THE BELL SYSTEM TECHNICAL JOURNAL
Vol. 57, No. 7, September 1978
Printed in U.S.A.

An Adaptive PCM System Designed for Noisy Channels and Digital Implementations*

By DEBASIS MITRA and B. GOTZ

(Manuscript received November 29, 1977)

We propose a new adaptive quantization scheme for digitally implementing PCM *and* DPCM *structures. The arithmetics we develop for the digital processing are useful as well in the implementation of previously existing schemes for adaptive quantization. Two objectives are stressed here:* (*i*) *The system must be robust in the presence of noise in the transmission channel which causes the synchronization between quantizer adaptations in the transmitter and receiver to deteriorate.* (*ii*) *It must also minimize the complexity of the digital realization. In addition to the above objectives, we require, of course, good fidelity of the processed speech waveform. The problem of synchronization in digital implementations where the constraint of finite precision arithmetic exists has not been addressed previously. We begin by examining an existing, idealized adaptation algorithm which contains a leakage parameter for the purpose of deriving robustness. We prove that, to provide the necessary synchronization capability without impairing the quality of speech reproduction, it is necessary to use a minimum, unexpectedly large, number of bits in the machine words and, additionally, to carefully specify the internal arithmetic, as is done here.*

The new scheme that we propose here uses an order of magnitude less memory in an ROM*-based implementation. The key innovations responsible for the improvement are:* (*i*) *modification of the adaptation algorithm to one where leakage is interleaved infrequently but at regular intervals into the adaptation recursion;* (*ii*) *a specification of the internal machine arithmetic that guarantees synchronization in the presence of channel errors. A detailed theoretical analysis of the statistical behavior of the proposed system for random inputs is given here. Results of a simulation of a realistic 16-level adaptive quantizer are reported.*

* A short version of this paper was presented at the International Conference on Communications, Toronto, June 1978.

I. INTRODUCTION

We propose a new scheme for adaptive quantization which is particularly well suited to the digital implementation of PCM and DPCM structures. In the course of this work, we have developed arithmetics for the digital processing that are useful as well in the implementation of previously existing schemes for robust quantization.

The exacting requirements on adaptive quantization stemming from the broad dynamic range and rapid transient behavior of speech are well known. Two additional objectives are given equal importance here: (*i*) To make the system robust in the presence of channel errors. Thus, while channel errors may cause the quantizer adaptations in transmitter and receiver to be put out of synchronization,* a mechanism must exist which acts to rapidly restore the synchronization during periods of error-free transmission. (*ii*) To minimize the complexity of the digital realization; specifically, to minimize the length of the internal words in the digital processors and to facilitate the multiplexing of the hardware.

Systems do exist in the literature for robust quantization in the presence of noisy channels; one such system is described below in some detail. However, the problem of synchronizing the quantizer adaptations in the transmitter and receiver in digital implementations, where the constraint of finite precision arithmetic exists, has not been addressed previously. We prove that, to provide the necessary synchronization capability without impairing the quality of speech reproduction, it is necessary to use an unexpectedly large number of bits in the internal words of the digital processors at both sites and, additionally, to carefully specify the internal arithmetic (which we do). If the digital processing is implemented using ROMs, as is being proposed, the long internal word length is reflected in large memory requirements and therefore costly implementations as well as exposure to new errors in the processing.

The scheme that we propose here uses an order-of-magnitude less memory in an ROM-based implementation in both the transmitter and receiver. This is for comparable performance with respect to loading characteristic, signal-to-noise ratio, and the synchronization capability. Another advantage not reflected in the above estimate is the fact that the essential costly digital component, the ROM, as distinct from other less costly components such as adders, is used only for a small fraction of the total operating time. Thus, further economies may be effected through multiplexing the ROM. The key innovations are: (*i*) the modification of the adaptation algorithm which allows the internal word length of the digital processors to be reduced significantly; and (*ii*) a specification of the internal arithmetic that guarantees synchronization in the presence of channel errors. As mentioned previously, the arithmetic is also applicable in digital implementations of previously existing adaptation algorithms.

* In our usage, synchronization is synonymous with tracking.

A byproduct of the work reported here is that it establishes a link between two hitherto unconnected areas, namely, finite-arithmetic digital signal processing and waveform quantization in the presence of a noisy channel. The problem of synchronizing two geographically separated digital processors gives rise to quite novel requirements on the processing, and we expect that the problem will be a subject of further investigation in the future.

The paper is organized as follows. In Section 1.1 we describe an existing quantizer adaptation scheme and the associated synchronization problem. Section II is devoted to the basic description of the new scheme. Section 2.1 introduces the key idea underlying the scheme. Section 2.2 considers the digital implementation of the system, and Section 2.3 considers the synchronization behavior of the resulting system. Section III is devoted to the probabilistic analysis of the behavior of the proposed algorithm. The basic notions of the bias functions, central log step sizes, and load curves are introduced, and the qualitative results proved in their connection are stated. In Section IV, some computational results are presented in the context of a realistic 16-level quantizer that has been proposed and investigated previously in connection with an industrial application. We try to illuminate the topics considered in Sections II and III through examples involving this particular quantizer. Four appendices to the paper present the detailed technical derivations.

On account of the length of the paper, we considered it desirable to include a final section, Section V, which summarizes and puts into perspective the key results obtained in the preceding sections.

We should mention that the digital implementation of adaptive DPCM systems is under investigation within Bell Laboratories in connection with TASI-D, subband voice coding, and new channel banks. The work reported here is a research study and not a description of a developed design.

1.1 Background and description of the problem

We begin by describing a system proposed in Ref. 1 which, unlike earlier systems upon which it is based,[2–5] possesses the capacity to recover from past channel errors during periods of error-free transmission.

1.1.1 An existing idealized scheme for robust quantization

Let $\Delta(i)$ (see Fig. 1) denote the *step size* of a quantizer, with $2N$ levels, at the ith sampling instant; $\Delta(i)$ is adapted according to the rule

$$\Delta(i+1) = \Delta(i)^{\beta} M(i), \qquad i = 0,1,2,\cdots \tag{1}$$

where β, $0 < \beta < 1$, is the leakage constant and $M(i)$ is the *multiplier* at time i. $M(i)$ is selected from a prespecified collection of multipliers $\{M_1, M_2, \cdots, M_N\}$ according to the rule:

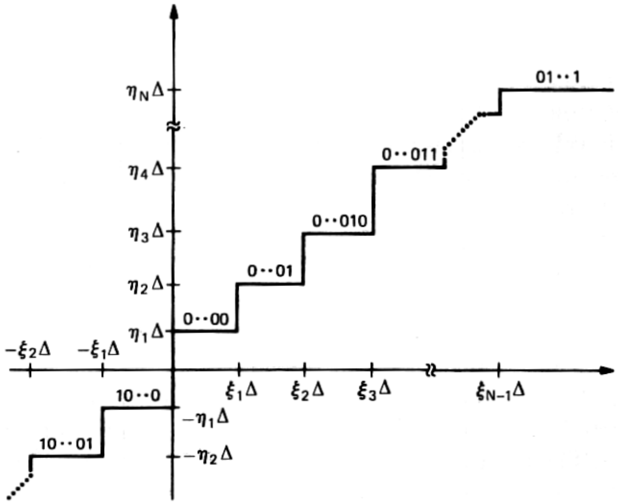

Fig. 1—The quantizer. A natural coding scheme is displayed. The step size is time-varying and the parameters (ξ_n) and (η_n) are prespecified and fixed.

$$\text{If } \xi_{r-1}\Delta(i) \le |x(i)| < \xi_r\Delta(i), \text{ then } M(i) = M_r, \tag{2}$$

where $x(i)$ is the input signal variable (speech or data) at time i and $0 = \xi_0, \xi_1, \cdots, \xi_{N-1}, \xi_N = \infty$ are fixed, ordered parameters of the quantizer,* Fig. 1. The multipliers are also ordered, i.e.,

$$M_1 \le M_2 \le \cdots \le M_N.$$

It is widely recognized[6,7] that (1) is not in a form convenient for implementation, even analog implementation. To utilize conventional multipliers, it is necessary to work with the log-transformed version of (1).

Denote the *log step size* by $d(i)$, where

$$d(i) \triangleq \log_Q \Delta(i), \tag{3}$$

Q being a fixed number greater than 1, and the *log multipliers* by

$$m(i) \triangleq \log_Q M(i),\ m_r \triangleq \log_Q M_r,\ 1 \le r \le N. \tag{4}$$

Also let

$$\bar{\xi}_r \triangleq \log_Q \xi_r, \qquad 1 \le r \le N. \tag{5}$$

Thus, from (1) and (2),

$$d(i+1) = \beta d(i) + m(i), \qquad i = 0,1,\cdots \tag{6a}$$

where

* When the parameters $\{\xi_r\}$ and $\{\eta_r\}$ are spaced equal distances apart, the quantizer is usually referred to as a uniform quantizer and it is natural to call Δ the "step size." However, for nonuniform quantizers, the term "step size" is less natural and other candidates are "scale" and "range." However, since there is no reason for confusion, we retain the familiar term "step size."

$$m(i) = m_r \text{ iff } \bar{\xi}_{r-1} + d(i) \le \log_Q |x(i)| < \bar{\xi}_r + d(i). \qquad (6b)$$

The only information that is coded and transmitted at time i is that concerning the quantizer output which uniquely determines the selected log multiplier $m(i)$. A natural coding scheme is exhibited in Fig. 1. The recursion in (6) is implemented at both the transmitter and receiver. We let $m'(i)$ denote the log multiplier corresponding to the received code word at time i, and we employ the natural notation $d'(i)$ to denote the log step size in the receiver. The reconstruction, $R(i)$, at the receiver of the input signal variable is done according to the rule:

$$\text{If } m'(i) = m_r \text{ then } |R(i)| = \eta_r Q^{d'(i)}, \qquad (7)$$

where η_r, $1 \le r \le N$, are also prespecified, fixed parameters of the quantizer, as shown in Fig. 1. The sign of the reconstructed value is obtained from the sign bit, usually the first and shown as such in Fig. 1, in the received code word.

The synchronization capability of the system, i.e., the capability possessed by the solutions of the recursions, $\{d(\cdot)\}$ and $\{d'(\cdot)\}$, at the transmitter and receiver to approach each other during error-free transmission is entirely due to the presence of the leakage parameter β. For if $d(0)$ and $d'(0)$ are two, possibly different, initial values of the log-step sizes at the commencement of an epoch of error-free transmission, then during the epoch

$$|d(i) - d'(i)| = \beta^i |d(0) - d'(0)|, \qquad i \ge 0. \qquad (8)$$

The notion of introducing leakage as a mechanism for deriving robustness in the presence of a noisy channel is a well-known one in communication practice; witness, the leaky delta-modulator.[8]

As far as the synchronization of the transmitter and receiver adaptations is concerned, eq. (8) implies that decreasing β provides improved quality. However, there is an accompanying price. The data in Fig. 5 of Ref. 1 together with the theory developed here in Sections 3.2 and 3.3 on the load curves (which describe the statistical behavior of the step size for random inputs) show that the statistical dynamic range of the step size is reduced rapidly with decreasing β, with a concomitant deterioration of the quality of the reconstruction.* Recent subjective tests[10] have shown that it is very unlikely that β less than $^{63}/_{64}$ can provide acceptable quality speech reproduction.

Herein lies the gist of the problem: For good quality reproduction, the leakage parameter must necessarily be very close to 1, and this, on the other hand, makes it difficult to provide good quality synchronization. It is thus necessary to walk a narrow path between too small leakage and too large leakage. As we see next, the constraint of finite precision

* Numerous related topics are treated analytically in Ref. 9.

arithmetic imposed by a digital implementation compounds the design problem.

1.1.2 Digital implementations

Equation (6) assumes continuous values of $d(\cdot)$ and infinite precision arithmetical operations, and hence it can only serve as an ideal in a digital implementation. An all-digital coder will have only a limited dictionary or total number (typically, ≥ 32, ≤ 128) of possible log step sizes. We will consider the log step sizes to be integers varying from 0 to $2^K - 1$; thus, typically, $5 \leq K \leq 7$. It is necessary to introduce the notion of an *internal machine word* with K integer bits and, say, F fractional bits (the need for fractional bits will become apparent shortly); the log step size is obtained from the internal machine word at time i, $y(i)$, by means of an external arithmetic, such as truncation. Although later we will consider other possibilities, for the purpose of this discussion let us assume that the external word at time i, which is the log-step size at that time, is simply the integer part of the internal word at time i, i.e.,

$$d(i) = \lfloor y(i) \rfloor_{\text{truncate}}, \qquad i = 0,1,2 \cdots. \tag{9}$$

The machine implementation of the ideal recursion in (6) is

$$y(i+1) = \langle \beta y(i) \rangle + m(i), \qquad i = 0,1,2,\cdots, \tag{10}$$

where $\langle \beta y(i) \rangle$ denotes some procedure, such as rounding, for taking $\beta y(i)$ into a $(K+F)$-bit word. It will turn out later that this operation is best viewed with greater generality as a mapping f of $(K+F)$-bit words, with F fractional bits into other such words. Thus we restate (10) as*

$$y(i+1) = f\{y(i)\} + m(i), \qquad i = 0,1,2,\cdots. \tag{10$'$}$$

It will be assumed that all the log multipliers $\{m_r\}$ have at most F fractional bits each, which ensures that if $y(i)$ is a $(K+F)$-bit word then so is $y(i+1)$.

Figure 2 shows an example of the most direct procedure for generating the discrete map $f(y)$, namely, by rounding βy to the nearest machine word. In the example, considered $F = 1$ so that the spacing between machine words is $2^{-F} = 1/2$. A feature common to such maps is that segments of unit slope are juxtaposed between other segments of zero slope which we call "breaks."

If, as before, we distinguish the quantities associated with the receiver by the superscript $'$, we see that the offset in the machine words behaves

* In (10) and (10′) we have not made allowances for overflow. This however can be done conventionally by employing saturation where:

$$\begin{aligned} y(i+1) &= 0 \text{ if } \langle \beta y(i) \rangle + m(i) < 0, \\ &= 2^K - 2^{-F} \text{ if } \langle \beta y(i) \rangle + m(i) > 2^K - 2^{-F}, \end{aligned}$$

and in every other case (10) holds. Saturation acts to attenuate the offset in the machine words at the two sites.

as follows during epochs of error-free transmission [i.e., periods in which $m(\cdot) = m'(\cdot)$]:

$$|y(i+1) - y'(i+1)| = |f\{y(i)\} - f\{y'(i)\}| \tag{11}$$

[compare with (8)].

The synchronization problem motivates us to impose the following two rather stringent requirements on the behavior of the offset.

Synchronization requirements:

(*i*) The offset is nonincreasing at all instants of error-free transmission.

(*ii*) The integer parts of the machine words at the two sites, and hence the respective log step sizes, differ in at most a finite (preferably small) number of time instants during error-free transmission.

We require the above to hold independent of the statistics of the input process. It is clear from (11) that these requirements imply restrictions on the discrete map f which are investigated below.

Let us digress to better motivate the second of the above requirements. If the integer parts of the machine words at the two sites at any instant are not identical, then the respective log step sizes differ by at least unity and, hence, the ratio of the two step sizes is at least Q [see eq. (3)]; this factor may be unacceptably large since values of Q as high as 1.5 are being

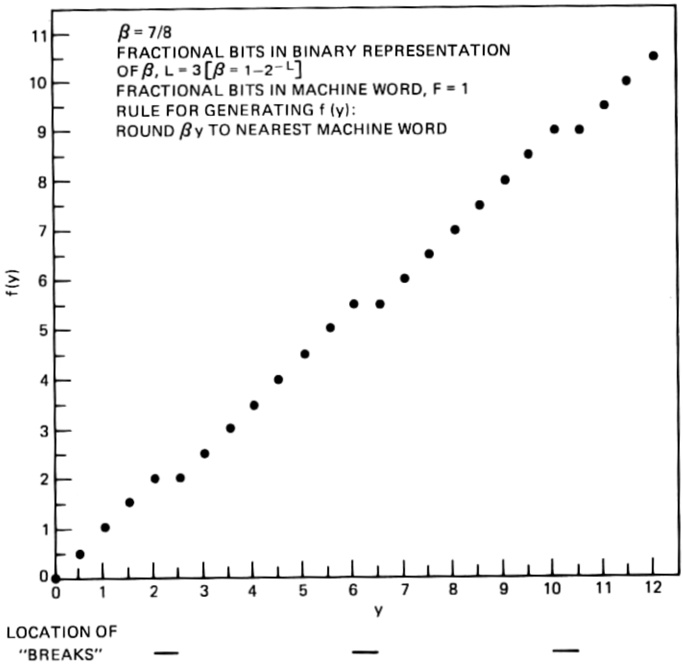

Fig. 2—An example of a naive machine arithmetic.

considered in practical designs.* To illustrate another facet of the second requirement, consider the case where, at a particular instant, the transmitter and receiver machine words are rather close, say, 1.9375 and 2.0625 ($F = 4$). Yet the integer parts are 1 and 2, respectively. Thus the step sizes are Q and Q^2, rather far apart. This example serves to illustrate that the mere proximity of the two machine words is not enough to guarantee that the log step sizes are identical.

In the following discussion, we will need to know the value of L, an integer, which is such that

$$1 - 2^{-L+1} < \beta \leq 1 - 2^{-L};$$

if $\beta = 7/8$, as in Fig. 2, then $L = 3$ and if $\beta = 63/64$ then $L = 6$. To simplify the following discussion, we shall assume that

$$\beta = 1 - 2^{-L}, \tag{12}$$

i.e. $\beta \epsilon \{1/2, 3/4, 7/8, \cdots\}$; with this form for β, L is the minimum number of fractional bits required for the binary representation of β. The assumption on the form of β is unessential, and later in Section 2.2 we indicate that no difficulties are presented if β is not of the assumed form.

We give two different but connected reasons which separately lead to the rather consequential conclusion that $F \geq L$ if the resulting system is to have certain essential properties, including the synchronization capability. The first reason stems directly from the synchronization requirements. We show that the latter requires the map f to incorporate certain contraction properties which in turn can be possible only if the internal machine word has at least L fractional bits. The second related reason is that fewer than L fractional bits gives rise to rounding errors in each iteration of the recursion which makes it hard to predict the effective value of the leakage parameter. Recall from Section 1.1.1 the stringent requirements on the leakage parameter.

Below we amplify both the above arguments. This discussion will motivate a more exact treatment in Section 2.2, which will also provide answers to the questions raised here.

Consider (10′) in conjunction with the synchronization requirements (i) and (ii). For the first of the synchronization requirements to be satisfied, it is apparent that it is necessary and sufficient that

$$|f(y) - f(y')| \leq |y - y'| \tag{13}$$

for all machine words y and y'. We refer to the above property of the map f as the *weak contraction everywhere* property. The map f shown in Fig.

* This is the case if K is 5 or 6. If K is larger, then it is possible to relax the second requirement by requiring that the offset in the integer parts of the machine words be reduced to a small number (instead of requiring them to be identical). Thus it is possible to trade a higher K for a lower F while keeping $K + F$ fixed. In any case, only minor modifications to the framework that is developed here will allow such cases to be handled.

2 possesses this property by virtue of the fact that the slope of the graph of f is everywhere either 0 (at the breaks) or 1.

For the second of the synchronization requirements to be satisfied, we claim that it is necessary and sufficient that the map f have the following property:

If y and y' are any machine words with different integer parts, then

$$|f(y) - f(y')| \le \delta |y - y'| \text{ for some } \delta < 1. \tag{14}$$

We call the above the *strong contraction across integer boundaries* property. Sufficiency is clear, since we have that during epochs where the machine words do not have identical integer parts and error-free transmission exists,

$$|y(i) - y'(i)| \le \delta^i |y(0) - y'(0)|. \tag{15}$$

Conversely, if (14) is not true, then it is easy to construct examples where the integer parts of the two machine words are different at an unbounded number of time instants. Referring to Fig. 2 we see that the graph of f does *not* possess the strong contraction property (14). To illustrate, suppose that initially the two machine words have different integer parts and that both words occur in the range [2.5,6]; we see from the figure that no mechanism exists to prevent the two words from indefinitely remaining in this range and simultaneously having different integer parts.

We will now argue that the above two contraction properties, together with any weak fidelity criterion relating $f(y)$ to βy, implies that $F \ge L$. Observe that the strong contraction property, (14), requires a "break" (see "breaks" in Fig. 2) in the graph of $f(y)$ just prior to *every* integral value of y. Reason: $y = k - 2^{-F}$ and $y = k$, k integral, have different integer parts. Further, if the local slope of the graph of $f(y)$ is not zero, then by virtue of the weak contraction property it is either 1 or -1. Finally, if F fractional bits are used, then each unit interval of y is composed of 2^F intervals of equal length corresponding to that many distinct machine words. These three considerations show that the

$$\text{average slope of the graph of } f(\cdot) \le \frac{2^F - 1}{2^F} = 1 - 2^{-F}. \tag{16}$$

But $f(y)$ is supposed to approximate βy, $\beta = 1 - 2^{-L}$. Thus, just about any weak fidelity criterion will give that the smallest value of F, which allows the map f to have the properties required of it, is L.

Our second reason is closely related to the aforementioned fidelity criterion. Implicit in a choice of a leakage parameter β with a large number of fractional bits, L, in its binary representation (e.g., $\beta = 63/64$) is the requirement that the absolute rounding error in each iteration of (10′), $|f\{y(i)\} - \beta y(i)|$, be not larger (at least not by much) than an error in the least significant bit of β, i.e. 2^{-L}:

$$|f(y) - \beta y| < 2^{-L}, \qquad \text{for all machine words } y. \tag{17}$$

Otherwise, there is no *a priori* need to specify β to that degree of precision. (Our experience with the idealized system, discussed previously, shows that it is indeed necessary to specify β to a high degree of precision.) A little thought will convince the reader that for such a bound, (17), on the rounding error to be valid it is necessary that the internal machine word have at least L fractional bits.

In Section 2.2 we show that it is possible to obtain maps f with the weak and strong contraction properties that satisfy the fidelity criterion with the minimum possible number of fractional bits, i.e., $F = L$. We show that, in fact, the maps obtained are *unique*. The results will show that, for our maps, the offset in machine words during error-free transmission decreases exponentially fast to a value less than unity, after which there may be at most $(2^L - 1)$ occasions at which the integer parts differ.

Let us now consider in broad terms what the preceding results imply in terms of the cost and complexity of the digital implementation of the scheme for adaptive quantization discussed in Section 1.1.1. Consider the fairly typical case where the total number of integral log step sizes is 64 and $\beta = 63/64$, i.e. $K = 6$ and $L = 6$. We now know that the total word length should be at least 12 bits. Consider the implications on the associated ROM size. The table stored in the ROM will have 2^{12} addresses, each address containing 12 bits, giving a total memory size in the transmitter and receiver of about 50K bits each! Moreover, with each additional bit in the internal word, the memory requirement more than doubles.*

In the next section, we propose a new adaptation algorithm and specify the required arithmetic. The new algorithm requires significantly fewer fractional bits in the machine words while possessing the necessary synchronization capability.

II. THE PROPOSED SYSTEM

2.1 Idealized description

We propose the following *interleaved-leakage algorithm* (ILA) as the basis for the machine adaptation of the log step size. For fixed parameters I and γ, $I \geq 2$ and $0 < \gamma < 1$ [see eq. (6)]:

$$\left.\begin{aligned} d(i+1) &= \gamma d(i) + m(i) \\ d(i+2) &= d(i+1) + m(i+1) \\ d(i+I) &= d(i+I-1) + m(i+I-1) \end{aligned}\right\} i = 0, I, 2I, \cdots. \tag{18}$$

* We have considered the possibility of exploiting the idea due to Croisier et al. (Ref. 11) and Peled and Liu (Ref. 12) wherein the ROM size may be reduced at the cost of increased processing time. The processing times available and the relative costs do not make this approach particularly promising at the present time. However, it is an approach worth keeping in mind.

Here γ is the leakage constant, and leakage is introduced only once in every I iterations. Thus we refer to I as the *interleaving interval*. The $m(\cdot)$ terms are the log multipliers, $m(\cdot) \epsilon \{m_1, \cdots, m_N\}$, and the selection rule is as in (6b). However, in general, the optimum values of the multipliers may be different from the ones in the scheme described in Section 1.1.1 (we refer to the latter scheme as the uniform-leakage algorithm, or sometimes only as ULA).

We observe that for two geographically separated implementations, $\{d(\cdot)\}$ and $\{d'(\cdot)\}$, of the recursion in (18) subject to possibly different initial values, $d(0)$ and $d'(0)$, but identical $\{m(\cdot)\}$ sequences, as is the case during error-free transmission, we have for the offset,

$$|d(i) - d'(i)| = (\gamma^{1/I})^i |d(0) - d'(0)|, \qquad i = 0, I, 2I, \cdots. \tag{19}$$

Comparing (19) with the similar expression in (8) for the offset in ULA, we find that the capability for recovery from channel errors is comparable in the two schemes if

$$\gamma^{1/I} = \beta. \tag{20}$$

The above is a key relation. Table I tabulates typical values of β and the corresponding choices of γ and I which give comparable recovery capabilities. There are small, inconsequential errors in the table which has been obtained from the approximation $\gamma = [1 - (1 - \beta)]^I \approx 1 - I(1 - \beta)$ for small values of $(1 - \beta)$.

The important point about the table is that, for given β, the fractional bits required for a binary representation of the equivalent value of γ is reduced by an additional bit for every doubling of the interleaving interval, I, in ILA. This simple fact is at the heart of the system that is proposed.

Table I — Leakage parameters (β, γ) and interleaving intervals (I) for comparable synchronization capabilities in the uniform and interleaved leakage algorithms*

	γ (ILA)				
β (ULA)	$I = 2$	$I = 4$	$I = 8$	$I = 16$	$I = 32$
127/128	63/64	31/32	15/16	7/8	3/4
63/64	31/32	15/16	7/8	3/4	
31/32	15/16	7/8	3/4		

* We have stopped short of using $\gamma = 1/2$ for two reasons. First, there may be no advantage in reducing γ beyond 3/4 because two fractional bits may be required in any case on account of the specification of the log multipliers, m_r. Second, the change in the step size may be too drastic, and this may be reflected in the subjective quality. However, it is a possibility worth keeping in mind.

A slight generalization of the proposed scheme would have the multiplier set in the iteration where leakage γ is inserted to be different from the common multiplier set in all other iterations. This generalization provides no gain when the midpoint of the input signal intensities ($\hat{\sigma}$ of Section IV) is scaled to be unity, which is the case considered in the simulations reported in Section IV. Goodman[10] has suggested that, when $\hat{\sigma} \pm 1$, the log multipliers in the leaky iterations be $m(\cdot) + (1 - \gamma)\log_Q \hat{\sigma}$, where $\{m(\cdot)\}$ are the log multipliers in the nonleaking iterations.

2.2 The digital implementation

We now consider the digital implementation of the idealized recursion (18).

Here we let L, an integer, be such that $1 - 2^{-L+1} < \gamma \leq 1 - 2^{-L}$. We make the simplifying, and inessential, assumption that $\gamma = 1 - 2^{-L}$; in this case, the binary representation of γ requires L fractional bits. (Later we indicate through an example that it is easy to make the modifications which allow other values of γ to be used.) Assume K integer and L fractional bits for the internal machine words. Thus, following the discussion on the synchronization requirements in Section 1.1.2, we are assuming that the fractional bits in the machine words are the minimum necessary for the system objectives to be satisfied. Finally, assume that the log multipliers $\{m_r\}$ are specified to L fractional bits.

The internal description of the machine is

$$\left.\begin{array}{l} y(i+1) = f\{y(i)\} + m(i) \\ \underline{y(i+2) = y(i+1) + m(i+1)} \\ \hdashline y(i+I) = y(i+I-1) + m(i+I-1) \end{array}\right\} i = 0, I, 2I, \cdots, \qquad (21)$$

where $y(\cdot)$, the internal machine word, is a $(K+L)$-bit word with L fractional bits. In (21), f maps $(K+L)$-bit words with L fractional bits into other such words. The mapping f may be implemented most easily using ROMs; the characterization of the map f that we give below is a recipe for the programming of the ROMs.*

The integral log step size $d(\cdot)$ is obtained from the internal word $y(\cdot)$ by a rule determined by an *external arithmetic*. We consider two natural and simple external arithmetics, rounding and truncation. Thus,

$$\text{Rounding: } d(\cdot) = [y(\cdot)]_{\text{round}} \qquad (22a)$$

$$\text{Truncation: } d(\cdot) = [y(\cdot)]_{\text{truncate}}. \qquad (22b)$$

We mean that if, for integral k, $k - 0.5 < y \leq k + 0.5$, then $[y]_{\text{round}} = k$; if $k \leq y < k + 1$ then $[y]_{\text{truncate}} = k$.

* Observe that the specifications of the maps given here and in Appendix A apply as well to the uniform leakage algorithm described in Section 1.1, provided β replaces γ and the appropriate value of the parameter L associated with the leakage parameter β in ULA is substituted.

We consider first the truncating external arithmetic. Following the discussion in Section 1.1.2, we impose the following requirements on the map f. (It is understood that all arguments of the map have L fractional bits.)

$$(i)\ \forall\, \sigma_1,\sigma_2, \qquad |f(\sigma_1) - f(\sigma_2)| \le |\sigma_1 - \sigma_2|:$$

"weak contraction everywhere." (23)

$$(ii)\ \begin{array}{l}\sigma_1 \epsilon [k,k+1)\\ \sigma_2 \epsilon [k+1,k+2)\\ k \text{ integral}\end{array} \Rightarrow \frac{|f(\sigma_1) - f(\sigma_2)|}{|\sigma_1 - \sigma_2|} \le \delta < 1:$$

"strong contraction across integer boundaries." (24)

$$(iii)\ \forall\, \sigma, \qquad |f(\sigma) - \gamma\sigma| < 2^{-L}:$$

"fidelity of discrete map to continuous map." (25)

Recall from Section 1.1.2 that the first two properties are equivalent to the synchronization requirements. We also know that these two conditions together with almost any weak fidelity criterion relating $f(\sigma)$ to $\gamma\sigma$ implies that the number of fractional bits in the machine words is at least L. We find that we can construct maps f which satisfy in addition the fidelity criterion in (*iii*) without incurring the penalty of using more than L fractional bits. Also, as discussed previously, the fidelity criterion in (*iii*) is important in itself.

In Appendix A we give the complete specification of a map for each value of L. In Fig. 3a, we show the graph of the map f for the example of $\gamma = 3/4$, where $L = 2$. In Appendix A we also show that there is only one such map f for any given L which satisfies conditions (*i*) to (*iii*), (23) to (25). Further, for this unique map the value of the contraction parameter δ in (24) is $2\gamma/(1+\gamma)$.

When the external arithmetic is the rounding arithmetic (22a), the

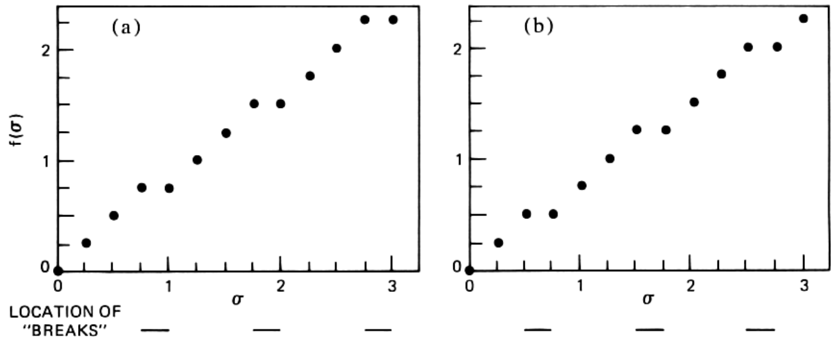

Fig. 3—Machine arithmetics incorporating contraction properties and fidelity criterion for (a) truncating and (b) rounding external arithmetics. $\gamma = 3/4$ and $L = 2$ (see Section 2.2).

resulting map f is somewhat different. Appendix A gives the complete specifications of the maps for all values of L; these maps are also unique. Figure 3b shows the graph of one such map.

Recall that earlier we made the simplifying assumption that $\gamma = 1 - 2^{-L}$. In general, L is defined to be such that $1 - 2^{-L+1} < \gamma \leq 1 - 2^{-L}$. Figure 4 illustrates a map f for the case of $\gamma = 5/8$ ($L = 2$) and the truncating external arithmetic. It may be verified that all the requirements in (23) to (25) are satisfied. We may similarly generate maps satisfying the requirements for arbitrary rational values of γ.

Note that the maps obtained are rather special and quite distinct from the usual maps encountered in digital signal processing.

Another point to note is that while we have specified arithmetics which use the minimum number of fractional bits, $F = L$, additional fractional bits, if they are available, may be put to use by incorporating more than one break in the graph of $f(\sigma)$ per unit interval of σ. The net effect is to give superior synchronization capability.

Finally, note that the implementation of (21) requires by way of hardware only the ROMs, for implementing the map f, and adders. However, the ROMs are used only once in every I iterations. This provides an ideal opportunity for multiplexing the ROMs between different channels and different frequency bands in subband coding[13] applications.

2.3 Synchronization in the digital implementation

We give some bounds on the offset between transmitter and receiver during periods of error-free transmission.

By y and y', two machine words, having different integer parts we mean in the following that $[y]_{\text{round}} \neq [y']_{\text{round}}$ or $\lfloor y \rfloor_{\text{truncate}} \neq \lfloor y' \rfloor_{\text{truncate}}$, depending on the external arithmetic chosen. Thus, depending upon whether the two machine words have identical or different integer parts, the corresponding log step sizes are identical or different, respectively.

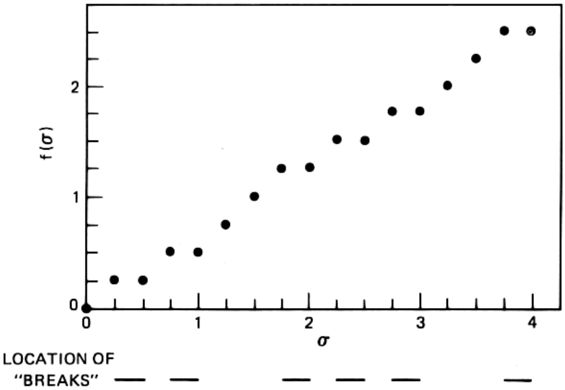

Fig. 4—Machine arithmetic for $\gamma = 5/8$ ($L = 2$) for two fractional bits in machine word and truncating external arithmetic. The contraction requirements and fidelity criterion are satisfied.

Suppose the machine implementations of the recursions in (18) in the transmitter and receiver during error-free transmission are: $i = 0, I, 2I, \cdots$

$$\begin{array}{l} y(i+1) = f\{y(i)\} + m(i) \\ y(i+2) = y(i+1) + m(i+1) \\ \hdashline y(i+I) = y(i+I-1) + m(i+I-1) \end{array}$$

$$\begin{array}{l} y'(i+1) = f\{y'(i)\} + m(i) \\ y'(i+2) = y'(i+1) + m(i+1) \\ \hdashline y'(i+I) = y'(i+I-1) + m(i+I-1). \end{array} \quad (26)$$

Observe that

$$\begin{aligned} |y(i+I) - y'(i+I)| = \cdots = |y(i+1) - y'(i+1)| \\ = |f\{y(i)\} - f\{y'(i)\}|. \end{aligned}$$

Now from (23) and (24),

$$\begin{aligned} &|f\{y(i)\} - f\{y'(i)\}| \\ &\leq |y(i) - y'(i)| \text{ if } y(i) \text{ and } y'(i) \text{ have identical integer parts,} && (27) \\ &\leq \delta |y(i) - y'(i)| \quad \text{if } y(i) \text{ and } y'(i) \text{ have different integer parts.} && (28) \end{aligned}$$

By repeated application of (28) we see that, if $|y(0) - y'(0)| > 1$, then

$$|y(j) - y'(j)| < 1 \text{ for all } j > I \log \{|y(0) - y'(0)|\}/\log(1/\delta). \quad (29)$$

Thus, once the offset is reduced to less than unity it subsequently remains thus.

Now consider the case where $|y(0) - y'(0)| < 1$. Consider the time instants j which are integral multiples of I. There can be at most $(2^L - 1)$ such time instants at which the integer parts differ. This is because a reduction of 2^{-L} in the offset is guaranteed by (28) in every such time instant. However, at time instants which are not integral multiples of I, the convergence of the integer parts is not quite as strong and is a penalty (which we believe to be insignificant) of ILA.

III. ANALYSIS: PROBABILISTIC ASPECTS

In this section, we investigate the probabilistic behavior of the log step sizes, $\{d(\cdot)\}$, when the input signal variables, $\{x(\cdot)\}$, are random and channel errors are absent. Clearly such an analysis is called for if we are to be able to guarantee certain qualitative features of performance that are basic and necessary in adaptive PCM systems.[4,5] The key notions of the bias function, central log step sizes, and load curves are introduced and their qualitative behavior pinned down.

For our purposes here, the defining equations for the log step sizes are in (18); the selection rule for the multipliers are in (6b). The key assumption that is made throughout this section is that $\{x(\cdot)\}$ is a sequence of independent, identically distributed random variables with mean zero and standard deviation σ. We sometimes refer to σ as the signal intensity. In keeping with the characteristics of speech, we are interested in σ in the range of $\sigma_{\max}/\sigma_{\min} = 100$, or even 400 (40 and 52 dB ranges, respectively).

3.1 The bias function

Define the bias function $B(\cdot|\sigma)$ to be

$$B(d|\sigma) \triangleq E[d(i+I)|d(i)=d] - d, \qquad i = 0, I, 2I, \cdots. \tag{30}$$

A little thought will show that the right-hand side of (30) does not depend on i—a consequence of the iid assumption on the input signal variables. Different values of σ will generally yield different bias functions, which explains the notation. In engineering parlance, $B(d|\sigma)$ measures, for initial log step size d, the mean drift of the log step size after one cycle of updating of the log step size.

We are able to show for a wide range of values of σ that the bias functions consistently have a distinctive form, depicted in Fig. 5, of considerable significance. In particular, we show that $B(d|\sigma)$ is positive when d is sufficiently small, and negative when d is sufficiently large. Further, under a rather mild restriction, we can prove the consequential result that $B(d|\sigma)$ is monotonic, decreasing with increasing d. The above results in their precise forms are proven in Appendix B. The restriction that is mentioned above is interesting in itself and, roughly, it calls for a propensity for the expected log step sizes after one iteration to be ordered in the same way as the initial log step sizes. This turns out to require, roughly, that $(m_N - m_1)$ be not too large.

The importance of the above results is on account of the following corollary which we state in qualitative terms:

> If $(m_N - m_1)$ is not too large, then there exists a unique root, or zero-crossing, of the bias function $B(\cdot|\sigma)$.

Without the monotonicity of the bias function, the possibility exists of

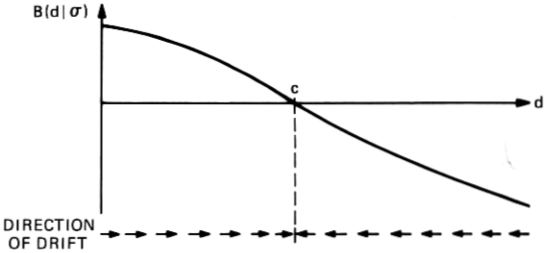

Fig. 5—Sketch of a bias function.

there being many roots with a consequent dilution of the importance that we attach to the root.

Let c denote such a root for a fixed value of σ, Fig. 5:
Definition of c:

$$B(c \mid \sigma) = 0. \tag{31}$$

We refer to c as the *central log step size* (for signal intensity σ). For a different value of σ and hence a different bias function, the root will generally be different, and to make this dependence quite clear we use the notation $c(\sigma)$.

As the terminology implies, we expect the probability distribution of the log step size to have a concentration of mass around $c(\sigma)$ whenever the signal intensity is σ. The reason for expecting this (see direction of drift indicated by arrows at bottom of Fig. 5) is that, whenever the log step size is not at $c(\sigma)$, the mean drift of the log step size is toward $c(\sigma)$.

The above conclusion is amply borne out by computational results (see Section IV). We find, for instance, that the fit between $c(\sigma)$ and the mean log step size in steady state is extremely good for a rather broad range of values of σ.

In summary, the dual properties of the central log step size (namely, that it predicts so well the mean log step size and that it is so much more tractable and easily obtained) explain the emphasis that we place on the notion of the central log step size.

3.1.1 Method for generating the bias function

The following recursive formula which is developed in Appendix B is the most effective method we know for obtaining the bias function. First, it is necessary to define the following functionals:

$$b_r(\tau) \triangleq 2 \int_{\xi_{r-1}Q^{\tau}}^{\xi_r Q^{\tau}} p(\mu) d(\mu), \qquad 1 \leq r \leq N, \tag{32}$$

where $p(\mu)$ is the common pdf of the input signal variables $\{x(\cdot)\}$. (It is slightly simpler to make as we do the inconsequential assumption that $p(\cdot)$ is symmetrical about 0.) Then $B(d \mid \sigma)$ is obtained as the solution of the following functional recursion:

$$B_0(d \mid \sigma) = 0, \ \forall d$$

$$B_k(d \mid \sigma) = \begin{cases} \sum\limits_{r=1}^{N} b_r(d)\{B_{k-1}(d + m_r \mid \sigma) + m_r\}, \ 1 \leq k \leq I - 1 \\ -(1-\gamma)d + \sum\limits_{r=1}^{N} b_r(d)\{B_{k-1}(\gamma d + m_r \mid \sigma) + m_r\}, \ k = I. \end{cases} \tag{33}$$

Finally, $B(d \mid \sigma) = B_I(d \mid \sigma)$.

The above formula is used in the following manner: Assume that the function $B_{k-1}(d\,|\,\sigma)$ is known for all values of d. Use (33) to generate next the complete function $B_k(d\,|\,\sigma)$. After I such iterations, the resulting function $B_I(d\,|\,\sigma)$ is in fact $B(d\,|\,\sigma)$.

The reader is referred to eq. (50), Appendix B, for the probabilistic interpretations of the ancillary functions $B_k(\cdot\,|\,\sigma)$.

The above formula is used in the analysis presented in Appendix B to determine the previously mentioned qualitative properties of the bias function $B(d\,|\,\sigma)$.

Figure 6 is a plot of the bias function $B(d\,|\,1)$ for a 16-level quantizer and normally distributed input signal variables. The interleaving interval, I, is 16. Observe in the figure that the graph is for d in the range [−200,800]. Values of d outside this range are not of much interest, since the maximum range of the log step sizes in this example is $[\mathrm{Im}_1/(1-\gamma), \mathrm{Im}_N/(1-\gamma)] = [-163{,}828]$.

3.2 Load curves

The load curves provide information regarding the manner in which the log step sizes depend on the input signal intensity, σ. We use the term to describe a graph of $\log_Q \sigma$ vs. $\overline{d}$, where $\overline{d}$ is the mean log step size in steady state for signal intensity σ. Naturally, the range of σ should cover the range of values expected in the specific application.

From our previous discussion on bias functions and their roots, the

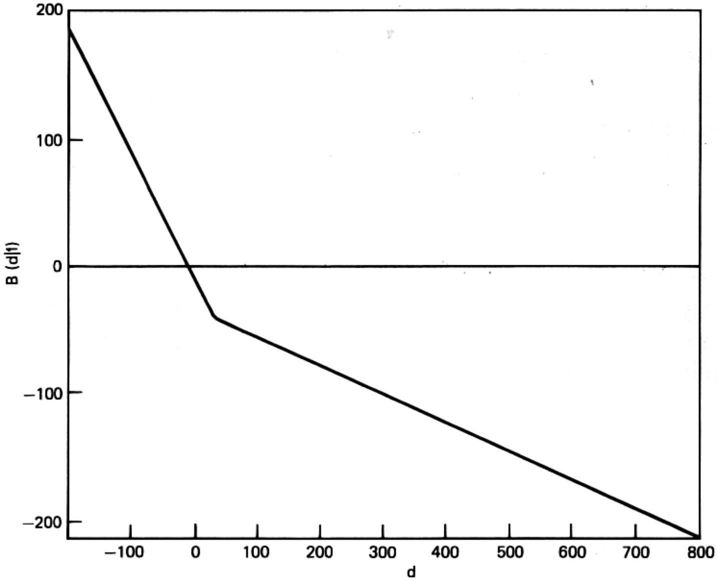

Fig. 6—The bias function for uniform 16-level quantizer and normally distributed input signal variables, $\sigma = 1$. Interleaving internal, $I = 16$ and $\gamma = 0.777$. The log multipliers are given in (39).

central log step sizes, we expect a plot of $\log_Q \sigma$ vs. $c(\sigma)$ to be a rather good fit to the load curves.

The utility of the load curve derives from the fact that it may be visually compared with a plot of the ideal log step size with respect to σ. This information may be obtained from solving a variational problem as is done by Max,[14] who has also tabulated the solutions for the case of normally distributed input signal variables. In any case, the solutions to the variational problem for the *optimum log step size* $\hat{d}(\sigma)$ have the following form

$$\hat{d}(\sigma) = \log_Q \sigma + \hat{D}, \tag{34}$$

where $\hat{D}$ is a constant which depends on the fixed parameters of the quantizer and, importantly, on the common pdf of the input signal variables.

Figure 7 is a plot of the load curve obtained for the 16-level quantizer.

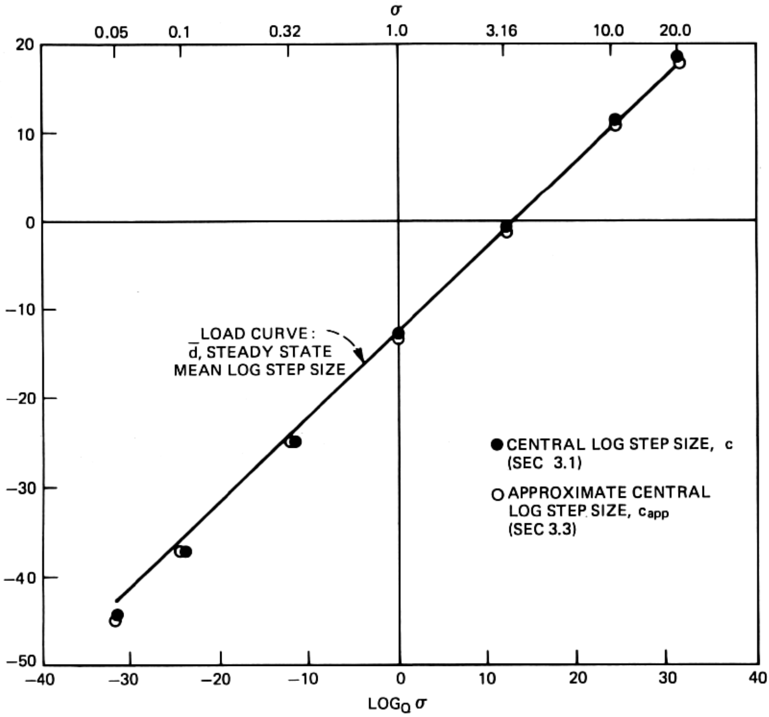

Fig. 7—Load curve ($\bar{d}$), central log step size (c), and approximate log step size (c_{app}) for uniform 16-level quantizer and Gaussian, zero-mean, input signal variables of variance σ^2. The log multipliers are given in (39) and $Q = 1.1$. Interleaving interval, $I = 16$ and leakage, $\gamma = 0.777$.

3.3 The almost-linear dependence of the central log step sizes on signal intensity

Even though a plot of $\log_Q \sigma$ vs $c(\sigma)$ may be expected to be a rather good approximation to, and certainly simpler to obtain than, the load curve ($\log_Q \sigma$ vs $\overline{d}$), it is an unfortunate fact that it is not a very simple matter to obtain $c(\sigma)$. However, our graphs of $c(\sigma)$ have consistently displayed a most remarkable trait, namely, the almost-linearity of $c(\sigma)$ with respect to σ. Intrigued by this feature, we found in an earlier study[9] that it could be explained if the following rather unusual approximation is effective:

$$\int_0^y p(\mu) d\mu \approx \alpha_1 \log y + \alpha_2, \tag{35}$$

where α_1 and α_2 are constants and $p(\cdot)$ is the common pdf of the input signal variables scaled to have unit variance.

Certainly, the above cannot be a good approximation when either y is very small or y is very large. But, as we see in Appendix C, we need the above to be a good approximation only for a limited range of y; specifically, the range of y is required to include the range encountered by $\xi_1 Q^{d(\cdot)}$ at one end, and $\xi_{N-1} Q^{d(\cdot)}$ at the other end, where $d(\cdot)$ is the typical log step size. It turns out that in the important cases where $p(\cdot)$ is either Gaussian or Laplacian, the range of validity of (35) is adequate, at least for the analysis of quantizers with up to 16 levels ($N = 8$). Further details may be found in Ref. 9. For both these distributions, we have found (35) to be an effective approximation in the range $1/3 \leq y \leq 2$. For the former distribution, we have found good fits to be obtained if

$$\alpha_1 = 0.44 \text{ and } \alpha_2 = 0.34.$$

(Below, we find it more convenient to express the rhs of (35) as $\alpha_1 \log_Q y + \alpha_2$.)

With (35) as the sole approximation, in Appendix C we go through the involved and tedious process of approximating the bias function and thence deriving its root. The final result, however, is the following remarkably informative formula ($c_{\text{app}}(\sigma)$ is the *approximate central log step size* for signal intensity σ):

$$c_{\text{app}}(\sigma) = S \log_Q \sigma + D, \tag{36}$$

where

$$S = \frac{1}{1 + \dfrac{(1-\gamma)\{1 - 2\alpha_1(m_N - m_1)\}^{I-1}}{1 - \{1 - 2\alpha_1(m_N - m_1)\}^I}} \tag{37}$$

and

$$D = \frac{m_N - 2 \sum_{r=1}^{N-1} (m_{r+1} - m_r)(\alpha_1 \overline{\xi}_r + \alpha_2)}{2\alpha_1(m_N - m_1)} S. \tag{38}$$

Let us remark on certain features of the formula. Observe that, on account of α_1 being small, $1 - 2\alpha_1(m_N - m_1) > 0$ almost certainly; for example, $\alpha_1 = 0.018$ when Q [see eq. (3)] is 1.1 and the input signal variables are Gaussian. Consequently, we observe, from the formula in (37) for the slope S, that $S < 1$. Now the ideal slope is 1 [see (34)]. Thus eq. (37) expresses the undesirable but expected fact, alluded to earlier in Section 1.1, that decreasing the leakage parameter γ has the effect of driving the load curve away from the ideal, as sketched in Fig. 8.

As a digression, note that when $\gamma = 1$, the slope S is unity. This is, of course, known to be the case.[4,5] We may also compare the expression for S with a similar expression for ULA derived in Ref. 9—the two expressions are practically identical when $\gamma = \beta^I$ [eq. (20)] and β is close to unity. This important fact, also confirmed in simulations in the example of Section IV, shows that in terms of the loading we expect the behavior in ILA and ULA to be roughly equivalent.

One of the uses that formulas (36) to (38) can be put to is in the optimum choice of the multipliers. The approach we take is that γ and $(m_N - m_1)$ are determined *a priori* on the basis of requirements arising from the quality of synchronization and transient response, respectively. This then fixes the value of S, eq. (37). However, there is still considerable freedom in the choice of the quantities $(m_{r+1} - m_r)$, $1 \le r \le N - 1$, and thereby in the choice of the value of D, eq. (38). This degree of freedom may be exploited to determine the point of intersection of the graph of $c_{\text{app}}(\sigma)$ and the ideal graph, which are shown in Fig. 8. A sensible choice for the point of intersection is at the signal intensity, σ, that is most likely to be encountered. Usually,[1] this is at the midpoint of the range of signal intensities expected to be encountered in the application.

IV. COMPUTED RESULTS

Throughout this section, the input signal variables $\{x(\cdot)\}$ are independent, Gaussian, random variables with mean zero and standard deviation σ. The signal intensity σ is varied about a central value of 1.0.

The quantizer is a 16-level, uniform quantizer, i.e., $N = 8$, $\xi_r = r$, $1 \le r \le N - 1$, and $\eta_r = r - ½$, $1 \le r \le N$. Throughout, the log base for the step sizes and multipliers, Q, is 1.1.

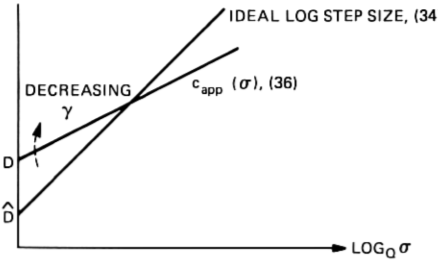

Fig. 8—The behavior of the central log step size compared to the ideal. See eqs. (34) and (36).

For the uniform-leakage algorithm, ULA, we used as the leakage constant $\beta = 63/64$. The multipliers for ULA are approximately those used by Rosenthal et al.[15] after correction, in the manner suggested in Ref. 1, for the following specifications: In the notation of Ref. 1, $\hat{\sigma}$ = midpoint of signal intensities = 1.0, the ideal loading factor = ideal step size/signal intensity = 0.257. This procedure gave the following values for the log-multipliers for ULA,

$$m(1) = m(2) = m(3) = m(4) = -2.25;\ m(5) = m(6) = 2.50;\ m(7) = 7.25;\ m(8) = 11.50. \quad (39)$$

The multipliers used for the interleaved algorithm, ILA, were also selected to be those given above. We are aware of the advantages of fine tuning the multipliers and Q to take advantage of the special features of ILA, but decided on balance to keep the multipliers and Q unchanged. We found that, as it stands, the transient behavior for ILA is slightly superior to that of ULA; reducing Q in ILA equalizes the transient behavior in the two schemes and yields s/n ratios slightly better than those reported here for ILA.

4.1 Computed load curve, central log step sizes, and their approximation

We illustrate the above notions for the interleaved leakage algorithm for the case of the interleaving interval, $I = 16$. We set $\gamma = \beta^I = 0.777$. Figure 7 plots three quantities with respect to $\log_Q \sigma$: (*i*) $\bar{d}$, the steady-state, mean log step size. This was obtained from 10,000 iterations; (*ii*) $c(\sigma)$, the central log step size defined in (31); (*iii*) $c_{\text{app}}(\sigma)$, the approximate central log step size as given by (36) to (38).

For the given specifications,

$$c_{\text{app}}(\sigma) = 0.99 \log_Q \sigma - 13.20.$$

To clarify Fig. 7, we have also tabulated in Table II the values of the above variables at seven values of σ.

Table II — Computed load curve, central log step sizes, and their approximation (I = 16 and γ = 0.777)

σ, signal intensity	0.05	0.10	0.3162	1.0	3.162	10.0	20.0
$\bar{d}$, steady state mean log step size	−42.40	−35.53	−24.14	−12.46	−0.84	10.81	17.88
$c(\sigma)$, central log step size	−44.35	−37.07	−25.00	−12.93	−0.85	11.24	18.51
$c_{\text{app}}(\sigma)$, approximate central log step size	−44.63	−37.36	−25.28	−13.20	−1.12	10.96	18.23

4.2 *S/N ratios and load curve for ULA and ILA*

Table III compares signal-to-noise ratios for the two schemes for a variety of interleaving intervals. The signal energy is simply the energy of the variables $\{x(\cdot)\}$. The noise is exactly the difference between the input signal variable and its reconstruction at the receiver, *assuming error-free transmission.* Thus, the reported s/n ratios reflect the effect of the step-size adaptation algorithms but do not measure synchronization capabilities of the systems—the latter is measured separately in Section 4.3.

Note the almost identical s/n ratio performance for the two algorithms, ULA and ILA.

Tables IV and V compare the mean and standard deviations of the log step sizes. Again, note the uniformity of the results for the ULA and ILA; the loading characteristics of the two approaches are almost identical.

Table III — Signal-to-noise ratios (dB)

		ILA; $I = 2$	ILA; $I = 4$	ILA; $I = 8$	ILA; $I = 16$
	ULA	$\gamma = \beta^2 =$	$\gamma = \beta^4 =$	$\gamma = \beta^8 =$	$\gamma = \beta^{16} =$
σ	$\beta = 63/64$	0.969	0.939	0.881	0.777
0.10	14.89	14.92	14.90	14.70	14.16
0.3162	14.55	14.57	14.56	14.48	14.17
1.0	14.19	14.16	14.18	14.14	14.13
3.162	13.80	13.77	13.63	13.84	13.76
10.0	13.37	13.30	13.36	13.31	13.24

Table IV — Steady-state mean log step sizes

		ILA; $I = 2$	ILA; $I = 4$	ILA; $I = 8$	ILA; $I = 16$
	ULA	$\gamma = \beta^2 =$	$\gamma = \beta^4 =$	$\gamma = \beta^8 =$	$\gamma = \beta^{16} =$
σ	$\beta = 63/64$	0.969	0.939	0.881	0.777
0.10	−35.75	−35.78	−35.72	−35.66	−35.53
0.3162	−24.12	−24.11	−24.10	−24.13	−24.14
1.0	−12.47	−12.54	−12.47	−12.54	−12.46
3.162	−0.88	−0.90	−0.87	−0.82	−0.84
10.0	10.74	10.81	10.84	10.78	10.81

Table V— Standard deviation of log step size in steady state

		ILA; $I = 2$	ILA; $I = 4$	ILA; $I = 8$	ILA; $I = 16$
	ULA	$\gamma = \beta^2 =$	$\gamma = \beta^4 =$	$\gamma = \beta^8 =$	$\gamma = \beta^{16} =$
σ	$\beta = 63/64$	0.969	0.939	0.881	0.777
0.10	4.48	4.50	4.57	4.75	5.26
0.3162	4.56	4.63	4.64	4.74	4.97
1.0	4.70	4.69	4.74	4.74	4.85
3.162	4.80	4.80	4.81	4.81	4.80
10.0	4.87	4.90	4.88	4.89	4.97

4.3 The steady-state mean offset in the transmitter and receiver log step sizes

Here we present some computational results connected with the *steady state,* joint distribution of the transmitter and receiver log step sizes assuming, as we have done throughout Section IV, that the input signal variables are independent, normally distributed.

The channel is assumed to be memoryless; further, the event that a transmitted "1" is received as a "0" and the event that a transmitted "0" is received as a "1" have the common probability p. Thus, p is the *bit error probability.* In the numerical results presented below, the following typical value for the bit error probability is assumed: $p = 10^{-4}$.

Two geographically separated implementations of the interleaved leakage algorithm, (18), are assumed to be occurring: $i = 0, I, 2I, \cdots$

$$\begin{array}{l} d(i+1) = \gamma d(i) + m(i) \\ \underline{d(i+2) = d(i+1) + m(i+1)} \\ \hdashline d(i+I) = d(i+I-1) + m(i+I-1) \end{array}$$

$$\begin{array}{l} d'(i+1) = \gamma d'(i) + m'(i) \\ \underline{d'(i+2) = d'(i+1) + m'(i+1)} \\ \hdashline d'(i+I) = d'(i+I-1) + m'(i+I-1) \end{array}. \qquad (40)$$

The information regarding the log multipliers $m(\cdot)$ are assumed to be coded in the manner shown in Fig. 1 and transmitted through the channel described above. The log multipliers $m'(\cdot)$ are the log multipliers corresponding to the received code word.

By the "steady state mean offset in the transmitter and receiver log step sizes" we mean the quantity $\overline{e}$ where

$$\overline{e} = \lim_{i \to \infty} E\{d(i) - d'(i)\} \qquad (41)$$

In Appendix D we show that $\overline{e}$ is given by the following expression:[9]

$$\overline{e} = \frac{I}{1-\gamma} \sum_{r,s=1}^{N} (m_r - m_s) T_{sr} p_r, \qquad (42)$$

Table VI — Steady state mean offset in transmitter and receiver log step sizes. Bit error probability in channel, p = 10^{-4}

σ	ULA $\beta = 63/64$	ILA; $I = 2$ $\gamma = \beta^2 =$ 0.969	ILA; $I = 4$ $\gamma = \beta^4 =$ 0.939	ILA; $I = 8$ $\gamma = \beta^8 =$ 0.881	ILA; $I = 16$ $\gamma = \beta^{16} =$ 0.777
0.10	−0.025	−0.025	−0.025	−0.026	−0.026
0.3162	−0.022	−0.022	−0.022	−0.023	−0.024
1.0	−0.020	−0.020	−0.020	−0.021	−0.022
3.162	−0.018	−0.018	−0.018	−0.018	−0.020
10.0	−0.015	−0.015	−0.015	−0.016	−0.017

where $T = \{T_{sr}\}$ is the *channel transition matrix* given below and p_r is the steady state probability that the rth code word is transmitted ($00\cdots0$ is the first code word, $11\cdots1$ is the last, Nth, code word; the sign bit is ignored).

The channel transition matrix T is defined thus:

$$T_{sr} \triangleq \Pr\,[s\text{th code word recd.}\,|\,r\text{th code word trans.}].$$

In the special case where the codes are as shown in Fig. 1, the elements of the matrix are obtained in a simple manner from the Hamming distance between the code words. Thus, if $d(s,r)$ is the Hamming distance between the sth and rth code words, then

$$T_{sr} = p^{d(s,r)}(1-p)^{\log_2 N - d(s,r)}, \qquad 1 \le s,r \le N. \tag{43}$$

In the example under consideration where $N = 8$, $T_{11} = (1-p)^3$, $T_{12} = p(1-p)^2$, etc.

The formula given in (42) for $\bar{e}$, the mean offset in log step sizes, is extremely useful. To see this, recall that $\bar{e}$ is defined in (41) in terms of the joint behavior of the transmitter and receiver in steady state, yet (42) provides the means for calculating $\bar{e}$ provided only that the transmitter log step size distribution is known, since the quantities $\{p_r\}$ are statistics of the latter distribution. Thus, the considerably harder task of evaluating the joint distribution of the log step sizes at the two different sites is circumvented.

Table VI enumerates the computed steady-state mean offset in transmitter and receiver log step sizes for various signal intensities and designs; note the almost identical performance.

V. SUMMARY

We consider it important that digitally implemented adaptive quantization systems possess two properties which, regardless of the statistics of the input signal, ensure that synchronization in the step-size adaptations at the transmitter and receiver is restored during periods of error-free transmission: The offset in step sizes is monotonic and nonincreasing and the step sizes differ in at most a finite number of sampling time instants. A detailed examination of the uniform-leakage algorithm (ULA) shows that a necessary and sufficient condition for the synchronization requirements to be satisfied is that the internal machine arithmetic, given by the nonlinear map f, possesses certain contraction properties. It is further shown that these contraction properties may exist only if the number of fractional bits (F) in the internal machine word is at least L where the leakage parameter β is such that $1 - 2^{-L+1} < \beta \le 1 - 2^{-L}$. Thus, if $\beta = 1 - 2^{-L}$ then L is the number of fractional bits required for the binary representation of β. We proceed to show that it is actually possible to obtain internal machine arithmetics which satisfy all the requirements with the minimum possible number of fractional bits, i.e., $F = L$. The arithmetics that we obtain are moreover unique. With these arithmetics the offset in machine words during error-free transmission decreases exponentially fast to a value less than unity, after which there may be at most $(2^L - 1)$ occasions in which the step sizes differ.

We give a complete specification of the unique maps f. Thus, in the case where truncation is used to obtain the log step size from the internal machine word, the formula that generates f is:

If $\sigma = k + \mathrm{j}2^{-L}$, where k and j are integral and $0 \leq j \leq 2^L - 1$, then

$$f(\sigma) = k(1 - 2^{-L}) + j2^{-L}.$$

Figure 3a is the graph of the map f for the example of $L = 2$.

Even the minimum length of the machine words translate into large memory requirements in ROM-based implementations. Thus, in the fairly typical case where the total number of step sizes is 64 and the leakage parameter $\beta = 63/64$, we find that the minimum word length is 12 bits, which translates into a ROM size of about 50K bits.

We propose a new adaptation algorithm which is considerably more efficient in terms of the memory used in the implementation. In this algorithm, ILA, leakage is interleaved infrequently but at regular intervals into the recursion for the step-size adaptation. Thus, this scheme has as parameters γ, the leakage parameter, and I, the interleaving interval. We find that, for comparable synchronization capabilities in ULA and ILA, the parameters are related thus:

$$\gamma^{1/I} = \beta.$$

Thus for β close to unity, $\gamma \approx 1 - I(1 - \beta)$. Table I shows that for given β the fractional bits required for the binary representation of the equivalent value of γ is reduced by an additional bit for every doubling of the interleaving interval.

To illustrate, consider the example given above where $\beta = 63/64$; the new scheme provides the option of interleaving leakage once in 8 iterations ($I = 8$) with a leakage parameter $\gamma \approx 7/8$, which has three fractional bits. Thus, for the same total number of step sizes, the total word length required is 9 bits, which translates into an ROM size of about 5K bits and an order-of-magnitude reduction in memory size. Furthermore, the essential costly element of the system, the ROM, is used only once in 8 iterations, thus allowing for the additional multiplexing of the ROM.

The internal machine arithmetic that is proposed for ILA is identical to that specified for ULA, except that the machine word in the former system is of shorter length.

A detailed theoretical analysis of the statistical behavior of the step sizes for independent random inputs is undertaken. Perhaps the most insightful result obtained is a simple formula giving the approximate dependence on the input signal intensity, σ, of the central log step size, $c(\sigma)$, which is the particular log step size about which the distribution of log step sizes is concentrated. The formula depends on only two parameters, α_1 and α_2, of the input signal distribution; in the case of Gaussian input distributions, $\alpha_1 \approx 0.44 \log Q$ and $\alpha_2 \approx 0.34$. This simple formula is given in (36) to (38).

The idealized adaptation algorithms were simulated for a representative 16-level quantizer and independent, Gaussian inputs. In the simulations, the multipliers in ILA were selected to be identical to those used in ULA, although *in general we expect the optimal multipliers to be different for the two schemes.* The results of the simulations show that the performances of the systems are almost identical.

APPENDIX A

Specification of the Machine Arithmetics

We describe first the maps f corresponding to the truncating external arithmetic in (22b) which satisfy conditions (*i*) to (*iii*) given in (23) to (25), Section 2.2. In the example shown in Fig. 3a, observe that the breaks, i.e., zero slope segments between pairs of points, occur just prior to the integral values of σ. This is also the rule by which f is obtained for general values of L.

The following formula generates f for general values of L:

$$\text{If } \sigma = k + j2^{-L},\ k \text{ and } j \text{ integral and } 0 \le j \le 2^L - 1, \tag{44}$$

then $f(\sigma) = k(1 - 2^{-L}) + j2^{-L}$.

Condition (*i*), (23), is trivially verified. For condition (*ii*), (24), note that for all integral k

$$f(k + 1 - 2^{-L}) - f(k + 1) = 0. \tag{45}$$

Thus a strong contraction across integer boundaries exists and, in fact, for σ_1 and σ_2 with different integer parts

$$\frac{|f(\sigma_1) - f(\sigma_2)|}{|\sigma_1 - \sigma_2|} \le \frac{2\gamma}{1 + \gamma}, \tag{46}$$

so that we may take

$$\delta = 2\gamma/(1 + \gamma) < 1. \tag{47}$$

For the final condition (*iii*), we find that

$$0 \le f(\sigma) - \gamma\sigma \le 2^{-L}(1 - 2^{-L}), \tag{48}$$

where the two inequalities become equalities at $\sigma = k$ and $\sigma = k - 2^{-L}$, respectively, whenever k is integral.

We can also show rather easily that the map f given by (44) is unique, i.e., there does not exist any other map satisfying the requirements (*i*) to (*iii*). Uniqueness follows from the following two reasons: (*a*) Condition (*ii*) requires that there be a break in the graph of f between $\sigma = k - 2^{-L}$ and $\sigma = k$, k integral, i.e., $f(k - 2^{-L}) = f(k)$. Reason: $\sigma = k - 2^{-L}$ and $\sigma = k$ have different integer parts. (*b*) In order to satisfy at once both the fidelity condition (*iii*) and the weak contraction (*i*) there can be at most one break in the typical integer interval $[k, k+1]$.

We now describe the slightly different map f which is obtained for the rounding external arithmetic, (22a). For the requirements on f, the only difference is in condition (*ii*) which now reads as follows:

$$(ii')\ \begin{matrix}\sigma_1 \epsilon (k - \tfrac{1}{2}, k + \tfrac{1}{2}] \\ \sigma_2 \epsilon (k + \tfrac{1}{2}, k + \tfrac{3}{2}]\end{matrix} \Rightarrow \frac{|f(\sigma_1) - f(\sigma_2)|}{|\sigma_1 - \sigma_2|} \le \delta < 1. \qquad (24')$$

The graph of f shown in Fig. 3b is obviously similar to the one displayed in Fig. 3a, the main difference being the locations of the breaks which are here positioned immediately following the midpoint of the integer intervals.

We rapidly summarize the key features of f. The formula for generating f for general L is:

If $\sigma = k - \tfrac{1}{2} + j2^{-L}$, $\quad$ k and j integral, $1 \le j \le 2^L$,

then

$$f(\sigma) = k(1 - 2^{-L}) - \tfrac{1}{2} + j2^{-L}. \qquad (44')$$

The weak contraction condition (*i*) is trivially satisfied as well as the strong contraction condition (*ii'*), (24'), with the same value of δ that was previously obtained:

$$\delta = 2\gamma/(1 + \gamma) < 1. \qquad (47')$$

Finally,

$$|f(\sigma) - \gamma\sigma| \le 2^{-L-1}, \qquad (48')$$

and hence condition (*iii*) is also satisfied. It is noteworthy that in keeping with the familiar properties of rounding and truncating, the above error bound is generally smaller than the corresponding bound in (48) for the truncating external arithmetic.

The arguments used previously for establishing uniqueness apply as well for the above construction.

APPENDIX B

On the Bias Function

We give here the derivations of the results on the bias function that are stated in Section 3.1, accompanied by more detailed insights and interpretations. It is convenient to drop the adjunct σ in $B(\cdot|\sigma)$, the bias function, with the understanding that here σ is arbitrary, but fixed.

B.1 Generating the bias function

We derive (33), which is a functional recursion yielding the bias function,

$$B(d) = E[d(i)|d(0) = d] - d. \tag{49}$$

Define the ancillary functions

$$B_k(d) \triangleq E[d(I)|d(I-k) = d] - d, 0 \leq k \leq I, \tag{50}$$

so that

$$B(d) = B_I(d).$$

Observe that

$$\begin{aligned} E[d(I)|d(I-k) = d] &= \sum_s s\Pr[d(I) = s|d(I-k) = d] \\ &= \sum_t \Pr[d(I-k+1) = t|d(I-k) = d] \\ &\qquad \times E[d(I)|d(I-k+1) = t], \end{aligned} \tag{51}$$

where the Markov property has been used to obtain (51). Now t can take only N possible values. In fact, from (18), we see that if $k < I$, then $t \epsilon \{d + m_r | r = 1, \cdots, N\}$, and if $k = I$ then $t \epsilon \{\gamma d + m_r | r = 1, \cdots, N\}$. Further, the respective probabilities are easily given in terms of the functionals $b_r(y)$, $1 \leq r \leq N$, defined in (32), of the common pdf of the input signal variables. Thus,

$$\begin{aligned} b_r(d) &= \Pr[\xi_{r-1}Q^d \leq |x(\cdot)| < \xi_r Q^d] \\ &= \begin{cases} \Pr[d(i-k+1) = d + m_r | d(I-k) = d], \\ 1 \leq k \leq I-1 \\ \Pr[d(I-k+1) = \gamma d + m_r | d(I-k) = d], \\ k = I. \end{cases} \end{aligned} \tag{52}$$

Substituting in (51), we arrive at the relations

$$\begin{aligned} &E[d(I)|d(I-k) = d] \\ &= \begin{cases} \sum_{r=1}^{N} b_r(d) E[d(I)|d(I-k+1) = d + m_r], & 1 \leq k \leq I-1 \\ \sum_{r=1}^{N} b_r(d) E[d(I)|d(I-k+1) = \gamma d + m_r], & k = I. \end{cases} \end{aligned} \tag{53}$$

Substituting in the expressions in (50) for the functions $B_k(\cdot)$, we obtain the recursive formula given in the main text:

$$B_0(d) \equiv 0,$$

$$B_k(d) = \begin{cases} \sum_{r=1}^{N} b_r(d)\{B_{k-1}(d+m_r)+m_r\}, \quad 1 \le k \le I-1 & (54) \\ -(1-\gamma)d + \sum_{r=1}^{N} b_r(d)\{B_{k-1}(\gamma d + m_r) + m_r\}, \quad k = I, & (55) \end{cases}$$

and $B(d) \equiv B_I(d)$.

B.2 The range of the bias function

Note that, as $d \to -\infty$, the values of all the probabilities $b_1(d), \cdots, b_{N-1}(d)$ approach 0, while $b_N(d) \to 1$. Similarly, as $d \to \infty$, the values of all the probabilities $b_2(d), \cdots, b_N(d)$ approach 0, while $b_1(d) \to 1$. Thus, from (54) we have that

$$\text{As } d \to -\infty, B_1(d) \to m_N, \qquad \text{and as } d \to \infty, B_1(d) \to m_1. \tag{56}$$

Iterating, we obtain that

$$\text{As } d \to -\infty, B_{I-1}(d) \to (I-1)m_N, \qquad \text{as } d \to \infty, B_{I-1}(d) \to (I-1)m_1. \tag{57}$$

Finally, for the bias function we obtain from the above and (55) that

$$\begin{aligned} d \to -\infty, \qquad & B(d) \approx -(1-\gamma)d + \mathrm{Im}_1 > 0 \\ d \to \infty, \qquad & \approx -(1-\gamma)d + \mathrm{Im}_N < 0. \end{aligned} \tag{58}$$

The above is the basis for the claim that at least one zero-crossing of the bias function is guaranteed from observing the values of the function at the two limits.

B.3 The monotonicity of the bias function

We establish here sufficient conditions which imply the rather important monotonicity property of the bias function. Equations (54) and (55) provide the working definition of the bias function. Observe from (54) that for $1 \le k \le I-1$,

$$\begin{aligned} B_k'(d) &= \sum_{r=1}^{N} b_r'(d)\{B_{k-1}(d+m_r)+m_r\} + \sum_{r=1}^{N} b_r(d)B_{k-1}'(d+m_r) \\ &= -\sum_{r=1}^{N-1} F_r'(d)\{B_{k-1}(d+m_{r+1}) - B_{k-1}(d+m_r) + m_{r+1} - m_r\} + ''. \end{aligned} \tag{59}$$

We have found it convenient to introduce

$$F_r(d) \triangleq \sum_{s=1}^{r} b_s(d), \qquad 1 \le r \le N. \tag{60}$$

The reason for this is that $F_r'(d)$ is positive since

$$F_r(d) = 2\int_0^{\xi_r Q^d} p(\mu)d\mu. \tag{61}$$

At this point, it is worth noting from (59) that $B_{k-1}' < 0$ is not enough to establish that $B'_k < 0$; it is necessary in addition that B'_{k-1} be not excessively negative. This motivates the bounding of the derivative of B_{k-1} from both below and above. We therefore introduce the quantities

$$\alpha_k \le \min_y B'_k(y); \quad \max_y B'_k(y) \le \beta_k, \tag{62}$$

where it is understood that we are only interested in y having values in the finite dynamic range of the log step size. Further, let

$$\delta(d) \triangleq \sum_{r=1}^{N-1} F'_r(d)(m_{r+1} - m_r) \tag{63}$$

and

$$0 < \delta_{\min} \le \delta(d) \le \delta_{\max}. \tag{64}$$

From (59) we obtain

$$\begin{aligned} B'_k(d) &\le -\delta(d)(\alpha_{k-1} + 1) + \beta_{k-1} \\ &\le -\delta_{\min}(\alpha_{k-1} + 1) + \beta_{k-1}, \textit{ assuming } \alpha_{k-1} \le -1. \end{aligned}$$

Thus, we may take

$$\beta_k = -\delta_{\min}(\alpha_{k-1} + 1) + \beta_{k-1}, \tag{65}$$

provided $\alpha_{k-1} \le -1$. In identical fashion, we also obtain

$$\alpha_k = -\delta_{\max}(\beta_{k-1} + 1) + \alpha_{k-1}, \tag{66}$$

again assuming $\alpha_{k-1} \le -1$.

Summarizing, we have at this stage a coupled pair of recursions for the upper and lower bounds on the derivatives of the functions B_k, $1 \le k \le I - 1$, provided $\alpha_{k-1} \ge -1$, $1 \le k \le I - 1$. Finally, we also have from (55) that

$$B'(d) = B'_I(d) \le (1 - \gamma) - \delta_{\min}(\alpha_{I-1} + 1) + \gamma\beta_{I-1}. \tag{67}$$

We may now solve the linear recursions in (65) and (66) for (α_k, β_k) with the initial conditions $\alpha_0 = \beta_0 = 0$. The following solution is obtained: $1 \le k \le 1$,

$$\alpha_k = \frac{1}{2}\{(1 + \bar{\delta})^k + (1 - \bar{\delta})^k\} - \frac{1}{2}\cdot\frac{\delta_{\max}}{\delta}\cdot\{(1 + \bar{\delta})^k - (1 - \bar{\delta})^k\} - 1. \tag{68}$$

$$\beta_k = \frac{1}{2}\{(1 + \bar{\delta})^k + (1 - \bar{\delta})^k\} - \frac{1}{2}\cdot\frac{\bar{\delta}}{\delta_{\max}}\cdot\{(1 + \bar{\delta})^k - (1 - \bar{\delta})^k\} - 1. \tag{69}$$

We have denoted by $\bar{\delta}$ the geometric mean of $\delta_{\max}$ and $\delta_{\min}$, i.e.,

$$\bar{\delta} = (\delta_{\min}\delta_{\max})^{1/2}. \tag{70}$$

The reader will recall that the recursions (65) and (66) were contingent upon $\alpha_{k-1} \geq -1$. We find, upon examining the "solutions," that we can ensure its validity over the range $1 \leq k \leq I - 1$ provided $\alpha_{I-1} \geq -1$, i.e.,

$$\delta_{\max} \leq \bar{\delta} \cdot \frac{(1+\bar{\delta})^{I-1} + (1-\bar{\delta})^{I-1}}{(1+\bar{\delta})^{I-1} - (1-\bar{\delta})^{I-1}}. \tag{71}$$

The above is a key relation. The first observation on it is that the relation implies not only that $\alpha_{I-1} \geq -1$ but also that $\beta_{I-1} \leq 0$, which is of primary interest. This may be verified either directly from the expression in (69) or, more conveniently, from the recursion in (65) for β_k and the fact that $\beta_0 = 0$. But, as an examination for the bound on $B'(d)$ in (67) shows, these two conclusions, namely, $\alpha_{I-1} \geq -1$ and $\beta_{I-1} \leq 0$, are sufficient to guarantee that $B'(d) < 0$. We have thus arrived at the main result of this section:

If $\delta_{\max}$ satisfies the inequality (71), then $B'(d) < 0$. (72)

Some insight into the nature of the inequality (71) may be gained by considering the case of $\bar{\delta} \ll 1$. In this case, the rhs of (71) reduces to $1/(I - 1)$. Further, we observe from (68) and (69) that $\alpha_k \approx -k\delta_{\max}$ and $\beta_k \approx -k\delta_{\min}$. Thus, summarizing, we have that

If $\bar{\delta} \ll 1$ then $\alpha_k \approx -k\delta_{\max}$, $\beta_k \approx -k\delta_{\min}$, $1 \leq k \leq I - 1$
and (71) requires that $\delta_{\max} \leq 1/(I - 1)$. (73)

Thus, we have demonstrated that the monotonicity of the bias function is implied if the quantity $\delta(d)$ defined in (63) is uniformly small.

Let us now examine the probabilistic import of the condition in (71), namely, that

$$\delta(d) = \Sigma F_r'(d)(m_{r+1} - m_r)$$

be not large. First, recall from the definition of $F_r(d)$ in (61) that

$$F_r'(d) = 2(\ln Q)(\xi_r Q^d)p(\xi_r Q^d), \qquad 1 \leq r \leq N-1. \tag{74}$$

Thus,

$$\begin{aligned}\delta(d) &= 2(\ln Q) \sum_{r=1}^{N-1} (m_{r+1} - m_r)(\xi_r Q^d)p(\xi_r Q^d) \\ &= 2 \sum_{r=1}^{N-1} \ln (M_{r+1}/M_r)(\xi_r Q^d)p(\xi_r Q^d).\end{aligned} \tag{75}$$

Requiring that $\delta(d)$ be not too large is tantamount to requiring that the ratios of the multipliers, M_{r+1}/M_r, be not too large. To make this connection quite transparent, we see that

$$\delta(d) \leq 2 \ln (M_N/M_1) \left[\max_y yp(y) \right]. \tag{76}$$

For $p(\cdot)$ Gaussian with variance σ^2, observe that

$$\max_{y} yp(y) = p(\sigma) = 0.242, \tag{77}$$

so that, in this case, (76) states that

$$\delta(d) \leq 0.484 \ln (M_N/M_1). \tag{78}$$

The above is not a particularly good bound, relative to the expression in (75), but it does illuminate the manner in which $\delta_{\max}$ depends on the ratios of the multipliers.

Finally, in summary let us recall in purely qualitative terms the reasons for requiring that $\delta(d) = \Sigma F_r'(d)(m_{r+1} - m_r)$ be not large. This condition is tied in a natural way to the conditions that $B_k'(d) \geq -1, 1 \leq k \leq I - 1$, which is at the core of the above analysis since it follows rather easily from these conditions that $B_k'(d) \leq 0$, also. The conditions "$B_k'(y) \geq -1$" have an entirely natural, underlying probabilistic interpretation. It merely states that, for two starting log step sizes, $d(0) = d$ and $d(0) = d'$, where, say, the ordering is $d < d'$, the respective expected log step sizes after k iterations should also be ordered in the same way. A little thought is enough to convince one that such a condition can only be guaranteed by requiring that $\delta(d)$ be not too large, since $\delta(d)$ itself measures the potential for initial orderings to be reversed in one iteration.

APPENDIX C

Approximate Formula for the Central Log Step Sizes

The object here is to derive the following approximate formula for the dependence of the central log step size on the signal intensity, σ:

$$c_{\text{app}}(\sigma) = S \log_Q \sigma + D, \tag{79}$$

where S and D, given in (37) and (38), are obtained from the fixed parameters of the system. The sole approximation that is made is in approximating the distribution of the input signal variables in the following manner:

$$\int_0^y p(\mu) d\mu \approx \alpha_1 \log_Q y + \alpha_2, \tag{80}$$

where $p(\cdot)$ is the pdf of the input signal variables normalized to have unit variance.

The procedure that is followed consists of first deriving the approximation to the bias function, using the recursive formula in (33), and subsequently deriving the root of the approximate bias function. Observe that the recursive formula in (33) calls for the quantities $b_r(\cdot), 1 \leq r \leq N$. We find it essential to work with the partial sums

$$\begin{aligned} F_r(d) &= \sum_{s=1}^{r} b_s(d), \qquad 1 \le r \le N-1 \\ &= 2 \int_0^{\xi_r Q^d} p(\mu) d\mu \\ &\approx 2\alpha_1 \log_Q(\xi_r Q^d/\sigma) + 2\alpha_2, \text{ from (80),} \\ &= 2\alpha_1 d - 2\alpha_1 \log_Q \sigma + (2\alpha_2 + 2\alpha_1 \bar{\xi}_r), \end{aligned} \tag{81}$$

where σ^2 is the variance of the input signal variables. Note that $F_N(d) = 1$.

Examining (33), we find that we may also write it as follows [for notational simplicity, we drop the adjunct σ in $B_k(d|\sigma)$]:
for $1 \le k \le I-1$

$$B_k(d) = B_{k-1}(d+m_N) + m_N - \sum_{r=1}^{N-1} F_r(d)\{B_{k-1}(d+m_{r+1}) - B_{k-1}(d+m_r) + m_{r+1} - m_r\}. \tag{82}$$

Now suppose that $B_{k-1}(d)$ may be expressed in the form

$$B_{k-1}(d) = (f_{k-1} - 1)d + g_{k-1} \log_Q \sigma + h_{k-1}, \tag{83}$$

where $(f_{k-1}, g_{k-1}, h_{k-1})$ do not depend on either d or σ. Certainly, $B_0(d)$ may be expressed in this form since $B_0(d) \equiv 0$. We now show that $B_k(d)$ may also be expressed in the above manner.

Upon substituting the above expression for $B_{k-1}(d)$ and the expression in (81) for $F_r(d)$, in (82) we find that

$$B_k(d) = (f_k - 1)d + g_k \log_Q \sigma + h_k, \tag{84}$$

where

$$\begin{aligned} f_k &= \{1 - 2\alpha_1(m_N - m_1)\} f_{k-1}, \\ g_k &= g_{k-1} + 2\alpha_1(m_N - m_1) f_{k-1}, \\ h_k &= h_{k-1} + \left\{ m_N - 2 \sum_{1}^{N-1} (m_{r+1} - m_r)(\alpha_1 \bar{\xi}_r + \alpha_2) \right\} f_{k-1}. \end{aligned} \tag{85}$$

Certainly, the newly defined quantities are independent of d and $\log_Q \sigma$. Thus, the basis exists for an inductive construction. Further, the coupled recursions in (85) are trivial to solve for the initial conditions $f_0 = 1, g_0 = 0, h_0 = 0$; thus, we obtain $(f_{I-1}, g_{I-1}, h_{I-1})$.

As is apparent from (23), the final iteration in the recursion for generating the bias function differs from all the others. In fact,

$$\begin{aligned} f_I &= \{\gamma - 2\alpha_1(m_N - m_1)\} f_{I-1} \\ g_I &= g_{I-1} + 2\alpha_1(m_N - m_1) f_{I-1} \\ h_I &= h_{I-1} + \left\{ m_N - 2 \sum_{1}^{N-1} (m_{r+1} - m_r)(\alpha_1 \bar{\xi}_r + \alpha_2) \right\} f_{I-1}. \end{aligned} \tag{86}$$

The complete solution for the approximation to the bias function is:

$$B(d) = (f_I - 1)d + g_I \log_Q \sigma + h_I, \tag{87}$$

where

$$f_I = -(1-\gamma)\{1 - 2\alpha_1(m_N - m_1)\}^{I-1} + \{1 - 2\alpha_1(m_N - m_1)\}^I, \tag{88}$$

$$g_I = 1 - \{1 - 2\alpha_1(m_N - m_1)\}^I,$$

$$h_I = \frac{\left\{m_N - 2 \sum_1^{N-1} (m_{r+1} - m_r)(\alpha_1 \bar{\xi}_r + \alpha_2)\right\}\{1 - \{1 - 2\alpha_1(m_N - m_1)\}^I\}}{2\alpha_1(m_N - m_1)}$$

Recall that the central log step size is the root of the bias function $B(d)$. Thus, denoting by $c_{\text{app}}(\sigma)$ the root of the function in (87), we obtain

$$c_{\text{app}}(\sigma) = \frac{g_I}{1 - f_I} \log_Q \sigma + \frac{h_I}{1 - f_I} \tag{89}$$

$$= S \log_Q \sigma + D, \tag{90}$$

where S and D, trivially identified by comparing the two expressions, are as given in the main text, (37) and (38).

APPENDIX D

Formula for the Steady State, Mean Offset in Transmitter and Receiver Log Step Sizes

We derive the formula for $\bar{e}$ given in (42). First, it is necessary to define certain quantities in connection with (40), which describes the step-size adaptations at the two sites.

$$e(\cdot) \triangleq d(\cdot) - d'(\cdot), \quad \text{the offset at time } \cdot, \tag{91}$$

and $u(\cdot) \triangleq m(\cdot) - m'(\cdot)$, the offset in the log multipliers at time $\cdot$. From (40) we obtain

$$\left.\begin{array}{l} e(i+1) = \gamma e(i) + u(i) \\ e(i+2) = e(i+1) + u(i+1) \\ \hdashline e(i+I) = e(i-I-1) + u(i+I-1) \end{array}\right\} \qquad i = 0, I, 2I, \cdots. \tag{92}$$

Thus,

$$e(i+I) = \gamma e(i) + \{u(i) + u(i+1) + \cdots + u(i+I-1)\}. \tag{93}$$

Taking expectations of both sides of the equation,

$$\bar{e}(i+I) = \gamma \bar{e}(i) + \{\bar{u}(i) + \bar{u}(i+1) + \cdots + \bar{u}(i+I-1)\}, \tag{94}$$

where the bar has been used to denote mean values.

Consider $\bar{u}(i)$, the first term inside the parentheses. Observe that $u(\cdot) \epsilon \{m_r - m_s \mid 1 \le r,s \le N\}$. Also,

$$\bar{u}(i) = \sum_{r,s=1}^{N} (m_r - m_s) \Pr \left[\begin{array}{l} r\text{th code word transmitted } \textit{and } s\text{th} \\ \text{code word received at time } i. \end{array} \right]$$

$$= \sum_{r,s=1}^{N} (m_r - m_s) \Pr \left[s\text{th code word recd.} \middle| \begin{array}{l} r\text{th code} \\ \text{word trans.} \end{array} \right]$$

$$\times \Pr \left[\begin{array}{l} r\text{th code word trans.} \\ \text{at time } i \end{array} \right] = \sum_{r,s=1}^{N} (m_r - m_s) T_{sr} p_r(i), \quad (95)$$

where T_{sr} is simply the (s,r)th element of the channel transition matrix, and $p_r(i)$, $1 \le r \le N$, is simply obtained from the pdf of the transmitter log step size at time i.

Expressions for $\overline{u}(i+1), \cdots, \overline{u}(i+I-1)$ may similarly be derived. Thus, for $i = 0, I, 2I, \cdots$

$$\overline{u}(i) + \cdots \overline{u}(i+I-1) = \sum_{r,s=1}^{N} (m_r - m_s) T_{sr} \{p_r(i) + \cdots + p_r(i+I-1)\}. \quad (96)$$

To proceed further, it is necessary to assume ergodicity, i.e., more specifically, convergence in the mean for the time-evolving distributions of the transmitter log step size. With this assumption, as $i \to \infty$

$$\overline{e}(i) \to \overline{e} \quad (97)$$

and

$$p_r(i) + \cdots + p_r(i+I-1) \to I p_r, \qquad 1 \le r \le N, \quad (98)$$

where $\overline{e}$ and p_r have the interpretations mentioned in the main text. Substituting in (94) and (96) yields

$$\overline{e} = \frac{I}{1-\gamma} \sum_{r,s=1}^{N} (m_r - m_s) T_{sr} p_r, \quad (32)$$

which is what we set out to establish.

REFERENCES

1. D. J. Goodman and R. M. Wilkinson, "A Robust Adaptive Quantizer," IEEE Trans. Communication (Correspondence) (November 1975), pp. 1362–1365.
2. N. S. Jayant, "Adaptive Quantization with a One-Word Memory," B.S.T.J., *52*, No. 7 (September 1973), pp. 1119–1144.
3. R. M. Wilkinson, "An Adaptive Pulse Code Modulator for Speech," Proc. Int. Conf. Commun., Paper 1C (June 1971), pp. 1.11–1.15.
4. D. Mitra, "New Results From A Mathematical Study of an Adaptive Quantizer," B.S.T.J., *54*, No. 2 (February 1975), pp. 335–368.
5. D. J. Goodman and A. Gersho, "Theory of an Adaptive Quantizer," IEEE Trans. Commun., *COM-22* (August 1974), pp. 1037–1045.
6. P. Cumminskey, N. S. Jayant, and J. L. Flanagan, "Adaptive Quantization in Differential PCM Coding of Speech," B.S.T.J., *52*, No. 7 (September 1973), pp. 1105–1118.
7. S. Bates, unpublished work.
8. R. Steele, *Delta Modulation Systems*, New York: John Wiley, 1975.
9. D. Mitra, "An Almost Linear Relationship Between the Step Size Behavior and the Input Signal Intensity in Robust Adaptive Quantization," to be presented at the National Telecommunications Conference, 1978.
10. D. J. Goodman, private communication.

11. A. Croisier, D. J. Esteban, M. E. Levilion, and V. Rizo, "Digital Filter for PCM Encoded Signals," US Patent 3,777,130, December 3, 1973.
12. A. Peled and B. Liu, "A New Hardware Realization of Digital Filters," IEEE Trans. Acoust., Speech, Sig. Proc., *ASSP-22* (December 1974), pp. 456–462.
13. R. E. Crochiere, S. A. Webber, and J. L. Flanagan, "Digital Coding of Speech in Subbands," B.S.T.J., *55,* No. 8 (October 1976), pp. 1069–1085.
14. J. Max, "Quantization for Minimum Distortion," Trans. IRE, *IT-6* (March 1960), pp. 7–12.
15. L. H. Rosenthal, L. R. Rabiner, R. W. Schafer, P. Cummiskey, and J. L. Flanagan, "A Multiline Computer Voice Response System Utilizing ADPCM Coded Speech," IEEE Trans. ASSP, *ASSP-22* (October 1974), pp. 339–352.