EFFECTS OF QUANTIZATION NOISE IN DIGITAL FILTERS

Bernard Gold and Charles M. Rader
Lincoln Laboratory, Massachusetts Institute of Technology*
Lexington, Massachusetts

GENERAL EXPRESSIONS FOR QUANTIZATION NOISE

If a discrete time linear system, hereafter called a digital filter, is programmed on a digital computer or realized with digital elements, computational errors due to finite word length are unavoidable. These errors may be subdivided into three classes, namely, the error caused by discretization of the system parameters, the error caused by analog to digital conversion of the input analog signal, and the error caused by roundoff of the results which are needed in further computations. The first type of error results in a fixed deviation in system parameters and is akin to a slightly wrong value of (say) an inductance in an analog filter. We shall not treat this problem here; it has been treated in some detail by Kaiser.[1] The other two sources of error are more complicated but if reasonable simplifying assumptions are made they can be treated by the techniques of linear system noise theory.[2] It is our aim to set up a model of a digital filter which includes these two latter sources of error and, through analysis of the model, to relate the desired system performance to the required length of computer registers.

Both analog to digital conversion and roundoff may be considered as noise introducing processes, very similar in nature. In each case a quantity known to great precision is expressed with considerably less precision. If the digitized or rounded quantity is allowed to occupy the nearest of a large number of levels whose smallest separation is E_0, then, provided that the original quantity is large compared to E_0 and is reasonably well behaved, the effect of the quantization or rounding may be treated as additive random noise. Bennett[3] has shown that such additive noise is nearly white, with mean squared value of $E_0^2/12$. Furthermore the noise is reasonably assumed to be independent from sample to sample, and roundoff noises occurring due to different multiplications should be independent. It is possible to show pathological examples which disprove each of these assumptions, but they are reasonable for the great majority of cases. Ultimately our results must rest on experimental verification, of course.

Since the noise of A-D conversion is assumed independent of the noise created by roundoff, we can compute the output of any filter due to either excitation alone, or due to the signal alone, and combine them to get the true filter output (of course the noise terms are known only statistically); therefore, we will begin by finding an expression for the mean squared output of an arbitrary filter excited by a single noise source. Let the filter function be $H(z)$; it is understood that $H(z)$ is the transfer function between the output of the filter and the node where noise is injected; $H(z)$ may thus be different from the transfer function between the filter's normal input and output. Let us thus consider the

*Operated with support from the U.S. Air Force.

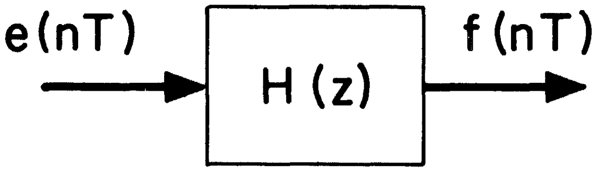

Figure 1. Random noise applied to a filter.

situation of Fig. 1, where a given noise sequence $e(nT)$ is applied to $H(z)$, resulting in an output noise sequence $f(nT)$.

We can conveniently examine this model using the convolution sum. Thus,

$$f(nT) = \sum_{m=0}^{n} h(mT)e(nT - mT) \qquad (1)$$

where $h(mT)$ is the inverse z transform of $H(z)$. The input noise $e(nT)$ is presumed to be zero for $m < 0$ and the system is initially at rest. Squaring Eq. (1) yields

$$f^2(nT) = \sum_{m=0}^{n} \sum_{l=0}^{n} h(mT)h(lT) \times e(nT - mT)e(nT - lT) \qquad (2)$$

Now, if $e(nT)$ is a random variable with zero mean and variance σ^2, and recalling our assumption that $e(nT)$ is independent from sample to sample, the statistical mean of Eq. (2) reduces to

$$E[f^2(nT)] = \sigma^2 \sum_{m=0}^{n} h^2(mT) \qquad (3)$$

For a system for which the right side of (3) converges, the steady state mean squared value of $f(nT)$ can be obtained by letting n approach infinity. For this case, a formula which is usually more convenient can be obtained in terms of the system function $H(z)$. Noting the definition.

$$H(z) = \sum_{m=0}^{\infty} h(mT)z^{-m} \qquad (4)$$

of the z transform, we can form the product $H(z) H\left(\frac{1}{z}\right)z^{-1}$ and, by performing a closed contour integration in the z plane within the region of convergence of both $H(z)$ and $H\left(\frac{1}{z}\right)$, arrive at the identity

$$\sum_{m=0}^{\infty} h^2(mT) = \frac{1}{2\pi j}\oint H(z)H\left(\frac{1}{z}\right)z^{-1}dz \qquad (5)$$

Either the right- or left-hand side of (5) may be used to evaluate the steady state mean squared value of $f(nT)$.

EXAMPLE—FIRST ORDER SYSTEM

As an example, consider the first order system of Fig. 2. Let the analog-digital conversion noise $e_1(nT)$ have variance σ_1^2 and the roundoff noise $e_2(nT)$ have variance σ_2^2. The system function $H(z)$ of Fig. 2 is given by $1/(1 - Kz^{-1})$ and $h(mT) = K^m$. The output $y(nT)$ can be expressed as the sum of a signal term $y_0(nT)$, caused by $x(nT)$, and a noise term $f(nT)$, whose mean squared value can be written, from (3), as

$$E[f^2(nT)] = (\sigma_1^2 + \sigma_2^2) \sum_{m=0}^{n} (K^m)^2 \qquad (6)$$

from which the steady state value can be instantly written as

$$\sigma_n^2 = \lim_{n\to\infty} E(f^2(nT)) = \frac{(\sigma_1^2 + \sigma_2^2)}{1 - K^2} \qquad (7)$$

The implications of Eq. (7) are tricky. The mean squared value of the noise clearly increases as K approaches unity. The maximum gain of the filter also increases (the gain of the system of Fig. 2 at dc is $(1/(1 - K))$. For this filter with low frequency input the signal power to noise power ratio (S^2/N^2) is proportional to $(1 + K)/(1 - K)$ which approaches infinity as the pole of the filter approaches the unit circle. This is a general result. However, with a finite word length, the input signal must be kept small enough that it does not cause overflow

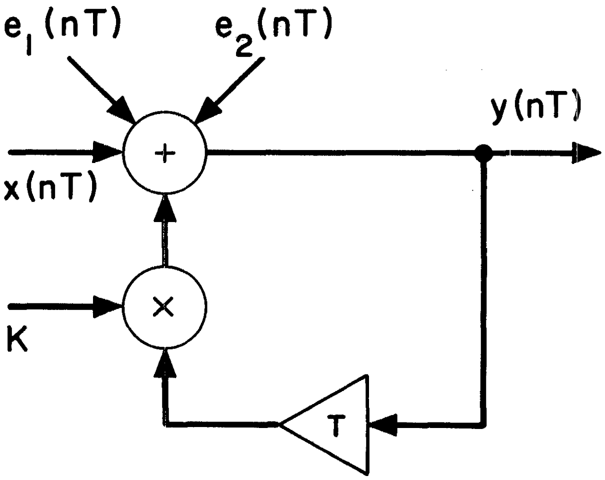

Figure 2. Noise mode for first order system.

in the computation. Thus, the obtainable signal-to-noise ratio decreases as K approaches unity. Clearly, each case deserves its own considerations, as the signal-to-noise ratio in the filter depends very much on the actual conditions of the use of the filter.

Finally, we comment that the cases $K = 0$, $K = 1$, in Eq. (7) are unique because σ_2 becomes zero since no multiplications are performed.

EFFECT OF DIFFERENT REALIZATIONS OF THE SAME FILTER

There are a variety of ways of programming a second order digital filter (or in general a filter with more than two singularities). Suppose a particular system function $H(z)$ is desired. If quantization is ignored, then only the relative speed and memory requirements of the different methods are of interest in deciding which way to use. However, Kaiser's work shows that the truncation of system constants affects different realizations differently, and may in fact lead to instability in some realizations. The noise effects described here also yield different results for different programming configurations. The point is illustrated through the examination of the two systems of Fig. 3. Fig. 3a represents a noisy programmed realization of the difference equation:

$$y(nT) = 2r\cos bTy(nT - T) - r^2y(nT - 2T) + x(nT) - r\cos bTx(nT - T) \quad (8)$$

and Fig. 3b represents the pair of simultaneous difference equations:

$$\left.\begin{aligned} w(nT) &= x(nT) + 2r\cos bT\,w(nT - T) \\ &\quad - r^2w(nT - 2T) \\ y(nT) &= w(nT) - r\cos bT\,w(nT - T) \end{aligned}\right\} \quad (9)$$

Both systems have the transfer function

$$H(z) = \frac{1 - r\cos bT z^{-1}}{1 - 2r\cos bT z^{-1} + r^2z^{-2}}$$

By examination of the poles and zeros of $H(z)$ in Fig. 4, we see that our network behaves as a resonator tuned to the radian center frequency b for the sampling interval T.

In Figs. 3a and 3b, $X(nT)$ represents the noiseless input to the filter, $e_1(nT)$ represents the noise due to A-D conversion of the input, and $e_2(nT)$ represents the noise added by rounding. The roundoff noise can be caused either by a single roundoff after all products are summed, or by the sum of the roundoff error due to each of the multiplications. It is simpler to program the latter, but more noise is created. Note that, while in the realization of Fig. 3b the noise terms $e_1(nT)$ and $e_2(nT)$ are injected into the filter at the same place as the input $X(nT)$ and thus see the same transfer function $H(z)$, in Fig. 3a the noise term $e_2(nT)$ is injected in a different part of the filter and sees a different transfer function:

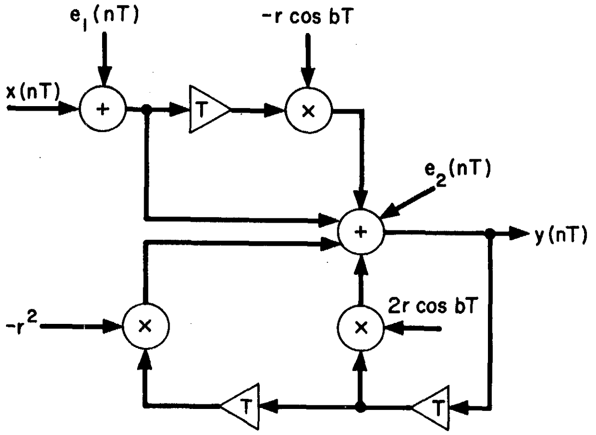

Figure 3a. Noise model for second order system—direct realization.

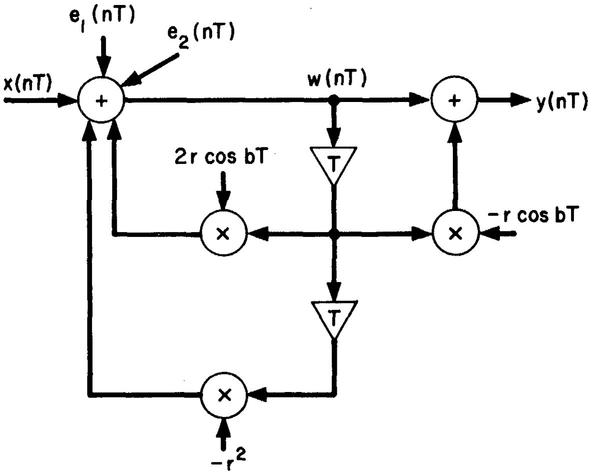

Figure 3b. Noise model for second order system—canonical realization.

$$H_1(z) = \frac{1}{1 - 2r\cos bT z^{-1} + r^2z^{-2}} \quad (10)$$

Thus we can expect that the noise due to $e_2(nT)$ will be different for the filters of Figs. 3a and 3b.

Considering first the realization of Fig. 3a, we can, after some manipulation, obtain the result,

$$\sigma_n^2 = \sigma_1^2u_1(r,bT) + \sigma_2^2u_2(r,bT) \quad (11)$$

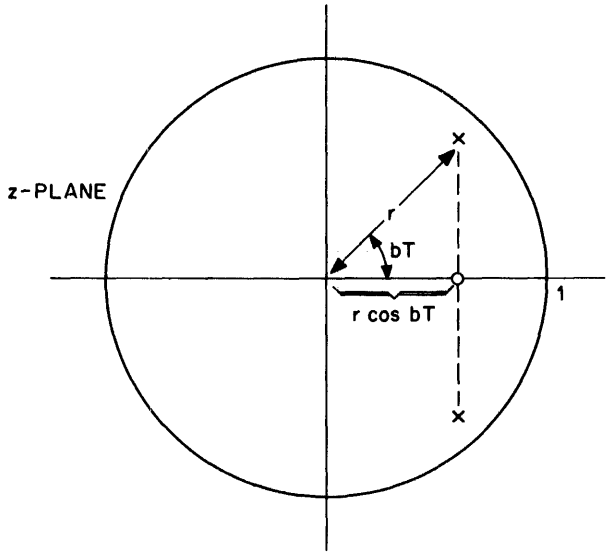

Figure 4. Pole zero representation of Eqs. (8) or (9).

where σ_1^2 and σ_2^2 are the variances of $e_1(nT)$ and $e_2(nT)$, and with

$$u_2 = \frac{1 + r^2}{1 - r^2} \times \frac{1}{r^4 + 1 - 2r^2 \cos 2bT}$$

and

$$u_1 = \frac{1}{1 - r^2}\left[1 - \frac{r^2 \sin^2 bT(1 + r^2)}{r^4 + 1 - 2r^2 \cos 2bT}\right]$$

More insight can be obtained into these results by letting $r = 1 - \epsilon$ and allowing ϵ to be quite small, of the order of 0.05 or less. Then (11) reduces to the simple form

$$\sigma_n^2 = \frac{1}{4\epsilon}\left[\sigma_1^2 + \frac{\sigma_2^2}{\sin^2 bT}\right] \qquad (12)$$

Carrying through a similar computation for the realization of Fig. 3b yields

$$\sigma_n^2 = (\sigma_1^2 + \sigma_2^2)u_1(r,bT) \qquad (13)$$

which can also be reduced, for small values of ϵ, to

$$\sigma_n^2 = \frac{1}{4\epsilon}[\sigma_1^2 + \sigma_2^2] \qquad (14)$$

Several important facts can be deduced from Eqs. (12) and (14). First, the so-called "straightforward" realization of Fig. 3a leads to increased noise for low resonance frequencies whereas the "canonic" realization of Fig. 3b does not. Physically, this result can be explained by noting that, in the straightforward realization, the noise "passes through" only the poles of the filter, so that at low frequencies, the complex conjugate poles interact to form a low pass filter. In the "canonic" realization the noise is also filtered by a zero which is close to dc and thus the output noise is of a band-pass nature and less total energy is able to pass through the filter. Second, we note that Eqs. (13) and (14) have the same functional dependence on pole positions, namely, that the mean squared output noise is inversely proportional to the distance from the pole to the unit circle and therefore directly proportional to the gain of the filter.

From these results one can, for example, estimate the word length needed for a simulation requiring many filters. One such system is a vocoder synthesizer shown in Fig. 5. Typically, a vocoder synthesizer will contain about 100 resonators. Assuming that the noise from each resonator is additive to the noise from all other resonators and picking an effective average ϵ of 0.01, we arrive at a total noise output of about 7 or 8 bits. It is clear that word lengths of at least 20 bits are needed to avoid audible noise outputs superimposed on the vocoder generated synthetic speech.

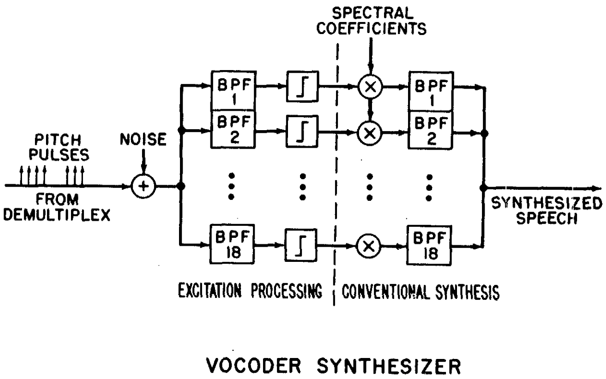

Figure 5. Vocoder synthesizer.

EXPERIMENTAL VERIFICATION FOR FIRST AND SECOND ORDER FILTERS

The results of the preceding computations were experimentally verified by programming various realizations of first and second order difference equations on the TX-2 digital computer. To perform a measurement of output noise for a given digital filter, the computations were performed with rounded arithmetic using a 36-bit word, and simultaneously, using rounded arithmetic with a shorter word and exactly the same input. The outputs of the two filters were subtracted, squared, integrated and divided by the number of iterations of the equation. The inputs to the filters were random noise or sampled sinusoids. The filters were programmed using the PATSI[4] compiler, and the various waveforms of interest, including the mean squared output noise, were displayed during the computation. The measurement was taken when the mean squared output noise seemed to reach a steady value, or in the case of the very high gain filters, when the patience of the observer was exhausted. As we shall see below, the necessary observation time for confidence in such a measurement is highly dependent on the gain of the filter.

Figure 6 shows the predicted and measured output noises for some one-pole filters, as Eq. (7), with $\sigma_1^2 = 0$. The horizontal axis is the pole position and the vertical axis is the mean squared output noise normalized to σ_2^2. Table 1 gives the predicted versus measured output noises for several two-pole filters (no real zeros) with various pole positions, along with the measurement error. All of the results seem to confirm the theory.

It is advisable to determine, on a statistical basis, the measurement time required before the variance of such statistical observations is sufficiently small. Thus, consider a random variable q defined as

$$q = \frac{1}{n} \sum_{m=0}^{n} f^2(nT) \tag{15}$$

where $f(nT)$ is an output noise signal as indicated in Fig. 1 due to a set of mutually independent input noise samples $e(nT)$.

Assuming $f(nT)$ to have zero mean, we can immediately perceive that the mean value of the measurement q is given by

$$E[q] = \sigma_f^2 \tag{16}$$

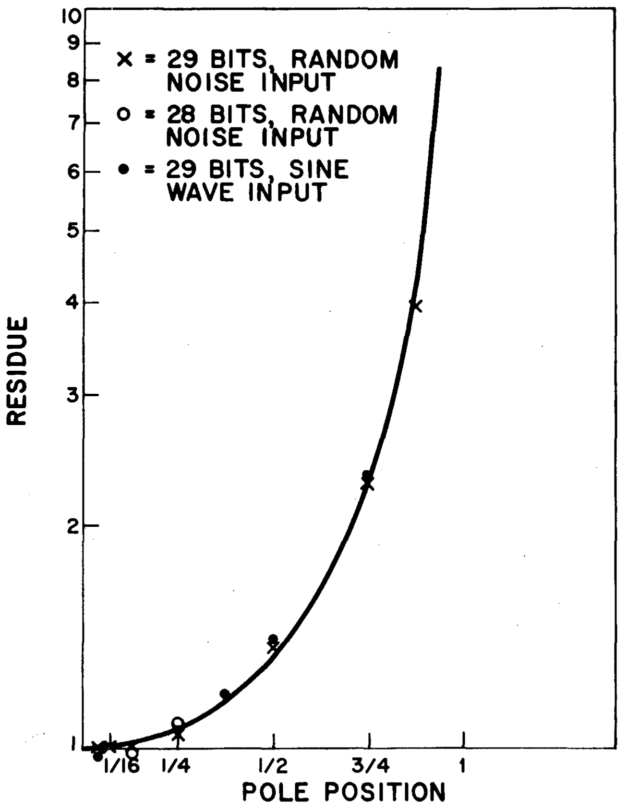

Figure 6. Predicted vs measured quantization noise for first order system.

Table 1. Two-Pole Filter Noise Measurement

Mean Squared Output Noise Predicted	 Measured	Error %	Pole Positions
204	203	0.49%	.5 ± .5j
289	297	2.77	.5 ± .707j
508	520	2.36	.5 ± .778j
1011	1058	4.65	.75 ± .56j
2824	2880	1.98	.875 ± .332j
5553	5933	6.40	.90625 ± .235j
5553	5503	0.90	.90625 ± .235j
11014	11450	3.96	.921875 ± .169j
11014	11079	0.59	.921875 ± .169j
3306	3740	13.12	.75 ± .654j
3306	3359	1.60	.75 ± .654j

where σ_f^2 is the variance of the (stationary) random variable $f(nT)$. Now assuming that $f(nT)$ is a set of stationary Gaussian variables with correlation coefficient $\rho(rT)$, then it can be shown that[5]

$$E[f^2(mT)f^2(lT)] = \sigma_f^4 + 2R^2(mT - lT) \qquad (17)$$

where R^2 is defined as the covariance between $f(mT)$ and $f(lT)$. From Eqs. (16) and (17), we arrive at the expression for the variance of q,

$$\sigma_q^2 = E[q^2] - E^2[q]$$

$$= \frac{2}{n^2}\sum_{m=0}^{n}\sum_{\ell=0}^{n} R^2(mT - lT) \qquad (18)$$

This can be evaluated for first order system of Fig. 2. For that case $R(mT - lT) = K^{2|m-l|}$ and, for large n, Eq. (18) reduces to

$$\sigma_q^2 = \frac{4\sigma_e^4}{n(1 - K^2)^3} \qquad (19)$$

where σ_e^2 is the variance of the input $e(nT)$ as in Fig. 1. Of major interest in determining the time needed to perform the measurement is the ratio of the standard deviation to the mean of q. Using an argument similar to the one that leads to Eq. (7) we can for the first order system relate σ_e^2 to σ_f^2 by the formula $\sigma_f^2 = \sigma_e^2/(1 - K^2)$, which combined with Eqs. (19) and (16) yields

$$\frac{E[q]}{\sigma_q} = \frac{(1 - K^2)\sqrt{n}}{2} \qquad (20)$$

Thus, for example, if $K^2 = 0.99$, we need 10^8 terms in the measurement of Eq. (15) in order to reduce the standard deviation of the measurement to 2% of the mean of the measurement. Assuming that an iteration could be done in 100 μsec, 10^4 seconds would be required for such accuracy.

NOISE CONSIDERATIONS IN PROGRAMMING ITERATIVE SINE WAVE GENERATORS

One must be especially attentive to noise considerations in the programming of iterative sine wave generators. Various efficient routines exist to compute the sine or cosine of a random argument rapidly, but for instances where the argument is nT for successive integers n, the most efficient way to generate sinusoidal functions is by the use of iterative difference equations. These are, of course, digital filters with poles directly on the unit circle, inputs equal to zero, and initial conditions which specify the magnitude and phase of the output. Since the poles of the filter are directly on the unit circle, the noise, according to Eq. (12) or (14) becomes infinite. This is indeed the situation.* The saving feature is the *gradual* increase of the noise term, so that if one runs the program for a limited time, or periodically resets the initial conditions, catastrophe can be avoided. To study this problem theoretically, consider the simultaneous difference equations

$$\left.\begin{aligned} y(nT + T) &= \cos bT\, y(nT) \\ &\quad + \sin bT\, x(nT) \\ x(nT + T) &= -\sin bT\, y(nT) \\ &\quad + \cos bT\, x(nT) \end{aligned}\right\} \qquad (21)$$

with initial conditions $x(0) = 1$, $y(0) = 0$. The "circuit" is shown in Fig. 7.

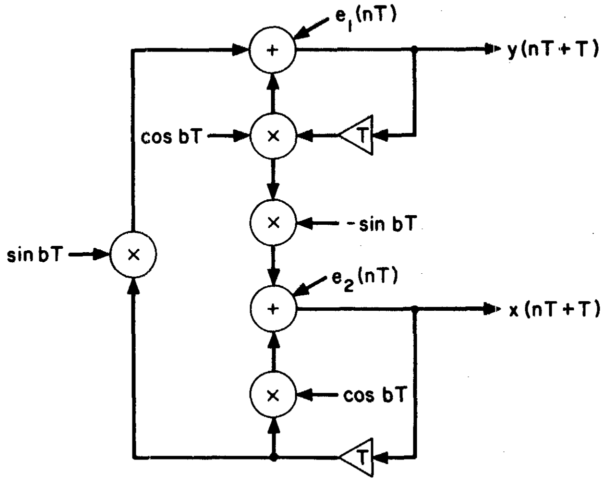

Figure 7. Iterative sine and cosine generator.

The z transform $X(z)$ of one output $x(nT)$ can be written

$$X(z) = \frac{z^2 - z\cos bT + zE_2(z) - \cos bTE_2(z) - \sin bTE_1(z)}{z^2 - 2z\cos bT + 1} \qquad (22)$$

We see that the first two terms of the numerator correspond to the signal and the remaining terms to the noise, $E_1(z)$ and $E_2(z)$ being respectively the z

*Various nonlinearities can be introduced to keep the noise finite. This is adequate for many applications since the selectivity of the filter can be relied on to keep the output spectrally pure even if the phase of the output is unpredictable.

transforms of the added noises $e_1(nT)$ and $e_2(nT)$, both introduced by roundoff error.

Defining:

$$h_1(nT) = Z^{-1}\left\{\frac{z - \cos bT}{z^2 - 2z\cos bT + 1}\right\}$$
$$h_2(nT) = Z^{-1}\left\{\frac{-\sin bT}{z^2 - 2z\cos bT + 1}\right\} \qquad (23)$$

where Z^{-1} is the inverse z transform, we can from Eq. (3) write the total noise as

$$E(f^2(nT)) = \sigma_1^2 \sum_{m=0}^{n} h_1^2(nT) + \sigma_2^2 \sum_{m=0}^{n} h_2^2(nT) \qquad (24)$$

Solving Eq. (23) explicitly and letting $\sigma_1^2 = \sigma_2^2 = \frac{E_0^2}{12}$ we arrive at the result

$$E(f^2(nT)) = \frac{E_0^2}{12}\left\{\sum_{m=0}^{n} \cos^2(nbT - bT) + \sin^2(nbT - bT)\right\} = \frac{E_0^2}{12}\, n \qquad (25)$$

Notice that is was impossible to use Eq. (5), since the result obtained would be infinite and thus no time-dependent result could be formulated. Equation (25) tells us that the noise increases linearly with the number of iterations of the difference equations. For example, after 10^6 iterations, the noise is about 10 bits. Assuming that one iteration is performed in 100 μsec, several minutes could certainly pass, even in an 18-bit machine, before the generated sine and cosine waves begin to look noisy.

Another program for generating a cosine wave is expressed by the iteration

$$y(nT + 2T) = 2\cos bT\, y(nT + T) - y(nT) \qquad (26)$$

with initial conditions $y(0) = 1$, $y(T) = \cos bT$. Noise analysis of Eq. (26) leads to a functional dependence of the mean squared noise, of the form $\frac{n}{\sin^2 bT}$; thus appreciably greater quantities of noise are generated at low frequencies, and fewer iterations are available before the program becomes unusable.

The comparison of Eqs. (21) and (26) was performed qualitatively on TX-2 by programming identical sine wave generators using both methods. For all frequencies, the method of Eq. (21) produced sinusoids of more nearly constant amplitude than the method of Eq. (26), but this difference in behavior was negligible for frequencies greater than one fourth of the sampling frequency, and, using 36-bit arithmetic, the distortions were almost unobservable for these frequencies. For low frequencies (of the order of one thousandth of the sampling rate) the method of Eq. (26) was completely unusable, with the generated sine wave being terribly distorted in the first period.

REFERENCES

1. J. F. Kaiser, "Some Practical Considerations in the Realization of Linear Digital Filter," 3rd Allerton Conference (Oct. 20–22, 1965).

2. J. B. Knowles and R. Edwards, "Effect of a Finite-Word-Length Computer in a Sampled-Data Feedback System," *Proc. IEEE,* vol. 112, no. 6, (June 1965).

3. W. R. Bennett, "Spectra of Quantized Signals," *Bell System Technical Journal,* vol. 27, pp. 446–472 (July 1948).

4. C. M. Rader, "Speech Compression Simulation Compiler," *J. Acoust. Soc. Am.* (A), June 1965.

5. J. L. Lawson and G. E. Uhlenbeck, *Threshold Signals,* MIT Rad. Lab. Series 24, McGraw-Hill, New York, 1950.