PDF) of the the quantizer input x is band limited such that:

$$\phi_x(t) = 0; |t| > \pi/q \tag{5}$$

where q is equal to 1 Least Significant Bit (LSB), then the CF of the input to the quantizer maybe derived uniquely from the CF of the output x', the same statement follows for PDF of x and x'. In essence, the theorem states that the PDF's of input and output of the Quantizer are uniquely related to each other.
The Quantizing Theorem II on the other hand puts forward an important and a stronger result that the moment of the quantized variable x' (the output of quantizer) is equal to the moment of the sum of the input and a uniformly distributed noise which has a zero mean and mean square equal to $\frac{q^2}{12}$.
The details of these theorem along with their proofs have been covered in detail in [**Kollar**], the book on Quantization Noise.

2.2 PQN Model: Additive Noise Approximation

To model the quantization noise, ρ in a form that is easy to analyze, we take help of the principle from statistics which states that the PDF of the sum, when two statistically independent signals are added together, is equal to the convolution of the PDF's of the individual signals. This implies that the CF of the sum would be the product of the two individual CF's (due to the duality property of fourier transform). Hence, the PDF of Quantizer output x' $f_{x'}(x)$, which is a discrete signal, is equal to the samples of the smooth PDF of the the input signal added with a uniform noise n, $f_{x+n}(x)$. These two PDF's correspond to each other in such a way that the moments of the two are equal when quantizing theorems QT I and QT II are satisfied. (This is the analogy between Sampling and Quantization, that Sampling of a signal is discretization on time scale (X axis) and quantization is discretization of the pdf (Y axis)!). In this case, the quantizer can be replaced by Pseudo Quantization Noise model (PQN), which is an additive uniform noise model. This replacement by additive noise holds to a very good approximation for Gaussian signals for $\sigma > q$ and even for higher values of q ($\sigma = q$). For other distributions, the approximation model is valid to a good approximation for low values of q. (σ is the standard deviation of the PDF of the signal). There are other kinds of modeling techniques for floating point quantization and various other complex modeling techniques. The book [**Kollar**] is being referred to the reader for all such analysis.

2.3 Assumptions

Some assumptions need to be made to enable us to replace the quantization noise source with white an additive PQN model in the control loop. Also, for multiple quantizers in the same loop it is assumed that the two quantization noises are uncorrelated to each other. We shall also assume that the signal is always scaled such that the quantizers are not underloaded or overloaded, thus assuming the absence of limit cycles and highly nonlinear behaviors in our modeling.

3 Noise Sources

3.1 Analog to Digital Controller

In a mixed signal control system, an ADC is used to convert the output analog signal into digital for processing in the digital controller. ADC is a hardware which samples the analog signal values at a frequency which should be more than twice the sampling frequency of the signal (Nyquist Sampling Theorem). After Sampling, the signal is discretized in its amplitude, which is called the Quantization of signal. It is here that the signal is actually converted into discrete values which is then called the converted digital form of the signal. There are various ways to improve upon the design of an ADC for a better quantization noise performance.

Although, ADC performance improvements and rigorous noise analysis is out of scope for this project but basic simulations and a literature review was done to cover completely the quantization noise sources in the control system. The following section describes modeling of an ADC:

Quantizer Model: An ADC Example For an example on Quantization noise in ADCs, [**Quantization**], let us take a ramp input signal. A ramp is an input signal which is a straight line when drawn on a 2D plane, i.e., y=x. One kind of quantizer model can be given by:

$$Q(x) = k.q.sgn(x). \left\lfloor \left| \frac{x}{q} \right| + \frac{1}{2} \right\rfloor \tag{6}$$

where,
k is any arbitrary scaling factor
q is equal to 1 Least Significant Bit (LSB) which shows the resolution of the ADC and,
x is the input to which Q(x) is the output.

The output of the quantizer will be a staircase waveform around the ramp input. The error between input and output can be approximated as (the difference between input and output) shown in Figure 1. Root Mean Square (RMS) of the error can then be derived to finally be able to calculate the SNR (Signal to Noise Ratio) of the ADC which would give an idea about the noise level due to the ADC.

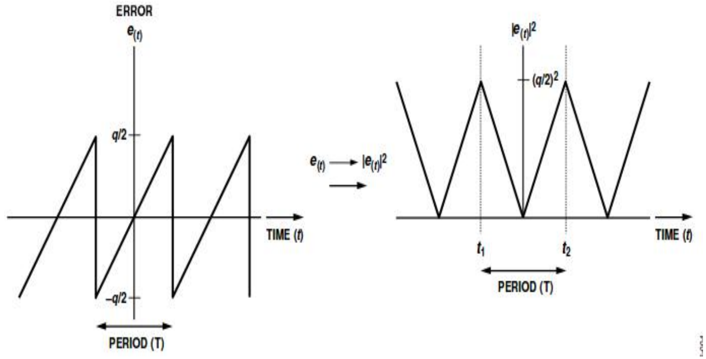

Figure 1: The Approximated Wave Form for the Quantization Error taken from Analog.com

To calculate the RMS we take a time interval $T = t_1 - t_2$ and write the equation of straight line in $y = mx + c$ form as:

$$e(t) - \frac{q}{2} = \left(\frac{\frac{q}{2} - \frac{-q}{2}}{t_2 - t_1}\right).(t - t_2) \tag{7}$$

On simplyifying, we get,

$$e(t) = \left(\frac{q}{T}\right).t + q.\left(\frac{1}{2} - \frac{t_2}{T}\right) \tag{8}$$

For the quantization error the RMS is given by:

$$e_{rms}^2 = \left|e^2\bar{(}t)\right| = \frac{1}{T}.\int_{t1}^{t2} |e(t)|^2\, dt \tag{9}$$

Simplifying,

$$e_{rms}^2 = \frac{1}{T}.\int_{t1}^{t2} \left(\frac{q^2}{T^2}\right).t^2 dt - \frac{1}{T}.\int_{t1}^{t2} q^2.\left(\frac{(t_1 + t_2)^2}{4T^2}\right) dt \tag{10}$$

On solving we get (Skipping the simple algebraic manipulations) that the RMS of the error is given by:

$$e_{rms} = \frac{q}{\sqrt{12}} \tag{11}$$

where q is the LSB of the quantizer representing the resolution of the taken ADC.

On simulation of the above in MATLAB and calculating the RMS, we observe that the RMS of the error is indeed equal to as given in equation 11. The script files for the same are available for reference on the GitHub Repository at the URL mentioned in [**Git**].
Now, to improve the noise performance and analysis for ADCs various techniques have been suggested. Mostly, the inner hardware of the ADC needs to be changed to improve noise performance. Also, dither signal is added to the input to randomize the input signal so that the noise and the input independence is followed to a better approximation. Some analysis on dithering signals[**Pandey**] was also done in this study, but as mentioned previously, ADCs and ADC noise were not the primary focus of the project and hence were not studied in detail.

3.2 Digital Filters

Digital Filters, also known as Compensators in a control system, perform mathematical operations on the input to produce desired outputs. The design is represented in terms of transfer function, say H(z), where z is the z-transform variable. For more on z-transform and transfer fuctions, refer [**Z transform**].
A transfer function is a unique representation for given position of poles and zeros on the z-plane. The implementation of a given transfer function can be done in infinitely different ways, depending on different State Space descriptions or the so-called different SOS (second order sections) matrices. Various different filter structures are described in detail in [**Oppenheim**]. The quantization noise, our primary concern in this project, also depends on the filter form realization. Intuitively, by different realization of filter (the filter structure), we basically mean different order in which the mathematical operations (additions, multiplications etc.) are performed by the filter.

Structure The following section details about some of the most commonly used filter structures along with their important features.

1. Direct Form I: The direct form I is an implementation of the difference equation describing the filter, as is. The roundoff noise for this realization has been found to be much more compared to other structures which has been proved in [**Oppenheim**]. The advantage of having direct form I structure is in the applications where least hardware complexity is required. DF I leads to lowest chip area required

for implementation on a Digital Signal Processor [**Rahmanian**].

2. Direct Form II : The number of multiplications and delay registers are least for this form. [**Oppenheim**] shows the quantization noise analysis for fixed point implementation. DF II form minimizes the coefficient sensitivities of the filter with respect to the quantization error which is proved in [**Rahmanian**].

3. State-Space Representation of Digital Filters is used widely to realize low noise form structures. The major disadvantage of using low noise form filters described in state space notation, is that it requires a very high computational time to realize these filters, i.e. a complexity of $\mathbb{O}((N+1)^2)$ for an Nth order filter[**T L Chang**], while others such as DF2 are realized in linear time complexity.

3.3 Digital to Analog Converter

Truncation A DAC converts the digital signal from the controller to analog, to be fed to the actuators or other analog parts of the control systems. This conversion introduces quantization noise due to the limited precision of the DAC hardware. Hence, if the precision of the DAC is p bits for an n-bit precise digital signal ($n > p$), the n-bit number is truncated to p-bit precision before giving it to the DAC for conversion to analog. This truncation operation (decrease in precision) leads to quantization noise called the DAC Quantization Noise.
The DAC is hence a major source of quantization noise in the control system since usually p is very less than n. Commonly, n=64 while p=16 or 18.
The noise analysis and the measurement of DAC noise have been described in a later section.

4 Noise Analysis

It is trivial that quantization noise will be lower for a system that has a higher resolution. A higher resolution implies that the value of 1 LSB or q is lower. Another trivial fact is that the maximum quantization error in one quantization operation would always be less than or equal to half of one LSB i.e. $\pm\frac{q}{2}$. The following sections describe in detail the noise analysis for digital filters.

4.1 Digital Filter Noise Analysis

There are some basic trade-offs in the design of filter which are mentioned as follows:

1. The Hardware Complexity in the Design of the filter (the chip area required of a DSP), though this doesn't play any role if specialized DSP chips are not in use and implementation is being done on a computer.
2. The time complexity, i.e. the computational time the filter takes in calculations and the complexity of the structure algorithm.
3. Quantization noise level in the filter and its effects.
4. Sensitivity to disturbances and perturbations of coefficients.

[**Kaiser**] showed for direct form filters that if poles and zeros are tightly clustered on the z-plane, then even small coefficient quantization errors may cause large shifts in the position of poles and zeros and hence changing the response of the filter and even tending to become unstable.

Fixed Point Precision Noise Analysis [**Oppenheim**] describes in detail the noise in fixed-point precision calculations. Quantization noise in fixed point precision is both well researched and not relevant to this project because the calculations are being performed in floating point in the digital controllers being studied in this project, and hence the details are being skipped here and [**Oppenheim**] is referred for all such analysis.

Floating-Point Precision Noise analysis For floating point representations, a key point to keep in mind is that the quantization noise can no more be assumed to be independent of the input signal. In fact, the noise is directly dependent on what input is being given to the filter. Also, the noise cannot be assumed to be white in this representation and hence the noise analysis in floating point precision is a difficult task. An advantage due to the use of an exponent in the floating point representation is that the Overflow condition is eliminated and hence the complexity of the structure reduces as we don't need to care about the overflow during the calculations. The book by Widrow and Kollár on Quantization Noise in Chapter 12, [**Kollar**] looks at floating point quantization in depth and provides detailed floating-point noise analysis. Some direct formula to calculate quantization noise level in different filter structures have been given in [**Matts**]. The presentation [**Matts**] shows that noise is lower in a state space representation of a filter realized analogous to the analog biquad filter, compared to the

Direct Form II (which is widely considered as the best filter structure). This "biquad filter" involves one more addition operation than the DF2 filter and hence is just a little more computationally complex than DF2. This filter structure having low noise has been referred to as a "biquad filter" in [**Matts**]. For floating point precision some very important results are available in the literature already existing. [**Zeng**] mentions two theorems proving that the floating point quantization noise level is independent both of the ordering of sections in a cascade form and the ordering of poles and zeros between sections. The paper assumes that ordering of calculations and the poles and zeros within a section remain the same throughout.

Any filter can be represented in the general State variable representation as:

$$x_{k+1} = Ax_k + Bu_k \tag{12}$$
$$y_k = Ax_k + Du_k \tag{13}$$

On a given transformation T, to achieve low noise structure, we have

$$T_{k+1} = A'T_k + B'u_k \tag{14}$$
$$y_k = C'T_k + Du_k \tag{15}$$

where $A' = T^{-1}AT; B' = T^{-1}B; C'^t = C^tT$
To implement a filter as shown above, a time complexity of $\mathbb{O}((N+1)^2)$ is required and hence these filters are not widely used. Moreover, as shown in [**Chang**], error feedback could be used to achieve super low noise forms. This is hard to implement practically as we don't have the error signal accessible to feed back. [**Mullis**] shows how the error feedback helps in minimizing the error. Another important result that comes from the literature is that when extra bits are used in floating point representation for the accumulation register, the optimization methods developed for fixed point analysis can be applied 'as is' for floating point representation[**Bomar**]. This is one of the reasons why [**Dehner**] can be used for optimization using reordering of poles and zeros for floating point even though the analysis given in [**Dehner**] is for fixed point precision.

4.2 Noise Analysis in Frequency Domain

All signals are sampled at a particular frequency 'fs'. The noise is random but its characteristics can be pinned down on proper analysis in the frequency domain. Fourier analysis reveals the frequency components that a signal is made up of [**Oppenheim**]. The power spectral density (PSD) [**Cerna**] gives a major piece of information as the SNR can be calculated directly by taking the ratio of the PSDs of the output and the noise signals. The approach to calculate PSD involves taking the Fourier transform of the autocorrelation fucntion. There are various ways which affect the analysis and are described in detail in [**Kanner**]. The power spectrum density for the noise, the input and the output is calculated using Welch's method [**Welch**] and plotted on the same log log plot for proper analysis.

5 LIGO

The Laser Interferometer Gravitational Wave Observatory (LIGO) [**LIGO**] aims to detect gravitational waves [**GWD**] which would give us an opportunity to get access to completely new and exciting astronomical insights and scientific research. Just like electromagnetic waves, gravitational waves have been predicted to have a frequency spectrum. The study of this frequency spectrum of gravitational waves along with many other new frontiers could lead to a completely different perception about science and astronomy. The existence of gravitational waves was first predicted by Einstein in his theory of general relativity. The detection (or the failure of it!) would help in establishing a stronger background in this field and would even prove (or disprove!, as the case maybe) Einstein's theory. Though the latter is almost impossible, as indirect detection of Gravitational waves has been done but LIGO aims to directly detect the gravitational waves. Other than this, if Gravitational waves are detected, various new information would be available about a host of different kind of astrophysical bodies. Apart from all these exciting results one can expect from LIGO, it also boasts of largest sustained ultra-high vacuum in the world (8x the vacuum of space) and being the most sensitive detector ever built.

5.1 Basic Working Principle

The LIGO setup consists of a Laser interferometer [**Interferometer**] which is at the heart of detection of the gravitational waves. Gravitational waves have the property of stretching and squeezing the space-time through which they pass. This stretch and squeeze in the space-time changes the distance (length of the interferometer arm) between two points. This displacement produced is measured as strain, which transforms into the constructive or destructive interference between the original wave and the reflected laser light (which is stretched & squeezed in space-time by presence of a gravitational wave). To achieve reflection, a mirror is kept at the other end of the interferometer in both of its perpendicular arms [**Interferometer**]. There are many controllers which are used to control and suppress the mirror motion. [**Carbone**] describes why movement of the mirrors and hence the motion control is necessary. Since, the strain measurement being done is of the order of 10^{-19} metres, all kinds of noises and disturbances need to be analyzed and looked into closely so as to maximize the Signal to Noise Ratio (SNR), which is of prime importance in the process of Gravitational wave detection of such a low magnitude (the astronomical strains being measured are one thousand times lesser than the diameter of a proton!).

5.2 LIGO and Quantization Noise

Quantization noise, is one such noise source among thousands others and this project was dedicated towards the analysis, study and improvements with respect to Quantization Noise. The problem of quantization noise is important to be solved for LIGO because it may be possible that at some places in the digital controllers the noise level might be so high that the signal wouldn't be detected leading to some very important signals being suppressed under the noise.

5.3 LIGO Filter Design

For the Advanced LIGO digital controllers the filter design was upgraded to a low noise form, in iLIGO, previously the digital filters were implemented in the direct form II (DF2) structures pertaining to its lower number of additions and multiplications. But, as described above, state-space representations lead to low noise forms, increasing the computational time in the process. [**Matts**] went on this line and suggested the use of a kind of biquad filter derived from state-space representation but only using one addition extra compared to DF2, in the process. The noise analysis shown in [**Matts**] proves that this indeed is a better choice for the digital filter structure as it provides for great increase in the SNR compared to the computation time penalty suffered. The aLIGO upgrade changed all digital filters from DF2 form to the low noise form as suggested by [**Matts**].

5.4 LIGO DAC

The aLIGO DAC is a 18-bit DAC. Off-the-shelf DACs are usually 16-bit precise DACs, but pertaining to the higher precision needs of the LIGO detector and the controller, an 18 bit DAC is in use in the aLIGO. This 18-bit DAC is made using a 16-bit and a 2-bit DAC together. These DACs work on integer 18-bit precision hence the floating values which are 64 bit precise are truncated to 18-bit precise integers. The quantization error during this operation is a big problem for various kinds of filter designs of aLIGO and hence analysis and mitigation techniques are important.

6 Part I: Filter Noise

6.1 Estimation

All digital controllers in the Advanced LIGO setup specifically use the floating point double precision representation of numbers. A good analysis has been done by [**Matts**] for floating-point representation wherein, calculations for various digital filter forms have been provided.
To measure single precision noise [**Martynov**] suggested a way, by taking the difference between the single precision output and the double precision signal output. The double precision output noise is so less compared to the single precision noise that the difference output is equal to single precision quantization error. Hence, in essence, approximating double precision to be a perfect representation (i.e. with no quantization error). Since, all digital filters are already implemented in double precision in the LIGO setup, single precision noise is of no use. To obtain double precision noise, in this code, an extrapolation is done to estimate the approximate quantization noise occurring in digital filters implemented in double precision. The approximation method has been described by Denis in [**Martynov2**]. An empirically obtained factor equal to 10^{-7} is multiplied to the quantization noise values obtained for single precision. But, as is obvious that this is not the best way to achieve the estimation and isn't very foolproof.
So, this project improves upon the already existent technique by calculating the same output using the long double precision which is better than double for most [**longdouble**] compilers, and then subtracting the two outputs, which would result in the quantization noise occurring in double precision filter implementation directly.

6.2 Analysis

For digital filter noise analysis, the following improvements were made to the previously existing analysis tool developed by Denis Martynov [**Martynov**].

1. Accurate Noise Calculation

2. SNR Warning message(s) on screen and SNR Plots
3. Fast and Optimized Code: takes far less time compared to previous implementation.

One complete software tool was developed by incorporating all the changes and improvements described above which could automatically analyze quantization noise for all the digital filters. The software tool is a MATLAB function which can be called without giving any arguments. Once called, it starts checking all the filters one by one on its own until the end of the list. So, basically analyzing all the digital filters is just one click away. The list that this function looks into is a list of filter bank module names mentioned in the Foton's description of the filters.
NDS servers were used to login remotely to the sites to analyze the digital filters of the aLIGO sites at Livingston and Hanford. Proper log files were generated for errors along with the PSD plots which were saved to the disk for future reference. The complete saved collection of vector graphics of the plots is available for reference at [**Collection**].

6.3 Results

Testing at the 40m Interferometer Prototype The tool developed was run to check all the digital filters implemented in the 40m Interferometer at Caltech.

Testing at aLIGO sites Some results:

1. More than 90% of the filters are "safe". The quantization noise PSD levels for these filters are orders of magnitude below the output PSD level. This type of response was observed for the complete frequency range except at very high frequency (near the twice of Nyquist frequency) where the output of the filter is usually rolled off to lower magnitudes. Some assorted results for some of the filters from both aLIGO sites are shown in the figure 2 and the ones following it. Figure 4 shows the SNR distribution. This type of SNR was common for these kinds of filters, realized in the Low Noise Form.

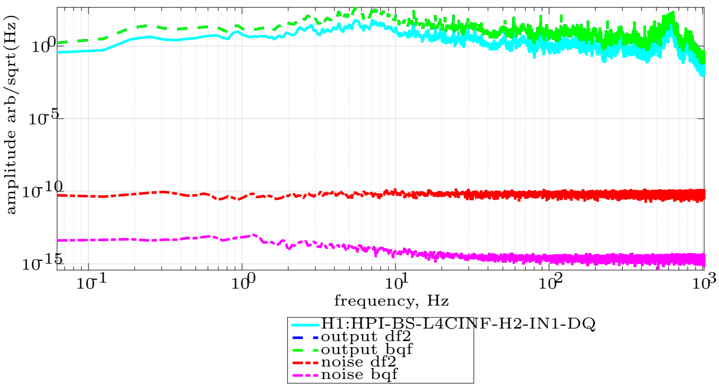

Figure 2: Analysis for Hanford HPI BS filter

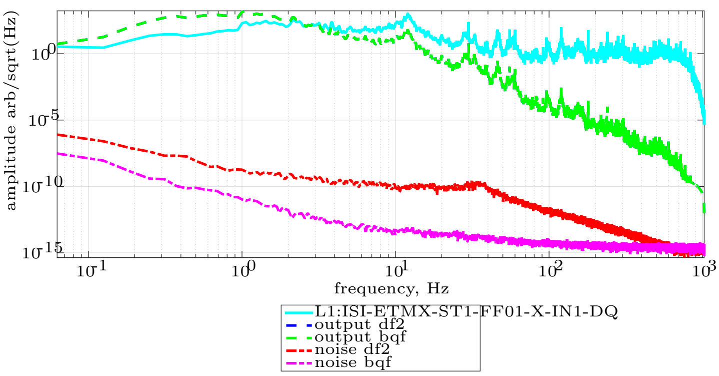

Figure 3: Analysis for Livingston ISI ETMX filter

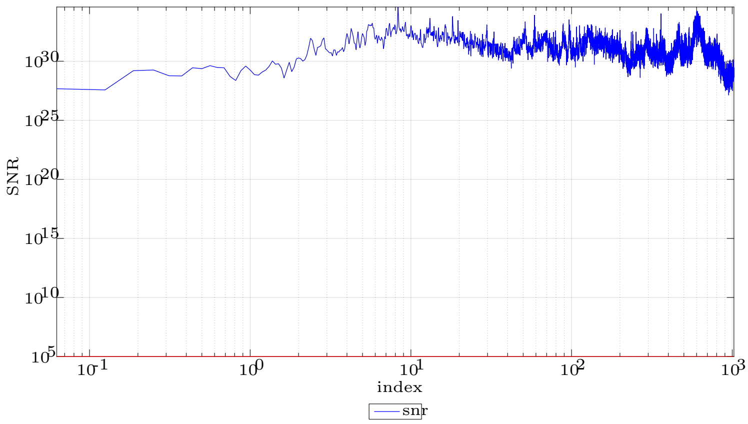

Figure 4: SNR distribution for HPI ITMY filter

2. For a few filters, the DF2 quantization noise level is even above the output for a range of frequencies. Though this is an alarming observation, but this does not affect the aLIGO digital control system at all because all digital filters have been implemented in the low noise form. This result further strengthens and proves the results shown by [**Matts**] and others. For these filters, the low noise form quantization noise level is considerably lower and the filter is safe. Some results are shown in the figure 5 and the ones following it.

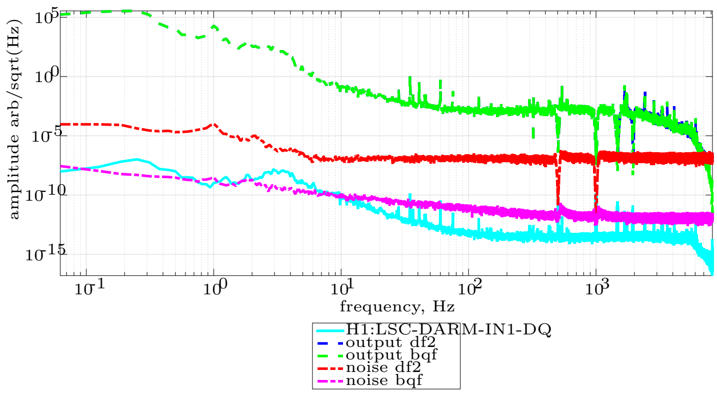

Figure 5: Analysis for Hanford LSC DARM filter

3. For some filters, the DF2 performs equally well as the LNF. This result

is shown in figure 6 and the ones following it. The reason behind this observation is clear from the figures which show the input and output signal as well. The output for these kinds of filters are mostly representing filters which only perform a gain operation (i.e. a simple multiplication).

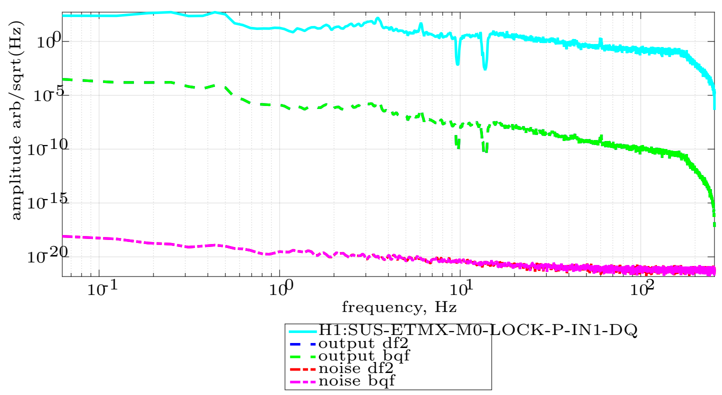

Figure 6: Analysis for Hanford SUS ETMX filter

7 Limitations

Some limitations of the analysis are:

1. Only Data Acquisition Channels: As all work is being done remotely, hence only the channels for which the data is stored on a hard disk is available. So, only the channels ending in _DQ are being analyzed.

2. Only Recorded Input Channels analyzed.

3. User friendliness of the tool isn't that great yet but is being improved and certainly it would be a good tool to check quantization noise in the coming time.

4. Memory Size being used by the software tool is higher.

8 Part II : DAC Noise

8.1 Estimation and Analysis

DAC noise is easy to estimate in the code as access to both the input and the quantized output is available. A block diagram showing the measurement procedure is given in figure 7.

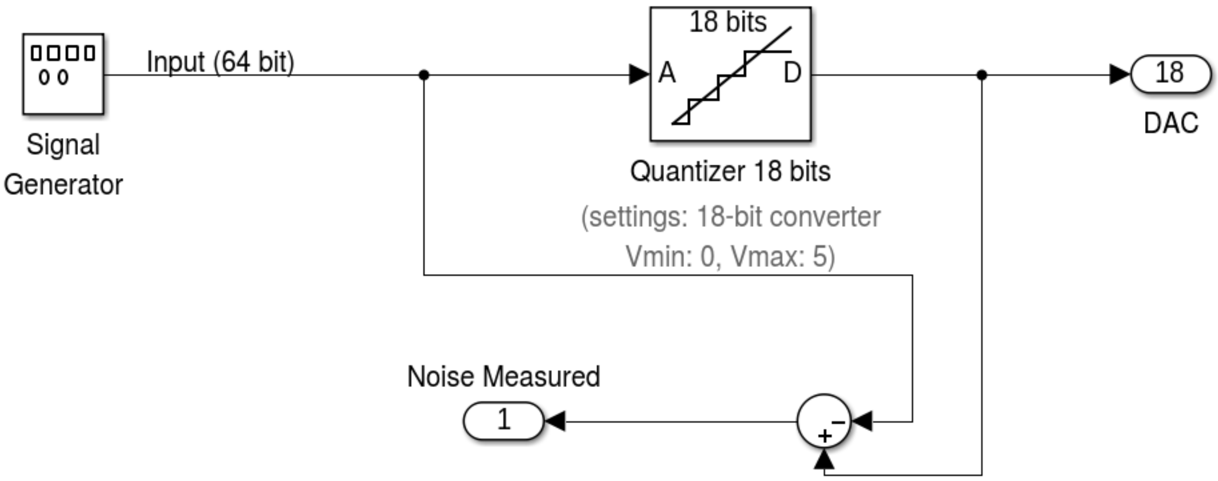

Figure 7: DAC Noise Measurement

The difference between the two gives us the noise. This noise can then be analyzed in the frequency domain, like before, by plotting the PSD for the input, output and the noise.
As described earlier, that the DAC noise is a major limiting factor in the digital controller as it truncates from a highly precise 64 bits to 18 bit integer precision. This fact was observed in the analysis, as the DAC noise floor was observed to be well above the digital filter noise floor.
The code developed for DAC noise analysis and visualization on MATLAB is available on [**Git**] and can be used for DAC noise analysis for all kinds of signals.

8.2 Results

The already known or rather predicted DAC noise problem, was realized again through the DAC noise analysis done in this project. The DAC noise floor is well above the output level and is limiting the signals for a wide range of frequencies for some signals. A good DAC denoising technique is needed to prevent such high noise levels.

8.3 DAC denoising Techniques

Some techniques existent in the Digital Signal processing theory to improve DAC performance are :

1. DAC Hardware Improvement
2. DAC Noise Shaping
3. DAC of higher precision

9 Noise Shaping

Noise Shaping is a technique used to modify the frequency spectrum of the error signal in such a way that the noise power of the spectrum is more in the undesirable frequency band which leads to a higher SNR in the desirable frequency band. This technique is perfect for our implementation as the frequency band of interest for gravitational wave detection is fixed and is < 100Hz. Hence, with the same DAC hardware, very low noise levels in this frequency band can be achieved using Noise shaping. An algorithm was developed to implement noise shaping technique in the DAC code.

9.1 Algorithm

A general example Consider the given system (see figure 8) where the input (denoted by x) which is double precision, is fed into the quantizer. The quantizer rounds it off and feeds to the DAC according to the DAC's resolution, hence incurring quantization error. To shape the quantization error, e (quantization error) is fed back as shown. The output of the quantizer is denoted by x' and the following equations explain how noise shaping is achieved.

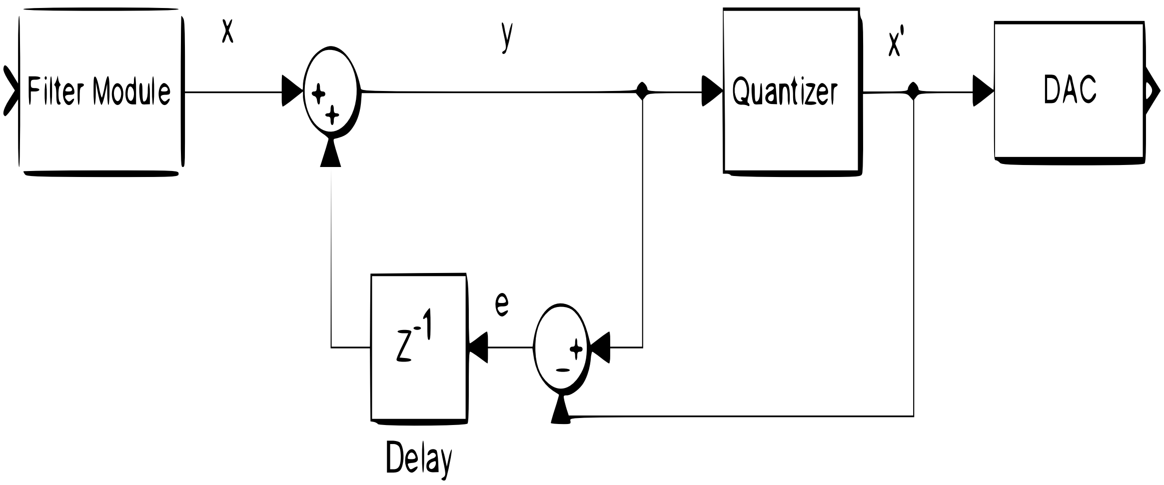

Figure 8: An explainatory Block Diagram Example for Noise Shaping in DAC

From the block diagram (in time domain), we have,

$$y[n] = x[n] + e[n-1] \tag{16}$$
$$e[n] = y[n] - x'[n] \tag{17}$$

We have,

$$x'[n] = y[n] - e[n] \tag{18}$$

Now taking Z-Transform, we have

$$Y(z) = X(z) + \frac{E(z)}{z} \tag{19}$$
$$X'(z) = Y(z) - E(z) \tag{20}$$

Now from equations 19 and 20, we get the final result which shows how quantization error is added to the input to form the output x', and since we have fed the error back we will see how the quantization noise is now shaped according to the transfer function we chose (in this case 1/z):

$$X'(z) = X(z) + E(z)(1 - \frac{1}{z}) \tag{21}$$

The noise is shaped by a factor of $\frac{z-1}{z}$ which has a zero at z=1 and a pole at z=0. This means that at the frequency corresponding to z=0, the gain will be high and at frequency corresponding to z=1 the gain will be low since a zero is occurring there. In essence, we have a one-pole digital filter in front of us resulting due to the feedback of the error. In this way, noise shaping can be achieved to increase the SNR in the desired frequency band by choosing any arbitrary shaping filter (with some restrictions).

9.2 Generalized Noise Shaping Algorithm for aLIGO DAC

Using the above mentioned concept, a generalized noise shaping algorithm was developed and simulated on MATLAB. This noise shaping algorithm was designed such that it provided the elegance to the user by providing the option to design any arbitrary noise shaping filter to achieve any arbitrary shape of the noise [**Nentwig**]. So, for example, if a designer fancies the noise to be very low for a particular notch design in the filter design, then the noise shaping filter can be designed to be a notch filter of that frequency. This would lead to noise being shaped such that the noise would be very low at the given notch frequency. In this way, any arbitrary shape could be given to the noise, which is a very useful result and something which could be very helpful in various ways.
The way this is achieved is shown below: After the delay element in the above description, if a H_{shaper} filter is put, then following similar analysis arbitrary noise shape could be achieved. The block diagram for the same is shown in the figure 9.

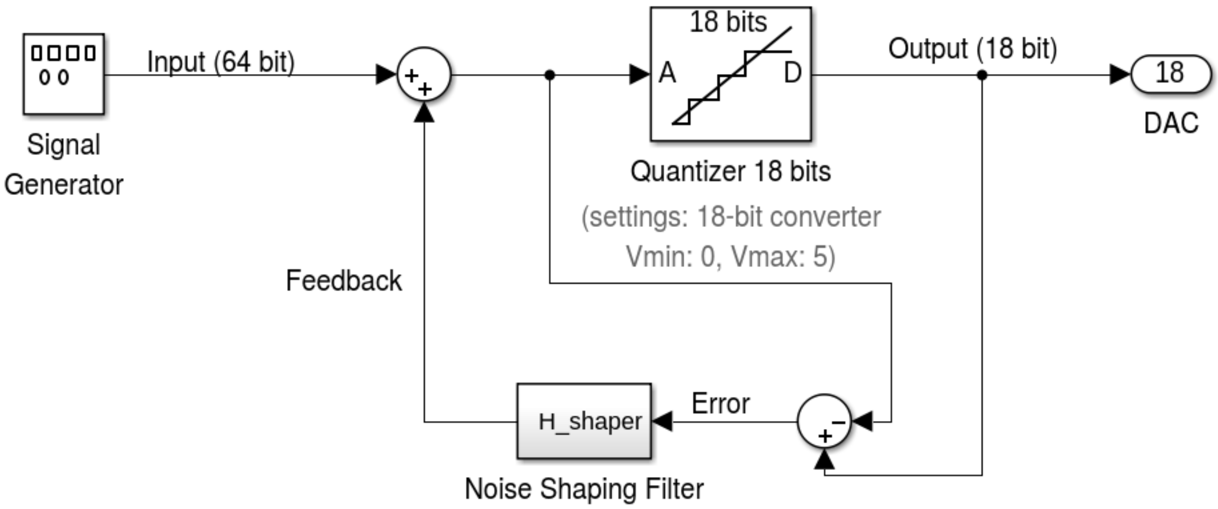

Figure 9: Generalized Noise Shaping Block Diagram

Following the analysis shown above, the following expression results:

$$X(z) = X(z) + E(z)(-1 + H_{shaper}(z)) \tag{22}$$

$H_{shaper}(z)$ allows for arbitrary filter design, and hence arbitrary noise shaping of the quantization noise.

9.3 Applications

General Applications Noise shaping is a widely used technique in Digital Signal Processing (DSP) applications. One of the fields making use of noise shaping extensively is audio processing. The audio industry has progressed greatly, partly due to such DSP techniques. In digital audio, it is applied as a bit-reduction scheme. The quantization is spread according to the frequencies ear is more sensitive to. This leads in more pleasant sounding audio as noise is removed from it.
Apart from wide applications in the audio industry, the noise shaping technique is being used more and more in the modern ADCs. Along with this, the noise shaping technique is used in video and image processing as well. In these applications, the noise shaping is done in combination with dither. One such description for image processing is given in [**Christou**].

Possible applications at LIGO As mentioned before, the noise shaping technique could be a very useful application for LIGO digital controllers as well because the GW detection is done at low frequencies, i.e. below 100 Hz. Hence, the DAC noise can be pushed out of this band using the noise shaping technique. Also, the generalized noise shaping technique presented could find very wide and varied applications especially in freedom of filter design.

9.4 Simulation

The algorithm was implemented for simulation on MATLAB. The code developed was tested for various different filter designs for H_{shaper}. One such, notch filter design is shown in figure 10. The elegancy of the method is visible in the noise spectrum which is almost zero for the given notch frequency of the noise shaping filter. In this way, the noise can be shaped as desired. Also is interesting to observe in the given spectra is the noise level without any noise shaping. The comparison between the two spectra clearly accounts for the fact that the overall noise level is more when it is shaped.

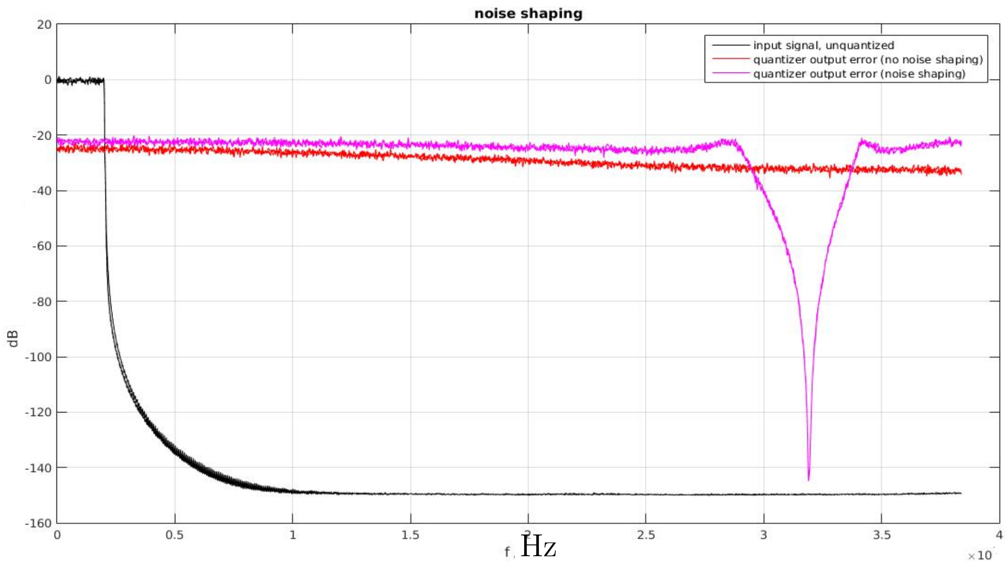

Figure 10: Notch Shaped Quantization Noise

A more applicable filter would be a high pass filter for implementation in LIGO. The figure 11 shows how the quantization noise level is very low for lower frequencies which is compensated by very high noise level in the higher frequency band.

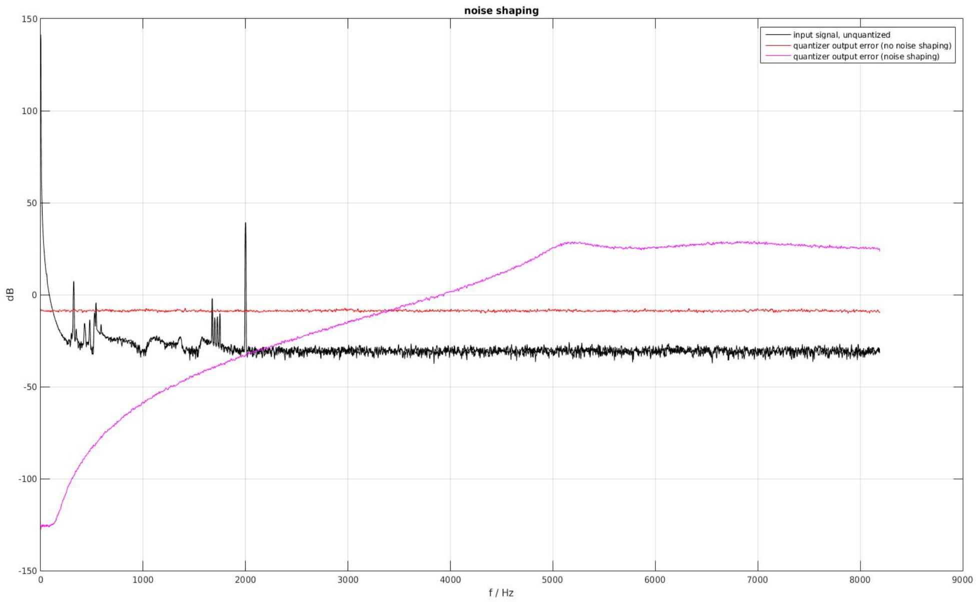

Figure 11: High Pass Shaped Quantization Noise : Good for GW Detection

10 Conclusions and Future Work

10.1 Part I: Digital Filters

A favorable conclusion and a positive note is that more than 90% of the filters don't have any problem with quantization noise level, when the filters are realized in the LNF structure.
The complete analysis is available at [**drive**]. It has all the plots for the thousands of filters analyzed using the software tool. There still remains a lot to be done with respect to achieving a complete digital controller analysis as only the signals recorded to the disk were analyzed in the project. The future work would demand a complete in-depth analysis of all filters and all kinds of signals.

10.2 Part II: DAC

The DAC noise problem was a known fact, and the denoising algorithm in the name of noise shaping presented could be a really helpful technique not just for DAC denoising but also for various other kinds of noise shaping requirements that the filter designer might want. The future work requires a complete development and implementation of the noise shaping technique in the real time code to reap advantages of the results shown in the simulations.

References

[1] Tutorials Point, http://www.tutorialspoint.com/dip/Concept_of_Quantization.htm

[2] B. P. Abott et. al., *LIGO: the laser Interferometer Gravitational Wave Observatory* http://dx.doi.org/10.1088/0034-4885/72/7/076901

[3] *On Gravitational Wave Detection:* https://nodus.ligo.caltech.edu:30889/wiki/doku.php?id=gw_detection_101_for_surf

[4] Frederick J Raab, *Overview of LIGO Instrumentation*

[5] L. Carbone et al., *Sensors and Actuators for the Advanced LIGO Mirror Suspensions* http://arxiv.org/pdf/1205.5643v1/

[6] Widrow and Kollár, *Quantization Noise* http://www.mit.bme.hu/books/quantization

[7] Dehner, *Noise Optimized Filter Design* https://dx.doi.org/10.1016/s0165-1684(03)00075-6

[8] Denis Martynov, *Checking Digital System* http://github.com/denismartynov/quantization http://nodus.ligo.caltech.edu:8080/40m/?mode=full&attach=1&reverse=0&npp=510&Author=den&subtext=digital

[9] Software Tool Code, http://github.com/ayush9pandey/quantization

[10] Bomar, *Roundoff Noise Analysis of State-Space Digital Filters implemented on Floating-Point Digital Signal Processors*

[11] T. L. Chang, *A Unified Analysis of Roundoff Noise Reduction in Digital Filters*

[12] Bing Zeng, *Analysis of Floating-Point Roundoff errors using dummy multiplier coefficient sensitivities*

[13] S. Rahmanian, *An Optimal Structure for Implementation of Digital Filters*

[14] Kaiser, *An approach to implement Digital Filters*

[15] Chang, *Comparison of Roundoff Noise variances in several low round off noise digital filter structures*

[16] Matthew Evans, *Digital Filter Noise*

[17] C T Mullis, *Roundoff Noise in digital filters: Frequency transformations and invariants*

[18] Widrow, *Chapter 12, Basics of Floating-Point Quantization, Quantization Noise Book*

[19] Oppenheim and Schafer, *Discrete-Time Signal Processing*

[20] Quantization of Signals, http://en.wikipedia.org/wiki/Quantization_(signal_processing)

[21] DAC Denoising http://github.com/rxa254/DACdenoising/

[22] Ayush Pandey, *Progress Report I : Quantization Noise in Digital Control Systems*

[23] Wikipedia : Long Double Precision, https://en.wikipedia.org/wiki/Long_double

[24] Jonah Kanner https://dcc.ligo.org/LIGO-G1500862

[25] Michael Cerna and Audrey F. Harvey (2000) *The Fundamentals of FFT-Based Signal Analysis and Measurement*

[26] Welch (1967), *The Use of Fast Fourier Transform for the Estimation of Power Spectra: A Method Based on Time Averaging Over Short, Modified Periodograms* , IEEE Transactions on Audio Electroacoustics, AU-15, 7073.

[27] Z-Transform, http://www.ele.uri.edu/Courses/ele541/tutorials/ztransforms/

[28] Denis Martynov, http://nodus.ligo.caltech.edu:8080/40m/?mode=full&attach=1&reverse=0&npp=510&Author=den&subtext=digital

[29] Ayush Pandey, *A complete collection of all plots for all digital filters at Hanford and Livingston* https://dcc.ligo.org/T1500351-v1

[30] Markus Nentwig, http://www.dsprelated.com/showarticle/184.php

[31] Cameron Nicklaus Christou,*Optimal Dither and Noise Shaping in Image Processing*