A Normalized Digital Filter Structure

AUGUSTINE H. GRAY, JR., MEMBER, IEEE, AND JOHN D. MARKEL, MEMBER, IEEE

Abstract—A normalized digital filter structure is presented, based upon an orthonormal polynomial expansion. This structure is recursively designed, has several predictable stability properties in the presence of time-varying parameters, and appears to have roundoff noise properties which are superior to other known filter structures, particularly in the presence of clustered poles. Each section of the filter can be precisely implemented by one complex multiply.

I. Introduction

In a previous paper [1], we presented a procedure for synthesizing several new digital filter structures through the use of an orthogonal polynomial expansion. A brief history of these polynomials, attributed to Szego [2], may be found in Kailath [3]. They appear to have been first applied in least-squares estimation, at least implicitly, by Levinson [4] in a recursion relationship. Applications in digital filtering were extensively studied during 1954–1957 by the Geophysical Analysis Group at the Massachusetts Institute of Technology. An account of this study is contained in Robinson [5], [6]. The first explicit application of orthogonal polynomials to more general recursive digital filters was apparently first suggested by Itakura and Saito [7].

Further study by the authors has resulted in an orthonormal polynomial structure which is optimal in the sense that all nodes have unity energy, or unity L_2 scaling norms [8]. Thus, with respect to a unit sample input, input scaling is not required, internal overflow cannot occur (since all coefficients are bounded by unity), and the dynamic range of the node energies or norms is minimal. This filter will be referred to as the normalized structure because of its orthonormal properties.

It is also the first digital filter structure we are aware of that can be proven stable (with two different stability definitions) in the case of time-varying filter parameters. For this case, the classical result of Szego concerning location of the zeros of the orthogonal polynomials [2] cannot be used to demonstrate stability, for in the case of time-varying parameters, z transforms and polynomials cannot be used to describe the system.

Manuscript received February 1, 1974; revised October 2, 1974. This work was sponsored by the Department of Defense.

A. H. Gray, Jr. is with the Department of Electrical Engineering and Computer Science, University of California, Santa Barbara, Calif. 93106.

J. D. Markel is with the Speech Communications Research Laboratory, Santa Barbara, Calif. 93109.

It is conjectured that the normalized structure will be most useful in the case where bandwidths are narrow and poles are clustered. Preliminary results indicate that the normalized form has roundoff noise properties superior to the best commonly known structure, the parallel 1-P form [8], with the improvement increasing as the poles become more clustered. This has been recently demonstrated, both theoretically and experimentally, with a number of digital elliptic filters [9].

This paper is separated into five sections, ordered for a logical flow of mathematics. Section II summarizes the synthesis procedure as an elementary extension or the orthogonal polynomial expansion [1]. Section III demonstrates the stability results for the case of time-varying parameters. Section IV treats the scaling considerations for fractional arithmetic and includes a roundoff noise analysis of a simple second-order form to illustrate the advantages of the normalized form. The results are summarized in Section V.

II. Synthesis of the Normalized Form

Let $G(z)$ represent the transfer function of a stable rational filter having the form

$$G(z) = \frac{P_M(z)}{A_M(z)} = \frac{\sum_{m=0}^{M} p_{M,m} z^{-m}}{1 + \sum_{m=1}^{M} a_{M,m} z^{-m}}. \tag{1}$$

It has been shown [1] that the filter can be implemented through a recursively generated parameter set which results in an expansion of the transfer function numerator as a sum of orthogonal polynomials. The recursion relations generate two sets of orthogonal polynomials, $A_m(z)$ and $B_m(z)$ for $m = 0, 1, \cdots, M$, a set of k-parameters, k_m for $m = 0, 1, \cdots, M-1$, a set of tap parameters, ν_m for $m = 0, 1, \cdots, M$, and a set of polynomial norm squares, α_m for $m = 0, 1, \cdots, M$.

The filter transfer function is implemented through the recursion relations

$$A_m(z) = A_{m+1}(z) - k_m B_m(z) \tag{2a}$$

and

$$z\,B_{m+1}(z) = k_m A_m(z) + B_m(z) \tag{2b}$$

with the boundary conditions

$$A_0(z) = z\,B_0(z) = 1. \tag{3}$$

The numerator of the transfer function, $P_M(z)$, can be expanded in terms of the polynomials $z\,B_m(z)$ giving

$$G(z) = \sum_{m=0}^{M} \frac{\nu_m\, z\, B_m(z)}{A_m(z)}. \tag{4}$$

To generate different orthogonal polynomial structures, we define a set of pi-parameters, π_m (with $\pi_M = 1$), in order to define modified polynomials and tap parameters,

$$\hat{A}_m(z) = \pi_m A_m(z),\ \hat{B}_m(z) = \pi_m B_m(z),\ \hat{\nu}_m = \nu_m/\pi_m. \tag{5}$$

Equation (4) can therefore be rewritten as

$$G(z) = \sum_{m=0}^{M} \frac{\hat{\nu}_m\, z\, \hat{B}_m(z)}{A_m(z)}. \tag{6}$$

Assume that the filter $G(z)$ is driven by a sequence $\{x(n)\}$ whose z transform is $X(z)$. Define the sequences $\{x_m^+(n)\}$ and $\{x_m^-(n)\}$ as those having z transforms $\hat{A}_m(z)X(z)/A_M(z)$ and $\hat{B}_m(z)X(z)/A_M(z)$, respectively. The output sequence $\{y(n)\}$ will have the z transform $Y(z) = G(z)X(z)$.

These definitions can be combined with (2) to give the recursive implementation

$$x_m^+(n) = [\pi_m/\pi_{m+1}]\, x_{m+1}^-(n) - k_m x_m^-(n) \tag{7a}$$

$$[\pi_m/\pi_{m+1}]\, x_{m+1}^-(n+1) = k_m x_m^+(n) + x_m^-(n). \tag{7b}$$

The boundary conditions are given by

$$x_M^+(n) = x(n) \quad \text{and} \quad x_0^-(n+1) = x_0^+(n), \tag{8}$$

and from (6), the output sequence is given by

$$y(n) = \sum_{m=0}^{M} \hat{\nu}_m x_m^-(n+1). \tag{9}$$

The implementation of (9) is shown in Fig. 1, where the separate blocks $G_m(z)$, $m = 0, 1, \cdots, M-1$ are used to implement the relations of (7). The two-multiplier formulation [1] is defined by the trivial case $\pi_0 = \pi_1 = \cdots \pi_M = 1$. In that case, (7) is directly implemented in each section, with two multiplications per section as shown in [1, Fig. 2].

If (7a) is used to eliminate $x_m^+(n)$ from (7b) one finds

$$x_{m+1}^-(n+1) = k_m x_{m+1}^+(n) + [\pi_{m+1}/\pi_m]\,(1 - k_m^2)\, x_m^-(n). \tag{10}$$

Using (7a) and (10), one arrives at the one-multiplier implementations of [1, Fig. 4] if

$$\pi_m/\pi_{m+1} = 1 + \epsilon_m k_m \qquad \text{with } \pi_M = 1 \tag{11}$$

and each ϵ_m is plus or minus 1. The parameters $\epsilon_0, \epsilon_1, \cdots, \epsilon_{M-1}$ are called sign parameters. The choice of sign parameters to optimize scaling has previously been discussed [1].

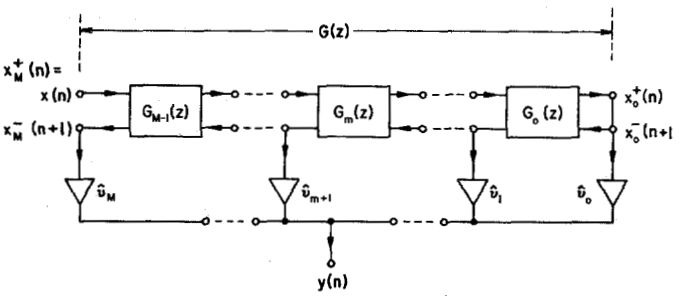

Fig. 1. Block diagram implementation of the orthogonal polynomial filters.

The normalized structure is synthesized by choosing the pi-parameters so as to make the polynomials $z\,\hat{B}_m(z)$ orthonormal. As α_m represents the norm square of $z\,B_m(z)$, dividing by $\sqrt{\alpha_m}$ will result in polynomials having unit norm. For the normalized structure we then take

$$\pi_m = 1/\sqrt{\alpha_m}. \tag{12}$$

From [1, eq. (15)] these pi-parameters can be seen to satisfy the recursion relation

$$\pi_m = (1 - k_m^2)^{1/2}\pi_{m+1} \qquad \text{with } \pi_M = 1. \tag{13}$$

Using (13) with (7a) and (10) we obtain the recursion equations for the normalized form

$$x_m^+(n) = [1 - k_m^2]^{1/2} x_{m+1}^+(n) - k_m x_m^-(n) \tag{14a}$$

$$x_{m+1}^-(n+1) = k_m x_{m+1}^+(n) + [1 - k_m^2]^{1/2} x_m^-(n). \tag{14b}$$

The implementation of these relations is shown in Fig. 2. The filter is called the normalized form, since for a unit sample input, all node energies or norms are unity, i.e.,

$$\sum_{n=0}^{\infty} [x_m^+(n)]^2 = \langle \hat{A}_m(z), \hat{A}_m(z) \rangle = 1, \tag{15a}$$

and

$$\sum_{n=0}^{\infty} [x_m^-(n)]^2 = \langle \hat{B}_m(z), \hat{B}_m(z) \rangle = 1. \tag{15b}$$

If an angle ϕ_m is defined as the inverse sine of k_m, then

$$\sin \phi_m = k_m, \tag{16a}$$

and

$$\cos \phi_m = [1 - k_m^2]^{1/2}. \tag{16b}$$

If (16) are used in (14), then the separate filter sections effect a rotation since each section takes a vector with components $x_{m+1}^+(n)$ and $x_m^-(n)$ and rotates it through an angle ϕ_m to obtain a new vector with components $x_m^+(n)$ and $x_{m+1}^-(n+1)$. A rotation of 0 rad makes the section into a straight feedthrough section, with the exception of the delay in the lower

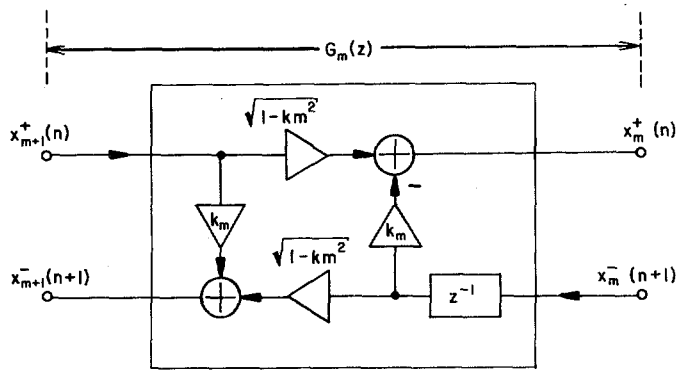

Fig. 2. Implementation of $G_m(z)$ using the normalized structure.

branch. A rotation of $\pm\pi/2$ rad essentially opens the section, isolating the right ports.

The rotation concept makes the normalized form ideally suited to existing signal processing computers which carry out complex multiplication as basic arithmetic operations. To see this more readily, (14) and (16) can be combined with Euler's formula

$$e^{i\phi m} = \cos\phi_m + i \sin\phi_m$$

where $i^2 = -1$, to note that each section is implementable using one complex multiply as

$$[x^+_m(n) + i\, x^-_{m+1}(n+1)] = [x^+_{m+1}(n) + i\, x^-_m(n)]\, e^{i\phi m}. \quad (17)$$

Further significance of the rotation is considered in the remaining sections.

III. Filter Stability with Time-Varying Parameters

The normalized structure appears to be unique in that two time-varying parameter stability relations can be shown by the simple requirement that the k-parameters are bounded in magnitude by a number less than 1. While it is not true in general that stability requirements for a time-invariant system automatically imply stability when the parameters are allowed to vary, this is the case for the normalized form.

To demonstrate stability relationships, the notation of filter energy for the normalized structure is introduced. Assume now that the filter defined by (9) and (14) or Figs. 1 and 2, now has parameters k_0, $k_1, \cdots, k_{M-1}$, $\hat{\nu}_0$, $\hat{\nu}_1, \cdots, \hat{\nu}_M$, that may be time-varying functions of n. If the modified tap parameters remain bounded, stability of the filter will depend only upon the stability of the interconnected sections.

The energy of a discrete signal is usually defined as the sum of the squares of the time sequence. The net energy flowing into a single filter section from index $n = 0$ through $n = N - 1$ can then be thought of as given by ΔE_m where

$$\Delta E_m = \sum_{n=0}^{N-1} ([x^+_{m+1}(n)]^2 - [x^-_{m+1}(n+1)]^2 - [x^+_m(n)]^2 + [x^-_m(n+1]^2).$$

Squaring and adding (14) allows the above equation to be rewritten as

$$\Delta E_m = \sum_{n=0}^{N-1} [x^-_m(n+1)]^2 - [x^-_m(n)]^2$$

$$= [x^-_m(N)]^2 - [x^-_m(0)]^2. \quad (18)$$

Equation (18) can be considered as analogous to a conservation of energy equation in continuous systems theory if the two terms on the right of (18) are used to represent stored energy of the filter section at times $n = N$ and $n = 0$. Thus $[x^-_m(n)]^2$ can be referred to as a stored energy of a filter section and (18) simply indicates that the increase in the energy of the filter section is equal to the net input signal energy. Pursuing this analogy further, the total filter energy $E(n)$ (at each instant) must be the sum of the stored energy in each section, i.e.,

$$E(n) = \sum_{m=0}^{M-1} [x^-_m(n)]^2. \quad (19)$$

Since $x^-_m(n)$ is the output of the delay element in section m, the filter energy of (19) can be thought of as being stored in the m separate delay elements. Squaring and summing (14), and then applying (19), leads to the conservation of energy equation for the entire filter,

$$E(n+1) - E(n) = [x(n)]^2 - [x^-_M(n+1)]^2. \quad (20)$$

From (20) it is noted that the only way in which energy can be added to the filter is through the input, $x(n)$, and the only way in which energy can be removed is through the node $x^-_M(n+1)$.

Considering the energy in the separate sections to be simply the square of the stored values in the delays, it can be seen that the energy is moved through the sections diffusively on the upper paths (for there are no delays there) and propagated in unit time steps along the lower paths (due to the delays). At the right-hand end of the filter the energy is reflected as noted from the boundary condition (8).

If the input is removed, then (20) indicates that the energy must be a nonincreasing function of time. This fact defines the basis of the stability derivations. If the k-parameter for any section equals plus or minus 1 at any time, then that section effectively isolates two portions of the filter. It reflects all of the energy arriving from the lower right of the section, back into the right portion of the filter and all of the energy arriving from the upper left of the section back into the left portion of the filter. While the k-parameter remains equal to one in magnitude, no energy is transferred through that section of the filter. The portion to the right is an oscillator and that to the left is a filter of smaller order than the original.

If the only bound on the time-varying k-parameters

is that they are all of less than unity magnitude, then it is possible to create an energy trap and thus an unstable filter, provided that the input is not allowed to equal zero exactly at any point in time. This is accomplished by simply having the time-varying k-parameter $k_{M-1}(n)$ satisfy

$$k_{M-1}(n)\,x(n) + [1 - k_{M-1}^2(n)]^{1/2} x_{M-1}^-(n) = 0$$

or

$$k_{M-1}(n) = -\sin\{\tan^{-1}[x_{M-1}^-(n)/x(n)]\}.$$

As long as $x(n)$ is not equal to zero, this equation can be satisfied—$|k_{M-1}(n)|$ will remain less than one, yet the filter will be unstable for the energy will monotonically increase without limit if the input remains nonzero.

In order to obtain theoretical stability, it is necessary that the time-varying k-parameters be bounded in magnitude by any constant K less than one, i.e.,

$$|k_m(n)| \leqslant K < 1. \tag{21}$$

Finite Energy Stability

If the input $x(n)$ has finite energy, E_{in}, then

$$\sum_{n=0}^{N-1} [x(n)]^2 \leqslant E_{in} \tag{22}$$

for all $N > 0$. By summing (20) from $n = 0$ through $n = N - 1$ the result

$$E(n) + \sum_{n=1}^{N} [x_M^-(n)]^2 = E(0) + \sum_{n=0}^{N-1} [x(n)]^2 \leqslant E(0) + E_{in} \tag{23}$$

is obtained. From the upper bound indicated in (23), the summations of the squares of $x(n)$ and $x_M^-(n+1)$ must converge as N goes to infinity, and hence the terms in the summation must approach zero as n approaches infinity.

As a result, $x(n) = x_M^+(n)$ and $x_M^-(n+1)$ must have a common bound which itself approaches zero as n goes to infinity. This bound can be utilized to bound other nodes in the filter, provided that the inequality of (21) is satisfied. Having placed a bound on both $x_m^+(\cdot)$ and $x_m^-(\cdot)$ which approaches zero as the argument goes to infinity, (14) can be rewritten with $k_m = \sin\phi_m$ as

$$x_m^+(n) = \sec\phi_m x_{m+1}^+(n) - \tan\phi_m x_{m+1}^-(n), \tag{24a}$$

and

$$x_m^-(n) = -\tan\phi_m x_{m+1}^+(n) + \sec\phi_m x_{m+1}^-(n). \tag{24b}$$

By combining (24) with the result

$$|\sec\phi_m| + |\tan\phi_m| = [(1 + |k_m|)/(1 - |k_m|)]^{1/2} \leqslant [(1+K)/(1-K)]^{1/2}, \tag{25}$$

$x_m^+(n)$ and $x_m^-(n)$ can be bounded by

$$|x_m^+(n)| \leqslant \left[\frac{(1+K)}{(1-K)}\right]^{1/2} \max\,[x_{m+1}^+(n), x_{m+1}^-(n)] \tag{26a}$$

and

$$|x_m^-(n)| \leqslant \left[\frac{(1+K)}{(1-K)}\right]^{1/2} \max\,[x_{m+1}^+(n), x_{m+1}^-(n)]. \tag{26b}$$

By starting with the bound on $x_m^+(\cdot)$ and $x_m^-(\cdot)$, (26) can be successively applied for $m = M - 1, M - 2, \cdots, 0$, to bound each of the nodes of the filter by a sequence which approaches zero. Thus a finite energy input implies that the filter energy, and each node value within the filter approaches zero.

Bounded Mean-Square Stability

If the input has a bounded mean square, then by definition

$$\frac{1}{N}\sum_{n=0}^{N-1} [x(n)]^2 \leqslant E_b. \tag{27}$$

As in the derivation of (23), one can sum (20) from $n = 0$ to $N - 1$ to obtain

$$\frac{1}{N}E(N) + \frac{1}{N}\sum_{n=1}^{N} [x_M^-(n)]^2 = \frac{1}{N}\sum_{n=0}^{N-1} [x(n)]^2 + \frac{1}{N}E(0) \leqslant E_b + E(0)/N. \tag{28}$$

From (28) one can note that the mean square of $x_M^-(n)$ must also be bounded. In fact it is bounded above by $E_b + E(0)/N$, which will approach E_b as N goes to infinity.

Using these bounds on the input $x_M^+(n)$ and $x_M^-(n)$, one can apply (26) for $m = M - 1, M - 2, \cdots 0$, to show that $x_m^+(\cdot)$ and $x_m^-(\cdot)$ must have bounded mean squares as well.

External Stability

The strongest stability definition is that of external stability, where bounded input must imply bounded output. While it is conjectured that this is true for the normalized form, we have been unable to prove it in a rigorous manner. If the input is bounded it will have a bounded mean square, and this implies that each node will have a bounded mean-square response, but this fact is insufficient to guarantee that the total response is bounded.

Other Filter Forms

While the concept of energy in filter forms other than the normalized form may be useful for some applications, it does not appear to be applicable for demonstrating stability. The difficulty in extension

occurs because the energy of each section must take on the form

$$\Delta E_m = [x_m^-(n)]^2/\pi_m^2 \alpha_m$$

in order to arrive at any simple conservation of energy relations for the separate sections. If the filter has parameters which vary with time, then the section energy depends not only upon information stored in the delay sections, but upon the filter parameters as well. Thus, energy can be gained or lost by simply changing the parameters of the filter, even though no energy is actually transferred from adjacent sections.

The important result of (20) remains valid only when the filter parameters are constant or when the filter is structured in the normalized form. Thus, stability proofs using the concept of energy of the filter are only valid for the normalized form.

IV. Filter Comparisons and Roundoff Errors

The orthogonal polynomial filter forms are canonic with respect to the number of delays, but only the one-multiplier forms are canonic with respect to the number of multiplies. Their advantages lie not with efficiency of calculation but in scaling properties and roundoff noise considerations, when these forms are compared with the more standard forms.

First, in terms of scaling considerations we shall utilize the L_2 norm as discussed by Jackson [8]. Let $\{h(n)\}$ represent the unit sample response time sequence at some filter node and $H(z) = Q(z)/A_M(z)$ represent its z transform. If the L_2 norm is denoted by $\| F_h \|$, then

$$\| F_h \|^2 = \sum_{n=0}^{\infty} [h(n)]^2 = \langle Q(z), Q(z) \rangle. \tag{29}$$

If $\{w(n)\}$ represents the response at that node due to a more general input sequence $\{x(n)\}$ then these are related by the convolution equation

$$w(n) = \sum_{k=0}^{\infty} h(k) x(n-k). \tag{30}$$

Equation (30) can be utilized to place numerous different types of bounds at the node in question, each dependent upon the input and the L_2 norm (the square root of the energy due to a unit sample input).

From (30) one can note that when the input has a finite energy, then the output energy is bounded by

$$\sum_{n=0}^{\infty} [w(n)]^2 \leqslant \| F_h \|^2 \sum_{n=0}^{\infty} [x(n)]^2. \tag{31}$$

As the energy places an elementary bound on the magnitude square of the sequence, a bound on the magnitude of the response at a node can be defined as simply the product of the L_2 norm, $\| F_h \|$, at that node and the square root of the input energy.

When the input is a stationary random sequence, whose autocorrelation sequence $r_x(n)$ has a spectrum $R(e^{j\theta})$, where

$$R(z) = \sum_{n=-\infty}^{\infty} r_x(n) z^{-n},$$

then the mean square of $w(n)$ is given by

$$\sigma_w^2 = \sum_{k=0}^{\infty} \sum_{l=0}^{\infty} h(k) h(l) r_x(k-l)$$

$$= \int_{-\pi}^{\pi} H(e^{j\theta}) H(e^{-j\theta}) R(e^{j\theta}) \frac{d\theta}{2\pi},$$

leading to the simple bound

$$\sigma_w^2 \leqslant R_{\max} \int_{-\pi}^{\pi} H(e^{j\theta}) H(e^{-j\theta}) \frac{d\theta}{2\pi} = R_{\max} \| F_h \|^2. \tag{32}$$

The maximum value of $R(e^{j\theta})$, given by $R_{\max}$, is equal to $R(e^{j\theta})$ in the case where the input is uncorrelated and is simply the mean-square input, a constant independent of θ.

When the input is bounded in magnitude by $x_{\max}$, (30) can be used to generate the simple bound

$$|w(n)| \leqslant x_{\max} \sum_{n=0}^{\infty} |h(n)|. \tag{33}$$

It is shown in Appendix A that the summation of (33) can be bounded by utilizing the L_2 norm in the form

$$\sum_{n=0}^{\infty} |h(n)| \leqslant \sqrt{N_{\text{eff}}} \| F_h \|, \tag{34}$$

where N_{eff} is a property of the filter denominator, representing an effective filter length. In particular, for a finite impulse response (FIR) filter which has exactly N nonzero coefficients for its unit sample response, the Cauchy-Schwarz inequality yields (34) with $N_{\text{eff}} = N$. For more general filters as discussed here, N_{eff} is found from

$$\sqrt{N_{\text{eff}}} = 1 + \sum_{n=1}^{\infty} \sqrt{E_h(n)}, \tag{35}$$

where $E_h(n)$ is the filter energy resulting from a unit sample input, as discussed in the preceding section.

In each of these cases, the L_2 norm can be combined with input signal properties and possibly the value of N_{eff} to obtain results for filter scaling to avoid overflow. In the case of random inputs, one can only work with probability of overflow; but in the other cases absolute bounds can be utilized.

Finite Word Length Scaling

In using finite word length arithmetic, it is desirable that all numerical values be as large as possible, without causing overflow at critical nodes, so that the number of significant figures in the respective calculations can also be as large as possible. Using the L_2

norms as a measure of relative numerical values, it has been shown [1] that the two-multiplier implementation has norms that monotonically increase from the filter input where $m = M$ to the reflection point where $m = 0$. This is shown in an example for a twelfth-order elliptic bandpass filter in Fig. 3. Scaling has been effected so that the peak L_2 norm is precisely unity.

Fig. 3 also shows the norms for one-multiplier implementations, using all plus signs, all minus signs, and optimal sign parameters. These curves do not reach the peak value of unity, since node values for the multiplier inputs (as indicated in [1, Fig. 4]) are not shown in these figures, and it is these that determine the final scaling. The optimal sign parameter choices appear to utilize the best properties of the plus and minus sign parameter choice in minimizing dynamic range. An algorithm for choosing these optimal sign parameters has been previously discussed [1].

The alternation in direction of the norms in the one-multiplier form for identical signs is due to an alternation in signs of the k-parameters, a property which appears to be common for filters whose poles lie in the right-half z plane. The one-multiplier form with optimal sign parameters has an obviously smaller dynamic range than the one-multiplier forms with fixed sign parameters and the two-multiplier form.

The top axis of the figure shows the node values for the normalized form, where all of the L_2 norms are unity. In this case each of the calculations is carried out (on the average) with the same number of significant figures, and the dynamic range of these node values is zero.

One qualitative way to note the advantage of the normalized form lies in the scaling of the input before entry into the filter. If one chooses as a criterion of scaling, for example, the fact that all L_2 norm values must be bounded by unity, then the two-multiplier input must be scaled down by a factor of $1/\sqrt{\alpha_0}$ whereas the normalized form does not require scaling. A direct form filter implementation must have an input scaled down by at least $1/\sqrt{\alpha_0}$, while a parallel form must have the input scaled down by at least $1/\sqrt{\alpha'}$, where α' represents the largest L_2 norm of the separate parallel sections. This scaling down automatically reduces the dynamic range of the input, and increases the output noise due to roundoff or truncation errors within the filter since the final output must be multiplied by the reciprocal factor to have a correct overall gain.

A Two-Pole Example

As an example of scaling and roundoff error properties, consider the simple two-pole filter defined by

$$G(z) = \frac{1}{1 - 2r\cos(\theta)z^{-1} + r^2 z^{-2}}. \tag{36}$$

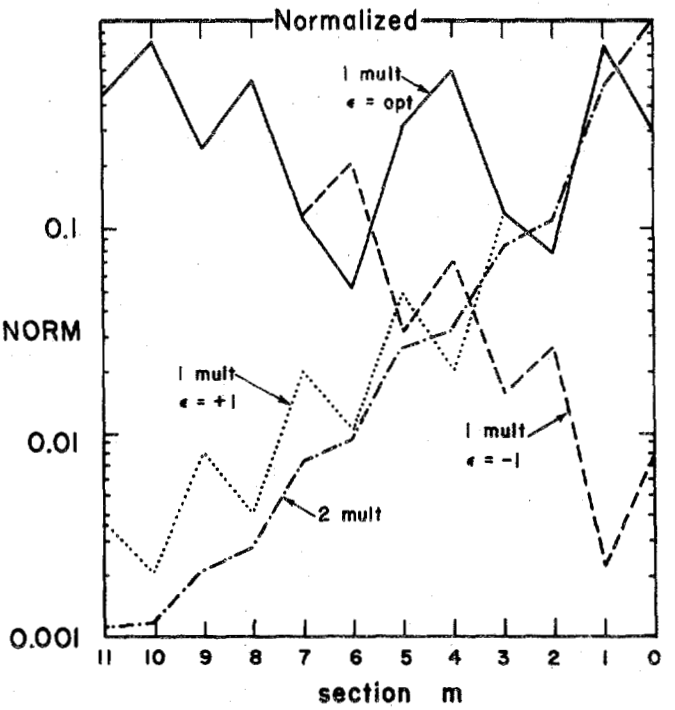

Fig. 3. Norms of various structures for the m sections $G_m(z)$ $m = 0, 1, \cdots, M-1$ normalized by the maximum norm within each filter.

Following the recursion relations [1], we find

$$k_1 = r^2, \qquad k_0 = -2r\cos(\theta)/(1 + r^2) \tag{37a}$$

$$\alpha_1 = 1/(1 - k_1^2), \qquad \alpha_0 = 1/[(1 - k_1^2)(1 - k_0^2)] \tag{37b}$$

$$\nu_2 = 0, \nu_1 = 0, \qquad \nu_0 = 1. \tag{37c}$$

Three implementations of this filter are shown in Fig. 4. In each case, the portion within the solid lines is scaled so that the maximum L_2 norm (at internal nodes in $1/A(z)$) is unity. Fig. 4(a) gives the direct form, which for a simple two-pole filter is also equivalent to the cascade and parallel forms. The multiplication by $2r\cos(\theta)$ is separated into two parts to allow for the case where $2r\cos(\theta)$ is greater than 1. First a multiplication by $r\cos(\theta)$ is introduced, followed by a multiplication by two (a binary shift). The order of computation is chosen so as to keep the L_2 norms at the multiplier inputs less than or equal to 1. Fig. 4(b) and (c) show the structure of the two-multiplier form and normalized form, respectively. In both Fig. 4(b) and (c) only those parts of the filter that effect the output are shown. The two-multiplier form of Fig. 4(b) has been drawn in a slightly different manner than the lattice of [1, Fig. 2] for ease of analysis, yet is identical to the lattice form. The one-multiplier form is not considered here, for its advantages lie in filters with multiple sections.

If multiplications are performed in finite word arithmetic subject to rounding, then the multipliers

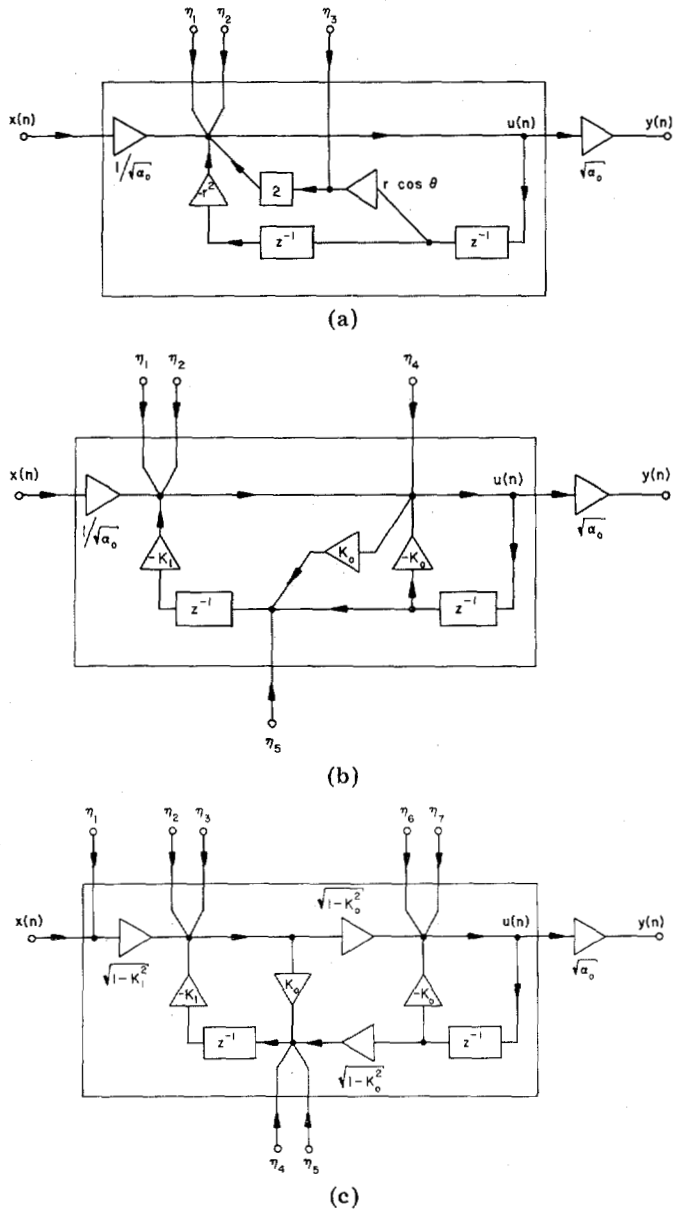

Fig. 4. Block diagrams of second-order filters with roundoff noise terms (a) direct form, (b) two-multiplier lattice form, and (c) normalized form.

[not including the binary shift of Fig. 4(a)] introduce uncorrelated random noise into the filter, as indicated by the η_k in the separate parts of the filter. An additional noise source, η_1, is included at the input of the normalized form to allow for possible noise introduced by additional scaling which might be needed for arbitrary inputs.

The separate transfer functions from the noise sources to the node $u(n)$ and L_2 norms of the roundoff noise at $u(n)$ are given in Appendix B. For the direct or transversal form, the result is $6\alpha_0$. Should $2r\cos(\theta)$ be less than unity, the binary shift can be included with the multiplication, and this result decreases to $3\alpha_0$, but we shall restrict our discussion here to the former case.

The L_2 norm square for the noise in the two-multiplier form, Fig. 4(b), is $(3 + k_1^2)\alpha_0$. For the case of a narrow bandwidth, where r is near unity, this norm square is approximately $4\alpha_2$, or 2/3 of the result for the direct form. This improvement lies basically in the fact that no binary shifts are needed in the two-multiplier form.

The L_2 norm square for the noise in the normalized form [Fig. 4(c)] is given by

$$[5 + k_1^2 - 3k_0^2 + k_1^2 k_0^2 - 4k_0^2 k_1]\alpha_0 = [6(1 - k_0^2) + 2(1 - k_1)^2 - (3 - k_1)(1 - k_0^2)(1 - k_1)]\alpha_0 .$$

If the filter has a narrow bandwidth so that

$$r = 1 - \Delta r \quad \text{with } \Delta r << 1 \tag{38}$$

and terms of order $(\Delta r)^2$ and higher are ignored, then

$$(1 - k_0^2)(6 - 4\,\Delta r)\alpha_0 \approx 6\sin^2(\theta)\,[1 - 2\,\Delta r/3]\alpha_0 .$$

Thus, the ratio of the noise norm for the normalized structure to the standard form is less than $|\sin(\theta)|$, where θ is the angular location of the poles. As the complex conjugate poles move closer together (cluster) the angle θ approaches zero or π, the degree of improvement over the standard second-order form becomes unbounded.

V. Summary

A normalized filter structure for optimizing finite word length scaling, in terms of L_2 norms, has been presented. The design is entirely recursive and requires only that the starting point, a direct form filter, be stable. The normalized form represents an extension of the class of filters designed by using orthogonal polynomial expansions [1].

It was shown that a concept of filter energy could be utilized with the normalized structure to demonstrate two forms of stability in the presence of time-varying parameters. This theoretical result can be useful in some time-varying filter applications, such as linear prediction speech synthesis.

In the time-invariant case, it was shown that the filter energy could be utilized in any of the filter structures to obtain an effective filter length, N_{eff}, which can be used with the knowledge of L_2 norms to bound outputs resulting from bounded filter input, eliminating the need of summing magnitudes of unit sample responses.

A two-pole example was utilized to illustrate roundoff noise comparisons between the standard form, two-multiplier form, and normalized form of the filter structure. In that simple example it was shown that the two-multiplier form is better (in the sense of L_2 noise norms) than the standard form, and that the normalized form was better than both.

When bandwidths are narrow, the degree of improve-

ment of the normalized form over the other forms greatly increases as the poles begin to cluster around zero frequency or the half-sampling frequency. While improvement was illustrated with only a simple two-pole example, similar results are observed for much more complicated filters [9].

Each of the three forms based upon orthogonal polynomial theory are canonic with respect to the number of delays. Only the one-multiplier forms are canonic with respect to the number of multiplications. If time-invariant filters are being utilized, the best choice with regard to roundoff noise considerations appears to be the normalized form, which requires four multiplications and two additions per section. However, in an array processor, utilizing complex arithmetic, each section can be implemented with only one complex operation. If the number of multiplications is an important consideration, then the next choice among these forms is the one-multiplier form with optimal sign parameters.

If filters are to be used in a time-varying environment, then the best choice again appears to be the normalized form, simply because of its theoretical stability relations. In speech synthesis filters the number of calculations involved may be of great importance, in which case the one-multiplier forms seem more advantageous. If optimal sign parameters are used, time must be spent in calculating them and changing them along with the other filter parameters. We have observed that one-multiplier forms with all sign parameters equal to minus 1 are an improvement over the two-multiplier form, while not being as good as the use of optimal sign parameters. This result depends upon the nature of the signals being generated and appears to hold for the synthesis of voiced sounds, which requires more accuracy than the synthesis of unvoiced sounds.

A detailed theoretical and experimental analysis of roundoff noise considerations is presented elsewhere [9]. In particular the orthogonal polynomial expansion forms, two-multiplier, one-multiplier, and normalized are compared with the worst and best of the more standard filters, the worst represented by the direct or transversal form and the best by the parallel (Jackson's 1-P form [8]). Particular interest is being paid to those filters which have at least one pole pair near the unit circle. Preliminary results have shown that the two-multiplier form is consistently better than the direct or transversal form. When there are a number of sections, the one-multiplier form with optimal sign parameters is better than the two-multiplier form.

As might be expected, the parallel form is an improvement over the one-multiplier forms. The normalized form appears to be consistently better than the parallel form. In cases where the filter poles cluster, the normalized form is far better than the other forms, as might be noted from the elementary example considered in this paper. We conjecture that in the case of narrow bandwidths and clustered poles, the normalized form represents the best possible digital filter structure for finite word length implementation, in the sense that minimum roundoff noise is obtained.

Appendix A

The transfer function from the filter input to any node in the filter can be expressed as a ratio of polynomials, as in the case of the overall transfer function (1), having the same denominator, $A_M(z)$. As in the derivation of [1, eq. (10)] this can be expanded in a series of orthonormal polynomials in the form

$$H(z) = \sum_{m=0}^{M} \frac{\hat{\mu}_m z \hat{B}_m(z)}{A_M(z)}. \tag{A-1}$$

The unit sample responses can then be related as in (9) for the overall system as

$$h(n) = \sum_{m=0}^{M} \hat{\mu}_m x_m^- (n+1). \tag{A-2}$$

Though the expansions of (A-1) and (A-2) are in terms of the normalized variables, it should be kept in mind that here these are simply mathematical relations and imply nothing about the actual structure of the filter.

From the orthonormal relations of the terms in the expansion, an elementary result for the L_2 norm square of $h(n)$ is obtained as

$$\|F_h\|^2 = \sum_{n=0}^{\infty} [h(n)]^2 = \sum_{m=0}^{M} \hat{\mu}_m^2 . \tag{A-3}$$

Applying the Cauchy–Schwarz inequality to (A-2) leads to the result

$$[h(n)]^2 \leq \sum_{i=0}^{M} \hat{\mu}_i^2 \sum_{m=0}^{M} [x_m^-(n+1)]^2 . \tag{A-4}$$

This can be simplified by using the filter energy concept. Let $E_h(n)$ denote the filter energy due to a unit sample input. From the definition of (19) and the property of (20) one can write

$$\begin{aligned} \sum_{m=0}^{M} [x_m^-(n+1)]^2 &= E_h(n+1) + [x_M^-(n+1)]^2 \\ &= E_h(n) + [x(n)]^2 \\ &= E_h(n) + \delta_{n,0} \end{aligned} \tag{A-5}$$

where $\delta_{n,0}$ is the unit sample at the origin (zero for $n \neq 0$ and one for $n = 0$).

Combining (A-3)–(A-5)

$$|h(n)| \leq \|F_h\| [E_h(n) + \delta_{n,0}]^{1/2} . \tag{A-6}$$

Summing this equation from $n = 0$ to $n = \infty$ and

utilizing the fact that $E_h(0) = 0$,

$$\sum_{n=0}^{\infty} |h(n)| \leq \|F_h\| \left[1 + \sum_{n=1}^{\infty} \sqrt{E_h(n)}\right]. \quad \text{(A-7)}$$

Equations (34) and (35) are simply a restatement of (A-7). It might be noted that the summation of the square roots of the filter energy due to a unit sample response cannot be recursively evaluated as the filter parameters were. However, the summation can be approximately evaluated once for the filter, and then utilized in scaling all filter implementations for the case of arbitrary bounded inputs.

Appendix B

The filter of (36) and (37) is implemented as shown in the separate parts of Fig. 4. In order to find the L_2 norm at the node $u(n)$ in each part, due to each noise term, one first obtains the transfer function from the node in question to $u(n)$. This can be done either by ordinary algebra and solution of simultaneous equations or by an application of linear signal flow graph theory. In each case the result will be a ratio of polynomials of the form $Q(z)/A_2(z)$ where

$$A_2(z) = 1 + k_0(1 + k_1)z^{-1} + k_1 z^{-2}$$
$$= 1 - 2r\cos(\theta)z^{-1} + r^2 z^{-2}. \quad \text{(B-1)}$$

L_2 norms are then most simply found by expanding $Q(z)$ as a series of the orthogonal polynomials $zB_m(z)$, where

$$zB_0(z) = 1 \quad \text{(B-2a)}$$

$$zB_1(z) = k_0 + z^{-1} \quad \text{(B-2b)}$$

$$zB_2(z) = k_1 + k_0(1 + k_1)z^{-1} + z^{-2} \quad \text{(B-2c)}$$

so that the orthogonality relations of the $zB_m(z)$ polynomials can be utilized.

Direct Form

The transfer function from either η_1 or η_2 to $u(n)$ is found to be simply $1/A_2(z)$. The numerator polynomial is then

$$Q(z) = 1 = zB_0(z),$$

so that the L_2 norm square from either η_1 or η_2 is given by

$$\langle Q(z), Q(z)\rangle = \alpha_0.$$

The transfer function from η_3 to $u(n)$ is twice that of the preceding transfer functions, and thus its L_2 norm square is $4\alpha_0$. The total norm square of the noise is thus,

$$\alpha_0 + \alpha_0 + 4\alpha_0 = 6\alpha_0.$$

Two-Multiplier Form

The transfer functions from η_1, η_2, and η_4 to $u(n)$ are each given simply by $1/A_2(z)$. As in the first portion of the direct form calculation, a resultant L_2 norm square of α_0 is obtained for each term.

The transfer function from η_3 to $u(n)$ has a numerator polynomial $Q(z) = k_1 z^{-1}$. Using the polynomial expansion

$$Q(z) = k_1[zB_1(z) - k_0 zB_0(z)]$$

one finds

$$\langle Q(z), Q(z)\rangle = k_1^2\,\alpha_1 + k_0^2 k_1^2\,\alpha_0 = k_1^2(1 - k_0^2)\,\alpha_0 + k_0^2 k_1^2\,\alpha_0 = k_1^2\,\alpha_0.$$

Summing the separate L_2 norm squares, one obtains

$$3\alpha_0 + k_1^2\,\alpha_0.$$

Normalized Form

The numerator of the transfer function from η_1 to $u(n)$ is simply $(1 - k_1^2)^{1/2}(1 - k_0^2)^{1/2}$ giving as its L_2 norm square the result $(1 - k_1^2)(1 - k_0^2)\alpha_0$. From η_2 and from η_3, the numerator polynomial is $(1 - k_0^2)^{1/2}$, yielding the L_2 norm squares $(1 - k_0^2)\cdot\alpha_0$.

From η_4 and from η_5 the numerator polynomial is $-k_1(1 - k_0^2)^{1/2}z^{-1}$. This leads to the L_2 norm square $k_1^2(1 - k_0^2)\alpha_0$ for η_4 and η_5.

From η_6 and η_7 to $u(n)$, one finds the numerator polynomials given by

$$Q(z) = 1 + k_0 k_1 z^{-1} = (1 - k_0^2 k_1)zB_0(z) + k_0 k_1 zB_1(z)$$

resulting in the L_2 norm squares

$$\langle Q(z), Q(z)\rangle = (1 - k_0^2 k_1)^2\alpha_0 + k_0^2 k_1^2\alpha_1 = (1 - 2k_0^2 k_1 + k_0^2 k_1^2)\alpha_0.$$

Adding the L_2 norm squares from η_1 to η_7 results in

$$(1 - k_1^2)(1 - k_0^2)\alpha_0 + 2(1 - k_0^2)\alpha_0 + 2k_1^2(1 - k_0^2)\alpha_0 + 2(1 - 2k_0^2 k_1 + k_0^2 k_1^2)\alpha_0$$
$$= [5 + k_1^2 - 3k_0^2 + k_1^2 k_0^2 - 4k_0^2 k_1]\alpha_0$$
$$= [6(1 - k_0^2) + 2(1 - k_1)^2 - (3 - k_1)(1 - k_0^2)\cdot(1 - k_1)]\alpha_0.$$

References

[1] A. H. Gray, Jr., and J. D. Markel, "Digital lattice and ladder filter synthesis," *IEEE Trans. Audio Electroacoust.*, vol. AU-21, pp. 491–500, Dec. 1973.

[2] G. Szegö, "Ein Grenszwertsatz uber die Toeplitzschen Determinanten einer reellen positiven Funktion," *Math. Ann.*, vol. 76, pp. 490–503, 1915.

[3] T. Kailath, "A view of three decades of linear filtering theory," *IEEE Trans. Inform. Theory*, vol. IT-20, pp. 146–181, Mar. 1974.

[4] N. Levinson, "The Wiener rms (root mean square) error criterion in filter design and prediction," *J. Math. Phys.*, vol. XXV, no. 4, pp. 261–278, Jan. 1974; also N. Wiener, *Extrapolation, Interpolation, and Smoothing of Stationary Time Series.* Cambridge, Mass.: Mass. Inst. Tech. Press, 1949, Appendix B.

[5] E. A. Robinson, *Multichannel Time-Series Analysis With Digital Computer Programs.* San Francisco, Calif.: Holden-Day, 1967.

[6] E. A. Robinson, *Statistical Communication and Detection*

with Special Reference to Digital Data Processing of Radar and Seismic Signals. New York: Hafner, 1967.
[7] F. Itakura and S. Saito, "Digital filtering techniques for speech analysis and synthesis," in *7th Int. Acoust. Cong.*, Budapest, Paper 25C-1, 1971.
[8] L. B. Jackson, "An analysis of roundoff noise in digital filters," Ph.D. dissertation, Stevens Inst. Tech., Hoboken, N.J., 1969.
[9] J. D. Markel and A. H. Gray, Jr., "Roundoff noise characteristics of a class of orthogonal polynomial structures," *IEEE Trans. Acoust., Speech, Signal Processing*, to be published.

Quantization Errors in the Fast Fourier Transform

DAVID V. JAMES

Abstract—**When a fast Fourier transform (FFT) is implemented on a digital machine, quantization errors will arise due to finite word lengths in the digital system. The magnitudes and characteristics of these errors must be known if an FFT is to be designed with the minimum word lengths needed for acceptable performance.**

Two forms of FFT quantization, coefficient rounding and floating point arithmetic quantization, are analyzed in this paper. A theory is presented from which several new results can be obtained. The error characteristics of FFT's using exact and truncated values for the coefficients 1 and $-j$ are found to be roughly equivalent. The accuracy of the theory is tested by computer simulations. Using the models introduced in this paper, new and accurate models can be derived to model quantization errors in high-speed convolution filters.

I. Introduction

A. Problem Statement

In general, a fast Fourier transform (FFT) cannot be implemented exactly. Consider the FFT butterfly of Fig. 1 and (1).

$$\begin{aligned} X(k) &= x(k) + W^m x(k) \\ X(k+n) &= x(k) - W^m x(k+p) \end{aligned} \tag{1}$$

where the constant W^m is called an FFT coefficient

$$W^m = \exp(-j2\pi m/N)$$
$$j = \sqrt{-1}.$$

Each multiplication and addition shown in (1) may introduce an error caused by the rounding or truncation of the arithmetic results; these errors will be called FFT arithmetic quantization errors. Also, the FFT butterfly operations will not (in general) be performed with exact FFT coefficients W^m, but with rounded coefficients $W'(m)$; the errors in the FFT which are caused by such inexact coefficients will be referred to as FFT coefficient quantization errors.

Manuscript received February 2, 1974; revised February 2, 1975. This work was sponsored by the Advanced Research Projects Agency of the Department of Defense.
The author is with the Massachusetts Institute of Technology Lincoln Laboratory, Lexington, Mass. 02173.

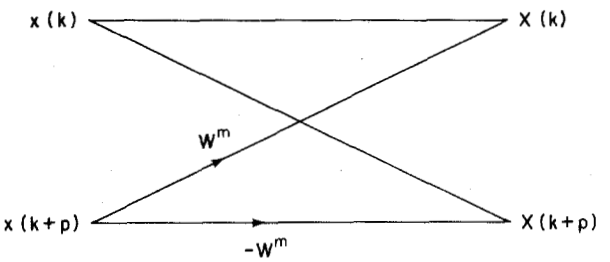

Fig. 1. FFT butterfly.

Several researchers have successfully analyzed fixed point arithmetic quantization in the FFT [6], [10], [12]. FFT floating point arithmetic quantization errors have also been modeled by Weinstein [10], [11] and Kaneko and Liu [5]. These papers were concerned primarily with the error characteristics of FFT's performed on white noise or sine wave inputs.

FFT coefficient rounding errors have also been analyzed by several researchers [6], [9], [10]. Weinstein's [6], [10] analysis evaluated FFT coefficient quantization errors in FFT's performed on white noise sequences. Tufts *et al.* [9] also analyzed FFT coefficient rounding errors, and his analysis is applicable to FFT's performed on sine wave inputs.

B. Paper Summary

All forms of quantization error in the FFT have been analyzed. However, both FFT coefficient rounding errors and floating point arithmetic quantization errors need further analysis and accurate models to predict their behavior; these models should accurately predict both the magnitude and the structure of errors arising from 1) quantized FFT's which are performed on signals $x(n)$ of known spectra and 2) quantized FFT's used in high-speed convolution filtering applications. Such models are presented in this paper.

The models presented in this paper are obtained from a previous work of the author [4]. For the sake of brevity, the simpler results are presented, and rigorous derivations are deleted.