THE BELL SYSTEM
TECHNICAL JOURNAL

DEVOTED TO THE SCIENTIFIC AND ENGINEERING
ASPECTS OF ELECTRICAL COMMUNICATION

Volume 58 March 1979 Number 3

Approximations to Stochastic Service Systems, with an Application to a Retrial Model

By A. A. FREDERICKS and G. A. REISNER

(Manuscript received February 1, 1978)

This paper illustrates the usefulness of state-dependent, birth-death processes in reducing the dimensions of stochastic service systems. The approximation techniques introduced have wide applicability to general (finite) multidimensional, state-dependent, birth-death processes. These techniques are introduced by considering the "classical" telephony problems dealing with trunk group overflow traffic from the point of view of state-dependent, birth-death processes. The main part of the paper then applies these techniques to a two-dimensional trunk group retrial model of Wilkinson and Radnik. The method, which reduces the W-R model to an approximate, easily-solved, one-dimensional model, makes use of the transition probabilities for state-dependent, birth-death processes. These are obtained via a simple extension of known results. We use the one-dimensional results to compute blocking for a range of parameter values (trunk group sizes and retrial rates) exceeding the computational limits of the W-R model. Maximum relative errors do not exceed 10 to 15 percent, while for most cases of practical interest the relative errors are less than 5 percent. The approximation also provides insight into the region of applicability of even simpler retrial models. This one-dimensional retrial model actually applies to more general (finite) state-dependent, birth-death processes (e.g., loss-delay systems).

I. INTRODUCTION

The purpose of this paper is to illustrate the usefulness of state-dependent birth and death processes in reducing the dimensions of stochastic systems. The principal application is the reduction of a two-dimensional retrial model, proposed by R. I. Wilkinson and R. C. Radnik,[1] to an approximate one-dimensional model, which is then readily solved. While algorithms[2] exist for the numerical solution of the two-dimensional Wilkinson-Radnik model, the large number of states that are often needed can result in convergence difficulties.

Section II presents the history of and motivation for the techniques used throughout the paper. The equations for the reduced one-dimensional retrial model are described in Section III. Their solution is discussed in Section IV. Section V contains numerical results and comparisons, and Section VI discusses the theoretical accuracy of our one-dimensional approximation.

The main points of this paper are the dimension reduction of stochastic models using a state-dependent birth-and-death process and a method of solution of the resulting approximate model. However, the retrial problem considered here as an example is of interest in itself and has been extensively studied in the past. L. Kosten[3] and J. Riordan[4] considered retrials coming back in a secondary, uncorrelated Poisson stream. J. W. Cohen[5] allowed for negative exponential distributions in the interarrival times of calls, the holding times of calls, the duration of the time interval between two successive attempts by a subscriber whose call was blocked at the first attempt, and the time during which a subscriber continues to make repeated attempts. All these works* attest to the difficulty of modeling and obtaining numerical solutions to the retrial problem. We hope that the ease with which one can obtain reasonable numerical results by using state-dependent, birth-death processes will motivate readers to consider this as one possible approach to the simplification of probabilistic systems.

II. HISTORY AND MOTIVATION FOR THE USE OF STATE-DEPENDENT BIRTH RATES

One early use of state-dependent birth rates to reduce a multidimensional system to a one-dimensional model is given in Ref. 6. The motivating problem was the analysis of an alternate-routed telephone network. The system to be analyzed consists of one or more primary groups of servers, each with its own arrival process. Arrivals which find all servers in its primary group busy are offered to a common overflow group (see Fig. 1). With the common assumption of Poisson traffic for the underlying arrival processes and exponential service times, one can readily write down the appropriate birth-death equations. For the case where there is only one primary group, the analysis

* Those noted are only meant to be representative of various works concerned with the retry phenomenon. They are not meant to provide a complete list of such works.

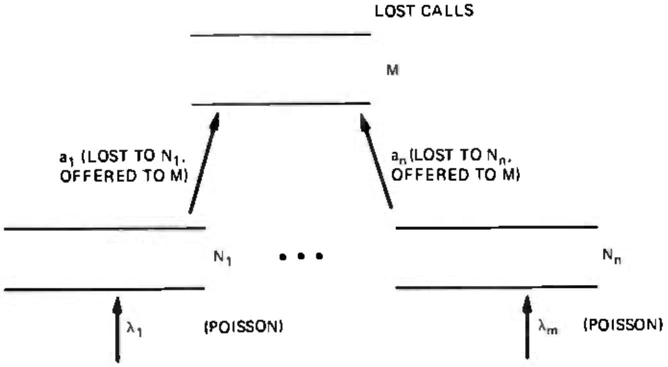

LOST CALLS
M
a_1 (LOST TO N_1, OFFERED TO M)
a_n (LOST TO N_n, OFFERED TO M)
N_1
• • •
N_n
λ_1
(POISSON)
λ_m
(POISSON)

Fig. 1—Overflow problem.

is already somewhat complicated, although it has been carried out.[7] Moreover, reasonably sized trunk groups quickly lead to systems where even numerical solutions are not feasible. However, one is often only interested in the behavior of the overflow group, e.g., finding which attempts are blocked there and hence lost from the combined system. In this case, the primary trunk groups are of interest only inasmuch as they supply the input to the overflow group. It is here that state-dependent, birth-rate modeling has proved of value.* But before noting some of the previous work on state-dependent birth rates related to this overflow problem, it is useful to consider the application of our basic ideas on dimensionality reduction.

For the simplest case of one primary group of N servers overflowing to a group of M servers, Poisson input rate λ, and (unit) exponential holding time, the birth-death equations can be written as

$$\begin{gathered}
(\lambda + i + j)P_{ij} = \lambda P_{i-1,j} + (j+1)P_{i,j+1} + (i+1)P_{i+1,j} \\
i < N \\
(\lambda + N + j)P_{Nj} = \lambda P_{N-1,j} + (j+1)P_{N,j+1} + \lambda P_{N,j-1} \\
i = N \qquad j < M \\
(N + M)P_{NM} = \lambda P_{N-1,M} + \lambda P_{N,M-1} \\
(P_{ij} = 0 \quad \text{if} \quad i \quad \text{or} \quad j < 0),
\end{gathered} \tag{1}$$

where P_{ij} is the probability that there are i busy servers in the primary group and j busy servers in the secondary group.

Since our main interest is in the marginal distribution $P_{\cdot j}$, we begin

* Actually, if total lost calls were the only item of interest, then the equivalent random method (Ref. 8) would perhaps be more applicable. However, we will be considering more general questions here.

by summing eq. (1) over i. The result after simplification is

$$\begin{gathered} jP_j + \lambda P_{Nj} = (j+1)P_{j+1} + \lambda P_{N,j-1} \qquad j < M \\ MP_M = \lambda P_{N,M-1} \qquad j = M, \\ (P_j, P_{N,j} = 0 \quad \text{if} \quad j < 0), \end{gathered} \tag{2}$$

where we have denoted the marginal distribution $P_{\cdot j}$ by P_j.

If we subtract the equation for $j = M$ from that for $j = M - 1$, and then proceed to subtract the new equation for each j from the old one for $j - 1$ we obtain the equivalent, but simpler, system

$$\lambda P_{Nj} = (j+1)P_{j+1}, \qquad 0 \le j < M. \tag{3}$$

Note that we could have obtained eq. (3) directly by balancing the upward transitions from j to $j + 1$ with the downward transitions from $j + 1$ to j, in equilibrium. In any event, by now using the fact that P_{Nj} can be written as $P_{N|j}P_j^*$ and denoting the term $\lambda P_{N|j}$ by λ_j, we obtain

$$\lambda_j P_j = (j+1)P_{j+1}\; 0 \le j < M, \tag{4}$$

an apparent one-dimensional birth-death process. Care must be taken in this interpretation. The quantity λ_j is the average "birth" rate when there are j busy on the overflow group. That is, the input process can be characterized by a state-dependent birth rate only in an average sense. Thus, while (4) is a valid equation satisfied by the equilibrium probabilities P_j, other quantities that might be obtained by viewing this as a birth-death process (e.g., transitory behavior) would at best be approximate. Indeed, even to obtain the P_j's from (4) it would be necessary to determine the λ_j exactly. Since this can usually only be done by solving the combined system (primary plus overflow trunk group), a problem we wish to avoid, we turn to approximations for the λ_j's.

Linear birth rates (e.g., $\lambda_j = a + bj$) were suggested in Ref. 6 for the case where the overflow group is infinite. This results in a negative binomial distribution for the state probabilities. Determining the parameters a and b by matching the mean and variance of the number of busy servers was found to result in a reasonably good approximation for the state probabilities. The idea of relating the birth rates to conditional probabilities was presented in Ref. 7. The resulting approximation for the state probabilities for an infinite overflow group were found to be better than those obtained with the negative binomial distribution, particularly for large values of the number busy.

Extensions of the negative binomial approximation (linear birth

* The notation $P_{N|j}$ refers to the conditional probability of N (busy on the primary group) given j (busy on the overflow group).

rates) to a system with a finite overflow group are given in Refs. 8, 9, and 10. The approach in Ref. 9 is to solve for the equilibrium probabilities $P_i^{(\infty)}$ for an infinite overflow group, and then simply terminate and normalize these to approximate the state probabilities $P_i^{(N)}$ for the finite group, i.e.,

$$P_i^{(N)} = \frac{P_i^{(\infty)}}{\sum_{i=0}^{N} P_i^{(\infty)}}. \tag{5}$$

Equation (5) still implies a linear birth rate of the form $\lambda_i = a + bi$. The problem is that if the parameters a, b are adjusted to match moments on an infinite trunk group, then for the finite case the total offered load $\Lambda = \sum_{i=0}^{N} \lambda_i P_i^{(N)}$ will no longer match the true offered load. This problem can be rectified by adjusting the offered load (see, for example, Refs. 9 and 10); however, this generally results in the need for an iteration procedure. For example, a choice of λ_i results in a set of probabilities $P_i^{(N)}$, and hence an actual offered load $\Lambda = \sum_{i=0}^{N} \lambda_i P_i^{(N)}$, which, if not the desired value, results in a need to adjust the λ_i. This feedback effect, where the equilibrium probabilities must be used to adjust the offered load, is a dominant feature in this type of approach to dimensionality reduction. We will see shortly that this interaction between the reduced state probabilities and the assumed state-dependent offered load is even stronger for the retrial model considered.

An important point to note is that, independent of the initial motivation for the above approximations, they can all be interpreted as attempting to approximate the conditional probability that the primary group is busy, given that there are j busy on the secondary group. This interpretation is important since it can often lead to insight into the applicability of the resulting approximation.

Before proceeding we note that state-dependent birth rates have also been used to study multilink systems offered overflow traffic and to obtain approximations for the blocking seen by the various parcels of traffic offered to an overflow group as depicted in Fig. 1.[11-13]

III. APPLICATION TO THE WILKINSON-RADNIK RETRIAL MODEL

A diagram of the Wilkinson-Radnik (W-R) retrial model is given in Fig. 2. The underlying offered load is assumed to be Poisson with rate λ. If these attempts find all c servers busy, they may defect from the system (probability D_1) or they may wait a period of time and then retry (probability $R_1 = 1 - D_1$). Thus, the number of people j waiting to retry is increased by one with probability R_1, whenever a first offered attempt arrives to find the number of busy servers i equal to c. When a customer retries, he either finds an idle server ($i < c$) and

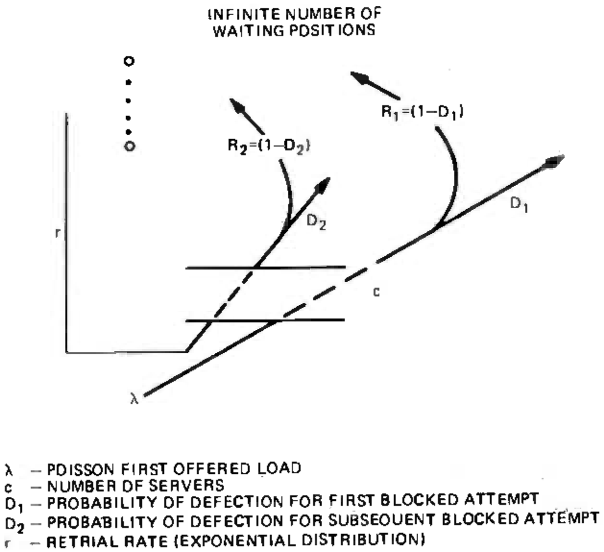

INFINITE NUMBER OF WAITING POSITIONS
R1=(1−D1)
R2=(1−D2)
D2
D1
c
r
λ
λ – POISSON FIRST OFFERED LOAD
c – NUMBER OF SERVERS
D1 – PROBABILITY OF DEFECTION FOR FIRST BLOCKED ATTEMPT
D2 – PROBABILITY OF DEFECTION FOR SUBSEQUENT BLOCKED ATTEMPT
r – RETRIAL RATE (EXPONENTIAL DISTRIBUTION)

Fig. 2—Wilkinson-Radnik retrial model.

hence is carried by the system, or, if $i = c$, he may defect with probability D_2 or again wait and retry with probability $R_2 = 1 - D_2$. With the assumption of exponential times to retry, this system is completely characterized by the state (i, j), i = number of busy servers $(i = 0, \cdots, c)$, j = number of customers waiting to retry $(j = 0, \cdots, \infty)$.

Denoting the mean time to retry by $1/r$ and the mean server holding time by $1/\mu$, we can readily write the state equations for the probabilities $P_{ij} = P^*$ (i busy server, j waiting to retry). Assuming for simplicity that $R_1 = R_2 = R$, we obtain:

$$\begin{aligned}(\lambda + rj + \mu i)P_{ij} &= \lambda P_{i-1,j} + r(j+1)P_{i-1,j+1} \\ &\quad + \mu(i+1)P_{i+1,j} \qquad i < c \\ (\lambda R + rj(1-R) + \mu c)P_{cj} &= \lambda P_{c-1,j} + r(j+1)P_{c-1,j+1} \\ &\quad + r(j+1)(1-R)P_{j+1} + \lambda R P_{c,j-1} \\ &\quad (P_{i,j} = 0 \quad \text{if} \quad i \quad \text{or} \quad j < 0).\end{aligned} \tag{6}$$

While the W-R model is a reasonable one for the customer retry phenomenon in telephone systems, eq. (6) quickly lead to numerical problems, even for modest values of c. Here, as with the overflow problem, we see that the dimensionality difficulty is caused by an

* P denotes probability.

aspect of the vector state (i, j), namely j, that we often are not concerned with, except as it influences i, the number of busy servers. Hence, following the procedure outlined in Section II, we sum (6) with respect to j and obtain

$$(\lambda + rE(j|i) + \mu i)P_i = \lambda P_{i-1} + rE(j|i-1)P_{i-1} + \mu(i+1)P_{i+1} \qquad i \neq c \quad (7)$$

and

$$(\mu c)P_c = \lambda P_{c-1} + rE(j|c-1)P_{c-1},$$

where

$$P_i = \sum_j P_{ij}$$

$$E(j|i) = jP_{ij}/P_i.$$

The term $\lambda_i' = rE(j|i)$ represents the mean input intensity associated with the retries. If this quantity were known exactly, then (7) could be solved to yield the exact solution for the equilibrium state probabilities P_i. The use of (7) to compute other quantities such as transitory probabilities would again be an approximation. However, one would expect such an approximation to be good if in the two-dimensional model, the value of j (number of retry sources) did not vary much from its mean for a given value of i (number of servers busy). This idea will be explored later.

Before discussing how to obtain an approximation for the λ_i' which clearly depends on the unknown $P(i)$, we note that direct balance of flows across the $(i, i+1)$-state boundary as before yields the simpler (but equivalent to (7)) state equations

$$\lambda_i P_i = \mu(i+1)P_{i+1}, \qquad (8)$$

where

$$\lambda_i = \lambda + \lambda_i' = \lambda + rE(j|i).$$

It is clear that there is a strong interaction between the state probabilities P_i and the retry intensity λ_i'. We now turn our attention to modeling this rather complicated relationship, and hence obtaining a solution to (8) which will hopefully approximate the two-dimensional W-R retry model adequately.

IV. SOLUTION OF THE ONE-DIMENSIONAL RETRIAL MODEL

The "solution" to a one-dimensional birth-death process is, of course, well known, provided the birth (and death) rates are *given*. Thus the main problem we are faced with is determining the λ_i's for the one-dimensional model so that they capture the essence of the two-dimen-

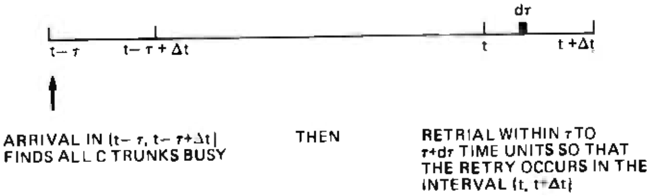


Fig. 3—Necessary events for a retrial arrival in the interval $(t, t + \Delta t)$ conditioned on an inter-retrial time equal to τ.

sional model. As indicated earlier, λ_i is the sum of λ (first offered traffic intensity) and λ_i' (retrial intensity when i trunks are busy). We thus have, to first-order in Δt, that the λ_i''s satisfy the relationship

$$\lambda_i'\Delta t = P^{(1)}\{\text{exactly one retrial in } (t, t + \Delta t] \mid N(t) = i\}\,, \qquad (9)$$

where $N(t)$ is the number of busy servers at time t and the superscript (1) indicates that this probability is for the one-dimensional model.

What we would like is for $P^{(1)}$ in (9) to be close (in some sense) to $P^{(2)}$, the corresponding probability for the two-dimensional model. Using the law of total probability, conditioned on the inter-retrial time of the arrival under consideration, we have*

$$P^{(2)}\{\text{exactly one retrial in } (t, t + \Delta t] \mid N(t) = i\}$$

$$= \int_\tau P\{N(t-\tau) = c,\, a(t-\tau, \Delta t),\, r(\tau, d\tau) \mid N(t) = i\}\,, \qquad (10)$$

where $a(t - \tau, \Delta t)$ is the event that an arrival occurs in $(t - \tau, t - \tau + \Delta t]$ and $r(\tau, d\tau)$ is the event that he will retry if blocked within $(\tau, \tau + d\tau]$ time units. Figure 3 represents these events pictorially. (Note that the assumptions made for the two-dimensional model imply that each blocked arrival can be tagged with a time to retrial, τ, taken independently from a distribution $F_r(\tau)$, at the time of his arrival.)

Using $P(A \mid B) = P(B \mid A)P(A)/P(B)$ in (10) we obtain

$$P^{(2)}\{\cdot\} = \int_\tau P\{N(t) = i \mid N(t-\tau) = c,\, a(t-\tau, \Delta t),\, r(\tau, d\tau)\}$$

$$\cdot\, P\{N(t-\tau) = c,\, a(t-\tau, \Delta t),\, r(\tau, d\tau)\}\,/P\{N(t) = i\}\,. \qquad (11)$$

Denoting the events $\{a(t - \tau, \Delta t), r(\tau, d\tau)\}$ by $\{A_r\}$ and using the law of total probability conditioned on $J(t - \tau)$ and $J(t)$, the number waiting to retry at times $t - \tau$ and t, result in

* For simplicity, we omit all terms of higher order than Δt in eqs. (10) to (16).

$$P^{(2)}\{\cdot\} = \int_{\tau} \sum_{j_1=0}^{\infty} \sum_{j_2=1}^{\infty}$$

$$\cdot\, P\{N(t) = i, J(t-\tau) = j_1, J(t) = j_2 \,|\, N(t-\tau) = c, A_r\}$$

$$\cdot\, P\{A_r\}/p\{N(t) = i\}\,. \quad (12)$$

Using $P(A, B\,|\,C) = P(A\,|\,B, C)P(B\,|\,C)$, this becomes

$$P^{(2)}\{\cdot\} = \int_{\tau} \sum_{j_1=0}^{\infty} \sum_{j_2=1}^{\infty} \cdot\, P\{N(t) = i, J(t)$$

$$= j_2 \,|\, N(t-\tau) = c, J(t-\tau) = j_1, A_r\}$$

$$\frac{\cdot\, P\{J(t-\tau) = j_1 \,|\, N(t-\tau) = c, A_r\}\, P\{A_r\}}{P\{N(t) = i\}}\,. \quad (13)$$

Now the value of $J(t-\tau)$ represents the retrial intensity at $(t-\tau)$. In general, for the two-dimensional model, $P\{J(t-\tau) = j_1\}$ does indeed depend not only on the value of $N(t-\tau)$, but on the fact that an arrival has just occurred. However, this latter dependence is inconsistent with the one-dimensional model. More specifically, we have assumed that the retrial intensity depends only on the state of the one-dimensional system, $N(t-\tau)$. Thus we are led to making the approximation

$$P\{J(t-\tau) = j_1 \,|\, N(t-\tau) = c, A_r\}$$

$$= P\{J(t-\tau) = j_1, N(t-\tau) = c\}\,.$$

Note that this approximation should tend to underestimate the retrial intensity λ'_c when there are c busy servers and hence underestimate the blocking, particularly as seen by the retrials.

Using this approximation, the formula $P(A\,|\,B, C)P(B\,|\,C) = P(AB\,|\,C)$, and noting that

$$p\{N(t) = i, J(t) = j_2 \,|\, N(t-\tau) = c, J(t-\tau) = j_1, A_r\}$$

$$= P\{N(t) = i, J(t) = j_2 - 1 \,|\, N(t-\tau) = c, J(t-\tau) = j_1\}\,,$$

we obtain

$$P^{(2)}\{\cdot\} \approx \int_{\tau} \sum_{j_1=0}^{\infty} \sum_{j_2=0}^{\infty} P\{N(t) = 1, J(t) = j_2, J(t-\tau)$$

$$= j_1 \,|\, N(t-\tau) = c\}$$

$$\cdot P\{N(t-\tau)=c\}P\{A_r\}/P\{N(t)=i\}$$

$$=\int_\tau P\{N(t)=i\,|\,N(t-\tau)=c\}\,P\{N(t-\tau)=c\}$$

$$\cdot P\{A_r\}/P\{N)t)=i\}. \tag{14}$$

Thus we require that

$$\lambda_i'\Delta t=\int_\tau P^{(1)}\{N(t)=i\,|\,N(t-\tau)$$

$$=c\}P^{(1)}\{N(t-\tau)=c\}P^{(1)}\{A_r\}/P^{(1)}\{N(t)=i\}, \tag{15}$$

where we have used the superscript (1) to emphasize that the probabilities are for the one-dimensional model. Using the Markovian properties and independence assumptions for a birth-death process, (15) can be written as

$$\lambda_i'\Delta t=\int_\tau P_{ci}(\tau)\ (\lambda_c\Delta t)P_cRdF_r(\tau)P_i, \tag{16}$$

where $P_{ci}(\tau)=P\{N(\tau)=i\,|\,N(\mathrm{o})=c\}$, i.e., the transition probabilities

λ_c = birth rate when $N=c$

R = Pr {a blocked attempt will retry}

$F_r(\tau)$ = distribution function for the time of retry

$$\left.\begin{aligned}P_c&=P\{N=c\\P_i&=P\{N=i\end{aligned}\right\}\text{ equilibrium probabilities.}$$

All are from the one-dimensional model.

Thus, finally, we have that the overall state dependent birth rates are given by

$$\lambda_i=\lambda+\lambda_i', \tag{17}$$

where from (16)

$$\lambda_i'=\frac{P_c\lambda_cR}{P_i}\int_\tau P_{c_i}(\tau)\ dF_r(\tau). \tag{18}$$

Up to this point, we have not made any assumptions regarding the retrial distribution F_r. Before doing so, it is worth pointing out that eq. (18) provides the correct answer for the two limiting values of the retrial rate r. As r goes to infinity, retrials occur "immediately." If we represent this by setting F_r equal to a unit step at zero, then (18) reduces to

$$\lambda_i'=\frac{P_c\lambda_cR}{P_i}\delta_{ci}, \tag{19}$$

where δ_{ci} is the Kronecker delta function. Combining eqs. (17) and (18) we find that $\lambda_i = \lambda$ for i less than c, and $\lambda_c = \lambda/(1 - R)$. Thus, when retrials occur immediately, they do not influence the state of the system. With one or more servers idle, the load remains at λ; with all servers occupied, an arrival retries infinitely fast and (for $R < 1$) will exit from the system before a server becomes idle.

In the other limiting case, as r goes to zero, customers "come back in an uncorrelated stream." Proceeding as above, and noting that

$$\int_0^\infty P_{ci}(\tau)\,dF_r(\tau) = P_i \qquad \text{as } r \to 0$$

(if we let F_r approach a unit jump function at infinity), we find that $\lambda_i = \lambda + P_c\lambda_c R$ for all i, and thus $\lambda_i = \lambda/(1 - P_c R)$. That is, the traffic intensity increases by a factor of $(1 - P_c R)^{-1}$, independent of the number of servers occupied. This is a familiar retrial model. In Section V we see that there is a wide range of parameter values for which this simple model is not a useful approximation for the proportion of retrials blocked.

We now return to eqs. (17) and (18), and use them to calculate recursive formulas for the λ_i's. We assume, as in the two-dimensional retrial model, that the time to retrial is given by a negative exponential distribution with rate r, namely

$$F_r(\tau) = 1 - e^{-r\tau}. \tag{20}$$

In this case, the equations become

$$\lambda_i = \lambda + \frac{P_c\lambda_c R}{P_i}\, r\hat{P}_{ci}(r) \qquad i = 1, \cdots, c, \tag{21}$$

where $\hat{}$ denotes the Laplace transform. In particular,

$$\lambda_c = \frac{\lambda}{1 - rR\hat{P}_{cc}(r)}. \tag{22}$$

If we let

$$\delta_j = \frac{\lambda_j - \lambda}{\lambda_{j-1} - \lambda} \qquad j = 1, \cdots, c, \tag{23}$$

then we can write

$$\lambda_{j-1} = \lambda + \frac{\lambda_j - \lambda}{\delta_j} \qquad j = 1, \cdots, c. \tag{24}$$

Substituting (17) into (19) and simplifying using $\lambda_{j-1}P_{j-1} = \mu_j P_j$, we obtain

$$\delta_j = \frac{\mu_j \hat{P}_{cj}(r)}{\lambda_{j-1}\hat{P}_{cj-1}(r)} \qquad j = 1, \cdots, c. \tag{25}$$

Using formula for $\hat{P}_{cj}(r)$ developed in the appendix,[*] we obtain a recursion for δ_j:

$$\delta_1 = 1 + r/\lambda_0 \quad \text{and} \quad \delta_j = \frac{(r + \lambda_{j-1} + \mu_{j-1})\delta_{j-1} - \mu_{j-1}}{\lambda_{j-1}\delta_{j-1}} \quad \text{for} \quad j = 2, \cdots, c. \tag{26}$$

Combining (26) with (22), (24), and the formula for $\hat{P}_{cc}(r)$,

$$\hat{P}_{cc}(r) = \frac{\delta_c}{(r + \mu_c)\delta_c - \mu_c}, \tag{27}$$

we get an iteration scheme for finding the $\lambda_i i = 0, \cdots, c$ which satisfies the birth-death equations and is consistent with the derivation of the $S_{jj} = 0, \cdots$, c.

Before leaving this section, we note that the above derivation holds equally well if the underlying system (without retries) is characterized by an arbitrary (finite) state-dependent, birth-death process. For example, it applies to the loss delay system considered in Ref. 4, and to overflow traffic characterized via state-dependent birth rates (as discussed in Section II). Moreover, using the results given in the appendix, one can compute transition probabilities and other related quantities for this system (e.g., correlation function).

V. NUMERICAL RESULTS

We assess the accuracy of the one-dimensional approximation by comparing our results to those obtained from direct numerical solution of the two-dimensional W-R model. The particular comparisons presented here were chosen to give the reader an understanding of the value of the one-dimensional approximation for a wide range of parameter values. However, due to the difficulty of computing "correct" values (i.e., from the two-dimensional model), the results may not cover every possible region of interest. In particular, it is difficult to analyze the convergence behavior of the two-dimensional algorithm for large trunk groups or small retrial rates. This difficulty arises because the number of waiting positions must increase to obtain a good approximation to an infinite waiting room; in turn, the number of states increases markedly and roundoff errors may become significant. Fortunately (see Section VI), the two models give the same results as r tends to infinity or zero. The approximation is worst for values of r around 2 and gets progressively better as r gets larger or smaller.

[*] The appendix shows that the $\hat{P}_{ij}(t)$ of a general one-dimensional birth-death process may be obtained in precisely the same way they were obtained for a combined delay and loss system in Ref. 4. This fact was recognized and used in Ref. 14 (Appendix B).

We first look at retrial blocking, i.e., the proportion of reattempts blocked. Figure 4 shows the retrial blocking for a system with two servers, as a function of offered load. We assume a probability of $R = 0.8$ that a blocked customer will retry. As expected, the proportion of reattempts blocked increases as r, the retrial rate, increases. For the two cases shown, r equal to 2.0 and 0.5, the relative difference between the two models is approximately 10 to 15 percent (as noted earlier, this is the worst case). We also see that the retrial blockings for the two retrial rates approach one another as offered load increases.

For comparison, we show similar retrial blocking curves for systems with 5 and 30 servers, in Figs. 5 and 6, respectively. An interesting phenomenon can be seen here, but first observe that the offered loads in each of Figs. 4 through 6 correspond to values of call congestion ranging from 0.01 to 0.30, without considering retrials. For a given design load, without retrials, the percentage of reattempts blocked is much higher for a smaller number of servers. In particular, at 1-percent total blocking and $r = 2.0$, the retrial blocking is 51 percent for 2 servers and 17 percent for 30 servers. This poor retrial performance of small server groups may be of importance in understanding customer satisfaction (or annoyance). We make one last point regarding retrial

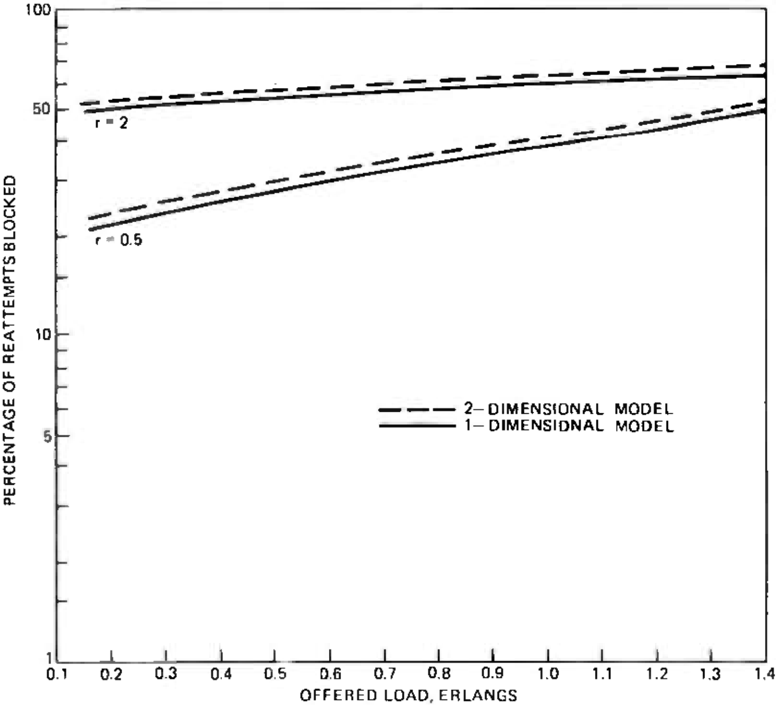


Fig. 4—Retrial blocking, 2 servers, retrial probability = 0.8.

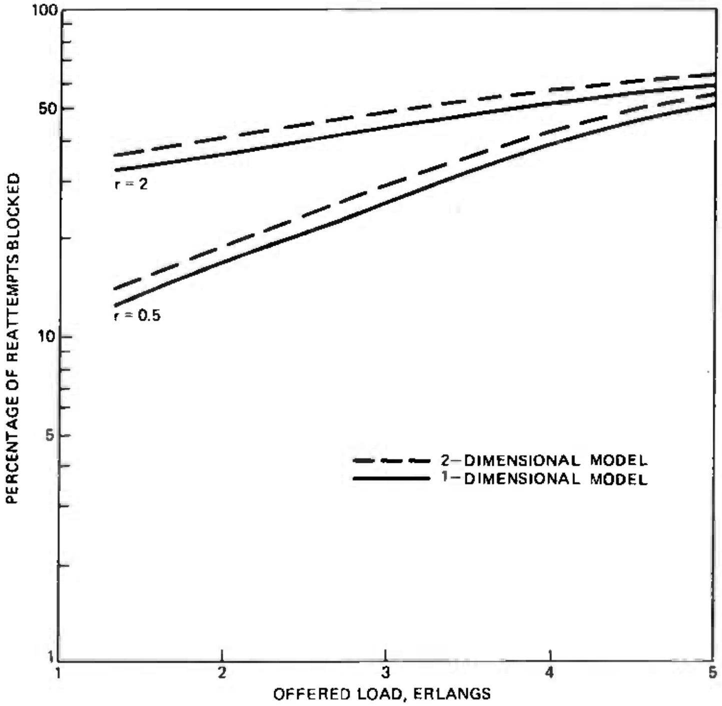


Fig. 5—Retrial blocking, 5 servers, retrial probability = 0.8.

blocking by referring the reader back to Fig. 6. The curve for $r = 0$ corresponds to the simply-computed "uncorrelated retrial" model. We see that, for a high-design blocking, both the one- and two-dimensional models are well approximated by this simpler model.

Figure 7 illustrates the dependence of retrial blocking and call congestion on trunk group size. The proportion of retrials blocked generally decreases as the group size N increases and seems to approach an asymptote. The total call congestion also decreases for small to medium size groups, but then increases for N larger than 30 and approaches the same asymptote. Notice that, if we restricted our attention to the region where the two-dimensional model applies, we would not get a good view of the limiting behavior. The reason for the seemingly anomalous behavior of total call congestion is that the larger and exceedingly efficient trunk groups are correspondingly more sensitive to traffic above the design load. We now briefly discuss the limiting behavior.

If we assume that retrials return in an uncorrelated Poisson stream ($r = 0$ case), then the inflated load Λ is given by the solution (determined by iteration, e.g.,) to

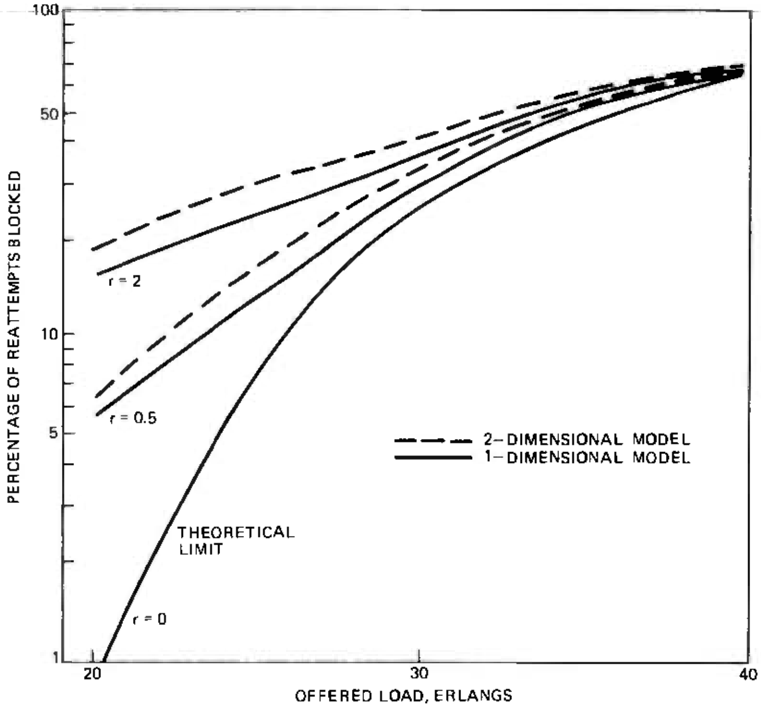


Fig. 6—Retrial blocking, 30 servers, retrial probability = 0.8.

$$\Lambda = \frac{\lambda}{1 - B(N, \Lambda)\cdot R}, \tag{28}$$

where $B(N, \Lambda)$ is the Erlang-B blocking function. The resulting blocking (for both retrials and first attempts) is the lowest curve in Fig. 7. For large values of N, we use the well-known approximation $1 - N/\lambda$ for $B(N, \lambda)$ (see Ref. 15) in conjunction with (28) to obtain the asymptote,

$$\text{call congestion} = \frac{B}{1 - R(1 - B)}, \tag{29}$$

for a fixed design blocking B. In Fig. 7, the asymptote 0.048 is shown by a dashed line.

Another way of obtaining (29) is to assume that, for large trunk groups, $\lambda - N$ equals the traffic lost on first attempts, $(\lambda - N)R$ equals the traffic lost on second attempts, $\cdots$, $(\lambda - N)R^{k-1}$ equals the traffic lost on the kth attempts, etc. Then the total blocking is given by

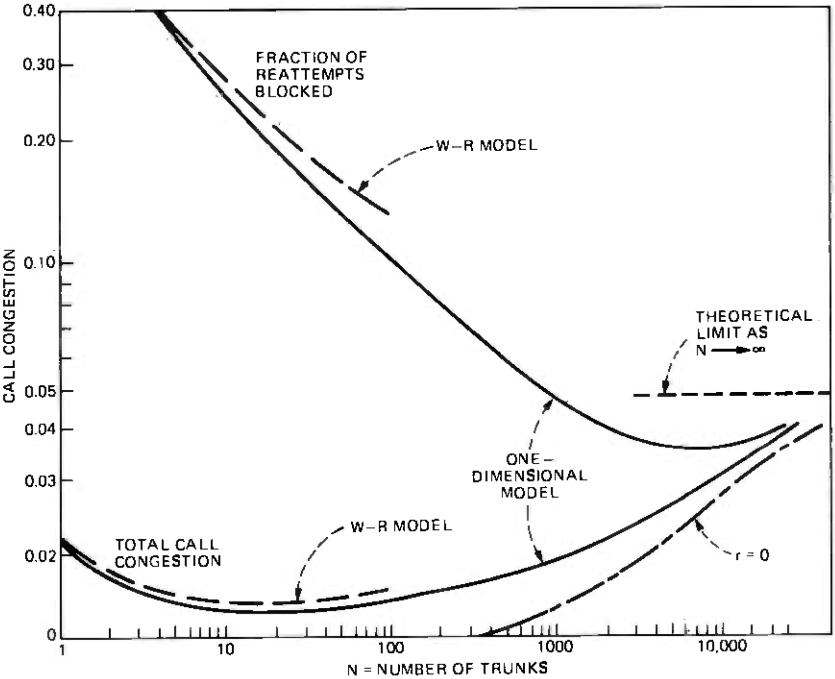


Fig. 7—Call congestion with retrials for trunk groups engineered at B.01.

$$\frac{\text{lost}}{\text{offered}} = \frac{(\lambda - N) + (\lambda - N)\left(\dfrac{R}{1-R}\right)}{\lambda + (\lambda - N)\left(\dfrac{R}{1-R}\right)}$$

$$= \frac{1 - N/\lambda}{1 - R(N/\lambda)} = \frac{B}{1 - R(1-B)}.$$

We close this section by pointing out that the one- and two-dimensional models yield practically the same values for the call congestion and the time congestion. A few sample values are shown in Table I.

VI. THEORETICAL ACCURACY—A NUMERICAL LOOK

As indicated in the derivation in Section III, the one-dimensional state equations may be obtained by summing the two-dimensional state equations over the number of waiting positions occupied. When this is done, the arrival rate of retries (when i servers are busy) equals the retrial rate times the expected number of waiting positions occupied (given i servers busy). In equation form,

$$\lambda_i' = r \cdot E(j \mid i). \tag{30}$$

If one knew the conditional expectations $E(j \mid i)$ precisely, then all the λ_i's, and hence all the P_i's of the one-dimensional equations, would be known exactly. The approximation occurs in the iteration procedure for finding the λ_i's where we implicitly assume that the one-dimensional equations actually come from a one-dimensional Markov model that adequately describes the retrial situation.

Intuitively, if either (i) the standard deviation-to-mean ratio $\sigma_{j \mid i}/E(j \mid i)$ is very small, or if (ii) $E(j \mid i) \pm k\sigma_{j \mid i}$ is "equivalent" to $E(j \mid i)$ (for some reasonable k), then for practical purposes we have a proper one-dimensional system; since knowing the number of busy servers i implies that we know j {namely, $E(j \mid i)$}. Hence, if either of conditions (i) or (ii) above hold, then we expect the one- and two-dimensional models to yield similar results. We look at a numerical example to illustrate the point. Table II shows the effect of varying the retrial rate, for five servers offered 2.22 Erlangs, while keeping all other parameters fixed. As the retrial rate goes to zero, the ratio $\sigma_{j \mid 5}/E(j \mid 5)$

Table I—Time and call congestion (r = 2.0, R = 0.8)

		One-Dimensional Model		Two-Dimensional Model	
Offered Load	No. of Servers	Time Cong	Call Cong	Time Cong	Call Cong
0.381	2	0.055	0.089	0.055	0.092
2.22	5	0.062	0.084	0.063	0.088
10.6	15	0.072	0.085	0.073	0.087
24.8	30	0.081	0.092	0.084	0.100
0.595	2	0.116	0.179	0.116	0.186
2.88	5	0.136	0.176	0.136	0.188
12.5	15	0.169	0.195	0.167	0.204
28.1	30	0.194	0.212	0.197	0.227

Table II—Effect of retrial rate (c = 5 servers, offered load = 2.22 Erlangs)

Retrial Rate	$E(j \mid 5)$	λ'_5/r	$\sigma_j \mid 5$	$\sigma_j \mid 5/E(j \mid 5)$
128.0	0.058	0.058 (0%)*	0.24	4.14
64.0	0.102	0.101 (1%)	0.32	3.14
32.0	0.166	0.161 (3%)	0.42	2.53
16.0	0.250	0.234 (6%)	0.52	2.08
8.0	0.349	0.316 (9%)	0.64	1.83
4.0	0.460	0.403 (12%)	0.76	1.65
2.0	0.577	0.499 (14%)	0.88	1.53
1.0	0.702	0.609 (13%)	1.00	1.42
0.5	0.856	0.758 (11%)	1.14	1.33
0.25	1.095	0.997 (9%)	1.31	1.20
0.125	1.527	1.440 (6%)	1.55	1.02
0.0625	2.398	2.300 (4%)	1.99	0.83
0.03125	4.107	4.010 (2%)	2.62	0.64
0.015625	7.515	7.422 (1%)	3.55	0.47

* The number in parentheses is the relative difference between $E(j|5)$ and our one-dimensional approximation to it, λ'_5/r.

tends to zero [as do $\sigma_{j\,|\,i}/E(j\,|\,i)$, $i = 0, 1, \cdots, 4$—not shown in Table II]. At the same time, our approximation λ_5'/r converges to $E(j\,|\,5)$.

At the other extreme, as the retrial rate tends to infinity, Table II shows again that λ_5'/r converges to $E(j\,|\,5)$. However, the standard deviation-to-mean ratio tends to infinity, proving that this is not a sufficient condition for convergence. On the other hand, $E(j\,|\,5) + k\sigma_{j\,|\,5} \simeq E(j\,|\,5)$, for any k. Intuitively, if we ignore the state $i = c$ $(= 5)$, we again have a one-dimensional Markovian system. Indeed, as r increases retrials occur instantaneously and the probability of having anyone in the waiting room tends to zero, if any server is free.

In summary, λ_c/r converges to $E(j\,|\,c)$ as r tends to zero or infinity. In these cases, the one-dimensional model gives the exact same answer as the W-R model but is much easier to compute (especially for r small since in this case a very large waiting room is needed). Unfortunately, the approximation seems worst for values of r around 2.0. Nevertheless, Table II shows only a 14-percent relative error in the approximation for $E(j\,|\,5)$ in this case.

VII. ACKNOWLEDGMENTS

This work has benefited greatly from many helpful discussions with A. E. Eckberg, J. M. Holtzmann, and J. S. Kaufman. We would also like to thank R. H. Harris and H. D. Jacobsen for assistance in obtaining needed data from the two-dimensional W-R retrial model.

APPENDIX

Transition Probabilities of a One-Dimensional Birth-Death Process

We show that one may obtain a solution for the transition probabilities for any finite-state, one-dimensional, birth-death process. The solution is not original; it precisely follows the derivation of the transition probabilities for the simplest combined delay and loss system given by J. Riordan (Ref. 4, pp. 96–98).

Assume we have arbitrary state-dependent birth and death rates, λ_i and μ_i, respectively. Further assume that $\lambda_i = 0$ for $i \geq c + 1$ and define $\mu_0 = \lambda_{-1} = \mu_{c+1} = 0$. Then one obtains the usual system of ordinary differential equations

$$P_{ij}'(t) = \lambda_{j-1}P_{ij-1}(t) - (\lambda_j + \mu_j)P_{ij}(t) + \mu_{j+1}P_{ij+1}(t) \qquad k, j = 0, 1, \cdots, c. \quad (31)$$

Since $P_{ij}(0) = 0$ for $i \neq j$ and $P_{ii}(0) = 1$, the Laplace transform of $P_{ij}'(t)$ is given by $sP_{ij}(s)$ for $i \neq j$ and by $-1 + sP_{ij}(s)$ for $i = j$. Hence, the Laplace transform of eq. (31) is

$$\lambda_{j-1}\hat{P}_{ij-1}(s) - (s + \lambda_j + \mu_j)\hat{P}_{ij}(s) + \mu_{j+1}\hat{P}_{ij+1}(s) = \delta_{ij} \qquad i, j = 0, 1, \cdots, c. \quad (32)$$

For any $i = 0, 1, \cdots, c$ we can write

$$D\Pi_i = \delta_i, \tag{33}$$

where δ_i is a $c + 1$ dimensional vector whose $(i + 1)$st component is 1, with all other components equal to zero,

$$\Pi_i = \begin{bmatrix} P_{io} \\ P_{ij} \\ P_{ic} \end{bmatrix},$$

and

$$D = \begin{bmatrix} s + \lambda_0 & -\mu_1 & 0 & \cdot & \cdot & 0 & 0 \\ -\lambda_0 & s + \lambda_1 + \mu_1 & -\mu_2 & 0 & & & 0 \\ 0 & -\lambda_2 & s + \lambda_2 + \mu_2 & -\mu_3 & 0 & & \\ \cdot & & & & & & \\ \cdot & & & & & & 0 \\ \cdot & & & 0 & -\lambda_{c-2} & s + \lambda_{c-1} + \mu_{c-1} & -\mu_c \\ 0 & 0 & & & 0 & -\lambda_{c-1} & s + \mu_c \end{bmatrix}.$$

If we define D_i, Δ_i via

$$D_0 = 1$$

$$D_1 = s + \lambda_0$$

$$D_{i+1} = (s + \lambda_i + \mu_i)D_i - \lambda_{i-1}\mu_i D_{i-1} \qquad 1 = 2, \cdots, c$$

and

$$\Delta_0 = 1$$

$$\Delta_1 = s + \mu_c$$

$$\Delta_{i+1} = (s + \lambda_{c-i} + \mu_{c-1})\Delta_i - \lambda_{c-1}\mu_{c-i+1}\Delta_{i-1},$$

then the transforms, $\hat{P}_{ij}(s)$, of the transition probabilities are given by

$$\text{Det}(D)\hat{P}_{ij}(s) = \begin{cases} \lambda_i\lambda_{i+1} \cdots \lambda_{j-1} D_i \Delta_{c-j} & i < j \\ \mu_{j+1}\mu_{j+2} \cdots \mu_i D_j \Delta_{c-i} & i > j, \\ D_i \Delta_{c-i} & i = j \end{cases} \tag{34}$$

where the determinant of D can be written as

$$\text{Det}(D) = D_c(s + \mu_c) - \lambda_{c-1}\mu_{nc} D_{c-1}.$$

Equation (34) is the key formula used in Section IV.

We can go one step further and write down a "closed-form" solution for the $P_{ij}(t)$, namely,

$$P_{ij}(t) = \sum_r \frac{D_i(r)D_j(r)}{\ell_i M_i S_c(r)} e^{rt},$$

where

$$\ell_i = \lambda_o \cdots \lambda_{i-1}$$

$$M_j = \mu_o \cdots \mu_i$$

$$S_c(r) = \sum_{i=0}^{c} \frac{D_i^2(r)}{\ell_i M_i}$$

and the sum is over the c roots of $\det(D) = 0$. Of course, the use of this solution is limited by one's ability to find characteristic roots.

REFERENCES

1. R. I. Wilkinson and R. C. Radnik, "Customer Retrials in Toll Circuit Operation," Conference Record, 1968 IEEE International Conference on Communications, June 12–14, 1968, Philadelphia, Pa.
2. R. H. Harris and S. R. Neal, unpublished work, and H. D. Jacobsen, unpublished work.
3. L. Kosten, "Over de invloed van herhaald oproepen in de theorie der blokkeringskansen" (On the Influence of Repeated Calls in the Theory of Probabilities of Blocking), De Ingenieur, *47* (1947), pp. 1–25.
4. J. Riordan, *Stochastic Service Systems* (1962).
5. J. W. Cohen, "Basic Problems of Telephone Traffic Theory and the Influence of Repeated Calls," Philips Telecommunications Review, *18*, No. 2 (August 1957).
6. R. I. Wilkinson, "Theories for Toll Traffic Engineering in the U.S.A." B.S.T.J., *35*, No. 2 (March 1956), pp. 421–514.
7. E. Brockmeyer, "The Simple Overflow Problem in the Theory of Telephone Traffic," Teleteknik, *5*, 1954, pp. 361–374.
8. B. Wallstrom, "A Distribution Model for Telephone Traffic with Varying Call Intensity, Including Overflow Traffic," Ericsson Technics, *20*, No. 2 (1964), pp. 183–202.
9. B. Wallstrom, "Congestion Studies in Telephone Systems with Overflow Facilities," Ericsson Technics, 22, No. 3 (1966).
10. R. R. Mina, "Some Practical Applications of Teletraffic theory," Fifth International Teletraffic Conference, 1967, pp. 428–434 (appendix by R. Syski).
11. J. M. Holtzman, "Point-to-Point Blocking Probabilities for Non-Poisson Traffic," unpublished work.
12. A. A. Fredericks, "On the Determination of Individual Parcel Blocking Probabilities for Overflow Traffic via State Dependent Birth Rates," unpublished work.
13. A. A. Fredericks, "A New Approach to Parcel Blocking via State Dependent Birth Rates," unpublished work.
14. A. A. Fredericks, "Impact of Traffic Factors and Other System Parameters on TASI-D Performance," unpublished work.
15. R. Syski, *Introduction to Congestion Theory in Telephone Systems*, Edinburgh: Olner and Boyd, 1960.

THE BELL SYSTEM TECHNICAL JOURNAL
Vol. 58, No. 3, March 1979
Printed in U.S.A.

Variable Rate Coding of Speech

By J. J. DUBNOWSKI and R. E. CROCHIERE

(Manuscript received July 31, 1978)

In this paper, we examine a number of concepts and issues concerning variable-rate coding of speech. We formulate the problem as a multistate coder (i.e., a coder that can operate at several bit rates) coupled with a time buffer. We first analyze the theoretical aspects of the problem by examining it in the context of a block processing formulation. We then suggest practical methods for implementing a variable rate coder based on a dynamic buffering approach. We also allude to a multiple user configuration of variable-rate coding for TASI*-type applications. A practical example of a variable rate* ADPCM *coder is presented and applied to speech coding. It is shown that by careful design the algorithm can be made to be as robust to channel errors as that of a fixed rate* ADPCM *coder.*

I. INTRODUCTION

In the design of digital speech coders it is often assumed that the coder and channel operate at fixed bit rates. In reality, however, it is known that speech is an intermittent and nonstationary process, and that in many applications the user demand on a communication system is a variable process. In practice, these intermittent properties can be utilized to make the design of a communication system more efficient. For example, the first property, that of an intermittent source, is utilized in communication systems such as TASI (Time Assignment Speech Interpolation).[1-3] The second property, that of a variable demand on the system, is being explored by various authors for use in packet transmission systems[4,5] and results in a variable rate channel from the point of view of the user.

In both of the above systems, an important element of the system is a variable-rate coder. In its simplest form, it may amount to a trivial transmit/no transmit decision as was used in the initial TASI systems. More generally, we might characterize a variable-rate coder according to a configuration shown in Fig. 1 where both the source activity and the channel rate are assumed to be variable.

In this paper, we examine a number of concepts of variable rate coding. We formulate the problem as a multi-state coder (i.e., a coder with several transmission states) coupled with a buffer to take up the "slack" between the desired source rate and the channel rate. In Section II we investigate theoretical aspects of variable rate coding using block processing concepts and rate distortion theory. Section III covers practical aspects of implementing variable-rate coders and in Section IV we present an example of a variable-rate ADPCM coder.

II. A BLOCK PROCESSING ANALYSIS OF VARIABLE RATE CODING

2.1 Theoretical consideration

To examine the theoretical performance of a variable-rate versus a fixed-rate coder, we can consider the problem in terms of a block processing problem. Figure 2a illustrates an example of a block of N samples of a zero mean nonstationary signal $s(n)$ as a function of time n. For convenience, we assume that this signal is uncorrelated from sample to sample (as in the difference signal of a DPCM coder).

Figure 2b illustrates the "short-time" variance or power of this signal, denoted as $\sigma^2(n)$. The noise power introduced by the coder is

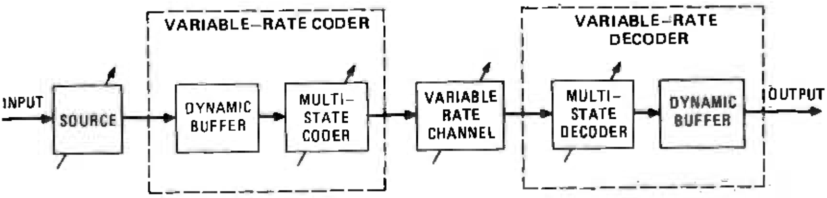


Fig. 1—A general characterization of a variable-rate coder.

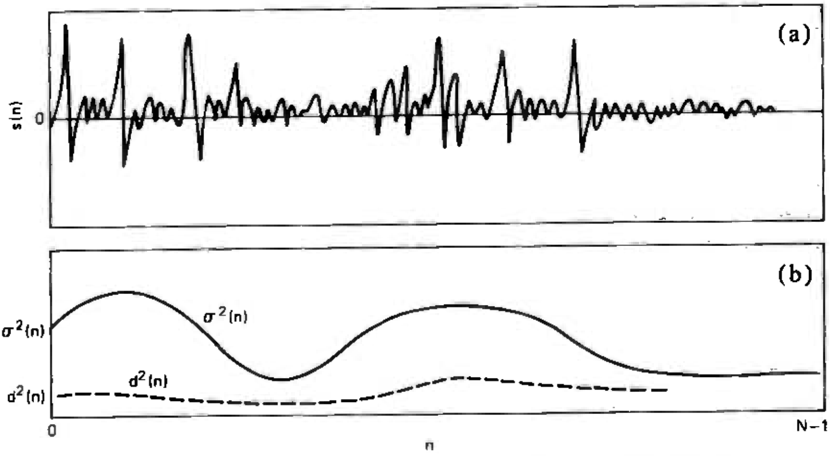


Fig. 2—Illustration of (a) a speech waveform and (b) its variance and quantization distortion after coding.

denoted as $d^2(n)$ and is also illustrated in Fig. 2b. As a performance criterion, we assume that the signal power to noise power ratio over the block, defined as

$$\text{s/n} = 10 \log \left[\frac{\sum_{n=0}^{N-1} \sigma^2 (n)}{\sum_{n=0}^{N-1} d^2(n)} \right], \tag{1}$$

is a sufficient measure for comparison. We discuss the practical merits of this measure later.

For a fixed-rate coder, the same number of bits/sample, R_f, is used for quantizing each sample $s(n)$. Therefore, the number of bits, B, used to encode the total block is

$$B = R_f N. \tag{2}$$

Also, from rate-distortion theory,[6,7] it is known that an approximate relationship between the bit rate and distortion of a quantizer is

$$R_f = \theta + \frac{1}{2} \log_2 \left(\frac{\sigma^2(n)}{d^2(n)} \right), \tag{3}$$

where $\sigma^2(n)$ is the variance of the signal as a function of time n, $d^2(n)$ is the variance of the quantization noise, and θ is a constant which is dependent on the characteristic of the quantizer and on the probability distribution of the signal. By rearranging (3), the distortion of the quantizer as a function of time can be shown to be

$$d^2(n) = \sigma^2(n) 2^{2(\theta - R_f)}. \tag{4}$$

By averaging $d^2(n)$ over N samples and applying the results to (1) and (2), the s/n of the fixed rate coder can then be shown to have the form

$$\text{s/n} \Big|_{\substack{\text{fixed}\\ \text{rate}}} = 20\,(R_f - \theta) \log_{10} 2 \tag{5a}$$

$$= 20 \left(\frac{B}{N} - \theta \right) \log_{10} 2. \tag{5b}$$

This s/n does not include the additional prediction gain that can be obtained if the input to the coder is correlated. For our purposes in this section, we assume that all correlations in the signal have been removed prior to quantization and that this s/n represents only the signal-to-noise ratio of the residual (uncorrelated) signal.

For the variable-rate coder, the number of bits/sample used to encode the nth sample is denoted as $R(n)$ (where it is assumed that $R(n)$ does not have to be an integer). The choice of $R(n)$ for $n = 0, 1, \cdots, N - 1$ is then made such that the signal-to-noise ratio in (1) is maximized and the total number of bits used to encode the block is B, i.e.,

$$B = \sum_{n=0}^{N-1} R(n). \tag{6}$$

The solution to this maximization problem is well known[7,8] and results in the condition that the distortion power $d^2(n)$ at each sample must be identical, i.e.,

$$d^2(1) = d^2(2) = \cdots d^2(N-1) = d_v^2. \tag{7}$$

Therefore, to maximize the block s/n the noise generated by the variable-rate coder must be flat across time. The number of bits/sample which must be used by the coder as a function of time is then

$$R(n) = \theta + \frac{1}{2}\log_2\left(\frac{\sigma^2(n)}{d_v^2}\right). \tag{8}$$

By applying (8) to (6), a relationship between the total number of bits in the block, B, and the distortion d_v^2 can be expressed in the form

$$\begin{aligned} B &= \sum_{n=0}^{N-1} R(n) \\ &= N\theta + \frac{1}{2}\log_2 \prod_{n=0}^{N-1} \sigma^2(n) \\ &\quad - \frac{N}{2}\log_2 d_v^2. \end{aligned} \tag{9}$$

Rearranging terms and solving for d_v^2 gives

$$d_v^2 = 2^{2(\theta - B/N)}\left[\prod_{n=0}^{N-1} \sigma^2(n)\right]^{1/N}. \tag{10}$$

The signal-to-noise ratio for the variable-rate coder can now be determined as

$$\text{s/n}\,\Big|_{\substack{\text{var.}\\ \text{rate}}} = 10\log_{10}\left(\frac{\sum_{n=0}^{N-1}\sigma^2(n)}{N d_v^2}\right) \tag{11}$$

and substituting in d_v^2 from (10) gives the revealing form

$$\begin{aligned} \text{s/n}\,\Big|_{\substack{\text{var.}\\ \text{rate}}} &= 20\left(\frac{B}{N} - \theta\right)\log_{10} 2 \\ &\quad + 10\log_{10}\left[\frac{\frac{1}{N}\sum_{n=0}^{N-1}\sigma^2(n)}{\left(\prod_{n=0}^{N-1}\sigma^2(n)\right)^{1/N}}\right]. \end{aligned} \tag{12}$$

In comparing the s/n of the variable rate coder (12) to that of the fixed rate coder (5b), it is seen that the first term in (12) is identical to that of the fixed-rate coder. The second term therefore represents the improvement in block s/n that can be expected by using a variable-rate coder instead of a fixed-rate coder. As seen by the form of this term, this improvement is signal-dependent and is in fact equal to the ratio of the arithmetic to geometric means of the signal variance $\sigma^2(n)$ over the block. If $\sigma^2(n)$ varies widely over the block, i.e., if the signal is highly nonstationary, then this gain can be large. If $\sigma^2(n)$ is relatively constant over the block, i.e., the signal is approximately stationary, then the arithmetic and geometric means are essentially equal, and no improvement can be expected.

It is interesting to note that this result is similar in form to that in transform coding.[8] In transform coding, the variation of the signal variance across the block corresponds to a variation in the frequency domain and occurs due to correlations in the input signal (i.e., a nonflatness of the signal spectrum). In the variable-rate coding application, we assumed that these correlations have already been removed and that the variation of the signal variance across the block occurs due to the nonstationarity of the signal in time. By a careful reformulation, however, both the effects of correlated inputs (i.e. prediction gain) and nonstationarity can be incorporated into the above relations for the variable rate coder.

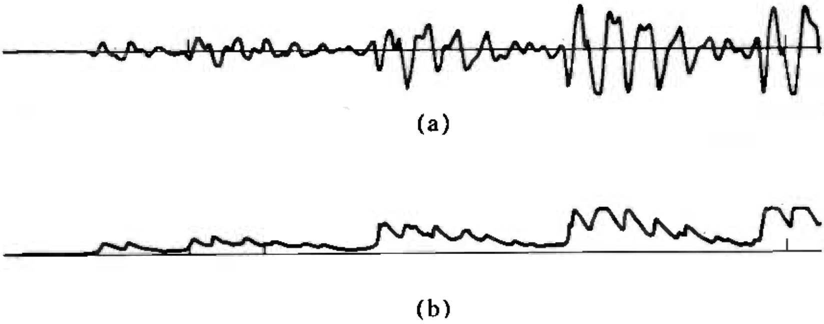
The reader should also be cautioned that while block s/n is appealing mathematically it may not be the most appropriate criterion in terms of perception.[9,10] In practice, some modification of the block s/n criterion may be required. Since little is presently understood about the perceptual effects of the distribution of s/n in time, this is a subject that requires further study before a more perceptually meaningful criterion can be proposed. Further comments on this subject are presented in Section V.

2.2 Potential improvements of variable rate over fixed rate coding

2.2.1 Single speaker

To obtain estimates of the theoretical improvement in block s/n for a variable rate coder, we have measured the arithmetic-to-geometric mean ratio of the signal variance $\sigma^2(n)$ over blocks for speech data for a (one-sided) telephone conversation. The signal variance $\sigma^2(n)$ was obtained by running a 4-bit ADPCM coder[11] on the sentence and using the step-size of the coder as a (scaled) estimate of $\sigma(n)$ of the differential input to the quantizer. Figure 3a shows an example of the speech waveform and Fig. 3b shows the corresponding scaled estimate of $\sigma(n)$ for the differential input to the quantizer in this coder.

The variance estimate $\sigma^2(n)$ of the coder was partitioned into blocks of size N and the ratio



(a)

(b)

Fig. 3—Example of (a) a speech signal and (b) the estimate of $\sigma(n)$ of its first-order predicted difference signal (8-kHz sampling rate).

$$G = 10 \log_{10} \left[\frac{\dfrac{1}{N} \sum_{n=0}^{N-1} \sigma^2(n)}{\left(\prod_{n=0}^{N-1} \sigma^2(n) \right)^{1/N}} \right] \text{(dB)} \tag{13}$$

was computed for each block to obtain an estimate of the potential s/n gain for variable rate coding.

The solid line in Fig. 4 shows a plot of the average G, denoted as $\bar{G}$, for this sentence as a function of the block size N in milliseconds. As seen in Fig. 4, significant gains in s/n cannot be expected with variable-rate coding of a single speech source until block sizes greater than about 100 ms are used. That is, the size of the block must be greater than the typical duration of phonemes and micro-silence in speech before improvements in s/n can be realized.

In real-time communications systems, blocks of this size may not be acceptable because they imply large transmission delays. Other potential applications exist, however, in voice-storage and message "store-and-forward" systems where delays may not be of concern. An alternate advantage that is offered with variable-rate coding is that it allows greater flexibility in gracefully varying the transmission rate of the coder rather than restricting it to rates which are a multiple of the sampling rate. Block sizes can be relatively small to achieve this purpose.

2.2.2 Multiple speakers (TASI)

When several sources share a channel, possibilities exist for greater improvements in overall performance due to TASI advantages. One possible approach to encoding P sources into a single channel, in a block fashion, is to assign each source a sub-block of size N/P. By concatenating the P sub-blocks into a single large block of size N, the problem can again be treated as a single source problem.

Figure 5 illustrates such an example where the variances of P concatenated sources $\sigma_1^2(n), \cdots, \sigma_P^2(n)$ are plotted as a function of time n. If the sources have greatly different variances, as depicted in Fig. 5, then the effective concatenated signal over the block will appear

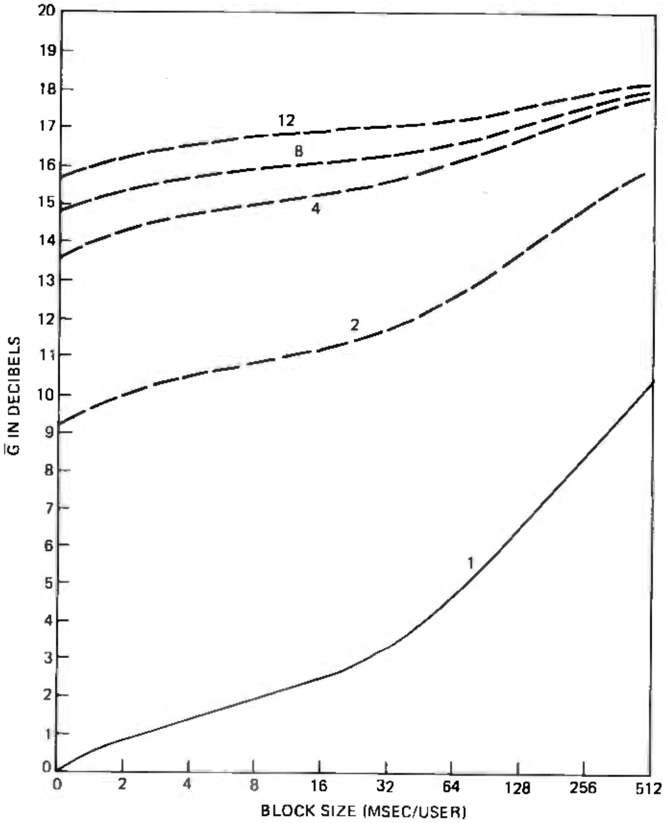


Fig. 4—Arithmetic-to-geometric mean ratio of the signal power (expressed in decibels) of a sentence as a function of the block size.

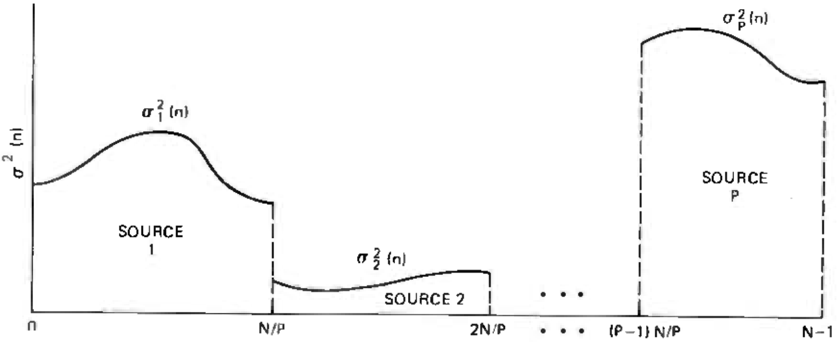


Fig. 5—Block formulation of a multi-user variable rate coder.

to be highly nonstationary and will therefore have a large arithmetic-to-geometric mean ratio. This suggests that a large s/n gain over the block can be obtained using variable-rate coding instead of a fixed bit/sample assignment. In effect, sources with larger variances will receive more bits and sources with lower variances will receive fewer bits. Each source will effectively receive the same amount of noise power as shown by eq. (7). Whether this is the most appropriate choice from a subjective point of view is again a question which remains unanswered at this time.

The dashed lines in Fig. 4 indicate measured values of $\overline{G}$ for 2, 4, 8, and 12 shared users. The gains along the left vertical axis are strictly due to TASI gains alone.

III. PRACTICAL CONSIDERATIONS IN IMPLEMENTING VARIABLE RATE CODERS

3.1 A block processing approach

In Section II, we assumed for purposes of analysis that the variable rate coder is implemented in a block processing manner with a fixed total number of bits B allowed in each block. Practical bit allocation schemes for this type of implementation have been investigated for use in transform coding[7,8] and can be carried over to the variable-rate coding application as well. Since this can be done in a relatively straightforward manner, we will not go into detail on this approach.

3.2 Dynamic buffer approach

An alternative approach to variable rate coding can be realized using a dynamic buffering strategy. A similar approach has been investigated by Tescher and Cox for use in image coding.[12] The method is illustrated in Fig. 6 for a single source example. The coder receives speech samples $s(n)$ at a fixed sampling rate and encodes them with a variable number of bits/sample. The output bits of the coder are then stored serially in a dynamic first-in, first-out buffer and the channel receives output bits from the buffer at the channel rate. A buffer control monitors the state of the buffer and a variance estimate of the input signal and regulates

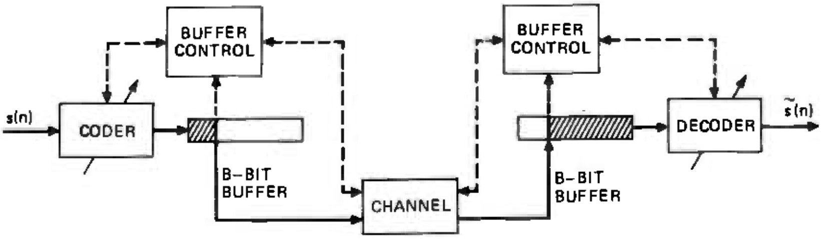


Fig. 6—Block diagram of a variable-rate coder based on buffering the output bit stream.

the number of bits/sample used by the coder. At the receiver, a similar variable rate decoding process takes place.

When the activity in the source is high, the buffer control increases the number of bits/sample used by the coder and decoder respectively above the channel rate. The transmitter buffer begins to fill up, and the receiver buffer begins to drain out. When the source activity is low, the coder and decoder use less than the average number of bits/sample and the reverse process takes place. The total signal delay across the coder/decoder is fixed at a value equivalent to the buffer size, B bits, divided by the channel rate (bits/second).

An alternative dynamic buffer strategy, based on buffering the data samples, is shown in Fig. 7. In this case, the buffer supplies samples to the coder at a variable rate. The buffer control adjusts this rate as a function of buffer status and signal variance while matching the output rate of the coder to that of the channel rate (bits/sample). When the source activity is high, the actual sampling rate transmitted through the channel lags the source rate. This causes the transmit buffer to fill and the receive buffer to empty, as in the scheme of Fig. 5. Conversely, when the source activity is low, the channel transmits samples at a rate greater than the input source rate. This results in filling the receiver buffer while depleting the transmitter buffer. The total signal delay for the system is equal to the buffer size N (samples).

3.3 Buffer control

In both the above dynamic buffering approaches, a key element in the algorithm is the buffer control. In this section, we propose a technique for implementing this control in a recursive manner which applies to either of the above methods.

The algorithm is based on the rate distortion relation (8), which can be expressed in the form

$$R(n) = \log_2 \frac{\sigma^2(n)}{d_s^2(n)}, \tag{14}$$

where $d_s^2(n)$ denotes the (scaled) distortion level in the quantizer and includes the factor θ in (8). Therefore,

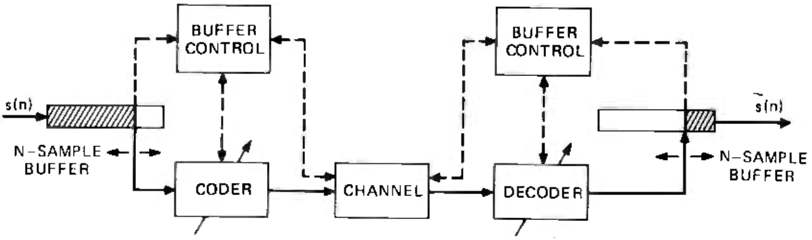


Fig. 7—Block diagram of a variable-rate coder based on buffering the input samples.

$$d_s^2(n) = d_v^2/2^{\theta}, \tag{15}$$

where $d_s^2(n)$ will be allowed to vary "slowly" with time in a manner which will be described shortly.

In practice, (14) must be modified to account for overflow and underflow of the buffers. Let B denote the size of the buffer and $b(n)$ denote the number of bits stored in the transmitter buffer at time n. Furthermore, let $R_c(n)$ denote the actual number of bits/sample used by the coder at time n and $R_d(n)$ be the number of bits removed from the buffer during the sample period at time n for transmission over the channel. If the transmitter buffer is full, then the coder cannot be permitted to use more than $R_d(n)$ bits/sample and if the buffer is empty the coder cannot be permitted to use less than $R_d(n)$ bits/sample. Therefore, the following algorithm applies:

$$R_c(n) = \begin{cases} [R(n)] \leq R_d(n) & \text{if} \quad b(n) = B \\ [R(n)] & \text{otherwise} \\ [R(n)] \geq R_d(n) & \text{if} \quad b(n) = 0, \end{cases} \tag{16}$$

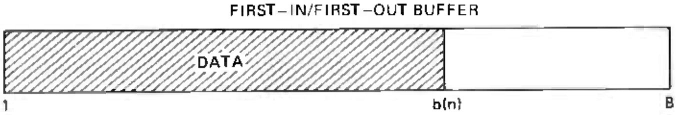
where $[R(n)]$ implies the operation of rounding $R(n)$ in (14) to the nearest integer, $\leq$ implies reducing $R_c(n)$ to be less than or equal to $R_d(n)$, and $\geq$ implies increasing $R_c(n)$ to be greater or equal to $R_d(n)$.

While the constraints in (16) prevent the buffers from overflowing or underflowing, they are not sufficient to assure that the buffers will be effectively utilized. If the average $R(n)$ is too large or too small, the buffers will remain in a state of being near full or near empty, respectively. To efficiently utilize the buffers, the average rate of $R(n)$ should be close to that of the channel rate. This is similar to eq. (6) in the block processing approach, which states that the average bit rate over the block is equal to the channel rate. This condition must be realized by adjusting the distortion level $d_s^2(n)$. If the transmitter buffer is excessively full for long periods of time, then it can be seen that $d_s^2(n)$ is too small and should be increased. Alternatively, if the transmitter buffer is empty for long periods of time, then $d_s^2(n)$ is too large and should be reduced.

The algorithm that we have investigated for controlling $d_s^2(n)$ is based on the recursive relation

$$d_s^2(n) = d_s^2(n-1) \cdot H(b(n-1)), \tag{17}$$

where $d_s^2(n)$ is the distortion level at time n, $d_s^2(n-1)$ is the distortion level at time $n-1$, and $H(b(n-1))$ is a multiplication factor which is dependent on the number of bits $b(n-1)$ in the transmitter buffer at time $n-1$. Figure 8 illustrates an example of $H(b(n-1))$ as a function of $b(n-1)$. The exact shape of $H(b(n-1))$ is not overly critical, except that it should be monotonically increasing and be less than 1 for $b(n-1)$ near zero and greater than 1 for $b(n-1)$ near B. The intercept where $H(b(n-1)) = 1$ determines the average buffer level



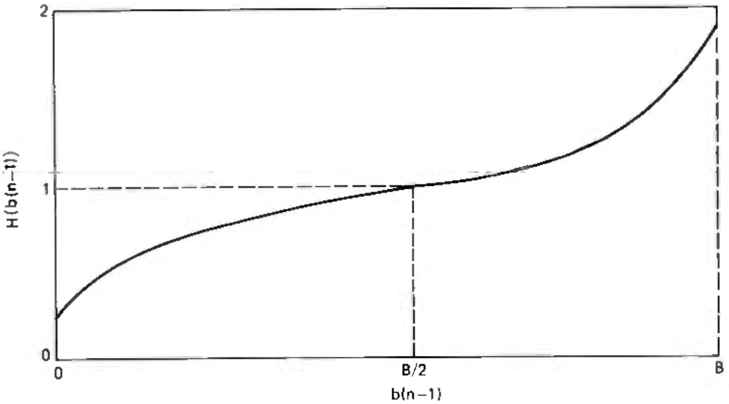


Fig. 8—Multiplier value $H(b(n-1))$ as a function of the buffer status $b(n-1)$.

about which the $b(n)$ fluctuates. If $H(b(n-1))$ is close to 1 for all $b(n-1)$, the algorithm will adapt slowly and if it becomes greatly different than 1, the algorithm will adapt rapidly. Typically, the time constant for adaption should be on the order of the total buffer delay, B. In practice, a piecewise approximation to $H(b(n-1))$ is probably sufficient. Also, it is desirable in practice to set maximum and minimum levels for $d_s^2(n)$, i.e.,

$$d_{\min}^2 \leqslant d_s^2(n) \leqslant d_{\max}^2. \tag{18}$$

This algorithm for controlling $d_s^2(n)$ is similar in many respects to the one-word memory algorithm proposed by Jayant, Flanagan, and Cummiskey[11] for adapting the step-size of an ADPCM coder.

A choice exists in generating the buffer control algorithm at both the receiver and transmitter, or at the transmitter alone. If the latter choice is made, additional information must be transmitted along with the serial data to indicate code word size. In either case, recovery from channel errors is essential. One example for accomplishing this recovery is discussed in the next section.

IV. AN EXAMPLE OF A VARIABLE-RATE ADPCM CODER

4.1 Basic design

To investigate the properties of a variable-rate coder, we have implemented a modified version of the algorithm in Fig. 7 by computer simulation. A block diagram of this implementation is shown in Fig. 9. The variable rate coder was designed around an ADPCM (adaptive differential PCM) coder[11] that can operate at 2, 3, 4, or 5 bits/sample.

The output of the ADPCM coder is framed into packets of typically 60 bits with a 2-bit header preceding each packet. The buffer control updates the number of bits/sample, $R_c(n)$, used by the ADPCM coder once per packet and transmits this decision to the receiver by means of the 2-bit header. It receives information on the buffer status $b(n)$ from the input buffer and an estimate of the signal variance $\sigma^2(n)$ from the ADPCM coder.

Each packet is encoded with either 2, 3, 4, or 5 bits/sample corresponding to 30, 20, 15, or 12 samples of data per packet respectively. A packet length of 60 bits (plus 2 header bits) is chosen because it is the smallest common multiple of 2, 3, 4, and 5 and results in a fixed packet size independent of the number of bits/sample used by the ADPCM coder.

Because the buffer control transmits the number of bits/sample, $R_c(n)$, used for encoding each packet the receiver algorithm is simplified and does not require a buffer control computation. This, coupled with the fixed packet size, allows for an overall variable rate coder that is more robust in recovering from channel errors. If a buffer control computation were to be used in the receiver, its variance information $\sigma^2(n)$ would have to be obtained from the ADPCM decoder and it would be highly susceptible to channel errors in the data. Once synchronization between the transmitter and receiver is lost it may not be able to recover because the receiver may be using incorrect bits for decoding. With the algorithm proposed here, this cannot happen. Synchronization is unaffected by errors in the data stream. If an error occurs in the header bits, a misalignment of the transmitter and receiver buffers can occur. This type of error is not disastrous, however, and is recoverable with this algorithm as will be seen later.

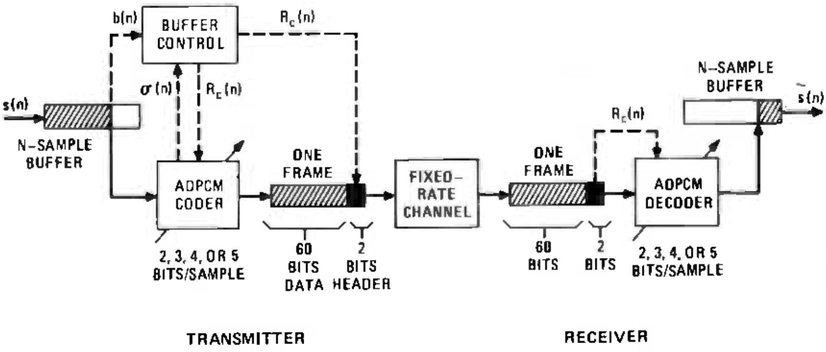


Fig. 9—Block diagram of the variable rate ADPCM coder.

4.2 The ADPCM coder

Figure 10 is a block diagram of the ADPCM coder/decoder. The signal, $e(n)$, resulting from the difference of the input $x(n)$ and its predicted value $y(n)$, is quantized using an adaptive step-size quantizer. The predicted signal, $y(n)$, is obtained from a first-order predictor, as seen in the figure. In the receiver, the difference signal $\hat{e}'(n)$ is decoded from an adaptive step-size decoder and the first-order predictor loop is used to generate the output signal $\hat{x}'(n)$.

The step-size logic adapts the quantizer step-size to track the rms level $\sigma(n)$ of the error signal $e(n)$ and is based on the one-word memory algorithm proposed by Jayant, Flanagan, and Cummisky.[11] Letting $\Delta(n)$ represent the step-size at time n and $\Delta(n-1)$ represent the step-size at time $n-1$, this algorithm is described by the relation

$$\Delta(n) = \Delta(n-1)\cdot M(|c(n-1)|). \tag{19}$$

$M(|c(n-1)|)$ is a multiplication factor that depends on the magnitude of the code word $c(n-1)$ at time $n-1$. If upper quantizer levels are used, a value of M greater than one is used and if lower quantizer levels are used, a value of M less than one is used. The M values that were used are close to the values proposed by Jayant.[11]

In the operation of the variable rate coder, the number of bits/sample used by the ADPCM coder changes and the step-size must be adjusted accordingly. This adjustment is made in such a way that the

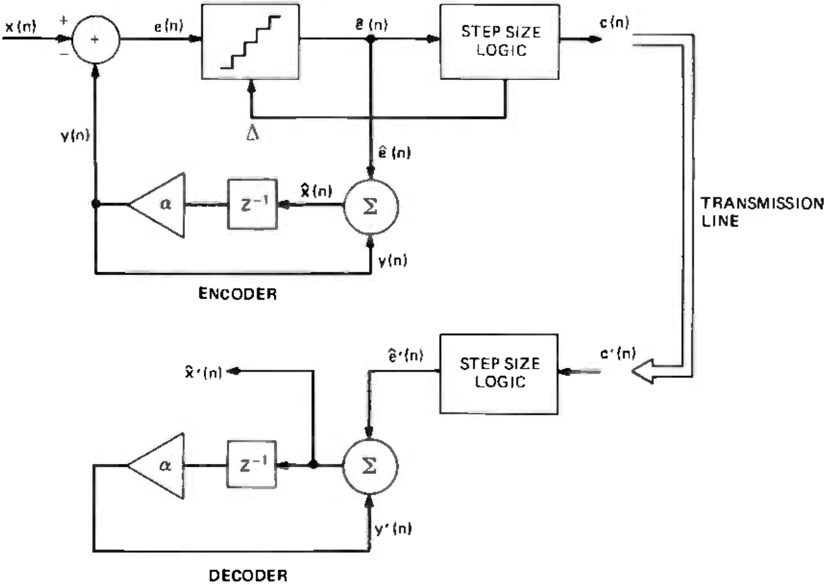


Fig. 10—Block diagram of the ADPCM coder.

center of the quantizer characteristic for the new bit rate is matched to the center for the previous bit rate. This alignment is illustrated in Fig. 11 for the 2-, 3-, 4-, and 5-bit quantizer characteristics. The horizontal scale denotes the (appropriately normalized) input signal $e(n)$ to the quantizer and the vertical scale denotes the (appropriately normalized) output signal $\hat{e}(n)$ from the quantizer (plotted only for positive values of $e(n)$ and $\hat{e}(n)$). The step-sizes Δ_2 to Δ_5 denote the relative step-sizes for the 2- to 5-bit/sample quantizer characteristics, respectively. By adjusting the step-size in this way, the loading factor and the dynamic range of the quantizer remains approximately the same when the number of bits/sample is changed—only the resolution changes.

4.3 Variance estimation

The buffer control requires an estimate of the variance of the signal $e(n)$ to compute the bit allocation $R_c(n)$. This estimate is obtained directly from the step-size adaptation algorithm in the ADPCM coder. It can be observed that, for a given loading factor and a given number of bits/sample, the square root of the signal variance, $\sigma(n)$, of the signal $e(n)$ is proportional to the step-size.

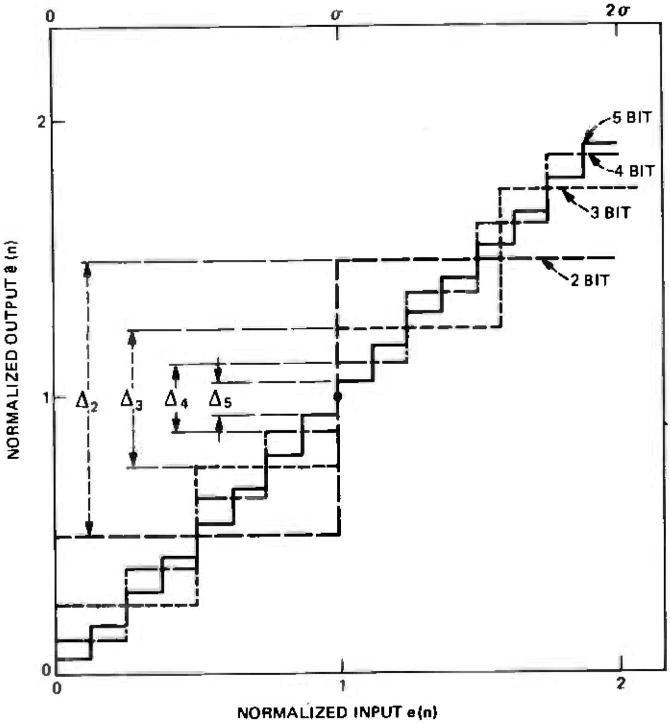


Fig. 11—Quantizer characteristic for 2- to 5-bit/sample characteristics.

The M values that were used here resulted in approximately a $\pm 2\sigma$ loading factor for each of the quantizer characteristics which is close to the optimum loading (in the mean-square error sense) proposed by Max.[13] This results in a quantizer characteristic centered about the variance of the signal as illustrated in the scale above Fig. 11. The estimate $\sigma(n)$ is therefore identified as the center of the quantizer (magnitude) characteristic which varies adaptively with the step-size adaptation.

4.4 Bit rate assignment

The buffer control determines the bits/sample assignment of the ADPCM coder, and it is based on the rate distortion relation in eq. (14). To give added flexibility to the algorithm, we also allowed a scale factor L in this equation to regulate the sensitivity of the bit allocation decision. This relation has the form

$$R(n) = L \log_2\left(\frac{\sigma^2(n)}{d_s^2(n)}\right), \tag{20}$$

where L is a parameter that can be adjusted.

4.4.1 Open loop control

To obtain an understanding of the range of values that L and $d_s^2(n)$ can take, we first ran the variable rate coder with an unlimited size input buffer and an open loop control of the bit assignment (i.e., $d_s^2(n)$ was fixed). The parameters L and $d_s^2(n)$ were adjusted as control parameters and the bit allocation was chosen on the basis of eq. (20) rounded to the nearest value 2, 3, 4, 5, or 6. The average bit rate used to encode a single sentence was then measured as a function of L and $d_s^2(n)$.

Figure 12 shows a plot of the average bits/sample, $\overline{R}(n)$, used by the coder, for this sentence, as a function of L and $d_s^2(n)$. As seen in the plot, as L increases, $\overline{R}(n)$ becomes more sensitive to variations in $d_s^2(n)$. Also, the range of $d_s^2(n)$ over which the average coder bit rate is between 2 and 6 bits/sample is clearly observed in this figure. These results were found to be useful in establishing practical limits for $d_s^2(n)$ when the adaptive buffer control is used.

4.4.2 Closed-loop dynamic buffer control

In the closed-loop buffer control, a limited size buffer was used, and a bit/sample allocation was made once per 60-bit packet, as described earlier. The bit allocation $R_c(n)$ was made on the basis of the scaled-rate distortion relation of eq. (20), and the allowed distortion $d_s^2(n)$ was "slowly" varied according to the relation in eq. (17). A two-piece approximation to the multiplier value $H(b(n-1)$ (see Fig. 8) was used in simulations according to the relation

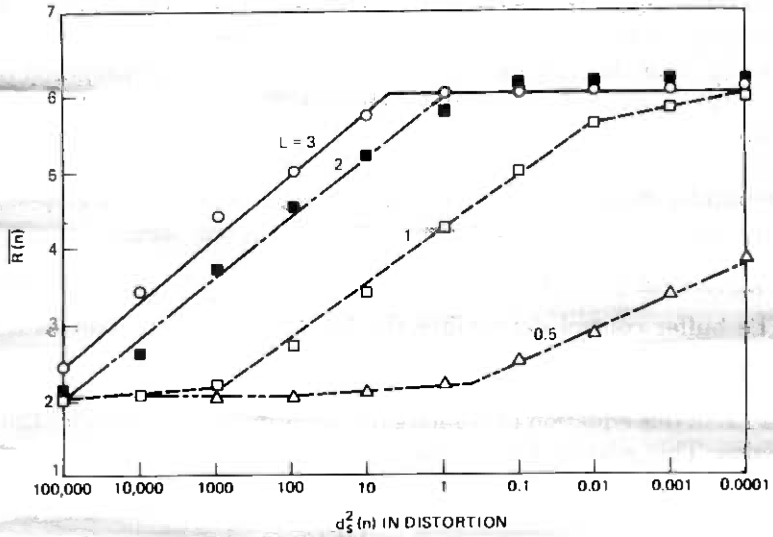


Fig. 12—Range of $\bar{R}(n)$ as a function of L and $d_s^2(n)$ for an open loop control.

$$H(b(n-1)) = \begin{cases} A > 1, & \text{if} \quad b(n-1) \geqslant N/2 \\ 1/A < 1, & \text{if} \quad b(n-1) < N/2, \end{cases} \tag{21}$$

where $b(n-1)$ is the number of samples in the input buffer in Fig. 9 and N is the size of the buffer. The value of A is greater than 1 and can be adjusted to control the speed at which $d_s^2(n)$ is allowed to vary. In general, the algorithm attempts to keep the buffer approximately one-half full (on the average).

If the buffer becomes full or empty, an additional constraint on the number of bits/sample, equivalent to that of eq. (16), is imposed to keep the buffer from overflowing or underflowing.

4.5 Performance of the variable rate ADPCM coder

The operation of the variable rate coder was observed with various parameters. In this section, we briefly illustrate the effects of some of these parameters.

Figure 13 shows a typical response of the variable rate coder for the sentence, "A lathe is a big tool." The parameters of the coder were:

N	= buffer size	=	1024
L	= rate distortion scale factor	=	1
A	= buffer adaption parameter	=	1.05
R_c	= fixed channel bit rate	=	32 kb/s
S_i	= input sampling rate	=	8 kHz

Figure 13a shows the input speech waveform, Fig. 13b shows the

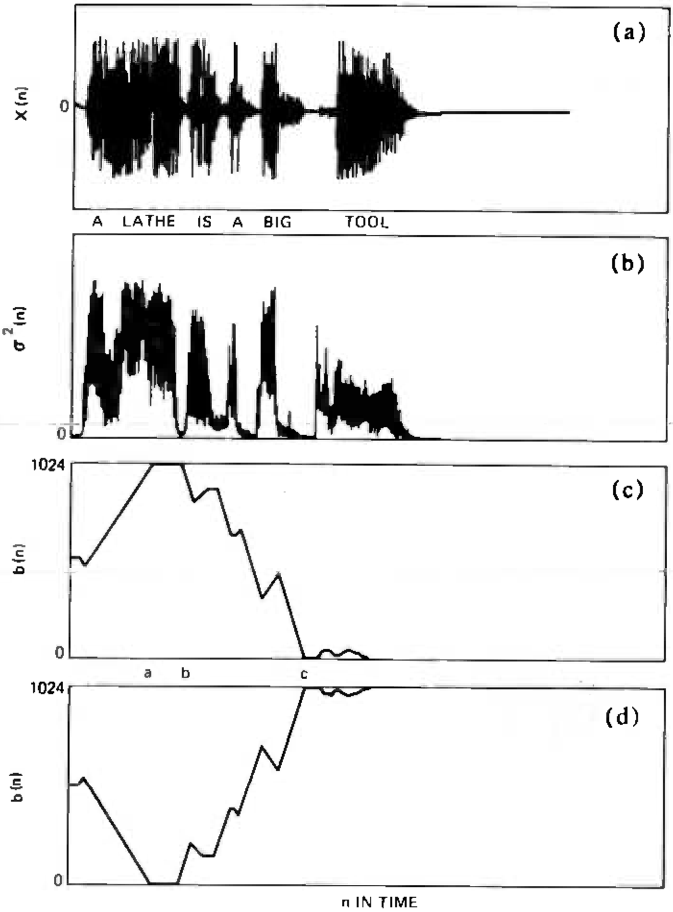


Fig. 13—(a) Input speech waveform. (b) Variance $\sigma^2(n)$. (c) Transmitter buffer status. (d) Receiver buffer status for the variable-rate ADPCM coder.

variance estimate of the difference signal $e(n)$, and Figs. 13c and 13d show the number of samples in the transmitter and receiver buffers respectively. It can be noted that the receiver buffer status is the complement of the transmitter buffer status as discussed in Section 3.2.

As seen in Fig. 13, when the signal maintains a high level of activity, the transmitter buffer fills to capacity. When it becomes full, at time a (see Fig. 13c), the bit rate of the coder is limited to that of the channel rate to prevent overflow. At time b, the speech activity drops and the buffer begins to drain out. It fluctuates with speech activity until a silent region is encountered at time c. At this point, the coder rate is again fixed to that of the channel rate to prevent underflow of the buffer.

The effects of buffer adaptation corresponding to values of $A = 0.99$, 1.0, 1.025, 1.05, 1.11, and 1.2 are shown in Fig. 14b for the same sentence with a buffer size of $N = 1024$ samples. It can be seen that, when A is less than one, the buffer control is unstable and as A becomes larger (≈ 1.2) the activity of the buffer is reduced. Figure 14c shows a similar result for a buffer size of $N = 256$ samples and it can be seen that, for smaller buffer sizes, the buffer fills up or drains out more often.

4.6 Recovery from channel errors

As pointed out earlier, the synchronization of the transmitter and receiver in the algorithm of Fig. 9 is unaffected by channel errors in the data. Resistance to these types of errors can be improved with a robust modification of the ADPCM step-size adaption algorithm.[14]

An error in the header, however, can result in an incorrect bit allocation in the receiver and the loss of a 60-bit packet of data. In addition, the receiver buffer will receive an incorrect number of samples resulting in an audible click and a misalignment of the transmitter

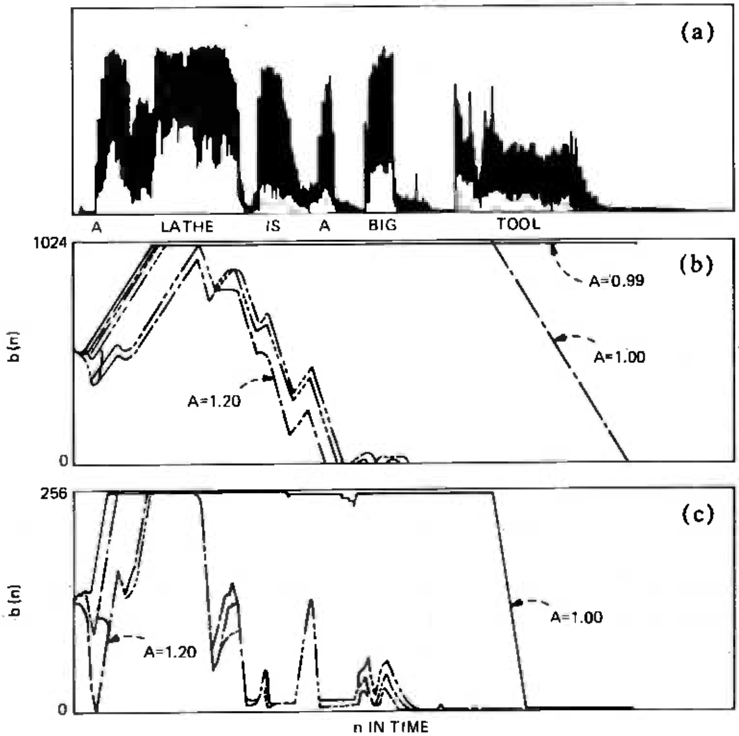


Fig. 14—(a) Variance of input speech waveform. (b) Buffer status for $A = 0.99$, 1.0, 1.025, 1.05, 1.1, and 1.2 (block size = 1024 samples). (c) Buffer status for $A = 0.99$, 1.0, 1.025, 1.05, 1.1, and 1.2 (block size = 256 samples).

and receiver buffers. This misalignment results in a temporary time shift between the transmitter and receiver (i.e., a change in total input to output delay) which is not audible to the listener.

A re-alignment of the receiver buffer will occur automatically when the buffer becomes full or empty, at which time a second signal error and time shift will occur. Two types of errors can occur, depending on whether the receiver buffer has an excess of samples or is missing samples. The first type of error is corrected when the receiver buffer becomes full and the second type of error is corrected when the receiver buffer becomes empty. In the first case, if the receiver buffer has more samples than it should and is driven into overflow (the transmitter buffer becomes empty), the excess samples can simply be discarded and the receiver and transmitter are again realigned. Since this occurs during a condition of low speech activity or silence, the loss of samples during this time is generally not audible. In the second case, when the receiver buffer is missing samples, the buffer will become empty prematurely during a period of high-speech activity (when the transmitter buffer fills up). In this case, zero-valued "dummy" samples can be inserted until realignment occurs between the transmitter and receiver. This silent period inserted during an active speech interval is not usually detectible by a listener. As a result, the realignment phases following an overflow or underflow error condition do not (in general) disrupt the audible speech.

Figure 15 is an example of a simulation of error recovery. Figure 15a shows the speech waveform and Fig. 15b shows the receiver buffer status. At time a (Fig. 15b), a header error was encountered resulting in an excess number of bits in the buffer, indicated by the shaded regions. At time b, during low speech activity, re-alignment with the transmitter buffer occurs. Following the error at a, no effects of the misalignment and realignment were audible to the listener.

V. ADDITIONAL CONSIDERATIONS AND COMMENTS ON VARIABLE RATE CODING

In this section, we examine a number of additional issues concerned with variable rate coding and comment on further directions and potential applications that need to be investigated.

5.1 Interaction of prediction gain and rate distortion criteria

In the coder example in Section IV (and the theoretical analysis in Section III), we have used the variance of the difference signal $e(n)$ (see Fig. 10) in controlling the buffer feedback. In this section, we briefly show the relationship between this variance and the signal variance of the input signal $x(n)$ for the case of a first-order predictor and demonstrate how the prediction gain interacts with the buffer

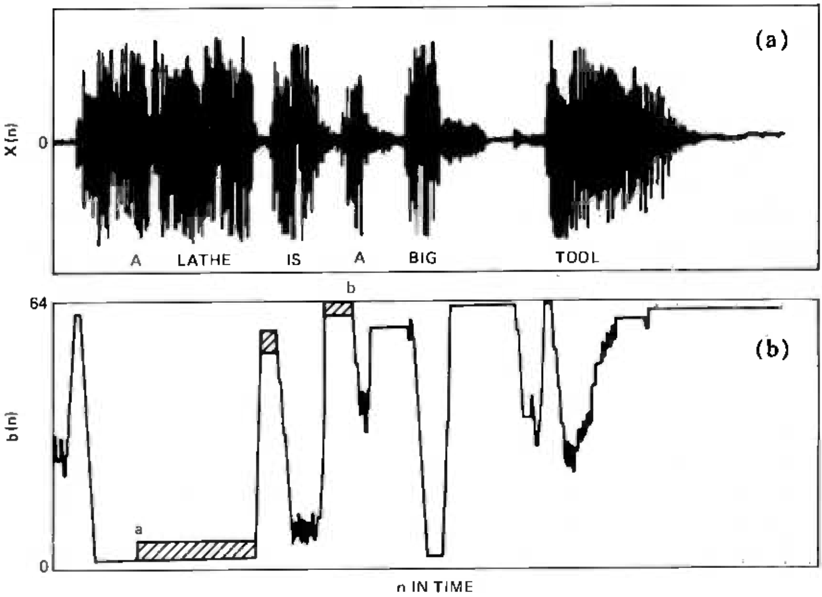


Fig. 15—Recovery of the buffer alignment after channel errors. (a) Speech waveform. (b) Receiver buffer status.

control. The interaction between the first-order predictor gain and the buffer control is also examined.

The differential signal $e(n)$ is (see Fig. 10)

$$e(n) = x(n) - \alpha\hat{x}\ (n). \tag{22}$$

After substituting the first-order correlation,

$$c = \frac{<x(n)\hat{x}(n-1)>}{<x(n)>}, \tag{23}$$

the expected value of $e^2(n)$ becomes

$$<e^2(n)> = <x^2(n)> \cdot [1 - 2\alpha c + \alpha^2]. \tag{24}$$

The result is that the difference signal variance is equal to the input signal variance multiplied by a factor dependent upon the signal correlation. Typically, for voiced speech, c corresponds to a signal correlation on the order of 0.9 (depending on the sampling rate) and a typical value of α might be about 0.9 for a fixed predictor. The result is that, for voiced speech the variance $<e^2(n)>$ is approximately proportional to the input variance, i.e.,

$$<e^2(n)>|_{\substack{\text{voiced}\\ \text{speech}}} \approx 0.2 < x^2(n)>. \tag{25}$$

During unvoiced speech, the signal correlation c is on the order of 0.1 and

$$<e^2(n)>|_{\substack{\text{unvoiced}\\ \text{speech}}} \approx 1.6 < x^2(n)>. \tag{26}$$

A comparison of the input signal variance $<x^2(n)>$ and the difference signal variance $<e^2(n)>$ is shown in Figs. 16b and 16c, respectively. Both variances appear to track relatively closely during voiced regions. During the unvoiced sounds, as for example /t/ in the word "tool," this loss of prediction increases the variance $<e^2(n)>$. Since the bit allocation is based on $<e^2(n)> \cong \sigma^2(n)$, this implies that a larger number of bits will be used to encode these unvoiced regions where the prediction gain becomes low but where the input signal variance is still significant.

5.2 Alternative criteria for buffer control based on code word magnitude

Throughout this paper, we have assumed that the buffer control is driven by the signal variance $\sigma^2(n)$ which is a result of the application of rate distortion theory. From the point of view of speech quality and perception, however, it is not clear that signal variance is the most appropriate parameter to be used for driving the bit allocation and buffer control.[9,10] Other, more perceptually meaningful parameters might be used as a driving function to produce better performance for speech.

In this section, we allude to one alternative candidate for this driving function based on a short-time average of the "code word energy." This function has been shown to be a sensitive indicator of speech and nonspeech activity.[15,16] The short-time, code-word energy is defined as

$$E(n) = \sum_{m=n-J}^{n} q_B \cdot |c(m)|, \tag{27}$$

where J is the number of samples over which the code word energy is averaged, $c(m)$ is the code word at time m (see Fig. 10), and q_B is a scale factor which normalizes the code words for different numbers of bits/sample. The code word $c(m)$ is more specifically defined as the quantizer level (see Fig. 11) expressed as an integer. The presence of small-magnitude code words are associated with silence, and the presence of large-magnitude code words are associated with speech.

Figure 16d illustrates an example of the short-time code word energy, $E(n)$, for parameters $J = 80$, and $q_2 = q_3 = q_4 = q_5 = 1$. It can be seen that $E(n)$ provides a more sensitive indication of speech activity than $<x^2(n)>$ or $<e^2(n)>$ and therefore may be a perceptually more desirable driving function to use for bit allocation and buffer control. It also provides a more reliable indication of when silence occurs in the

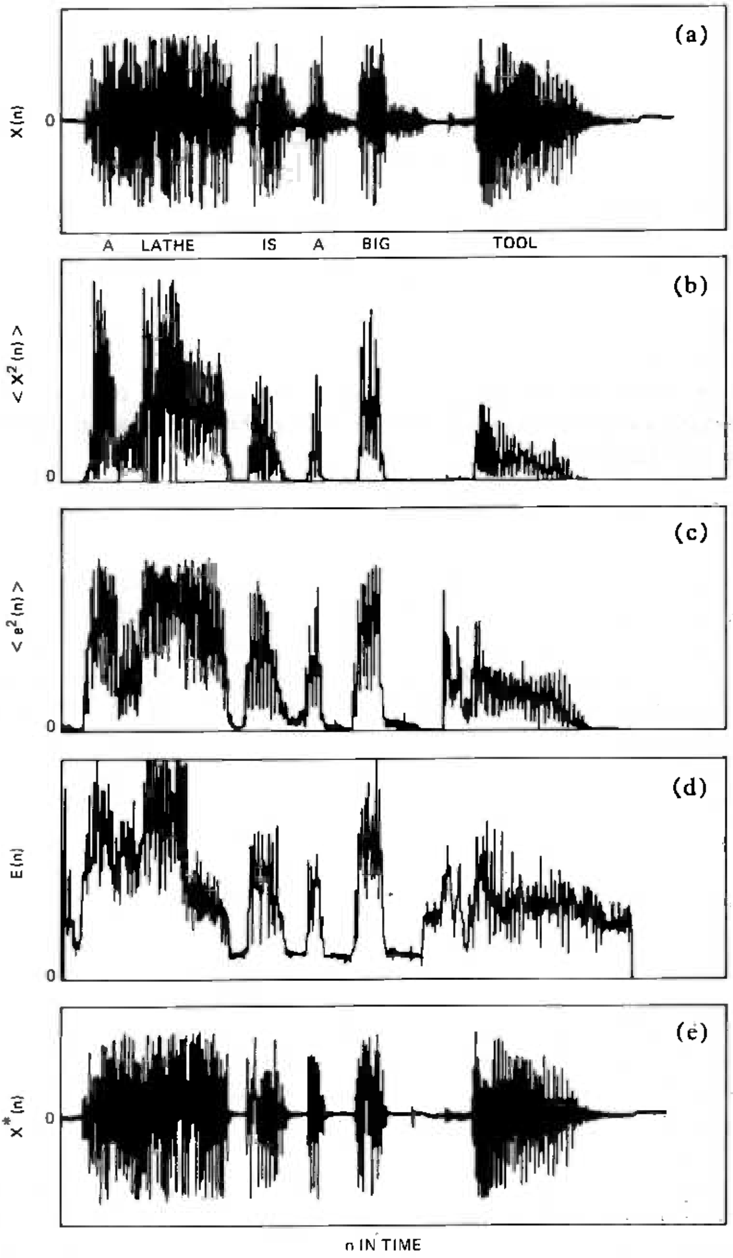


Fig. 16—(a) Input speech waveform. (b) input signal variance $<x^2(n)>$. (c) Difference signal variance $<e^2(n)>$. (d) Short-time code word energy $E(n)$. (e) Speech waveform with silence decision based on code word energy.

sentence.[15,16] For example, Fig. 16e shows the speech waveform $x(n)$ with the silence regions set to zero. It was not possible to distinguish between this sentence and the original in Fig. 16a when listening. This silence detector feature may be useful, particularly for a multiple-user application where some users can be turned off during silence.

5.3 Multiple user (TASI) applications of variable rate coding

In Section II, we illustrated a block approach for implementing variable-rate coding with multiple users and have pointed out that TASI-type advantages can be gained with this approach. A similar approach can also be implemented using dynamic buffering for each speaker. This idea is intuitively appealing in the sense that it couples TASI advantages with variable-rate coding advantages, i.e., it is a TASI with memory. By buffering the inputs of the speakers, bursts of strong activity from some speakers can be time-aligned with micro-silence regions of other speakers. Speakers whose buffers are full can receive short-time priority over other speakers whose buffers are not full. Thus, the statistics of speech activity seen by the channel is a combination of activity over time as well as across speakers. Flexible tradeoffs should be possible between the size of the input buffers (i.e., time delay) and the number of allowed users in the system.

VI. CONCLUSIONS

In summary, we can draw a number of conclusions concerning variable-rate coding:

(*i*) A block processing analysis shows that, for a single user, the improvements in block s/n of a variable-rate coder over that of a fixed-rate coder are dependent on the nonstationarity of the source and are related to the ratio of the arithmetic-to-geometric means of the signal variance.

(*ii*) For a single speech source, block sizes greater than about 100 ms are required before any substantial improvement over fixed-rate coding can be realized. Alternatively, flexibility in transmission rate is obtainable with very short block sizes with no loss in performance over fixed rate coding.

(*iii*) A multiple user variable-rate coding offers an interesting approach to implementing a TASI system.

(*iv*) Practical methods exist for designing variable-rate coders, and they can be made to be robust to channel errors.

REFERENCES

1. K. Bullington and J. M. Fraser, "Engineering Aspects of TASI," B.S.T.J., *38*, No. 2 (March 1959), pp. 353–364.
2. J. M. Fraser, D. B. Bullock, and N. G. Long, "Over-all Characteristics of a TASI System," B.S.T.J., *41*, No. 4 (July 1962), pp. 1439–1454.

3. J. P. Adoul and F. Daaboul, "Digital TASI Generalization with Voiced/Unvoiced Discrimination for Tripling T1 Carrier Capacity," Proc. IEEE Int. Conf. Commun., Chicago, June, 1977, pp. 32.5–310 to 32.6–314.
4. B. Gold, "Digital Speech Networks," Proc. IEEE, *65,* No. 12 (December 1977), pp. 1636–1658.
5. S. A. Webber, C. J. Harris, and J. L. Flanagan, "Use of Variable-Quality Coding and Time-Interval Modification in Packet Transmission of Speech," B.S.T.J., *56,* No. 8 (October 1977), pp. 1569–1573.
6. T. Berger, *Rate Distortion Theory—A Mathematical Basis for Data Compression,* Englewood Cliffs, N.J.: Prentice-Hall, 1971.
7. J. Huang and P. Schultheiss, "Block Quantization of Correlated Gaussian Random Variables," IEEE Trans. Commun. Sys., *CS-11* (September 1963), pp. 289–296.
8. R. Zelinsky and P. Noll, "Adaptive Transform Coding of Speech Signals," IEEE Trans. Acoust., Speech, and Sig. Proc., *ASSP-25* (August 1977), pp. 299–309.
9. R. E. Crochiere, L. R. Rabiner, N. S. Jayant, and J. M. Tribolet, "A Study of Objective Measures for Speech Waveform Coders," Proc. of the 1978 Zurich Seminar on Digital Communications, Zurich, Switzerland, March 1978.
10. J. M. Tribolet, P. Noll, B. J. McDermott, and R. E. Crochiere, "Complexity vs. Quality for Speech Waveform Coders," Proc. of the IEEE Int. Conf. on Acoust., Speech, and Sig. Proc., Tulsa, Okl, April 1978, pp. 586–590.
11. N. S. Jayant, "Digital Coding of Speech Waveforms: PCM, DPCM, and DM Quantizers," Proc. IEEE, *62,* (May 1974), pp. 611–632.
12. A. G. Tescher and R. V. Cox, "Image Coding: Variable Rate DPCM Through Fixed Rate Channel," Proc. Society of Photo-Optical Instrumentation Engineers, *119* (August 1977), pp. 147–154.
13. J. Max, "Quantizing for Minimum Distortion," IRE Trans. Inform. Theory, *IT-16* (May 1960), pp. 7–12.
14. D. J. Goodman and R. M. Wilkinson, "A Robust Adaptive Quantizer," IEEE Trans. Comm., November 1975, pp. 1362–1365.
15. R. W. Schafer, J. J. Dubnowski, K. Jackson, and L. R. Rabiner, "Detecting the Presence of Speech Using ADPCM Coding," IEEE Tran. Comm., May 1976, pp. 563–567.
16. L. H. Rosenthal, R. W. Schafer, and L. R. Rabiner, "An Algorithm for Locating the Beginning and End of an Utterance Using ADPCM Coded Speech," B.S.T.J., *53,* No. 6 (July-August 1974), pp. 1127–1135.

Copyright © 1979 American Telephone and Telegraph Company
THE BELL SYSTEM TECHNICAL JOURNAL
Vol. 58, No. 3, March 1979
Printed in U.S.A.

Objective and Subjective Performance of Tandem Connections of Waveform Coders with an LPC Vocoder

By D. J. GOODMAN, C. SCAGLIOLA, R. E. CROCHIERE, L. R. RABINER, and J. GOODMAN

(Manuscript received June 9, 1978)

In a recently proposed communication system, there would be tandem connections of 16-kb/s delta modulators and 2.4 kb/s vocoders. Preliminary work has indicated that such tandem links would be of substantially lower quality than either the delta modulator link or the vocoder link alone. The present study, which includes an elaborate subjective speech quality experiment, confirms this preliminary conclusion. It also shows that two other differential waveform coders are no better than the proposed delta modulator in tandem links. On the other hand, a 5-band sub-band coder does offer substantially higher quality than the delta modulator. Still, its performance in tandem with the vocoder is poorer than that of the vocoder or the sub-band coder alone and is probably of only marginal value for practical communication. We have obtained several objective measures of speech quality which, for the most part, show relatively little correlation with subjective quality. The most successful objective predictor of subjective ratings is a linear combination of linear predictive coding distances.

I. BACKGROUND AND INTRODUCTION

1.1 Background

Recent plans for United States government digital communication networks have focused attention on the compatibility of 2.4-kb/s (narrowband) vocoder systems and 16-kb/s (wideband) waveform coding schemes. An important question arises in the implementation of such a system: If both narrowband and wideband systems are designed to provide adequate speech communication individually, will a tandem connection of them also function adequately?

A recent study of this question,[1, 2] using signal-to-noise ratio (s/n) and a spectral distance measure as criteria of merit, has cast doubt on the viability of circuits containing a 2.4-kb/s LPC (linear predictive coding) vocoder and a 16-kb/s CVSD (continuously variable slope delta modulation) waveform coder. In that study, it appeared that CVSD was the weak link in these tandem connections. However, the conclusions could only be regarded as tentative because the speech material included in the study was very limited and because the relationship of the objective performance measures to the quality of communication experienced by human users was by no means evident.

1.2 Aims

In the work reported here, we extend previous results by:

(*i*) Studying three 16-kb/s waveform coders in addition to CVSD.

(*ii*) Presenting subjective as well as objective performance measures.

(*iii*) Greatly enlarging the variety of speech material processed by the various communication systems.

The specific questions addressed in our study are:

(*i*) What is the subjective quality of tandem connections of narrowband and wideband systems, relative to the quality of individual systems?

(*ii*) Are there alternatives to CVSD that offer higher quality in either (or both) individual or tandem performance?

(*iii*) What is the relationship of objective measures of system performance to subjective assessment of speech quality?

1.3 An experiment

To answer these questions we produced, in software on a Data General Eclipse computer, a 2.4-kb/s LPC vocoder and four different 16-kb/s waveform encoders. They are:

(*i*) The CVSD studied in Refs. 1 and 2.

(*ii*) A double integration version of CVSD, which we call ADM (adaptive delta modulator).

(*iii*) A two-bit 8 kHz ADPCM (adaptive differential PCM).

(*iv*) SBC (sub-band coding) with five separate channels spanning the 200 to 3200-Hz band of speech energy.

Relative to CVSD, ADM has virtually the same circuit complexity (requiring only one additional resistor and one capacitor), ADPCM is perhaps 2 to 3 times as complex, and SBC is approximately 10 to 20 times as complicated.

The five coding schemes (four waveform coders and LPC) were used in 13 different communication systems (five coders individually, four waveform coders preceding LPC, four waveform coders following LPC). These systems processed a total of 148 speech samples from four talkers (two male and two female) at three different power levels (spanning a 30-dB range).

Twenty-two subjects rated each of the processed speech samples on a 9-point scale. Each sample consisted of one sentence of 2 to 3 seconds duration, and no sentence was heard more than once by any individual subject. Although the subjects were asked to rate overall speech quality, it is felt that intelligibility had a greater influence over their ratings than it does in experiments in which a few sentences are repeated many times.

In addition, four different objective measures of system quality were calculated. These include the s/n and spectral distance measure used in Refs. 1 and 2, and also two segmental signal-to-noise ratios[3] that have been shown in other work to be more closely related to subjective quality than s/n.[4]

Statistical analysis of the subjective data reveal a complicated pattern of interactions among the experimental variables. The relative performances of the various coding schemes are dependent in a complicated way on talker and on input level as well as on (individual or tandem) system configuration. In spite of the complicated dependence of subjective quality on physical conditions, clear patterns in the data emerge to answer our original questions.

Among the individual circuits, SBC has on the average the highest subjective quality, followed by LPC, CVSD, ADM, and ADPCM, in that order. Tandem connections all are substantially degraded relative to individual circuits. Among the waveform coders, SBC provides the best tandem connections with LPC, but the SBC-LPC tandems are substantially worse than either individual system. The tandems involving the other waveform coders are probably inadequate for effective speech communication.

Among the objective measures, s/n as in other studies[3, 4] was found to be very poorly correlated with subjective quality. Moreover, with the diversity of circuit conditions and speech material presented here, the segmental signal-to-noise ratios were also of little use in predicting subjective quality. The spectral distance measure was the only one that was somewhat useful: a linear regression model based on distance measures of both tandem links and on overall distance accounted for 60 percent of the variance in the average ratings.

II. SYSTEM DESCRIPTION

2.1 Overview

In the narrowband-to-wideband tandem link shown in Fig. 1, the input speech appears as 16-bit PCM with 8-kHz sampling rate. It is first bandpass filtered to a bandwidth of 200 to 3200 Hz by means of a 6th order elliptic bandpass filter. It is then vocoded by the LPC vocoder. At the output of the vocoder, the sampling rate is converted (if necessary) by digital techniques[5] to the sampling rate of the waveform coder. This conversion has no effect on the tandem connection and is virtually "transparent" in terms of quality. The gains G and $1/G$ before and after the coder are used in measuring the dynamic range (i.e., variations in performance as a function of signal level) of the waveform coder. The output of the coder is lowpass filtered to 3200 Hz, and its sampling rate is converted back to 8 kHz and the output signal is processed by a 3200-Hz lowpass filter. In Fig. 2, the same signal processing operations are shown with the ordering that provides a wideband-to-narrowband connection.

2.2 The narrowband system (LPC)

The narrowband system consists of a linear predictive coding (LPC) system based on an all-pole model of the speech production mechanism. The all-pole model implies that, within a frame of speech, the output speech sequence is approximated by

$$s_n = \sum_{k=1}^{p} a_k s_{n-k} + G' u_n, \tag{1}$$

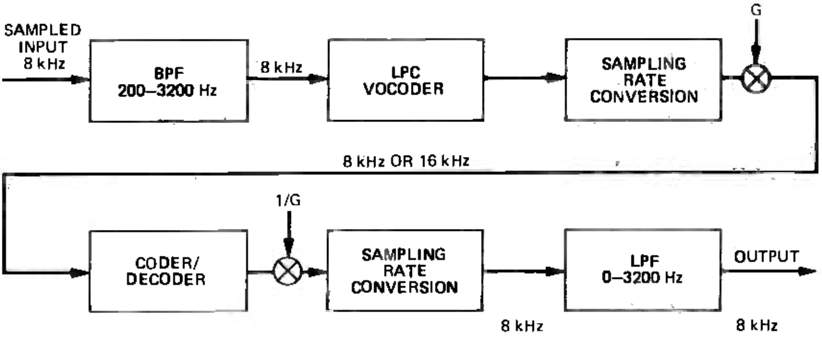


Fig. 1—Narrowband-to-wideband system.

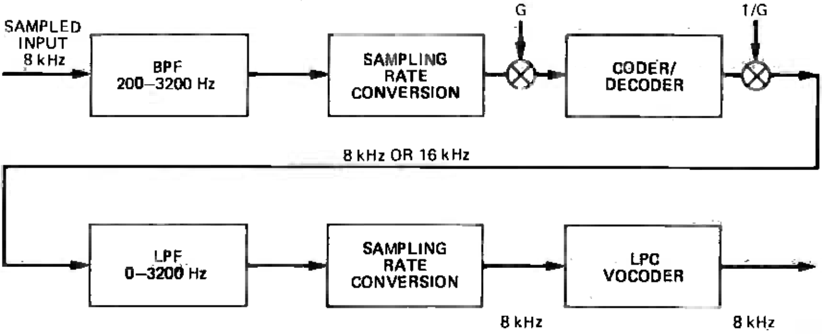


Fig. 2—Wideband-to-narrowband system.

where p is the number of poles, u_n is the appropriate input, G' is the gain, and the a_k's are the LPC coefficients that represent the spectral characteristics of the speech frame. For a voiced speech segment, u_n is a sequence of pulses separated by the pitch period. If the segment is unvoiced, pseudorandom white noise is used as input.

In our study, the LPC coefficients were calculated by the autocorrelation method with $p = 12$. The analysis was performed every 20 ms (50 times/s) with a variable analysis frame size. The frame size was proportional to a running average of the pitch period as obtained at the pitch detector output.[6] A Hamming window was used prior to the LPC analysis. Pitch detection and voiced-unvoiced (V/U) analysis were done using the modified autocorrelation method.

For quantization purposes, the LPC coefficients were converted to log area ratio coefficients, which were coded by means of ADPCM techniques.[8] An overall bit rate of 2.4 kb/s was obtained by allocating 48 bits to each of the 50 frames per second. Details of the encoding scheme are given in the references.

2.3 Delta modulators, CVSD and ADM

The experiment includes two delta modulators, CVSD, and a double integration version of CVSD which we refer to as ADM. Both of them can be represented by the block diagram in Fig. 3. Their principal difference is in the nature of the signal feedback path which is a single integrator in CVSD and a double integrator in ADM.

In both coders, the step size voltage can be generated by an RC integrator as described in Ref. 1. The integrator input depends on the three most recent output bits. If they are identical, the step size increases; otherwise, it decreases. The adaptation equation is

$$\Delta_{k+1} = \beta\Delta_k + (1 - \beta)(V_k + \Delta_{\min}), \tag{2}$$

where

Δ_k is the step size at the kth sampling instant,

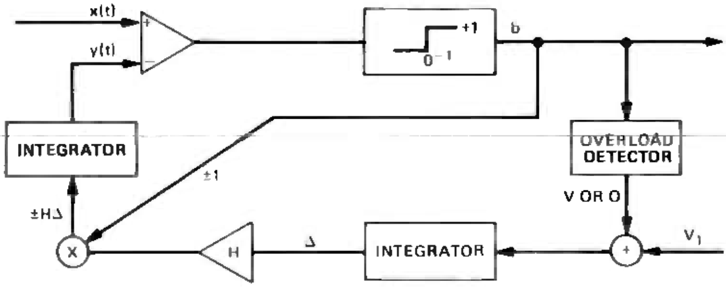


Fig. 3—Block diagram of the CVSD and ADM coders.

$\beta = 0.99$ is the step size leakage constant corresponding to an RC time constant of 6.4 ms,

$\Delta_{\min}$ is the minimum step size, and

$V_k = \Delta_{\max}$ when the three most recent outputs are identical and

$V_k = 0$ otherwise.

The dynamic range of the coder is determined by $\Delta_{\max}/\Delta_{\min}$, which is 150 (44 dB) in the CVSD and 256 (48 dB) in the ADM.

2.3.1 CVSD

As in Ref. 1, the signal feedback loop is a single integrator with a 1-ms time constant. The difference equation is

$$y_{k+1} = \alpha_1 y_k + H(1-\alpha_1)b_k\Delta_k, \tag{3}$$

where

y_k is the integrator output at the kth sampling instant,

$\alpha_1 = 0.94$ is the integrator leakage constant, corresponding to an RC time constant of 1 ms,

H = 3 is the integrator gain, and

$b_k = \pm 1$ is the kth output bit.

In the CVSD, $\Delta_{\max} = 2$ dBm, which places the center of the coder dynamic range near −21 dBm, the central value of the three signal input levels used in the experiment.

2.3.2 ADM

The double integration version of CVSD was selected for evaluation after a large number of other delta modulators were simulated. The other delta modulators differed from CVSD in one or more of the following respects:

(*i*) An exponential expandor was used in the step-size feedback loop to produce the step size

$$\Delta'_{k+1} = \exp(\Delta_{k+1}),$$

where Δ_{k+1} is given by (2). This changes the adaptation from essentially additive to essentially multiplicative.

(*ii*) The most recent two bits rather than the most recent three bits were used to determine whether the step size would increase or decrease.

(*iii*) The signal feedback loop contained a double integrator rather than a single integrator.

A limited amount of speech material was processed to evaluate these modifications. Segmental s/n was measured for each delta modulator configuration and, although some modifications resulted in better performance than CVSD for certain input levels, none of them produced substantially better results either in terms of dynamic range or peak segmental s/n. However, to provide one delta modulation alternative to CVSD, we chose the double integration ADM. Double integration is known to enhance performance significantly at higher sampling rates and to be essentially ineffective in 8-kHz DPCM (see Section 2.4). Our purpose here was to assess the effectiveness of a particular double integrator in a coder with 16-kHz sampling.

The double integrator in this coder is a second-order FIR filter that conforms to the block diagram in Fig. 4. The difference equations of the integrator are

$$u_{k+1} = y_k + H(1 - \alpha_1 - \alpha_2)b_k\Delta_k \tag{4}$$

$$y_{k+1} = \alpha_1 u_{k+1} + \alpha_2 u_k, \tag{5}$$

where

u_k is the decoder output,

y_k is the output of the encoder feedback loop,

$\alpha_1 = 1.38$, $\alpha_2 = -0.43$ are the filter coefficients, and

H = 10 is the gain.

The z-transform of the integrator is

$$\frac{\alpha_1 z^{-1}(1 - c_3 z^{-1})}{(1 - c_1 z^{-1})\,(1 - c_2 z^{-1})}, \tag{6}$$

where the integrator poles are related to the filter coefficients by

$$c_1 + c_2 = \alpha_1 \quad c_1 c_2 = -\alpha_2 \tag{7}$$

and the zero is

$$c_3 = -\alpha_2/\alpha_1. \tag{8}$$

The corresponding real poles and zero of the filter frequency response are

$$f_i = \frac{1}{2\pi T}\cos^{-1}\left[\frac{4c_i - c_i^2}{2c_i}\right], \tag{9}$$

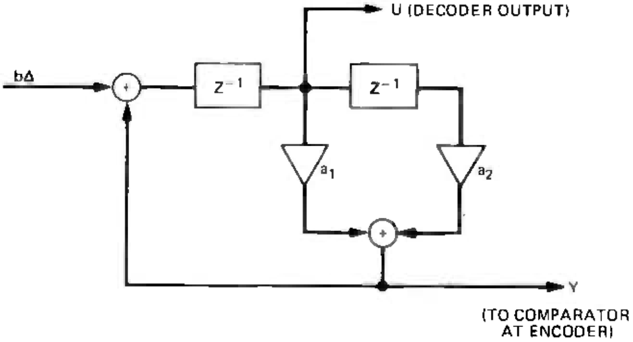


Fig. 4—Double integrator circuit for the ADM coder.

where $T = 1/16000$ s in a 16-kb/s delta modulator. With $\alpha_1 = \alpha_2 = 1.38$, -0.43, the pole frequencies are 200 and 2000 Hz, and the zero is at 3500 Hz, so that the integrator frequency response is approximately that shown in Fig. 5. In the ADM, $\Delta_{max} = -5$ dBm, which approximately centers the coder dynamic range at −21 dBm.

2.4 ADPCM

Figure 6 shows the block diagram of the ADPCM system. In an error-free environment, the primed quantities at the receiver are equal to the corresponding ones at the transmitter. In the encoder error sample $e(k)$ is generated as the difference between the input speech sample $s(k)$ and a predicted sample $\hat{s}(k)$. After quantization with 2 bits/sample, the prediction error at both receiver and transmitter is added to the predicted sample to give the reconstructed sample $r(k)$. The predicted sample $\hat{s}(k)$ is derived from the previous reconstructed one, $r(k-1)$, by a first-order transversal predictor:

$$\hat{s}(k) = 0.78r(k-1). \tag{10}$$

The coefficient 0.78 was computed by the usual mean-square error minimization technique[9] under the hypothesis of an overall s/n of about 10 dB.

The 2-bit coding of the prediction error is effected by means of the adaptive step size $\Delta(k)$, which is computed as proportional to a short-time estimate $\sigma(k)$ of the absolute magnitude of the quantizer input. The estimate $\sigma(k)$ is computed recursively from the decoded prediction error $d(k)$ so that no side information has to be transmitted. The

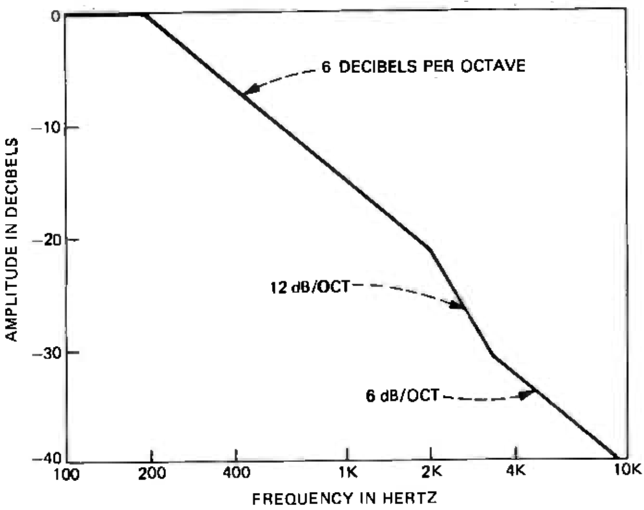


Fig. 5—Frequency response of the double integrator circuit.

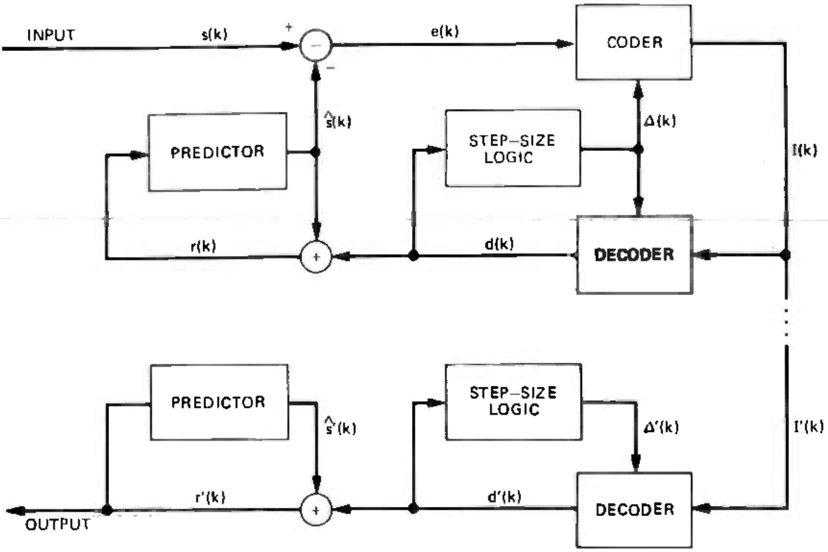


Fig. 6—Block diagram of the ADPCM coder.

entire adaptation process is summarized by the following two equations:

$$\Delta(k) = C\sigma(k) \tag{11}$$

$$\sigma(k) = \alpha\sigma(k-1) + (1-\alpha)\,|\,d(k-1)\,|\,. \tag{12}$$

In eq. (11), the parameter C, the load-constant, determines in the steady state the magnitude of the average step-size and hence the amount of granular noise and overload distortion. In eq. (12), the parameter α determines the speed of response of the adaptation algorithm to input level changes: a relatively small value of α produces fast response but an inaccurate estimate in steady state.

For simulating a practical implementation, the step size $\Delta(k)$ was constrained to assume values in the range $(\Delta_{\min}, \Delta_{\max})$ with

$$\frac{\Delta_{\max}}{\Delta_{\min}} = 256. \tag{13}$$

Values of C, α, and $\Delta_{\min}$ were calculated by optimizing a prediction of the subjective opinion score, obtained from separate measures of granular noise and overload distortion. The integrator coefficient was $\alpha = 0.875$, corresponding to a time constant of 1 ms. The minimum step size which produced the same degradations at the high and low end of the input level range of interest was found to be −63 dBm. The maximum step size is −15 dBm, while the rms speech input level assumes values in the range −36 dBm to −6 dBm.

2.5 The sub-band coder

Sub-band coding is a waveform coding technique in which the speech band is partitioned into typically 4 or 5 sub-bands by bandpass filters. Each sub-band is then lowpass-translated to dc, sampled at its Nyquist rate, and then digitally encoded using adaptive PCM (APCM) encoding. By this process of dividing the speech band into sub-bands, each sub-band can be preferentially encoded according to perceptual criteria for that band. On reconstruction, sub-band signals are decoded and bandpass-translated back to their original bands. They are then summed to give a replica of the original speech signal.

A particularly attractive implementation of the sub-band coder in terms of hardware is based on an integer band sampling approach.[10-12] With this approach, the modulations to lowpass at the transmitter and to bandpass at the receiver are inherent in the sampling process. The implementation is illustrated in Fig. 7. Bandpass filters BP_1 to BP_N in the transmitter and receiver serve to partition the input speech into N sub-bands. The coders and decoders encode the sub-band signals and the multiplexer combines these digital signals and synchronizing bits into a single bit stream for transmission over the digital channel.

Table I shows the choice of bands and bit allocations used in the 16

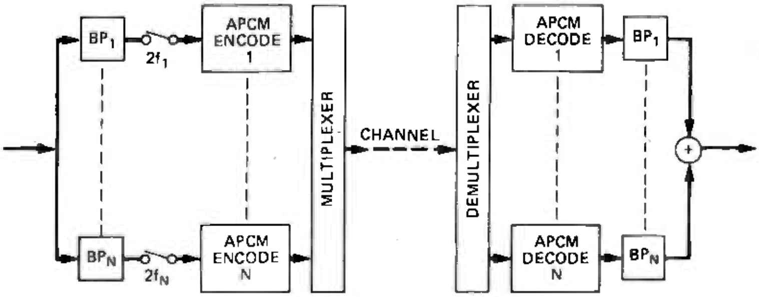


Fig. 7—Block diagram of the sub-band coder.

Table I—16 kb/s 5-band sub-band coder

Band	Band Edges (Hz)	Sampling Freq (Hz)	Min. Step-size (dB)	Bit Allocation	Kb/s
1	178–356	356	(Ref)	4	1.42
2	296–593	593	0	4	2.37
3	533–1067	1067	0	3	3.20
4	1067–2133	2133	−3	2	4.27
5	2133–3200	2133	−8	2	4.27
Sync					0.47
					16.00

kb/s coder. The coder is a 5-band design which was proposed in Ref. 11. Column 2 shows the frequency range covered by each sub-band. The bit allocation refers to the number of bits/sample used by the coders in each sub-band. As seen from the table, more accuracy is allowed for encoding the lower bands for reasons explained in Ref. 11.

The frequency range of the coder extends from 200 to 3200 Hz. A plot of the frequency response, shown in Fig. 8, reveals small notches at 1067 and 2133 Hz. These notches are due to the transition bands of the filters in bands 4 and 5. Subjectively, they are not very perceptible. Bands 1 to 3 are overlapped to avoid such notches at lower frequencies. The filters are sharp cutoff, 200 tap, FIR filters.

Column 4 in Table I contains the minimum step sizes of the APCM coders, expressed in decibels, relative to the minimum step size of band 1. This choice of minimum step sizes is different than that suggested in Ref. 11 and was found to give a better matching of the dynamic ranges of the sub-bands.

III. OBJECTIVE MEASUREMENTS

Four different objective measurements were made on the waveform coders. They are conventional signal-to-noise ratio, *SNR*, two types of segmental signal-to-noise ratios, *SEG*1 and *SEG*2, and an LPC spectral distance measure, D. In addition, D was used to evaluate the performance of the LPC vocoder and the tandem connections of the waveform coders and the LPC vocoder. In this section we briefly define each of these objective measures.

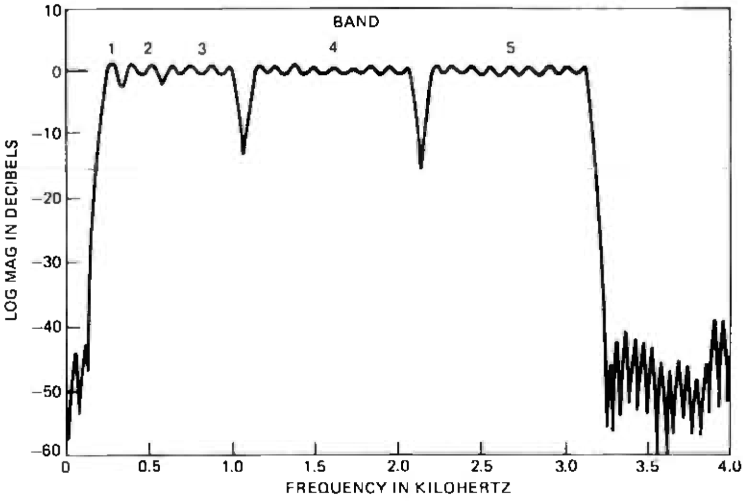


Fig. 8—Frequency response of the sub-band coder.

3.1 Conventional SNR

The most commonly used measure of performance of digital coders has been the conventional signal-to-noise ratio evaluated over an utterance of speech. The speech power is defined as

$$\hat{p} = \sum_m x^2(m), \tag{14}$$

and the noise power is defined as

$$\hat{n} = \sum_m (x(m) - y(m))^2, \tag{15}$$

where $x(m)$ and $y(m)$ are the input and output signals of the coder, respectively, and the summations in (14) and (15) are taken over the entire speech utterance. The conventional s/n is then defined as

$$\textit{SNR} = 10 \log(\hat{p}/\hat{n}). \tag{16}$$

In measuring the input and output signals of the coders, it is generally desirable to compensate for the effects of lowpass or bandpass filtering. This is done by the arrangement shown in Fig. 9. The input speech signal $s(m)$ is coded to form the output speech signal $y(m)$. It is also filtered with the same filters used in the coder to generate a compensated reference signal $x(m)$ which is used as the input signal in (14) and (15). *SNR* is, therefore, strictly a measure of coder distortions and is not affected by bandlimiting or group delay in the filters.

3.2 SEG1

While *SNR* is the most widely used criterion in measuring coder distortion, it has also long been known that in many situations it does not correlate well with subjective performance.[13] Another definition of signal-to-noise ratio, however, recently proposed by Noll,[3] does appear to correlate better with subjective performance. This measure is based on s/n measurements made over segments of speech which are typically about 20 ms in duration. An average over all of the segments in the speech utterance is then taken to obtain a composite measure of

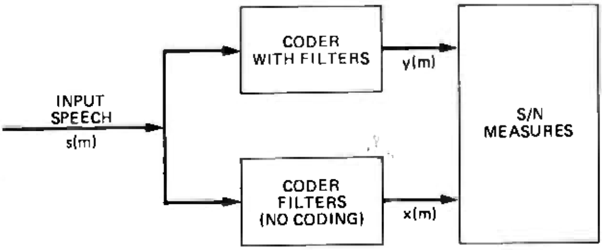


Fig. 9—Circuit for measuring signal-to-noise ratios.

performance for the entire utterance. If $(\mathrm{s/n})_i$ corresponds to the signal-to-noise ratio in decibels for a segment, i [computed in the same manner as in (16)], the segmental s/n (SEG1) is then defined as

$$\text{SEG1} = \frac{1}{N} \sum_{i=1}^{N} (\mathrm{s/n})_i \quad (\mathrm{dB}), \tag{17}$$

where it is assumed that there are N 20 ms segments in the speech utterance.

Problems arise in this definition of segmental s/n when intervals of silence exist in the speech utterance. In segments where the input signal $x(n)$ is essentially zero, any slight noise will give rise to large negative $(\mathrm{s/n})_i$, and these segments may unduly dominate the average in (17). To prevent this anomaly, we first identify those segments which correspond to silence and exclude them from the average in (17). This is achieved by means of a simple threshold. Let $\hat{p}_i$ represent the (average) speech energy in a segment, i, so that

$$\hat{p}_i = \frac{1}{K} \sum_{m=1}^{K} x^2(m), \tag{18}$$

where K corresponds to the number of speech samples in the segment. Then the segment will be included in the computation of SEG1 in (17) if its energy exceeds a threshold T, i.e., if $\hat{p}_i > T$. If it does not exceed this threshold, it is not included in the average in (17). Furthermore, to prevent any one segment from dominating the average we also limit the value of $(\mathrm{s/n})_i$ to a range of -10 to $+80$ dB. That is, $-10 \leqslant (\mathrm{s/n})_i \leqslant 80$ dB. In computer simulations, the 16-bit wordlength admitted signal levels in the range ±32767 and we set T to 900, corresponding to -55 dBm.

3.3 SEG2

The definition of this measure is

$$\text{SEG2} = \frac{1}{N} \sum_{i=1}^{N} 10 \log_{10}(1 + \hat{p}_i/\hat{n}_i) \quad (\mathrm{dB}), \tag{19}$$

where $\hat{p}_i$ is the signal power in segment i and $\hat{n}_i$ is the noise power in segment i. They are defined (on segments) according to eqs. (14) and (15), respectively.

Unlike SEG1, SEG2 does not have any thresholds. It is self-limiting to a lower value of 0 dB due to the addition of the constant 1 inside the logarithm. As in the SEG1 measure, SEG2 uses 20 ms segments.

3.4 LPC distance measure

The fourth objective measure was the LPC distance proposed by Itakura.[14] The distance between two frames of speech with LPC coefficient vectors **a** and **â**, and with autocorrelation matrices **V** and **V̂** is defined as

$$D_1 = d(\mathbf{a}, \hat{\mathbf{a}}) = \log\left[\frac{\mathbf{a}\hat{\mathbf{V}}\mathbf{a}^t}{\hat{\mathbf{a}}\hat{\mathbf{V}}\hat{\mathbf{a}}^t}\right], \tag{20}$$

where **a** and **â**, are $(p + 1)$ component vectors and **V** and **V̂** are $(p + 1) \times (p + 1)$ matrices, where p is the order of the LPC analysis.

D_1 is a measure of the distance between frames of speech. This distance, however, does not satisfy exactly all the properties of a true distance metric, i.e.,

$$d(\mathbf{a}, \hat{\mathbf{a}}) \neq d(\hat{\mathbf{a}}, \mathbf{a}). \tag{21}$$

However, for cases when $d(\mathbf{a}, \hat{\mathbf{a}})$ is small, the inequality of eq. (21) is almost an equality. To compensate for this lack of symmetry, it is convenient to define a second distance, D_2, as

$$D_2 = d(\hat{\mathbf{a}}, \mathbf{a}) = \log\left[\frac{\hat{\mathbf{a}}\mathbf{V}\hat{\mathbf{a}}^t}{\mathbf{a}\mathbf{V}\mathbf{a}^t}\right] \tag{22}$$

and an average distance between the two frames is now given by

$$D = \frac{D_1 + D_2}{2}. \tag{23}$$

Equation (23) defines a true distance metric which can be used to measure the distance (dissimilarity) between two frames of speech. It can readily be shown[15] that either D_1 or D_2 can be expressed in terms of spectral differences between the LPC models for the two frames of speech.

D_1 and D_2 were measured for every utterance used in the tests to be described later. They were measured on a frame-by-frame basis and averaged across the entire utterance to give an overall LPC distance between the original and processed version of a sentence. Figure 10 shows the system used to measure LPC distance for a single coder. The box labeled **delay** was used to compensate any delay inherent in the coder, and the bandpass filters were used both to eliminate out-of-

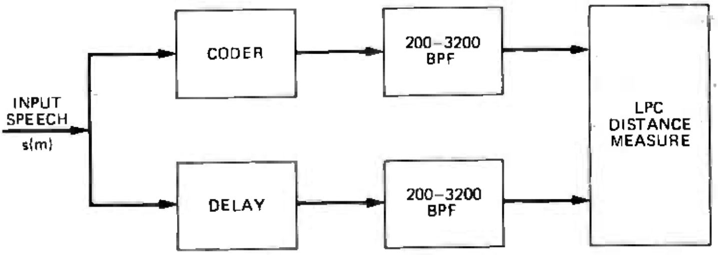


Fig. 10—Circuit for measuring LPC distance measures.

band quantization noise generated in the coder and to guarantee that the bandwidths of both the original and coded utterances were the same.

3.5 Comparison of SEG and D

Figures 11 and 12 show a series of plots for two of the utterances used in the experiments (encoded with the ADPCM coder). Part **a** of each figure shows the rms energy of the utterance as a function of time (frame number), part **b** of each figure shows the segmental s/n versus frame number, and part **c** of each figure shows the LPC distances

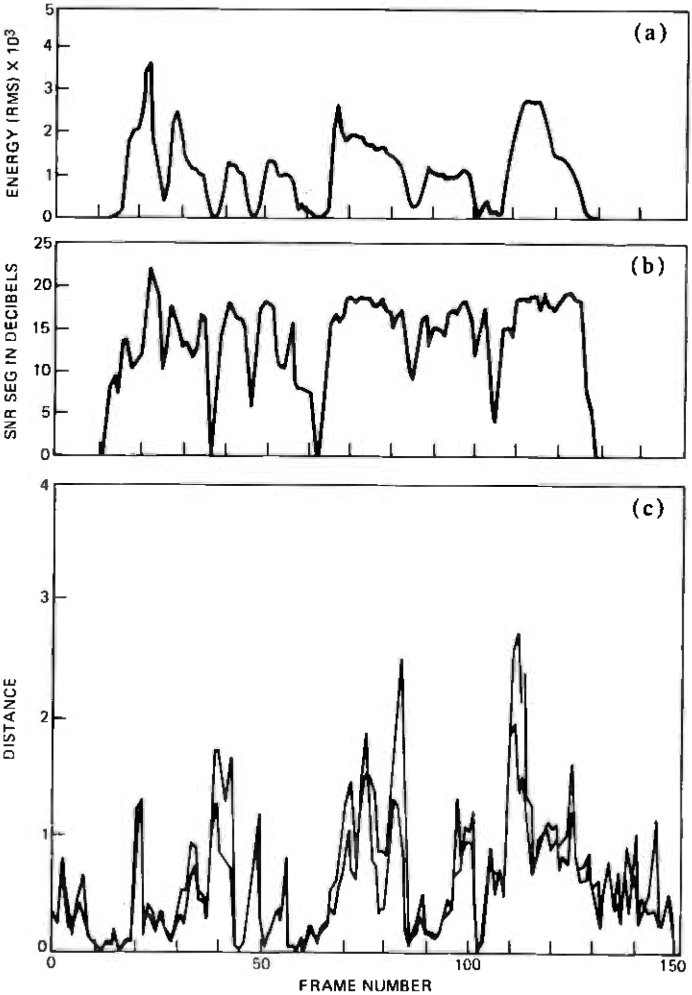


Fig. 11—Objective measurements as a function of time for utterance A. (a) rms energy of the input signal. (b) Segmental s/n-*SEG*1. (c) LPC distance.

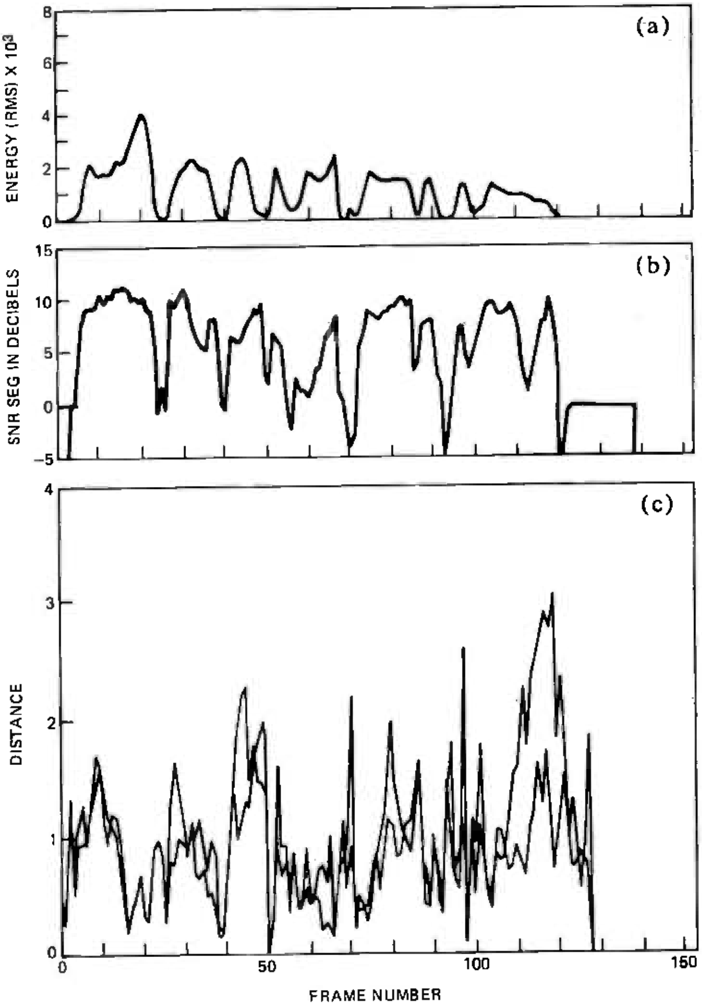


Fig. 12—Objective measurements as a function of time for utterance *B*. (a) rms energy of the input signal. (b) Segmental s/n-*SEG*1. (c) LPC distance.

(both D_1 and D_2) versus time. The utterance of Fig. 11 had an average LPC distance of about 0.60, whereas the utterance of Fig. 12 had an average LPC distance of 0.97. It can be seen in both figures that most of the time $D_1 \approx D_2$; however when either D_1 or D_2 was large, the differences between D_1 and D_2 were often significant. It can also be seen in these figures that the LPC distance and the segmental s/n do not measure similar types of distortion—i.e., when the segmental s/n is small (indicating temporal distortion of the waveform) the LPC distance is not necessarily large (indicating spectral distortion of the signal). Finally, it can be seen that a large degree of variation (on a frame-by-frame basis) occurs with both the segmental s/n and the LPC distance. Thus, a single number which characterizes the "distortion" across the entire utterance may have little meaning in many cases.

IV. EXPERIMENTAL DESIGN

4.1 Circuit conditions

The experiment tested 37 different communication circuits, each characterized by three parameters: direction, coder, and level. There were three directions: (*i*) single link with a waveform coder or vocoder alone, (*ii*), LPC-to-waveform, as in Fig. 1, (*iii*) waveform-to-LPC as in Fig. 2. There were five different single links, four with waveform coders and one with a vocoder. Each waveform coder was tested with speech at three different input levels, −36 dBm, −21 dBm, and −6 dBm. The corresponding settings of the gain parameter, G, were 0.178, 1.00 and 5.62, respectively. The speech level at the vocoder input was always −21 dBm. Thus, there were 13 single link configurations, in all. Each of the other two directions had 12 circuit configurations, comprising all combinations of four waveform coders and three input levels.

4.2 Speech material

For this experiment, a substantial digital speech library was prepared. Four talkers, two male and two female, read 40 different sentences, 2 to 3 seconds long, each talker reading from a different phonetically balanced list. The talkers were seated in a sound-proof booth and spoke into a high-quality dynamic microphone. The amplified microphone signal was lowpass-filtered at 3.2 kHz, sampled and converted into digital form by a 16-bit A/D converter operating at 8-kHz sampling frequency and finally written onto a magnetic disk. All the sentences were digitally equalized to the mean power level of −21 dBm.

For each of the 37 circuit conditions, sentences spoken by each of the four talkers were processed, generating a total of 148 stimuli. Different sentences were used in each case so that in the set of 148 stimuli the same sentence was never heard twice. With this format, we speculate that intelligibility of the processed speech played an important role in determining quality judgments. In tests containing a few sentences, presented many times, each sentence becomes recognizable to subjects even in conditions severe enough to make it quite unintelligible at first hearing. It is our hypothesis that, in such tests, there is a lower correlation between intelligibility and subjective quality than in the tests reported here.

4.3 Procedure

The 148 stimuli were recorded in different random orders on 4 analog tapes. Twenty-two students from the junior and senior classes of local high schools served as paid subjects. They listened to the processed speech monaurally over Pioneer SE 700 earphones at 80 dB SPL while seated in a double-walled sound booth with frequency-weighted room noise introduced at a level of 50 dBA. The total listening time for each group of subjects was about 30 minutes, with a short break after the 80th sentence. After each stimulus, the subjects had 4 seconds for recording their judgments. They were instructed to rate the quality of the stimulus by checking on their answer sheets a value between 1 and 9, using 1 for the worst conditions, 9 for the best ones, and intermediate numbers for intermediate qualities. Before the actual test, the subjects listened to 12 practice sentences, different from those used in the experiment, spanning the range of quality in the experiment.

V. SPEECH QUALITY RESULTS

Variability in the subjective and objective measures of quality of the 148 processed speech samples can be attributed to several (variable) sources, namely:

(*i*) The "direction" of the circuit: LPC-to-waveform coder, waveform-coder-to-LPC, or single link.

(*ii*) The waveform coder.

(*iii*) The speech level at the input to the coder.

(*iv*) The talker.

(*v*) The sentence.

(*vi*) The listener (subjective data only).

(*vii*) Inconsistency of each listener (subjective data only).

We are primarily interested in how the first two variables, circuit direction and waveform coder, influence quality. Inferences about these variables would be simple if they accounted for most of the variance in the data or if they did not interact substantially with the other variables. Unfortunately, neither of these conditions is met by our data, and many of our inferences about circuit direction and waveform coder will be more qualified than we would like them to be.

5.1 Subjective data

5.1.1 Listeners

The amount of listener agreement was fairly low relative to other speech quality experiments.[3,13] For each pair of listeners, we computed the correlation coefficient of the 148 ratings. The median of the correlations was only 0.49. The 25th and 75th percentiles were 0.41 and 0.60, respectively. With respect to the 148 mean ratings (averaged over the 22 listeners), individual listener correlations ranged from 0.50 to 0.85, which suggests that no subject was very idiosyncratic in his ranking of stimulus conditions.

Figure 13 gives plots of the rating scores of each of the 22 subjects for the LPC system alone and for each of the four talkers. The large variability among subjects is readily seen. For example, for talker 3 the average rating was 7.3. However, two subjects gave this circuit rating of 1 (the lowest possible), whereas 13 subjects gave it a rating of 8 or 9 (the highest possible). Similar variability was found in the scores for almost every test condition.

The 148 listener averages are presented in Table II, where we also provide aggregates of these averages across input level and talker. The aggregated mean values show the overall effects of circuit direction and waveform coder.

5.1.2 Sentences

In many subjective testing experiments, listeners hear one or a few sentences repeatedly. To achieve closer conformity to practical communication situations, a different sentence for each stimulus condition was used. A disadvantage of this design is the lack of any control for or means of testing the effect of sentence content on the quality measures. The variability due to sentences appears in and enhances the experimental error; i.e., the variance that cannot be accounted for in statistical analyses.

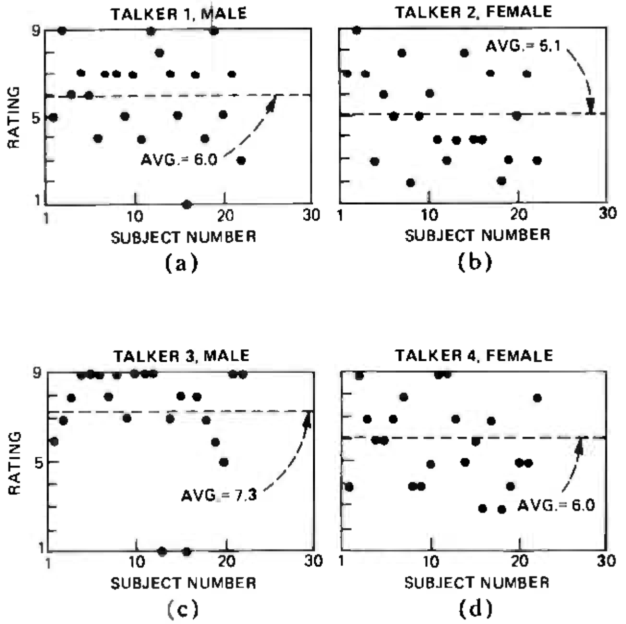


Fig. 13—Rating scores as a function of subject for the individual LPC circuit for each of the four talkers.

5.1.3 Talker effects

Averaged over all 37 circuit conditions (combinations of input level, coder, and direction), the ratings of speech from the two male talkers were 4.98 and 4.94. The averages for the two females were 3.99 and 4.00. A three-way analysis of variance (listener by talker by circuit condition) revealed a very significant talker effect. Clearly, this effect is predominantly due to listeners giving lower ratings to distorted female speech than to distorted male speech. However, there is also a substantial talker-circuit interaction, indicating that differences in ratings of male and female speech are by no means uniform across experimental conditions. (In fact, with fairly low distortion as in subband coding in a single link, the male and female averages are virtually the same—7.13 and 7.20, respectively.) This nonuniformity is evident in Table II which also reveals that, although the overall ratings of the two males are virtually identical, there are substantial differences from condition to condition in the ratings of the male voices, and likewise for the female voices.

5.1.4 Input level

The step sizes of the waveform coders were adaptable over a range of 44 dB (for the CVSD) or 48 dB (for the other three coders). With the rms input level varying over a range of 30 dB and individual sounds within a sentence exhibiting a wide range of levels, the weak sounds of the low-level signals were subject to greater-than-average granular quantizing noise, while the strong sounds of the high level sentences were susceptible to overload. The maximum and minimum step sizes of each waveform coder were chosen with the aim of centering the dynamic range of subjective quality in the −15 to +15 dB range of input levels.

Table III shows that this design effort was entirely successful with the CVSD and ADPCM coders in which the dynamic range of subjective performance is exactly symmetric around the 0-dB input level. In the SBC and ADM coders, the overload distortion of the +15 dB input level was less harmful subjectively than the granular noise produced with the input set at −15 dB. In these coders, a better balance of granularity and overload would have been achieved with lower minimum step sizes.

5.1.5 Coder and direction

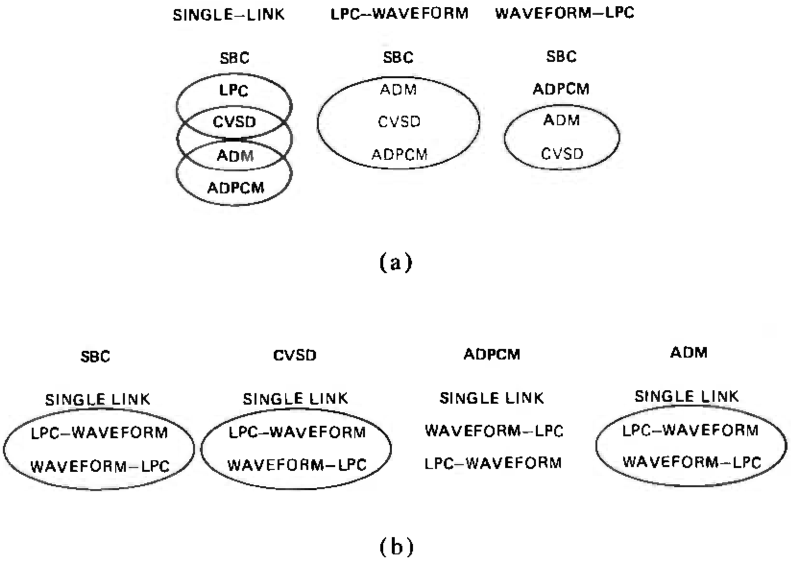
We have used the Tukey HSD criterion[16] to evaluate the relative merits of the 13 communication system configurations listed in Fig. 14. Figure 14a shows, for each circuit direction, groupings of coders for which the null hypothesis cannot be rejected at the 0.05 level. In all cases, SBC is superior to any of the other waveform coders. In the LPC

Table II—Average subjective ratings (over 22 listeners)

		Single Link					LPC → Waveform				Waveform → LPC			
		SBC	CVSD	ADPCM	ADM	LPC	SBC	CVSD	ADPCM	ADM	SBC	CVSD	ADPCM	ADM
	−15	6.8	6.3	5.4	4.9		4.5	3.2	2.9	3.1	4.7	2.5	5.7	3.9
M1	0	6.3	6.7	7.2	7.0	6.0	4.3	4.9	4.3	4.5	4.5	2.9	5.1	4.0
	+15	7.5	6.2	5.9	6.8		5.0	4.8	4.0	4.5	4.5	3.8	5.5	4.2
	Ave (level)	6.9	6.4	6.2	6.2		4.6	4.3	3.7	4.0	4.6	3.0	5.4	4.0
	−15	7.6	7.0	6.5	5.5		5.1	3.9	3.8	3.7	4.8	4.3	3.5	3.6
M2	0	7.6	6.9	5.6	5.9	7.3	5.7	2.9	4.0	3.5	6.1	3.6	4.7	3.5
	+15	6.9	5.5	5.6	5.9		6.1	3.1	2.5	4.6	5.0	3.8	4.0	3.4
	Ave (level)	7.4	6.5	5.9	5.8		5.6	3.3	3.5	3.9	5.3	3.9	4.1	3.5
Ave (M1, M2, level)		7.1	6.5	6.0	6.0	6.6	5.1	3.8	3.6	4.02	4.9	3.5	4.8	3.8
	−15	7.0	4.8	3.5	3.2		3.8	2.3	2.1	3.1	2.0	3.9	2.4	2.0
F1	0	7.9	6.2	5.4	4.5	5.1	5.3	2.8	3.7	2.2	1.6	3.3	2.3	2.7
	+15	7.8	3.9	4.8	6.5		6.5	2.9	2.5	3.1	6.9	3.0	3.4	3.3
	Ave (level)	7.6	5.0	4.5	4.7		5.2	2.7	2.8	2.8	3.5	3.4	2.7	2.7
	−15	5.7	5.8	4.6	4.1		3.4	3.6	3.5	2.9	4.2	1.8	3.3	2.8
F2	0	7.4	4.7	3.4	6.0	6.0	4.8	2.6	3.2	4.9	4.3	1.9	3.2	2.5
	+15	7.4	5.8	4.5	4.6		3.9	3.0	2.1	3.1	4.5	3.2	2.5	2.7
	Ave (level)	6.8	5.4	4.2	4.9		4.0	3.1	3.0	3.6	4.3	2.3	3.0	2.7
Ave (F1, F2, level)		7.2	5.2	4.3	4.8	5.6	4.6	2.9	2.9	3.2	3.9	2.8	2.9	2.7
Ave (Talker, level)		7.2	5.8	5.2	5.4	6.1	4.9	3.3	3.2	3.6	4.4	3.2	3.8	3.2

Table III—Average subjective ratings (over listeners, talkers and direction)

		Coder			
		SBC	CVSD	ADPCM	ADM
Level	−15 dB	5.0	4.1	3.9	3.6
	0 dB	5.5	4.1	4.3	4.3
	+15 dB	6.0	4.1	3.9	4.4



(a)

(b)

Fig. 14—Relative subjective quality of coding systems. Circles indicate that it is impossible to reject the hypothesis that the coders have the same quality.

→ waveform circuits, ADM, CVSD, and ADPCM have essentially the same performance. In the waveform → LPC direction, ADPCM is better than ADM and CVSD, which exhibit essentially the same quality.

Figure 14b shows the equivalent groupings across direction. The salient inferences from these groupings is that, for each waveform coder, the single link substantially outperforms either of the tandem connections. The two tandem directions have essentially the same quality when SBC, CVSD, or ADM is the waveform coder. The ADPCM → LPC tandem is significantly better than the LPC → ADPCM tandem.

5.2 Objective measurements of quality

Results of the objective measurements discussed in Section III are presented in Tables IV and V. Table IV gives results for the performance of the single-link circuits in terms of *SNR*, *SEG*1, *SEG*2, and LPC

Table IV—Objective measurements of single link coders

	Talker: M1						Talker: F1				
Level	Coder	*SNR*	*SEG*1	*SEG*2	*D*	Level	Coder	*SNR*	*SEG*1	*SEG*2	*D*
−15	SBC	13.2	11.9	8.7	0.82	−15	SBC	17.7	15.3	12.3	0.53
0	SBC	14.4	13.4	8.8	0.50	0	SBC	14.4	13.3	10.5	0.51
+15	SBC	13.8	10.2	8.6	0.80	+15	SBC	13.2	14.9	12.0	0.34
−15	CVSD	12.1	10.2	9.1	0.65	−15	CVSD	14.8	12.1	11.6	0.84
0	CVSD	8.4	12.1	11.0	0.54	0	CVSD	17.5	15.2	13.8	0.65
+15	CVSD	3.5	8.3	9.8	0.54	+15	CVSD	8.2	11.8	12.3	0.55
−15	ADPCM	12.6	12.8	10.3	0.54	−15	ADPCM	16.2	16.2	13.9	0.61
0	ADPCM	10.1	12.3	12.0	0.34	0	ADPCM	16.5	14.0	13.9	0.50
+15	ADPCM	3.6	10.6	12.0	0.37	+15	ADPCM	8.2	12.6	13.1	0.58
−15	ADM	14.6	10.5	9.5	0.64	−15	ADM	16.5	14.7	12.6	0.89
0	ADM	12.3	13.2	11.1	0.47	0	ADM	18.4	16.5	14.9	0.70
+15	ADM	4.7	10.7	12.2	0.47	+15	ADM	10.4	12.7	13.0	0.48
—	LPC				0.31	—	LPC				0.38
	Talker: M2						Talker: F2				
Level	Coder	*SNR*	*SEG*1	*SEG*2	*D*	Level	Coder	*SNR*	*SEG*1	*SEG*2	*D*
−15	SBC	12.3	10.6	7.6	0.76	−15	SBC	14.6	14.1	12.0	0.60
0	SBC	13.8	14.5	10.8	0.38	0	SBC	13.4	14.1	11.5	0.33
+15	SBC	12.3	14.1	11.1	0.27	+15	SBC	12.2	13.8	12.1	0.28
−15	CVSD	12.0	9.9	8.1	0.81	−15	CVSD	12.8	11.5	10.0	0.74
0	CVSD	9.4	9.6	9.4	0.43	0	CVSD	11.1	14.9	12.5	0.62
+15	CVSD	4.4	9.9	11.5	0.43	+15	CVSD	4.3	11.3	12.1	0.40
−15	ADPCM	12.8	13.0	10.5	0.50	−15	ADPCM	14.5	14.8	12.5	0.64
0	ADPCM	12.0	13.8	13.4	0.45	0	ADPCM	14.3	14.9	14.5	0.38
+15	ADPCM	5.1	11.3	13.0	0.47	+15	ADPCM	4.5	11.2	12.6	0.67
−15	ADM	14.1	10.5	8.2	0.73	−15	ADM	16.5	12.5	10.5	0.66
0	ADM	11.6	12.1	11.5	0.38	0	ADM	17.8	16.4	14.7	0.58
+15	ADM	8.8	10.7	11.6	0.35	+15	ADM	8.9	13.1	12.6	0.51
	LPC				0.35		LPC				0.47

Table V—Overall LPC distances for tandem links

First Link	Second Link	Talker: M1	Talker: F1	Talker: M2	Talker: F2
LPC	SBC	1.18	0.93	0.83	0.81
LPC	SBC	0.20	0.73	0.672	0.66
LPC	SBC	0.61	0.57	0.46	0.56
LPC	CVSD	0.91	0.92	0.88	1.10
LPC	CVSD	0.66	1.00	0.54	0.91
LPC	CVSD	0.60	0.66	0.64	0.83
LPC	ADPCM	0.79	0.90	0.69	1.09
LPC	ADPCM	0.71	0.75	0.53	0.68
LPC	ADPCM	0.61	0.72	0.58	0.76
LPC	ADM	0.90	1.40	0.82	0.91
LPC	ADM	0.68	0.80	0.68	0.77
LPC	ADM	0.59	0.64	0.60	0.90
SBC	LPC	1.50	1.08	0.86	0.93
SBC	LPC	1.05	0.88	0.78	0.78
SBC	LPC	0.71	0.57	0.61	0.62
CVSD	LPC	0.91	0.86	0.98	1.21
CVSD	LPC	0.62	0.75	0.63	0.84
CVSD	LPC	0.62	0.82	0.56	0.89
ADPCM	LPC	0.75	0.76	0.71	0.79
ADPCM	LPC	0.63	0.80	0.67	0.87
ADPCM	LPC	0.51	0.74	0.60	0.76
ADM	LPC	0.79	0.71	0.86	1.06
ADM	LPC	0.58	0.81	0.64	0.80
ADM	LPC	0.47	0.62	0.56	0.69

distance D for each of the four talkers used in the experiment. Due to the large variability of the objective measures across talkers and sentences (a different sentence was used for each condition), it is difficult to make meaningful comparisons across conditions. A similar variability was observed for the objective measurements across individual coders in the tandem links.

Table V gives results for LPC distance for the overall tandem links. Again, a large variability is seen across conditions due to the different sentences used for each measurement.

5.3 Relationship of subjective and objective measures

5.3.1 Correlations

Previous studies have demonstrated the inadequacy of *SNR* as an indicator of subjective quality and have pointed to segmental signal-to-noise ratio and to LPC distance metrics as more promising measures. In the present experiment, the diversity of speech material and of signal-processing approaches exceed those of previous studies, and thus the merits of single measures and combinations of measures as subjective quality indicators are tested more critically than ever before.

Table VI shows correlations of average rating with each of the objective measures. The subscripts A, B, and AB, appended to *SNR*, *SEG*1, *SEG*2, and D, refer to measures taken on the first link of a tandem circuit (or the entire single-link circuit), the second link of a tandem circuit, and the overall circuit, respectively.

Table VI indicates that the diversity of conditions either eliminates or dilutes the value of each of the measures as a predictor of speech quality. The table gives correlations of average rating (over 22 subjects) with each one of the objective measures. There are nine objective measures; 3 s/n's and one LPC distance for each half of a tandem connection, and the overall LPC distance. Except for D_A, the LPC distance of the first link, and D_{AB}, the overall LPC distance, none of the measures is applicable to all conditions. (For example, s/n is measured only in the first link in the single-link and waveform-to-LPC circuits. It is measured only in the second link in the LPC-to-waveform circuits.) In addition to the correlation, the table shows the number of data points used in the computation and the significance (two-tailed) of the null hypothesis that the coefficient is zero.

It should be noted that, for all talkers, the only statistic for which the null hypothesis can be rejected at the 0.01 level is D_{AB}, the overall distance. The two-tailed significance level for *SEG*2_A is 0.001, but the correlation is negative. Surely a one-tailed test applies here, and the null hypothesis cannot be rejected. Computing correlations for ratings of male and female talkers separately, we see the same situation, except that *SEG*1_B is significant at the 0.01 level as a predictor of male speech quality on the LPC-to-waveform tandems.

Table VI—Correlations of average ratings with objective measures

	All Talkers			Male Talkers			Female Talkers		
	Corr.	No. of Conditions	Signif.	Corr.	No. of Conditions	Signif.	Corr.	No. of Conditions	Signif.
SNR_A	−0.115	96	0.1	−0.038	48	0.4	0.014	48	0.5
$SEG1_A$	−0.175	96	0.04	−0.043	48	0.4	0.046	48	0.4
$SEG2_A$	−0.309	96	0.001	−0.196	48	0.09	−0.167	48	0.1
D_A	−0.089	148	0.1	0.233	74	0.02	−0.193	74	0.05
SNR_B	0.087	48	0.3	0.254	24	0.1	0.382	24	0.03
$SEG1_B$	0.072	48	0.3	0.510	24	0.005	0.381	24	0.03
$SEG2_B$	−0.123	48	0.2	0.318	24	0.06	0.031	24	0.4
D_B	−0.149	96	0.07	−0.056	48	0.4	−0.086	48	0.3
D_{AB}	−0.590	148	0.001	−0.415	74	0.001	−0.709	74	0.001

The poor correlations of practically all s/n measures with subjective quality has led us to abandon all of them as performance indicators of the tandem circuits and to focus our attention on the LPC distance measures.

5.3.2 Prediction of subjective quality

Working with the LPC distance measure for the first link of a tandem connection, D_A, the distance measure for the second link, D_B, and D_{AB}, the overall distance measure, linear regression procedures, were applied to find formulas for predicting the average ratings, $\overline{R}$, of the 148 circuit conditions. The best linear combination of the three distances was

$$\overline{R} = -5.48D_A - 6.47D_B + 2.52D_{AB} + 7.38. \qquad (24)$$

The standard deviation of the 148 mean ratings was 1.55 units on the 9-point scale and the standard error of this regression was 1.10. The proportion of variance accounted for is thus 51 percent, and the multiple correlation coefficient is 0.712.

The prediction accuracy can be improved somewhat by accounting for the fact that ratings and LPC distances are related differently for male and female talkers. We have done so by introducing a new variable, M, where $M = 1$ for male talkers and $M = 0$ for female talkers. Introducing M to the regression, we have

$$\overline{R} = -4.99D_A - 5.98D_B + 2.14D_{AB} + 0.48M + 6.85. \qquad (25)$$

Here the standard error is 1.08, i.e., 53 percent of the variance is accounted for and the multiple correlation coefficient is 0.727.

Various transformations of the distance data were also studied and a simple log transform proved useful in regression equations. We define the transform variables

$$L_A = \ln(D_A); \qquad L_{AB} = \ln(D_{AB})$$

and

$$\begin{aligned} L_B &= \ln(D_B) \text{ in tandem circuits and} \\ &= -4.0 \text{ in single-link circuits.} \end{aligned}$$

The value -4.0 has been chosen empirically. (It corresponds to a distance of 0.018. The lowest measured distance was 0.21, which was observed for several sentences processed by LPC.)

Using the log-transformed distances, the regression equations corresponding to (24) and (25) are

$$\overline{R} = -1.55\mathrm{L}_A - 0.785\mathrm{L}_B - 0.211\mathrm{L}_{AB} + 1.59 \qquad (26)$$

and

$$\bar{R} = -1.34L_A - 0.782L_B - 0.0622L_{AB} + 0.643M + 1.51. \qquad (27)$$

The standard error of (26) is 1.02, which accounts for 58 percent of the variance in average ratings and the multiple correlation coefficient is 0.760. The corresponding statistics for (27) are 0.973, 62 percent, and 0.785.

VI. DISCUSSION

These data analyses allow us to make generalized statements in answer to the three questions posed in Section 1.2. Owing to the interactions in the data, there are specific exceptions to many of the general conclusions of the following subsections.

6.1 Quality of tandem connections

A strong conclusion of the study is that any tandem connection of the vocoder is substantially worse than either of the two corresponding single links. Although we did not attach descriptive adjectives to rating categories, we have the impression that ratings below about 4.0 reflected degradations severe enough to render a circuit inadequate for effective communication.

In our judgment, the results of this experiment strongly suggest that a tandem connection involving any of the three differential waveform coders (CVSD, ADPCM, or ADM) is inadequate. It appears that the LPC-SBC tandem could provide reasonable communication in many circumstances, but that the SBC-LPC tandem is of marginal use.

6.2 Alternatives to CVSD

Only the sub-band coder, which is substantially more complicated, offers significantly better performance than CVSD over all circuit conditions, talkers, and input levels. ADM, a double integration version of CVSD, has the same subjective quality (within the bounds of experimental error) and ADPCM is better than CVSD in one tandem direction, equal to CVSD in the other tandem direction and worse than CVSD in the single link configuration. The ADPCM coder was designed by extrapolating, to 16 kb/s, results of an experiment involving 24 kb/s and 32 kb/s coders. The result of this design optimization was a coder that adapts somewhat more slowly than ADPCM coders used elsewhere. It may be that higher quality could be obtained with a faster adaptive quantizer in the ADPCM coder.

6.3 Objective measures

The wide variety of circuit conditions and speech material either destroyed or strongly diluted the value of the objective measures as indicators of speech quality. With the wide range of input levels, the outputs of differential waveform coders contained various types of additive noise and signal distortion. Meanwhile, the sub-band coder and LPC each have their own peculiar distortions; a reverberant effect and a mechanical buzziness, respectively. The presence of all these impairments in the single link circuits and their combinations in the tandem circuits together present a diversity of quality that would be very hard to describe with a single measure.

While the wide range of circuit conditions produces great subjective variability, the variety of speech material seems to have a strong effect on the objective measures. We speculate that sentence-to-sentence fluctuation in objective measures is greater than that of corresponding subjective impressions.

These irregularities led to regression formulas of considerably less accuracy [about 60 percent of variance accounted for by eqs. (28) and (29)] than the 70 to 90 percent obtained in other studies.[4,13] Our work lends support to the value of current efforts to find more robust objective measures.[17–19]

VII. ACKNOWLEDGMENT

We are grateful to Barbara McDermott for advice and assistance with the experimental design and the data processing.

REFERENCES

1. R. E. Crochiere, D. J. Goodman, L. R. Rabiner, and M. R. Sambur, "Tandem Connections of Wideband and Narrowband Speech Communications Systems: Part 1—Narrowband-to-Wideband Link," B.S.T.J., *56*, No. 9 (November 1977), pp. 1701–1722.
2. L. R. Rabiner, M. R. Sambur, R. E. Crochiere, and D. J. Goodman, "Tandem Connections of Wideband and Narrowband Speech Communications Systems: Part 2—Wideband-to-Narrowband Link," B.S.T.J., *56*, No. 9 (November 1977), pp. 1723–1741.
3. P. Noll, "Adaptive Quantizing in Speech Coding Systems," Int. Zurich Seminar, March 1974.
4. B. J. McDermott, C. Scagliola, and D. J. Goodman, "Perceptual and Objective Evaluation of Speech Processed by Adaptive Differential PCM," B.S.T.J., *57*, No. 5 (May–June, 1978), pp. 1597–1618.
5. R. E. Crochiere and L. R. Rabiner, "Optimum FIR Digital Filter Implementations for Decimation, Interpolation, and Narrowband Filtering," IEEE Trans. Acoust., Speech, Sig. Proc., *ASSP-23* (October 1975), pp. 444–456.
6. L. R. Rabiner, "On the Use of Autocorrelation Analysis for Pitch Detection," IEEE Trans. Acoust., Speech, and Sig. Proc., *ASSP-25*, No. 1 (February 1977), pp. 24–33.
7. J. J. Dubnowski, R. W. Schafer, and L. R. Rabiner, "Real-Time Hardware Pitch Detector," IEEE Trans. Acoust., Speech, and Sig. Proc., *ASSP-24*, No. 1 (February 1976), pp. 2–8.
8. M. R. Sambur, "An Efficient Linear Prediction Vocoder," B.S.T.J., *54*, No. 10 (December 1975), pp. 1693–1723.
9. R. A. McDonald, "Signal-to-Noise and Idle Channel Performance of Differential Pulse Code Modulation Systems," B.S.T.J., *45*, No. 7 (September 1966), pp. 1123–1151.
10. R. E. Crochiere, S. A. Webber, and J. L. Flanagan, "Digital Coding of Speech in Sub-bands," B.S.T.J., *55*, No. 8 (October 1976), pp. 1069–1086.
11. R. E. Crochiere, "On the Design of Sub-band Coders for Low-Bit-Rate Speech Communication," B.S.T.J., *56*, No. 5 (May–June 1977), pp. 747–770.
12. R. E. Crochiere, "An Analysis of 16 kb/s Sub-band Coder Performance: Dynamic

Range, Tandem Connections, and Channel Errors," B.S.T.J., *57*, No. 8 (October 1978), pp. 2927–2952.

13. D. J. Goodman, B. J. McDermott, and L. H. Nakatani, "Subjective Evaluation of PCM Coded Speech," B.S.T.J., *55*, No. 8 (October 1976), pp. 1087–1109.
14. F. Itakura, "Minimum Prediction Residual Principle Applied to Speech Recognition," IEEE Trans. Acoustics, Speech, and Signal Proc., *ASSP-23*, No. 1 (February 1975), pp. 67–72.
15. A. H. Gray, Jr. and J. D. Markel, "Distance Measures for Speech Processing," IEEE Trans. Acoustics, Speech, and Signal Proc., *ASSP-24* (October 1976), pp. 380–391.
16. N. H. Nie, C. H. Hull, J. H. Jenkins, K. Steinbrenner, and D. H. Bent, *Statistical Package for the Social Sciences*, second ed., New York: McGraw Hill, 1975, pp. 426–428.
17. C. Scagliola, "Evaluation of Adaptive Speech Coders Under Noisy Channel Conditions," Int'l Conf. on Commun., June 1978.
18. R. E. Crochiere, L. R. Rabiner, N. S. Jayant, and J. M. Tribolet, "A Study of Objective Measures for Speech Waveform Quantization," Int'l Zurich Seminar, 1978.
19. J. M. Tribolet, P. Noll, B. J. McDermott, and R. E. Crochiere, "A Study of Complexity and Quality of Speech Waveform Coders," Proceedings of the 1978 IEEE International Conf. on Acoustics, Speech and Signal Proc., Tulsa, Okla., April 1978.

THE BELL SYSTEM TECHNICAL JOURNAL
Vol. 58, No. 3, March 1979
Printed in U.S.A.

Motion-Compensated Television Coding: Part I

By A. N. NETRAVALI and J. D. ROBBINS

(Manuscript received August 29, 1978)

We present methods of estimating displacements of moving objects from one frame to the next in a television scene and using such displacements for frame-to-frame prediction. Displacement is estimated by a recursive algorithm which seeks to minimize a functional of the prediction error. Several simplifications of the algorithm are presented which make it attractive for hardware implementation. Performance of the algorithm is evaluated by computer simulations on two sequences of moving images containing various amounts and types of motion. In both cases, the use of displacement-based (or motion-compensated) prediction results in bit rates that are 22 to 50 percent lower than those obtained by simple "frame-difference" prediction, which is used commonly in the interframe coders.

I. INTRODUCTION

Television signals are generated by scanning a scene several times a second even though there may not be any change in the scene from one frame to the next. This results in a considerable frame-to-frame redundancy in the signal. Existence of this redundancy has long been recognized, and several measurements have been made to quantify it. However, the first real demonstration of a frame-to-frame coder which used redundancy between successive frames was made in 1969 by Mounts.[1] Since then, several improvements have been made to the basic frame-to-frame encoder resulting in prototypes or real implementations of coder-decoder pairs.[2-5] These are the subjects of two excellent surveys,[6,7] the first one covering material up to 1972 and the second one up to 1978. As is evident from these works, most frame-to-frame coders are based on the following:

(*i*) Segmenting each television frame into two parts, one part which is the same as the previous frame and the other part (called the moving area) which has changed from the previous frame.

(*ii*) Transmitting two types of moving area information: (*a*) addresses specifying the location of the picture elements in the moving area, and (*b*) information by which the intensities of the moving area elements can be updated.

(*iii*) Matching the coder bit rate to the channel rate. Since the motion in a real television scene occurs randomly and in bursts, the amount of information about the moving area will change as a function of time. To transmit it over a constant bit rate channel; (*a*) smooth out the transmitted information rate by storing it in a buffer prior to transmission, and (*b*) use the buffer fullness to regulate the encoded bit rate by varying amplitude, spatial, and temporal resolution of the television signal.

Intensities of the moving area picture elements are transmitted by predictive coding which sends frame difference, element difference, or line difference (or their combination) as the differential signal. Attempts have been made to optimize the picture quality within the constraints of the buffer size and the channel rate.

Simultaneously with these implementations, computer simulations have been used to explore other improvements of the frame-to-frame coders. It has long been recognized[8,9] that, if an estimate of the translation of an object is obtained, more efficient predictive encoding can be performed by taking differences of elements in the moving areas with respect to elements in the previous frame that are appropriately translated. We refer to such schemes as "Motion-Compensated Coding Schemes." Their success depends obviously on the following: (*i*) the amount of purely translational motion of objects in a real television scene,* (*ii*) the ability of an algorithm to estimate the translation with an accuracy that is desirable for good prediction of intensity, and (*iii*) robustness of the displacement estimation algorithm when amplitude, spatial, and temporal resolution of the transmitted picture are lowered due to buffer fullness.

Several methods of estimating the displacement of an object in a television scene have been proposed. Methods of point-by-point correlation[9] or pattern matching used in scene analysis[10–12] appear to be too complex for present-day implementation, especially if displacement needs to be defined with resolution finer than the sampled grid of a television frame. Simpler displacement estimation techniques[13,14] utilize the relation between the spatial differential and temporal differential signals. They would be easier to implement. Another approach is adaptive linear prediction using elements in the previous frame (or field) which are displaced in all directions (horizontal left, right; vertical up, down) by a certain maximum amount and adapting the coefficients to minimize an intensity error function.[15] All these techniques assume

* So that the television camera sees the objects in pure translation, it is also necessary that the lighting be uniform in the camera field of view.

that the displacement is constant within a block of picture elements. This assumption presents difficulties in scenes with multiple moving objects, occluding objects, as well as different parts of the same object moving with different displacements. Of course, decreasing the size of the block makes this assumption more realistic, but then the quality of displacement estimate suffers.

The techniques proposed by Haskell[15] do not require an explicit estimate of translational displacement and are applicable even if the motion is not precisely translational. However, they also work on blocks of picture elements and require a matrix inversion in addition to other complicated operations and, therefore, do not appear to be easily implementable without a significant simplification.

We present several new techniques for estimating motion. They attempt to minimize recursively a measure of the motion-compensated prediction error. Thus, given an ith estimate of displacement, we obtain the $(i + 1)$th estimate such that, in general, the motion-compensated prediction error resulting from $(i + 1)$th estimate is lower than that using the ith estimate. The recursive minimization is performed by a gradient or steepest descent algorithm. Considerable freedom exists in the specific choice of this algorithm. Our choices have been guided primarily by a desire to implement this algorithm in real time.

In Section II, we present a derivation of several motion-estimation algorithms and describe some simulations to evaluate their performance. Section III contains modifications of one of the algorithms of Section II for use in a frame-to-frame coder. Here we evaluate the performance of the algorithm in the context of a frame-to-frame coder. Section IV contains a discussion and the conclusions of our study. Many enhancements of the algorithm are possible, and some of these will be presented in a companion paper.[16]

II. MOTION ESTIMATION

In this section, we derive some simple algorithms for estimating motion. They attempt to minimize recursively a certain quantity (function of the motion estimation error). If the changes in successive television frames are due to translation of an object, then the algorithm iterates in a gradient or steepest descent direction such that the consecutive estimates converge to an estimate of translation. A proof of convergence of such a scheme under certain assumptions is given in the appendix and is supported by a large number of computer simulations presented at the end of this section.

2.1 Motion estimation in a block of pels

We mentioned in the introduction that most algorithms in the literature for estimating translation of an object from a television scene assume that the translation is constant within a block of picture elements (pels). We start with one such algorithm developed by Limb and Murphy[13] and Cafforio and Rocca[14] and show how it can be modified to obtain a better estimate of translation. This is done primarily to define a quantity which is fundamental to our recursive algorithm introduced in the next section.

Assuming a 2:1 interlaced raster format, let $I(\mathbf{x}, t-\tau)$ and $I(\mathbf{x}, t)$ be the intensity values of the two successive frames as a function of spatial location $\mathbf{x}$ (a two-dimensional vector) and time t. The time between the two frames is τ. If an object moves in translation, then in the moving area (disregarding the uncovered background):

$$I(\mathbf{x}, t) = I(\mathbf{x} - \mathbf{D}, t - \tau), \tag{1}$$

where $\mathbf{D}$ is the translation vector of the object during the time interval $[t-\tau, t]$. The frame difference at spatial position $\mathbf{x}$ is given by

$$\begin{aligned} FDIF(\mathbf{x}) &= I(\mathbf{x}, t) - I(\mathbf{x}, t-\tau) \\ &= I(\mathbf{x}, t) - I(\mathbf{x} + \mathbf{D}, t), \end{aligned} \tag{2}$$

which can be written, for small $\mathbf{D}$, by Taylor's expansion about $\mathbf{x}$ as

$$FDIF(\mathbf{x}) = -\mathbf{D}^T \nabla I(\mathbf{x}, t) + \text{Higher Order Terms in } \mathbf{D}, \tag{3}$$

where ∇ is the gradient with respect to $\mathbf{x}$ and superscript T on a vector or matrix denotes its transpose. If the translation of the object is constant over the entire moving area (except for the uncovered background) and if the higher order terms in $\mathbf{D}$ can be neglected, then both sides of the above equation can be summed over the entire moving area to obtain a good estimate for translation. We recognize that ∇I can be taken to be a vector of element and line differences ($EDIF$, $LDIF$) if the intensity is available on a discrete grid as is the case in most coding situations. Using linear regression, we get $\hat{\mathbf{D}}$, an estimate of $\mathbf{D}$ as:

$$\hat{\mathbf{D}} = -\left[\sum_{\substack{\text{moving}\\\text{area}}} \nabla I(\mathbf{x}, t)\cdot\nabla I(\mathbf{x}, t)^T\right]^{-1}\left[\sum_{\substack{\text{moving}\\\text{area}}} FDIF(\mathbf{x})\cdot\nabla I(\mathbf{x}, t)\right],$$

which can be written as

$$\hat{\mathbf{D}} = -\begin{bmatrix} \sum EDIF^2(\mathbf{x}) & \sum EDIF(\mathbf{x})\cdot LDIF(\mathbf{x}) \\ \sum EDIF(\mathbf{x})\cdot LDIF(\mathbf{x}) & \sum LDIF^2(\mathbf{x}) \end{bmatrix}^{-1} \cdot \begin{bmatrix} \sum FDIF(\mathbf{x})\cdot EDIF(\mathbf{x}) \\ \sum FDIF(\mathbf{x})\cdot LDIF(\mathbf{x}) \end{bmatrix},$$

where all the summations are over the entire moving area. This can be approximated* by assuming that

$$\sum_{\substack{\text{moving}\\\text{area}}} EDIF(\mathbf{x})\cdot LDIF(\mathbf{x}) = 0$$

and then

$$\hat{\mathbf{D}} = -\begin{bmatrix} \sum FDIF(\mathbf{x})\cdot EDIF(\mathbf{x})/\sum EDIF^2(\mathbf{x}) \\ \sum FDIF(\mathbf{x})\cdot LDIF(\mathbf{x})/\sum LDIF^2(\mathbf{x}) \end{bmatrix}. \tag{4}$$

This can be further approximated* by avoiding the multiplications in the sums as:

$$\hat{\mathbf{D}} = -\begin{bmatrix} \dfrac{\sum FDIF(\mathbf{x})\text{sign}(EDIF(\mathbf{x}))}{\sum |EDIF(\mathbf{x})|} \\[2ex] \dfrac{\sum FDIF(\mathbf{x})\text{sign}(LDIF(\mathbf{x}))}{\sum |LDIF(\mathbf{x})|} \end{bmatrix}, \tag{5}$$

where

$$\text{sign}(z) = \begin{cases} 0, & \text{if } z = 0 \\ \dfrac{z}{|z|}, & \text{otherwise.} \end{cases} \tag{6}$$

This algorithm is identical to one of the algorithms given by Limb and Murphy.[13] Its accuracy may be improved† by several modifications suggested by Limb and Murphy[13] and Cafforio and Rocca.[14]

We suggest another modification which improves the above motion estimator further. First, we note that the above estimates are good as long as $\mathbf{D}$ is small. As $\mathbf{D}$ increases, the quality of the approximation becomes poor. This can be overcome to some extent by linearizing the intensity function around an initial estimate of $\mathbf{D}$. This is possible in a television situation, where there is an estimate of $\mathbf{D}$ for every field. Thus, for the ith field, displacement estimate $\hat{\mathbf{D}}^i$ can be obtained by linearizing the intensity function around the displacement estimate for the $(i-1)$th field. This process results in the following recursion:

$$\hat{\mathbf{D}}^i = \hat{\mathbf{D}}^{i-1} + \mathbf{U}^i, \tag{7}$$

* We make no attempts to justify these two approximations. They are made merely to derive the algorithm given in Ref. 13. Perhaps, instead of the least squares, some other criterion may lead to the same result (i.e., eq. (5)) with fewer assumptions.

† The improvement suggested by Limb (Ref. 13) was simulated for comparison purposes. However, for the type of pictures used in Section 3.3, the performance did not change noticeably by incorporating the improvement.

where $\hat{\mathbf{D}}^{i-1}$ is an initial estimate of $\hat{\mathbf{D}}^{i}$ and $\mathbf{U}^{i}$ is the update of $\hat{\mathbf{D}}^{i-1}$ to make it more accurate, i.e., an estimate of $\mathbf{D} - \hat{\mathbf{D}}^{i-1}$.

We now define the quantity DFD $(\mathbf{x}, \hat{\mathbf{D}}^{i-1})$, called the displaced frame difference, which is analogous to $FDIF(\mathbf{x})$ used in (4) and (5),

$$\text{DFD}(\mathbf{x}, \hat{\mathbf{D}}^{i-1}) = I(\mathbf{x}, t) - I(\mathbf{x} - \hat{\mathbf{D}}^{i-1}, t - \tau) \tag{8}$$

DFD is defined in terms of two quantities: (i) the spatial location $\mathbf{x}$ at which it is evaluated and (ii) the displacement $\hat{\mathbf{D}}^{i-1}$ with which it is evaluated. Obviously, in the case of a two-dimensional grid of discrete samples, an interpolation process would be used to evaluate $I(\mathbf{x} - \hat{\mathbf{D}}^{i-1}, t - \tau)$ for nonintegral values of $\hat{\mathbf{D}}^{i-1}$. As defined, DFD has the property of converging to zero as $\hat{\mathbf{D}}^{i}$ converges to the actual displacement, $\mathbf{D}$, of the image. Following the same steps as were used in the derivation of eq. (5), we get:

$$\begin{aligned}\text{DFD}(\mathbf{x}, \hat{\mathbf{D}}^{i-1}) &= I(\mathbf{x}, t) - I(\mathbf{x} + \mathbf{D} - \hat{\mathbf{D}}^{i-1}, t) \\ &= -(\mathbf{D} - \hat{\mathbf{D}}^{i-1})^{T} \cdot \nabla I(\mathbf{x}, t) + \text{Higher Order Terms.}\end{aligned} \tag{9}$$

Neglecting higher order terms and making approximations similar to the above results in an estimate of $\mathbf{D} - \hat{\mathbf{D}}^{i-1}$ which, when combined with eq. (7), yields:

$$\hat{\mathbf{D}}^{i} = \hat{\mathbf{D}}^{i-1} - \begin{bmatrix} \dfrac{\sum \text{DFD}(\mathbf{x}, \hat{\mathbf{D}}^{i-1})\operatorname{sign}(EDIF(\mathbf{x}))}{\sum |EDIF(\mathbf{x})|} \\[2ex] \dfrac{\sum \text{DFD}(\mathbf{x}, \hat{\mathbf{D}}^{i-1})\operatorname{sign}(LDIF(\mathbf{x}))}{\sum |LDIF(\mathbf{x})|} \end{bmatrix}, \tag{10}$$

where the summations are again carried over the entire moving area.

This may be simplified slightly by noting that if the initial estimate of displacement $\hat{\mathbf{D}}^{i-1}$ has only integral components, then DFD$(\cdot, \cdot)$ can be computed without interpolation. Let $[\mathbf{D}]$ denote an integer approximation to $\mathbf{D}$. This can be obtained either by truncating or rounding both the components of the vector $\mathbf{D}$.

$$\hat{\mathbf{D}}^{i} = [\hat{\mathbf{D}}^{i-1}] - \begin{bmatrix} \dfrac{\sum \text{DFD}(\mathbf{x}, [\hat{\mathbf{D}}^{i-1}]) \cdot \operatorname{sign}(EDIF(\mathbf{x}))}{\sum |EDIF(\mathbf{x})|} \\[2ex] \dfrac{\sum \text{DFD}(\mathbf{x}, [\hat{\mathbf{D}}^{i-1}]) \cdot \operatorname{sign}(LDIF(\mathbf{x}))}{\sum |LDIF(\mathbf{x})|} \end{bmatrix}. \tag{11}$$

We describe the performance of both the above algorithms later in this section.

2.2 Pel-recursive estimation of motion

We mentioned in the introduction that there is an advantage in recursive algorithms which iterate on a pel-by-pel (or on a small block of pels) basis, i.e., they revise their displacement estimate at every moving area pel. Such recursive algorithms overcome, to a large extent, the problems of multiple moving objects, as well as different parts of an object undergoing different displacements, provided the recursion has sufficiently rapid convergence. Since we intend to use the displacement estimator for predictive coding, our algorithm should in some manner attempt to minimize the resulting prediction error. Also, since the prediction error is calculated for transmission anyway, its use in the recursive estimation of displacement does not result in extra computations and is therefore advantageous. Thus, if a pel at location $\mathbf{x}_a$ is predicted with displacement $\hat{\mathbf{D}}^{i-1}$ and intensity $I(\mathbf{x}_a - \hat{\mathbf{D}}^{i-1}, t - \tau)$, resulting in prediction error $\mathrm{DFD}(\mathbf{x}_a, \hat{\mathbf{D}}^{i-1})$, the estimator should try to produce a new estimate, $\hat{\mathbf{D}}^i$, such that $|\mathrm{DFD}(\mathbf{x}_a, \hat{\mathbf{D}}^i)| \leqslant |\mathrm{DFD}(\mathbf{x}_a, \hat{\mathbf{D}}^{i-1})|$. To this end, we attempt to recursively minimize $[\mathrm{DFD}(\mathbf{x}, \hat{\mathbf{D}})]^2$ at each moving area element using a gradient type of approach. For example,

$$\begin{aligned}\hat{\mathbf{D}}^i &= \hat{\mathbf{D}}^{i-1} - (\epsilon/2)\nabla_{\mathbf{D}}[\mathrm{DFD}(\mathbf{x}_a, \hat{\mathbf{D}}^{i-1})]^2 \\ &= \hat{\mathbf{D}}^{i-1} - \epsilon\mathrm{DFD}(\mathbf{x}_a, \hat{\mathbf{D}}^{i-1})\nabla_{\mathbf{D}}\mathrm{DFD}(\mathbf{x}_a, \hat{\mathbf{D}}^{i-1}),\end{aligned}$$

where $\nabla_{\mathbf{D}}$ is the gradient with respect to displacement $\mathbf{D}$ and ϵ is a positive scalar constant. The gradient $\nabla_{\mathbf{D}}$ may be evaluated using the definition of DFD and noting that

$$\nabla_{\mathbf{D}}(\mathrm{DFD}(\mathbf{x}_a, \hat{\mathbf{D}}^{i-1})) = +\nabla I(\mathbf{x}_a - \hat{\mathbf{D}}^{i-1}, t - \tau), \tag{12}$$

where ∇ is the gradient with respect to x. This gives us

$$\hat{\mathbf{D}}^i = \hat{\mathbf{D}}^{i-1} - \epsilon\mathrm{DFD}(\mathbf{x}_a, \hat{\mathbf{D}}^{i-1})\nabla I(\mathbf{x}_a - \hat{\mathbf{D}}^{i-1}, t - \tau), \tag{13}$$

where DFD and ∇I are evaluated by interpolation for nonintegral $\mathbf{D}^{i-1}$. A significant reduction in computation of ∇I is achieved by quantizing $\hat{\mathbf{D}}^{i-1}$ to an integral value. Thus, if $[\hat{\mathbf{D}}^{i-1}]$ represents a rounded or truncated value of each of the components of $\hat{\mathbf{D}}^{i-1}$, then the estimator of eq. (13) can be simplified to

$$\hat{\mathbf{D}}^i = \hat{\mathbf{D}}^{i-1} - \epsilon\mathrm{DFD}(\mathbf{x}_a, \hat{\mathbf{D}}^{i-1})\nabla I(\mathbf{x}_a - [\hat{\mathbf{D}}^{i-1}], t - \tau). \tag{14}$$

It should be pointed out that ∇_D could have been evaluated using (9), resulting in an estimator in which ∇I is evaluated at $(\mathbf{x}_a, t)$ instead of $(\mathbf{x}_a - \hat{\mathbf{D}}^{i-1}, t - \tau)$ as above. This second method implies an assumption regarding the expansion of $I(\cdot, \cdot)$, which may not be valid if $D - \hat{\mathbf{D}}^{i-1}$ is large. Also, there is no difference in the computational complexity if it is assumed that a linear interpolation of $I(\mathbf{x}, t - \tau)$ is used to compute DFD, and the resulting displaced line and element differences are used to define ∇I of eq. (14).

It is interesting to observe that at every iteration we add to our old estimate a vector quantity parallel to the direction of the spatial gradient of image intensity and whose magnitude is proportional to the motion-compensated prediction error. It may be seen from eq. (9) that if the displacement error $(\mathbf{D} - \hat{\mathbf{D}}^{i-1})$ is orthogonal to the intensity gradient ∇I, the displaced frame difference DFD $(\cdot, \cdot)$ is zero, giving a zero update for recursion of eq. (14). This may happen even though the object may have actually moved. However, this is not a failure of the algorithm, but rather is identical to the situation in which an intensity ramp is translated and only motion parallel to the ramp direction (∇I) is perceived. Motion perpendicular to the ramp direction is unobservable, and as such is arbitrary. It is only through the occurrence of edges with differing orientations in real television scenes that convergence of $\hat{\mathbf{D}}^i$ to actual $\mathbf{D}$ is possible.

The quantities involved in the above algorithm are shown in Fig. 1. An initial estimate of the displacement at pel $\mathbf{x}_a$, $\hat{\mathbf{D}}^{i-1}$, is to be updated using (14), yielding $\hat{\mathbf{D}}^i$. Using the initial estimate $\hat{\mathbf{D}}^{i-1}$ and $\mathbf{x}_a$, the samples in the previous frame in the neighborhood of spatial position $\mathbf{x}_a - \hat{\mathbf{D}}^{i-1}$ are located (for example: samples b, c, d, e, and f). The

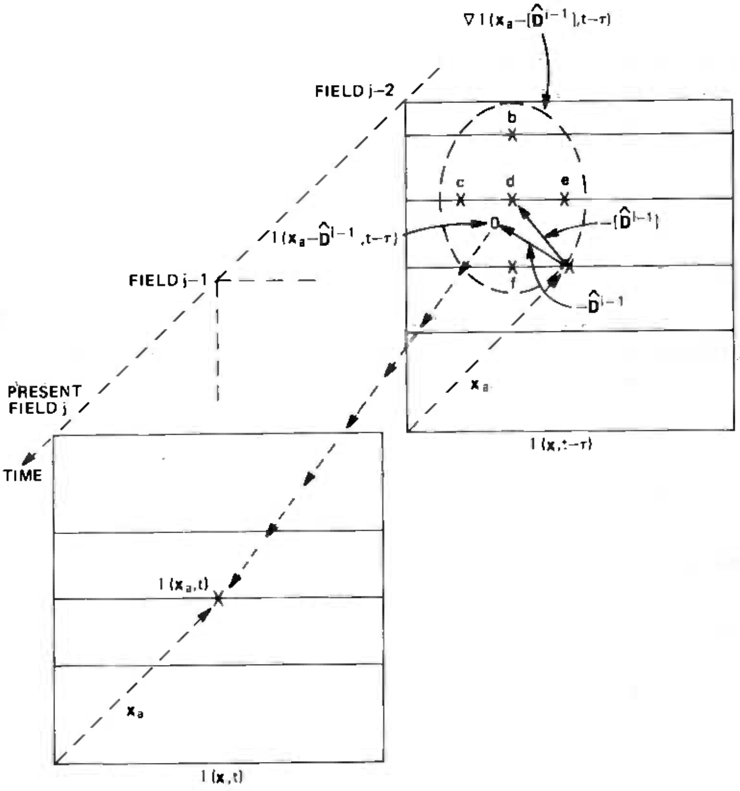

$\nabla I(\mathbf{x}_a - [\hat{\mathbf{D}}^{i-1}], t-\tau)$
FIELD j−2
b
c
d
e
$I(\mathbf{x}_a - \hat{\mathbf{D}}^{i-1}, t-\tau)$
$-[\hat{\mathbf{D}}^{i-1}]$
FIELD j−1
f
$-\hat{\mathbf{D}}^{i-1}$
PRESENT FIELD j
$\mathbf{x}_a$
$I(\mathbf{x}, t-\tau)$
TIME
$I(\mathbf{x}_a, t)$
$\mathbf{x}_a$
$I(\mathbf{x}, t)$

Fig. 1—Recursive motion estimation. Displacement estimate $\hat{\mathbf{D}}^{i-1}$ is updated at pel a. Gradient of intensity, $\nabla I(\mathbf{x}_a - [\hat{\mathbf{D}}^{i-1}], t - \tau)$, is obtained by using intensities at pels b, c, d, e, f in the field $(j - 2)$.

samples of this neighborhood are then used to compute the update term in conjunction with $I(\mathbf{x}_a, t)$. Thus, in Fig. 1, the components of intensity gradient may be approximated by

$$EDIF = (I_e - I_c)/2 \tag{15}$$

$$LDIF = (I_b - I_f)/2. \tag{16}$$

Similarly, $I(\mathbf{x}_a - \hat{\mathbf{D}}^{i-1}, t - \tau)$ may be approximated by a two-dimensional linear interpolation using the intensities of the neighborhood. In the next section, we present several simplifications of this basic algorithm and adapt it for frame-to-frame coding. Recursive algorithms, using a model of the video process for motion estimation in a different context, are described in Ref. 17.

2.3 Motion estimator performance

In this section, we give simulation results for the first two motion estimators, algorithm I (eq. (5)) and algorithm II (eq. (11)). Obviously, the performance depends upon the type of scene and, even then, it is not clear how to measure the performance. We have used two types of scenes. The first is produced synthetically using the following formula:

$$I(\mathbf{x}, t) = \begin{cases} 127 \quad \text{if} \quad \|\mathbf{R}\| > 100 \\ 127\ (1 + e^{-.05\|\mathbf{R}\|} \cos(0.2 \cdot \pi \cdot \|\mathbf{R}\|), \quad \text{otherwise,} \end{cases} \tag{17}$$

where $\mathbf{R} = \mathbf{x} - (\mathbf{x}_o + \mathbf{D}t)$, 127 is the background intensity (on a scale of 0 to 255), $\mathbf{x}_o$ is the location of the center of the moving pattern at $t = 0$, $\mathbf{D}$ is the displacement of the pattern per frame (i.e., velocity), and $|\cdot|$ denotes the Euclidean norm. This formula produces a series of alternating light and dark concentric rings with exponentially decreasing radial intensity variation. This pattern was chosen because it contains a distribution of edges with different direction and height. $I(\mathbf{x}, t)$ was sampled both in time t and space $\mathbf{x}$ to produce a set of four frame sequences with strictly horizontal translation ranging from 0.5 to 6.0 pels per frame.

The second type of scene is also a collection of four frame sequences containing an object approximately in horizontal translation. These are the same as shown in Fig. 1b of Ref. 15. They contain a mannequin's head which was moved at various nominal speeds from 0.4 to 4.7 pels per frame.

The first type of scene was chosen so that an exact velocity was known and, therefore, the performance of motion estimators could be evaluated rather easily by measuring the deviation from this known velocity. A second measure of performance was obtained by computing the "match entropy" of the elements in the moving area of each field.

That is, using the displacement estimate $\hat{\mathbf{D}}^i$, obtained from two consecutive fields, the entropy of DFD$(\mathbf{x}, \hat{\mathbf{D}}^i)$ was computed using a linear two-dimensional interpolation over the moving area elements of frame $I(\mathbf{x}, t)$. This quantity is similar to the entropy of the prediction error; however, some future information is used in its calculation. Due to the presence of shadows and nonuniform illumination, the second type of scene does not possess a precisely defined velocity, and therefore only match entropy was used to evaluate the motion estimators for this type of scene.

In all the simulations of the results of algorithms I and II of this section and the next section, the moving area elements were determined by an algorithm similar to that described in Ref. 13. Likewise, the definition of $EDIF(\mathbf{x})$ and $LDIF(\mathbf{x})$ used in the simulations of algorithm I may be found in Ref. 13. For $EDIF(\mathbf{x})$, this involved the averaging of the element differences at $I(\mathbf{x}, t)$ and $I(\mathbf{x}, t - \tau)$. $LDIF(\mathbf{x})$ was computed in a similar manner. To extend this definition for use in algorithm II, the corresponding differences at $I(\mathbf{x}, t)$ and $I(\mathbf{x} - [\hat{\mathbf{D}}^{i-1}], t - \tau)$ were averaged.

The performance of algorithms I and II is given in Figs. 2a and 2b for the synthetic scenes. As seen in Fig. 2a, the error in the estimated velocity* using algorithm II is considerably smaller than that of algorithm I, especially at higher velocities. The peak observed in the estimates of algorithm I is perhaps due to the insensitivity of the algorithms to a per-frame shift equal to the period (10.0 units) of the synthetic moving image. The initial estimate of displacement for algorithm II was assumed to be 0. It is interesting to note that the curve of algorithm I is approximately the same as that of algorithm II after the first iteration; but after only three iterations, algorithm II converges to its curve in Fig. 2a. In Fig. 2b, the match entropies resulting from the estimates of Fig. 2a are shown. Again, the superiority of algorithm II over algorithm I is clearly seen. Also for comparison, we have included the entropies of $FDIF(\mathbf{x})$ of picture elements in the moving area. As expected, both algorithms I and II show significant improvements compared to frame differences. These conclusions for synthetic pictures remained generally unchanged when the direction of motion was changed and when random frame-to-frame noise of −40 dB (s/n ratio) was added to the scenes.

Performance of algorithms I and II for the second type of scene is given in Fig. 3. Here, since we do not have available an exact velocity, we only give the match entropy at the various nominal horizontal velocities used to create the scenes. While both algorithms result in match entropies considerably smaller than the frame difference en-

* Average of last four out of six estimates.

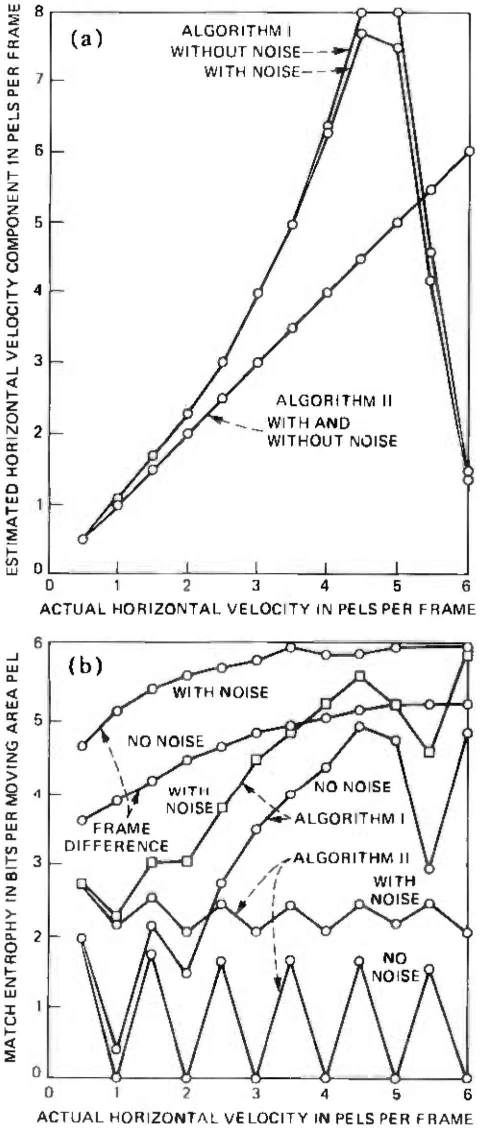


Fig. 2—Performance of two motion estimators which obtain one displacement estimate per field on synthetic pictures. Algorithm II results in considerable improvement over algorithm I. Zero entropy indicates that, in addition to a perfect estimate, there was no interpolation error in evaluating the prediction error. Estimated vertical velocity components (in lines/frame) were ± 0.3 for algorithm I and ± 0.02 for algorithm II.

tropy, we see that the performance of the two appears about the same. This result is perhaps due to the averaging of the displacement components over a large block (i.e., the moving area of an entire field). It should be mentioned that the performance of both the algorithms

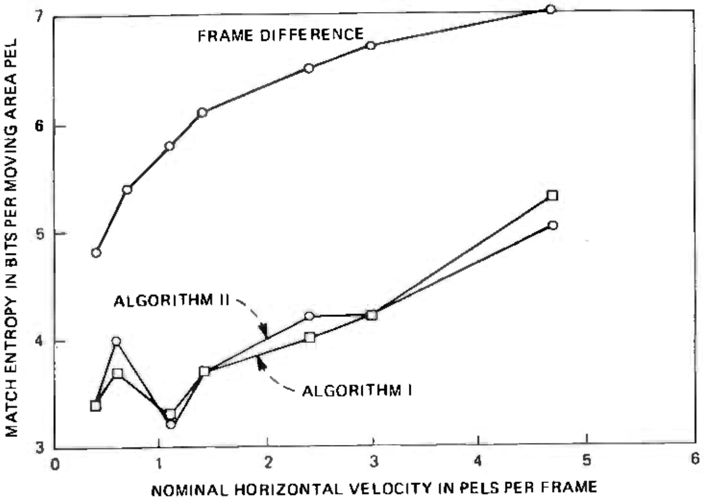


Fig. 3—Performance of two motion estimators which obtain one displacement estimate per field on moving mannequin.

can be improved by separating the uncovered background, as suggested by Cafforio and Rocca.[14]

III. CODER PERFORMANCE USING RECURSIVE DISPLACEMENT ESTIMATION

In the previous section, we developed motion estimators and discussed the performance of estimators which obtain one velocity estimate per field. We used scenes which had only one object, moving with a nearly uniform translational displacement. However, such scenes are unrealistic and, therefore, estimators which can dynamically adjust to motion of objects are more desirable. In this section, we first describe our recursive estimator in more detail and then evaluate its performance on scenes that are much more realistic. We then modify our basic estimator so that it can be incorporated in a frame-to-frame encoder and evaluate the coder performance. We note that we have paid little attention to a very important facet of frame-to-frame coders: resolution control using the contents of the buffer. It is important that motion estimation does not suffer immensely in lower resolution modes of the coder. Although some of these issues will be considered in part II,[16] more realistic performance evaluation can only be done using a hardware coder working on real scenes.

The first scene, called Judy, consists of 64 frames (2:1 interlaced fields) of 256 × 256 samples each, obtained at 30 times a second and sampled at Nyquist rate from a video signal of 1 MHz bandwidth, and contains head-and-shoulders view of a person engaged in a rather active conversation. The portion of a frame classified as moving area varies from 15 to 51 percent. Also, the motion is not translational, and different parts of the scene move differently (such as lips, eyes, and head). Four frames of this scene are shown in Fig. 4.

The second scene, called Mike and Nadine, consists of 64 frames with the same resolution as the scene Judy and contains a panned full body view of two people briskly walking around each other on a set with severe nonuniform illumination. The percentage of a frame classified as moving area varied from 92 to 96 percent. Four frames from this sequence are shown in Fig. 5.

3.1 Basic estimator performance

The basic recursive estimator consists of the following: the displacement estimate is updated at each moving area picture element using eq. (14), i.e.,

$$\hat{\mathbf{D}}^{i} = \hat{\mathbf{D}}^{i-1} - \frac{1}{1024} \cdot \mathrm{DFD}(\mathbf{x}_j, \hat{\mathbf{D}}^{i-1}) \cdot \nabla I(\mathbf{x}_j - [\hat{\mathbf{D}}^{i-1}], t - \tau), \qquad (18)$$

where $\hat{\mathbf{D}}^{i}$ is the estimated displacement of moving area pel $\mathbf{x}_j$, $\hat{\mathbf{D}}^{i-1}$ is the last estimate formed prior to pel $\mathbf{x}_j$ during the pel-by-pel, line-by-line (interlaced raster scan order) iteration through the moving area, ∇I is the spatial gradient of the intensity, approximated by *EDIF* and *LDIF* defined in (15) and (16), and $[\hat{\mathbf{D}}^{i-1}]$ denotes rounded $\hat{\mathbf{D}}^{i-1}$. A two-dimensional linear interpolation is used to evaluate DFD (interpolation is discussed in detail in Section 3.3.3). Instead of using the previous frame for the intensity $I(\mathbf{x}, t - \tau)$, we have used the previous field. Relative advantages of this choice are discussed later. Also, both the horizontal and vertical components in the displacement error estimate (i.e., the second term of the right-hand side of eq. (18)) are clipped at a magnitude of ($^{8}\!/_{128}$), so that the displacement from pel to pel is not allowed to change by more than $^{1}\!/_{16}$ pel/field and $^{1}\!/_{16}$ line/field. This avoids the possibility of rapid oscillations in displacement due to noise. The accuracy used in computation of displacement and interpolation is $^{1}\!/_{128}$ pel (or line) per field.

In all the simulations of this section, with the exception of the results of algorithm I appearing in Fig. 7, the moving area picture elements were determined by the following rule: pel z (Fig. 6) is classified as a moving area pel if either

$$(i) |FDIF(z)| > T_2$$

or

$$(ii) |FDIF(z)| > T_1$$

and

$$|FDIF| \text{ at a, b, c, or d} > T_1$$

and

$$|FDIF| \text{ at A, B, C, or D} > T_1,$$

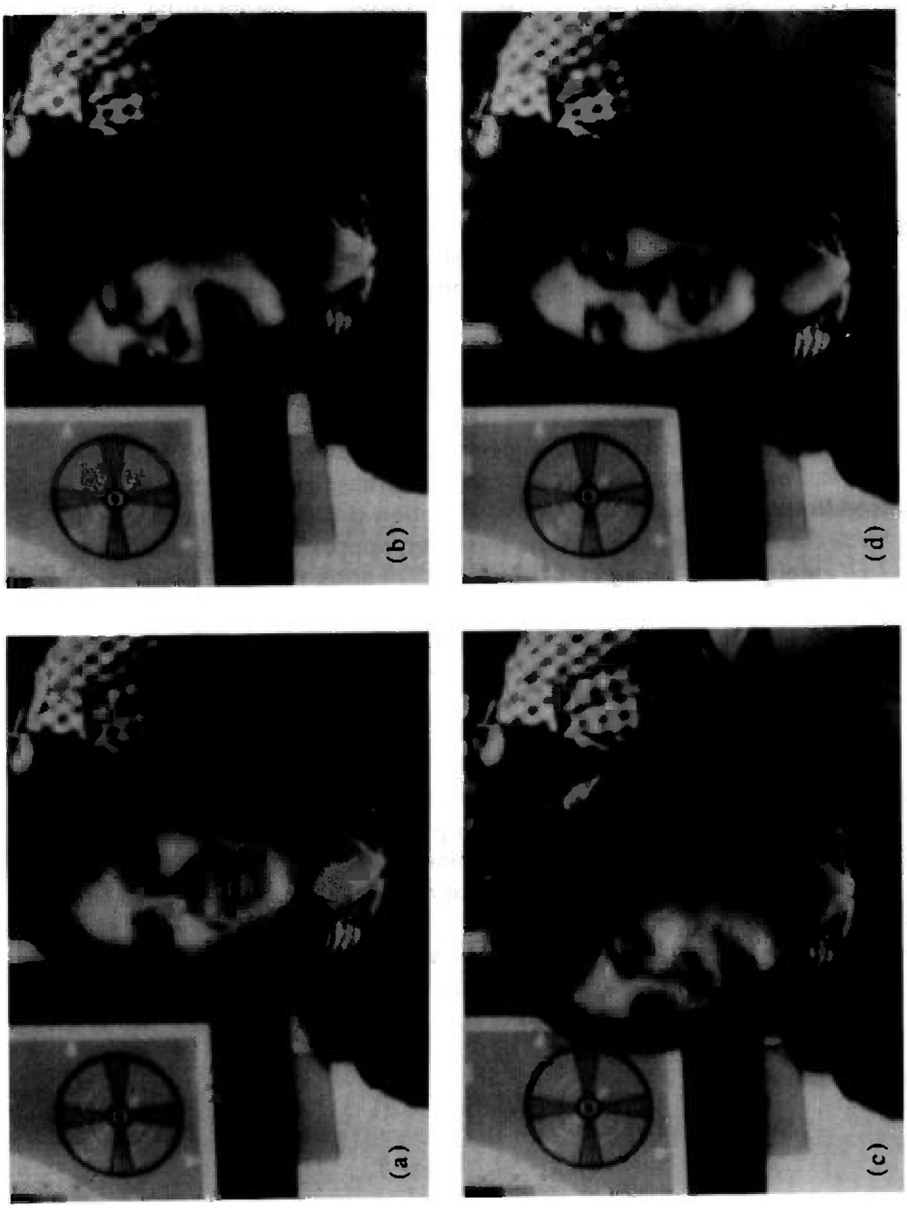


Fig. 4—Four frames of the scene Judy.

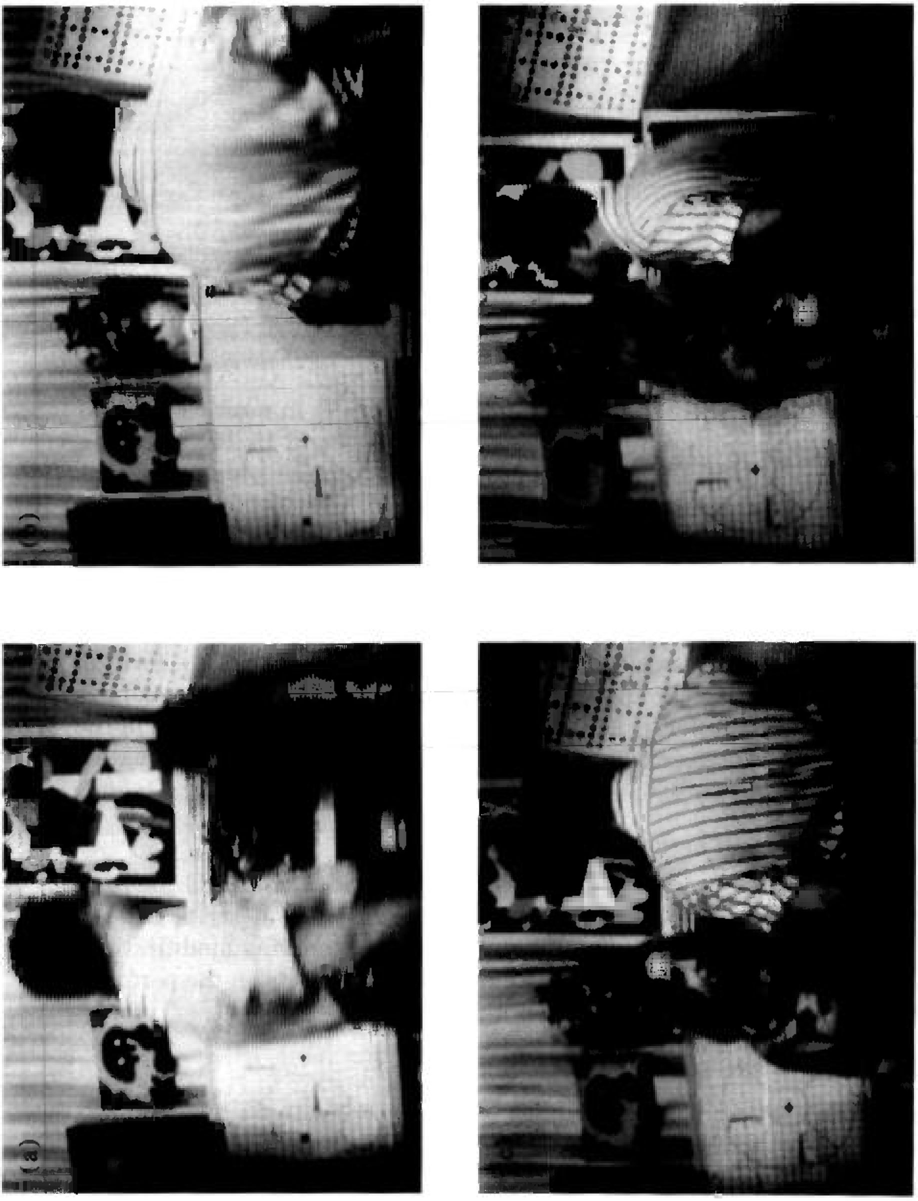


Fig. 5—Four frames of the scene Mike and Nadine.

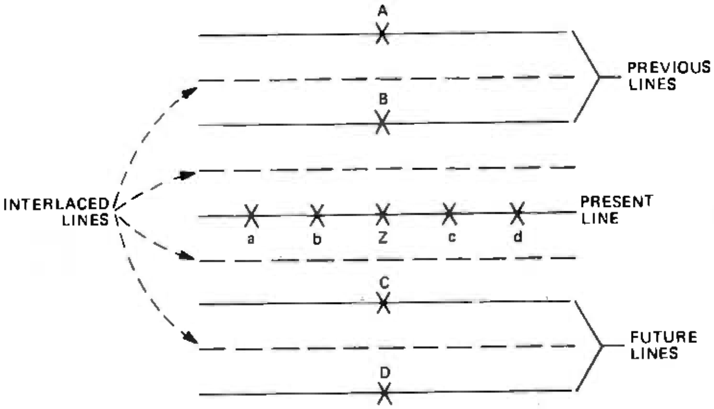


Fig. 6—Configuration of pels used in the moving area segmentor.

where T_1 and T_2 are two thresholds, and $T_2 \geqslant T_1$ in most cases. This segmenter is quite similar to the one in Refs. 4 and 15. It overcomes, to a large extent, the effects of frame-to-frame noise which otherwise produce a large number of isolated moving area elements. For the basic estimator, the moving area was segmented with thresholds $T_1 = 4$ and $T_2 = 255$ (on a scale of 0 to 255).

The performance of this basic estimator is given in Fig. 7 for the sequence Judy. Also for comparison, we have included the entropies of the moving area frame differences and the match entropy of algorithm I of Section II. On the average, match entropy for the recursive estimator was lower by about 1.4 bits/moving area pel than that of the frame differences, while algorithm I only resulted in approximately half of this decrease.*

3.2 Basic coder performance

In order to incorporate the estimator as a part of a coder, several other choices have to be made. In this section, we describe a basic coder and its performance. The next section contains modifications and simplifications of the basic coder and their effects on the performance. The basic coder consists of the following:

3.2.1 Displacement estimator

In the predictive coder, since the displacement estimate is not transmitted to the receiver, it has to be derived from the previously encoded and transmitted data. We derive a displacement estimate for a previously transmitted pel and use it for computation of the predic-

* Results in Fig. 7 for algorithm I were obtained by using a slightly different moving area segmenter given in Ref. 13.

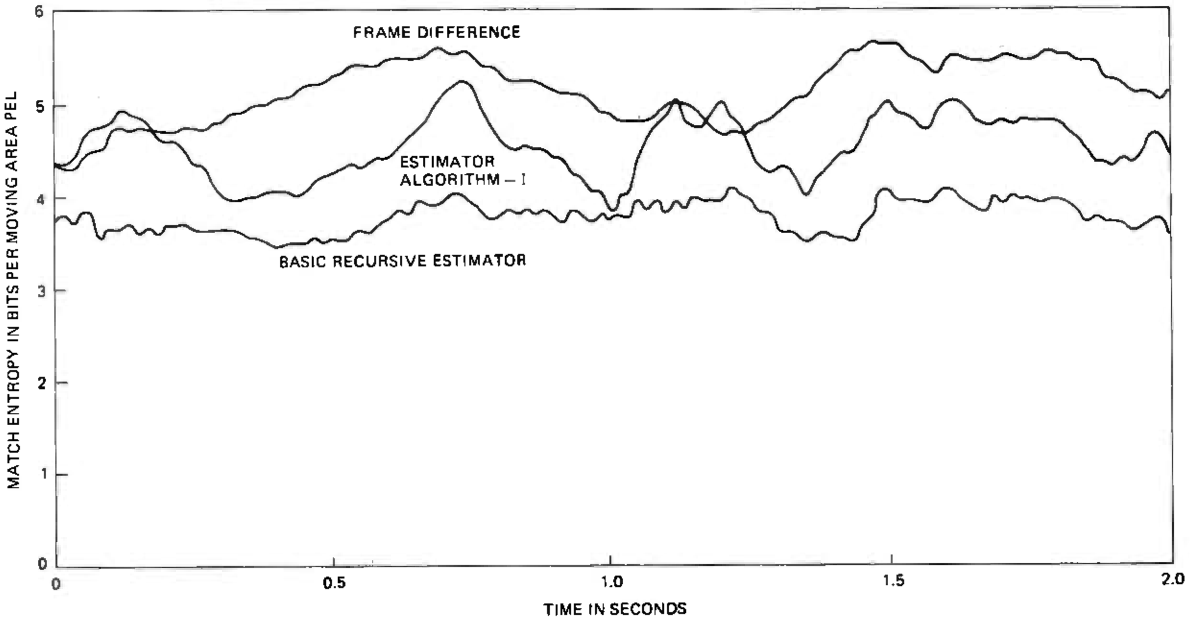


Fig. 7—Performance of basic recursive estimator on the scene Judy. Compared to frame difference and the estimator algorithm I, which obtains one displacement estimate per field, the recursive estimator results in a lower and relatively constant entropy.

tion of the present pel. The two natural choices for the relative location of the displacement estimate are the pel above and the pel to the left of the predicted pel. We have chosen the previous line (i.e., the pel above in the same field), since it relieves hardware timing constraints that would have otherwise resulted. If the previous line pel is not a moving area pel, we use the initial displacement estimate that would have been used by the above pel if it were moving. Such use of displacement estimates for prediction is shown in Fig. 8. We see that several elements (e.g., elements $j + 1, j + 2, \cdots, j + 4$ of the present line) may be predicted with the same displacement estimate if the corresponding elements of the previous line are not moving area elements. Other details of the estimator remain the same as those given in Section 3.1 for the basic estimator.

3.2.2 Segmentation

Every field was first divided into two segments: moving area and background, using the segmentation strategy described in Section 3.1. For a good picture quality, the thresholds T_1 and T_2 were chosen to be 1 and 3 (on a scale of 0 to 255), respectively. Moving area was further divided into two segments: (*i*) compensable regions, where motion-compensated prediction is adequate and, therefore, no update information need be sent; and (*ii*) uncompensable regions which require an update (i.e., transmission of prediction error). This segmentation of the moving area was performed using the following rule:

(*i*) If the magnitude of the motion-compensated prediction error for a moving area pel is greater than 3 (out of 255), then it is called uncompensable.

(*ii*) To reduce the occurrence of isolated compensable pels, compensable pel runs of length 1, 2, and 3 between uncompensable pels were bridged by calling them uncompensable.

This segmentation is shown in Fig. 9.

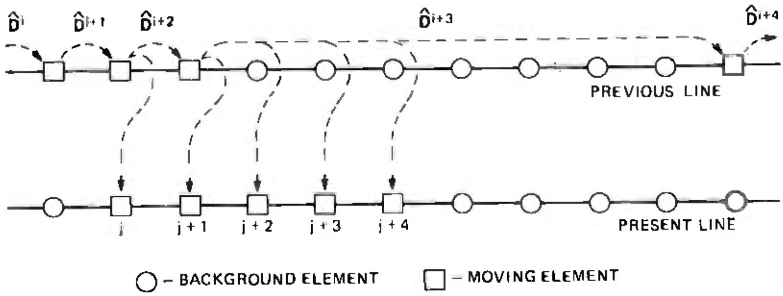


Fig. 8—Use of displacement estimates $(\hat{\mathbf{D}}^i)$ in prediction process. Displacement estimates are formed at every moving area pel in the same order as the scanning process. Displacement estimate of the previous line element is used for prediction of the present line picture elements.

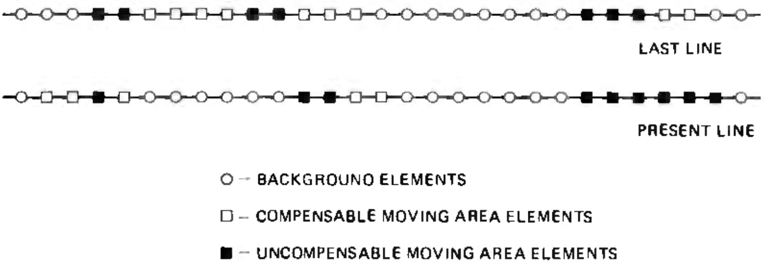

LAST LINE
PRESENT LINE
○ – BACKGROUND ELEMENTS
□ – COMPENSABLE MOVING AREA ELEMENTS
■ – UNCOMPENSABLE MOVING AREA ELEMENTS

Fig. 9—Motion-compensated coder segmentation. Elements whose motion-compensated prediction error is lower than a threshold are called compensable and are not transmitted.

DECISION LEVELS	1,7,13,21,30,39,49,60,71,82,95,108,121,134,147,160,173,255
REPRESENTATIVE LEVELS	0,4,10,17,26,35,44,55,66,77,87,102,115,128,141,154,167,179

Fig. 10—Quantizer for basic coder. Only the positive side of the symmetric quantizer is shown.

3.2.3 Quantization

The prediction error was quantized using a 35-level symmetric quantizer shown in Fig. 10. Other quantizers were also investigated. The relative performance of the coder did not depend heavily on the quantizer. However, the picture quality degraded as would be expected by using a very coarse quantizer. The quantizer of Fig. 10 was chosen since it gave good picture quality, although the quantization error was clearly visible to a trained eye.

The performance of this basic coder was evaluated by computing the following quantities: (*i*) the entropy of the prediction errors of uncompensable pels, and the entropy of the run lengths defining the (*ii*) background pels, (*iii*) compensable pels, and (*iv*) uncompensable pels. The last three quantities are the addressing costs associated with not sending the prediction errors for certain pels. Figure 11 shows the overall bit rate associated with frame difference conditional replenishment encoding and motion-compensated encoding for both of our test sequences. Conditional replenishment encoding was done using the quantizer of Fig. 10 and moving area segmentation similar to that of the motion-compensated coder. The quality of pictures for both cases appeared to be approximately the same. Two characteristics of the motion-compensated coding are worth noting: (*i*) the average bits/field is reduced by about 44 percent; (*ii*) more importantly, during the times when large amounts of frame difference data are generated due to either rapid movement or a high percentage of moving area, the motion-compensated coder does significantly better. This type of behavior may help to decrease the occurrence of temporal overload

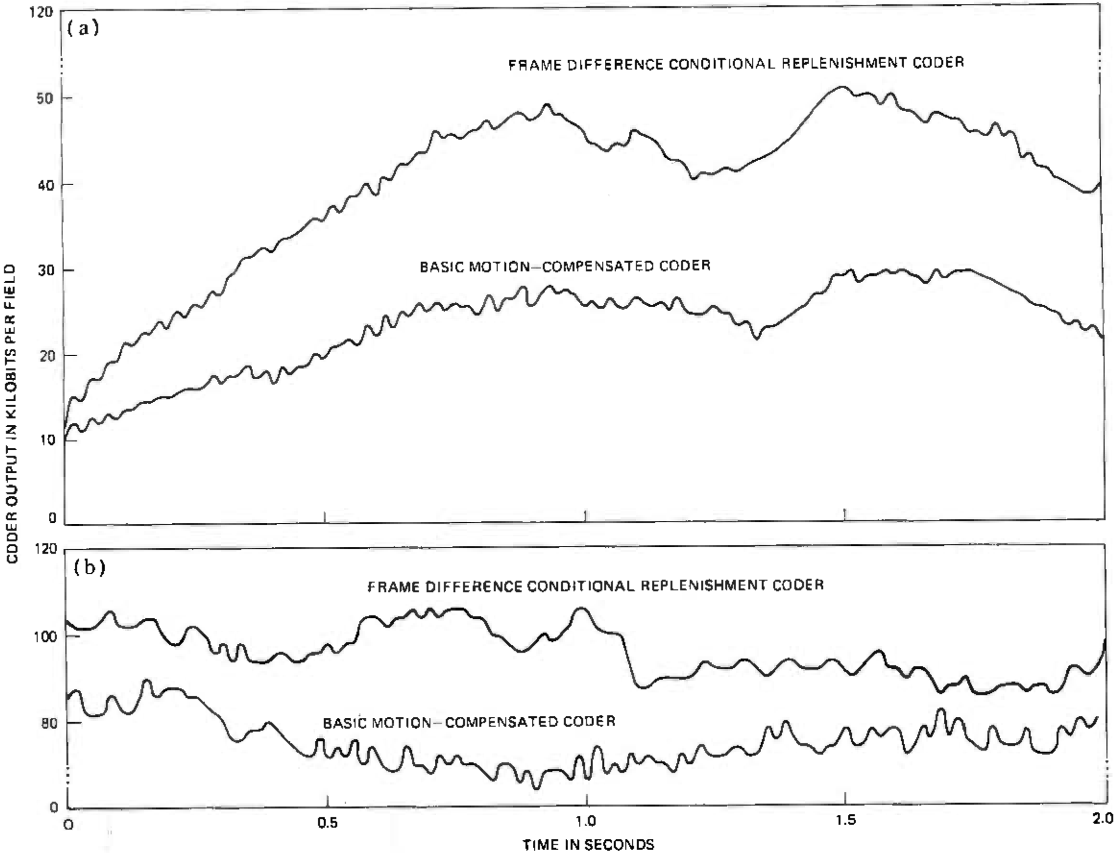


Fig. 11—Performance of basic motion-compensated coder and its comparison with a conditional replenishment coder for (a) the scene Judy and for (b) the scene Mike and Nadine.

associated with most conditional replenishment coders. Part of the reason for such a significant reduction in bit rate due to movement compensation is seen from Figs. 12 and 13 which show the results for conditional replenishment and motion-compensated coding, respectively. In these figures, the pel intensity is proportional to the bits required to code the prediction error for that pel. Uncompensable pels form a very small portion of the moving area and their prediction error is also small. For the scene Mike and Nadine, the output of each encoder is more than twice that obtained for Judy. This increase is mainly due to the panning of the camera. For this scene, intended as a severe test for the encoders, motion-compensated encoding resulted in a 22-percent reduction in average encoded bits.

Figure 14 shows a breakdown of the bits required for transmission of addresses and prediction error for scene Judy. It is interesting to note that, on the average, for the conditional replenishment coder the addressing bits comprise only about 15 to 30 percent of the total transmitted bits, but in the movement-compensated coder, since the magnitude of the prediction error is decreased, the addressing bits account for a much larger fraction of the total bits (50 to 65 percent). More efficient addressing methods are therefore desirable with motion compensation.

3.3 Variation in coder structure

In this section, we simplify the coder and evaluate the effect of various modifications on the coder performance.

3.3.1 Coder prediction

It is known[15] that, in conditional replenishment encoders, instead of using a frame differential prediction, line or element difference of the frame difference can be used with advantage. Thus in Fig. 8, for picture element $\mathbf{x}_{j+3}$, the conditional replenishment coder would send $[FDIF(\mathbf{x}_{j+3}) - FDIF(\mathbf{x}_{j+2})]$ as an element difference of frame difference prediction error. Similar modifications can be made to the motion-compensated coder. Here we transmit line or element differences of the displaced frame differences. Thus for picture element $\mathbf{x}_{j+3}$, we transmit $[\mathrm{DFD}(\mathbf{x}_{j+3}, \mathbf{D}^{i+3}) - \mathrm{DFD}(\mathbf{x}_{j+2}, \mathbf{D}^{i+3})]$ as an element difference of displaced frame difference prediction error. In cases such as pel $\mathbf{x}_j$ of Fig. 8, where the previous element is not in the moving area, their FD as well as DFD is assumed to be zero. The performance of this predictor modification is given in Fig. 15. As seen from this figure, element differencing improves both the frame differential and motion-compensated coders by about 5 to 15 percent.

3.3.2 Addressing of moving area

In motion-compensated coding, the addressing information takes up a large fraction of the total transmitted bits. We have therefore used

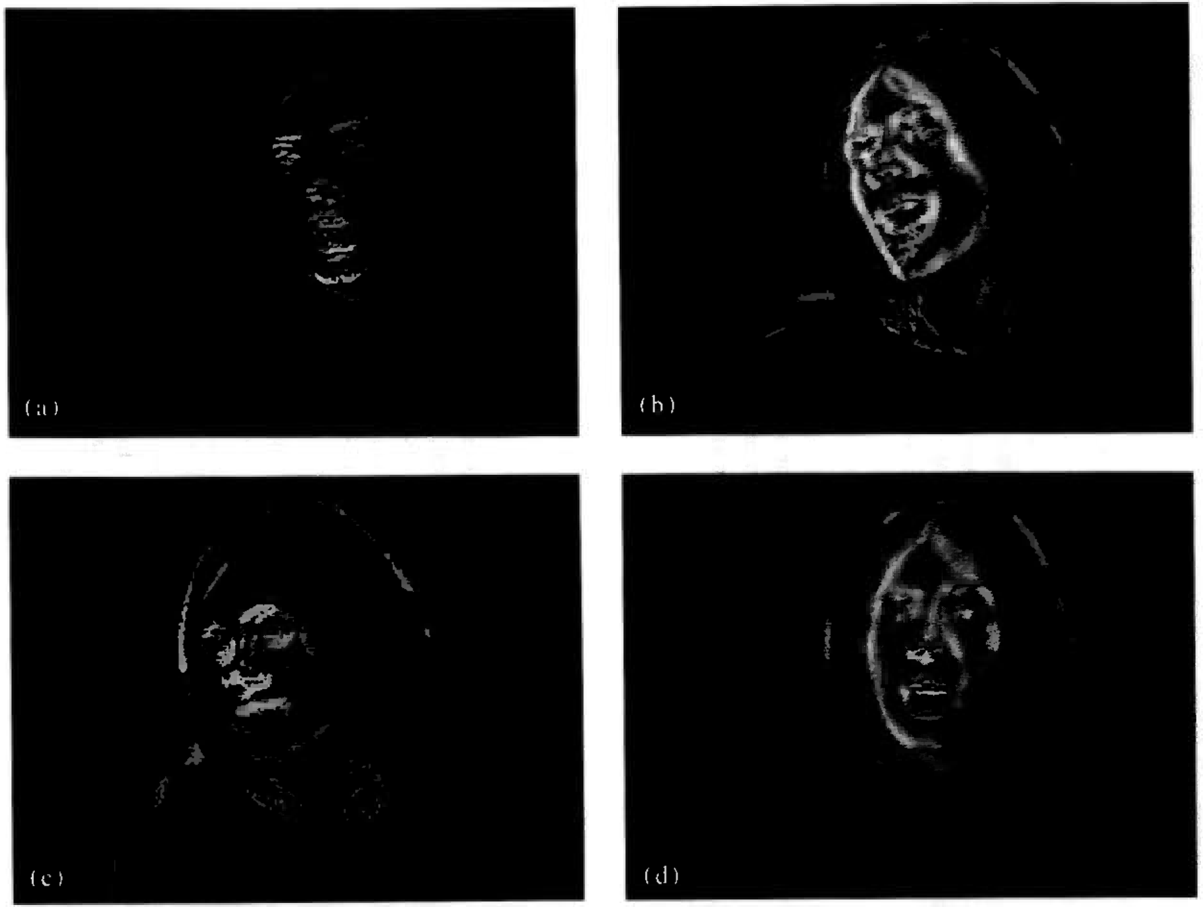


Fig. 12—Transmitted amplitude information for a conditional replenishment coder. Intensity of a picture element is proportional to the bits required to code the prediction error using frame difference prediction for four frames of scene Judy (Fig. 4). Addressing information is not included.

Fig. 13—Transmitted amplitude information for a motion-compensated coder. Intensity of a pel is proportional to the bits required to code the prediction error using the basic coder of Section 3.2 for four frames of scene Judy (Fig. 4). Addressing information is not included.

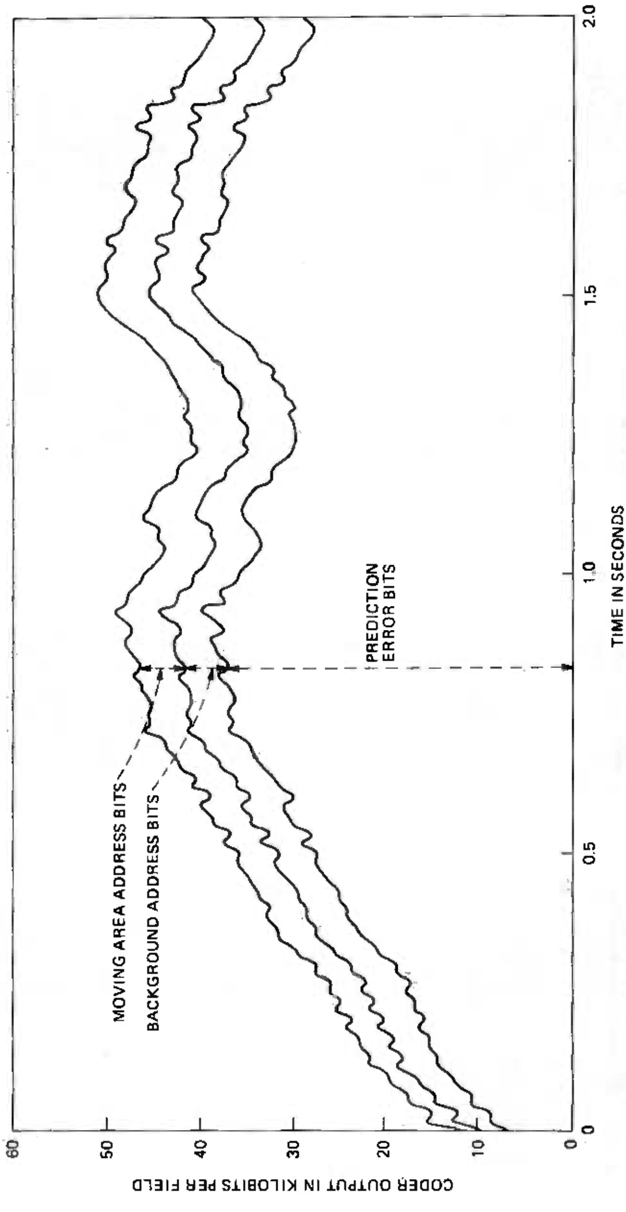


Fig. 14a—Components of conditional replenishment coder output. Approximately 80 percent of output is devoted to prediction error.

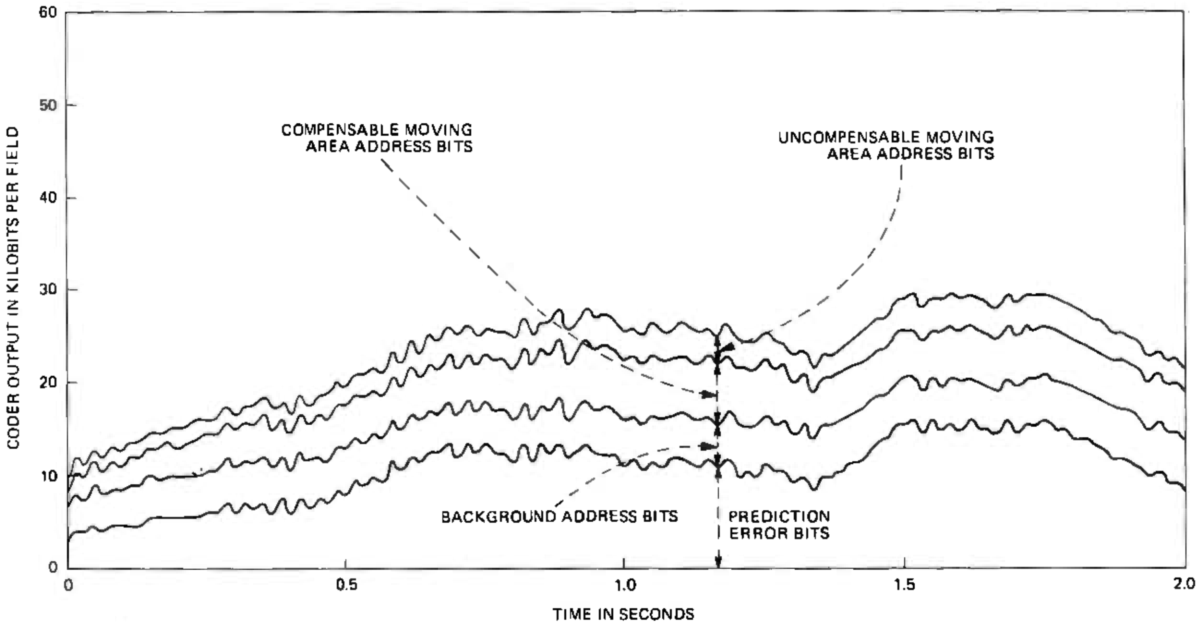


Fig. 14b—Components of motion-compensated coder output. Note that addressing information accounts for more than 50 percent of the coder's output, whereas the conditional replenishment output is mostly prediction error.

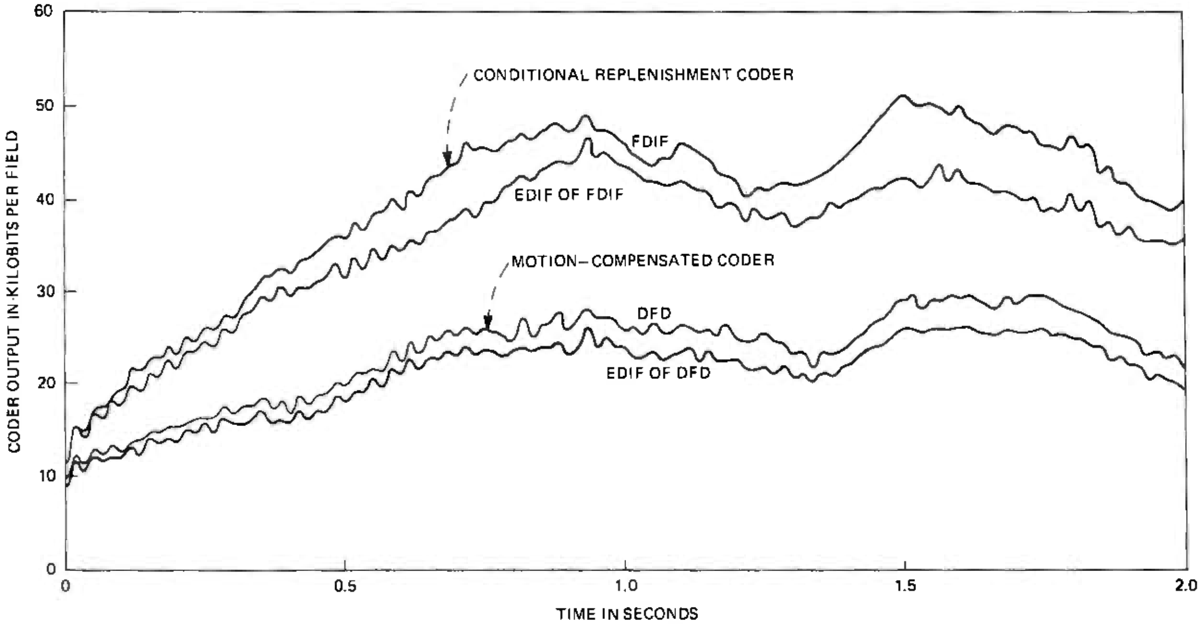


Fig. 15—Performance with predictor modification for sequence Judy. Element difference of either frame difference or displaced frame difference lower the entropy.

some known[18] addressing techniques and evaluated their effects on the total bit rate. As in Ref. 18, three techniques for addressing the moving area were tried. In the first technique, the beginning of each moving area was given an absolute address (8 bits) with respect to the beginning of the scan line, and the end of moving area cluster was given a special flag word different from all the prediction error codes. This is similar to the technique used in Ref. 2. The second technique[18] was to address the beginning of each cluster of moving area with respect to a similar cluster in the previous line as long as the magnitude of this differential address was less than ±8; otherwise, the cluster was addressed with respect to the beginning of the line. Maximum differential address of 8 was chosen after experiments with 2, 4, 16, and 32. The final technique is to address all the clusters as one-dimensional run lengths, as in our previous sections. The performance of these three addressing schemes for both the conditional replenishment and motion-compensated coders is given in Fig. 16 for sequence Judy. As expected, absolute addressing requires more bits than either differential or run-length addressing. Run-length addressing is the most efficient and decreases the overall bit rate by about 10 percent compared to the absolute addressing in a motion-compensated coder.

It should be noted that, to precisely evaluate the addressing cost associated with motion-compensated coding, it is not adequate to use only the run-length entropies of the three types of segments. Knowing the type of the last run, the receiver must be told, in some fashion, which of the two possible types the next run is. This information has been included in the simulations of Fig. 16. Comparing the motion-compensated coder using run-length addressing for moving area as in Fig. 16 with the results shown in Fig. 11 which does not include the next segment type of information, we see that this information accounts for about 8 percent of the bits.

3.3.3 *Interpolation*

The motion-compensated coder discussed above uses previous field intensities for interpolation rather than previous frame intensities. Use of previous field has two advantages: (*i*) the displacement values are generally smaller, since fields occur at every 1/60th of a second, whereas frames occur at 1/30th of a second, and (*ii*) in a hardware coder, since only a certain number of elements from the previous field or frame would be made available for interpolation with the same number of adjacent available elements, almost twice the amount of motion can be accommodated. On the other hand, one disadvantage of using the previous field intensities is that, for a perfectly horizontal motion, intensities have to be obtained from the previous field by interpolation (due to the interlace) which introduces error in the computation of DFD. Another factor, which appears to be a disadvan-

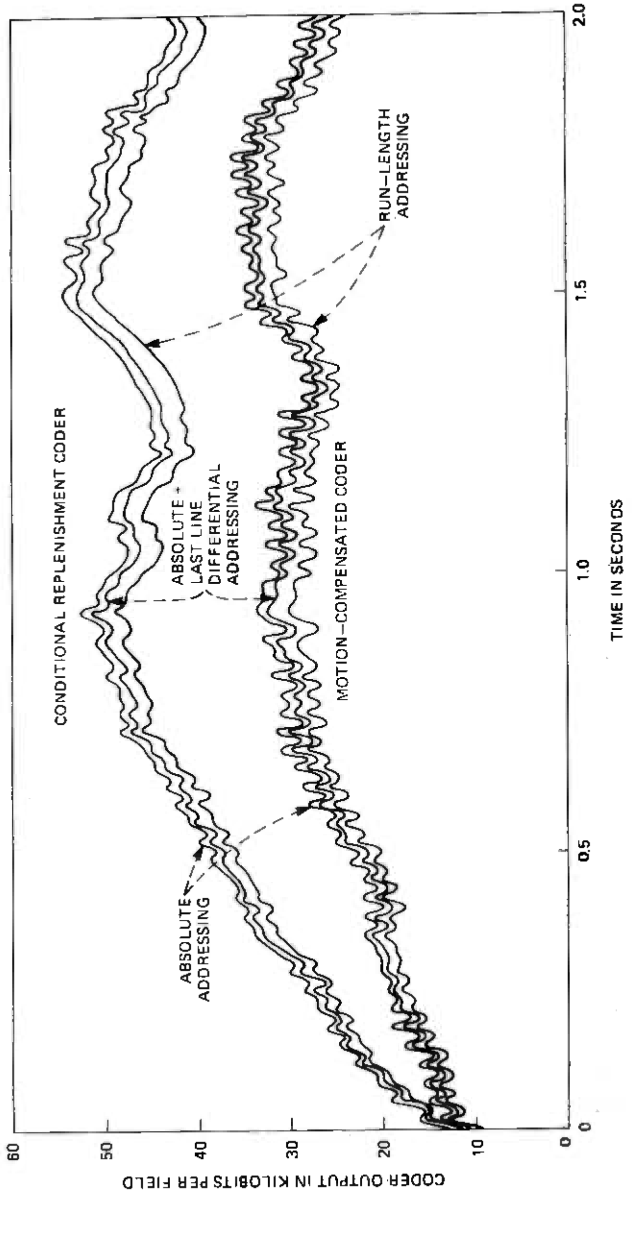


Fig. 16—Performance of moving area addressing schemes. Run-length addressing does better than absolute or differential addressing.

tage but which can be overcome, is the use of an alternate field dropping coder mode. To relieve a buffer overflow condition caused by a peak in coder output, it is advantageous to drop alternate fields as is done in conditional replenishment coders,[4,5] however; the previous field is then no longer available for motion estimation. In any case,we experimented with both the previous field and frame intensities and found that the use of previous field intensities resulted in entropies of unquantized moving area prediction errors which were 5 to 15 percent lower than those of previous frame intensities.

Of the many interpolation techniques possible, we restricted our attention to linear interpolation since our goal was a coder that could be implemented in real time. Thus, given the four surrounding pels as in Fig. 17, if the displacement $\mathbf{D}$ is written as the sum of an integral part $\mathbf{D}^I$, and a fractional part $\mathbf{D}^f$ with component magnitudes D_1^f and D_2^f, then the intensity can be interpolated by a standard two-dimensional linear interpolation as:

$$I = (1 - D_2^f)[(1 - D_1^f)I_D + D_1^f I_C] + D_2^f[(1 - D_1^f)I_B + D_1^f I_A], \tag{19}$$

where I is the interpolated intensity. It should be noted that eq. (19) has been used for all simulations previously described in this paper. Since this interpolation formula requires a large number of time-consuming multiplications, we tried another approach. We first select the three nearest (in the sense of the Euclidean norm) neighboring pels out of the four (e.g., pels B, C, and D in Fig. 17). This may be done by rounding $\mathbf{D}$ to form $\mathbf{D}^I$, thus locating the nearest neighbor, pel D, which is taken to be the corner of the three, then testing the sign of the rounding errors to locate the other two pels. The following formula is then used for interpolation:

$$I = I_D + D_1^f(I_C - I_D) + D_2^f(I_B - I_D). \tag{20}$$

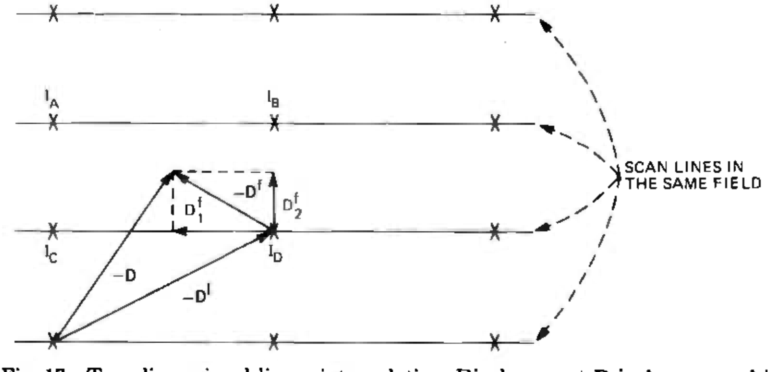


Fig. 17—Two-dimensional linear interpolation. Displacement $\mathbf{D}$ is decomposed into integral part $\mathbf{D}^I$ and fractional part $\mathbf{D}^f$.

Simulations using sequence Judy, comparing eqs. (19) and (20), showed very little differences in entropy, and since the interpolation of eq. (20) is significantly simpler, it is more suitable for a real-time implementation.

The precision with which the interpolation computation is performed was also varied. In particular, we used precisions of 2^{-n}; $n = 1, 2 \cdots, 7$ pel (or line) per field for displacement estimates, and found, as expected, that the average bits/field increased as the precision was decreased. The average increase in bits/field relative to a precision of 1/128 pel/field was as follows: negligible increase for precisions greater than 1/8 pel/field, 2 to 3 percent for 1/8 pel/field, 6 to 10 percent for 1/4 pel/field, and 15 to 25 percent for 1/2 pel/field. Thus, it appears that for interpolation, a resolution of 1/8 pel/field does an adequate job with much lower complexity compared to resolution of 1/128 pel/field. It should be noted that the recursion of eq. (18) was calculated by using a precision which is higher than the one used for the calculation of interpolation.

3.3.4 Estimator parameters

In eq. (18), the ϵ was taken to be 1/1024 and the update term was clipped to (1/16) pels/field. Both these quantities were varied over a wide range. Performance due to such variations is given in Fig. 18. This shows the robustness of the estimator. Of course, increase of both ϵ and the clipping level to a very high value results in a noisy estimator, but it gives the ability to adjust quickly to rapid changes in motion. On the other hand, a small value of ϵ allows us to converge to a finer value of displacement, thereby allowing low prediction error. A compromise between these two conflicting goals appears to be the values of ϵ and clipping level used in (18).

We mentioned earlier that the recursion of eq. (18) was calculated at a resolution of 1/128 pel/field. Simulation results on Judy indicate that, when the interpolation is calculated with resolution of 1/8 pel/field, the recursion of eq. (18) does not need to be calculated at resolutions higher than 1/16 pel/field, i.e, there is no significant improvement in performance by increasing the resolution beyond 1/16 pel/field in the calculation of displacement. To make the simulations realistic, all the computations were performed using integer arithmetic (on the computer).

Within our investigation into the effect of precision, the following simplification of the update term was simulated:

$$\hat{\mathbf{D}}^{i} = \hat{\mathbf{D}}^{i-1} - \epsilon \cdot \text{sign}(\text{DFD}(\mathbf{x}_j, \hat{\mathbf{D}}^{i-1})) \cdot \text{sign}(\nabla I(\mathbf{x}_j, [\hat{\mathbf{D}}^{i-1}])), \quad (21)$$

where sign of a vector quantity is the vector of signs of its components. The sign function, defined by (6), avoids the multiplication necessary for computation of the update term. Thus the update of each displace-

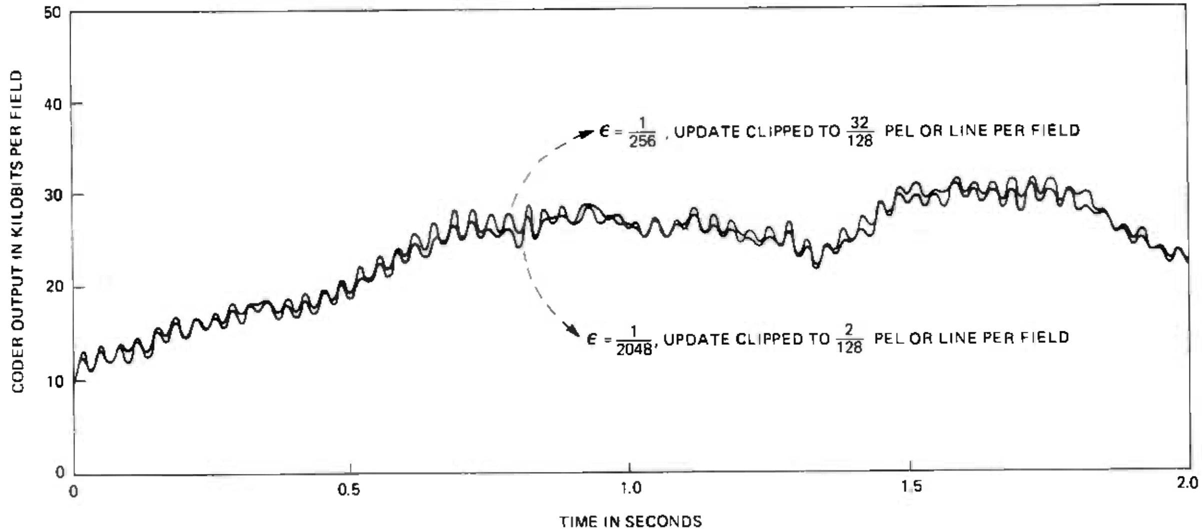


Fig. 18—Performance change due to the variation of rate and the maximum level of displacement update in motion estimator for sequence Judy. Results show robustness to variation of these parameters.

ment component from one picture element to the next consists of only three possibilities; 0 or ±ϵ. The performance using such a simplification is given in Fig. 19. It is clear from this figure that the performance in terms of bits/field is virtually unaffected by using the simplified update term of eq. (21).

Earlier we mentioned that we computed the displacement from the previous line of video and used it for computation of the motion-compensated prediction of the present line. This was done so that the displacement computation would have enough time to finish in a real-time encoder. It is possible, at least in simulations, to use the displacement of the previous moving area element in the same line for computation of the motion-compensated prediction of the present pel. This variation improves the performance of the encoder by about 6 to 10 percent in terms of bits/field.

The last simplification we attempted was in the calculation of the intensity gradient. Our hope was to use the same element and line differences which would be used in the computation of the DFD, so that this computation could be shared. Figure 20 shows three possible configurations of picture elements and the corresponding intensity gradients that were simulated. In the first and second configuration, the elements are those used for the interpolation of eqs. (19) and (20), respectively, while configuration III replaces pel D of configuration II with two pels further away. Coder performance using the three interpolation configurations was similar, and since the second configuration is the simplest, it is more suitable for hardware implementation.

3.4 Combined effects of simplifications

We have so far evaluated the changes in coder performance due to each individual simplification. In this section, we evaluate the combined effects of the following simplifications:

(*i*) The moving area is specified with absolute addresses and the next segment-type information is transmitted as in part 2 of the last section.

(*ii*) The sign $(\cdot) \cdot$sign $(\cdot)$ estimator of eq. (21) is used for displacement estimation.

(*iii*) The three-point interpolation of eq. (20) is used for intensity prediction and displacement estimation at a precision of ⅛ pel/field.

(*iv*) The line and element differences of interpolation eq. (20) are used as the intensity gradient components of eq. (21).

As seen from the simulated performance of the coder in Fig. 21, the modifications have had only a minor effect on the encoded bits/field. The simplified basic coder represents what we believe to be a practical algorithm for real-time implementation. A simplified block diagram of this coder appears in Fig. 22. The inner loop, which comprises the

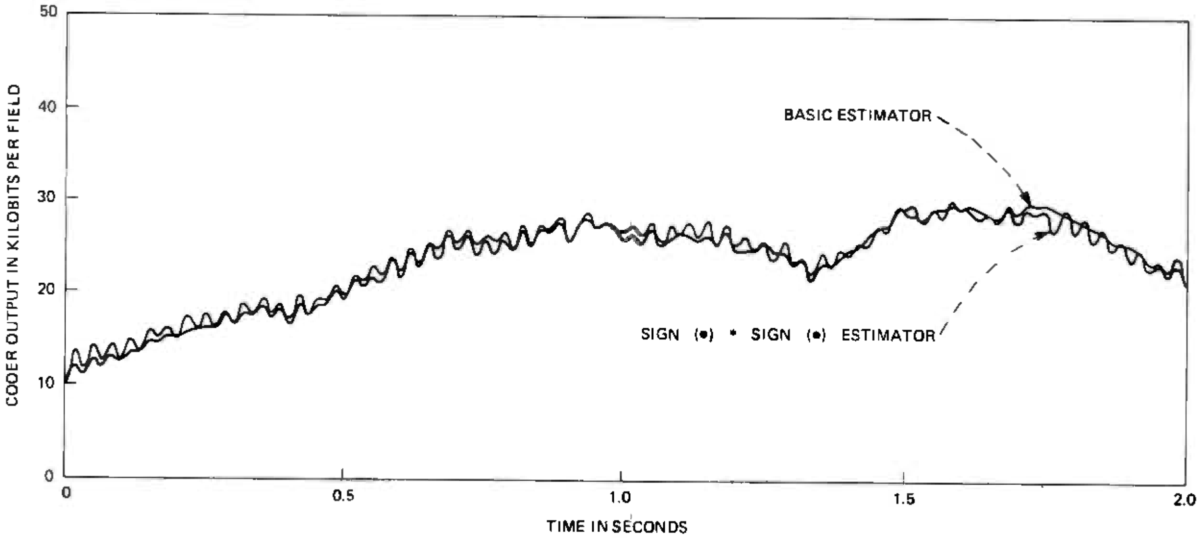


Fig. 19—Coder performance using a sign (·) * sign (·) motion estimator update for sequence Judy.

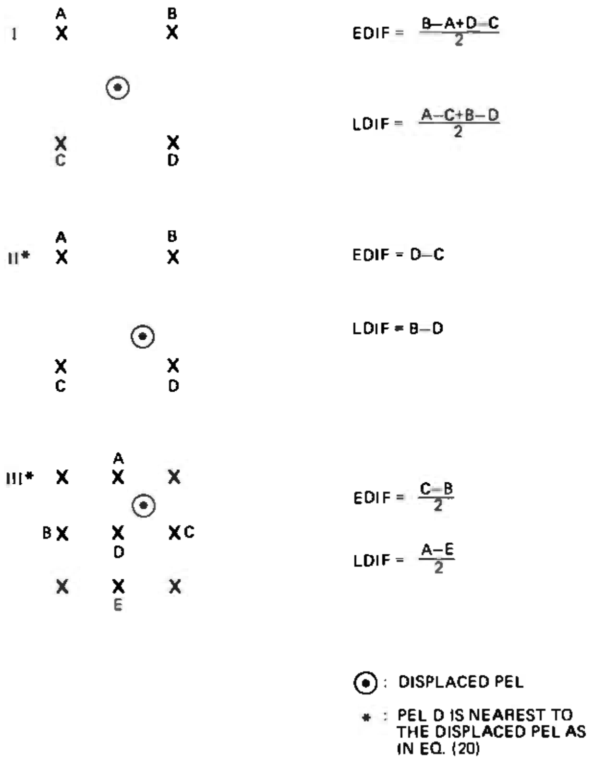


Fig. 20—Possible pel configurations for computing intensity gradient.

displacement estimator, attempts to match interpolated intensities of the last field with those of the last line by adjusting the displacement. The outer loop uses the displacement estimates to predict the video input with interpolated last field intensities. Based on frame difference, the segmentor limits displacement estimation (switch A) and motion-compensated prediction (switch B) to the moving areas. The segmentor also controls the selection of errors for transmission with switch C. Hardware for controlling the coder's output and matching it to a constant rate channel has been omitted in this representation. However, it has to be included for any real-time implementation.

IV. DISCUSSIONS AND CONCLUSIONS

We have developed recursive estimators in this paper and simplified them so that a real-time implementation would be possible. Some characteristics of our final estimator are that it needs computation of interpolation intensities at resolution no more than ⅛ pel/field (which can be done without multiplication). We believe that this is in the realm of present-day circuit speeds. Also, by the very nature of the

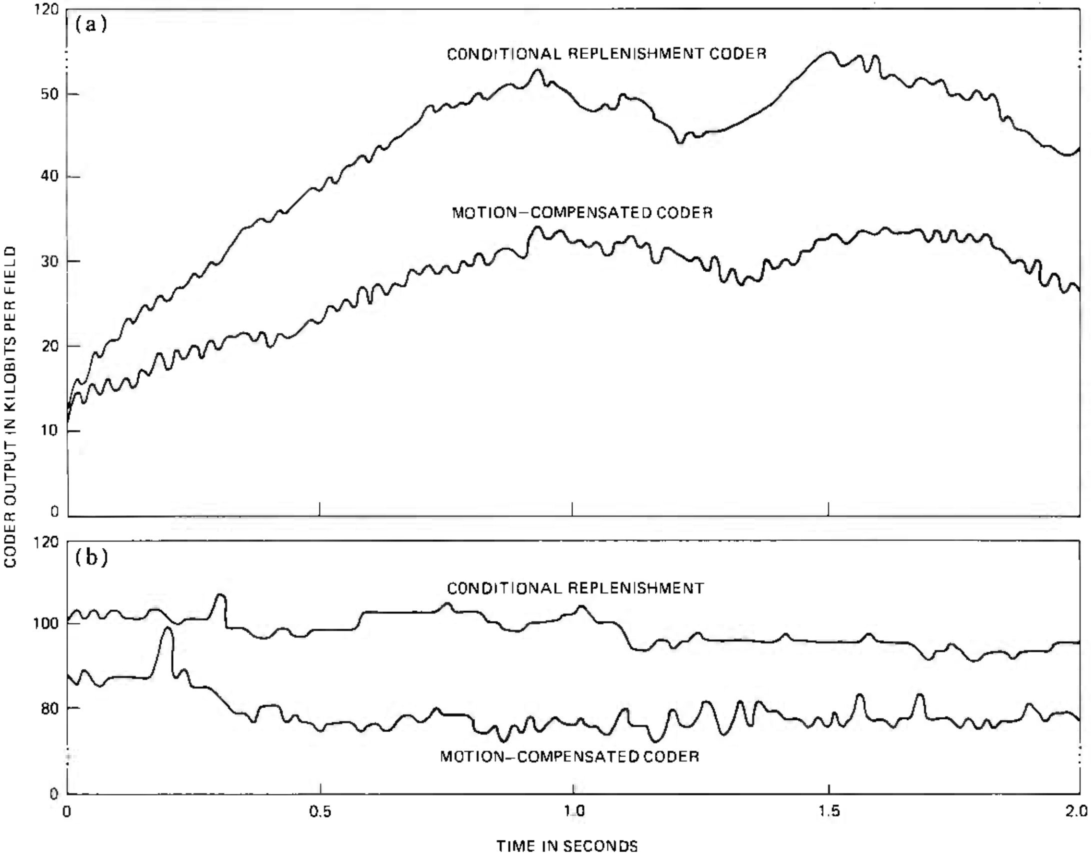


Fig. 21—Simplified basic coder performance for (a) scene Judy and (b) scene Mike and Nadine. Compared to the basic coder results of Fig. 11, the increase in output bit rate is mainly due to absolute addressing of moving area elements and the inclusion of the next segment type of information.

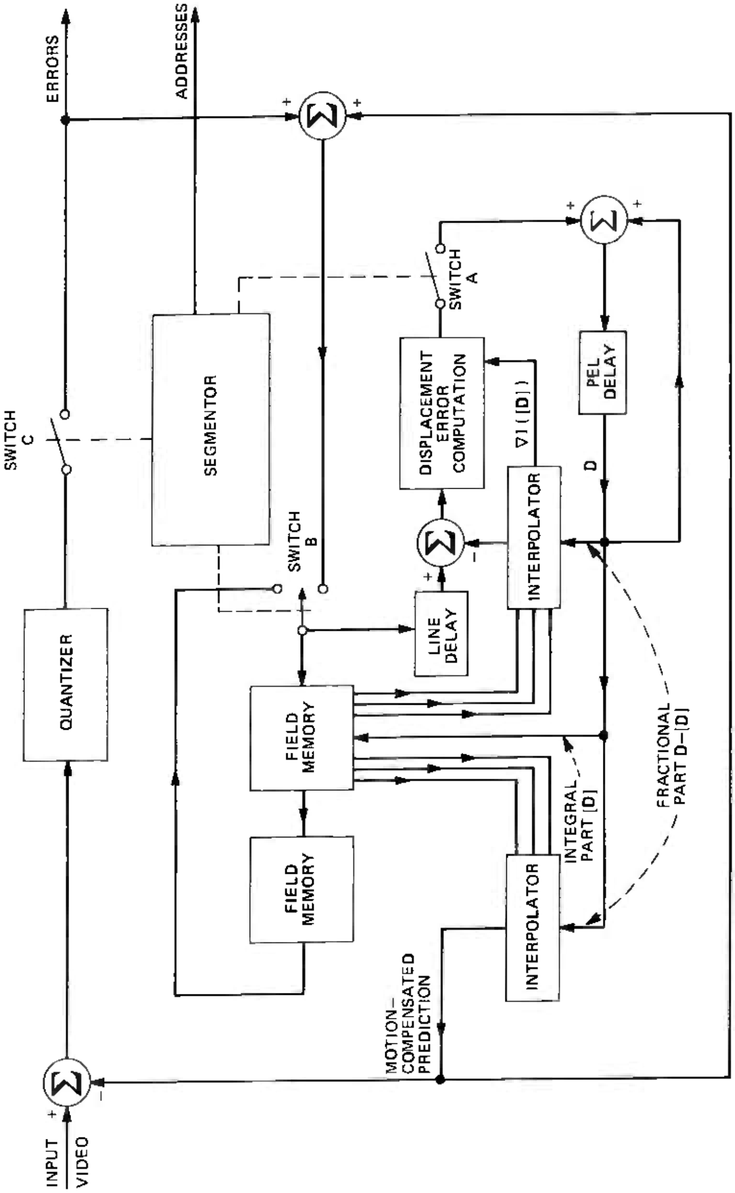


Fig. 22—A simplified block diagram of motion-compensated coder.

recursive estimation, we do not need the assumption of uniform translational motion of a single object, which appears to be necessary for most of the heretofore known estimates. Performance of the simple estimator on the sequence Judy shows that a 30-to-50 percent improvement in bits/field is possible compared to frame-difference conditional replenishment. We also evaluated the performance of our simple estimator on a more difficult scene: Mike and Nadine. Both conditional replenishment and motion compensation generate significantly more data for this scene than for Judy due to the high percentage of moving area pels. However, the motion-compensated coding decreases the transmitted bits/field by approximately 22 percent. Thus the improvement is somewhat lower for this scene. Some of the modifications suggested in Part II[16] improve the performance for this type of scene.

V. ACKNOWLEDGMENTS

The authors would like to thank Ed Bowen for his help in creating the scene Mike and Nadine and Mike Lukacs and Nadine Pinkerton for being subjects of this scene.

APPENDIX

In this appendix, we show that the gradient algorithm of eq. (13) which attempts to minimize DFD$(\cdot\,,\cdot)$ converges, under suitable conditions, to true displacement. As in Section II, let us assume that an object is moving with pure translation in the field of view of the camera. Neglecting the uncovered background and assuming a constant and uniform displacement per unit time, we get

$$I(\mathbf{x}, t) = I(\mathbf{x} - \mathbf{D}, t - \tau), \tag{22}$$

where $\mathbf{D}$ is the true displacement and τ is the frame time. The pel-recursive motion estimation algorithm of eq. (13) can be written as:

$$\hat{\mathbf{D}}^i = \hat{\mathbf{D}}^{i-1} - \epsilon\text{DFD}(\mathbf{x}_a, \hat{\mathbf{D}}^{i-1})\nabla I(\mathbf{x}_a - \hat{\mathbf{D}}^{i-1}, t - \tau). \tag{23}$$

Using the definition of DFD$(\cdot\,,\cdot)$ from eq. (8), w get

$$\hat{\mathbf{D}}^i = \hat{\mathbf{D}}^{i-1} - \epsilon\{I(\mathbf{x}_a, t) - I(\mathbf{x}_a - \hat{\mathbf{D}}^{i-1}, t - \tau)\}\nabla I(\mathbf{x}_a - \hat{\mathbf{D}}^{i-1}, t - \tau). \tag{24}$$

Substituting for $I(\mathbf{x}_a, t)$ from (22),

$$\hat{\mathbf{D}}^i = \hat{\mathbf{D}}^{i-1} - \epsilon\{I(\mathbf{x}_a - \mathbf{D}, t - \tau) - I(\mathbf{x}_a - \hat{\mathbf{D}}^{i-1}, t - \tau)\}\cdot\nabla I(\mathbf{x}_a - \hat{\mathbf{D}}^{i-1}, t - \tau). \tag{25}$$

Using Taylor's series, we can write

$$I(\mathbf{x}_a - \mathbf{D}, t - \tau) - I(\mathbf{x}_a - \hat{\mathbf{D}}^{i-1}, t - \tau) = (\hat{\mathbf{D}}^{i-1} - \mathbf{D})^T\cdot\nabla I(\mathbf{x}_a - \hat{\mathbf{D}}^{i-1}, t - \tau) + \text{higher order terms in } (\hat{\mathbf{D}}^{i-1} - \mathbf{D}).$$

Substituting in (25),

$$\begin{aligned}\hat{\mathbf{D}} &= \hat{\mathbf{D}}^{i-1} - \epsilon\{(\hat{\mathbf{D}}^{i-1} - \mathbf{D})^T \nabla I(\mathbf{x}_a - \hat{\mathbf{D}}^{i-1}, t-\tau)\}\, \nabla I(\mathbf{x}_a - \hat{\mathbf{D}}^{i-1}, t-\tau) \\ &\quad + \text{higher order terms in } (\hat{\mathbf{D}}^{i-1} - \mathbf{D}). \\ &= \hat{\mathbf{D}}^{i-1} - \epsilon\{\nabla I(\mathbf{x}_a - \hat{\mathbf{D}}^{i-1}, t-\tau)\cdot \nabla I(\mathbf{x}_a - \hat{\mathbf{D}}^{i-1}, t-\tau)^T\}\,(\hat{\mathbf{D}}^{i-1} - \mathbf{D}) \\ &\quad + \text{higher order terms in } (\hat{\mathbf{D}}^{i-1} - \mathbf{D}).\end{aligned}$$

Subtracting **D** from both sides, we get

$$\begin{aligned}(\hat{\mathbf{D}}^{i} - \mathbf{D}) = (\hat{\mathbf{D}}^{i-1} - \mathbf{D}) &- \epsilon\{\nabla I(\mathbf{x}_a - \hat{\mathbf{D}}^{i-1}, t-\tau) \\ &\cdot \nabla I(\mathbf{x}_a - \hat{\mathbf{D}}^{i-1}, t-\tau)^T\}\,(\hat{\mathbf{D}}^{i-1} - \mathbf{D}) \\ &+ \text{higher order terms in } (\hat{\mathbf{D}}^{i-1} - \mathbf{D}).\end{aligned}$$

Neglecting higher order terms,* for small $(\hat{\mathbf{D}}^{i-1} - \mathbf{D})$,

$$\begin{aligned}&(\hat{\mathbf{D}}^{i} - \mathbf{D}) \\ &\quad = [J - \epsilon\{\nabla I(\mathbf{x}_a - \hat{\mathbf{D}}^{i-1}, t-\tau)\cdot \nabla I(\mathbf{x}_a - \hat{\mathbf{D}}^{i-1}, t-\tau)^T\}](\hat{\mathbf{D}}^{i-1} - \mathbf{D})\end{aligned}$$

where J is an identity matrix of appropriate size. We now take statistical averages of both sides, assume statistical independence of the two right-hand terms,† and then apply Schwartz inequality to get

$$\begin{aligned}&\|\bar{\hat{\mathbf{D}}}^{i} - \bar{\mathbf{D}}\| \\ &\quad \leq \|J - \epsilon\{\overline{\nabla I(\mathbf{x}_a - \hat{\mathbf{D}}^{i-1}, t-\tau)\nabla I(\mathbf{x}_a - \hat{\mathbf{D}}^{i-1}, t-\tau)^T}\}\| \\ &\qquad\qquad \cdot \|\bar{\hat{\mathbf{D}}}^{i-1} - \bar{\mathbf{D}}\| \qquad (26)\end{aligned}$$

where a bar on top denotes a statistical average. This can be written as:

$$\|\bar{\hat{\mathbf{D}}}^{i} - \bar{\mathbf{D}}\| \leq |1 - \epsilon\lambda_{\max}| \cdot \|\bar{\hat{\mathbf{D}}}^{i-1} - \bar{\mathbf{D}}\|, \qquad (27)$$

where $\lambda_{\max}$ is the maximum eigenvalue of the positive semidefinite, symmetric matrix $\overline{\nabla I(\mathbf{x}_a - \hat{\mathbf{D}}^{i-1}, t-\tau)\nabla I(\mathbf{x}_a - \hat{\mathbf{D}}^{i-1}, t-\tau)^T}$. For convergence of the algorithm, we need

$$|1 - \epsilon\lambda_{\max}| < 1,$$

i.e.,

$$\frac{2}{\lambda_{\max}} > \epsilon > 0. \qquad (28)$$

Since the maximum eigenvalue is hard to compute, and since it is upper bounded by the trace, we get

* This may be a severe assumption in certain cases. In such cases, the ability of the algorithm to converge may depend upon the goodness of the initial estimate $\hat{D}$.

† This is traditionally done in stochastic gradient algorithms. See, for example, Refs. 19 to 21.

$$\frac{2}{tr[\overline{\nabla I(\mathbf{x}_a - \hat{\mathbf{D}}^{i-1}, t-\tau)\nabla I(\mathbf{x}_a - \hat{\mathbf{D}}^{i-1}, t-\tau)^T}]} > \epsilon > 0. \tag{29}$$

But since

$$tr[\overline{\nabla I(\mathbf{x}_a - \hat{\mathbf{D}}^{i-1}, t-\tau)\nabla I(\mathbf{x}_a - \hat{\mathbf{D}}^{i-1}, t-\tau)^T}] \cong \frac{1}{N}\sum_{i=1}^{N} EDIF_i^2 + LDIF_i^2,$$

we get the following condition for convergence

$$\frac{2N}{\sum_{i=1}^{N} EDIF_i^2 + LDIF_i^2} > \epsilon > 0, \tag{30}$$

where the summations are carried over the entire moving area containing N pels.

REFERENCES

1. F. W. Mounts, "A Video Encoding System Using Conditional Picture-Element Replenishment," B.S.T.J., *48*, No. 7 (September 1969), pp. 2545–2554.
2. J. C. Candy, M. A. Franke, B. G. Haskell, and F. W. Mounts, "Transmitting Television as Clusters of Frame-to-Frame Differences," B.S.T.J., *50*, No. 6 (July–August 1971), pp. 1889–1917.
3. J. O. Limb, R. F. W. Pease, and K. A. Walsh, "Combining Intraframe and Frame-to-Frame Coding for Television," B.S.T.J., *53*, No. 6 (August 1974), pp. 1137–1173.
4. B. G. Haskell, P. L. Gordon, R. L. Schmidt, and J. V. Scattaglia, "Interframe Coding of 525-Line, Monochrome Television at 1.5 MBS," IEEE Trans. Commun., *COM-25*, No. 10 (October 1977), pp. 1339–1344.
5. T. Ishiguro, K. Iinuma, Y. Iijima, T. Koga, S. Azami, and T. Mune, "Composite Interframe Coding of NTSC Color Television Signals," 1976 National Telecommunications Conference Record (Dallas, Texas, November, 1976), *1*, pp. 6.4-1 to 6.4-5.
6. B. G. Haskell, F. W. Mounts, and J. C. Candy, "Interframe Coding of Videotelephone Pictures," Proc. IEEE, *60*, No. 7 (July 1972), pp. 792–800.
7. B. G. Haskell, "Frame Replenishment Coding of Television," a chapter in *Image Transmission Techniques*, W. K. Pratt, Ed., New York: Academic Press, 1978.
8. F. Rocca, "Television Bandwidth Compression Utilizing Frame-to-Frame Correlation and Movement Compensation," *Symposium on Picture Bandwidth Compression* (M.I.T. Cambridge, Mass., 1969), Gordon and Breach, 1972.
9. B. G. Haskell and J. O. Limb, "Predictive Video Encoding Using Measured Subject Velocity," U.S. Patent 3,632,865, January 1972.
10. W. N. Martin and J. K. Aggaraval, "Dynamic Scene Analysis; The Study of Moving Images," Technical Report No. 184, Information Systems Research Laboratory, University of Texas, Austin, Texas, January 1977.
11. J. L. Potter, "Scene Segmentation Using Motion Information," Computer Graphics and Image Processing, *6* (1977), pp. 558–581.
12. L. Dreschler and H.-H. Nagel, "Using 'Affinity' for Extracting Images of Moving Objects from TV-Frame Sequences," Technical Report IFI-HH-B-44/78, The University of Hamburg, Hamburg, Germany, February 1978.
13. J. O. Limb and J. A. Murphy, "Estimating the Velocity of Moving Images from Television Signals," Computer Graphics and Image Processing, *4* (1975), pp. 311–327.
14. C. Cafforio and F. Rocca, "Methods for Measuring Small Displacements of Television Images," IEEE Trans. Inform. Theory, *IT-22*, No. 5 (September 1976), pp. 573–579.
15. B. G. Haskell, "Entropy Measurements for Nonadaptive and Adaptive, Frame-to-Frame, Linear Predictive Coding of Video Telephone Signals," B.S.T.J., *54*, No. 6 (August 1975), pp. 1155–1174.

16. J. D. Robbins and A. N. Netravali, "Motion-Compensated Television Coding: Part II," unpublished work.
17. A. O. Aboutalib, M. S. Murphy, and L. M. Silverman, "Digital Restoration of Images Degraded by General Motion Blurs," IEEE Trans. on Automatic Control, *AC-22*, No. 3 (June 1977), pp. 294–302.
18. B. G. Haskell, "Differential Addressing of Clusters of Changed Picture Elements for Interframe Coding of Videotelephone Signals," IEEE Trans. Commun. (January 1976), pp. 140–144.
19. B. Widrow, J. Glover, J. McCool, et al., "Adaptive Noise Canceling: Principles and Applications," Proc. IEEE, *63* (December 1975), pp. 1692–1716.
20. L. Ljung, "Analysis of Recursive Stochastic Algorithms," IEEE Transactions on Automatic Control, *AC-22*, No. 4 (August 1977), pp. 551–575.
21. J. Salz, "On the Behavior of Stochastic Gradient Algorithms," unpublished work.

Copyright © 1979 American Telephone and Telegraph Company
THE BELL SYSTEM TECHNICAL JOURNAL
Vol. 58, No. 3, March 1979
Printed in U.S.A.

Multiqueue Systems with Nonexhaustive Cyclic Service

By P. J. KUEHN

(Manuscript received July 14, 1978)

Queuing models with cyclic-type service are applicable for performance studies of polling mechanisms in data communication and switching systems or cyclic scheduling algorithms in real time computers. This paper provides an approximate analysis of the multiqueue system $M^{[X]}/G/1$ with batch Poisson input, general service times, general overhead (switchover) times, and a single server operating under a cyclic strategy with nonexhaustive service of queues. Based on a new concept of conditional cycle times, the generating function of the stationary probabilities of state, the Laplace-Stieltjes transforms of the delay distributions, and the mean waiting times are derived explicitly for each queue through an imbedded Markov chain approach and an independence assumption. The approximate analytic results are validated by computer simulations. Besides this analysis, a stability criterion is derived for the general case of GI/G/1 systems with cyclic priority service. The paper concludes with a number of studies of the behavior of cyclic queues discovering interesting properties such as the dependence of cycle times and waiting times on the arrival and service process types and on the efficiency of cyclic priorities.

I. INTRODUCTION

Cyclic service is a frequently used mechanism for the information transfer between peripheral units and their centralized control opposed to asynchronous or synchronous interrupt mechanisms. In a cyclic service operation, the centralized control scans the peripheral units in a cyclic sequence. At each peripheral unit, the queue of waiting items (user or control data) is served either completely ("exhaustive service") or up to a specified maximum number of transferred items per scan ("nonexhaustive service") until the centralized device switches over to the succeeding unit within the cycle sequence. Examples of this type of operation are found in data communications systems (polling, asynchronous multiplexing), telephone switching systems (device scanning), and certain I/O mechanisms of real-time computers. The performance of these cyclic service mechanisms is of considerable interest for traffic engineering, namely with respect to throughput and resource utilizations, delays, unbalanced load, overload behavior, and the influence of various statistical properties of the traffic.

In the sequel, we refer to the general cyclic queuing model shown in Fig. 1. There are g arrival groups of "customers" and their corresponding waiting lines (queues). Customers of group j arrive according to a general independent (GI) arrival process with probability distribution function (pdf) $A_j(t) = P\{T_{Aj} \leq t\}$, where T_{Aj} denotes the random variable of the interarrival time in queue j, $j = 1, 2, \cdots, g$, and $\lambda_j = 1/ET_{Aj}$ defines the arrival rate of customers in queue j. Special cases of the GI arrival processes are: D (deterministic), M (Markovian), E_k (Erlangian order k), or H_2 (hyperexponential order 2). In case of batch arrivals, the arrival process is defined by both the random interarrival time T_{Bj} of batches and the random batch size K_j, $j = 1, 2, \cdots, g$. The batch size in queue j is given by its probability distribution $q_{jk} = P\{K_j = k\}$, $k = 0, 1, \cdots$. The total arrival rate of j-customers λ_j and the arrival rate of batches λ_{Bj} are related to each other through $\lambda_j = \lambda_{Bj} \cdot EK_j$. For the special case of deterministic arrival processes in more

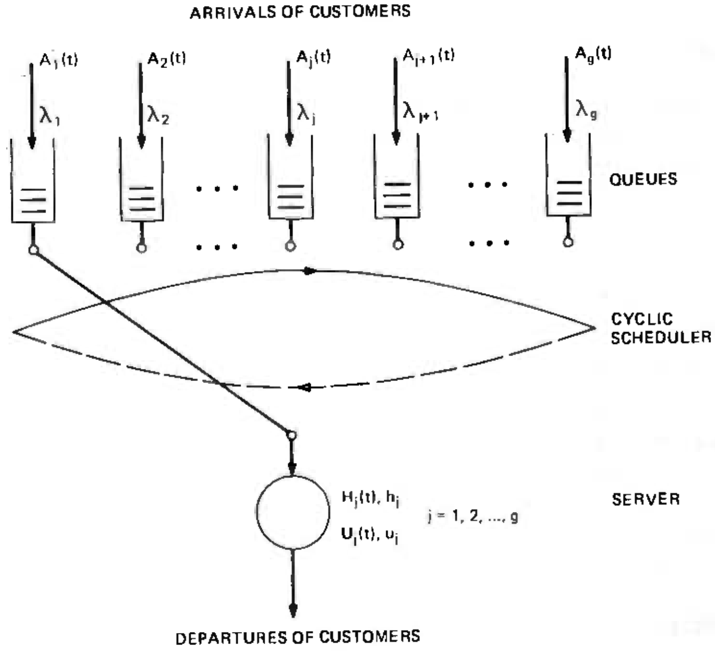


Fig. 1—Cyclic queuing model.

than one queue, a "phase shift parameter" is additionally needed which describes the relation between the periodic arrival patterns in those queues.

Similarly, customers of queue j receive a random service time T_{Hj} with pdf $H_j(t) = P\{T_{Hj} \leq t\}$ and mean $h_j = ET_{Hj}, j = 1, 2, \cdots, g$. Once the server has finished service at a particular queue j, it switches to the succeeding queue in a finite switchover (overhead) time T_{Uj} with pdf $U_j(t) = P\{T_{Uj} \leq t\}$ and mean $u_j = ET_{Uj}, j = 1, 2, \cdots, g$.

Finally, the general nonexhaustive cyclic operation of the server may be specified by a sequence ("cycle") $\{i_1, i_2, \cdots, i_l\}$, where $i_k \epsilon (1, 2, \cdots, g)$ denotes the number of that queue which is served in kth position within the cycle (l = cycle length). The sequence $\{i_1, i_2, \cdots, i_l\}$ is repeated in a cyclic manner. If there is no customer to serve from the i_kth queue, the server switches over to the i_{k+1}st queue (modulo l). An example of this general (mixed) cyclic sequence for $g = 3$ queues is $\{1, 2, 1, 3, 1, 2\}$, where $l = 6$. In this case, an overhead phase is inserted after every queue visit. Cyclic schedules with different frequencies of visits at the various queues within a cycle will also be referred to as "cyclic priority service." An important special case of cyclic priority service is obtained when all visits at a particular queue within a cycle are clustered such that the server attends queue 1 successively up to l_1 times, queue 2 successively up to l_2 times, and so on ($l = l_1 + l_2 + \cdots + l_g$). In this case, an overhead occurs only when changing to another queue. Limiting cases of this schedule are cycles with $l_j = 1, j = 1, 2, \cdots, g$ ("ordinary cyclic service" $\{1, 2, \cdots, g\}$) or cycles with $l_j \gg 1, j = 1, 2, \cdots, g$ ("exhaustive cyclic service"). The queuing analysis in this paper is limited to the practical important case of ordinary cyclic service; for stability and simulation studies the more general (nonmixed) cyclic priority service will be considered.

Queues with cyclic service have received considerable attention in literature (see Refs. 1 to 16). Cyclic queues with exhaustive service with or without overhead have been treated in case of $M/G/1$ models[1-8] and in case of discrete arrival and service processes.[9] The case of nonexhaustive cyclic service involves considerable mathematical difficulties and has been treated rigorously only for $M/G/1$ models with two queues without overhead.[10,11] Because of the mathematical intractability of most cyclic queuing problems, several approximate methods were suggested.[12-16] The approximate methods usually rest on some simplifying assumptions such as the "independence assumption"[12] under which the stochastic processes within a particular queue are considered more or less independent of the processes within the other queues.

In Section II of this paper, we first derive a stability criterion for queues of the type $GI/G/1$ with cyclic priority service. Section III deals with the cycle time analysis for $GI/G/1$ queues in case of ordinary cyclic service. Sections IV and V present an analysis of the probabilities of state and the waiting times for cyclic queuing models of the types $M/G/1$ and $M^{[X]}/G/1$ with ordinary cyclic service and general overhead, respectively. In Section VI finally, we report various numerical results of the approximate analysis and of computer simulations for validation and qualitative performance studies. Some of those results discover new insight into the properties of cyclic queues and could have direct consequences for system engineering and future research as well.

II. STABILITY OF CYCLIC QUEUES

Contrary to most standard queuing problems, an obvious and simple criterion does not exist under which a queue in a cyclic queuing system stays stable. In the following sections, we develop a stability criterion for $GI/G/1$ multiqueue systems with a (nonmixed) cyclic priority service.

2.1 A stability criterion for queues with cyclic priority service

Following analogously to a common definition for stability in system theory, a queuing system will be called "stable" if for positive service times and finite input rates the average queue lengths are limited (note that a stationary queue is stable, whereas a stable queue need not necessarily be stationary). Additionally, we assume that all arrival and service processes are stationary so that the following reasoning can be based on average values *independent* of specific distributional assumptions.

Let T_C be the random cycle time, $c = ET_C$ the average cycle time, and $c_0 = u_1 + u_2 + \cdots + u_g$ the average of the cycle time under the condition that no customer is served during a cycle. The average number of arriving j customers during a cycle is $n_j = \lambda_j c$. In the stationary state of the system, the average number of arriving j customers equals the average number of served j customers, $j = 1, 2, \cdots, g$. Thus, we have

$$c = c_0 + \sum_{j=1}^{g} (\lambda_j c) h_j,$$

from which we find the result

$$c = \frac{c_0}{1 - \rho_0}, \tag{1}$$

where $\rho_0 = \rho_1 + \rho_2 + \cdots + \rho_g$ defines the total server utilization and $\rho_j = \lambda_j h_j$ is the server utilization by j customers only, $j = 1, 2, \cdots, g$. The result according to (1) has already been discovered for cyclic queues with exhaustive and ordinary cyclic service. To find the boundary of system stability, we proceed as follows: First, we state a stability criterion for a particular queue j under the condition of stability of the residual queues. This condition can always be achieved for sufficiently small arrival rates in the residual queues. The whole system is stable if and only if all individual stability conditions are satisfied simultaneously.

Under the condition that all queues $\nu \neq j$ are stable, queue j approaches the stability boundary as $n_j \rightarrow l_j$; this corresponds to a maximum arrival rate $\lambda_{j\,\max}$ at the margin $n_j = l_j$ and an average cycle length c_j^*:

$$\lambda_{j\,\max} = \frac{l_j}{c_j^*}, \qquad \text{where} \qquad c_j^* = \frac{c_0 + l_j h_j}{1 - \rho_0 + \rho_j}. \tag{2a}$$

Thus, the system is stable if for all queues

$$\lambda_j < \lambda_{j\,\max} = \frac{l_j}{c_0 + l_j h_j} \cdot (1 - \rho_0 + \rho_j), \quad j = 1, 2, \cdots, g, \tag{2b}$$

are fulfilled simultaneously. In a similar way, criteria of partial stability can be stated in cases where some queues are saturated (a saturated queue i contributes to the average cycle time by $l_i \cdot h_i$). Finally, it should be noted that the average cycle time c stays always stable since $c \leq c_0 + l_1 h_1 + \cdots + l_g h_g$.

2.2 Examples

To further explore the stability criterion, consider the example of $g = 2$ queues. From (2b) we find the following relationships between λ_1 and λ_2:

$$\lambda_1 < \frac{l_1}{c_0 + l_1 h_1} \cdot (1 - \lambda_2 h_2), \qquad \lambda_2 < \frac{l_2}{c_0 + l_2 h_2} \cdot (1 - \lambda_1 h_1), \tag{3a, b}$$

where $c_0 = u_1 + u_2$. These relationships are shown graphically by two marginal lines in Fig. 2 with the intersection

$$\lambda_{jo} = \frac{l_j}{c_0 + l_1 h_1 + l_2 h_2}, \qquad j = 1, 2. \tag{4}$$

The absolute stable region is below the hatched area when both individual criteria (3a), (3b) are fulfilled simultaneously. For $\lambda_1 < \lambda_{1o}$ queue 2 always saturates first, whereas for $\lambda_2 < \lambda_{2o}$ queue 1 saturates first. At the intersection $(\lambda_{1o}, \lambda_{2o})$, both queues saturate simultaneously. Similarly, for $\lambda_1 > \lambda_{1o}$, $\lambda_2 > \lambda_{2o}$, both queues are saturated (absolute unstable region). Within the intermediate regions $\lambda_1 > l_1 \cdot (1 - \lambda_2 h_2)/(c_0 + l_1 h_1)$, $\lambda_2 < \lambda_{2o}$ (or $\lambda_2 > l_2 \cdot (1 - \lambda_1 h_1)/(c_0 + l_2 h_2)$, $\lambda_1 < \lambda_{1o}$), queue 1 (or queue 2) is saturated, whereas queue 2 (or queue 1) is stable; in these regions of partial stability, the cyclic queuing system

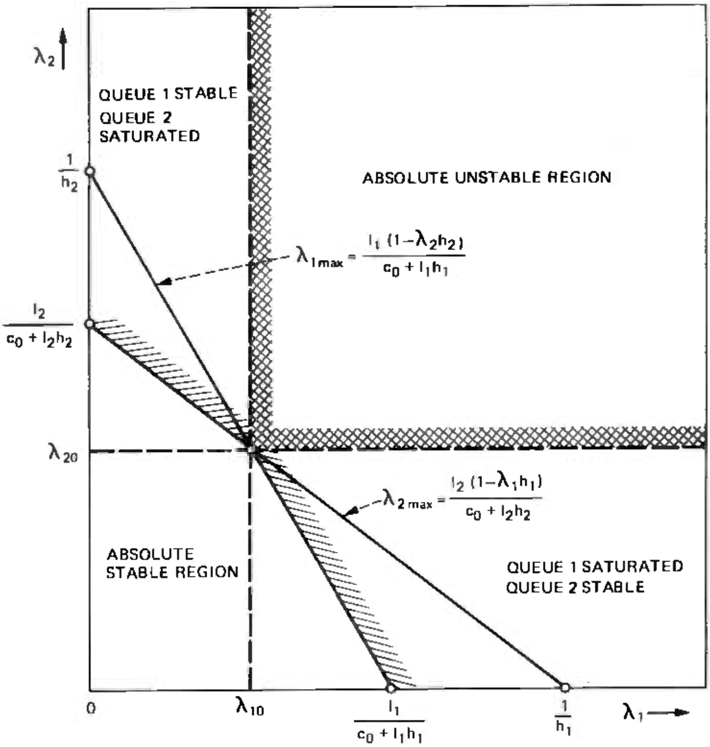


Fig. 2—Stability regions for a *GI/G/*1 queuing system with 2 queues and cyclic priority service.

can be considered as consisting only from the stable queue where the contribution of the unstable queue to the cycle time affects as an increased cycle time overhead of l_1 (or l_2) consecutive service times T_{H1} (or T_{H2}). Furthermore, starting from any point within the absolute stable region and increasing λ_1 and λ_2 simultaneously, we state that the queue with the greater λ_j/l_j ratio reaches saturation first, *independent* of the service and overhead time parameters. This statement differs from many other queuing stability criteria.

Finally, we discuss briefly two important special cases of the above example. The first special case is that of ordinary cyclic service ($l_1 = l_2 = 1$). The intersection in Fig. 2 falls on the median $\lambda_1 = \lambda_2$. This means that the queue with the greater arrival rate always saturates first. This result was already found by M. Eisenberg[11] for *M/G/*1 systems without overhead. Additionally, the average number of j customers served during a cycle is identical with the probability α_j that the server meets at least one customer in queue j:

$$\alpha_j = \lambda_j c = \frac{c_0}{1-\rho_0} \cdot \lambda_j, \qquad j = 1, 2. \tag{5}$$

The second special case is that of exhaustive cyclic service ($l_1, l_2 \gg 1$). In this case, both marginal lines fall together on $\lambda_1 h_1 + \lambda_2 h_2 = 1$. The stability criterion is simply $\lambda_1 h_1 + \lambda_2 h_2 = \rho_1 + \rho_2 = \rho_0 < 1$. Approaching the stability margin, both queues saturate simultaneously independent of c_0.

III. CYCLE TIME ANALYSIS

In this section, we consider multiqueue systems of the type $GI/G/1$ with overhead and ordinary cyclic service. Since the queuing analysis in Sections IV and V is based on the knowledge of cycle time, we briefly discuss a known result and then develop an improved approximate solution for the pdf of the cycle time.

3.1 Cycle time analysis by Hashida and Ohara

The exact solution for the pdf of the cycle time T_C is still unknown, except for the mean c in (1). Based on the probabilities α_j in (5) and the approximation assumption of independence, Hashida and Ohara[13] gave the following expression for the Laplace-Stieltjes transform (LST) of the cycle time pdf $C(t) = P\{T_C \leq t\}$:

$$\phi_C(s) = \prod_{i=1}^{g} \phi_{Ui}(s) \cdot \prod_{i=1}^{g} (\alpha_i \phi_{Hi}(s) + [1 - \alpha_i]). \qquad (6)$$

In (6), $\phi_C(s) = \int_{0-}^{\infty} e^{-st} dC(t)$ defines the LST of $C(t)$; similarly, $\phi_{Ui}(s)$ and $\phi_{Hi}(s)$ denote the LSTs of $U_i(t)$ and $H_i(t)$, respectively. The expression (6) follows directly when considering T_C as a sum of independent random variables.

From (6), the exact mean cycle time c follows straightforwardly and agrees with (1). However, it was found by intensive simulations (some of them are given in Section VI) that (6) underestimates the cycle time variance and, herewith, also the mean waiting times. For this reason, we shall now improve the cycle time analysis by introduction of a new concept of "conditional cycle times."

3.2 Conditional cycle times

The basic idea of the queuing analysis in Ref. 13 and in Sections IV and V of this paper is the description of the queue length of a particular queue j at the scan instant by an imbedded Markov chain. The influence of all queues $\nu \neq j$ on the considered queue j will be expressed only through the cycle time. The cycle time T_C is the time interval between two successive scan instants of a queue (say, j). However, a particular realization of T_C clearly depends on whether a j customer is served or not in a cycle. Therefore, we introduce two conditional cycle times T_{C_j} and $T_{C_j'}$, with respect to the considered queue j for cycles without or with a service time contribution to the cycle by a customer of queue j, respectively. The corresponding cycles are denoted by C_j'

and C_j''. Since $ET_{C_j''} > ET_{C_j'}$, it is more likely that after a long cycle another long one is followed, and vice versa. Thus, the concept of conditional cycle times increases the cycle time variance through a reduction of the independence assumption by explicit consideration of some history of a current cycle.

Let $C_j'(t)$ and $C_j''(t)$ be the pdfs of the conditional cycle times $T_{C_j'}$ and $T_{C_j''}$, and α_{ji}', α_{ji}'' the corresponding probabilities for the service of an i customer ($i \neq j$) during a conditional cycle without or with a j service, respectively. Then, it follows by similar reasoning as for (6):

$$\phi_{C_j'}(s) = \prod_{i=1}^{g} \phi_{Ui}(s) \cdot \prod_{i \neq j} (\alpha_{ji}' \phi_{Hi}(s) + [1 - \alpha_{ji}']), \tag{7a}$$

$$\phi_{C_j''}(s) = \prod_{i=1}^{g} \phi_{Ui}(s) \cdot \prod_{i \neq j} (\alpha_{ji}'' \phi_{Hi}(s) + [1 - \alpha_{ji}'']) \cdot \phi_{Hj}(s). \tag{7b}$$

For the (unconditional) cycle time T_C we find from the law of total probabilities

$$\phi_C(s) = (1 - \alpha_j)\phi_{C_j'}(s) + \alpha_j \phi_{C_j''}(s). \tag{8}$$

With $c_j' = ET_{C_j'}$ and $c_j'' = ET_{C_j''}$ we state the conditional cycle time balances:

$$c_j' = c_0 + \sum_{i \neq j} \alpha_{ji}' h_i, \tag{9a}$$

$$c_j'' = c_0 + \sum_{i \neq j} \alpha_{ji}'' h_i + h_j, \tag{9b}$$

$$c = (1 - \alpha_j) c_j' + \alpha_j c_j''. \tag{9c}$$

Similarly, as in (5), we assume

$$\alpha_{ji}' = \lambda_i c_j', \tag{9d}$$

$$\alpha_{ji}'' = \lambda_i c_j'', \qquad i \neq j. \tag{9e}$$

Inserting (9d), (9e) in (9a), (9b), we find

$$c_j' = \frac{c_0}{1 - \rho_0 + \rho_j}, \tag{10a}$$

$$c_j'' = \frac{c_0 + h_j}{1 - \rho_0 + \rho_j}. \tag{10b}$$

Note that the exact value of c in (1) follows from (9c) and (10a), (10b). It should also be mentioned that the solution (10a), (10b) holds only as long as $\alpha_{ji}'' \leq 1$. This condition is always fulfilled in case of symmetrical load ($\lambda_1 = \lambda_2 = \cdots = \lambda_g$, $h_1 = h_2 = \cdots = h_g$). In case of higher unsymmetrical loads, it can indeed happen that $\alpha_{ji}'' > 1$ so that α_{ji}'' can no longer be interpreted as probability; this difficulty can be overcome by a suitable limitation of α''_{ji} by 1 (i.e., queue i always contributes a service time to the conditional cycle time $T_{C''_j}$).

With (9d), (9e) and (10a), (10b), the conditional cycle time pdfs $C'_j(t)$ and $C''_j(t)$ are completely defined. The mean conditional cycle times c'_j and c''_j are given by (10a), (10b). For the variances, we find from (7a), (7b):

$$\text{VAR } T_{C'_j} = \sum_{i=1}^{g} \text{VAR } T_{Ui} + \sum_{i \neq j} (\alpha'_{ji} h_i^{(2)} - \alpha'^{2}_{ji} \cdot h_i^2), \tag{11a}$$

$$\text{VAR } T_{C''_j} = \sum_{i=1}^{g} \text{VAR } T_{Ui} + \sum_{i \neq j} (\alpha''_{ji} h_i^{(2)} - \alpha''^{2}_{ji} \cdot h_i^2) + \text{VAR } T_{Hj}, \tag{11b}$$

where $h_i^{(2)} = ET_{Hi}^2$ denotes the ordinary second moment of T_{Hi}. The cycle-time variance is finally given by

$$\text{VAR } T_C = (1 - \alpha_j) \cdot [\text{VAR } T_{C'_j} + c'^{2}_j] + \alpha_j \cdot [\text{VAR } T_{C''_j} + c''^{2}_j] - c^2. \tag{11c}$$

The second moments of the cycle times follow from the definition $\text{VAR } T_C = c^{(2)} - c^2$.

IV. QUEUING ANALYSIS OF *M/G/1* SYSTEMS

Based on the concept of the conditional cycle times of Section III, a queuing analysis is given for multiqueue systems of the type $M/G/1$ with general overhead times and ordinary cyclic service by means of an imbedded Markov chain. Basically, the derivation follows the approach of Hashida and Ohara.[13]

4.1 Probabilities of state

For an exact analysis, the state of the system at a time t has to be defined such that all past history is summarized in it so that the future development of the system state process is completely determined from it. In the present case, the system state could be described by a vector $\{N_1(t), N_2(t), \cdots, N_g(t), I(t), X_0(t)\}$, where $N_j(t)$ defines the number of waiting customers in queue j, $j = 1, 2, \cdots, g$, $I(t)$ points to the present location of the server within the cycle, and $X_0(t)$ specifies the age of the current service (or overhead) phase of the server. An exact analysis on this base seems not to be feasible.

In the following analysis, only the state N_j of a particular queue j is considered. Moreover, the analysis does not apply to continuous time but is restricted to a set of special points, namely the scan instants (or departure instants) of the considered queue j. The time intervals between the scan instants of queue j are the conditional cycle times $T_{C'_j}$ and $T_{C''_j}$; the influence of all the other queues on the queue length process in queue j is completely expressed by those cycle times. Although the following imbedded Markov chain solution is formally exact, the analysis approach is approximate since T_{C_j} $T_{C''_j}$ are assumed to be independent and identically distributed (iid) variables; the expressions for their pdf are only approximations, too.

The outlined method only renders results for the particular queue j under consideration. In the case of unsymmetrical systems, the procedure must be repeated for the other queues, too. For ease of reading, we suppress the subscript j in the following treatment, i.e., we write $\lambda, h, \rho, c', c'', \cdots$ instead of $\lambda_j, h_j, \rho_j, c'_j, c''_j \cdots$.

4.1.1 State distribution at scan instants

We assume that the queuing sytem is in the stationary state. Let N be the number of waiting customers at the server arrival instant (scan instant) of a particular queue. We are interested in the stationary distribution

$$p_n = P\{N = n\}, \qquad n = 0, 1, 2, \cdots. \tag{12}$$

Because of the memoryless property of the arrival process, the system state of the considered queue forms an imbedded Markov chain at the discrete set of scan instants (renewal points). The stationary distribution satisfies the equation (see Ref. 17, pp. 167–174):

$$p_n = p_0 \cdot p_{0n} + \sum_{m=1}^{n+1} p_m \cdot p_{mn}, \qquad n = 1, 2, \cdots, \tag{13a}$$

where the transition probabilities p_{mn} are given by

$$p_{mn} = \begin{cases} \displaystyle\int_{t=0-}^{\infty} e^{-\lambda t} \cdot \frac{(\lambda t)^{n-m+1}}{(n-m+1)!} \cdot dC''(t), & m > 0 \\ \\ \displaystyle\int_{t=0-}^{\infty} e^{-\lambda t} \cdot \frac{(\lambda t)^{n}}{n!} \cdot dC'(t), & m = 0. \end{cases} \tag{13b}$$

Together with the normalizing condition

$$\sum_{n=0}^{\infty} p_n = 1, \tag{13c}$$

the stationary probabilities of state at the scan instants are completely determined by the set of equations (13a), (13b), and (13c). Introducing the probability generating function of the state distribution p_n, $n = 0, 1, 2, \cdots$,

$$G(x) = \sum_{n=0}^{\infty} p_n x^n, \tag{14}$$

we obtain after some algebraic manipulations

$$G(x) = p_0 \cdot \frac{x\phi_{C'}(z) - \phi_{C''}(z)}{x - \phi_{C''}(z)}, \qquad \text{where} \qquad z = \lambda(1 - x). \tag{15}$$

Note that $G(x)$ is completely expressed by p_0 and the LSTs of the two conditional cycle times. Using the identity $G(1) = 1$, we find from (15) through evaluation of $\lim_{x \to 1} G(x)$ by L'Hospital's rule

$$p_0 = \frac{1 - \lambda c''}{1 - \lambda(c'' - c')} = 1 - \alpha. \tag{16}$$

The latter identity can be shown by using equations (10a), (10b), and (5).

The expected number of waiting customers at the scan instant follows from

$$EN = \frac{d}{dx} G(x) \bigg|_{x=1} .$$

This results in

$$EN = p_0 \cdot \lambda \cdot \frac{\lambda c'^{(2)} \cdot (1 - \lambda c'') + c'(\lambda^2 c''^{(2)} + 2 - 2\lambda c'')}{2(1 - \lambda c'')^2}, \tag{17}$$

where $c'^{(2)} = ET^2_{C'}$, and $c''^{(2)} = ET^2_{C''}$.

4.1.2 State distribution at departure instants

Let N^* be the number of waiting customers within the considered queue which are left behind by a departing customer of that queue with distribution

$$p^*_n = P\{N^* = n\}, \qquad n = 0, 1, 2, \cdots \tag{18}$$

and generating function

$$G^*(x) = \sum_{n=0}^{\infty} p^*_n x^n. \tag{19}$$

The probability p^*_n can be expressed through the probability of having m customers at the scan instant given that the considered queue is not empty, $p_m/(1 - p_0)$, and the probability of $n - m + 1$ new arrivals in that queue during the subsequent service time of one customer. Hence,

$$p^*_n = \sum_{m=1}^{n+1} \frac{p_m}{1 - p_0} \cdot \int_{t=0-}^{\infty} e^{-\lambda t} \cdot \frac{(\lambda t)^{n-m+1}}{(n - m + 1)!} \cdot dH(t),$$

$$n = 0, 1, 2, \cdots . \tag{20}$$

Substituting (20) in (19) and interchanging the order of summation and integration, we find

$$G^*(x) = \frac{1}{1-p_0} \cdot \frac{G(x) - p_0}{x} \cdot \phi_H(z), \tag{21}$$

where

$$z = \lambda(1 - x).$$

Therefore, it follows for the expected number of customers at the departure epoch:

$$EN^* = \frac{d}{dx} G^*(x)\bigg|_{x=1} = \frac{EN}{1-p_0} - 1 + \rho. \tag{22}$$

On the other hand, EN^* equals the expected number of customers which have arrived during the sojourn (waiting + service) time of the departing customer (for this, consider as an example the queue discipline FCFS, first come, first-served). Hence, $EN^* = \lambda \cdot (w + h)$, where $w = ET_W$ denotes the average waiting time in the considered queue (service being excluded). Solving for w, we find with (22)

$$w = \frac{1}{\lambda} \cdot \left[\frac{EN}{1-p_0} - 1\right]. \tag{23}$$

4.2 Delay analysis

For the following derivation, the queue discipline FCFS is assumed. Let T_W be the waiting time which an arbitrary customer of the considered queue (in the following denoted by "test customer") has to undergo with pdf $W(t)$ and LST $\phi_W(s)$. Through an analogous reasoning as in the previous section, p_n^* can alternatively be considered as the distribution of the number of arriving customers during the sojourn time T_S of the test customer. Since $T_S = T_W + T_H$ and since T_W and T_H are independent of each other, the pdf of T_S is the convolution of $W(t)$ and $H(t)$, symbolized by $W(t) \circledast H(t)$. Hence,

$$p_n^* = \int_{t=0-}^{\infty} e^{-\lambda t} \cdot \frac{(\lambda t)^n}{n!} \cdot d(W(t) \circledast H(t)), \qquad n = 0, 1, 2, \cdots. \tag{24}$$

Applying (19) in (24), we find $G^*(x) = \phi_W(z) \cdot \phi_H(z)$, where $z = \lambda(1 - x)$, which, with (21), finally results in

$$\phi_W(s) = \frac{1 - \lambda c''}{c'} \cdot \frac{1 - \phi_{C'}(s)}{s - \lambda \cdot [1 - \phi_{C''}(s)]}. \tag{25}$$

From (25) we find for the mean waiting time

$$w = -\frac{d}{ds}\phi_W(s)\bigg|_{s=0} = \frac{c'^{(2)}}{2c'} + \frac{\lambda c''^{(2)}}{2(1 - \lambda c'')}. \tag{26}$$

Equation (26) reveals that the mean waiting time depends basically on the first and second moments of the conditional cycle times. Note also that (26) agrees with (23) when the corresponding results for p_0, EN, c', c'', $c'^{(2)}$, and $c''^{(2)}$ from eqs. (16), (17), (10a), (10b), and (11a), (11b), respectively, are inserted.

It may be mentioned that the result (26) can also be derived directly through the application of renewal theory and Little's law: An arriving test customer of the considered queue meets either a cycle **C**′ or **C**″ in progress. Since the arrival process is a Markovian process, the probabilities of meeting a cycle **C**′ or **C**″ is simply the weighted ratio of frequencies, i.e. $(1-\alpha)\cdot(c'/c)$ or $\alpha\cdot(c''/c)$, respectively. According to our approximation assumption, the conditional cycle times $T_{C'}$ and $T_{C''}$ are iid-variables. Thus, the average residual cycle times are $c'^{(2)}/2c'$ and $c''^{(2)}/2c''$ according to renewal theory (see, for example, Ref. 17, pp. 158–161). The average waiting time w consists of the average residual cycle time and the product of the mean cycle time c'' and the average number L of customers met at the arrival instant of the test customer; the latter one can be expressed through Little's law (see, for example, Ref. 17, pp. 156–158) through $L = \lambda\cdot w$. The average waiting time w can now be balanced as

$$w = (1-\alpha)\cdot\frac{c'}{c}\cdot\frac{c'^{(2)}}{2c'} + \alpha\cdot\frac{c''}{c}\cdot\frac{c''^{(2)}}{2c''} + (\lambda w)c''. \qquad (27)$$

Solving for w, we yield precisely the result (26) from (27).

The pdf $W(t)$ can be obtained by the inversion of (25) either through a partial fraction expansion (in case of rational LSTs), by the numerical inversion technique of D. Jagerman,[18] or by an approximation using the ordinary first and second moments.[19]

Finally, we mention that (25) includes the exact result for the limiting case $g = 1$ of a single cyclic queue with overhead. Furthermore, another limit with zero overhead can be derived from (25); this case represents the worst case with respect to the approximation accuracy (see Section VI).

V. QUEUING ANALYSIS OF $M^{[X]}/G/1$ SYSTEMS

In this section, the solution of Section IV for single Poisson arrivals (M) is generalized to batch Poisson arrivals ($M^{[X]}$) in every queue. The analysis follows analogously to Section IV; i.e., we consider all processes with respect to a particular queue j. Again, the subscript j will be suppressed for ease of reading.

5.1 Probabilities of state

5.1.1 Arrival process

Customers of the considered queue arrive in batches of size K with distribution

$$q_k = P\{K = k\}, \qquad k = 0, 1, 2, \cdots \tag{28a}$$

and probability generating function

$$Q(x) = \sum_{k=0}^{\infty} q_k x^k. \tag{28b}$$

The interarrival times of batches are exponentially distributed with mean $1/\lambda_B = EK/\lambda$, where λ and λ_B are the arrival rates of customers and batches within the considered queue, respectively.

Let $N_B(t)$ be the number of batch arrival instants in $(0, t)$ with distribution

$$P\{N_B(t) = n\} = \frac{(\lambda_B t)^n}{n!} \cdot e^{-\lambda_B t}, \qquad n = 0, 1, 2, \cdots \tag{29a}$$

and probability generating function

$$g(x, t) = \sum_{n=0}^{\infty} \frac{(\lambda_B t)^n}{n!} \cdot e^{-\lambda_B t} \cdot x^n = e^{-\lambda_B t(1-x)}. \tag{29b}$$

Finally, let $N_A(t)$ be the total number of customers arriving at the considered queue in $(0, t)$. Then, the probability generating function of the distribution $P\{N_A(t) = k\}$, $k = 0, 1, 2, \cdots$, is given by

$$h(x, t) = \sum_{k=0}^{\infty} P\{N_A(t) = k\} \cdot x^k = g(Q(x), t) = e^{-\lambda_B t[1-Q(x)]}. \tag{30}$$

5.1.2 State distribution at scan instants

Since the cycle time approximation of Section III holds for $GI/G/1$ cyclic queues, the same pdfs $C(t)$, $C'(t)$, and $C''(t)$ can be used for batch arrival processes. As in Section 4.1., let p_n be the stationary probability of state for n customers waiting within the considered queue at the scan instants. The transition probabilities p_{mn} in (13a) are now

$$p_{mn} = \begin{cases} \displaystyle\int_{t=0-}^{\infty} \sum_{\nu=0}^{\infty} P\{N_B(t) = \nu\} \\ \quad \cdot P\{N_A(t) = n - m + 1 \mid N_B(t) = \nu\} \cdot dC''(t), & m > 0 \\ \\ \displaystyle\int_{t=0-}^{\infty} \sum_{\nu=0}^{\infty} P\{N_B(t) = \nu\} \\ \quad \cdot P\{N_A(t) = n \mid N_B(t) = \nu\} \cdot dC'(t), & m = 0. \end{cases} \tag{31}$$

The probabilities of state p_n are completely determined by (13a), (31), and (13c). The application of the generating function results finally in the same expression for $G(x)$ as in (15), however, with $z = \lambda_B \cdot [1 - Q(x)]$, where $Q(x)$ is defined by (28a), (28b). Also, for p_0 the identical result is obtained as in (16). Further results can easily be derived analogously as in Section 4.1.

5.1.3 State distribution at departure instants

Using the same definitions for p_n^* and $G^*(x)$ as in Section 4.1.2, we find, instead of (20),

$$p_n^* = \sum_{m=1}^{n+1} \frac{p_m}{1 - p_0} \cdot \int_{t=0-}^{\infty} \sum_{\nu=0}^{\infty} P\{N_B(t) = \nu\} \cdot P\{N_A(t)$$

$$= n - m + 1 \,|\, N_B(t) = \nu\} \, dH(t), \qquad n = 0, 1, 2, \cdots . \tag{32}$$

This again results in the same expression for $G^*(x)$ as in (21) with $z = \lambda_B \cdot [1 - Q(x)]$, from which further results could be derived analogously.

5.2 Delay analysis

Following the method outlined in Secton 4.2, p_n^* is also the distribution of the number of arriving customers during the sojourn time of a test customer of that queue. The number N^* of customers left behind in the considered queue by the departing test customer is now built up from *two* components:

$$N^* = N_1^* + N_2^*,$$

where

N_1^* = the number of customers that had arrived together with the test customer in one batch but that were *behind* the test customer

N_2^* = the number of customers that had arrived in *subsequently* arriving batches during the sojourn time T_S of the test customer.

Let $r_n = P\{N_1^* = n\}$, $n = 0, 1, 2, \cdots$, be the probability that the departing test customer leaves n customers behind which had arrived together with the test customer in one batch. The test customer arrived in a batch of size $K = k$ with probability (see Ref. 20)

$$q_k^* = \frac{kq_k}{EK}, \quad k = 1, 2, 3, \cdots . \tag{33}$$

The test customer is first, second, $\cdots$, kth in the batch of size k with probability $1/k$. Thus, q_k^*/k defines the probability that the test customer arrived in a batch of size k in $(k - n)$th position, $n = 0, 1, 2, \cdots, k - 1$. Then,

$$r_n = \sum_{k=n+1}^{\infty} \frac{q_k^*}{k} = \frac{1}{EK} \cdot \sum_{k=n+1}^{\infty} q_k, \qquad n = 0, 1, 2, \cdots, \tag{34}$$

and probability generating function

$$R(x) = \sum_{n=0}^{\infty} r_n x^n = \frac{1}{EK} \cdot \frac{1 - Q(x)}{1 - x}. \tag{35}$$

Now, we can establish the relation between p_n^* and T_S analogously as in Section 4.2:

$$p_n^* = \int_{t=0-}^{\infty} \left[\sum_{\mu=0}^{n} P\{N_1^* = \mu\} \cdot \sum_{\nu=0}^{\infty} P\{N_2^*(t) = n - \mu \mid N_B(t) = \nu\} \right. \\ \left. \cdot P\{N_B(t) = \nu\} \right] \cdot d(W(t) \circledast H(t)), \qquad n = 0, 1, 2, \cdots. \tag{36}$$

In (36), the bracket term expresses the probability of new arrivals within a sojourn time of length t through consideration of all principal possibilities of batch configurations of the departing test customer.

Introducing r_n and $R(x)$ from (34) and (35) and applying (19) on (36), we find after some intermediate calculations

$$G^*(x) = \phi_W(z) \cdot \phi_H(z) \cdot R(x), \tag{37}$$

where

$$z = \lambda_B[1 - Q(x)].$$

Equating both expressions in (21) and (37) yields the final result

$$\phi_W(s) = \frac{1 - \lambda c''}{\lambda c'} \cdot \frac{1 - \phi_{C'}(s)}{\phi_{C''}(s) - x} \cdot \frac{1}{R(x)}, \tag{38a}$$

where $x = f(s)$ the solution of

$$s = \lambda_B[1 - Q(x)]. \tag{38b}$$

From (38a), (38b), we find for the mean waiting time of a customer

$$w = \left(\frac{c'^{(2)}}{2c'} + \frac{\lambda c''^{(2)}}{2(1 - \lambda c'')} \right) + \frac{c''}{2(1 - \lambda c'')} \cdot \left[\frac{EK^2}{EK} - 1 \right]. \tag{39}$$

Note that the mean waiting time consists of two terms; the first term is identical with that of an $M/G/1$ system [see (26)], whereas the second term expresses the influence of *batch* arrivals.

Analogously to Section 4.2, the mean waiting time can be derived directly. Let $w(i)$ be the mean conditional waiting time of a customer who is ith in his batch. For $w(1)$, a similar balance can be stated as in (27):

$$w(1) = (1 - \alpha)\cdot\frac{c'}{c}\cdot\frac{c'^{(2)}}{2c'} + \alpha\cdot\frac{c''}{c}\cdot\frac{c''^{(2)}}{2c''} + (\lambda w)c''. \tag{40a}$$

The relationship between $w(1)$ and $w(i)$ is

$$w(i) = w(1) + (i - 1)\cdot c''. \tag{40b}$$

The mean waiting time w, irrespective of the test customer's position within the batch, follows by averaging over the conditional waiting times. Thus, with (33) and (40b), we have

$$w = \sum_{k=1}^{\infty} \frac{kq_k}{EK}\cdot\frac{1}{k}\cdot\sum_{i=1}^{k} w(i) = w(1) + \frac{c''}{2}\cdot\left[\frac{EK^2}{EK} - 1\right]. \tag{41a}$$

Inserting (41a) in (40a) and solving for $w(1)$ yields

$$w(1) = \frac{c'^{(2)}}{2c'} + \frac{\lambda c''^{(2)}}{2(1 - \lambda c'')} + \frac{\lambda c''^{(2)}}{2(1 - \lambda c'')}\cdot\left[\frac{EK^2}{EK} - 1\right]. \tag{41b}$$

The mean waiting time w is completely determined with (41a), (41b) and agrees with (39).

Finally, we give the explicit results for w in the case of two special batch size distributions. For *constant batch size k*, i.e., $q_i = \delta(i, k)$, we find

$$w\,|_{M^{[X]}/G/1} = w\,|_{M/G/1} + \frac{k - 1}{2}\cdot\frac{c''}{1 - \lambda c''}. \tag{42a}$$

In the case of *geometrically distributed batch sizes*, which are defined by $q_i = q^i\cdot(1 - q)$, $i = 0, 1, \cdots$, and $q = (EK + 1)/EK$, the result is

$$w\,|_{M^{[X]}/G/1} = w\,|_{M/G/1} + EK\cdot\frac{c''}{1 - \lambda c''}. \tag{42b}$$

The expressions (42a), (42b) demonstrate at first the increase of the waiting time through the batch Poisson arrival process compared to the pure Poisson arrival process and, second, the increase of w through geometrically distributed batches against constant batches.

VI. NUMERICAL RESULTS

In this section, the results of the approximate analysis are validated by computer simulations. Further results are given to show various properties of cyclic queuing systems.

6.1 Cycle time variance for ordinary cyclic service

Since the mean cycle time c according to (1) is always exact, the approximation accuracy can be judged in a first step by the cycle time variance VAR T_C (note that even the pdf $C(t)$ of the cycle time would not be sufficient for a complete validation since successive cycle times are not independent of each other; for a more complete validation, some covariance measure should be considered, too). We expect very good accuracy for low traffic (since the independence assumption is asymptotically exact for zero arrival rates) as well as for heavy traffic (since each of the queues contributes in the limit with a full service time to the cycle so that the cycle times become independent of each other again).

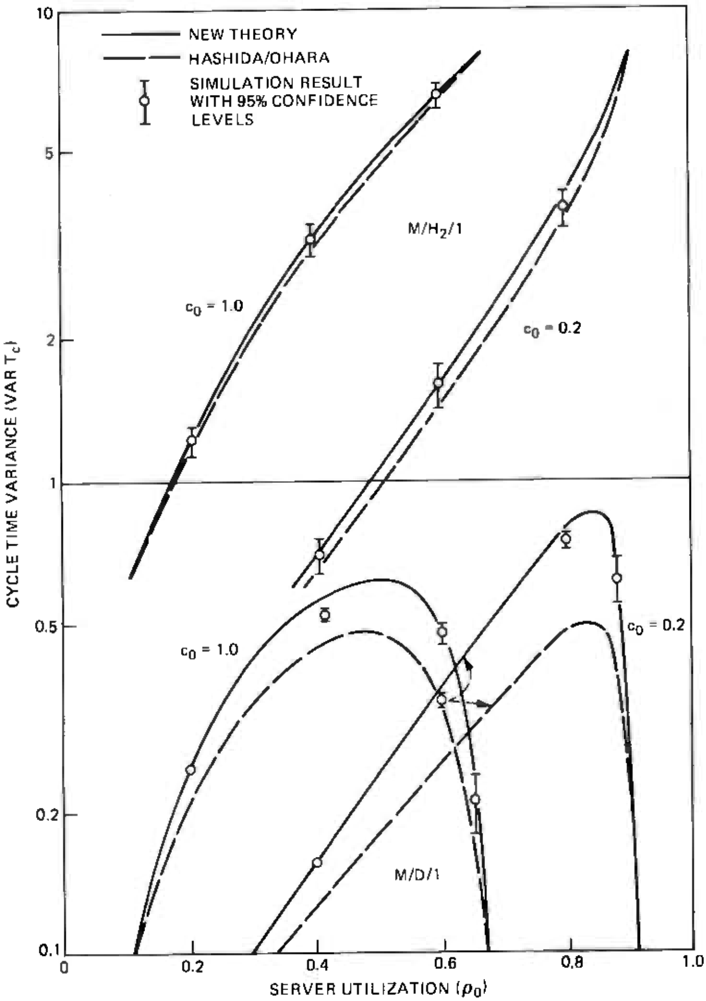


Fig. 3—Accuracy of cycle time variances

Parameters: g = 2 symmetrical queues $M/D/1$ and $M/H_2/1$
c_H = 2.0 coefficient of variation of service times for $M/H_2/1$
$h_1 = h_2 = 1$ average service times
c_0 = 0.2 and 1.0, constant overhead.

Table I—Cycle time variance for various $GI/M/1$ queues with ordinary cyclic service

Parameters: $g = 10$ symmetrical queues; $h_j = 1$, $u_j = 0.1$ (0.5), $\lambda_j = \lambda$, $j = 1, 2, \cdots, 10$; Constant overhead times; $D/M/1$: Equal phase shift between arrival instants; $H_2/M/1$: Interarrival time coefficient of variation $c_A = 2.0$; $M^{[X]}/M/1$: Constant batch size $k = 4$.

Parameters		VAR T_C (simulation)					VAR T_C (approx.)
ρ_0	c_0	$D/M/1$	$E_4/M/1$	$M/M/1$	$H_2/M/1$	$M^{[X]}/M/1$	$GI/M/1$
0	1.0	0	0	0	0	0	0
0.4	1.0	1.83	1.88	1.91	1.88	1.66	1.38
0.6	1.0	6.60	5.38	5.14	5.03	4.61	3.25
0.8	1.0	21.90	15.40	14.50	12.90	11.00	9.20
0.909	1.0	10.00	10.00	10.00	10.00	10.00	10.00
0	5.0	0	0	0	0	0	0
0.2	5.0	2.23	2.77	2.78	2.81	2.64	2.38
0.4	5.0	11.10	7.85	7.59	7.28	6.66	5.80
0.6	5.0	21.70	13.00	11.30	10.70	10.40	10.30
0.667	5.0	10.00	10.00	10.00	10.00	10.00	10.00

In Fig. 3, VAR T_C is shown dependent on the server utilization ρ_0 in case of $g = 2$ symmetrical queues of the types $M/D/1$ and $M/H_2/1$, each with two cases of constant overhead. As expected, the cycle time variance depends largely on the pdfs of the service and overhead times. The solid curves of the new approximation with the concept of conditional cycle times compare generally better with the simulation than the previous theory by Hashida and Ohara,[13] especially for low overhead. From a large number of computer simulations for $M/G/1$ systems, we made the following qualitative observations:

(*i*) The cycle time variance accuracy decreases with increasing number of queues and increasing service time variance.

(*ii*) The cycle time variance accuracy increases with increasing overhead and for approaching the low or heavy traffic region.

(*iii*) Observations (*i*) and (*ii*) apply to the new and old theory; the concept of conditional cycle times, however, yields generally a better accuracy.

Since the approximation for the pdf of the cycle time is independent of the arrival process type, it is interesting to know how the actual cycle time variance depends on various process types. For comparison, five different $GI/M/1$ systems with $g = 10$ queues (the accuracy is generally better for $g < 10$), two cases of overhead, and five cases of load have been considered (see Table I). Summarizing, we make the following observations:

(*i*) The cycle time variance depends indeed on the arrival process type. This dependence decreases, however, as the load approaches the low or the heavy traffic regions.

(*ii*) For medium loads, the cycle time variance may *decrease* as the arrival process peakedness increases.

(*iii*) The approximation generally underestimates the true cycle time variance. The accuracy increases with the overhead, the peakedness of the arrival process, and as the load approaches the low or heavy traffic region.

At first sight, observation (*ii*) is counterintuitive and surprising since the mean waiting time generally *increases* with the arrival process peakedness (see, for example, Fig. 9). However, regular arrival patterns may result in very short and very long cycles since many idle cycles could be produced after a service until the next arrival occurs.

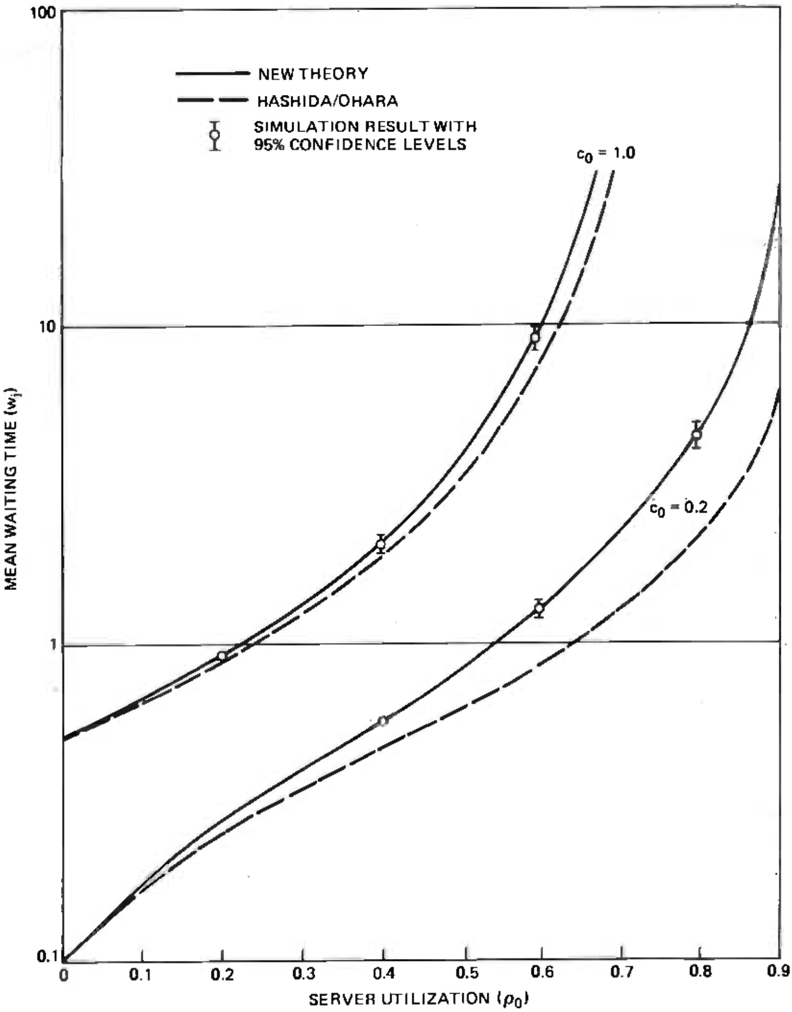


Fig. 4—Accuracy of mean waiting times for cyclic queuing systems $M/D/1$
Parameters: g = 2 symmetrical queues
$h_1 = h_2 = 1$ average service times
c_0 = 0.2 and 1.0, constant overhead.

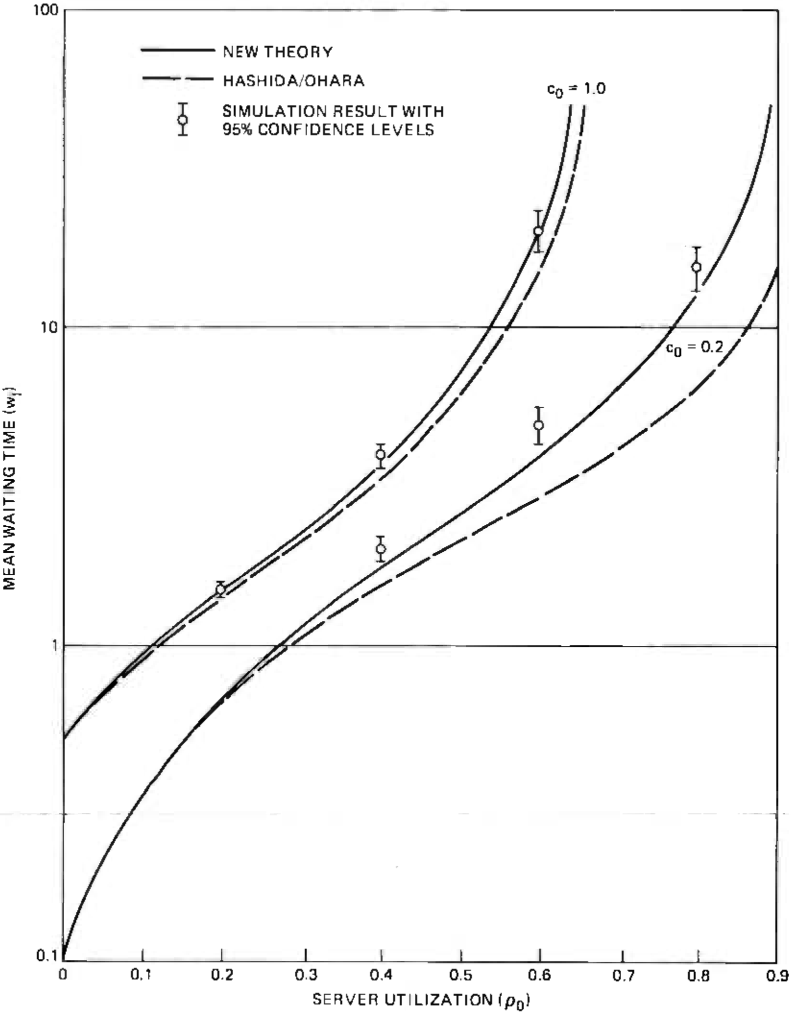


Fig. 5—Accuracy of mean waiting times for cyclic queuing systems $M/H_2/1$

Parameters: $g = 2$ symmetrical queues
$h_1 = h_2 = 1$ average waiting times
$c_0 = 0.2$ and 1.0, constant overhead
$c_H = 2.0$ coefficient of variation of service times.

On the contrary, batch arrivals may stabilize the cycle time since many cycles consist of one service time and the overhead only. Although these characteristics depend largely on the parameter combination, they indicate some interesting effects which may be important for applications and theory as well.

6.2 Mean waiting time

Since the mean waiting times in (26) are basically dependent on the first two moments of the conditional cycle times, we can expect the same accuracy trends as for the cycle time variance. Figures 4 and 5 show results for systems of the type $M/D/1$ and $M/H_2/1$ with two symmetrical queues for low and high overhead. The accuracy for $M/D/1$ is excellent, whereas for $M/H_2/1$ and low overhead the mean waiting time is underestimated. In any case, the new approach yields a better accuracy compared to Ref. 13, which results from the conditional cycle time concept.

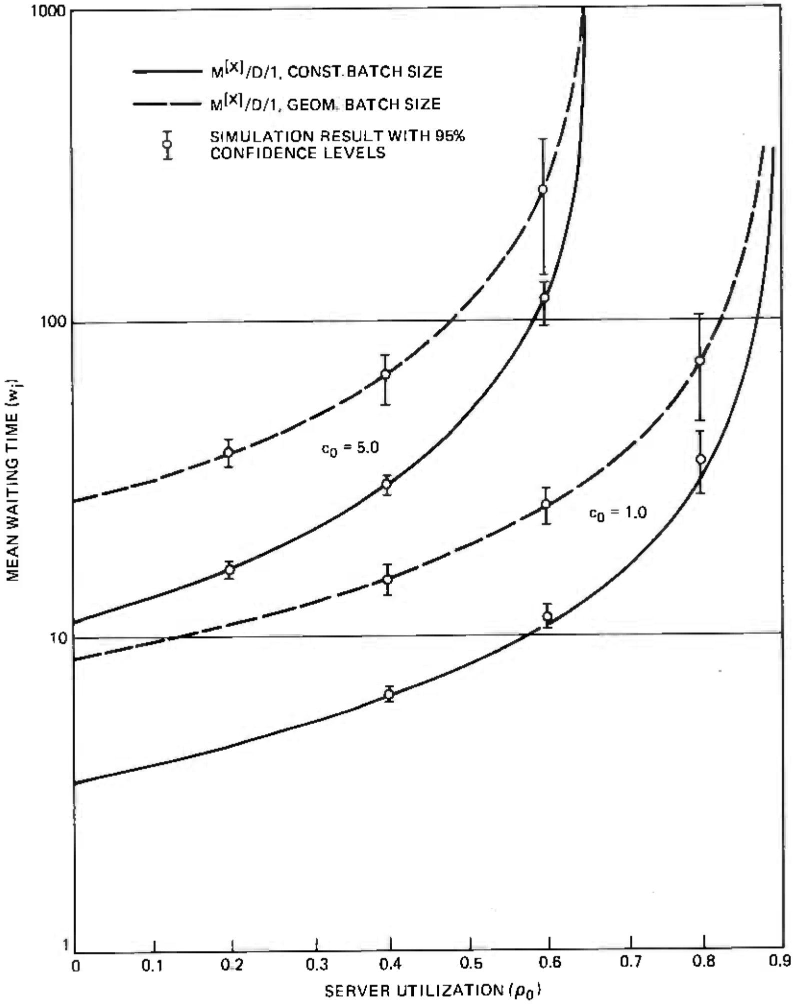


Fig. 6—Accuracy of mean waiting times for cyclic queuing systems $M^{[X]}/D/1$
Parameters: g = 10 symmetrical queues
h_j = 1 average service time, $j = 1, 2, \cdots, 10$
c_0 = 1.0 and 5.0, constant overhead
EK = 4 constant (average) batch size.

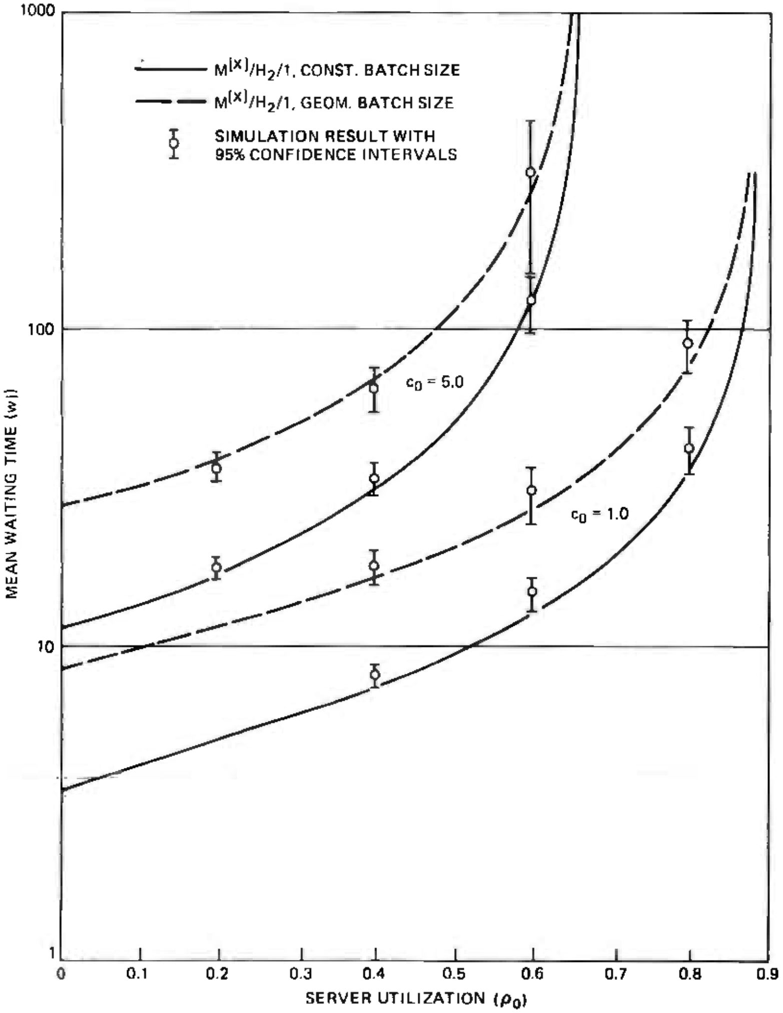


Fig. 7—Accuracy of mean waiting times for cyclic queuing systems $M^{[X]}/H_2/1$

Parameters: g = 10 symmetrical queues
h_j = 1 average service time, $j = 1, 2, \cdots, 10$
c_0 = 1.0 and 5.0, constant overhead
EK = 4 constant (average) batch size
c_H = 2.0 coefficient of variation of service times.

Figures 6 and 7 show the results for $g = 10$ symmetrical queues for systems $M^{[X]}/D/1$ (Fig. 6) and $M^{[X]}/H_2/1$ (Fig. 7), each with constant or geometrically distributed batch sizes, low and high overhead. All cases of batch arrival processes show an excellent accuracy. Many other validations have also shown that the accuracy is far less dependent on the parameters g, c_0, or G compared to single Poisson arrivals. This results from the fact that the cycle time analysis yields the best accuracy in case of batch arrivals; also, the contribution of the batch arrivals to the mean waiting time w dominates the expression (39) for larger batch sizes.

Another study on the influence of the service process type G and arrival process type GI on the mean waiting time w in case of ordinary cyclic service is shown in Figs. 8 and 9 for zero, low, and high overhead. The $M/G/1$ curves with overhead are analytic results according to (26), whereas the $GI/M/1$ curves are simulation results; the results for

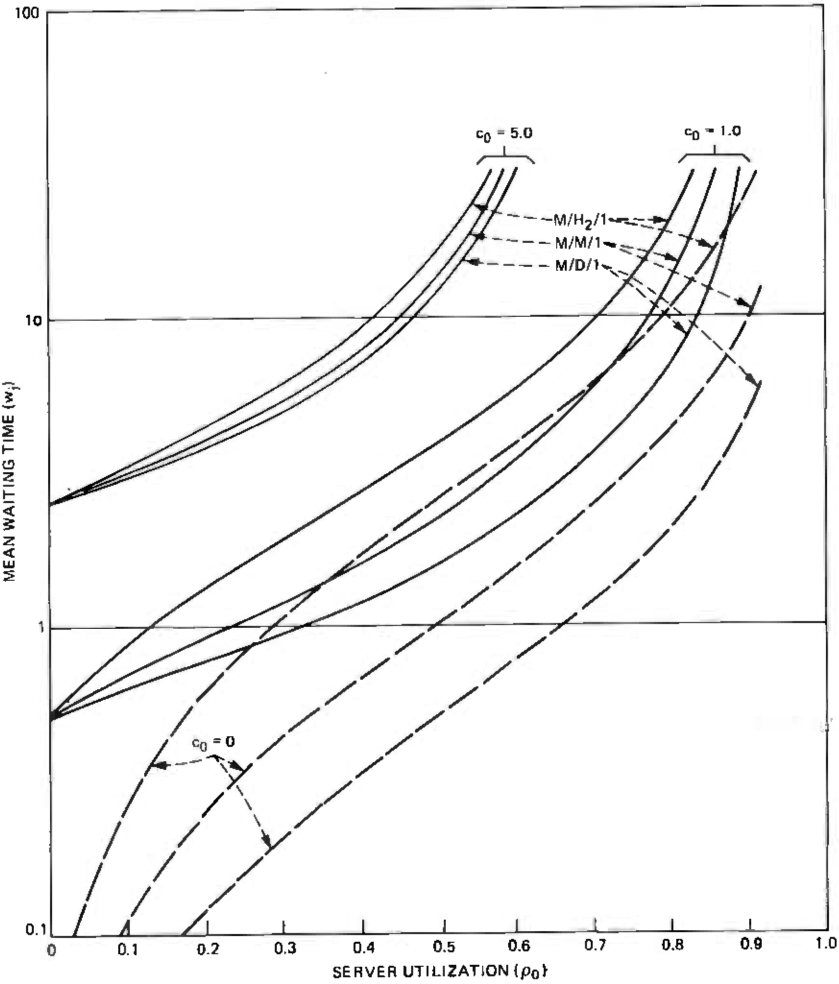


Fig. 8—Influence of service process type for cyclic queuing systems $M/G/1$
Parameters: g = 10 symmetrical queues
h_j = 1 average service time, $j = 1, 2, \cdots, 10$
c_0 = 0, 1.0, 5.0, constant overhead
Systems $M/D/1$, $M/M/1$, $M/H_2/1$ ($c_H = 2.0$).

zero overhead are exact and have been drawn from standard queuing tables by the author.[19] The main conclusions from Figs. 8 and 9 are:

(*i*) For $M/G/1$ systems with ordinary cyclic service, the influence of the service process decreases with increasing overhead.

(*ii*) For $GI/M/1$ systems with ordinary cyclic service, the influence of the arrival process does not remarkably decrease or may even increase with increasing overhead (see also Figs. 6 and 7 for batch arrivals).

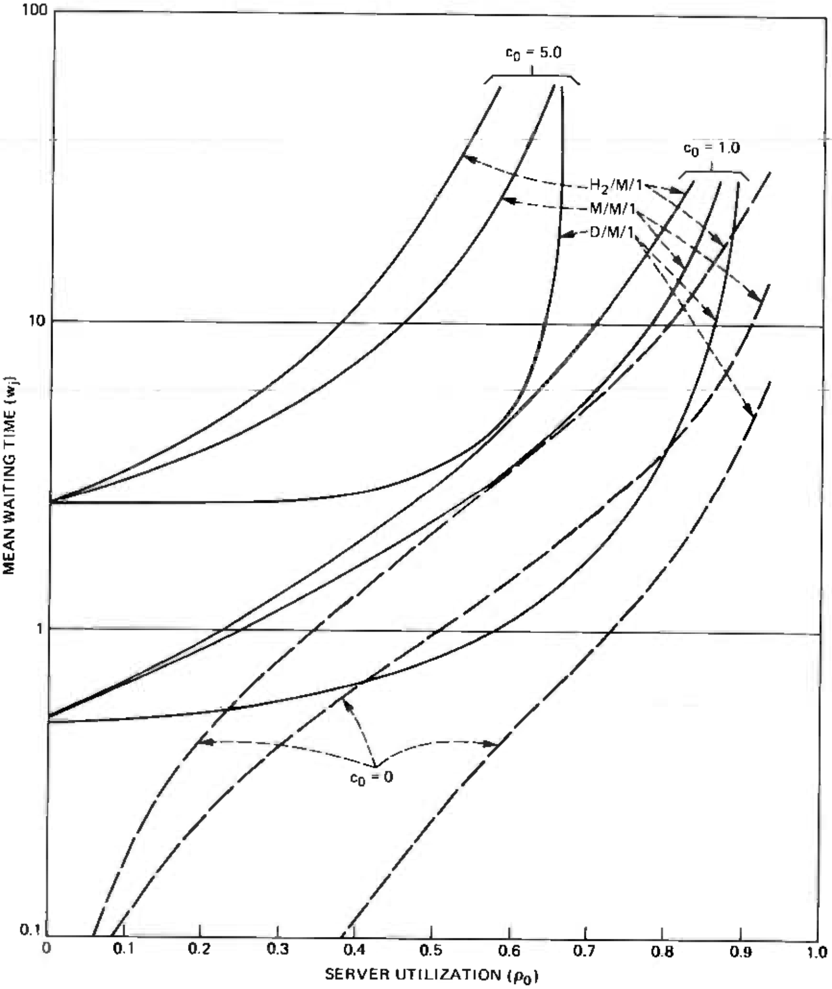


Fig. 9—Influence of arrival process type for cyclic queuing systems $GI/M/1$
Parameters g = 10 symmetrical queues
h_j = 1 average service time, $j = 1, 2, \cdots, 10$
c_0 = 0, 1.0, 5.0, constant overhead
Systems $D/M/1$, $M/M/1$, $H_2/M/1$ (c_A = 2.0).

These properties are most important for applications since they show that the results are much more sensitive to arrival processes than to service processes and that the usual approximation of arrival processes by Poisson processes may result in a quite dramatic error in the performance estimation. Therefore, future analytic studies on cyclic queuing systems should aim more to the generalization of arrival processes.

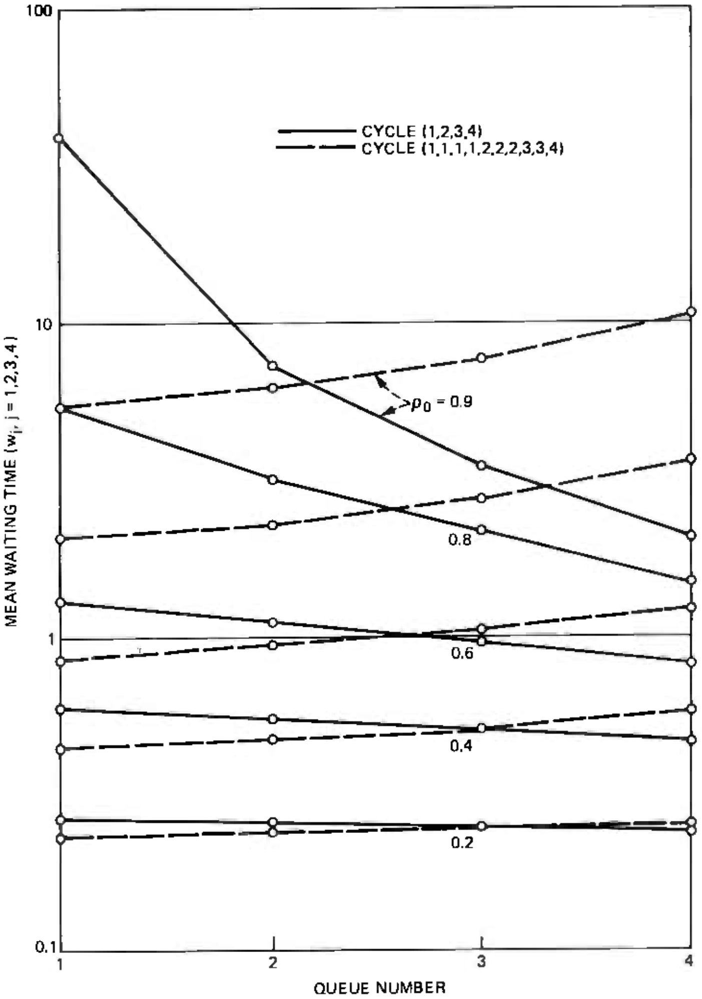


Fig. 10—Unbalanced load performance of cyclic queuing systems $M/D/1$
Parameters: g = 4 queues
h_j = 1, j = 1, 2, 3, 4, average service times
u_j = 0.05, j = 1, 2, 3, 4, constant overhead
$\lambda_1:\lambda_2:\lambda_3:\lambda_4 = 4:3:2:1$ arrival rate ratios.

6.3 Equity of service for unbalanced load

So far, we have concentrated on cases of ordinary cyclic service. The final study shows how cyclic priority service can be used to achieve equity of service in cases of unbalanced load. In this case, queue 1 is served up to l_1 times, queue 2 up to l_2 times, $\cdots$, queue g up to l_g times within a cycle, so that $l_1, l_2, \cdots, l_g$ could be considered as cycle priorities. Figure 10 demonstrates the use of cyclic priority service in case of unbalanced load in a system $M/D/1$ with $g = 4$ queues and arrival rate ratios $\lambda_1:\lambda_2:\lambda_3:\lambda_4 = 4:3:2:1$. In case of ordinary cyclic service with the cycle $\{1, 2, 3, 4\}$, the unbalanced load produces also unbalanced waiting times with increasing absolute load ρ_0. The dashed curves show the result of cyclic priority service where $l_1:l_2:l_3:l_4 = \lambda_1:\lambda_2:\lambda_3:\lambda_4$ with the cycle $\{1, 1, 1, 1, 2, 2, 2, 3, 3, 4\}$. For small ρ_0, both schedules do not remarkably differ in performance. In case of higher ρ_0, the unbalanced load effects can be compensated for by a cyclic priority service.

VII. CONCLUSION

This paper provides a new approximate analysis for cyclic queuing systems $M^{[X]}/G/1$ with batch Poisson arrivals, general service and overhead times, and ordinary cyclic service. The method allows a relatively easy evaluation of numerical results. The accuracy of the method has been validated by computer simulations. In addition to the analysis method, a new stability criterion for systems $GI/G/1$ with general cyclic service is developed. A number of traffic studies are reported revealing more insight in the traffic performance of cyclic queuing systems.

VIII. ACKNOWLEDGMENTS

The author wishes to thank J. M. Holtzman and H. Zucker for many stimulating discussions. The constructive comments by two reviewers are greatly appreciated.

REFERENCES

1. B. Avi-Itzhak, W. L. Maxwell, and L. W. Miller, "Queuing With Alternating Priorities," Opns. Res., *13* (1965), pp. 306–318.
2. L. Takács, "Two Queues Attended by a Single Server," Opns. Res., *16* (1968), pp. 639–650.
3. R. B. Cooper and G. Murray, "Queues Served in Cyclic Order," B.S.T.J., *48,* No. 3 (March 1969), pp. 675–689.
4. R. B. Cooper, "Queues Served in Cyclic Order: Waiting Times," B.S.T.J., *49,* No. 3 (March 1970), pp. 399–413.
5. J. S. Sykes, "Simplified Analysis of an Alternating-Priority Queuing Model with Setup Times," Opns. Res., *18* (1970), pp. 1182–1192.
6. M. Eisenberg, "Two Queues with Changeover Times" Opns. Res., *19* (1971), pp. 386–401.
7. M. Eisenberg, "Queues with Periodic Service and Changeover Time," Opns. Res., *20* (1972), pp. 440–451.

8. O. Hashida, "Analysis of Multiqueue," Review of the Electr. Comm. Laboratories, Nippon Telegraph and Telephone Public Corp., *20* (1972), pp. 189–199.
9. T. S. Brodetskaya, "Capacity-Time Analysis of a Cyclic-Service Queuing System." Kibernetica, No. 2 (1975), pp. 64–68 (in Russian). Translation: New York: Plenum, 1976.
10. S. S. Nair, "Alternating Priority Queues with Non-Zero Switch Rule," Comp. and Opns. Res., *3* (1976), pp. 337–346.
11. M. Eisenberg, "Two Queues With Alternating Service," unpublished work.
12. M. A. Leibowitz, "An Approximate Method for Treating a Class of Multiqueue Problems," IBM Journal, *5* (1961), pp. 204–209.
13. O. Hashida and K. Ohara,, "Line Accomodation Capacity of a Communication Control Unit," Review of the Electr. Comm. Laboratories, Nippon Telegraph and Telephone Public Corporation, *20* (1972), pp. 231–239.
14. S. Ekberg, "Strategies for a Telephone Central Marker Handling Different Queuing Jobs," Proceedings TIMS XX, Tel Aviv (1973), pp. 523–528.
15. P. Kuehn and M. Langenbach-Belz, "Effectiveness of Device Servicing Strategies in Real Time Systems," Fourth Annual Conference of the German Informatics Society, Berlin, October 9–12, 1974, Springer-Verlag, 1975, pp. 463–472. Bell Laboratories Translation Series No. TR 77-72.
16. S. Halfin, "An Approximate Method for Calculating Delays for a Family of Cyclic-Type Queues," B.S.T.J., *54*, No. 10 (December 1975), pp. 1733–1754.
17. R. B. Cooper, *Introduction to Queuing Theory*, New York: MacMillan, 1972.
18. D. Jagerman, "An Inversion Technique for the Laplace Transform with Application to Approximation," B.S.T.J., *57*, No. 3 (March 1978), pp. 669–710.
19. P. J. Kuehn, "Tables on Delay Systems," Institute of Switching and Data Technics, University of Stuttgart, 1976.
20. P. J Burke, "Delays in Single-Server Queues with Batch Input," Opns. Res., *23* (1975), pp. 830–833.

Copyright © 1979 American Telephone and Telegraph Company
THE BELL SYSTEM TECHNICAL JOURNAL
Vol. 58, No. 3, March 1979
Printed in U.S.A.

A Comparison of the Performance of Four Low-Bit-Rate Speech Waveform Coders

By J. M. TRIBOLET, P. NOLL, B. J. McDERMOTT, and R. E. CROCHIERE

(Manuscript received May 23, 1978)

Subjective ratings were obtained for four different speech waveform coders of varying complexity at each of three bit rates (24, 16, and 9.6 kb/s). ATC (adaptive transform coding) is rated the highest and ADPCM-F (adaptive differential PCM with a fixed predictor) the lowest, regardless of bit rate. Although there is large variability in the ratings due to different talkers and listeners, SBC (sub-band coding) is rated higher than ADPCM-F and about equal to ADPCM-V (adaptive differential PCM with a variable predictor) for most talkers and listeners. A weighted combination of two objective measures, one accounting for noise and the other for bandwidth effects, appears promising as a predictor of the subjective quality ratings.

I. INTRODUCTION

The quality of the reproduced speech from waveform coders can usually be improved by increasing their complexity. However, increasing the complexity also usually increases the cost. The practical problem in choosing a coder, at the current state of the art, becomes one of knowing how much loss in quality is sacrificed when opting for a less complex (less expensive) coder.

This paper compares subjective quality ratings for four different speech waveform coder algorithms of varying complexity. The algorithms, rated in order of their complexity are: ADPCM-F (adaptive differential pulse code modulation with a fixed predictor), SBC (sub-band coding), ADPCM-V (adaptive differential PCM with a variable predictor), and ATC (adaptive transform coding). Each of these four algorithms was studied at three different transmission rates: 24, 16, and 9.6 kb/s. These coders were chosen because they represent a number of different classes of coding techniques ranging from relatively simple (inexpensive) to highly complex (costly) schemes. The choice was also influenced by the availability of coder software at the time of the experiment. All coders are "non-pitch-predicting."

Since the coders being studied produce a broad range of qualities and types of degradations, the data also provided an opportunity to evaluate the relative merits of several objective measures that have been proposed for predicting quality ratings.[1]

II. THE WAVEFORM CODERS: A BRIEF DESCRIPTION

The four waveform coders used in this experiment are depicted in Figs. 1 through 4 and are briefly described below.

2.1 ADPCM with a fixed predictor (ADPCM-F)

The ADPCM-F coder is the simplest of the four coding techniques. As seen in Fig. 1, it consists of a quantizer with an adaptive step-size and a first-order fixed predictor. The step-size adaptation is based on the one-word memory approach of Jayant, Flanagan, and Cummiskey.[2] The number of bits used in the quantizers are 3, 2, and 1 bit, respectively, corresponding to the transmission rates of 24, 16, and 9.6 kb/s. In the case of the 9.6-kb/s transmission rate, the coder reduces to that of an adaptive delta modulator with a sampling rate of 9.6 kHz. The output of the 9.6-kb/s coder was filtered with a 0 to 2800-Hz lowpass filter to remove the high frequency noise. The parameters used for the ADPCM-F coders are close to those proposed by Jayant.[3]

2.2 Sub-band coding (SBC)

In the sub-band coder, the speech band is partitioned into sub-bands.[4,5] Each sub-band is effectively lowpass-translated and sampled at its Nyquist rate. It is then preferentially encoded using APCM

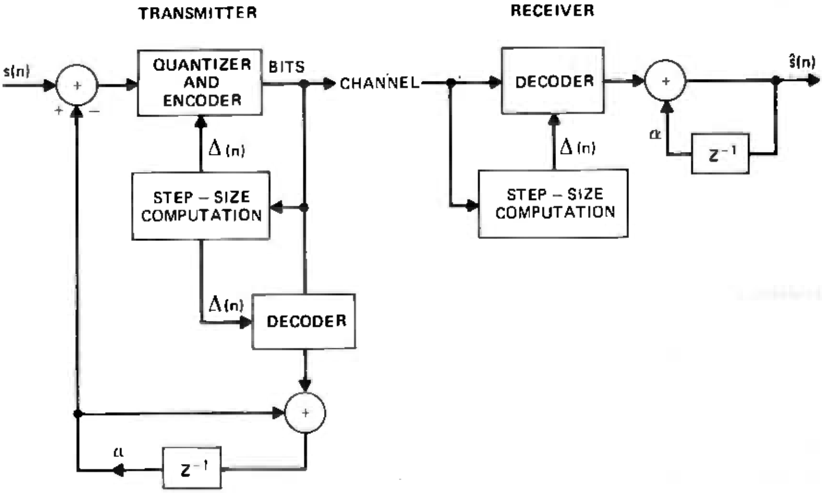


Fig. 1—Block diagram of ADPCM-F.

(adaptive step-size PCM) encoding with a backward step-size adaptation algorithm. The number of bits per sample in each band is chosen according to perceptual criteria for that band. On reconstruction, the sub-bands are decoded and bandpass-translated back to their original bands. They are then summed to give a replica of the original speech signal.

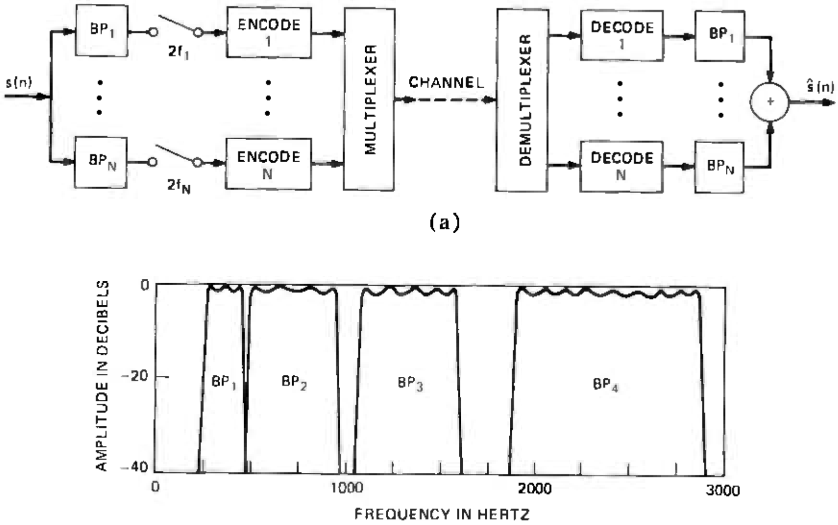
Figure 2a shows an implementation of the sub-band coder based on an integer-band sampling technique.[4] The speech band is partitioned into N sub-bands by bandpass filters BP_1 to BP_N. Typically, four to five bands are used and, at 9.6 kb/s, gaps are permitted between the bands to conserve bandwidth and, therefore, bit rate, as is illustrated in Fig. 2b.

The complexity of the sub-band coder is somewhat greater than that of the ADPCM-F coder. Using recent CCD (charge-coupled device) technology, the filters can potentially be implemented very efficiently. A single APCM coder can be multiplexed between the N sub-bands. The multiplexer requires digital logic and a ROM (read only memory) for storing the multiplexing pattern.

The coder configurations at 9.6 and 16 kb/s are those of examples A and D in Ref. 5. The 24-kb/s coder uses the same filters as the 16-kb/s coder, but the number of bits per sample per sub-band is increased by one.

2.3 ADPCM with a variable predictor (ADPCM-V)

The ADPCM-V coder is a more sophisticated version of ADPCM,[6,7] as seen in Fig. 3. The input data are first buffered and delayed. From this



(a)

(b)

Fig. 2—(a) Block diagram of SBC. (b) Sub-band partitioning.

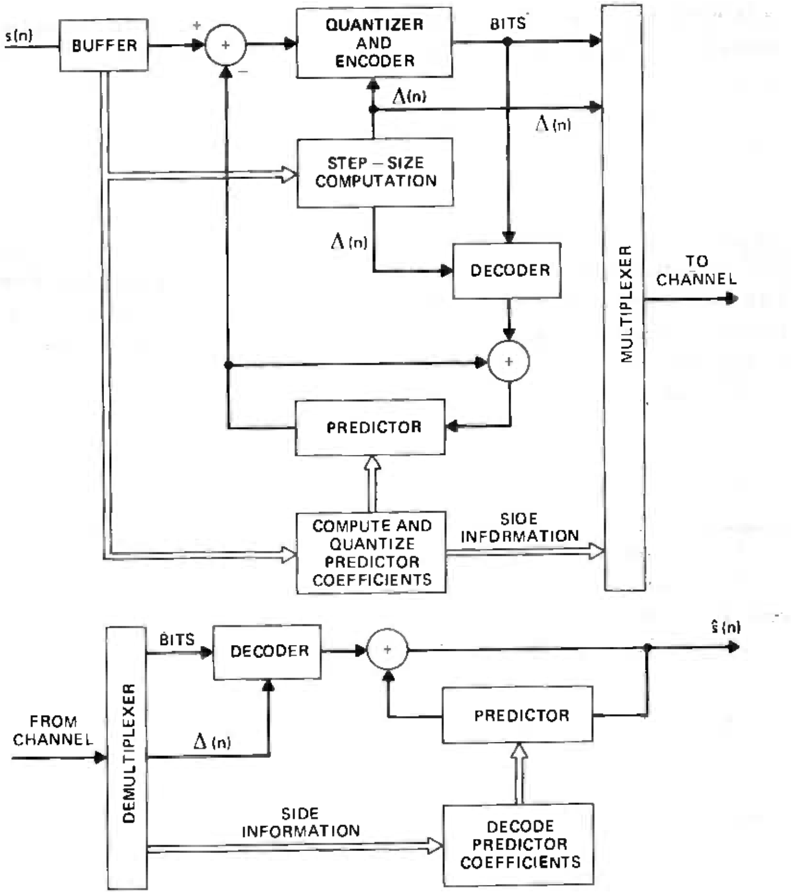


Fig. 3—Block diagram of ADPCM-V.

buffered block of speech, a short-time estimate of the variance of the input speech is computed and used to control the step-size of the quantizer. This local variance estimate is also quantized and transmitted for use in the receiver.

The predictor is an eighth-order adaptive predictor (no pitch prediction is involved in this scheme). The coefficients of the predictor are computed according to the relation

$$\mathbf{h}_N = \mathbf{R}_N^{-1}, \mathbf{r}_N, \tag{1}$$

where $\mathbf{R}_N$ and $\mathbf{r}_N$ are the matrix and vector of autocorrelation coefficients of the data in the buffer. The predictor coefficients are also quantized and transmitted to the receiver as side information. The total transmission of side information requires about 2 kb/s of data.

In the implementation of the ADPCM-V coder, 3, 2, and 1 bits/sample were used for the respective transmission rates. A sampling rate of

8 kHz was used. Since approximately 1.5 to 2 kb/s of additional information are required to be transmitted with this scheme, the actual transmission rates represented by this coder were 26, 18, and 9.6 kb/s instead of 24, 16, and 9.6 kb/s, which was used in the other coders.

The complexity of the ADPCM-V coder is primarily dominated by the implementation of the adaptive predictor and the computation of the predictor coefficients. As seen in eq. (1), this involves an autocorrelation computation and the solution of 8 simultaneous linear equations every 8 to 16 ms. This must be done using high-speed digital computation. Therefore, the complexity of the ADPCM-V coder is substantially greater than that of the ADPCM-V or sub-band coders.

2.4 Adaptive transform coding (ATC)

The adaptive transform coder is analogous in some respects to the sub-band coder in that it divides the speech band into a number of frequency components.[8] The resolution (number of bands), however, is generally much finer than that used in the sub-band coder, and the translation to the frequency domain is achieved by means of a fast transform algorithm. The transform is a 128-point, discrete, cosine transform. The transformed coefficients are encoded with APCM encoding.

Figure 4 is a block diagram of the transform coder. The input speech

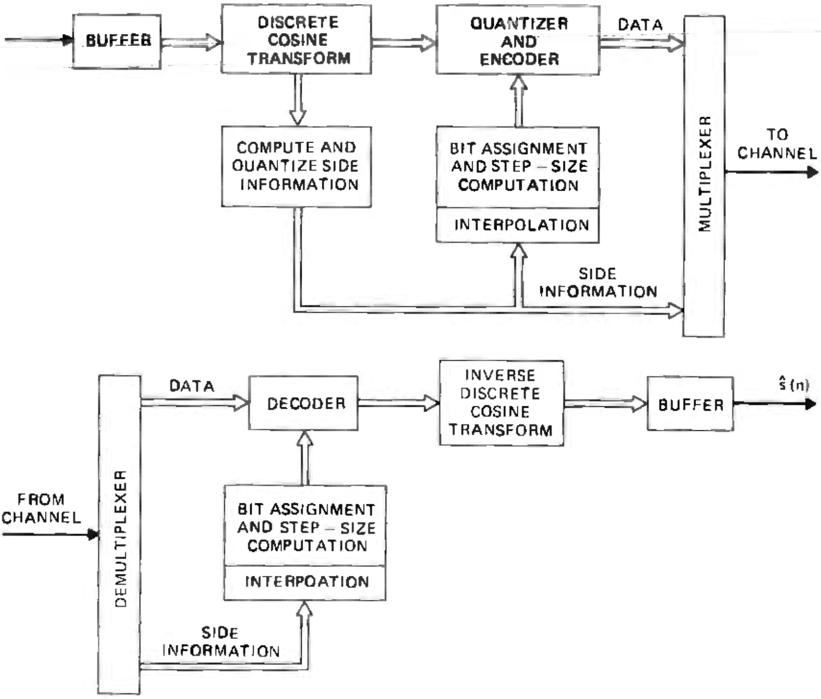

BUFFER
DISCRETE COSINE TRANSFORM
QUANTIZER AND ENCODER
DATA
COMPUTE AND QUANTIZE SIDE INFORMATION
BIT ASSIGNMENT AND STEP – SIZE COMPUTATION
INTERPOLATION
MULTIPLEXER
TO CHANNEL
SIDE INFORMATION
DATA
DECODER
INVERSE DISCRETE COSINE TRANSFORM
BUFFER
$\hat{s}(n)$
FROM CHANNEL
DEMULTIPLEXER
BIT ASSIGNMENT AND STEP – SIZE COMPUTATION
INTERPOATION
SIDE INFORMATION

Fig. 4—Block diagram of ATC.

is buffered and transformed in blocks of data. The output values of the transform are smoothed and decimated to 16 values. These 16 values are quantized and sent as side information to the coder and the decoder, where they are interpolated, yielding a smoother version of the short-time spectrum. This smoothed spectrum forms the basis for a dynamic bit assignment and step-size adaptation of the quantizers, as a function of frequency. The quantizers are then used to encode the transform coefficients.

In the receiver, a similar bit assignment and step-size computation is performed and the transform coefficients are decoded. The decoded values are then inverse-transformed to give a replica of the input speech.

The degree of complexity of the transform coder is of approximately the same order as the ADPCM-V coder. The side information and bit assignment computation require high-speed digital computations. Some efficiency might be gained using CCD technology to implement the transform.

III. SUBJECTIVE EVALUATION

3.1 Experimental procedure

Digital recordings of sentences spoken by four talkers (two male and two female) were processed by each of the 12 coders. The processed utterances were equalized to the same mean power to eliminate loudness differences. Two analog test tapes were prepared that contained different permutations of four random orderings of the 12 coders. The talkers were assigned in a balanced design so that each coder was represented by the speech of a different talker in each of the random orders. Since each of the four talkers had recorded a unique set of eight sentences, the sentences were randomly assigned and none occurred more than twice.

Students from the junior and senior classes of local high schools served as paid subjects. They listened to the processed speech binaurally over Pioneer SE700 earphones while seated in a double-walled sound booth. Sixty-five subjects judged the 48 coded sentences (4 coders × 3 bit rates × 4 talkers). They were asked to rate the quality of each sentence on a scale from 1 to 9, using a 1 to represent the worst quality, 9 to represent the best quality, and the numbers between 1 and 9 for intermediate evaluations. Before the test session began, they judged six representative conditions for practice to familiarize them with the task and the range of quality.

3.2 Results

An initial analysis of the ratings by the 65 listeners revealed a large amount of unexpected variability in the data due to the different talkers and listeners.

The variability due to the different listeners is illustrated in Fig. 5. These histograms show the percent of the listeners who assigned each of the nine possible ratings to each of the 12 coders. While these values were computed by summing across the four talkers, comparable plots for the individual talkers produced essentially the same general pattern, indicating that the variability in the ratings cannot be attributed entirely to the effects of the different talkers. The extremely skewed distributions for ATC at 24 kb/s and ADPCM-F at 9.6 kb/s show strong agreement among the listeners about which coders had the best and worst qualities, but the wide range, and in some cases almost flat character, of other distributions indicate that the listeners differ in the

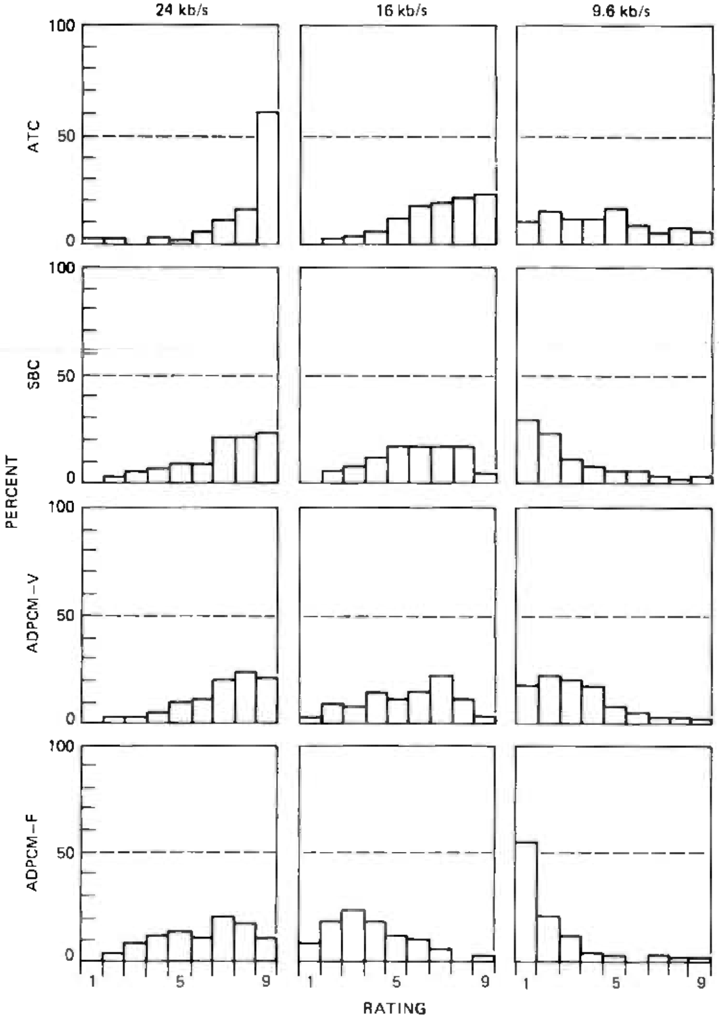


Fig. 5—Percent of ratings assigned each coder (65 listeners × 4 talkers).

amount and type of distortion that they will tolerate. However, it is not the purpose of this paper to examine the sources of listener variability but to gain some information about general trends and the trade-off between complexity and quality. Since so many of the distributions showed a lack of unimodality, the median ratings were used to compare the evaluations according to the experimental variables: type of coder, bit rate, and talkers.

The variability in the ratings due to the different talkers is shown in Fig. 6, where the median rating (bracketed by their 0.95 confidence interval) of the 65 listeners for the 12 coders are plotted for each of the four talkers. The overall pattern for all talkers shows that ATC had the highest rated quality, ADPCM-F the lowest rated quality, and the rating for each talker-coder combination drops as the bit rate is reduced. The ratings for SBC and ADPCM-V are about the same for all voices except female 2, indicating that they have about the same overall quality. For the same three voices, ATC at 16 kb/s is rated equal to, or better than, the other three coders at 24 kb/s.

Some specific differences in ratings are consistent with the usual male-female voice distinctions, and some are speaker idiosyncratic. ATC at 16 kb/s is rated almost equal to 24 kb/s for male voices but much lower for female voices. ADPCM-V at 16 kb/s is rated almost equal to 24 kb/s for female 1 but almost three categories lower for female 2. The coders span a much wider range of ratings at the lower bit rates for male 2 than for male 1. The separation of the four coders into only two classes for female voice 2 is a unique result in this small sample of four voices, but an analysis of the characteristics of her voice could provide important information about the effect of different voices in digital coding techniques.

IV. OBJECTIVE MEASURES

Several objective measures that may be more sensitive to the types of degradations produced by waveform coders are described in Ref. 1. The data from this experiment provided an opportunity to test these theoretically derived predictors of quality ratings. The median ratings (bracketed by the 0.95 confidence interval) computed across both listeners and talkers are plotted in Fig. 7 and served as a single basis for comparing the efficacy of these various objective measures.

4.1 Segmental S/N ratio

Perhaps the most widely used measure of performance has been the conventional signal-to-noise ratio, although it has not generally been found to be a very good indicator of subjective quality. An improved measure, proposed by Noll,[1,9] averages the s/n ratio in short (20 to 30

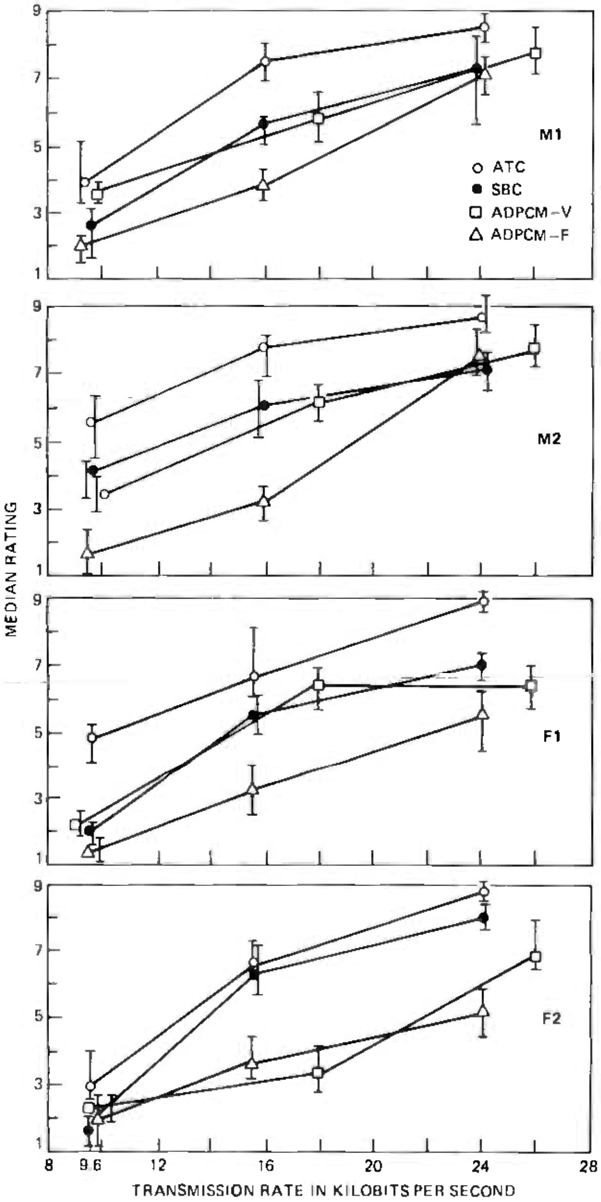


Fig. 6—Median rating of 12 coders for each talker.

ms) segments and has been found to be a more accurate predictor. In Fig. 8, the segmental s/n ratio, averaged across talkers, is plotted for each coder as a function of bit rate. These plots show that segmental s/n ratio is linearly related to the bit rate for each type of coder except

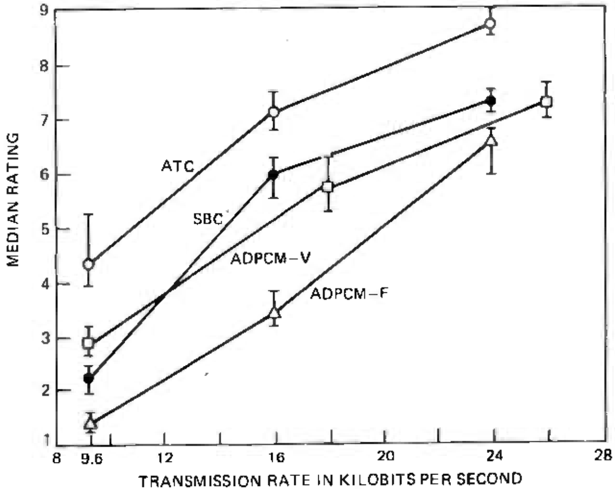


Fig. 7—Median rating of 12 coders (65 listeners × 4 talkers).

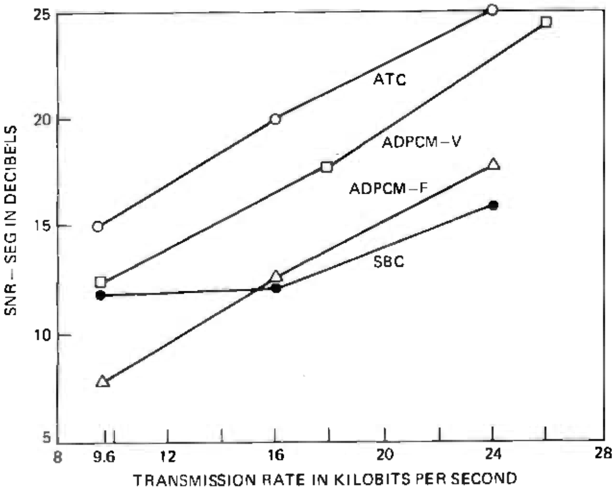


Fig. 8—Signal-to-noise ratio measured segmentally (15–20 ms) as a function of transmission rate for 4 types of coders.

SBC at low bit rates. While it orders ATC and the two ADPCM coders relative to each other at each bit rate, it underestimates the quality ratings of SBC shown in Fig. 7. In an effort to improve the effectiveness of s/n ratio as an indicator of quality, a number of frequency-weighted s/n ratio measures were computed, but they also were not highly successful in predicting the relative ordering of these four coders.[1,10]

4.2 Noise-to-signal measure

Another functional form for measuring the noise-to-signal power ratio in coders has been recently proposed[1] and analyzed.[10] This measure was derived through concepts of log likelihood ratios and has the following functional form:

$$\overline{l_m = 10 \log_{10} \left[1 + \sum_{j=1}^{B} e_j^2/s_j^2 \right]}, \tag{2}$$

where the summation is taken over $B = 16$ frequency bands spaced according to the articulation bands in the range of 200–3200 Hz.[1,10] The bar above the equation denotes that l_m is the result of an average over (20 to 30 ms) segmental measurements, where e_j^2 is the segmental noise power and s_j^2 is segmental signal power in band j.

The values of l_m, averaged over talkers, are plotted as a function of bit rate in Fig. 9a. This measure is a better predictor of the relative

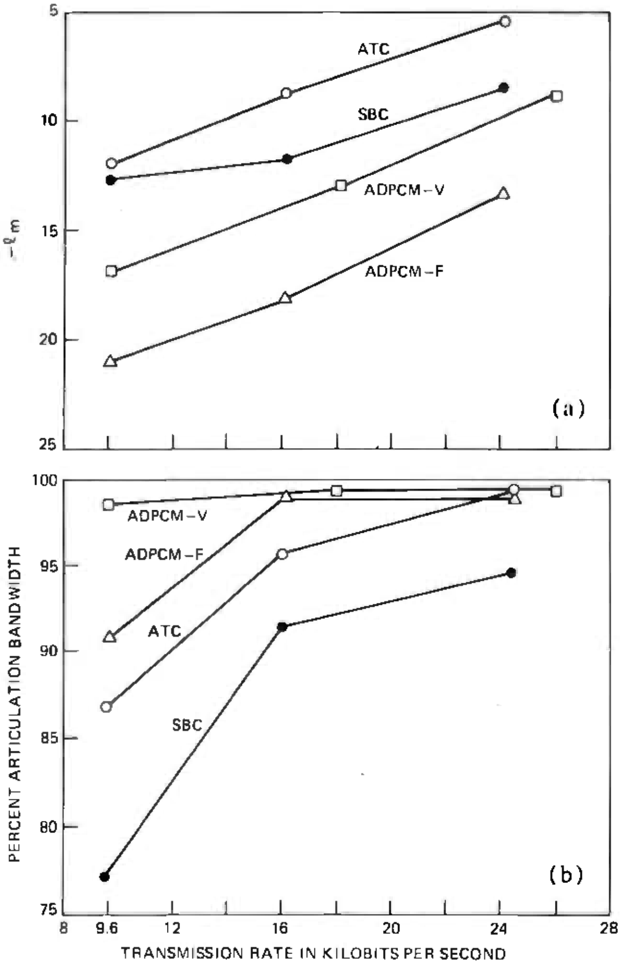


Fig. 9—(a) Noise-to-signal measure averaged over 16 articulation bands. (b) Percent of articulation bandwidth.

ordering of the coders at 24 and 16 kb/s where the distortions are primarily due to noise effects. At 9.6 kb/s, additional distortions due to bandwidth limitations are introduced by some of the coders. These distortions are not accounted for by the l_m measure and do not predict the subjective ratings at this bit rate.

4.3 Percent bandwidth

A bandwidth measure was defined[1,10] to account for the loss of bandwidth by the coders at low bit rates. The measure is defined as a simple percentage of the bandwidth on an articulation scale. Thus, the loss of a fixed bandwidth at a lower frequency has more importance than at a higher frequency. A coder with a flat frequency response from 200 to 3200 Hz is defined as one with a 100-percent articulatory bandwidth.

The bandwidth is measured on a segmental (32 ms) basis, and a smoothed spectral estimate of the input and output of the coder is computed for each segment. If, at any frequency point, the output power is less than, say, 10 dB of the input power, it is counted as a loss of bandwidth and is weighted according to the articulation scale. An average bandwidth is then computed over the entire utterance.

The percent bandwidth, plotted in Fig. 9b, complements the l_m measure in that it is sensitive to differences at 9.6 kb/s and relatively insensitive to differences at 16 and 24 kb/s. This measure correctly indicates that SBC at 9.6 kb/s has the lowest bandwidth due to the spectral gaps depicted in Fig. 2b. It also reflects the lower bandwidth of ATC at 9.6 kb/s due to the dynamic bit allocation in the algorithm and the 0 to 2800 Hz low-pass filter for ADPCM-F at this bit rate.

4.4 A combined measure

By forming a linear combination of the l_m and B_w (bandwidth) measures, a combined measure, Q, which accounts for both noise and bandwidth effects, can be defined. This measure has the form

$$Q = A_1 + A_2\, l_m + A_3\, B_w, \tag{3}$$

where A_1, A_2, and A_3 are computed by multiple regression techniques. The results of this combined measure for $A_1 = -3.78$, $A_2 = -0.42$, and $A_3 = 0.15$ are shown as the abscissa in Fig. 10, and the median ratings (of the 12 coder-bit rate combinations) as the ordinate. The solid dots in the scatterplot are the median ratings of the listeners summed over talkers and are the values used to compute the weighting coefficients. The four Δ's around each point are the median ratings for each of the four talkers. The high correlation (0.98) shows that this combined measure is a good predictor of the median ratings but will seriously err in predicting the ratings of some specific talker-coder combinations, particularly at low bit rates.

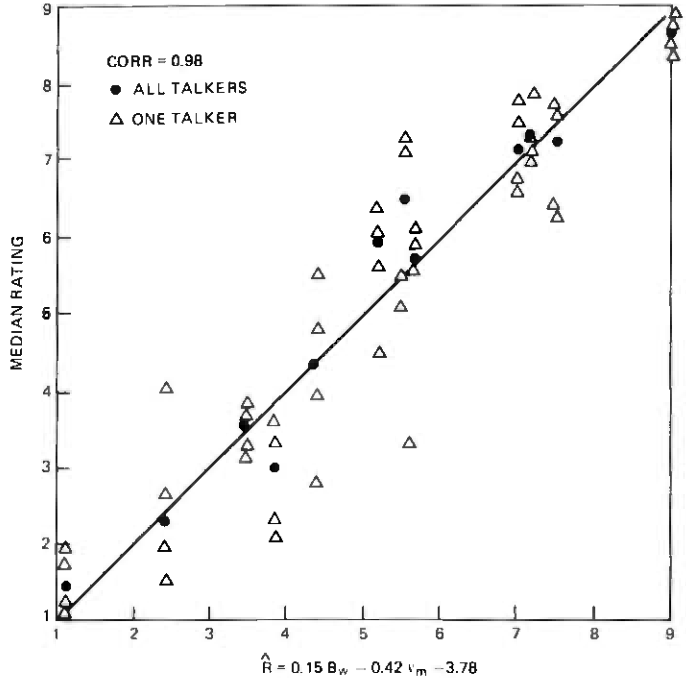


Fig. 10—Prediction of median rating by weighted combination of noise-to-signal and percent bandwidth measures.

V. DISCUSSION

Although the large variability in the ratings due to the different talkers and listeners rendered the quality ratings less precise than anticipated, an underlying pattern of relationships can be detected. ATC has the highest rating and ADPCM-F the lowest rating, regardless of bit rate, for most talkers and listeners. If cost/complexity is of no concern, then ATC is the most attractive of these coders. If cost/complexity is of concern, then SBC is an attractive choice since it is only slightly more complex than the ADPCM-F coder and about equal in quality to the costlier ADPCM-V coder.

The combination of two objective measures, l_m and B_w, each accounting for a different type of degradation, appears to be promising as a predictor of subjective ratings. Its precision is obscured by the interactive effects of the talkers and listeners. Of these two effects, the talker interaction is probably the easier to eliminate. The characteristics of different voices can be studied, and possibly the factors affecting the different coders could eventually be identified. However, the listener variability is a more difficult problem. Digital coding techniques are producing a variety of new and different types of degradations. As the bit rate is reduced, ATC produces a burble, SBC produces a reverberant quality, and the ADPCM coders produce a signal-dependent noise. The variability in the subjects' ratings indicates that the trade-off for these different types of degradations is not the same for all people.

REFERENCES

1. R. E. Crochiere, L. R. Rabiner, N. S. Jayant and J. M. Tribolet, "A Study of Objective Measures for Speech Waveform Coders," Proceedings of the 1978 Zurich Seminar on Digital Communications, Zurich, Switzerland, March, 1978, pp. H1.1–H1.7.
2. P. Cummiskey, N. S. Jayant, J. L. Flanagan, "Adaptive Quantization in Differential PCM Coding of Speech," B.S.T.J., *52,* No. 7 (September 1973), pp. 1105–1118.
3. N. S. Jayant, "Digital Coding of Speech Waveforms: PCM, DPCM, and DM quantizers," Proc. IEEE, *62* (May 1974), pp. 611–632.
4. R. E. Crochiere, S. A. Webber, and J. L. Flanagan, "Digital Coding of Speech in Subbands," B.S.T.J., *55* No.10 (October 1976), pp. 1069–1085.
5. R. E. Crochiere, "On the Design of Sub-Band Coders for Low Bit Rate Speech Communication," B.S.T.J., *65,* No. 5 (May–June 1977), pp. 747–770.
6. P. Noll, "Non-adaptive and adaptive differential pulse code modulation of speech signals," Polytechnisch Tijdschrift, Den Haag, 1972, No. 19, pp. 623–629.
7. P. Noll, "Untersuchungen zur Sprachcodierung mit adaptiven Prädiktionsverfahren," NTZ, *27* (1974), pp. 67–72.
8. R. Zelinski and P. Noll, "Adaptive Transform Coding of Speech Signals," IEEE Trans. Acoust. Speech and Sig. Proc., *ASSP-25* (August 1977), pp. 299–309.
9. P. W. Noll, "Adaptive Quantization in Speech Coding Systems," Int. Zurich Seminar on Digital Communication (October 1974), pp. B3.1 to B3.6.
10. J. M. Tribolet, P. Noll, B. J. McDermott, and R. E. Crochiere, "A Study of Complexity and Quality of Speech Waveform Coders," Proc. 1978 IEEE Int. Conf. on Acoustics, Speech, and Signal Proc., April 10–12, 1978, Tulsa, Okla. pp. 586–590.

Copyright © 1979 American Telephone and Telegraph Company
THE BELL SYSTEM TECHNICAL JOURNAL
Vol. 58, No. 3, March 1979
Printed in U.S.A.

LED Array Package for Optical Data Links

By A. ALBANESE and W. S. HOLDEN

(Manuscript received September 8, 1978)

A package with an array of six LEDs *is described. The design represents a first effort at reducing the cost and the size of electrooptical interfaces for applications in low-bit-rate transmission systems and computer data links. A package containing six GaAs* LEDs, *electrical connections, and a new fiber ribbon connector was fabricated, mounted, and encapsulated using presently available technology. Each* LED *in the array, with 80 mA dc applied, can couple an average optical power of 5 μW into a graded-index fiber having 0.22 N.A. and 55-μm core diameter. Cross-modulation between adjacent* LED *signals is smaller than −50 dB. Fiber ribbon connectors fabricated with matching plates which align the fibers with the light output from the* LEDs *showed insertion losses as low as 0.4 dB.*

I. INTRODUCTION

Optical fiber technology is of potential use in data links for computers and switching systems where optical fibers can replace existing copper wires.[1] In most of these applications, the space and the cost of the electro-optical interfaces can be decreased, as in the case of integrated circuits (ICs), by using a batch fabrication process and by having several components share the same package. A further reduction in cost and space would result from the integration of several optical devices and several ICs into a single package. This integration poses a new challenge for IC packaging where this package must include electronic components, heat sink, electrical connections, electro-optical interfaces, and optical connections.

In the present work, a package with an array of six LEDs is presented as an initial effort toward the integration of several electro-optical interfaces that share a single package as a way of reducing space and fabrication cost.

Figure 1 is a drawing of the experimental LED array package fabricated using standard dual-in-line packaging (DIP) techniques. The package has an array of six homojunction GaAs LEDs of the Burrus

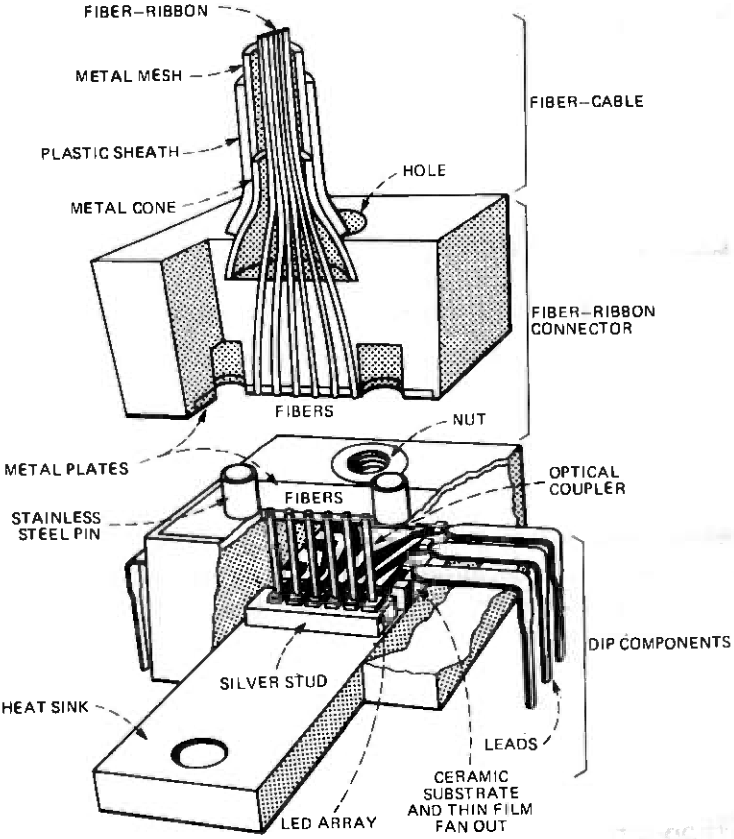


Fig. 1—LED array package.

type,[2] a thin-film fan-out to provide electrical connections, a screw-mountable copper heat-sink that supports the package and also serves as a ground terminal, and a new fiber ribbon connector that couples the light from the LEDs into an array of six fibers.

The LEDs described in this paper were fabricated on 625-μm centers with 60-μm-diameter emitting areas; however, arrays have also been made with 375-μm centers. Each diode in an array produced ≈3-mW total output power into air at 300-mA driving current. The variation in output from diode to diode in a given array is ±0.25 dB. The light output is linear in the current range of ≈25 mA to ≈250 mA. A component spacing of 625 μm was chosen for the LEDs in the transmitter array, the photodetectors in the receiver array, and the fibers in the fiber cable. Having a standard spacing will decrease the design and production costs of such a package and simplify the optical connection between the package and the fiber cable.

The optimum number of LEDs in an array will depend on particular system requirements, fabrication yield of the array package, and the reliability of the LEDs that form the array. A design consisting of six LEDs per package was selected to prove the feasibility of packaging an array of LEDs, to explore its performance, and to determine some fabrication problems that may limit the density of devices in a package.

To facilitate the description, the following discussion of the package design deals separately with the LED array, the DIP part, the optical coupler, and the fiber ribbon connector (see Fig. 1).

II. LED ARRAY

Figure 2 shows an array of six GaAs LEDs of the Burrus type fabricated using a batch process technique. A plasma-deposited silicon nitride film, 750 Å thick, is employed as the Zn-diffusion mask in

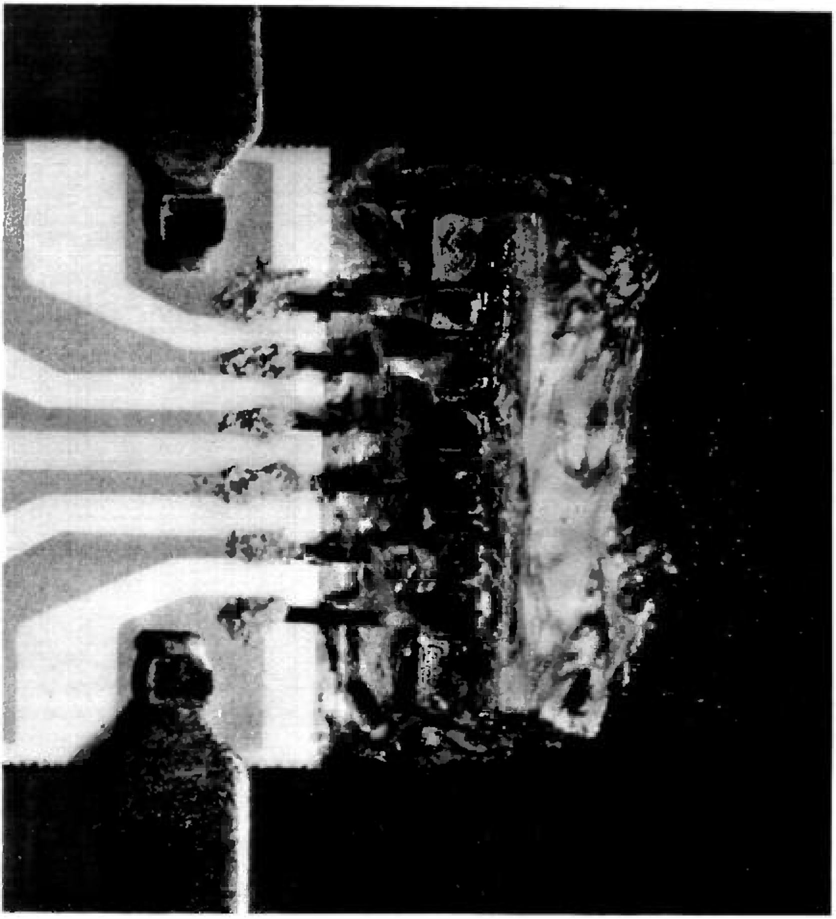


Fig. 2—LED array.

forming the planar p–n junctions.[3] The planar-diffused junction approach is ideal for lateral current confinement in the light-emitting area and thus minimizes the cross modulation between adjacent LEDs.

Burrus LEDs were selected because they were available in the laboratory, but arrays with other types of LEDs or lasers may be implemented.[4,5] However, the chemically etched "well" in the Burrus LEDs may facilitate the alignment of the fibers provided the emitting areas of the LEDs are centered within the "wells."

The LEDs have a 40-MHz bandwidth (3-dB point), and the intermodulation between the light intensity of two fibers placed in front of two adjacent LEDs is lower than −50 dB.

III. DUAL-IN-LINE PACKAGE (DIP) COMPONENTS

The LED array shown in Fig. 3 was mounted using standard IC technology. The stud of silver supporting the LED array is indium soldered to a copper plate that serves as the heat sink and as the electrical connection for all the anodes of the LEDs. Gold-plated silver

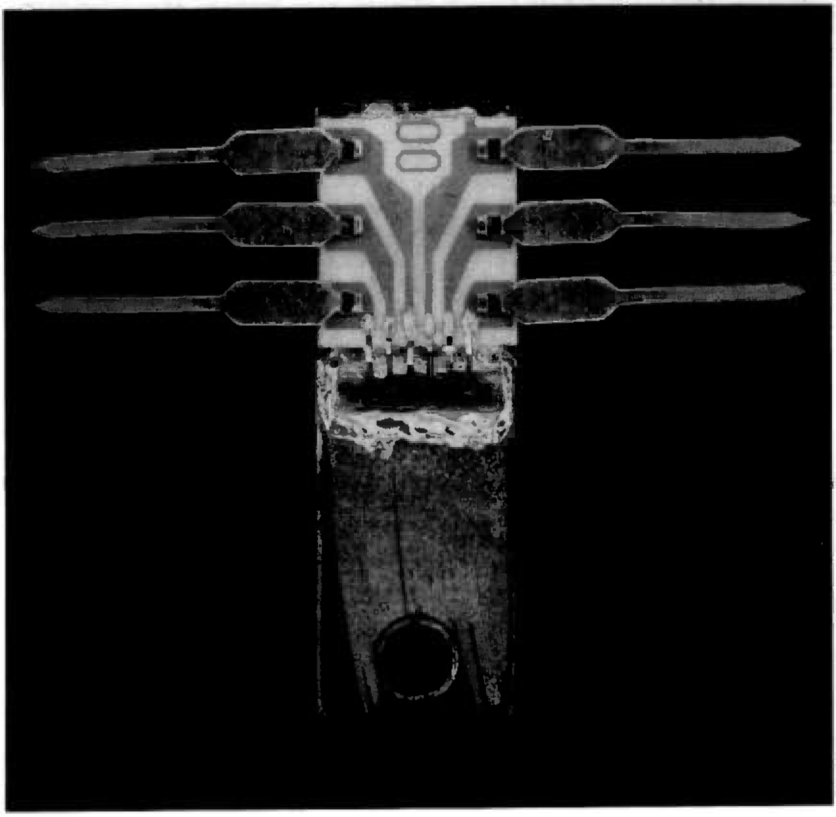


Fig. 3—LED array after being mounted on the copper plate.

tabs connect the cathode of every LED to a thin-film fan-out mounted on a ceramic board with copper leads. There is space in the package for several ICs that can serve as drive circuits for the LEDs.

IV. OPTICAL COUPLER

The optical coupler, which consists of six fiber stubs, serves the following two functions: first, it provides a protective housing for the LEDs and any associated circuitry, and second, it provides a means of light guidance from the LEDs to the fiber ribbon connector.

Several optical couplers have been implemented to improve the coupling efficiency between LEDs and fibers. Selfoc lenses,[6] glass spheres,[7] and tapers are all possible solutions. For an initial trial, short pieces (5-mm long) of graded-index fiber, similar to those in the fiber cable, were used because they were readily available. The characteristics of the fiber are: 0.22 N.A., 55-μm core diameter, and 110-μm overall fiber diameter.

Before inserting the LED array into the mold for the casting operation, the short pieces of fiber were mounted on the LEDs according to the following steps. The LED array is placed on a holder, and a current of 50 mA is applied to each LED. A 2-meter fiber with flat ends couples the light from one of the LEDs in the array to a power meter. The position of the fiber in front of the LED is adjusted until a maximum reading of the power meter is obtained. Then the fiber end is glued to the LED using a photopolymer (Norland Optical Adhesive No. 61). The fiber is subsequently fractured with a diamond scriber to yield a 5-mm length. This process is repeated for each LED in the array.

V. FIBER RIBBON CONNECTOR

A new demountable connector was designed to join a fiber ribbon to the package. The connector utilizes a metal plate with eight holes. Two large holes, 1.574 mm in diameter, serve as the guides for two stainless steel pins which align the two parts of the connector; six small holes, 114 μm in diameter, align the fibers on 625-μm centers. The position accuracy of the fiber must be better than ±2.5 μm to achieve an insertion loss objective for the optical connection of less than 0.5 dB.[8]

The first prototypes of the connector were made of stainless steel by machining the plates in pairs. The insertion loss measured on connections made of matched plates was 0.4 dB. However, insertion losses as high as 5 dB were measured on connections with plates of different pairs.

VI. CASTING

Figure 4 shows the mold used to cast the package of the LED array. The mold is made of brass and the movable parts are made of stainless steel. The metal plate with two stainless steel pins and six small holes aligns the fibers 625 μm apart. The post inside the mold serves to position a nut before casting. The two side plugs cover the leads of the array during casting.

The LED array with the six short pieces of glass fiber attached is placed into the mold, such that the six fibers enter into the six alignment holes of the stainless steel plate. Finally, the mold is covered and the casting material (epoxy, Bacon compound 84 plus an activator BA-63) is poured into the mold at room temperature.

After a curing time of 24 hours, the LED array package is removed from the mold. The metal plate with two stainless steel pins and the

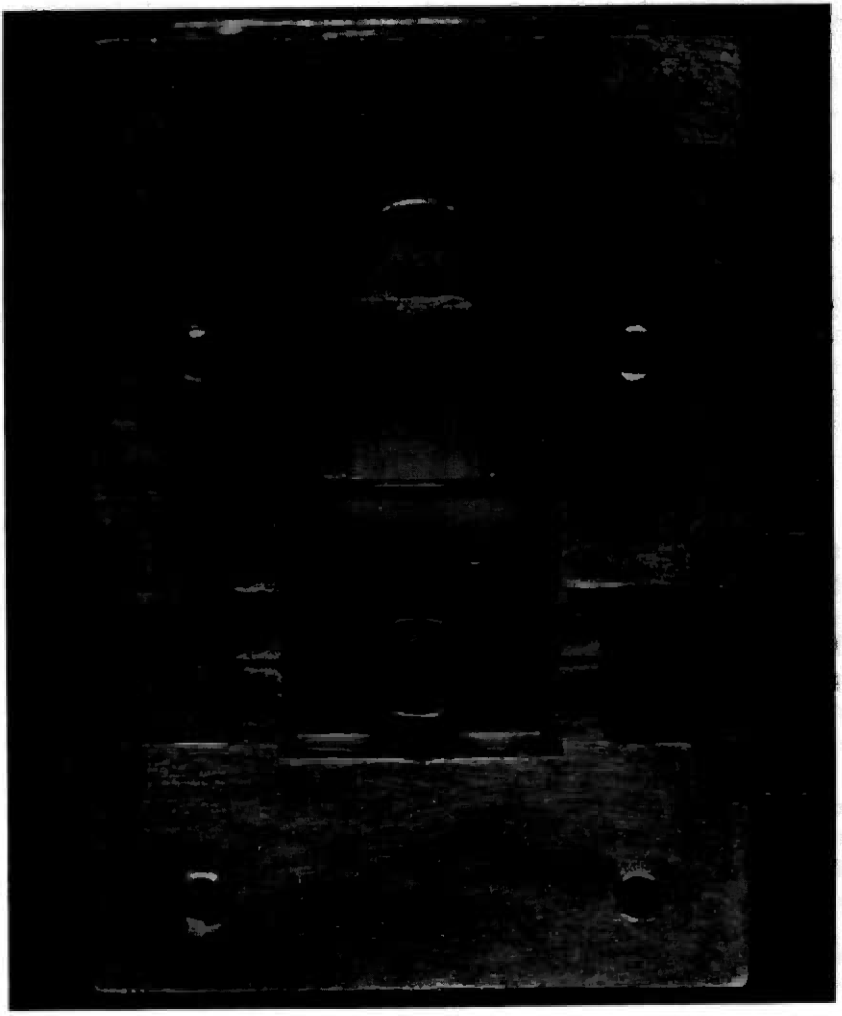


Fig. 4—Casting mold to encapsulate the LED array.

Fig. 5—LED array package and cable termination.

nut, inside the mold, becomes part of the package. The fibers sticking out of the connector surface are shortened using a diamond scriber. The two stainless steel pins are momentarily removed, and the connector surface is lapped and polished.

The cable termination of the ribbon connector was cast in a way similar to the LED array. The six fibers of the cable were threaded into the holes of a metal plate, and the connector was cast using the same mold as in the case of the LED package. Figure 5 shows the finished LED array package and the cable termination.

VII. CONCLUSIONS

A package with an array of six LEDs has been described. The design represents a first effort at reducing the cost and the size of electrooptical interfaces for applications in low-bit-rate transmission systems and computer data links. A package containing six GaAs LEDs, electrical connections, and a new fiber ribbon connector was fabricated, mounted, and encapsulated using presently available technology.

The LEDs in the array, with 80-mA dc applied, couple an average optical power of 5 μW into a graded-index fiber with 0.22 N.A. and 55-μm core diameter. Cross-modulation between adjacent LED signals is smaller than −50 dB. Fiber ribbon connectors fabricated with matching plates which align the fibers with the light output from the LEDs showed insertion losses as low as 0.4 dB.

REFERENCES

1. W. H. Hackett, Jr., C. A. Brackett, L. C. Dombrowski, L. E. Howarth, P. W. Shumate, R. G. Smith, A. W. Warner, R. S. Riggs, and J. R. Jones, "Optical Data

Links for Short-Haul High Level Performance at 16 and 32 Mb/s," Topical Meeting on Fiber Transmission II, February 22–24, 1977, Williamsburg, Virginia.
2. C. A. Burrus and R. W. Dawson, "Small-Area High-Current-Density GaAs Electroluminescent Diode and a Method of Operation for Improved Degradation Characteristics," Appl. Phys. Lett., *17,* No. 3 (August 1970).
3. W. S. Holden, T. P. Lee, C. J. Mogab, and F. B. Alexander, "High Radiance GaAs LEDs Made by Planar Diffusion of Zinc in GaAs using Low Temperature Plasma Deposited Silicon Nitride as a Diffusion Mask," unpublished work.
4. J. D. Crow, L. D. Comerford, J. S. Harper, M. J. Brady, and R. A. Laff, "AlGaAs Laser Array on Silicon Source Package," Appl. Opt., *17,* No. 3 (February 1978).
5. E. G. H. Lean, "Multimode Fiber Devices for Optical Fiber Links, Printing, and Display," IEEE Trans. Commun., *COM-26,* No. 7 (July 1978).
6. H. Yonezu, K. Kobayashi, K. Minemura, and I. Sakuma, "GaAs-Al$_x$Ga$_{1-x}$ As Double Heterostructure Laser for Optical Fiber Communication Systems," 1973 International Electron Device Meeting, Digest of Technical Papers, p. 324.
7. R. A. Abram, R. W. Allen, and R. C. Goodfellow, "The Coupling of Light-Emitting Diodes to Optical Fibers Using Sphere Lenses," J. Appl. Phys., *46,* No. 8 (August 1975), pp. 3468–3474.
8. T. C. Chu and A. R. McCormick, "Measurement of Loss Due to Offset, End Separation, and Angular Misalignment in Graded Index Fibers Excited by an Incoherent Source," B.S.T.J., *57,* No. 3 (March 1978), pp. 595–602.

Copyright © 1979 American Telephone and Telegraph Company
THE BELL SYSTEM TECHNICAL JOURNAL
Vol. 58, No. 3, March 1979
Printed in U.S.A.

DPCM with Forced Updating and Partial Correction of Transmission Errors

By R. STEELE, D. J. GOODMAN, and C. A. McGONEGAL

(Manuscript received September 28, 1978)

A speech signal is dpcm (differential pulse code modulation) encoded at the sampling rate f_s and also pcm encoded at a rate f_s/W. Blocks of W dpcm words and one pcm word are transmitted. The receiver compares the decoded dpcm signal at the end of every block with the decoded pcm sample. If the difference is above a threshold, it is assumed that one error exists within the block, and a search is made for the erroneous dpcm code word. Correction is accomplished by inverting the bits in this code word until the difference is below the threshold. Whether or not the error is corrected, the dpcm signal at the end of the block is forced to the value of the pcm sample, thereby preventing error propagation outside the block. The system improves segmental s/n ratio by 7 dB for W = 64 and bit error rates between 0.1 and 0.5 percent. Larger improvements are available with smaller block sizes. In the absence of transmission errors, there is no perceptible distortion due to the correction system.

I. INTRODUCTION

We describe a new method of protecting dpcm (differential pulse code modulation) speech signals against the effects of transmission errors. The dpcm bit stream is divided into blocks, and one pcm sample is transmitted with each block. At the receiver, the appropriate sample of the integrated dpcm signal is compared with the pcm sample. A disparity between the two samples is evidence of a transmission error within the block.

When an error is thus detected, the dpcm integrator is reset to the value of the pcm sample and an algorithm is invoked to locate the error within the block. When the algorithm is successful, the transmission error is completely corrected. Even when the algorithm is unsuccessful, the resetting of the integrator at the receiver prevents the error from propagating outside the block in which it occurs.

This approach to error protection is different in spirit from conventional channel coding aimed at protecting a digital information stream regardless of its nature. Our method is directly keyed to the dpcm character of the message. Because it introduces its own redundancy to a dpcm signal, it is more powerful than the DDC (difference detection and correction) system described by the authors.[1,2] DDC is implemented at the receiver only and infers transmission errors from anomalies within the integrated dpcm sample sequence.

Other authors have reported on the periodic transmission of pcm code words in a dpcm picture coding system.[3] The pcm samples were used to update the receiver integrator and thereby curtail visible streaks caused by dpcm transmission errors. The use of pcm samples for error detection and correction is new to this paper.

II. SYSTEM DEFINITION

2.1 The transmitter

The transmitter (Fig. 1) sends one block of data every W sampling intervals. The data consist of the W code words of a conventional dpcm encoder plus one pcm code word formed by quantizing the input sample at the end of the block. In our implementation, the pcm and dpcm samples are formed by the same quantizer. As a consequence, for intermediate and high level inputs, the pcm signal is unable to code the entire range of signal amplitudes. Therefore, at the receiver, the error control mechanism is disabled when the pcm quantizer is overloaded so that errors affecting high amplitude speech samples go uncorrected and are allowed to propagate beyond the blocks in which they occur. We accept this penalty in order to derive the convenience

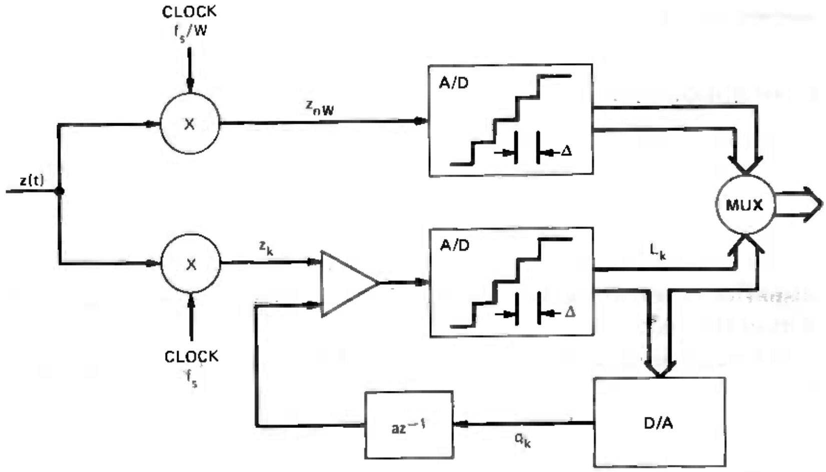


Fig. 1—Transmitter. The dpcm and pcm bit streams are multiplexed to form the transmitted sequence. The symbol rate is $f_s(W+1)/W$ Hz.

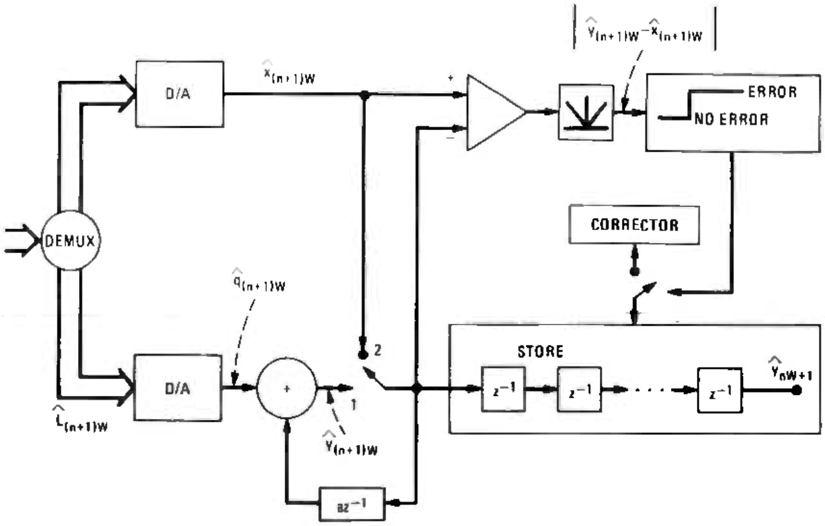


Fig. 2—Receiver. If, at the end of the block, the pcm sample and the dpcm sample differ by more than Δ, the correction logic is introduced. The switch in the integrator is in position 1 for W samples and goes to position 2 at the end of each block provided $\hat{x}$ is not an extreme value.

of a system with one quantizer, because high amplitude samples occur with relatively low probability in speech, and because we anticipate that errors in the idle channel and low amplitude regions of the signal are the most damaging ones subjectively.

2.2 The receiver

During reception of the nth block of data (Fig. 2), the integrator generates W dpcm output words $\hat{y}_{nW+j}$, from the received dpcm sequence $\{\hat{q}_k\}$ according to:

$$\hat{y}_{nW+j} = a\hat{y}_{nW+j-1} + \hat{q}_{nW+j}, \qquad j = 1, 2, \cdots, W. \tag{1}$$

These W samples are stored for possible revision by the error correction algorithm. At the end of the block, $\hat{y}_{(n+1)W}$ is reset to the pcm sample $\hat{x}_{(n+1)W}$ if $\hat{x}_{(n+1)W}$ is not the most positive or the most negative code word. When $\hat{x}_{(n+1)W}$ is at an extreme value, $\hat{y}_{(n+1)W}$ remains set at the value computed from (1).

2.2.1 Error detection

Provided $\hat{x}_{(n+1)W}$ does not indicate quantizer overload, it is compared with $\hat{y}_{(n+1)W}$ derived from (1). If these two samples differ by more than Δ, the quantizer step size, an error in the nth block is inferred and a search for the error is initiated. Otherwise the samples

$$\hat{y}_{nW+1}, \hat{y}_{nW+2}, \cdots, \hat{y}_{(n+1)W} \tag{2}$$

are sent to the system low-pass filter, $\hat{y}_{(n+1)W}$ is reset to $\hat{x}_{(n+1)W}$, and block $n + 1$ is processed.

2.2.2 Error correction

When $|\hat{x}_{(n+1)W} - \hat{y}_{(n+1)W}| > \Delta$, the system scans the samples in its buffer and finds the largest sample-to-sample difference in the block. That is, it computes

$$\delta_j = |\hat{y}_{nW+j} - \hat{y}_{nW+j-1}|, \qquad j = 1, 2, \cdots, W$$

and determines the maximum δ_j. If this maximum occurs at the rth position in the block, the corrector modifies $\hat{q}_r$ by successively inverting bits in the code word $\hat{L}_r$. With each bit inversion, $\tilde{q}_r$, a version of $\hat{q}_r$, is formed and (1) is iterated to produce a new trial value of $\hat{y}_{(n+1)W}$, which we call $\bar{y}_{(n+1)W}$. Thus we have

$$\bar{y}_{nW+r} = a\hat{y}_{nW+r-1} + \tilde{q}_r$$

and

$$\bar{y}_{nW+j} = a\hat{y}_{nW+j-1} + \hat{q}_{nW+j}, \qquad j = r + 1, r + 2, \cdots, W.$$

Now, if $\bar{y}_{(n+1)W}$ satisfies the test,

$$|\bar{y}_{(n+1)W} - \hat{x}_{(n+1)W}| \leq \Delta,$$

the modified sequence

$$\hat{y}_{nW+1}, \cdots, \hat{y}_{nW+r-1}, \bar{y}_{nW+r}, \cdots, \bar{y}_{(n+1)W}$$

becomes the system output. Otherwise, a new value of $\tilde{q}_r$ is obtained by inverting another bit in $\hat{L}_r$.

If no single-bit inversion in $\hat{L}_r$ succeeds in bringing $\bar{y}_{(n+1)W}$ sufficiently close to $\hat{x}_{(n+1)W}$, the correction attempt ceases and the samples in (2) are sent to the output filter.

Clearly, this correction scheme is effective only when there is one bit in error in a block of W samples and this single error leads to a large sample-to-sample difference. However, even when there is more than one error in the block, the updating of the integrator signal prevents long-term propagation of error effects.

Errors in pcm samples induce distortions that would not be present in ordinary dpcm. Our performance evaluations indicate that the effects of these distortions are substantially smaller than the benefits of error correction and integrator updating.

III. EVALUATION

The technique was evaluated by means of computer simulation of a 7-bit, 8-kHz single-integration dpcm system with prediction coefficient 0.9. To obtain the objective measurements displayed in Figs. 3 to 5, we repeatedly played a single sentence through the system: "I have two

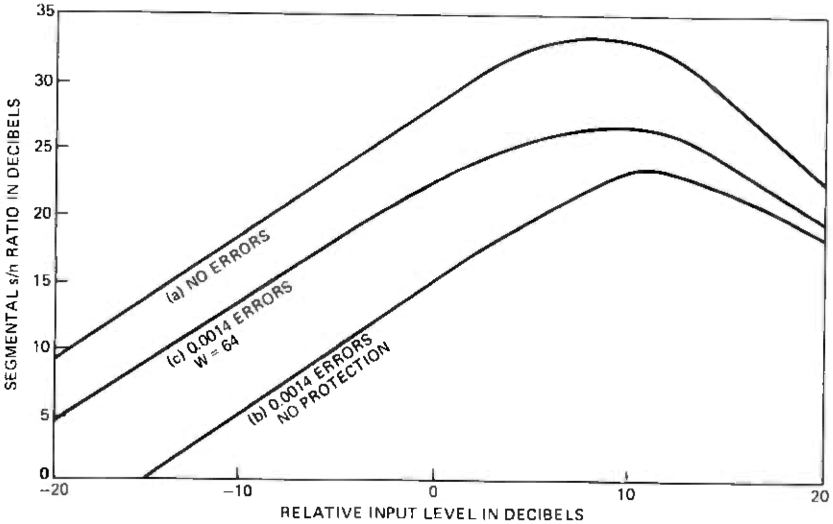


Fig. 3—Performance as a function of input level. (a) No transmission errors. Segmental s/n ratio reflects only quantizing noise. The curves with and without error protection virtually coincide. (b) Transmission error rate is 0.0014 and the system is unprotected. (c) Error rate is 0.0014 and protection block size is 64.

daughters, Lorna and Susan," spoken by a male. By listening to many samples of processed speech from a variety of talkers, we confirmed that the error protection indicated in these graphs is subjectively meaningful and not peculiar to a single utterance.

For each system configuration, we measured segmental signal-to-noise ratio,[4] defined as the decibel average of the signal-to-noise ratios in 214 speech segments, each of duration 16 ms. These are the segments (out of 224 in the 3.5-second utterance) in which the rms signal level exceeds −60 dB relative to the peak signal. Segmental s/n ratio is considered a better indicator of speech quality than ordinary s/n ratio. In a study of adaptive dpcm, there was a correlation of 0.93 between segmental s/n ratio and subjective ratings of speech quality.[5] The comparable correlation with ordinary s/n ratio was only 0.69.

Variables in our experiment were input signal level, block size W, and transmission error rate. The simulated channel introduces random errors to the serial bit stream consisting of W 7-bit dpcm code words and one 7-bit pcm code word per block.

Figure 3 shows segmental s/n ratio as a function of input level for a block size $W = 64$ and for unprotected dpcm. The top curve is for zero error rate, in which case s/n ratio is the same (to within 0.3 dB) with and without the error protection mechanism. Curve b shows the effect on dpcm of a channel with error rate 0.0014. In the granular noise region (signal level below the peak of curve a), these errors cause a degradation of about 13 dB in s/n ratio. Over most of this region, the

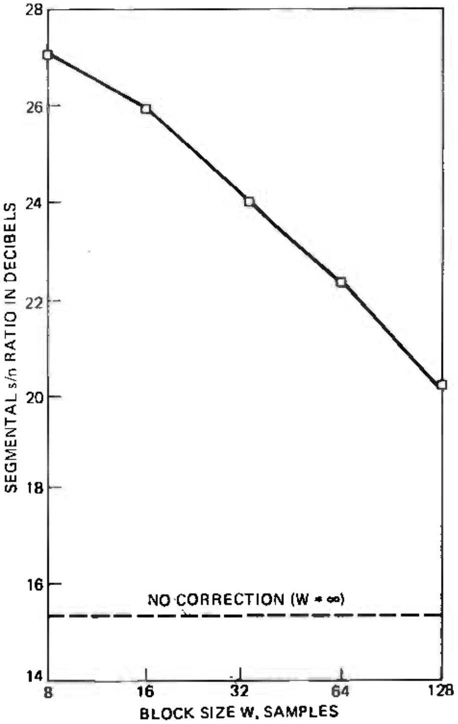


Fig. 4—The effect of block size on performance. The relative input level is 0 dB and the error rate is 0.0014. Over most of the range, s/n ratio decreases by about 2 dB per octave increase in block size.

protection system redeems about 9 dB of this loss. For reasons discussed in Section 2.1, the effectiveness of the error protection diminishes as signal level increases.

Figure 4 pertains to the 0-dB relative input level (see Fig. 3) and 0.0014 error rate. It shows the effect on segmental s/n ratio of varying the block size, W, over the range 8 to 128 samples. A small block size offers more protection but exacts a greater penalty in transmitted bit rate than a large block size. With $W = 8$ there is almost one extra bit per code word and an improvement of 11.5 dB in s/n ratio. The improvement decreases by about 2 dB per octave change in W over the range we investigated.

The effectiveness of the error protection as a function of channel quality is shown in Fig. 5 for the input level 0 dB and $W = 64$. Our error protection mechanism results in an s/n ratio improvement of about 6 to 8 dB at error rates between 0.001 and 0.01. For comparison, we also show the performance of a dpcm system protected by the DDC (difference detection and correction) scheme.[1,2] DDC requires no modification of the dpcm transmitter and no additional transmitted bits.

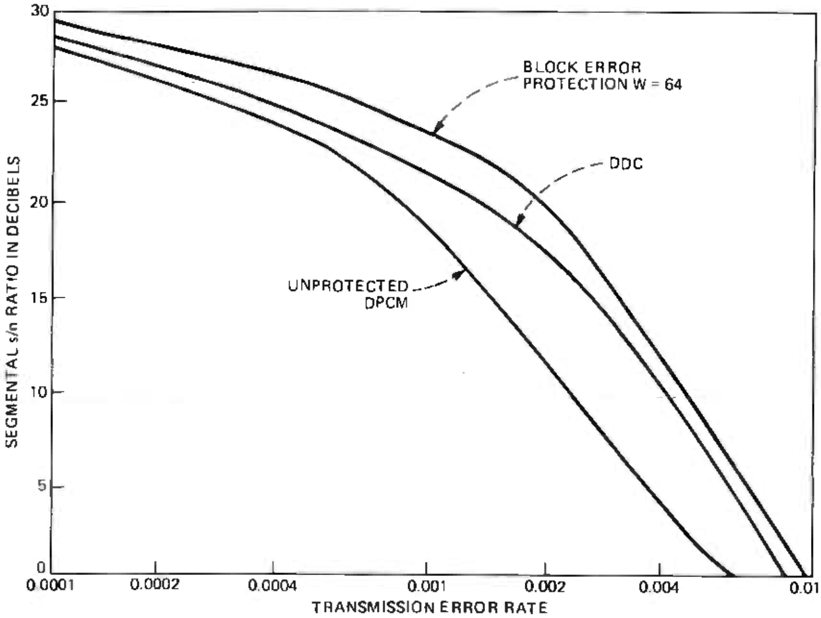


Fig. 5—Performance as a function of channel error rate, 0 dB relative input level. Here, the block protection scheme is compared with ordinary dpcm and with dpcm augmented at the receiver by DDC, a difference detection and correction system.

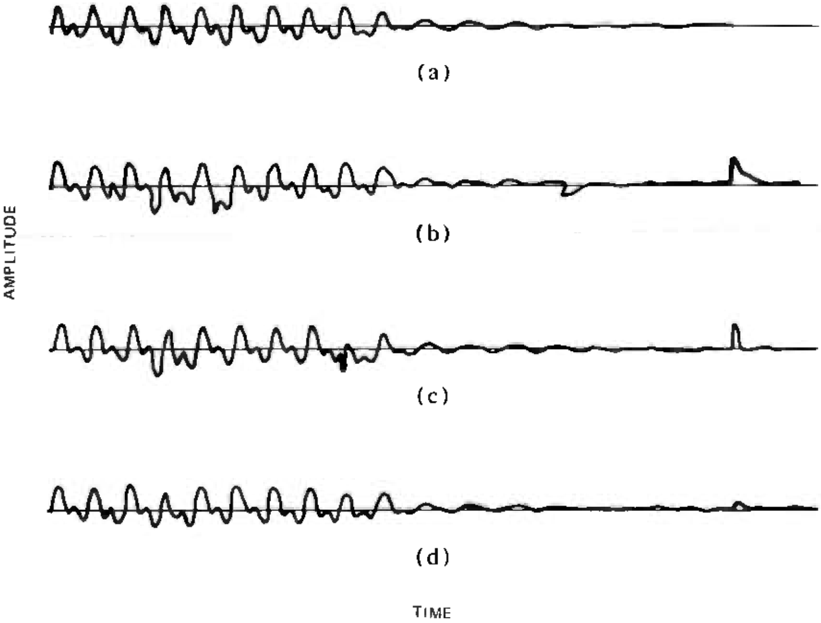


Fig. 6—Oscillograms of a segment of the test utterance. (a) Original speech, 0 dB relative input level. (b) dpcm output with 0.0042 error per bit. (c) After block protection with $W = 64$. (d) After selective smoothing of (c) by means of DDC.

Its performance, however, is about 2 dB poorer than that of the new scheme.

IV. REPROCESSING THE PARTIALLY CORRECTED SPEECH

The forced updating of the integrator signal itself causes sharp transients in the system output when a detected error cannot be corrected. Thus, the output signal often contains many spurious spikes. This phenomenon is illustrated by the speech waveforms in Fig. 6. A segment of the original speech signal is shown in Fig. 6a, and Fig. 6b shows the same segment corrupted by transmission errors (which occurred with probability 0.0042). Figure 6c shows the output of the protected dpcm system, which has suppressed most of the channel error noise but left a residual spike (click) in the signal. DDC[6] (as applied to pcm) is designed to smooth out such spikes and the effect of reprocessing the system output with DDC is seen in Fig. 6d. In general, appending DDC to the forced updating method is effective in suppressing residual impulse noice.

V. CONCLUSIONS

By periodically introducing pcm samples to a dpcm signal sequence, it is possible to reduce substantially the propagation of transmission errors in dpcm. Furthermore, at the cost of some delay, storage, and elementary signal processing in the receiver, many errors can be completely corrected. Small block sizes (frequent pcm transmissions) are more effective but introduce larger transmission rate penalties than large block sizes.

We have shown that this block protection scheme improves segmental s/n ratio of speech signals, and our experience of listening to several speech samples processed this way confirms that subjective quality is correspondingly enhanced. The method is also very appropriate to dpcm transmission of video signals in which error propagation causes very objectionable streaks in pictures.[3]

REFERENCES

1. R. Steele, D. J. Esdale, and D. J. Goodman, "Partial Correction of Transmission Errors in DPCM Without Recourse to Error Correction Coding," Elec. Lett., *13*, No. 12 (June 9, 1977), pp. 351–353.
2. R. Steele, D. J. Goodman, and C. A. McGonegal, "A Difference Detection and Correction Scheme for Combating DPCM Transmission Errors," IEEE Trans. Commun., *COM-27*, No. 1 (January 1979), pp. 252–255.
3. R. J. Arguello, H. R. Sellner, and J. A. Stuller, "The Effect of Channel Errors in the Differential Pulse-Code-Modulation Transmission of Sampled Imagery," IEEE Transactions on Communications, *COM-19*, No. 6 (December 1971), pp. 926–933.
4. P. Noll, "Adaptive Quantizing in Speech Coding Systems," International Zurich Seminar, Zurich, Switzerland, April 1974.
5. B. McDermott, C. Scagliola, and D. Goodman, "Perceptual and Objective Evaluation of Speech Processed by Adaptive Differential PCM," B.S.T.J., *57*, No. 5 (May–June 1978), pp. 1597–1618.
6. R. Steele and D. J. Goodman, "Detection and Selective Smoothing of Transmission Errors in Linear PCM," B.S.T.J., *56*, No. 3 (March 1977), pp. 399–409.

Copyright © 1979 American Telephone and Telegraph Company
THE BELL SYSTEM TECHNICAL JOURNAL
Vol. 58, No. 3, March 1979
Printed in U.S.A.

Equivalent Circuits for the Analysis and Synthesis of Switched Capacitor Networks

By K. R. LAKER

(Manuscript received August 17, 1978)

An extensive library of z-domain building block equivalent circuits is derived to facilitate the analysis and synthesis of switched capacitor (SC) networks. These SC building blocks, typically comprised of a single capacitor and from one to four switches, serve as basic circuit elements for SC networks in much the same spirit that resistors and capacitors serve analog networks. This building block library facilitates the derivation of canonic z-domain equivalent circuits for complex SC networks and the application of well-established mathematical network analysis and synthesis tools. What has been sought are easily applied techniques for achieving the same insights for SC networks that we have long enjoyed with analog networks.

I. INTRODUCTION

Over the past several years, many researchers[1-15] have searched for the means to realize monolithic analog recursive filters, particularly for voice frequency applications. Initial attempts to realize a monolithic filters technology led to the development of active-R or resistor-only active filters.[1-6] By removing the large external capacitors ($C \sim 5000$ pF), such filters are, in principle, highly suited to integration with standard bipolar processing. The frequency dependence for these filters is derived[1-5] from the single-pole rolloff due to a compensation capacitor ($C \sim 30$ pF) to achieve a unity gain frequency of 1 MHz. This method of operation posed two significant barriers to the practical application of active-R filters; namely, large resistor ratios[1-5] are required to reach audio frequencies and the unity gain frequencies are not sufficiently stable[1-4] for precise filter realization. Although these barriers have to some degree been overcome,[5, 6] it has become clear that the future of integrated circuits is in MOS large-scale integration and very large-scale integration processing. LSI has substantially reduced the cost of digital logic and memory, and VLSI will bring even further cost reductions.

With the overwhelming success of digital MOS LSI and the promises of VLSI as motivation, much productive effort[7-15] has been spent on the development of a compatible MOS LSI sample data technology. This work culminated in the development of compact operational amplifiers[12] with acceptable noise and power specifications, charge transfer device (CTD) transversal filters,[14, 15] and switched capacitor (SC) recursive filters[7-11] which fully utilize the advantages provided by MOS LSI. The transfer function coefficients[7-11] of an SC recursive filter are determined by a highly stable clock frequency and capacitor ratios which can be held to very tight tolerances (measured[13] errors of less than 0.2 percent have been achieved for binary valued capacitor ratios). Furthermore, MOS capacitors are nearly ideal, with very low dissipation factors and temperature coefficients[13] of less than 100 ppm/°C. This process of inherent precision and quality is sufficient to meet many filter and system specifications.

Considering the growing interest in MOS switched capacitor networks, the need is obvious for analytical and computer-aided tools[16-21] for the analysis and synthesis of SC networks. The pioneering work[16, 17] of Kurth and Moschytz provided a rigorous, network-theoretic foundation to the characterization of SC networks. They considered the analysis of SC networks comprised of capacitors and periodic, bi-phased switches. These networks, which are sampled data in nature, were shown to be characterized by nodal charge difference equations with periodically time-varying coefficients. This system of equations can be transformed into the z-domain[22, 23] to obtain the frequency response of the SC network. To reduce the analytical complexity, a building block approach was introduced[17] with the six basic building blocks: (*i*) shunt capacitor, (*ii*) series capacitor, (*iii*) shunt capacitor in parallel with a switch, (*iv*) series capacitor in parallel with a switch, (*v*) shunt switch, and (*vi*) series switch. These building blocks were expressed as four-port equivalent circuits with each two-port pair accounting for each of two signal paths which result from the two switch phases. The two signal paths, denoted even and odd, were shown to be linked by a common link two-port (LTP)[17] network. Any SC network, comprised of bi-phase switches, can be transformed into a z-domain equivalent circuit by interconnecting the appropriate combination of these building-block equivalent circuits. This equivalent circuit then provides the network designer with a pictorial representation of the circuit from which transfer relations can be derived between any pair of node voltages and provides the instant insight to circuit innovations we have long enjoyed with analog, linear, time-invariant networks.

The primary objective of this paper is to extend the Kurth-Moschytz[17] library of building blocks to include those higher-order SC elements which occur frequently in complex SC networks. These elements are typically comprised of one capacitor and from one to four switches. With this expanded library of equivalent circuits, one can efficiently derive canonic z-domain equivalent circuits for any SC network. It is shown that, by manipulating the equivalent circuit for a toggle-switched four-port element one can derive all the equivalent circuits in the library. Alternative interpretations of the Kurth-Moschytz LTP are also provided which facilitate the derivation of canonic equivalent circuits. Finally, several examples are given which demonstrate the ease in which equivalent circuits are constructed and the insight derived therefrom.

II. APPLICATION OF Z-TRANSFORM TECHNIQUES TO THE ANALYSIS OF SC NETWORKS WITH BI-PHASE SWITCHES

This section briefly reviews the basic assumptions[16, 17] regarding the sampled data nature of SC networks and the fundamentals germane to the derivations and procedures given in the succeeding sections. This review also provides an opportunity to define symbols and to acquaint the reader with the notation employed.

2.1 Operation of ideal SC networks

Consider now the operation of an ideal SC network, comprised of ideal capacitors, ideal switches, and ideal voltage-controlled voltage sources (i.e., ideal operational amplifiers) when excited by sampled data voltage inputs. Typically, the switches are controlled by a two-phase, nonoverlapping clock of frequency $f_c = 1/2T$, as shown in Fig. 1. Note that ϕ^e is used to denote the even clock phase that instantaneously closes the e-switch on the even $2kT$ times. Similarly, ϕ^o denotes the odd clock phase that instantaneously closes the o-switch on the odd $(2k + 1)T$ times. The switches are assumed to have a 50-percent duty cycle with equal (T-second) on- and-off time periods. It is further assumed that both the input and output of the SC network are sampled data signals which change in value only at switching instants kT. Thus, in their most general form, the voltage sources and internal circuit voltages are assumed to be sampled at times kT and held over a one-half clock period interval (T) as shown in Fig. 2a. With this assumption, we can apply[16-19] z-transform techniques to the general analysis and synthesis of SC networks. The z-transform, $z = e^{sT}$, where s is the complex analog frequency variable and T is the clock period, then provides us with a convenient means for performing frequency domain analysis. Of course, the z-domain transfer functions obtained from this procedure relate the input and output samples of switched capacitor networks. Thus, to obtain the response, the input must also be characterized in the z-domain. Furthermore, if the output is considered as a held[24] (staircase) signal, this computed response must be modified[24] by $(\sin x)/x$. When continuous inputs are applied directly[20, 21] to SC networks, the analysis can become considerably more complex.

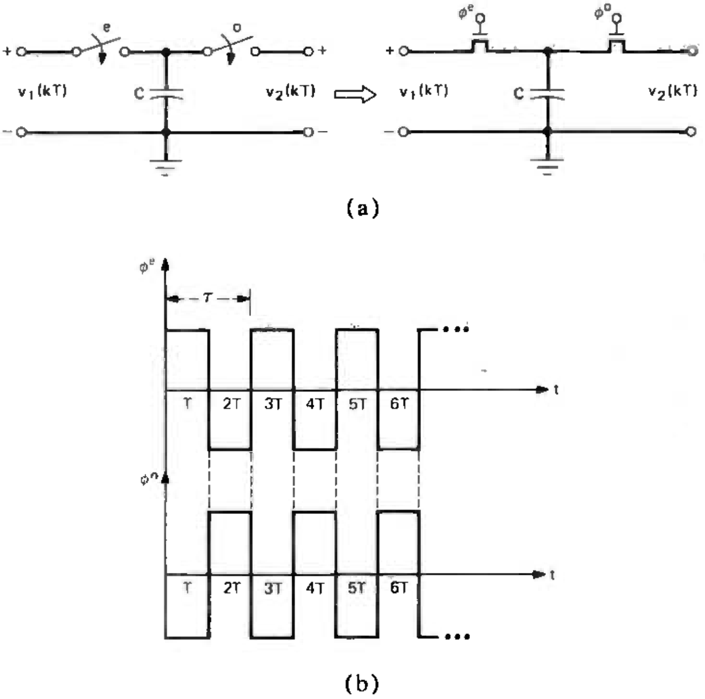


Fig. 1—(a) Simple switched capacitor network with (b) two-phase nonoverlapping clock.

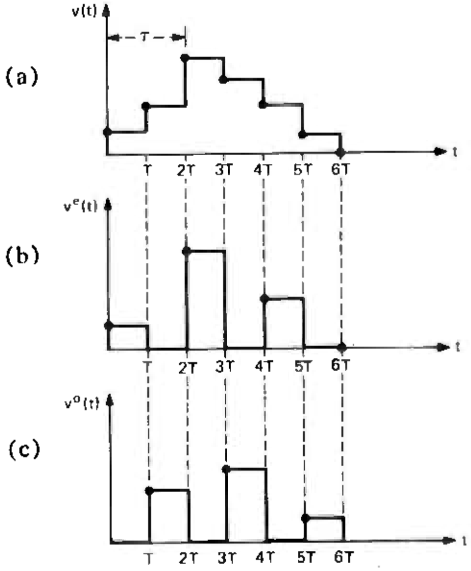


Fig. 2—(a) Sampled data voltage waveform portioned into its (b) even and (c) odd components.

As pointed out in Ref. 16, the switching action described in Fig. 1 provides a time-varying nature to the SC network. That is, as the switches open and close, the network graph changes, alternating between two topologies. One topology corresponds to the even clock phase and a second topology to the odd clock phase. Thus, one can view the time-varying SC network, with bi-phase switches, as two interrelated time invariant networks.[16–21] In lieu of this fundamental approach, it is mathematically convenient to partition[16] the sampled data voltage waveform in Fig. 2a into its even and odd components as shown, respectively, in Figs. 2b and 2c. Comparing v^e and v^o to the clock waveforms ϕ^e and ϕ^o, we observe that v^e is only nonzero when the e-switch is closed and v^o is only nonzero when the o-switch is closed. This fundamental observation[16, 17] has opened the door to a rigorous understanding of switched capacitor networks and has resulted in several methods for their analysis.

One way to interpret the relationship between the even and odd topologies is to consider them topologically decoupled, with the states of one determining the initial conditions for the other.[18, 19] This interpretation results in two distinct circuits coupled together via dependent sources which establish the initial conditions mentioned previously. This formulation has been found[18, 19] to be particularly convenient for computer-aided analysis. Another interpretation[16, 17] is to combine the even and odd networks topologically into a single z-domain equivalent circuit. In general, an n-port SC network[16, 17] will require a $2n$-port equivalent circuit, i.e., n-ports for the even clock phase and n-ports for the odd clock phase. It is this interpretation that provides the kinds of valuable insight that Laplace transform techniques have provided for analog linear-time invariant networks.

Since SC networks can be most rigorously characterized[16, 17] in terms of charge transfer operations, discrete time voltages $v_i(kT)$ and discrete time charge variations $\Delta q_i(kT)$ are used as port variables. At the switching times kT, charges are instantaneously redistributed with the principle of charge conservation maintained at every node in the network. It is this principle that allows us to write nodal charge equations in the same spirit with which we apply Kirchoff's current law to continuous networks. In general, due to the bi-phase switching operation, two distinct, but coupled, nodal charge equations are required to characterize the charge conservation condition at a particular node for all time instants kT. Namely, one equation for the even $2kT$ times and a second equation for the odd $(2k + 1)T$ times are required. These equations are written, for some node p, as

$$\Delta q_p^e(kT) = \sum_{i=1}^{M_{ep}} q_{pi}^e(kT) - \sum_{i=1}^{M_{op}} q_{pi}^o((k-1)T)$$

for k an even integer, (1a)

and

$$\Delta q_p^o(kT) = \sum_{i=1}^{M_{op}} q_{pi}^o(kT) - \sum_{i=1}^{M_{op}} q_{pi}^e((k-1)T)$$

for k an odd integer, (1b)

or equivalently in the z-domain

$$\Delta Q_p^e(z) = \sum_{i=1}^{M_{ep}} Q_{pi}^e(z) - z^{-1/2} \sum_{i=1}^{M_{ep}} Q_{pi}^o(z) \tag{2a}$$

and

$$\Delta Q_p^o(z) = \sum_{i=1}^{M_{op}} Q_{pi}^o(z) - z^{-1/2} \sum_{i=1}^{M_{op}} Q_{pi}^e(z), \tag{2b}$$

where q_{pi}^e, q_{pi}^o and Q_{pi}^e, Q_{pi}^o denote, respectively, the instantaneous charges stored on the ith capacitor connected to node p for the even, odd kT time instants and their z-transforms. Also M_{ep} and M_{op} denote respectively the total number of capacitors connected to node p during the even and odd clock phases.

For single capacitor SC blocks, z-transformed nodal charge equations[16] lead directly to the desired equivalent circuits as described in Section III. To characterize a complex SC network one simply substitutes, one-for-one, the appropriate z-domain block equivalent circuit for each SC element in the network schematic. As demonstrated in Section IV, transformed nodal charge equations for each node in the network are then written by inspection from the equivalent circuit. The desired voltage transfer function(s) is then obtained by algebraically manipulating these z-domain equations in the usual manner.

2.2 Sample data waveforms

It should be noted that there are several sample data waveforms that can be modeled as special cases of the waveform depicted in Fig. 2. These waveforms and their respective even and odd components are shown in Fig. 3. One can immediately invoke the z-transform to mathematically describe these waveforms. The return-to-zero waveforms in Figs. 3a and 3b can be expressed mathematically as

$$V_a(z) = V_a^e(z) + V_a^o(z), \tag{3a}$$

where

$$V_a^o(z) = 0 \tag{3b}$$

and

$$V_b(z) = V_b^e(z) + V_b^o(z), \tag{4a}$$

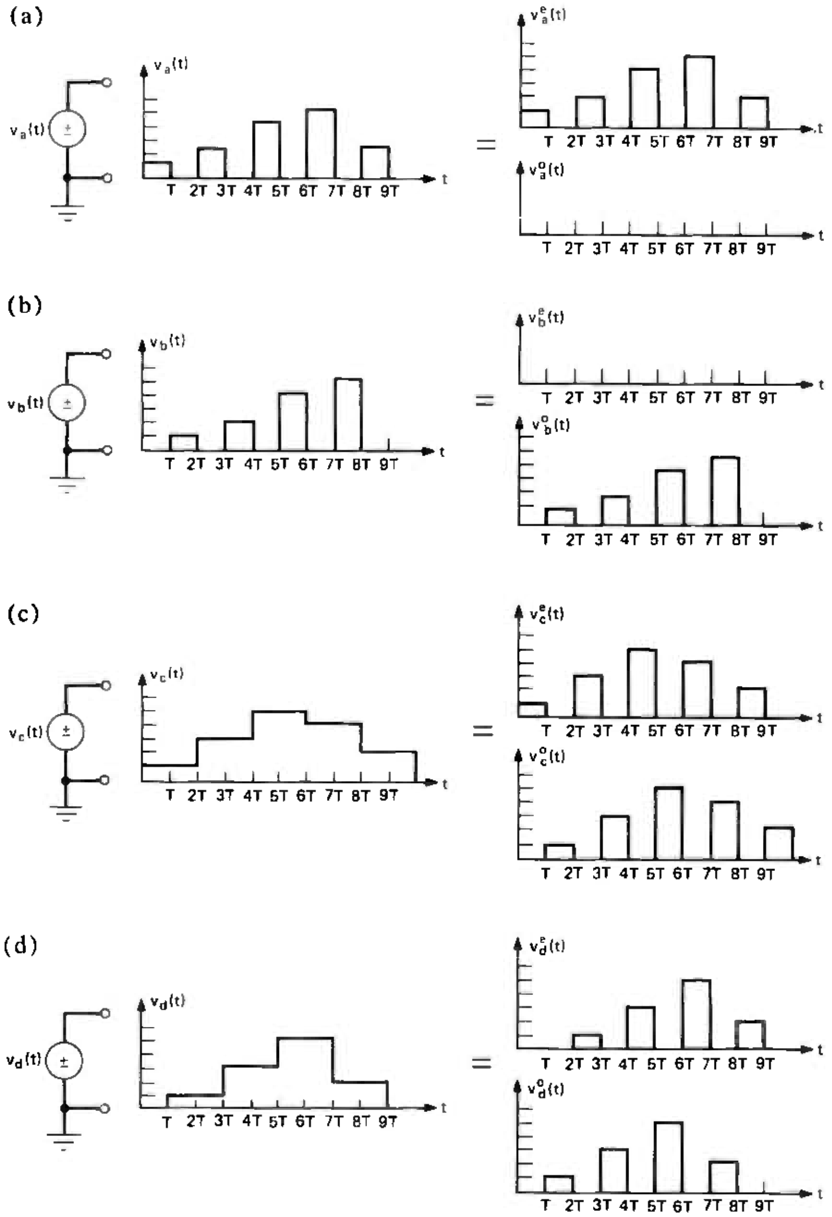


Fig. 3—Common special-case sampled data voltage waveforms and their respective even and odd components.

where

$$V_b^e(z) = 0. \tag{4b}$$

In a similar manner, we can characterize the full clock period sampled and held (*S/H*) waveforms in Figs. 3c and 3d as

$$V_c(z) = V_c^e(z) + V_c^o(z), \tag{5a}$$

where

$$V_c^o(z) = z^{-1/2} V_c^e(z) \tag{5b}$$

and

$$V_d(z) = V_d^e(z) + V_d^o(z), \tag{6a}$$

where

$$V_d^e(z) = z^{-1/2} V_d^o(z). \tag{6b}$$

If a capacitor of value C is placed across the terminals of the voltage source $v_c(t)$, we observe that the charge on the capacitor changes in value only once per clock cycle, i.e., at the even $2kT$ time instants when $v_c(t)$ changes. At the odd $(2k + 1)T$ time instants $v_c(t)$, the capacitor voltage is unchanged, thus, the charge remains constant. This phenomenon is described analytically, for the even and odd clock phases, in the following manner:

$$\Delta Q_c^e(z) = CV_c^e(z) - Cz^{-1/2} V_c^o(z) = C(1 - z^{-1}) V_c^e(z) \tag{7a}$$

and

$$\Delta Q_c^o(z) = CV_c^o(z) - Cz^{-1/2} V_c^e(z) = 0. \tag{7b}$$

It is noted that the condition $\Delta Q_c^o(z) = 0$ can also be obtained by disconnecting V_c from the capacitor with a switch which is open during the odd clock phase. Thus, in the sense that no charge is transferred, this open circuit condition also implies a full cycle sample and hold operation. Corresponding relations can also be written for the full cycle *S/H* waveform in Fig 3d.

It is useful to note that the return-to-zero waveforms $v_a(t)$ and $v_b(t)$ can be obtained by processing $v(t)$ in Fig. 2 with simple switch networks as shown in Fig. 4. When the switches in Fig. 4 are ideal, v_a and v_b are ideal, zero-impedance voltage sources with waveforms as depicted in Figs. 3a and 3b, respectively.

The various sample data waveforms considered in this section can be generated externally (i.e., an independent voltage source) and at any internal node by an appropriate combination of switches and capacitors. It is often crucial, particularly at the network output where one may either resample or couple to another SC network, to identify the waveform type of internal node voltages. From the properties described in this section, this identification process is usually straightforward.

2.3 SC network transfer function relations

The voltage transfer function is well recognized as a convenient mathematical tool for the analysis and synthesis of continuous, linear, time-invariant networks. The value of the voltage transfer function is not expected to diminish with SC networks. At this point, it should be obvious that SC network transfer functions are most conveniently written in the z-domain.

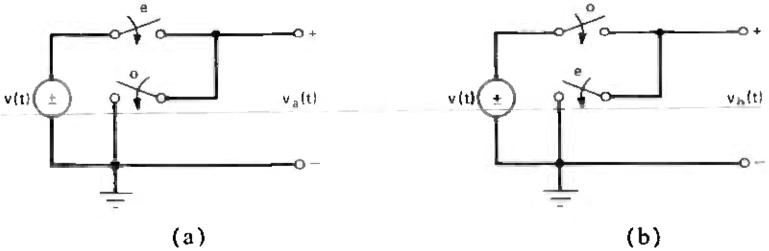
Let us, for simplicity, confine the discussion in this subsection to two-port SC networks with one input and one output. As noted previously, the two-port can be represented by an equivalent four-port, as shown in Fig. 5. In general, a 2 × 2 transfer matrix is required to fully characterize the input-output relations for this four-port network, i.e.,

$$\begin{bmatrix} V_{out}^{e}(z) \\ V_{out}^{o}(z) \end{bmatrix} = \begin{bmatrix} H_1(z) & H_2(z) \\ H_3(z) & H_4(z) \end{bmatrix} \begin{bmatrix} V_{in}^{e}(z) \\ V_{in}^{o}(z) \end{bmatrix} \tag{8}$$

where, by superposition,

$$V_{in}(z) = V_{in}^{e}(z) + V_{in}^{o}(z) \tag{9a}$$

$$V_{out}(z) = V_{out}^{e}(z) + V_{out}^{o}(z). \tag{9b}$$



(a) (b)

Fig. 4—Switch networks for return to zero voltage generation.

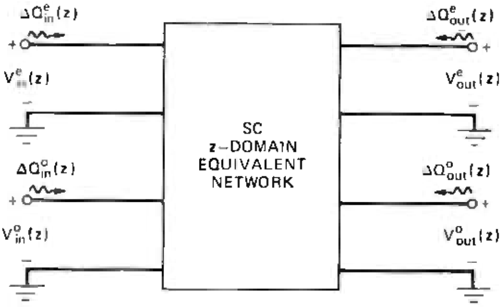


Fig. 5—Four-port z-domain equivalent circuit.

In many cases, the signal conditioning performed at the input and output imposes constraints on the form of the transfer relations. For example, consider the application of the return-to-zero source in Fig. 3a to the SC network in Fig. 5. Substituting $V_{in}^{o}(z) = 0$, obtained from eq. (3b), into eq. (8) yields the following transfer relations

$$V_{out}^{e}(z) = H_1(z)\,V_{in}^{e}(z) \tag{10}$$

and

$$V_{out}^{o}(z) = H_3(z)\,V_{in}^{e}(z). \tag{11}$$

Thus, depending on whether $v_{out}(kT)$ is sampled at the even $2kT$ times or the odd $(2k+1)T$ times, the voltage transfer function is either $H_1(z)$ or $H_3(z)$, respectively. However, if v_{out} is sampled at all kT times, then

$$V_{out}(z) = (H_1(z) + H_3(z))\,V_{in}^{e}(z). \tag{12}$$

In general, $H_1(z) \neq H_3(z)$; however, they are obviously interrelated. Therefore, one is not able to independently synthesize $H_1(z)$ and $H_3(z)$.

In practice, by appropriately conditioning the input and output signals, one can realize an SC network which is completely characterized by a single transfer function. Equations (10) and (11) describe examples of this class of SC network. One can in principle synthesize SC networks of this type directly in the z-domain using digital filter[22, 23] synthesis techniques. Several examples of multi-transfer function and single transfer function SC networks are provided in Section IV.

III. EQUIVALENT CIRCUIT MODELS FOR SC BUILDING BLOCKS

In this section, multi-port z-domain equivalent circuits will be derived for several SC building blocks. It has been shown[17] that any SC network can be constructed from the six blocks mentioned in Section I and voltage controlled voltage sources. The objective here is to facilitate the application of this approach by deriving a library of higher order building blocks which, when interconnected, lead to canonic z-domain equivalent circuits. SC elements comprised of one capacitor and from one to four switches are treated as basic circuit elements much like passive R's, L's, and C's in analog circuits. As noted in the previous section, the z-domain transfer relations can be derived from the equivalent circuit using familiar network analysis techniques.[25]

Figures 6 and 7 contain listings of the commonly occurring SC elements and their respective z-domain equivalent circuits or building blocks. In addition to the SC building blocks, z-domain models are also given for each of the sample data sources discussed in the previous section. This library is sufficiently general to accommodate all the

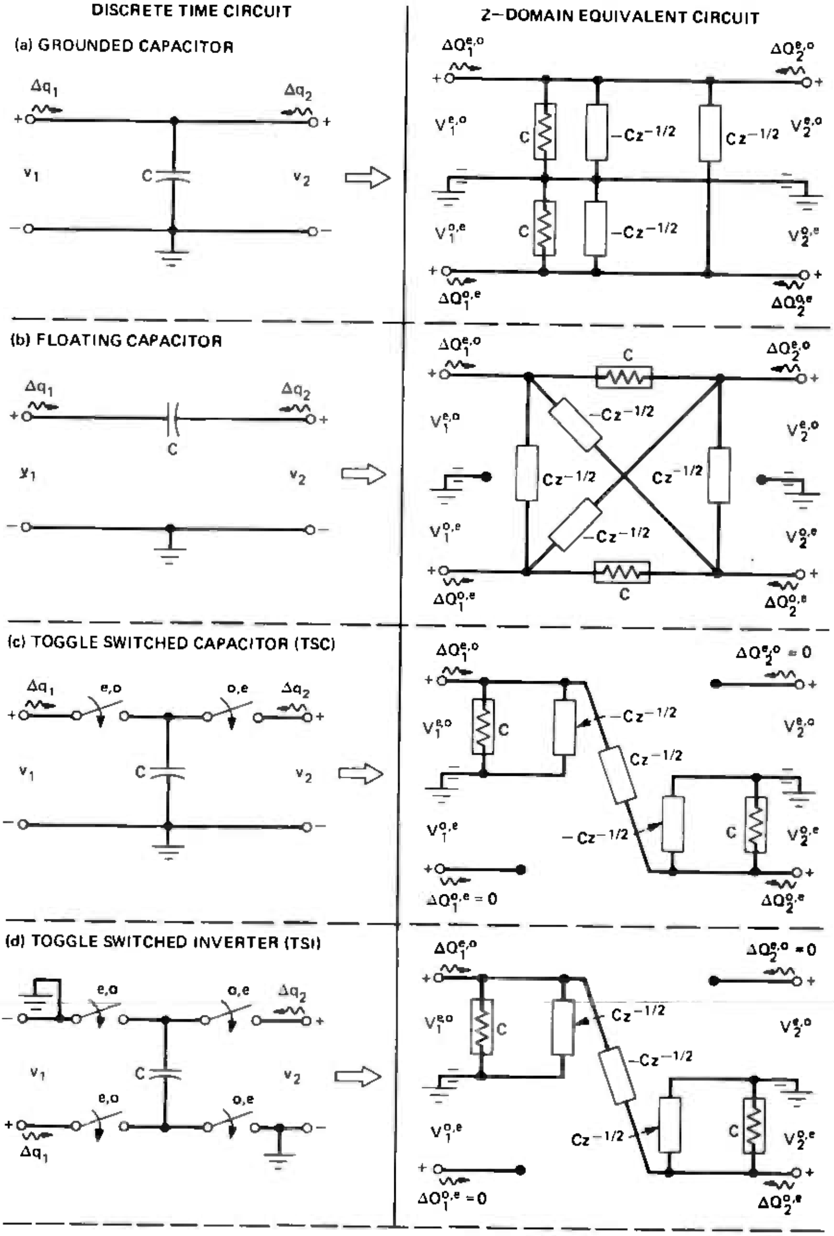
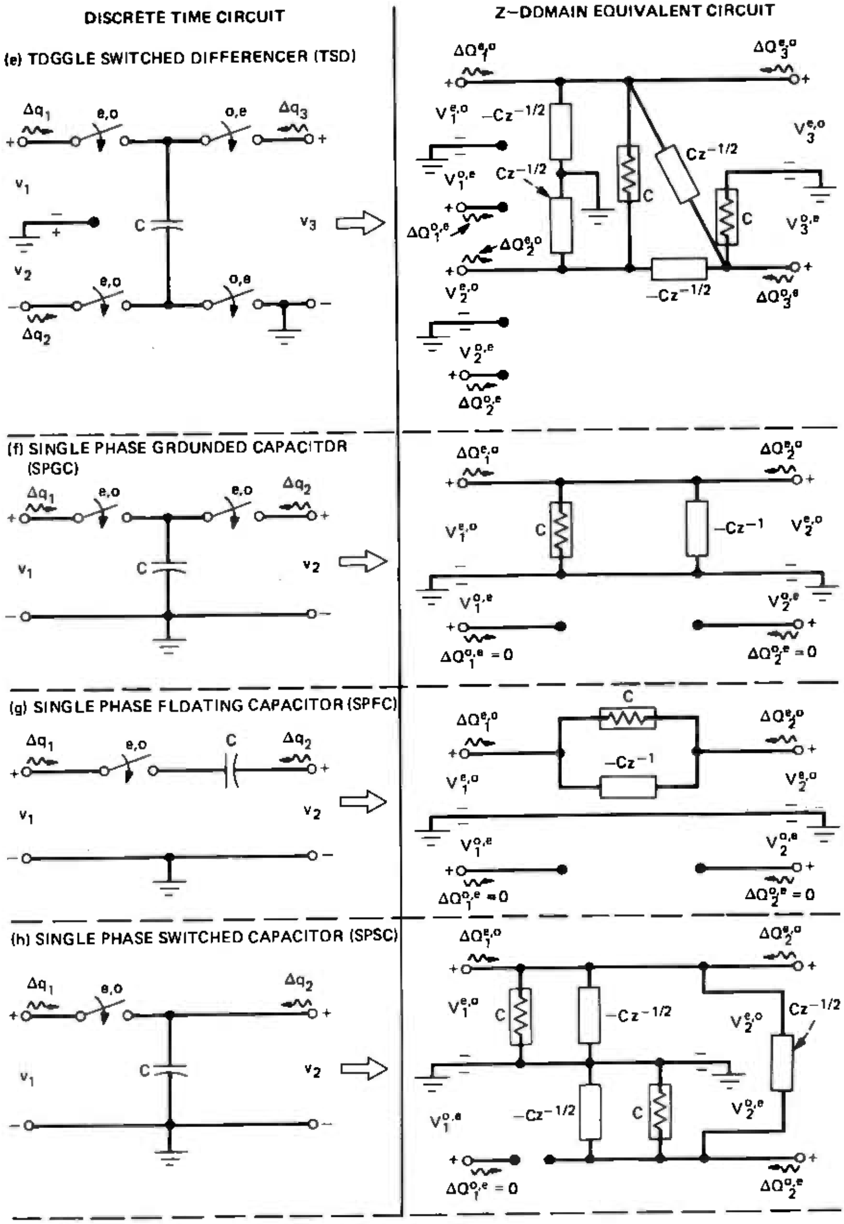


Fig. 6—General library of 2n port z-domain equivalent circuits for switched capacitor building blocks (*continued on pp 740–743*).


DISCRETE TIME CIRCUIT

Z–DOMAIN EQUIVALENT CIRCUIT

(e) TOGGLE SWITCHED DIFFERENCER (TSD)

(f) SINGLE PHASE GROUNDED CAPACITOR (SPGC)

(g) SINGLE PHASE FLOATING CAPACITOR (SPFC)

(h) SINGLE PHASE SWITCHED CAPACITOR (SPSC)

Fig. 6—(*continued*).

published[7-11, 17, 26, 27] SC networks which use nonoverlapping bi-phase switches. The equivalent circuits in Fig. 6 are derived in their most general $2n$-port form, assuming that all voltages update at one-half clock cycle intervals, as per $v(t)$ in Fig. 2. The e, o notation refers to the switch phasings as noted in the previous section. Similarly, super-

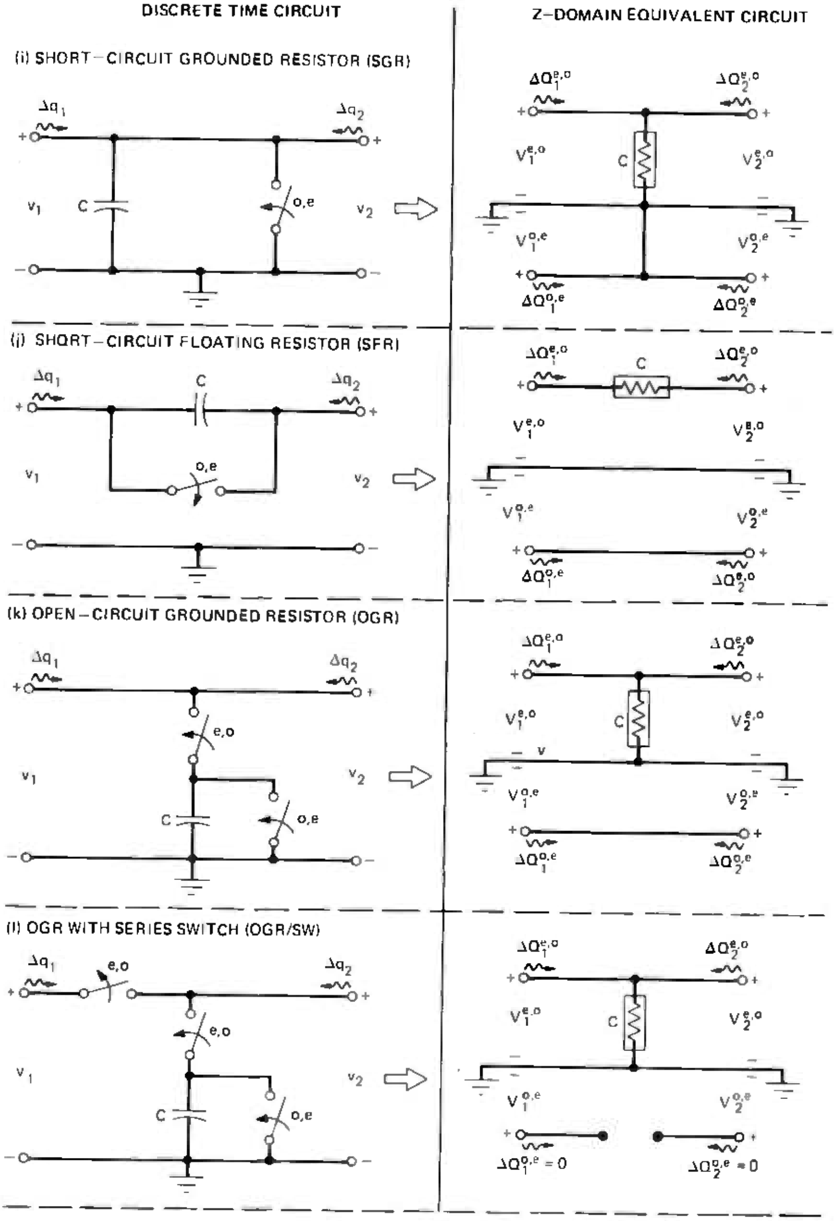
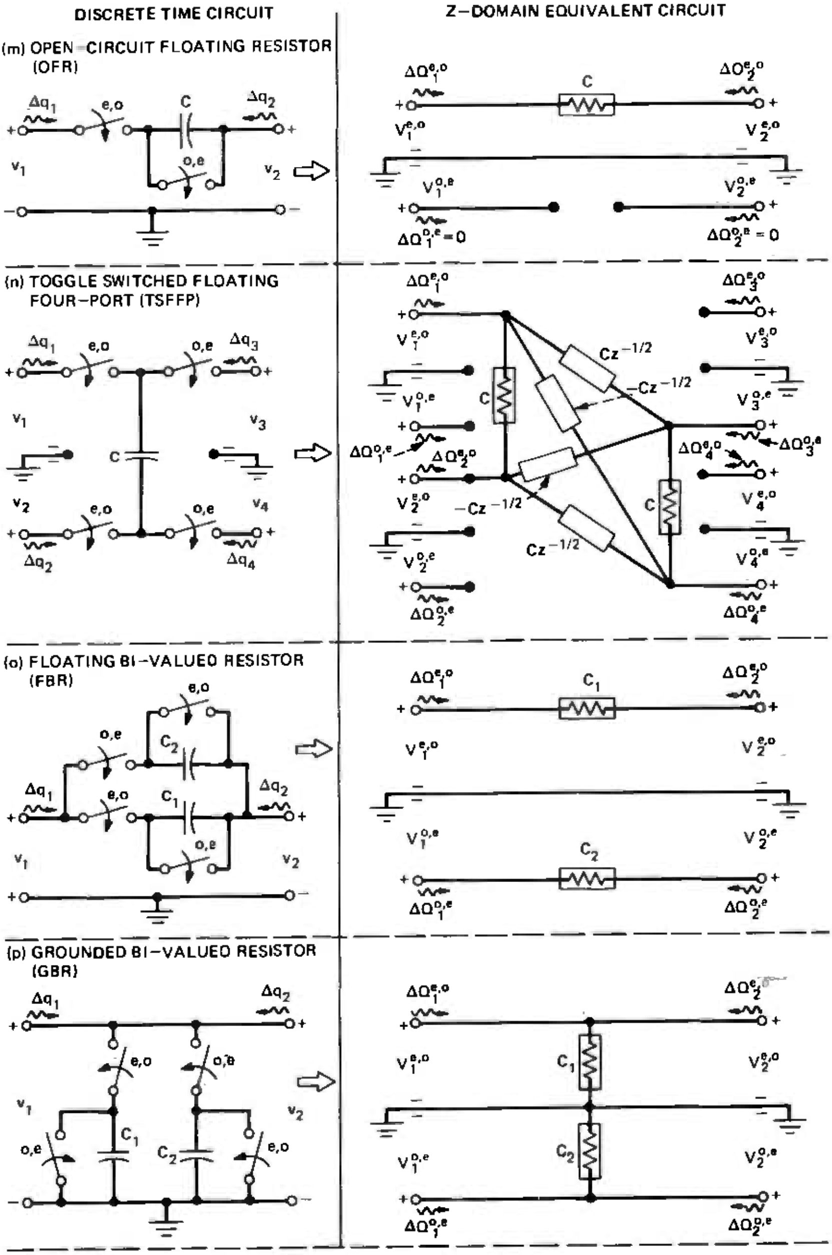


Fig. 6—(*continued*).

scripts e, o and o, e are used to denote the even or odd port variable (V_i, ΔQ_i) components and the complement odd or even port-variable components, respectively. This e, o notation conveniently provides the connectivity information for interconnecting the building blocks.



(m) OPEN-CIRCUIT FLOATING RESISTOR (OFR)

(n) TOGGLE SWITCHED FLOATING FOUR-PORT (TSFFP)

(o) FLOATING BI-VALUED RESISTOR (FBR)

(p) GROUNDED BI-VALUED RESISTOR (GBR)

Fig. 6—(*continued*).

In practice, there are many SC networks in which the charges and voltages update, due to the internal switching action of the SC network, only on full clock cycle intervals. This behavior, which is readily identified on a block-by-block basis, results in $2n$-port equivalent circuits with n open ports. Many of the SC blocks in Fig. 6 fall within

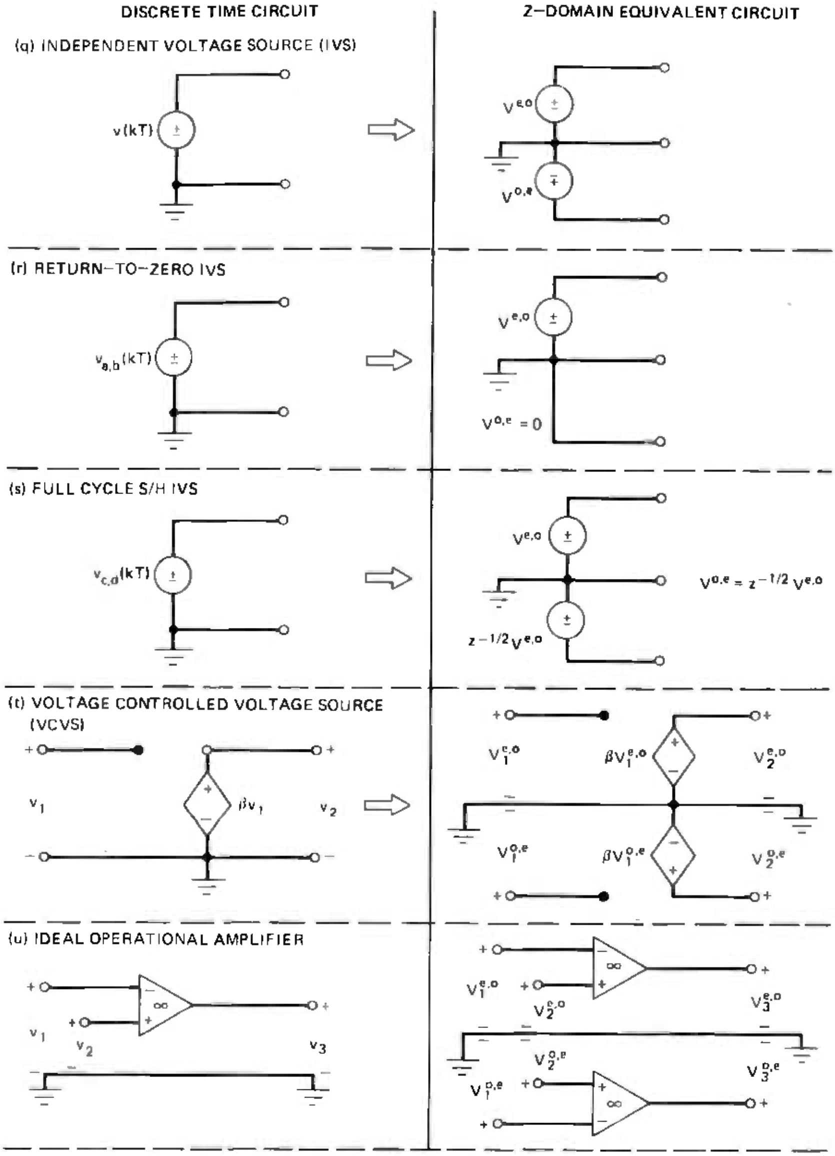


Fig. 6—(*continued*).

this category. When properly interconnected,[17] these $2n$-port equivalent circuits can be reduced to the n-port equivalent circuits in Fig. 7.

3.1 2n-port SC building block equivalent circuits

In this section, derivations are given for several of the SC equivalent circuits in Fig. 6. These derivations will be based on z-transformed nodal charge equations which can be derived by inspection from the

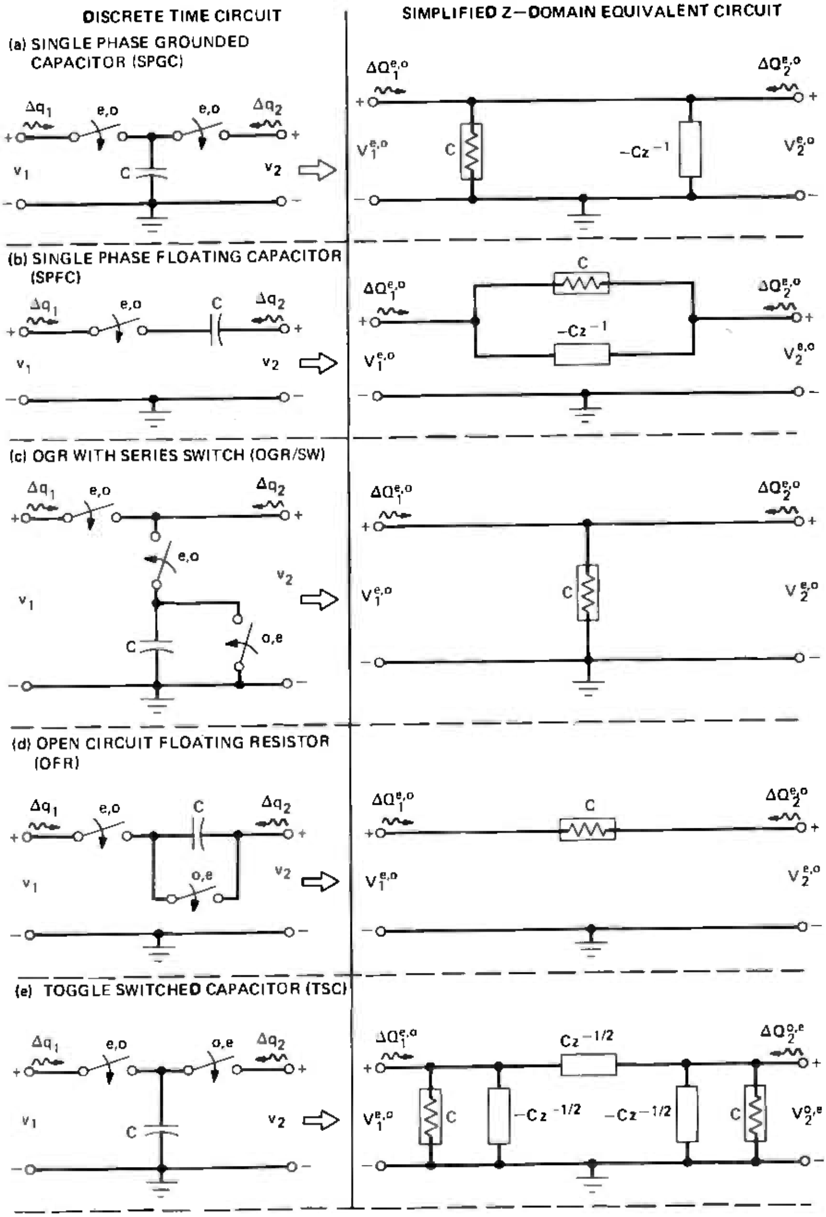


Fig. 7—Simplified library of n port z-domain equivalent circuits (*continued on pp. 745–746*).

sc circuit. As noted in the previous section, one can write a distinct nodal charge equation for each switch phase. Therefore an n-port sc block is characterized by $2n$ nodal charge relations. The desired $2n$-port z-domain equivalent circuit evolves directly from these relations.

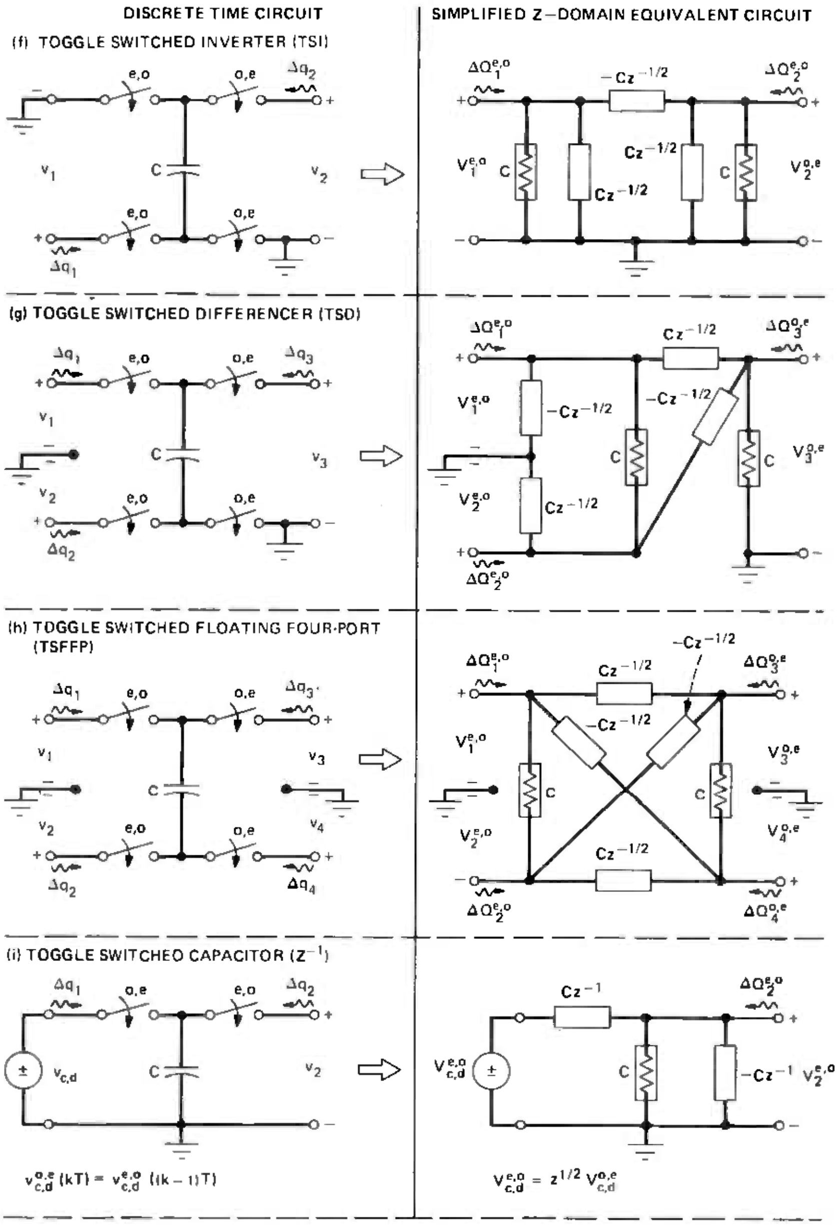


Fig. 7—(*continued*).

The equivalent circuit for a complex SC network is derived by properly interconnecting the appropriate block equivalent circuits. To avoid boring the reader with excessive repetition, derivations will only be provided for blocks b through f and l of Fig. 6. Once these derivations

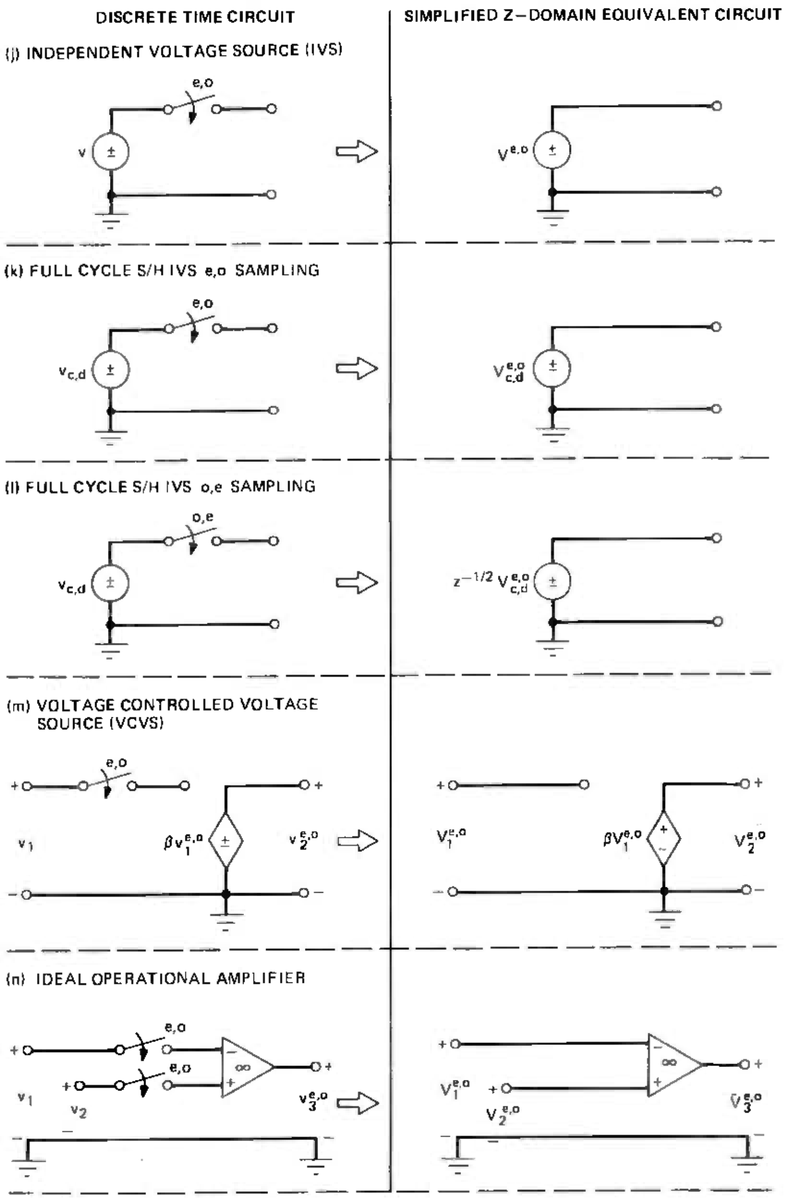

DISCRETE TIME CIRCUIT
SIMPLIFIED Z–DOMAIN EQUIVALENT CIRCUIT
(j) INDEPENDENT VOLTAGE SOURCE (IVS)
e,o
v
±
$V^{e,o}$
(k) FULL CYCLE S/H IVS e,o SAMPLING
e,o
$v_{c,d}$
$V^{e,o}_{c,d}$
(l) FULL CYCLE S/H IVS o,e SAMPLING
o,e
$v_{c,d}$
$z^{-1/2}\,V^{e,o}_{c,d}$
(m) VOLTAGE CONTROLLED VOLTAGE SOURCE (VCVS)
e,o
+
v_1
−
$\beta v^{e,o}_1$
$v^{e,o}_2$
$V^{e,o}_1$
$\beta V^{e,o}_1$
$V^{e,o}_2$
(n) IDEAL OPERATIONAL AMPLIFIER
e,o
e,o
∞
v_1
v_2
$v^{e,o}_3$
$V^{e,o}_1$
$V^{e,o}_2$
$V^{e,o}_3$

Fig. 7—(*continued*).

are understood, the validity of the remaining equivalent circuits can be established by inspection. The independent and dependent voltage source equivalent circuits are obtained directly from the relations in the previous section.

3.1.1 Floating capacitor equivalent circuit

One can derive the desired equivalent circuit in a straightforward manner directly from the nodal charge equations. In these equations, the even and odd voltage components (V^e and V^o) serve as independent variables and the even and odd charge variation components (ΔQ^e and ΔQ^o) serve as the dependent variables. Since the floating capacitor block in Fig. 6b contains no switches, the z-transformed nodal charge equations, where V^e, V^o, V_2^e and V_2^o are independent voltage excitations, are instantly written as

$$\Delta Q_1^{e,o}(z) = \\ CV_1^{e,o}(z) - Cz^{-1/2}V_1^{o,e}(z) - CV_2^{e,o}(z) + Cz^{-1/2}V_2^{o,e}(z), \quad (13a)$$

$$\Delta Q_2^{e,o}(z) = \\ CV_2^{e,o}(z) - Cz^{-1/2}V_2^{o,e}(z) - CV_1^{e,o}(z) + Cz^{-1/2}V_1^{o,e}(z), \quad (13b)$$

$$\Delta Q_1^{o,e}(z) = \\ CV_1^{o,e}(z) - Cz^{-1/2}V_1^{e,o}(z) - CV_2^{o,e}(z) + Cz^{-1/2}V_2^{e,o}(z), \quad (13c)$$

and

$$\Delta Q_2^{o,e}(z) = \\ CV_2^{o,e}(z) - Cz^{-1/2}V_2^{e,o}(z) - CV_1^{o,e}(z) + Cz^{-1/2}V_1^{e,o}(z). \quad (13d)$$

There are perhaps several circuit interpretations for this set of equations. One convenient interpretation is the balanced lattice equivalent circuit shown in Fig. 6b. Another circuit interpretation[17] for these equations is a four-port network comprised of an unbalanced floating LTP coupled to the even and odd transmission paths via ideal transformers. By interpreting eqs. (13) as a balanced lattice, one can eliminate the transformers. This balanced lattice is referred to in this paper as a balanced floating LTP; in contrast, the Kurth-Moschytz circuit is referred to as an unbalanced floating LTP. Both circuits are equivalent and valid under all port termination conditions.

3.1.2 Toggle switched capacitor[7] (TSC) equivalent circuit

Due to the switching action of the toggle switch, the capacitor C receives charge from v_1, only on the even (odd) times and charge from v_2 on the odd (even) times. When the switches are open, the corresponding ports are open and $\Delta Q = 0$. These observations are consistent with the z-transformed nodal charge equations:

$$\Delta Q_1^{e,o}(z) = CV_1^{e,o}(z) - Cz^{-1/2}V_2^{o,e}(z) \quad (14a)$$

$$\Delta Q_2^{e,o}(z) = 0 \quad (14b)$$

$$\Delta Q_1^{o,e}(z) = 0 \tag{14c}$$

$$\Delta Q_2^{o,e}(z) = CV_2^{o,e}(z) - Cz^{-1/2}V_1^{e,o}(z). \tag{14d}$$

These equations lead directly to the four-port equivalent circuit in Fig. 6c. As described in Ref. 17, an unbalanced LTP is seen to bridge the 1-*e,o* and 2-*o,e* ports. Note that ports 1-*o,e* and 2-*e,o* are always open; therefore, no transmission occurs at these ports. This is a property common to all bi-phase toggle switched SC blocks (e.g., equivalent circuits 6d and 6e in Fig. 6).

3.1.3 Toggle switched inverter[11] (TSI) equivalent circuit

The operation of this circuit is similar to the TSC element; with the exception that in the TSI the voltage is inverted as the charge on C is transferred from port 1 to port 2. This process is described by the following *z*-transformed nodal charge equations:

$$\Delta Q_1^{e,o}(z) = CV_1^{e,o}(z) + Cz^{-1/2}V_2^{o,e}(z) \tag{15a}$$

$$\Delta Q_1^{o,e}(z) = 0 \tag{15b}$$

$$\Delta Q_2^{e,o}(z) = 0 \tag{15c}$$

$$\Delta Q_2^{o,e}(z) = CV_2^{o,e}(z) + Cz^{-1/2}V_1^{e,o}(z). \tag{15d}$$

These equations are readily interpreted by the four-port equivalent circuit in Fig. 6d. Note for this block the 1-*e,o* and 2-*o,e* ports are bridged by an unbalanced LTP-like network in which the storage elements ($Cz^{-1/2}$) are all premultiplied by (-1). Since this network serves both as a link between even and odd transmission paths and as a signal inverter, it is referred to as an unbalanced inverting LTP.

3.1.4 Toggle switched differencer[11] (TSD) equivalent circuit

In this element, the charge on C is determined by the voltage difference $v_1^{e,o}(kT) - v_2^{e,o}(kT)$ during the *e,o* switch phase. When the *o,e* switches close this voltage, difference appears directly across port 3. This operation is described by the following *z*-transformed nodal charge equations:

$$\Delta Q_1^{e,o}(z) = CV_1^{e,o}(z) - CV_2^{e,o}(z) - Cz^{-1/2}V_3^{o,e}(z) \tag{16a}$$

$$\Delta Q_1^{o,e}(z) = 0 \tag{16b}$$

$$\Delta Q_2^{e,o}(z) = CV_2^{e,o}(z) - CV_1^{e,o}(z) + Cz^{-1/2}V_3^{o,e}(z) \tag{16c}$$

$$\Delta Q_2^{o,e}(z) = 0 \tag{16d}$$

$$\Delta Q_3^{e,o}(z) = 0 \tag{16e}$$

and

$$\Delta Q_3^{o,e}(z) = CV_3^{o,e}(z) - Cz^{-1/2}V_1^{e,o}(z) + Cz^{-1/2}V_2^{e,o}(z). \tag{16f}$$

The six-port equivalent circuit representation for these equations is given in Fig. 6e. Note that three of the six ports are open. The TSD element exhibits yet another form of LTP. In this element two *e,o* transmission paths are linked to a single *o,e* path through a differencing operation. It is perhaps appropriate to refer to this LTP as an unbalanced differencing LTP.

3.1.5 Single phase grounded capacitor (SPGC)

This SC element occurs frequently in SC networks, particularly in low-pass SC filters. In a sense, it serves as a companion element for the grounded capacitor in Fig. 6a. It is also a special case of the grounded capacitor. The nodal charge relations for this block are readily written as

$$\Delta Q_1^{e,o}(z) = C(1 - z^{-1})V^{e,o}(z) \tag{17a}$$

$$\Delta Q_2^{e,o}(z) = C(1 - z^{-1})V^{e,o}(z) \tag{17b}$$

$$\Delta Q_1^{o,e}(z) = 0 \tag{17c}$$

$$\Delta Q_2^{o,e}(z) = 0, \tag{17d}$$

where $V^{e,o}(z) = V_1^{e,o}(z) = V_2^{e,o}(z)$. It is noted that one can derive eqs. (17) from the grounded capacitor equivalent circuit in Fig. 6a by setting $\Delta Q_1^{o,e} = \Delta Q_2^{o,e} = 0$, which implies $V^{o,e}(z) = z^{-1/2}V^{e,o}(z)$. For the SPGC block, $V^{o,e}(z)$ represents the voltage stored and held on capacitor C and no longer refers to port voltages $V_1^{o,e}$ and $V_2^{o,e}$. As noted in Section II, this condition is equivalent to a full clock period S/H.

Equations (17a) through (17d) lead directly to the four port equivalent circuit in Fig. 6f. Due to the switches, two of the ports are open as described by eqs. (17c) and (17d). This network is equivalent to the open circuit LTP described in Ref. 17. The equivalent circuit for the floating capacitor is seen to similarly reduce to that in Fig. 6g when a series switch is added. Since these blocks occur frequently in complex SC networks, their recognition results in substantially simplified equivalent circuits.

3.1.6 Open circuit grounded resistor with series switch (OGR/SW) equivalent circuit

This block performs a function similar to the SGR in Fig. 6i, except that capacitor *C* is discharged while it is totally disconnected from the circuit. Therefore the shorted capacitor does not load the circuit during the discharging switch phase. The equivalent circuit, in Fig. 6l, for this block is obtained from the following *z*-transformed nodal charge equations

$$\Delta Q_1^{e,o}(z) = CV^{e,o}(z) \tag{18a}$$

$$\Delta Q_1^{o,e}(z) = 0 \tag{18b}$$

$$\Delta Q_2^{e,o}(z) = CV^{e,o}(z) \tag{18c}$$

and

$$\Delta Q_2^{o,e}(z) = 0, \tag{18d}$$

where

$$V^{e,o}(z) = V_1^{e,o}(z) = V_2^{e,o}(z).$$

This block is memoryless, and it is an open circuit during the *o,e* clock phase when capacitor C discharges. Thus during the *e,o* clock phase, the block serves as a resistor and during the *o,e* clock phase it does nothing. In many SC network arrangements, where it is only necessary to transmit during one clock phase, this block serves as an excellent resistor equivalent.

Equivalent circuits 6o and 6p represent straightforward, nevertheless useful, generalizations of circuits 6l and 6m. These SC networks are seen to provide, for the even and odd clock phases, different value resistor-like components. This type of component suggests the possibility of time-sharing capacitors and operational amplifiers to achieve different even and odd circuit behaviors.

This concludes the derivations for equivalent circuits in Fig. 6. At this point, the interested reader should be able to derive the remaining SC equivalent circuits easily.

3.2 Simplified SC building block equivalent circuits

In the previous section, it was observed that many of the four-port equivalent circuits result in n (of $2n$) open circuit ports. Obviously, any signals applied to one or more of these open ports will neither be processed nor transmitted. Therefore, SC networks comprised of these blocks will only provide transmission and filtering, when the switches are phased such that the blocks interconnect to provide one nonopen signal path[17] from input to output. Assuming this connection rule, the open ports are nonfunctional and can be removed from the equivalent circuits. The immediate identification of these blocks in a complex SC network results in much labor-saving equivalent circuit simplification. More specifically, $2n$-port equivalent circuits reduce directly to n-port equivalent circuits. To emphasize this point, the appropriate four-port equivalent circuits in Fig. 6 have been reconfigured as two-port equivalent circuits in Fig. 7. Many complex SC networks[9,11] can be modeled exclusively with these simplified equivalent circuits. For this class of SC networks, circuit analysis is no more complex than that for continuous (linear) time-invariant networks. Since the blocks listed in Fig. 7 perform all the necessary network functions, it is expected that one can synthesize general z-domain transfer functions using only these blocks. This restriction, with little sacrifice in generality, should lead to efficient z-domain synthesis procedures for SC networks.

In addition to the reconfigured equivalent circuits from Fig. 6, Fig.

7 contains two additional building blocks. These blocks are shown in Fig. 7h and 7i. Let us briefly discuss each of these blocks on an individual basis.

3.2.1 Toggle switched floating four-port (TSFFP) equivalent circuit

This element is the most general of the toggle switched (single) capacitor elements. Thus the equivalent circuits for the TSC, TSI, and the TSD can be derived directly from the equivalent circuit in Figure 7h, by simply shorting to ground the appropriate port or ports. The z-transformed nodal charge equations for the block are expressed as

$$\Delta Q_1^{e,o}(z) = CV_1^{e,o}(z) - CV_2^{e,o}(z) - Cz^{-1/2}V_3^{o,e}(z) + Cz^{-1/2}V_4^{o,e}(z) \tag{19a}$$

$$\Delta Q_1^{o,e}(z) = 0 \tag{19b}$$

$$\Delta Q_2^{e,o}(z) = CV_2^{e,o}(z) - CV_1^{e,o}(z) - Cz^{-1/2}V_4^{o,e}(z) + Cz^{-1/2}V_3^{o,e}(z) \tag{19c}$$

$$\Delta Q_2^{o,e}(z) = 0 \tag{19d}$$

$$\Delta Q_3^{e,o}(z) = 0 \tag{19e}$$

$$\Delta Q_3^{o,e}(z) = CV_3^{o,e}(z) - CV_4^{o,e}(z) - Cz^{-1/2}V_1^{e,o}(z) + Cz^{-1/2}V_2^{e,o}(z) \tag{19f}$$

$$\Delta Q_4^{e,o}(z) = 0 \tag{19g}$$

and

$$\Delta Q_4^{o,e}(z) = CV_4^{o,e}(z) - CV_3^{o,e}(z) - Cz^{-1/2}V_2^{e,o}(z) + Cz^{-1/2}V_1^{e,o}(z). \tag{19h}$$

To be completely general, eqs. (19) describe an eight-port equivalent circuit with four open ports. Such an eight-port description is shown in Fig. 6n. The more useful four-port equivalent circuit in Fig. 7h is obtained by deleting the open ports.

Comparing Figs. 6b and 7h, one observes that the four non-open ports of the TSFFP are coupled together via a 90-degree rotated, balanced floating LTP. In fact, if the TSFFP is rotated 90 degrees with ports 1 and 3 serving as the incoming ports and ports 2 and 4 as the outgoing ports, we indeed have the equivalent circuit for the floating capacitor in Fig. 6b. If ports 2 and 4 are then shorted to ground, one can then easily derive the equivalent circuit for the grounded capacitor in Fig. 6a. All the toggle-switched (the TSC, TSI, and TSD) elements can be readily derived from the TSFFP. For example, if ports 2 and 4 are shorted to ground, the TSFFP equivalent circuit reduces to that of the TSC in Fig. 7e. Also, the TSI in Fig. 7f is obtained when ports 1 and 4 are shorted to ground. In summary, by providing the proper termination conditions, one can derive the equivalent circuits for any of the single capacitor SC elements in Figs. 6 and 7 with the TSFFP.

3.2.2 Toggle switched blocks driven by full clock cycle S/H voltage sources

Equivalent full-cycle time delays can be experienced when toggle-switched SC blocks are driven with full cycle S/H voltage sources. A situation of this type is illustrated in Fig. 7i. The behavior of this circuit can be described in the following manner. When source $v_{c,d}(t)$ changes to value $v_{c,d}^{e,o}$, the *o,e* switch is open; thus, the charge on capacitor C remains unchanged. One-half clock period later when switch *o,e* closes, capacitor C acquires the charge $Cv_{c,d}^{e,o}$. Another one-half clock period later, the *o,e* switch opens, the *e,o* switch closes, and $v_{c,d}^{e,o}$ appears at the output with a net time delay of one full clock period. Obviously, when the source changes value in synchronism with the initial *o,e* switch, the net time delay is one-half of a clock period. The SC circuit in Fig. 7i can be modeled according to the equivalent circuit in Fig. 8. Writing a nodal charge equation at node 2 yields

$$CV_2^{e,o}(z) = Cz^{-1/2}(z^{-1/2}V_{c,d}^{e,o}(z)) = Cz^{-1}V_{c,d}^{e,o}(z). \tag{20}$$

The equivalent circuit in Figure 7i conveniently characterizes this relationship. Similar equivalent circuits can be derived for the TSI and TSD blocks as shown in Figs. 9a and 9b respectively. Full cycle time delays can readily[11] occur when appropriately phased toggle switched blocks are driven by operational amplifier integrator circuits.

IV. APPLICATIONS TO THE ANALYSIS AND SYNTHESIS OF SC NETWORKS

In this section, the concepts developed in the previous sections are applied to the analysis of several passive and active SC networks. Many examples are simple, to emphasize the insight provided by the equivalent circuits.

4.1 Passive SC networks

In this section we examine two single pole passive SC networks. The equivalent circuits in Figs. 6 and 7 allow one to examine a given SC network under an assortment of input-output conditions, as in Fig. 3. As we will see, such an examination can reveal some rather interesting circuit behavior that is not immediately obvious.

4.1.1 First-order, low-pass SC networks

As the initial example, consider the simple first-order, low-pass network depicted in Fig. 10a. An equivalent circuit for this network can be obtained by simply cascading blocks 6c and 6f, as shown in Fig. 10b. This circuit can obviously be reduced to that in Fig. 10c. One could have immediately written the equivalent circuit in Fig. 10c by

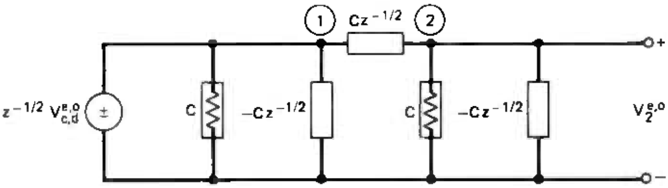


Fig. 8—Toggle switched capacitor driven by a full clock period S/H voltage source.

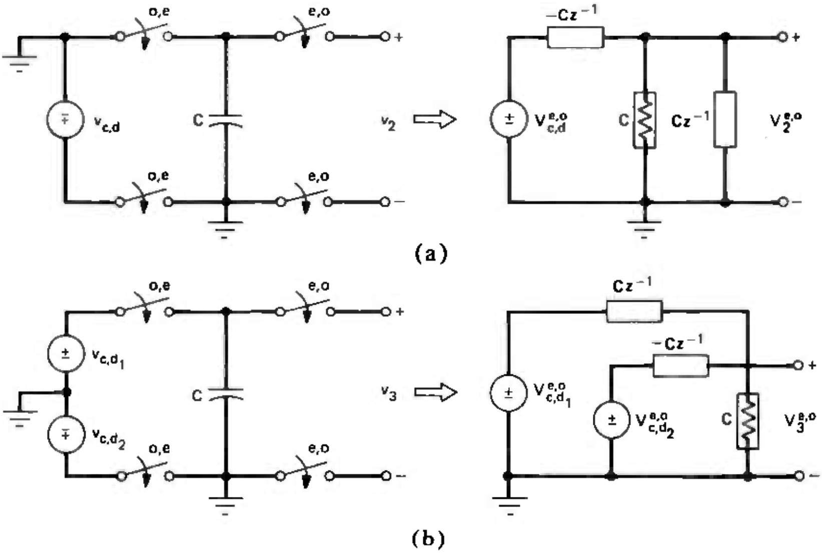


Fig. 9—(a) Toggle switched inverter and (b) toggle switched differencer, driven by full clock period S/H voltage sources.

cascading the simplified block equivalent circuits 7a and 7e. Writing a single nodal charge equation at node 2, namely,

$$(C_1 - C_1 z^{-1/2} + C_1 z^{-1/2} + C_2 - C_2 z^{-1}) V_{out}^o(z) = C_1 z^{-1/2} V_{in}^e(z) \quad (21)$$

yields the familiar low-pass z-domain transfer function

$$H_3(z) = \frac{V_{out}^o(z)}{V_{in}^e(z)} = \frac{C_1 z^{-1/2}}{C_1 + C_2 - C_2 z^{-1}}. \quad (22)$$

Note that V_{in}^o and V_{out}^e are removed by sampling operations at the input and output respectively. Thus, the transmission through this network is completely described by a single transfer function, namely H_3.

4.1.2 First-order, high-pass SC network

The simple first-order, high-pass circuit shown in Fig. 11 is a rather interesting circuit, as we shall soon see. Its interesting behavior stems from the input-to-output switch free path which permits both V_{in}^e and

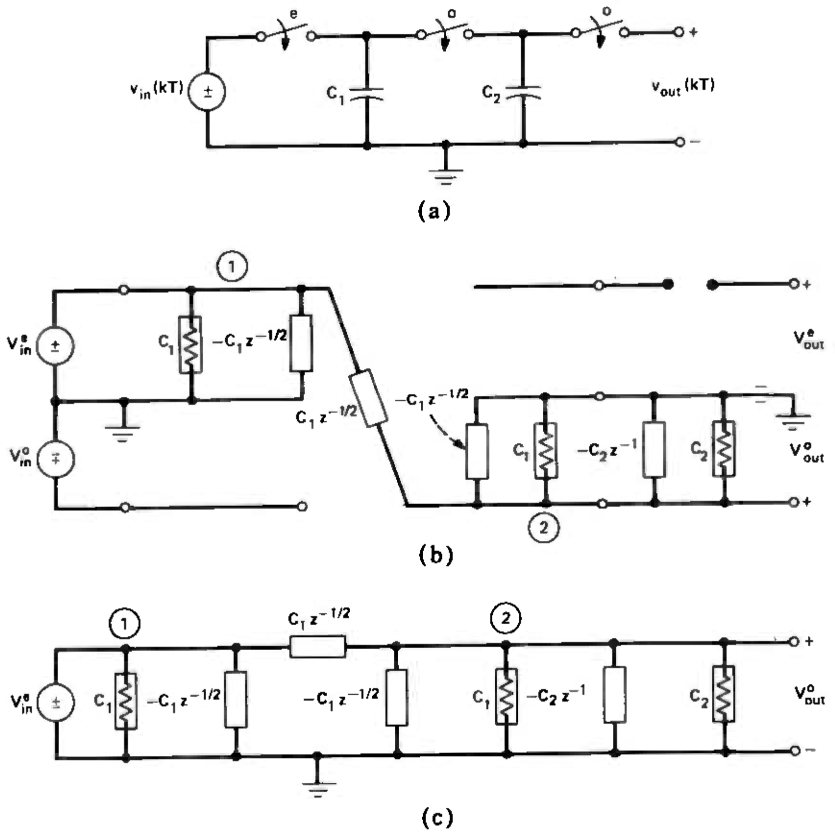


Fig. 10—Single pole "passive" SC low-pass network.

V^o_{in} to determine the e and o components of V_{out}. To study this circuit, we write the equivalent circuit in Fig. 11b by cascading blocks 6b and 6k.

Analysis of the equivalent circuit yields the following relations:

$$V^e_{out}(z) = \frac{C_1(1 - z^{-1})}{C_1 + C_2 - C_1z^{-1}} V^e_{in}(z) + OV^o_{in}(z) \tag{23a}$$

$$= H_1(z)V^e_{in}(z) \tag{23b}$$

and

$$V^o_{out}(z) = \frac{-C_2z^{-1/2}}{C_1 + C_2 - C_1z^{-1}} V^e_{in}(z) + V^o_{in}(z) \tag{24a}$$

$$= H_3(z)V^e_{in}(z) + H_4(z)V^o_{in}(z). \tag{24b}$$

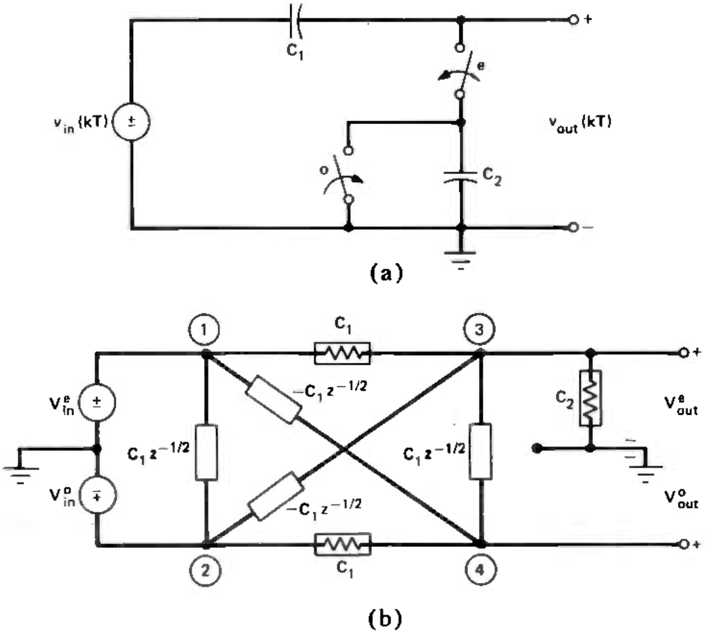


Fig. 11—Single pole "passive" sc high-pass network.

Note:

$$H_1(z) \triangleq \left.\frac{V_{out}^e(z)}{V_{in}^e(z)}\right|_{V_{in}^o = 0} = \frac{C_1(1 - z^{-1})}{C_1 + C_2 - C_1z^{-1}} \tag{25a}$$

is a first-order, high-pass function, while

$$H_3(z) \triangleq \left.\frac{V_{out}^o(z)}{V_{in}^e(z)}\right|_{V_{in}^o = 0} = \frac{-C_2z^{-1/2}}{C_1 + C_2 - C_1z^{-1}} \tag{25b}$$

is a first-order, low-pass function. This is a most interesting result, indeed. From eqs. (24), we observe that by forcing $V_{in}^o = 0$, as in Fig. 12a, this circuit behaves like a first-order, high-pass filter when the output is sampled on the even times and behaves like a first-order, low-pass filter when the output is sampled on the odd times. A circuit that achieves this bifunctional characteristic is shown in Fig. 12b. To achieve this behavior, a simple return-to-zero source of the form shown previously in Fig. 3a, or equivalently in Fig. 4a, is used to drive the high-pass circuit depicted in Fig. 11.

4.2 Active SC networks

Due to obvious reasons, much of our interest is in active[8-11,26,27] SC networks. These networks are typically[8-11,26,27] comprised of capacitors,

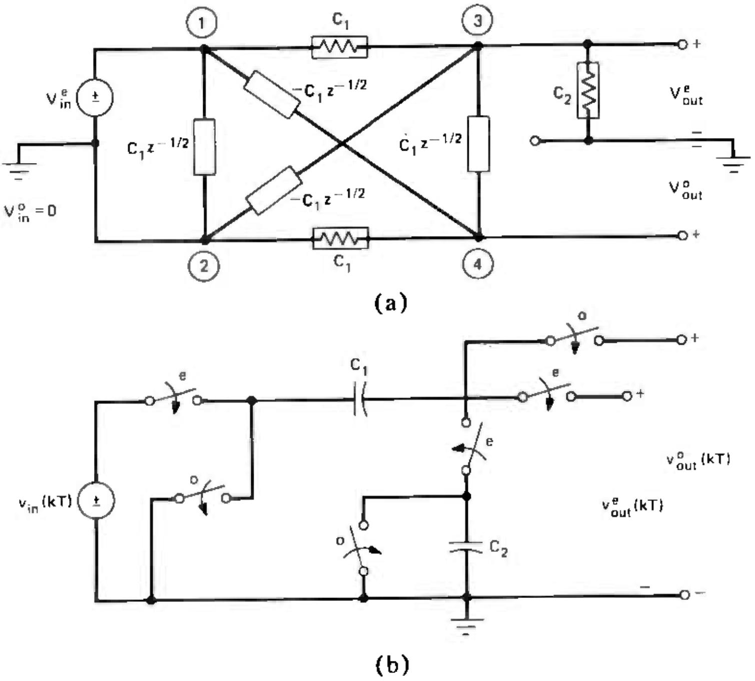


Fig. 12—Single pole "passive" sc high-pass/low-pass network.

switches and operational amplifiers. Many of the active sc networks appearing[8-11] in the literature are comprised of simple sc building blocks, of the form listed in Fig. 7, buffered by operational amplifiers. When these operational amplifiers can be assumed to be ideal, the virtual grounds result in further simplifications in the equivalent circuits. In the voltage-charge domain, a "virtual" ground at the input of an ideal operational amplifier shall be defined by the condition $\Delta Q = 0$, $V = 0$. Let us now consider the equivalent circuit representations for the following selection of first-order active sc networks.

4.2.1 Lossless integrator

The equivalent circuit for the lossless integrator in Fig. 13a is derived, in full generality, using blocks 6b, 6c, and 6u as shown in Fig. 13b. Of course, one may accommodate the finite gain of an actual operational amplifier using the voltage-controlled voltage source, cited in Fig. 6 as block 6t. The rather unwieldy circuit depicted in Fig. 13b can be immediately simplified by removing all elements shunting virtual ground points and voltage sources. This network is then redrawn in the form shown in Fig. 13c, which can be again reconfigured to yield the circuit in Fig. 13d. Finally, the second stage of Fig. 13d is noted to be a voltage-controlled voltage source with $\beta = z^{-1/2}$ as in

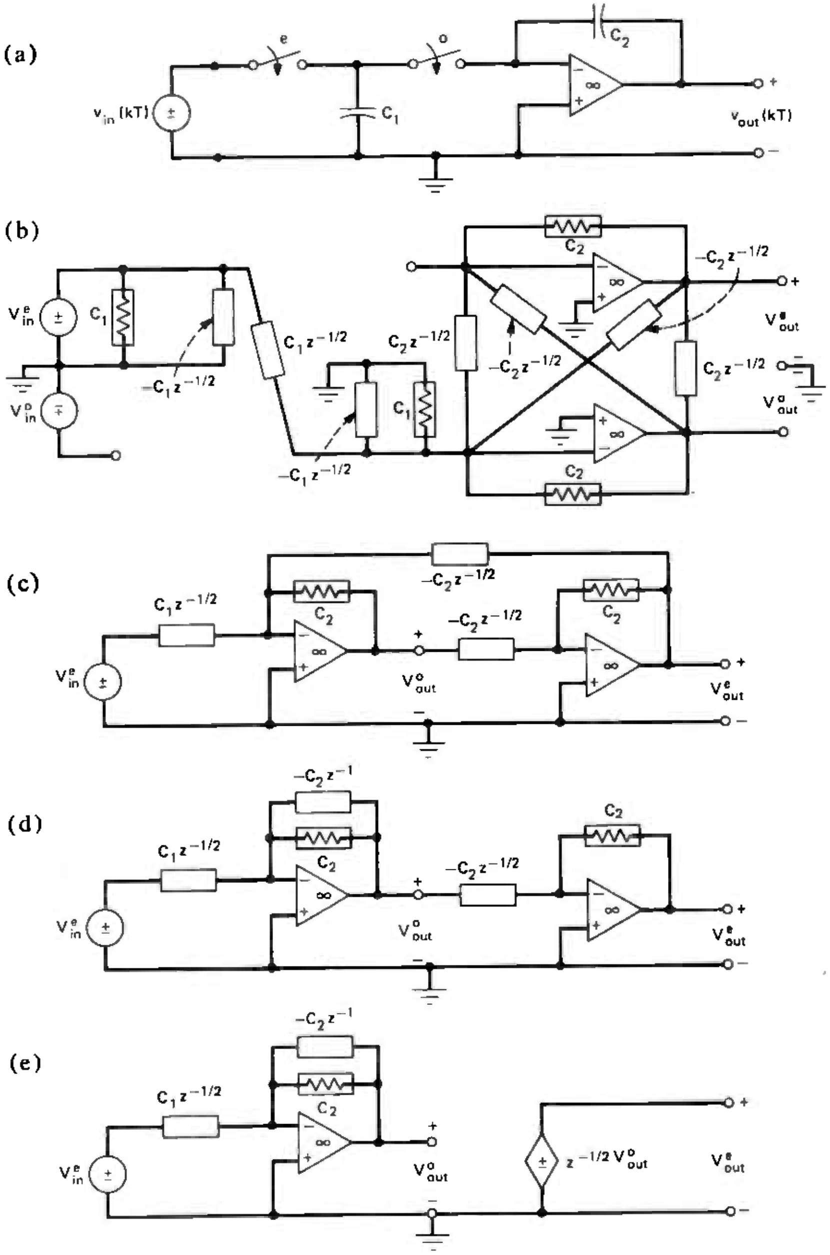
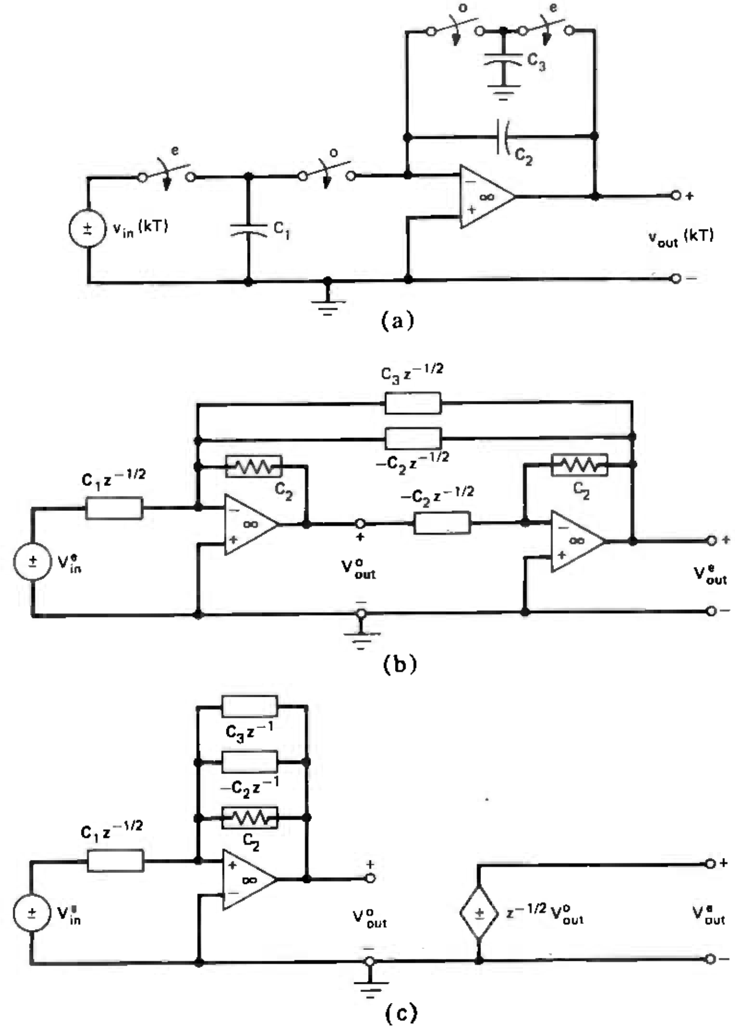


Fig. 13—Active-sc integrator.

Fig. 8. The final equivalent circuit in Fig. 13e implies that the lossless integrator could have been derived directly from the simplified equivalent circuits in Fig. 7.

The transfer functions for the lossless integrator are then readily



(a)

(b)

(c)

Fig. 14—Active-sc lossy integrator with TSC damping.

determined to be

$$H_3(z) = \frac{V^o_{out}(z)}{V^e_{in}(z)} = \frac{-\,(C_1/C_2)z^{-1/2}}{1 - z^{-1}} \tag{26a}$$

and

$$H_1(z) = \frac{V^e_{out}(z)}{V^e_{in}} = \frac{-\,(C_1/C_2)z^{-1}}{1 - z^{-1}}. \tag{26b}$$

Note that when the output of the lossless integrator is sampled at the odd times, the transfer function is H_3 and when sampled on the even times the transfer function is H_1.

4.2.2 Lossy integrator with TSC

As should be expected, the equivalent circuit for the lossy integrator, shown in Fig. 14a, is similar to that derived for the lossless integrator. To shift the pole to the left of $z = 1 + j0$, a toggle-switched capacitor (TSC) has been placed across the feedback capacitor. Obviously, the intent is for the TSC to play a role comparable to a resistor in active-RC lossy integrators. To analyze this network, let us first derive the equivalent circuit. This equivalent circuit can be derived step by step, as was done for the lossless integrator and shown successively in Figs. 14b and 14c, respectively. The final equivalent circuit in Fig. 14c, like that in Fig. 13e, can be readily derived from the simplified equivalent circuits in Fig. 7 by direct substitution. This result simplifies tremendously the equivalent circuits for complex, high-order active SC networks.

The transfer functions for the lossy integrator are then readily obtained from the circuit in Fig. 14c, namely,

$$H_3(z) = \frac{V^o_{out}(z)}{V^e_{in}(z)} = \frac{-(C_1/C_2)z^{-1/2}}{1-[1-(C_3/C_2)]z^{-1}} \tag{27a}$$

and

$$H_1(z) = \frac{V^e_{out}(z)}{V^e_{in}(z)} = \frac{-(C_1/C_2)z^{-1}}{1-[1-(C_3/C_2)]z^{-1}}\,. \tag{27b}$$

It is interesting to examine $H_3(z)$ for different values of C_3. Consider the following three conditions $C_3 = C_2$, $C_3 = 2C_2$, and $C_3 > 2C_2$. For $C_3 = C_2$:

$$H_3(z) = -\frac{C_1}{C_2}z^{-1/2} \tag{28}$$

an ideal half-delay element. For $C_3 = 2C_2$,

$$H_3(z) = \frac{-(C_1/C_2)z^{-1/2}}{1+z^{-1}}, \tag{29}$$

and the circuit is no longer stable. Finally, when $C_3 > 2C_2$ the pole of $H_3(z)$ lies outside the unit circle and the circuit is clearly unstable. Obviously, the TSC is much more than a resistor; a point which has been illustrated[17] in other ways.

Let us look briefly at the effect of alternating the phases of the switches which make up the feedback TSC. This SC network along with its equivalent circuits are shown in Fig. 15. The transfer functions are then readily written

$$H_3(z) = \frac{V^o_{out}(z)}{V^e_{in}(z)} = \frac{-(C_1/C_2)z^{-1/2}}{1-[1-(C_3/C_2)]z^{-1}} \tag{30a}$$

and

$$H_1(z) = \frac{V^e_{out}(z)}{V^e_{in}(z)} = \frac{-(C_1/C_2)[(C_2 - C_3)/C_2]z^{-1}}{1 - [1 - (C_3/C_2)]z^{-1}}. \qquad (30b)$$

Comparing the pole location for $H_i(z)$ in eqs. (27) and (30), one observes that the TSC switch phasing has no effect on this parameter. However, the dc gain for the even component V_{out} is altered by the factor $(C_2 - C_3)/C_2$. It should be noted that these observations were not totally expected.

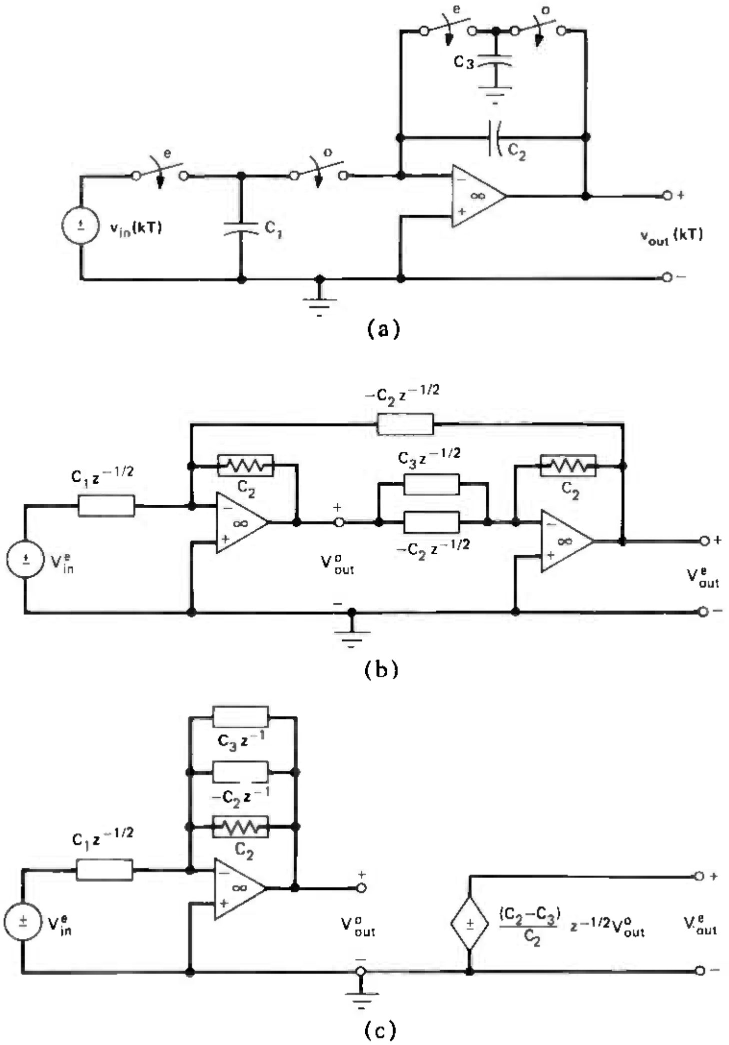


Fig. 15—Active-SC lossy integrator with TSC damping. The switches of the feedback TSC are phased opposite to that shown in Fig. 14.

4.2.3 Lossy integrator with OFR

Another lossy integrator realization is shown in Fig. 16a. Of perhaps only theoretical interest is the comparison of the behavior of this circuit with its counterpart in Fig. 14a. The equivalent circuit, shown in successive stages of simplification in Figs. 14b and 14c respectively, can be derived directly from the simplified block equivalent circuits in Fig. 7. The transfer functions for this circuit are

$$H_3(z) = \frac{V^o_{out}(z)}{V^e_{in}(z)} = \frac{[-C_1/(C_2 + C_3)]z^{-1/2}}{1 - [C_2/(C_2 + C_3)]z^{-1}} \tag{31a}$$

and

$$H_1(z) = \frac{V^e_{out}(z)}{V^e_{in}(z)} = \frac{[-C_1/(C_2 + C_3)]z^{-1}}{1 - [C_2/(C_2 + C_3)]z^{-1}}. \tag{31b}$$

The transfer functions expressed in eqs. (31) are seen to be truly representative of lossy integrators. $H_3(z)$ and $H_1(z)$ are absolutely stable for all finite values of C_1, C_2 and C_3.

4.2.4 Bilinear lossless integrators[26, 27]

Previously, in Section 4.1.1, an integrator was analyzed which achieved integration in the sample-data sense according to the transformations[28]

$$\frac{1}{s} = \tau \frac{z^{-1/2}}{1 - z^{-1}} \tag{32a}$$

or

$$\frac{1}{s} = \tau \frac{z^{-1}}{1 - z^{-1}}. \tag{32b}$$

It is well known[26, 28] that eq. (32b) only adequately approximates the function of an analog integrator for frequencies satisfying $\omega\tau \ll 1$. Although eq. (32a) overcomes[28] this difficulty, it cannot always be rigorously applied.[11, 28] An accurate and mathematically convenient integrator implementation is obtained via the well-known bilinear transform [22, 26, 28]

$$\frac{1}{s} = \frac{\tau}{2}\frac{1 + z^{-1}}{1 - z^{-1}}. \tag{33}$$

There are several ways bilinear integration can be realized[26, 27] with active-SC networks, as demonstrated in Figs. 17, 18, and 19. It is interesting to examine the behavior of each of these circuits.

Let us initially consider the bilinear integrator[26] shown in Fig. 17a. The z-domain equivalent circuit, obtained by interconnecting blocks 6b, 6c, 6m, and 6u, is shown in Fig. 17b. By straightforward nodal-

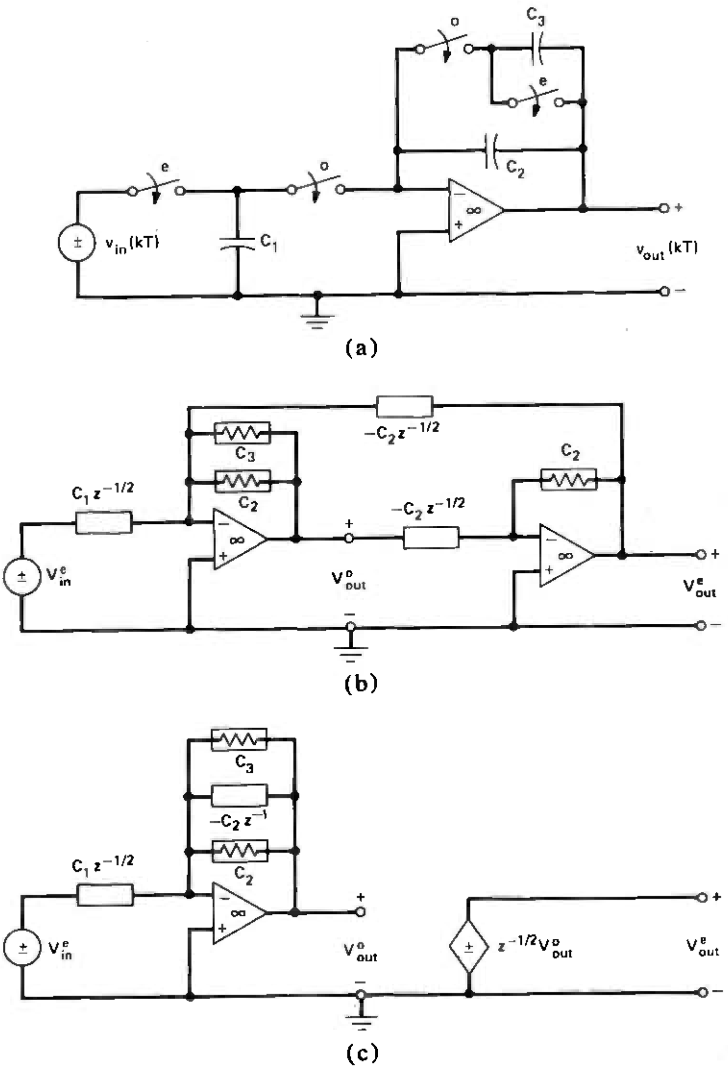


Fig. 16—Active-SC lossy integrator with OFR damping.

charge analysis, the following relations are easily obtained:

$$V^o_{out} = \frac{-(C_1/C_2)z^{-1/2}}{1-z^{-1}} V^e_{in} - \frac{(C_1/C_2)}{1-z^{-1}} V^o_{in} \tag{34a}$$

and

$$V^e_{out} = z^{-1/2} V^o_{out} . \tag{34b}$$

From eq. (34a), we observe that the desired bilinear integration is only obtained when

$$V^e_{in} = z^{-1/2} V^o_{in} . \tag{35}$$

Substituting eq. (35) into eq. (34a) yields the desired result

$$H_4(z) = \frac{V^o_{put}}{V^o_{in}} = \frac{-(C_1/C_2)(1 + z^{-1})}{1 - z^{-1}} ; \tag{36a}$$

also,

$$H_2(z) = \frac{V^e_{out}}{V^o_{in}} = \frac{-(C_1/C_2)z^{-1/2}(1 + z^{-1})}{1 - z^{-1}} . \tag{36b}$$

In summary, this circuit, with the switches phased as shown in Fig. 17a, will provide bilinear integration only when the input and output are sampled at the odd $(2k + 1)T$ times and the input is held for the entire clock period.

A second bilinear integrator realization[27] is shown in Fig. 18a. The z-domain equivalent circuit, shown in Fig. 18b, is obtained by interconnecting blocks 6b, 6e, and 6u. The transfer relations for this circuit are readily determined to be

$$V^o_{out} = \frac{-(C_1/C_2)(1 + z^{-1})}{(1 - z^{-1})} V^o_{in} - \frac{2(C_1/C_2)z^{-1/2}}{1 - z^{-1}} V^e_{in} \tag{37a}$$

and

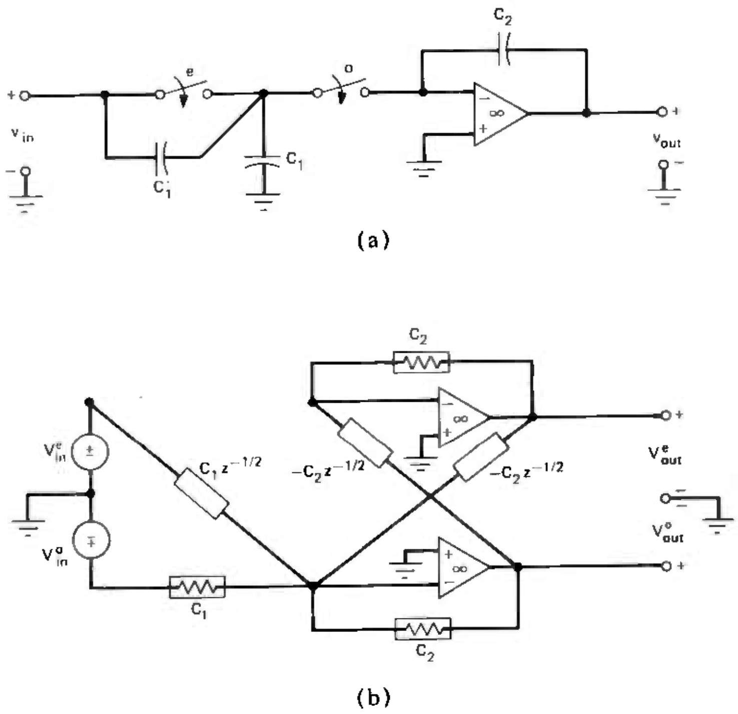
$$V^e_{out} = \frac{-(C_1/C_2)(1 + z^{-1})}{(1 - z^{-1})} V^e_{in} - \frac{2(C_1/C_2)z^{-1/2}}{1 - z^{-1}} V^o_{in} . \tag{37b}$$

The output, sampled at all (both even and odd) kT times, is obtained by summing eqs. (37a) and (37b) according to eq. (9) and cancelling the common factor $(1 + z^{-1/2})$; i.e.,

$$H(z) = \frac{V_{out}}{V_{in}} = \frac{V^e_{out} + V^o_{out}}{V^e_{in} + V^o_{in}} = \frac{-(C_1/C_2)(1 + z^{-1/2})}{1 - z^{-1/2}} . \tag{38}$$

Comparing equations (38) and (36a), we see that the effective sampling rate has been doubled with the circuit in Fig. 18a. Also, bilinear integration is obtained independent of the input sampling conditions. It is noted that sampling the output of this circuit at only the even $2kT$ times or only the odd $(2k + 1)T$ times will result in an erroneous output.

A third bilinear integrator is obtained by simply deleting one TSD from Fig. 18a, as shown in Fig. 19a. It is interesting to analyze this circuit and compare the results with that given in eqs. (37) and (38) for the integrator in Fig. 18a. The z-domain equivalent circuit in Fig. 19b is readily obtained by deleting the appropriate elements from the



(a)

(b)

Fig. 17—Active-sc bilinear integrator (Copeland, Ref. 26).

equivalent circuit in Fig. 18b. The transfer relations for this circuit are

$$V_{out}^{e} = \frac{-(C_1/C_2)(1 + z^{-1})}{1 - z^{-1}} V_{in}^{e} \tag{39a}$$

and

$$V_{out}^{o} = \frac{-2(C_1/C_2)z^{-1/2}}{1 - z^{-1}} V_{in}^{e} . \tag{39b}$$

Summing eqs. (39a) and (39b) yields

$$V_{out} = V_{out}^{e} + V_{out}^{o} = \frac{-(C_1/C_2)(1 + z^{-1/2})}{1 - z^{-1/2}} V_{in}^{e} . \tag{40}$$

Note that bilinear integration is obtained when the output is either sampled at the even $2kT$ times or at all (both even and odd) kT times. Again, bilinear integration is obtained independent of the input sampling conditions. This concludes the first-order active sc network examples. It should be noted that circuits similar to those in Figs. 13 through 16 could have been derived using the TSI and TSD elements shown in Figs. 6d and 6e or Figs. 7f and 7g. To further illustrate the procedure, the equivalent circuit for the fourth-order low-pass, leap-

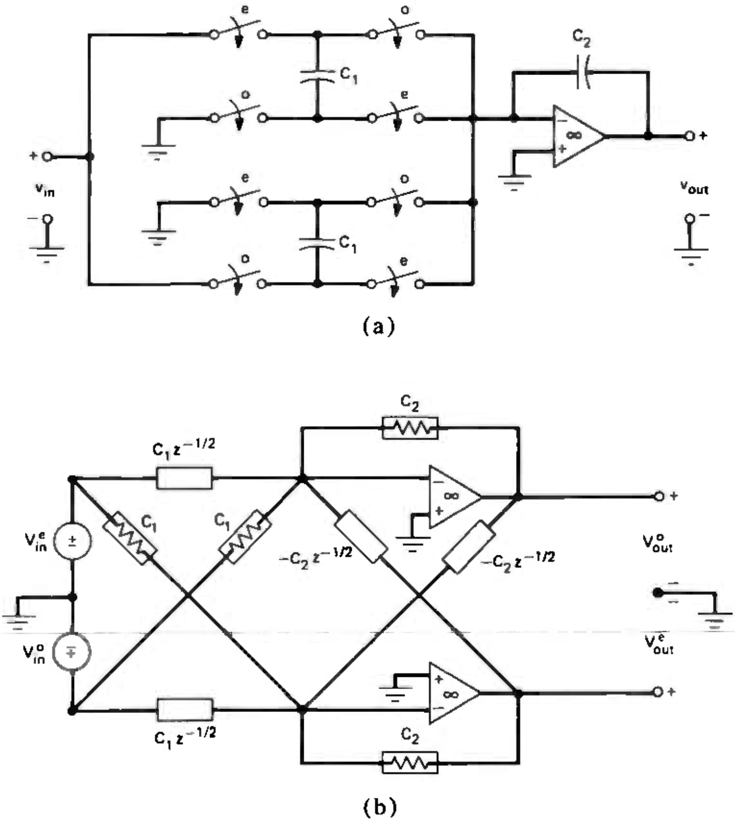
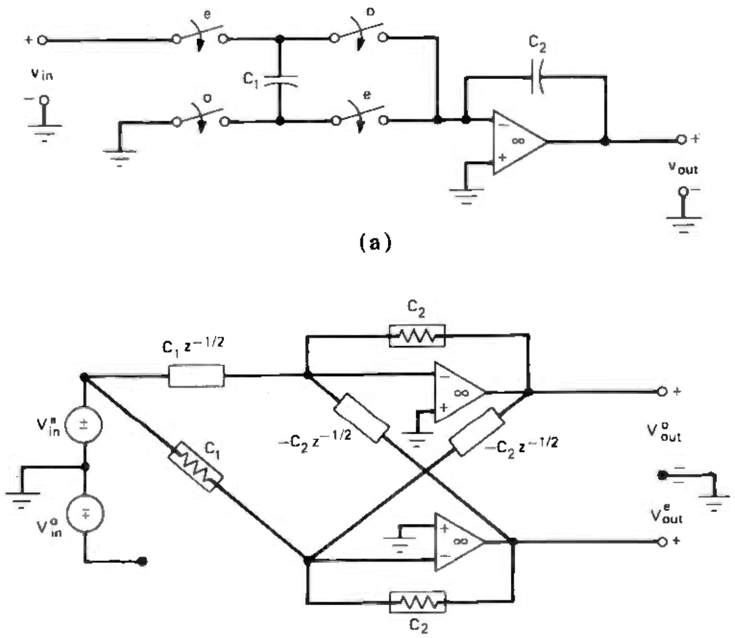


Fig. 18—Active-SC bilinear integrator (Temes and Young, Ref. 27).

frog, active-SC filter, depicted in Fig. 20a, is given in Fig. 20b. This equivalent circuit is readily derived from the equivalent circuit blocks in Fig. 7 and the principles discussed in this section. The equivalent circuit in Fig. 20b is seen to be no more complex than the equivalent active-RC circuit. Note that, in the absence of the output even switch, the only modification to the z-domain equivalent circuit in Fig. 20b is an additional voltage-controlled voltage source at the output. This voltage-controlled voltage source defines the relation $V^o_{out} = z^{-1/2}V^e_{out}$, as shown previously in Fig. 14c. The verification of this circuit is left as an exercise for the reader.

V. CONCLUDING REMARKS

The powerful tools which we commonly refer to as network or circuit theory have been indispensable in advancing the analog filter art to its present level of quality and sophistication. Such fundamental concepts as transfer functions, poles, and zeros have provided the filter designer with quick insight as to the behavior of a given filter. He can then



(a)

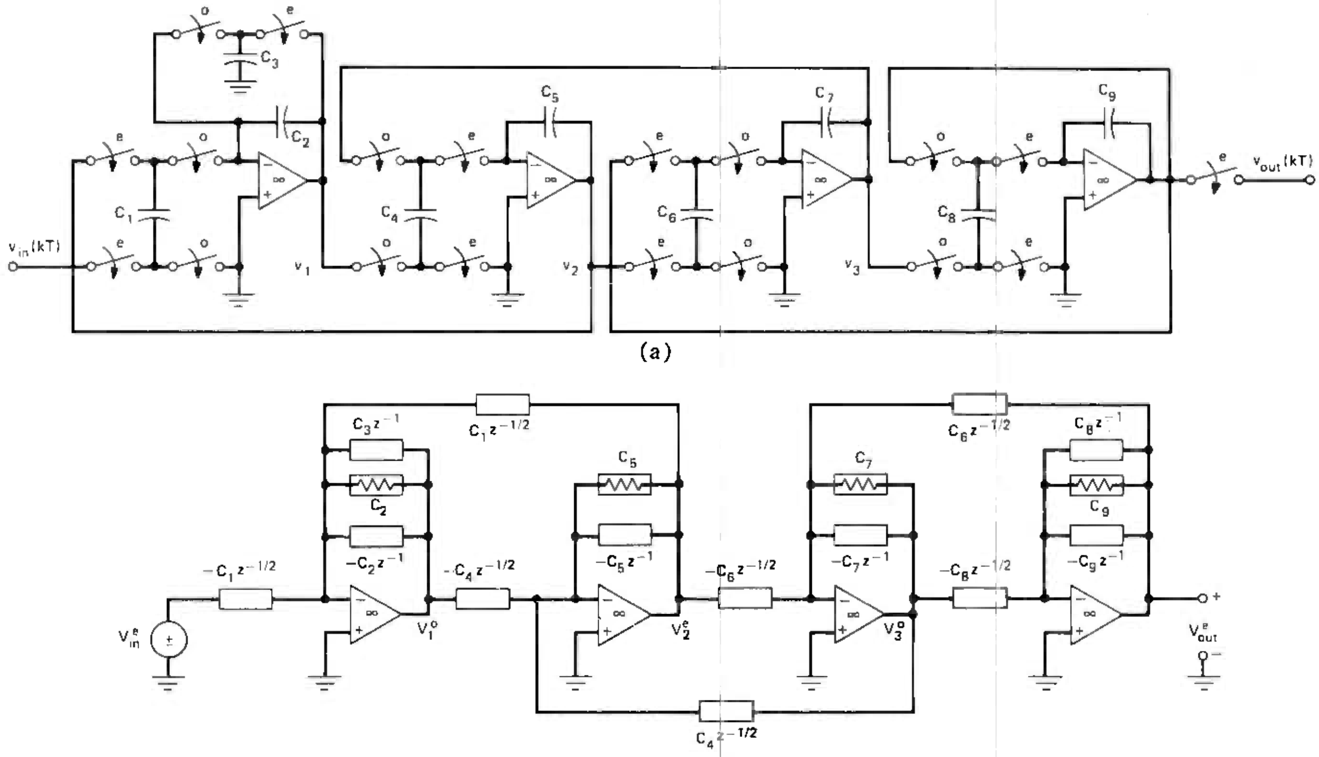
(b)

Fig. 19—A reduced version of the bilinear integrator in Fig. 18.

efficiently design the filter by relating pole-zero movements to specific circuit elements. It is interesting, in this era of high-speed computers and sophisticated analysis programs, that many of the classical networks tools still maintain their important role in filter design. The objective of this work has been to make tools[16, 17] of this kind more accessible to the designers of switched capacitor filters.

In keeping with this goal, a comprehensive library of building-block equivalent circuits has been given. This library extends that given in Ref. 17 by providing equivalent circuits for higher order SC elements and a variety of sampled data sources. These SC elements, typically comprised of one capacitor and from one to four switches, serve as circuit elements for SC networks much in the spirit in which resistors and capacitors serve analog circuits. Viewing switched capacitor elements in this way facilitates the derivation of canonic z-domain equivalent circuits for complex SC networks and the application of classical networks tools to their analysis and synthesis.

In deriving the equivalent circuits, it is pointed out that there are several interpretations and types of link two ports or LTPs. The interpretation of the floating LTP as a balanced lattice network, in lieu of an unbalanced structure, results in the elimination of two transformers in the equivalent circuit for a floating capacitor. In addition,



(a)

(b)

Fig. 20—Fourth-order, low-pass, leapfrog active-sc filter.

unbalanced inverting and differencing LTPs are also identified. An equivalent circuit is provided for a general toggle switched floating four port (TSFFP) element. The functioning ports are shown to be linked via a 90-degree rotated, balanced floating LTP. By applying the appropriate termination conditions to the TSFFP, its equivalent circuit can be used to derive the equivalent circuit for any single capacitor element in the library.

Several examples were worked out and discussed illustrating the case of application of the proposed equivalent circuits and the insight gained. Most interesting was the novel circuit depicted in Fig. 12 which exhibits a bifunctional capability; namely, a high-pass function at the even $(2kT)$ times and a low-pass function at the odd $(2k + 1)T$ times. The application to active-SC networks of various complexities was also discussed. Particularly noteworthy were the various subtle differences among the circuits shown in Figs. 17, 18, and 19, all professed to realize bilinear integration.

VI. ACKNOWLEDGMENTS

The author is grateful to J. J. Friend, T. M. Chien, and P. E. Fleischer for their helpful comments, discussions, and words of encouragement as this work evolved. I would also like to thank R. L. Adams and C. F. Kurth for carefully reading the initial manuscript and providing valuable constructive criticisms.

REFERENCES

1. R. Rao and S. Srinivasan, "Low Sensitivity Active Filter Using the Operational Amplifier Pole," Proc. IEEE (1974), pp. 1713–1714.
2 R. Schaumann, "Low Sensitivity, High Frequency, Tunable Active Filters Without External Capacitors," IEEE Trans. on Circuits and Systems, *CAS-22* (1975), pp. 39–44.
3. R. Schaumann, "On the Design of Active Filters Using only Resistors and Voltage Amplifiers," AEU-Elec. Commun. *30* (1976), pp. 245–252.
4. K. R. Laker, R. Schaumann and J. R. Brand, "Multiple Loop Feedback Active-R Filters," Proc. IEEE International Symposium on Circuits and Systems, April 1976, pp. 279–282.
5. J. R. Brand and R. Schaumann, "Temperature Compensation of Active R Filter Parameters," 1977 Midwest Symposium on Circuits and Systems, August 1977.
6. K. S. Tan and P. R. Gray, "High Order Monolithic Analog Filters Using Bipolar/IFET Technology," IEEE International Solid State Circuits Conference, February 1978, pp. 80–81.
7. D. L. Fried, "Analog Sample-data Filters," IEEE J. Solid State Circuits, *SC-7* (August 1972), pp. 302–304.
8. I. A. Young, P. R. Gray, and D. A. Hodges, "Analog NMOS Sampled-Data Recursive Filter," IEEE International Solid State Circuits Conference, February 1977, pp. 156–157.
9. B. J. Hosticka, R. W. Broderson, and P. R. Gray, "MOS Sampled Data Recursive Filters Using Switched Capacitor Integrators," IEEE J. Solid State Circuits, *SC-12*, No. 6 (December 1977), pp. 600–608.
10. J. T. Caves et al., "Sampled Analog Filtering Using Switched Capacitors as Resistor Equivalents," IEEE J. Solid State Circuits, *SC-12*, No. 6 (December 1977), pp. 592–599.
11. G. M. Jacobs, "Practical Design Considerations for MOS Switched Capacitor Ladder Filters," *Memorandum No. UCB/ERL M77/69*, University of California, Berkeley (1977).

12. Y. P. Tsividis and P. R. Gray, "An Integrated NMOS Operational Amplifier with Internal Compensation," IEEE J. Solid State Circuits, *SC-11*, No. 6 (December 1976), pp. 748–753.

13. J. L. McCreary and P. R. Gray, "All-MOS Charge Redistribution Analog-to-Digital Conversion Techniques—Part I," IEEE J. Solid State Circuits, *SC-10* (December 1975), pp. 371–379.

14. R. D. Baertsch, W. E. Engeler, H. S. Goldberg, C. M. Puckette, and J. J. Tiemann, "The Design and Operation of Practical Charge Transfer Transversal Filters," IEEE Trans. on Electron Devices, *ED-23* (February 1976), pp. 133–142.

15. C. H. Sequin, M. F. Tompsett, P. I. Sucin, D. A. Sealer, P. M. Ryan, and E. J. Zimany, "Self-Contained Charge-Coupled Split Electrode Filters Using a Novel Sensing Technique," IEEE J. Solid State Circuits, *SC-12* No. 6 (December 1977), pp. 626–632.

16. C. F. Kurth and G. S. Moschytz, "Nodal Analysis of Switched Capacitor Networks," to be published in IEEE Trans. On Circuits and Systems, February 1979.

17. C. F. Kurth and G. S. Moschytz, "Two-port Analysis of Switch-Capacitor Networks Using Four-port Equivalent Circuits," to be published in IEEE Trans. on Circuits and Systems, March 1979.

18. D. L. Fraser, Jr. and A. J. Vera, "Design Tools for Switched Capacitor Filters," CCD Signal Processing Workshop, New York, N.Y., May 1978.

19. P. E. Fleischer, "Computer Analysis of Switched Capacitor Networks in the Frequency Domain," unpublished work.

20. Y. Tsividis, "Switched Capacitor Network Analysis," CCD Signal Processing Workshop, New York, N.Y., May 1978.

21. M. L. Liou and Y. L. Kuo, "Exact Analysis of Switched Capacitor Circuits with Arbitrary Inputs," unpublished work.

22. A. V. Oppenheim and R. W. Schafer, Digital Signal Processing, Englewood Cliffs, N.J.: Prentice-Hall, 1975.

23. T. Fjällbrant, "Digital Filters with a Number of Shift Sequences in Each Pulse Repetition Interval," IEEE Trans. on Circuit Theory, *CT-17*, No. 8 (August 1970), pp. 452–455.

24. J. R. Ragazzini and G. F. Franklin, *Sampled-Data Control Systems*, Chapter 3, New York; McGraw Hill, 1958.

25. C. A. Desoer and E. S. Kuh, *Basic Circuit Theory*, New York: McGraw-Hill, 1969.

26. M. Copeland. "The Bilinear Transform Switched Operator," CCD Signal Processing Workshop, New York, N.Y., May 1978.

27. G. C. Themes and I. A. Young, "An Improved Switched Capacitor Integrator," Elec. Lett. *14*, No. 9 (April 27, 1978), pp. 287–288.

28. L. T. Bruton, "Low Sensitivity Digital Ladder Filters," IEEE Trans. on Circuits and Systems, *CAS-22*, No. 3 (March 1975), pp. 168–176.

THE BELL SYSTEM TECHNICAL JOURNAL
Vol. 58, No. 3, March 1979
Printed in U.S.A.

Determination of the Basic Device Parameters of a GaAs MESFET

By H. FUKUI

(Manuscript received July 28, 1978)

This paper describes a new technique to determine the basic properties of the active channel of a gallium arsenide (GaAs) metal-semiconductor field effect transistor (MESFET). The effective gate length, channel thickness, and carrier concentration are determined from dc parameters. A precise method of measuring the dc parameters is also given. The new techniques are demonstrated using a wide variety of sample devices. It is also shown that microwave performance parameters, such as the maximum output power and minimum noise figure, are well predicted by dc parameters. Calculated values of the intrinsic and extrinsic dc parameters, using simple analytical expressions developed in terms of the geometrical and material parameters of a device, are shown to be in excellent agreement with their measured values. These expressions can be used as a basis for device design.

I. INTRODUCTION

In a gallium arsenide (GaAs) metal-semiconductor field effect transistor (MESFET), the properties of the active channel are fundamental in describing its operation. The channel properties can be characterized by the four basic parameters: gate length, gate width, channel thickness, and channel doping.

In a recent paper,[1] the maximally obtainable value of channel current was defined as the maximum channel current, I_m. It was pointed out that I_m differs from either (fully open channel) saturation current, I_s, or zero-gate-bias drain current, which is often referred to as I_{dss}. Currents I_s and I_{dss} have conventionally been used to show upper limits of the drain current capability. However, neither I_s nor I_{dss} can represent the maximally obtainable value of channel current. It was emphasized that I_m plays an important role in determining the maximum capability of large-signal operation of the device. Simple expressions for I_m were then obtained in terms of the four basic channel parameters, as a result of an extended study of Shockley's gradual channel approximation[2] on Grebene-Ghandhi's two-section FET model[3] with Fukui's concept on the current limiting mechanism.[1]

Among the four basic channel parameters, the total gate width, Z, is usually a given factor or merely a scaling factor. Therefore, the other three parameters are noted to be the most crucial variables in the design work. For these three parameters, their effective values were adopted in Ref. 1. This was essential, especially for gate length. The effective gate length, L, may be either shorter or longer than the physical length of gate metallization, L_g, depending upon the gate junction topography. The channel thickness, a, and carrier concentration, N, represent their effective values in the active region of the channel.

In Ref. 1, practical expressions for the zero-gate-bias channel current, I_o, were also developed as functions of the basic channel parameters N, a, L, and Z, and an additional parasitic parameter of source series resistance, R_s. An approximate expression for the knee voltage, V_{kf}, corresponding to I_m in the drain I-V characteristic, was given by a combination of N, a, L, Z, R_s, and R_d on a semi-empirical basis, R_d being the drain series resistance. In addition, it has been known that the total pinch-off voltage, W_p, is determined by the Na^2 product and that W_p is equal to the sum of terminal pinch-off voltage, V_p, and Schottky-barrier built-in voltage, V_b.

It is now conceivable that N, a, and L may be determined from the measured values of I_m, I_o, V_{kf}, V_p, V_b, R_s, and R_d, provided that Z is known. The prime purpose of this paper is to present a new technique for carrying out this work. Throughout the paper, a transistor curve tracer is exclusively used as the tool necessary for measuring the dc parameters. However, test equipment of other types with the equivalent functions can, of course, be used as well.

There has been a common practice in which either a or N is determined from V_p, after knowing either N or a, respectively, and assuming an appropriate value of V_b. Also, Fair showed that an iterative analysis on I_o, R_s, and transconductance, g_m, makes it possible to determine N and a from known values of I_o, V_p, and terminal transconductance, g'_m.[4] However, as far as the author knows, there has been no published report on an evaluation technique for the effective gate length of a finished device. This paper presents such a technique as well as simultaneous determination of N and a, from known values of I_m, I_o, V_p, V_b, V_{kf}, R_s, and R_d.

The second purpose of this paper is to show prediction of the microwave performance parameters, such as the maximum output power and minimum noise figure, from the dc parameters and hence the basic channel parameters. To predict the minimum noise figure, the values of g_m and R_g, which is the gate series resistance, must be known. Therefore, the determination of g_m and R_g are also described in this paper. Once the detail of the structure outside the gate channel is given, the parasitic parameters, such as R_g, R_s, and R_d, can be analytically expressed in terms of the geometrical and material parameters of the corresponding sections of the device. The validity of such expressions is then examined with experimental results in this paper.

II. PRINCIPLE OF NEW TECHNIQUE

2.1 Analytical expressions for device dc parameters

To determine the basic channel parameters of a GaAs MESFET from the measured values of its dc parameters, expressions showing their relationships are essential. It has been well known that W_p and I_s are given by

$$W_p = \frac{qNa^2}{2\kappa\epsilon_0} \tag{1}$$

and

$$I_s = qv_sNaZ, \tag{2}$$

respectively,[5] in which q is the electronic charge, ϵ_0 is the permittivity of free space, κ is the specific dielectric constant, and v_s is the saturated velocity of electrons in n-GaAs. Substituting $q = 1.60 \times 10^{-19}$ C, $\epsilon_0 = 8.85 \times 10^{-14}$ F/cm, $\kappa = 12.5$ for GaAs, and a best fit value of $v_s = 1.4 \times 10^7$ cm/sec (Ref. 1) into (1) and (2) yields the following practical expressions:

$$W_p = 7.23Na^2 = V_p + V_b \quad \text{(V)} \tag{3}$$

and

$$I_s = 0.224ZNa \quad \text{(A)}, \tag{4}$$

where N is in units of 10^{16} cm^{-3}, a is in μm, and Z is in mm.

As mentioned earlier, analytical expressions for I_m, I_o and V_{kf} have been derived in Ref. 1. First, an expression for I_m is given in the form

$$I_m = \beta I_s, \tag{5}$$

where β is the maximum channel opening factor. Parameter β is expressed approximately as

$$\beta \approx 1 - \frac{0.18}{a}\sqrt{\frac{L}{N}} \tag{6}$$

provided that a best-fit value of 0.29×10^4 V/cm is assumed for the critical electric field, E_c. Another approximate expression for I_m is shown as

$$\frac{I_m}{Z} \approx \frac{0.18N^{1.3}a^{1.5}}{L^{0.28}} \quad \text{(A/mm)}. \tag{7}$$

Second, an expression for I_o is given in the form

$$I_o = \gamma I_s, \tag{8}$$

where

$$\gamma \approx 1 + \sigma - \sqrt{\delta + 2\sigma + \sigma^2} \tag{9}$$

$$\delta = \frac{V_b + 0.234L}{W_p} \tag{10}$$

and

$$\sigma = \frac{0.0155R_sZ}{a}. \tag{11}$$

Another approximate expression for γ is given by

$$\gamma \approx \left[1 - \frac{1}{\sqrt{W'_p}}\right]\left[1 - \frac{I_sR_s}{2\sqrt{W'_p}}\right], \tag{12}$$

where

$$W'_p = W_p + V_c. \tag{13}$$

In (13), V_c is a correction voltage that may vary from zero to a few tenths of a volt, depending upon the configuration of the channel structure. No analytical form is presently available for V_c.

Third, an expression for V_{kf} is given by

$$V_{kf} \approx (1-\beta)^2W_p + I_f(R_s + R_d) + V_c, \tag{14}$$

where I_f is the maximum value of total forward drain current, including the leakage current through the buffer layer and substrate.

2.2 Determination of basic channel parameters: Case I

If any one of the basic channel parameters N, a, and L is known, the other two can be determined from known values of V_p, V_b, and (I_m/Z) in a straightforward manner, using either a set of (3), (4), (5), and (6), or (3) and (7). For example, if the N value is known as a result of the epitaxial layer evaluation, a is readily determined by

$$a = \sqrt{\frac{V_p + V_b}{7.23N}} \quad (\mu\text{m}), \tag{15}$$

as is well known. The maximum channel opening factor is evaluated as

$$\beta = \frac{12I_m/Z}{\sqrt{N(V_p + V_b)}}. \tag{16}$$

Using this β value, L is determined by

$$L = 4.27(V_p + V_b)(1 - \beta)^2 \quad (\mu\text{m}). \tag{17}$$

In the case that L (or a) is known, similar evaluation for N and a (or N and L) can also be carried out as well.

Among dc parameters, I_m and W_p ($= V_p + V_b$) can be considered to be primary, because they are determined only by the active channel properties which are intrinsic. Other dc parameters, such as I_0 and V_{kf}, are secondary, since they are affected by extrinsic elements outside the gate channel region. The basic channel parameters can be determined only from the *primary* dc parameters, if any one of the three basic parameters for the device under evaluation is known. This is a characteristic of case I.

2.3 Determination of basic channel parameters: Case II

In this case, none of the three basic channel parameters is known. For the determination of these parameters, the *secondary* dc parameters play the major role and the primary dc parameters remain auxiliary. There are two ways to determine the basic channel parameters in this category.

The first method is based on parameter I_0. Rearranging (8) and (12) yields

$$I_s \approx \frac{\sqrt{V_p + V_b + V_c}}{R_s}\left[1 \mp \sqrt{1 - \frac{2I_0R_s}{\sqrt{V_p + V_b + V_c} - 1}}\right]. \tag{18}$$

After knowing I_s, N and a can be determined from (3) and (4), respectively, as follows:

$$N = \frac{1}{V_p + V_b}\left[\frac{12I_s}{Z}\right]^2 \quad (10^{16}\ \text{cm}^{-3}) \tag{19}$$

and

$$a = 0.031\ (V_p + V_b)\ \frac{Z}{I_s} \quad (\mu\text{m}). \tag{20}$$

Now (17) can be used with (5) to obtain L as

$$L = 4.27\ (V_p + V_b)\left[1 - \frac{I_m}{I_s}\right]^2 \quad (\mu\text{m}). \tag{21}$$

The second method is to utilize parameter V_{kf}. Rewriting (14) yields

$$\beta \approx 1 - \sqrt{\frac{V_{kf} - I_f(R_s + R_d) - V_c}{V_p + V_b}}. \tag{22}$$

After knowing the β value, N and a can be determined by (19) and (20), respectively, provided that $I_s = I_m/\beta$. The L value can be directly determined from (14) and (17) as

$$L = 4.27\,[V_{kf} - I_f(R_s + R_d) - V_c] \quad (\mu\text{m}). \tag{23}$$

The first method demands the known values of Z, V_p, V_b, V_c, I_o, I_m, and R_s, whereas the second method requires Z, V_p, V_b, V_c, V_{kf}, I_f, R_s, and R_d already known. In the computation process of determining N, a, and L, the subtraction of two major terms is included in both cases. Therefore, chances of introducing an intolerable error are inevitable. Thus, taking only a single method is not advisable. The results obtained from one method have to be checked with the other method. The simplified relationship given in (7) could be conveniently used as a guide to examination and adjustment. Expressions (8) through (11) for I_o could also be applied for an additional checking of the determined values of N, a, and L, in comparison with the directly measured value of I_o. Some adjustments on temporarily determined values of the parameters are often necessary to reach their most probable values.

III. MEASUREMENTS OF DC PARAMETERS

As previously mentioned, a transistor curve tracer is used as the test instrument in this paper. A good calibration of the measuring system is essential. Not only the curve tracer but also the test fixture must be taken into consideration. For example, lead resistances may introduce an intolerable measuring error in large-size devices. An excessive leakage current may mislead the determination of junction parameters. Instability and/or relatively low-frequency oscillation, often taking place in a high-performance device, are an annoying phenomena and require a special skill to suppress.

In the following sections, measuring methods for the dc parameters of a GaAs MESFET are described. Although some methods have been known or are easily derived from known methods, they are included with brief descriptions for completeness. The ideality parameter of a gate junction, n, and the open channel resistance, R_o, are needed neither for determination of the basic channel parameters nor for prediction of the microwave performance parameters. Nevertheless, they are secondarily obtained in the course of determination of the primary parameters. As they may be relevant to a further study of the device, their determination is also described.

3.1 Determination of gate barrier built-in voltage and ideality parameter

As is well known,[6] the forward current density, J, of a Schottky barrier junction for $V > 3kT/q$ is approximately written as

$$J = A^*T^2 \exp\left[-\frac{qV_b}{kT}\right]\exp\left[\frac{qV}{nkT}\right], \tag{24}$$

where A^* is the effective Richardson constant, T is the junction temperature in °K, k is the Boltzmann constant, n is the ideality parameter, and V is the forward bias voltage.

The extrapolated value of current density to zero bias gives the saturation current density, J_s. The barrier built-in voltage is then obtained from

$$V_b = \frac{kT}{q} \ln\left[\frac{A^* T^2}{J_s}\right]. \tag{25}$$

The ideality parameter is given by

$$n = \frac{q}{kT} \frac{\partial V}{\partial(\ln J)}. \tag{26}$$

Figure 1 is a multi-exposed photograph of the forward *I-V* characteristic of a gate junction at room temperature. As described in the caption, each curve corresponds to a current range in a decimal step for several orders of magnitude. In the highest current range, the gate was against three different connections, i.e., source and drain combined, source alone, and drain alone, in order to differentiate R_g, R_s, and R_d from each other later.

The *I-V* characteristic given in Fig. 1 is plotted as shown in Fig. 2. At high values of the gate bias, V_g, the gate current, I_g, tends to saturate due to the series resistance effect. At low values of V_g, I_g is often disturbed by a leakage current component around the gate

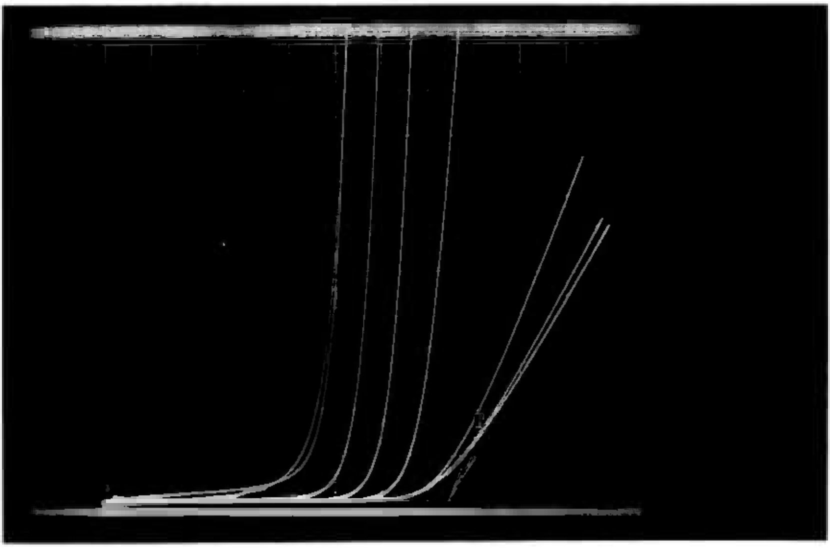


Fig. 1—A multi-exposed photograph of the forward *I-V* characteristic of a gate junction at room temperature. Each curve corresponds to a current range of decimal step. In the highest current range, three different ground connections are taken against the gate. They are the source and drain connected together, source alone, and drain alone. For all lower current ranges, the drain is connected to the source. The horizontal scale is in units of 0.1 V/div. The vertical scales are, left to right, 1 μA/div, 10 μA/div, 100 μA/div, 1 mA/div, 10 mA/div (source and drain together), 10 mA/div (source only), and 10 mA/div (drain only).

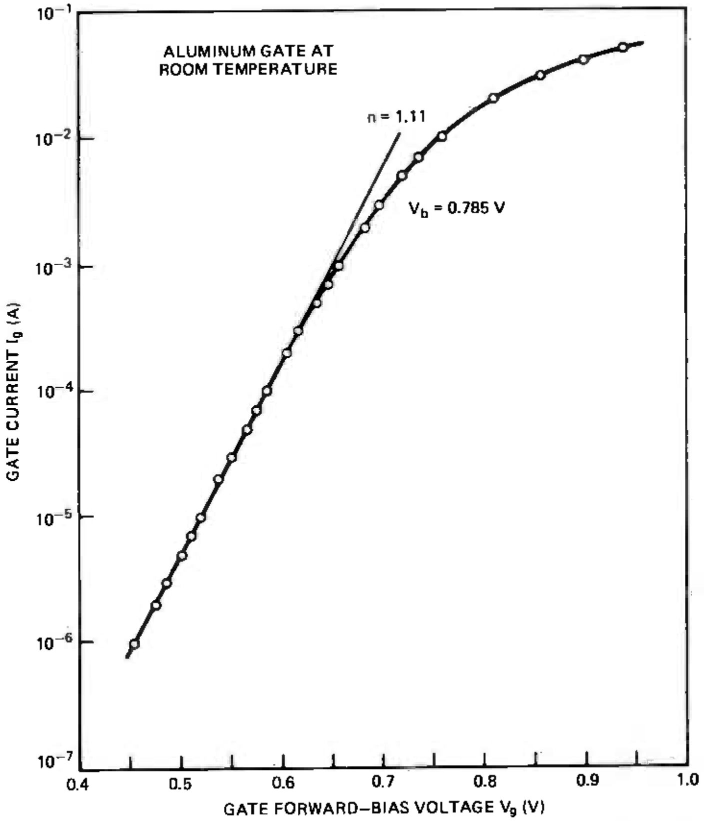


Fig. 2—The forward *I-V* characteristic of an aluminum gate diode at room temperature. The slope and the location of its linear portion on a semi-log paper give the ideality parameter, *n*, and the gate built-in voltage, V_b, respectively.

periphery or with the package. In the middle range where the log I_g vs V_g characteristic is linear, two gate biases, $V_{g(m)}$ and $V_{g(m-1)}$ in V, can be chosen corresponding to $I_g = 10^m$ and $I_g = 10^{m-1}$ in A, respectively. Usually, m takes a negative value.

If the effective mass of electrons in *n*-GaAs were taken into account, the effective Richardson constant would be 8.7 A/cm²/°K at room temperature.[7] Expression (25) for V_b can then be reduced to the following practical form at room temperature:

$$V_b = 0.768 - 0.06 \log J_s \quad \text{(V)}, \tag{27}$$

in which

$$J_s = \frac{10^y}{L_g Z} \quad (10^{-7}\ \text{A/cm}^2) \tag{28}$$

$$y = 12 + m - \frac{1}{1 - \dfrac{V_{g(m-1)}}{V_{g(m)}}} \qquad (29)$$

and L_g is in units of μm and Z is in mm. A formula to be used for deriving n is deduced from (26) as

$$n = 16.8[V_{g(m)} - V_{g(m-1)}]. \qquad (30)$$

3.2 Determination of pinch-off voltage, active channel resistance, and parasitic series resistances

Figure 3 shows the drain *I-V* characteristics in the so-called linear region. The characteristics were taken with the lowest scale of V_{ds} on the curve tracer, in which V_{ds} was the drain-source bias voltage. Each characteristic corresponds to a gate-source bias voltage, V_{gs}. The drain current, I_d, at $V_{ds} = 0.05$ V is then plotted as a function of V_{gs} as shown in Fig. 4. The terminal pinch-off voltage, V_p, can *temporarily* be determined by an extrapolation of the plot to the abscissa.

The current shown in Fig. 4 can now be converted into the resistance value as $R_{ds} = I_d/V_{ds}$. Such a resistance for $V_{ds} = 0.05$ V is shown in Fig. 5 as a function of V_p, V_b, and V_{gs}, in the same manner as used in Ref. 8, as compounded in parameter X defined as

$$X = \frac{1}{1 - \sqrt{\dfrac{V_b - V_{gs}}{V_b + V_p}}}. \qquad (31)$$

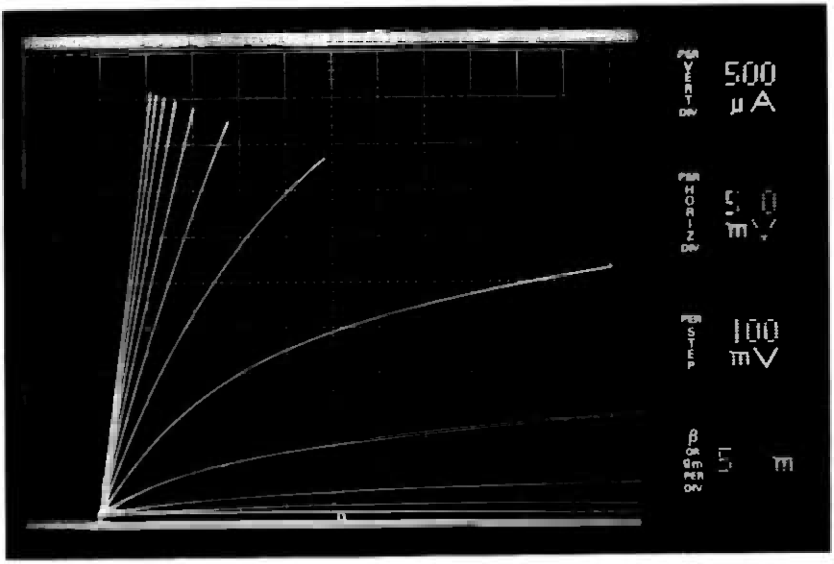


Fig. 3—An expanded view of the drain *I-V* characteristics in the so-called linear region near the origin on gate-bias offset mode (+0.4 V in this case).

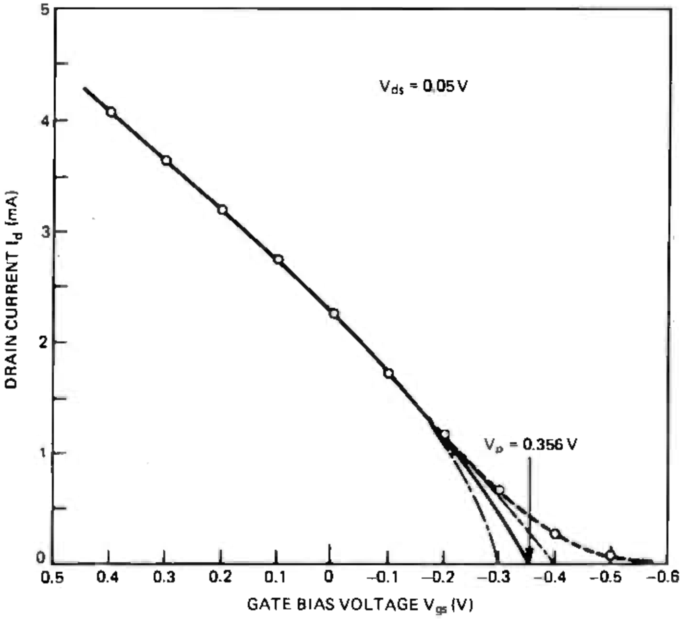


Fig. 4—Drain current as a function of gate bias voltage in both forward and reverse directions at a drain-source bias voltage, V_{ds}, of 0.05 V. An appropriate extrapolation of this curve to the abscissa gives the external (or terminal) pinch-off voltage, V_p. A misjudgment in the determination of V_p, as shown by the dash-dotted line, will cause a problem in the next step.

This plot should be a straight line. However, the plot may deviate from the line either upward or downward as X increases, as indicated by dashed lines in Fig. 5. Such a deviation depends upon the *temporarily* determined value of V_p. If the plot significantly departed from the straight line, the previously determined V_p had to be re-examined. Usually, a slight adjustment on the V_p value easily solves this problem and V_p is *finally* determined.

Such a way of determining V_p seems to be much more complicated and tedious than the conventional method. Usually, V_p is simply estimated from V_{gs} corresponding to the bottom line of the drain *I-V* characteristics. Indeed, the conventional method may not bring too much error in the determination of V_p in the case of thick active channels, i.e., devices with high pinch-off voltages. However, in devices with thin active channels, especially on buffer layers, the error may reach as high as 100 percent with the conventional method, as will be seen in Fig. 6. Therefore, the present method has been developed.

In Fig. 5, the linear extrapolation of the plot to the ordinate gives the value of $(R_s + R_d)$. The slope of the line is designated as R_o. The parameter R_oX represents the effective value of active channel resistance at a given V_{gs}.

As was mentioned previously, the forward gate current value is affected by a combination of series resistances at high current levels. Therefore, the slope of the I_g-V_g characteristic, measured at a current density of around 10^4 A/cm^2, gives an estimate of series resistance values. By measuring the gate current in three different ground connections, i.e., source and drain combined, source only, and drain only, three resistance values can be obtained. The differences between the last two values yields $(R_s - R_d)$. Since $(R_s + R_d)$ has been known, R_s and R_d are now readily separated. The gate series resistance R_g can be

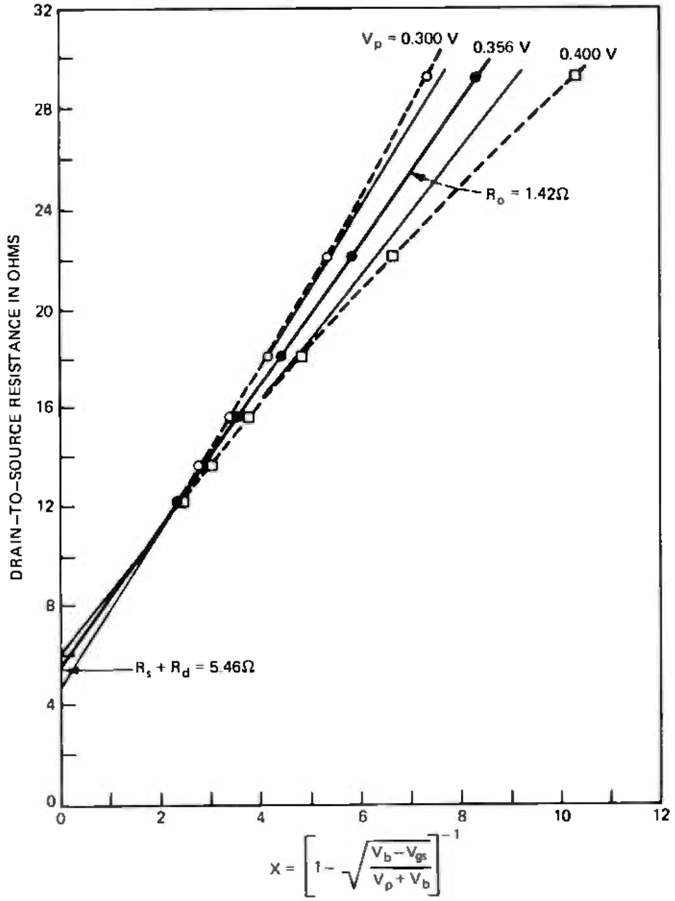


Fig. 5—Determination of the open channel resistance, R_o, and parasitic series resistances, R_s and R_d. The proper selection of V_p value in Fig. 4 (i.e., V_p = 0.356 V in this case) is essential for this determination, as illustrated in two wrong cases (V_p = 0.300 V and 0.400 V).

deduced from the first resistance value by subtracting the contribution of the paralleled R_s and R_d from the resultant.

3.3 Determination of specific voltages and currents

Figure 6 is a typical photograph of the drain I-V characteristics for negative gate potentials, taken with the nonoffset gate-bias mode of the curve tracer. In contrast with Fig. 6, Fig. 7 is an unconventional photograph of the drain characteristics of the same device when driven in the forward gate bias with the offset mode. As the positive value of gate offset is increased, the drain current increases. Beyond a certain value of the offset, V_f, however, the drain current no longer increases. Figure 7 shows such a state of offsetting.

The knee voltage of a drain I-V characteristic could be defined as the intercepting point between the extensions of two linear regions of the characteristic. The knee voltage of the I-V curve for V_f is denoted by V_{kf} as shown in Fig. 7. Also, the knee voltage for the zero-gate-bias curve is by V_{ko} as shown in Fig. 6. Total drain currents, I_f and I_{do}, for V_f and null gate bias, respectively, are measured at the corresponding knee voltages, V_{kf} and V_{ko}. Leakage current components, I_{pf} and I_{po}, are also measured at V_{kf} and V_{ko}, respectively, both for $V_{gs} = -V_p$. The maximum channel current, I_m, and zero-gate-bias channel current, I_o, are then evaluated as $I_f - I_{pf}$ and $I_{do} - I_{po}$, respectively.

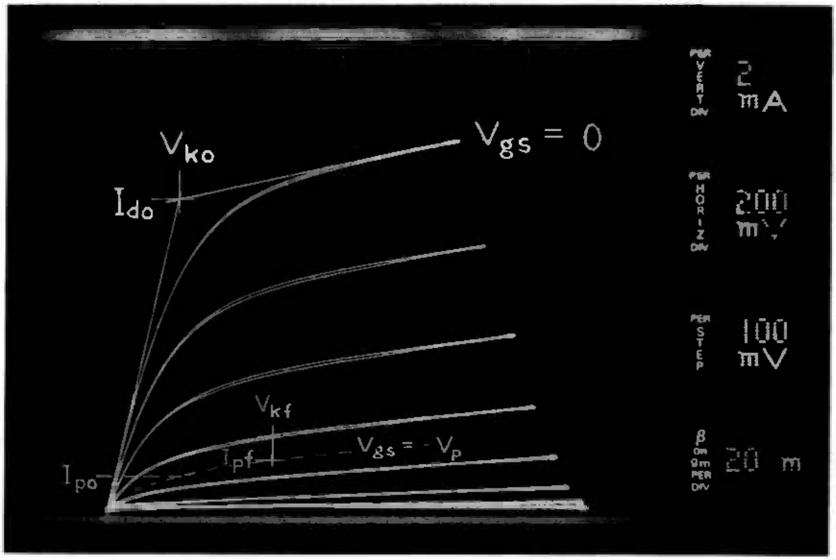


Fig. 6—A conventional drain I-V characteristics with nonoffset gate-bias mode as usual.

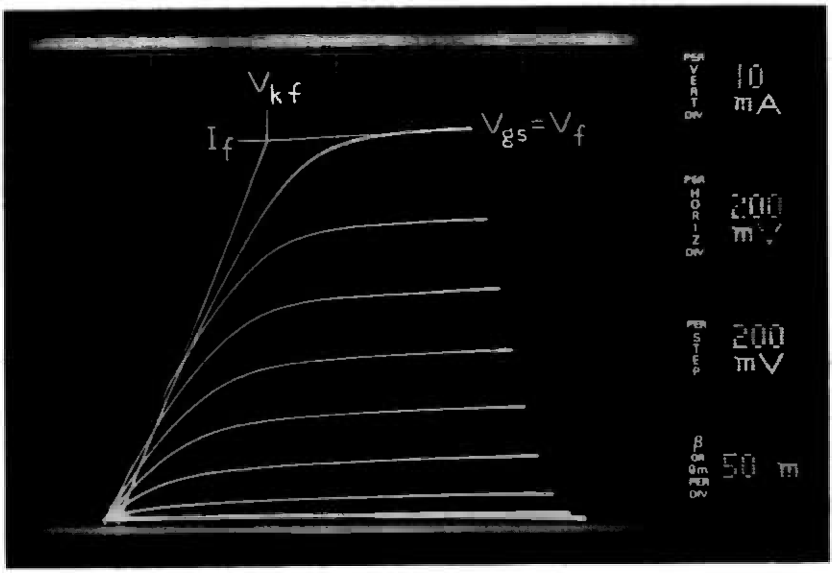


Fig. 7—A special drain *I-V* characteristics with a forward gate-bias offsetting of V_f. This is a critical value beyond which no increase in the drain current is observed.

3.4 Determination of transconductance

As is well known, the magnitude of the transconductance of a good device can be assumed to remain constant up to nearly the cutoff frequency. Therefore, the so-called dc transconductance can be considered a first-order approximation of the amplitude of microwave transconductance.

The following method of measuring g'_m and evaluating g_m is conventional. In the drain *I-V* characteristics, an increment of drain current, ΔI_d, between two adjacent curves for $V_{gs} = V_1$ and $V_{gs} = V_2$ at a given V_{ds} yields an average transconductance as

$$g'_m = \left|\frac{\Delta I_d}{V_2 - V_1}\right|. \tag{32}$$

However, this value is a result of degradation due to R_s. The intrinsic value of the small-signal transconductance at the bias points, $V_{ds} = V_{ds}$ and $V_{gs} = (V_1 + V_2)/2$, can thus be derived as

$$g_m \approx \frac{g'_m}{1 - g'_m R_s}. \tag{33}$$

IV. EXAMINATIONS OF NEW TECHNIQUE

4.1 Sample devices

To present the practical values of measured dc parameters and hence to demonstrate the new technique for determining the basic channel parameters, 11 GaAs MESFETs of various designs were chosen, as shown in Table I. The first five devices (A through E) were originally designed for high-power use[9] and the others (F through K) were for low-noise applications.[10]

All devices had a total device width of 0.5 mm, except for device E which had 1 mm. The distance between the source and drain electrodes was nominally 6 μm for devices A through E and 3 μm for devices F through K. The nominal gate length was 2.0 to 2.5 μm for devices A to D, 1.0 to 1.5 μm for device E, and 0.8 μm for devices F to K. The physical length of the gate electrode, however, was not necessarily equal to the effective gate length because the latter was subject to the shape of a gate junction.

All the sample devices had a multi-layer structure consisting of an undoped n-GaAs film 2 to 3 μm thick as the buffer, an n-GaAs channel, and n^+-GaAs layer, except for devices J and K. All the layers were grown sequentially on a semi-insulating GaAs substrate in an $AsCl_3$/Ga/H_2 CVD system.[11] After removing the n^+-GaAs layer and part of n-GaAs in the gate region, aluminum approximately 0.7 μm thick was deposited as the Schottky-gate metal. The ohmic contacts were formed with a 12 percent Ge/Au-Ag-Au system approximately 0.25 μm thick, alloyed at nearly 500°C. The final metallization was completed with a Ti-Pt-Au system 0.9 to 1.4 μm thick.

4.2 Results of measurements

First, dc parameters n, V_b, V_p, W_p, V_{kf}, V_f, I_f, I_{pf}, I_m, V_{ko}, I_{do}, I_{po}, I_o, R_o, R_s, R_d, R_g and g_m' were obtained in accordance with the measuring technique described in Section III. The measured values of these parameters are shown in Table I. Note that g_m' is an average value taken at approximately I_o.

Second, basic channel parameters L and N were deduced from each of the two methods described in Section 2.3. In the application of (18), (22), and (23), V_c was assumed to be zero for devices A to E and to be a single value of 0.17 V for devices F to K. The two deduced values for each of L and N were then averaged to obtain the most probable value. Using this mean value of N in (15), the most probable value of a was determined. All values mentioned here are shown in Table II.

4.3 Comparison of calculated and measured results

By inserting the determined values of N, a, and L into (4), (6), and (5), I_m was calculated. The calculated value of I_m was then compared with the measured value as shown in Table III. By adopting the measured value of V_b and R_s in (10) and (11), respectively, I_o was also calculated using (9), (4), and (8). The calculated I_o was compared with the measured value, again as shown in Table III. The comparison has shown excellent agreement between the calculated and measured

Table I—Measured values of dc parameters

Parameter		Device										
Symbol	Units	A	B	C	D	E	F	G	H	I	J	K
Z	mm	0.5	0.5	0.5	0.5	1.0	0.5	0.5	0.5	0.5	0.5	0.5
z	mm	0.5	0.25	0.25	0.5	0.25	0.25	0.25	0.25	0.25	0.25	0.25
n		1.09	1.07	1.26	1.34	1.26	1.46	1.12	1.23	1.23	1.34	1.26
V_b	V	0.72	0.73	0.76	0.79	0.74	0.74	0.78	0.76	0.74	0.76	0.76
V_p	V	5.84	6.07	3.51	1.16	4.88	1.15	0.51	0.34	0.54	0.52	0.36
W_p	V	6.56	6.80	4.27	1.95	5.62	1.89	1.29	1.10	1.28	1.28	1.12
V_{kf}	V	1.95	1.95	2.10	1.55	1.85	0.75	0.59	0.68	0.77	1.08	0.83
V_f	V	1.49	1.34	1.80	1.49	1.62	1.39	0.97	0.96	1.12	1.30	1.14
I_f	mA	176	196	179	136	400	83.0	86.0	82.0	107	94.0	87.5
I_{pf}	mA	1	1	2	2	5	4.0	1.5	1.5	4	3.0	3.0
I_m	mA	175	195	177	134	395	79.0	84.5	80.5	103	91.0	84.5
V_{ko}	V	1.5	1.5	1.4	0.7	1.4	0.55	0.3	0.3	0.4	0.3	0.3
I_{do}	mA	122	134	104	39.5	257	32.0	20	13	24.0	19.5	14.5
I_{po}	mA	0	1	1	1.5	3	2.7	1	1	2.5	2	2
I_o	mA	122	133	103	38.0	254	29.3	19	12	21.5	17.5	12.5
R_o	Ω	4.1	3.0	2.9	3.4	1.2	4.0	2.9	2.9	2.5	2.7	3.0
R_s	Ω	4.8	3.7	4.5	5.2	1.8	2.7	1.5	2.3	2.3	3.8	2.9
R_d	Ω	2.4	3.6	4.0	2.8	1.7	2.7	1.9	2.4	2.3	4.8	3.4
R_g	Ω	3.8	1.7	1.7	4.5	4.4	13.7	3.8	3.7	4.0	4.5	3.8
g'_m	m℧	28	32	28	53	72	30	45	51	52	45	48

Table II—Determination of the most probable values of active channel carrier concentration and thickness and effective gate length

Parameter			Device										
Equations Used	Symbol	Units	A	B	C	D	E	F	G	H	I	J	K
	V_c	V	0	0	0	0	0	0.17	0.17	0.17	0.17	0.17	0.17
(18, 19)	N	10^{16} cm^{-3}	6.01	5.98	11.19	21.16	7.96	3.52	6.34	7.75	9.84	7.59	7.91
(18, 21)	L	μm	3.07	2.06	2.71	2.08	2.04	0.568	0.466	0.537	0.502	0.491	0.486
(21, 5, 19)	N	10^{16} cm^{-3}	5.86	6.15	10.58	20.13	7.78	3.51	6.78	7.76	9.83	7.23	8.31
(23)	L	μm	2.91	2.22	2.47	1.97	1.92	0.563	0.545	0.532	0.503	0.436	0.538
(averaged)	N	10^{16} cm^{-3}	5.94	6.07	10.89	20.65	7.87	3.52	6.56	7.76	9.84	7.41	8.11
(averaged)	L	μm	2.99	2.14	2.59	2.03	1.98	0.566	0.508	0.534	0.502	0.463	0.512
(15)	a	μm	0.391	0.394	0.233	0.114	0.314	0.273	0.165	0.140	0.134	0.155	0.138

Table III—Calculated values of the maximum channel current and zero-gate-bias channel current and their comparison with measured values

Parameter			Device										
Equations used	Symbol	Units	A	B	C	D	E	F	G	H	I	J	K
(4, 5, 6)	I_m	mA	175	195	177	134	395	79.0	84.4	80.6	103	91.0	84.4
(4, 8, 9, 10, 11)	I_o	mA	118	133	99.0	36.2	272	31.0	18.5	11.0	23.1	18.4	12.1
I_m (cal)/I_m (meas)			1.000	0.999	1.000	1.000	1.000	1.000	0.999	1.001	1.000	1.000	0.999
I_o (cal)/I_o (meas)			0.965	1.002	0.971	0.953	1.069	1.058	0.973	0.916	1.076	1.054	0.966

values in both cases. Note that the equations applied to calculate I_m and I_o had not been used to determine L, a, and N in Section 4.2.

The 11 devices used in this experiment were from 11 different slices. The free carrier concentration of each slice was evaluated by a doping profiler.[12] It has been recognized that this particular profiler gives an N-value 5 to 40 percent higher than the true value of N.[13] Also, a standard deviation of ± 3 percent in the doping across a wafer has been known for these slices.[11] Under such circumstances, the evaluated value of N for each device was compared with an uncorrected, representative N value obtained for the corresponding slice. The results shown in Table IV seem to be very reasonable. A consistent pattern of difference between the two N values is seen there, as expected from the above observation.

V. ANALYSES OF EXPERIMENTAL RESULTS

5.1 Schottky-barrier built-in voltage

The built-in voltage of an aluminum Schottky-barrier gate junction at room temperature has been expressed in an analytical form as a function of N in n-GaAs[1] as follows:

$$V_b = 0.706 + 0.06 \log N \quad \text{(V)}, \tag{34}$$

where N is in units of 10^{16} cm^{-3}. This expression is compared with the measured value of V_b for all devices used, as shown in Fig. 8. This comparison indicates that (34) can be used for aluminum gates on n-GaAs at room temperature.

5.2 Transconductance

The small-signal transconductance of a GaAs MESFET has been described by the theoretical expression[5]

$$g_m = \frac{I_s}{2W_p\left[1 - \dfrac{I_d}{I_s}\right]}, \tag{35}$$

where I_d is the dc drain bias current. Since g'_m was measured approximately at I_o, the theoretical value of g_m was also calculated for $I_d = I_o$ in (35). This calculated value was then compared with the measured value obtained using (33). This comparison led to a conclusion that the measured value of g_m would agree well with the predicted value if the latter were calculated by

$$g_m \approx \frac{0.9\, I_s}{2(V_p + V_b)\left[1 - \dfrac{I_c}{I_s}\right]}, \tag{36}$$

Table IV—Comparison of the most probable value of free carrier concentration in the active channel of an individual device with an uncorrected value of doping in the epitaxial layer of the corresponding slice

Parameter		Device										
Symbol	Units	A	B	C	D	E	F	G	H	I	J	K
N (slice) uncorrected	10^{16} cm^{-3}	6.7	7.4	14.2	24.6	8.8	4.1	7.1	9.0	10.4	8.1	11.2
N (slice)/N(device) uncor. most prob.		1.13	1.22	1.30	1.20	1.12	1.16	1.08	1.16	1.06	1.09	1.38

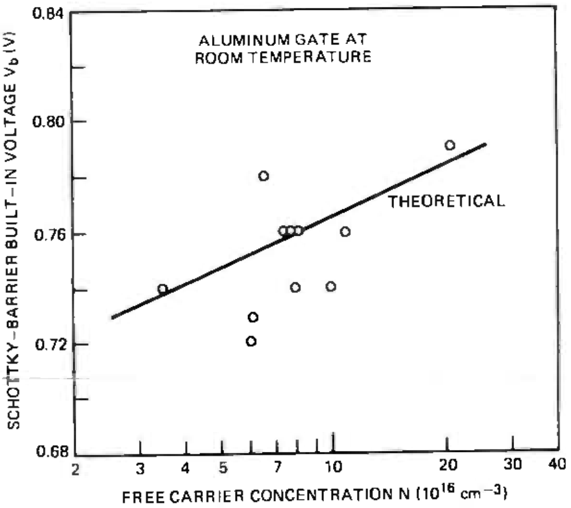


Fig. 8—Comparison of the measured values of gate built-in voltage for aluminum Schottky-barrier gates at room temperature, with a theoretical expression as a function of free carrier concentration in n-GaAs.

where I_c was the channel current, as shown in Table V.

5.3 Ohmic contacts and series channel resistance

The parasitic series resistance, R_s or R_d, consists of the ohmic contact resistance, R_{co}, and series channel resistance, R_{ch}, between the two concerned electrodes. These component resistances are now separately expressed. The expression for R_{co} given by Berger[14] and Murrmann and Widmann[15] can be simplified, as already shown by Macksey and Adams[16] as

$$R_{co} = \frac{1}{Z}\sqrt{\frac{R_c\rho_1}{a_1}}\coth\sqrt{\frac{\rho_1 L_c^2}{R_c a_1}} \approx \frac{1}{Z}\sqrt{\frac{R_c\rho_1}{a_1}}. \tag{37}$$

The series channel resistance can be further divided into two components which represent two individual parts of different structures in the space between the gate and one ohmic contacts.
Thus,

$$R_{ch} \approx R_2 + R_3, \tag{38}$$

where

$$R_2 = \frac{\rho_2 L_2}{Z a_2} \tag{39}$$

and

$$R_3 = \frac{\rho_3 L_3}{Z a_3}. \tag{40}$$

Table V—Measured and predicted values of transconductance at zero-gate-bias channel current

Parameter	Device										
g_m @ I_o	A	B	C	D	E	F	G	H	I	J	K
Measured (m℧)	32	36	46	69	83	33	48	58	59	54	56
Predicted (m℧)	33.6	35.2	47.4	71.2	81,9	35.1	50.0	55.4	60.9	52.1	56.2

In the above expressions, $a_{i=1,2,3}$, $\rho_{i=1,2,3}$ and $L_{i=2,3}$ are, respectively, the thickness, specific resistivity, and length of the GaAs epitaxial film at the corresponding place $i = 1$, 2 or 3, L_c is the length of the contacting metal electrode, and R_c is the specific contact resistance.

Parameters ρ and R_c are both functions of N. Using the experimental data taken by Matino on epitaxial n-GaAs films,[17] the doping dependence of ρ can be written in an analytical form

$$\rho \approx 0.11\ N^{-0.82} \quad (\Omega\text{-cm}) \tag{41}$$

in an N range of 10^{-1} to $10^3 \times 10^{16}$ cm^{-3}. Based on the so-called Shockley method,[8] R_c was statistically investigated using monitor areas provided within the same slices as those fabricated for either high-power[9] or low-noise use.[10] An empirical expression for R_c was then found to be

$$R_c \approx 4\ N^{-0.5} \quad (10^{-5}\ \Omega\text{-cm}^2) \tag{42}$$

for N values in the range of 3 to $10^3 \times 10^{16}$ cm^{-3}. This expression differs from that given by Heime et al.,[18] which is

$$R_c \approx 8\ N^{-1} \quad (10^{-5}\ \Omega\text{-cm}^2). \tag{43}$$

However, both expressions give close values of R_c for N in the vicinity of the mid-10^{16} cm^{-3} range. As the N value increases, the difference in R_c between the two expressions becomes recognizable. This would give rise to the case of ohmic contacts formed on n$^+$-GaAs layers. An estimate by (43) would result in too-optimistic prediction of R_c.

Substitution of (41) and (42) into (37), (39), and (40) yields practical expressions for R_{co}, R_2, and R_3 as follows:

$$R_{co} \approx \frac{2.1}{Za_1^{0.5}N_1^{0.66}} \quad (\Omega) \tag{44}$$

$$R_2 \approx \frac{1.1\ L_2}{Za_2N_2^{0.82}} \quad (\Omega) \tag{45}$$

and

$$R_3 \approx \frac{1.1\ L_3}{Za_3N_3^{0.82}} \quad (\Omega). \tag{46}$$

For the 11 sample devices with reasonable assumptions on N_1, a_1, L_2, N_2, a_2, L_3,, N_3, and a_3, component resistances R_{co}, R_2, and R_3 were calculated using (44), (45), and (46), respectively. The predicted value of R_s (or R_d) was thus obtained as the sum of these resistances. The average of the measured values of R_s and R_d for each device was then compared with the predicted value, as shown in Table VI. They were in good agreement.

5.4 Gate series resistance

Since the input signal applied to the feeding end of the gate travels along the gate metallization to the other end, the gate must be considered as a distributed network. The effective value of the gate metallization resistance in a lumped equivalent circuit is, therefore, different from the dc value measured from one end to the other. As theoretically analyzed by Wolf,[19] this effective value, R_g, is one-third of the end-to-end dc resistance as a first-order approximation. Thus,

$$R_g \approx \frac{\rho_g z^2}{3L_g hZ}, \tag{47}$$

where ρ_g is the specific resistivity, L_g is the mean length, h is the mean height, and z is the unit width of the gate metallization.

By substituting the nominal values of Z, z, L_g and h into (47) in conjunction with the measured value of R_g, ρ_g was evaluated as shown in Table VII. The range from 3.8 to 5.7 × 10^{-6} Ω-cm gives a mean value of 5.0 × 10^{-6} Ω-cm. Although this value is higher than a bulk aluminum resistivity of 2.8 × 10^{-6} Ω-cm, it seems to be very reasonable for such a fine, thin, and scaly structure of gate metallization. Based on this finding, a practical expression for R_g can be given by

$$R_g \approx \frac{17z^2}{L_g hZ} \quad (\Omega), \tag{48}$$

where Z and z are in units of mm and L_g and h are in μm.

It may be noted that annealing of a device sometimes results in an improvement in the effective value of ρ_g. The above measured values were obtained without additional heat treatment.

5.5 Open channel resistance

The open channel resistance mentioned in Section III can be expressed as

$$R_o = \frac{L}{q\mu_0 NaZ}, \tag{49}$$

where μ_0 is the low-field mobility of electrons.[8] Thus, μ_0 of an individual device can be obtained from the measured value of R_o in conjunction with Z, L, a, and N, which have already been determined.

On the other hand, μ_0 is related to ρ in the form

$$\mu_0 = \frac{1}{qN\rho}. \tag{50}$$

Table VI—Predicted values of parasitic source (or drain) series resistance and its component resistances in comparison with measured value

Parameter		Device										
Symbol	Units	A	B	C	D	E	F	G	H	I	J	K
Z	mm	0.5	0.5	0.5	0.5	1.0	0.5	0.5	0.5	0.5	0.5	0.5
N_1	10^{16} cm^{-3}	100	100	50	100	100	100	100	100	100	7.4	8.1
a_1	μm	0.15	0.15	0.15	0.15	0.15	0.15	0.15	0.15	0.15	0.35	0.45
L_2	μm	2.1	2.1	2.1	2.1	2.5	0.85	0.75	0.85	0.85	0.75	0.75
N_2	10^{16} cm^{-3}	5.9	6.1	10.9	20.7	7.9	100	100	100	100	7.4	8.1
a_2	μm	0.35	0.35	0.2	0.1	0.35	0.1	0.1	0.1	0.1	0.25	0.35
L_3	μm	0	0	0	0	0	0.4	0.3	0.4	0.4	0.3	0.3
N_3	10^{16} cm^{-3}	—	—	—	—	—	3.5	6.6	7.8	9.8	7.4	8.1
a_3	μm	—	—	—	—	—	0.273	0.165	0.140	0.134	0.155	0.138
R_{co}	Ω	0.52	0.52	0.82	0.52	0.26	0.52	0.52	0.52	0.52	1.89	1.57
R_2	Ω	3.07	3.01	3.26	3.86	1.45	0.68	0.59	0.68	0.68	1.28	0.85
R_3	Ω	0	0	0	0	0	1.15	0.86	1.17	1.00	1.13	0.86
Predicted R_s, R_d	Ω	3.59	3.53	4.08	4.38	1.71	2.35	1.97	2.37	2.20	4.30	3.28
Measured $(R_s + R_d)/2$	Ω	3.6	3.65	4.25	4.5	1.75	2.7	1.7	2.35	2.3	4.3	3.15

Table VII—Determination of the effective resistivity for aluminum gate metallization

Parameter		Device										
Symbol	Units	A	B	C	D	E	F	G	H	I	J	K
Z	mm	0.5	0.5	0.5	0.5	1.0	0.5	0.5	0.5	0.5	0.5	0.5
z	mm	0.5	0.25	0.25	0.5	0.25	0.25	0.25	0.25	0.25	0.25	0.25
L_g	μm	2.5	2.0	2.0	2.0	1.2	0.8	0.8	0.8	0.8	0.8	0.8
h	μm	0.7	0.7	0.7	0.7	0.2	0.2	0.65	0.65	0.65	0.65	0.65
R_g	Ω	3.8	1.7	1.7	4.5	4.4	13.7	3.8	3.7	4.0	4.5	3.8
ρ_g	10^{-6} Ω-cm	4.0	5.7	5.7	3.8	5.1	5.3	4.9	4.8	5.0	5.6	4.9

Substituting (41) into (50) yields μ_0 as a function of N

$$\mu_0 \approx 5.7\, N^{-0.18} \quad (10^3 \mathrm{cm}^2/\mathrm{V\text{-}sec}). \tag{51}$$

The μ_0 value deduced from the measured value of R_o and that predicted by (51) are shown in Table VIII for all the sample devices. The evaluation of the latter value was based on the average value of N shown in Table II. In the high-power devices, these two values of μ_0 were close enough to confirm Matino's results[17] and to support the use of (41) and (51). In the low-noise devices, however, the μ_0 value deduced from R_o appeared to be approximately one-half the predicted value by (51). Such a discrepancy might be caused by a special two-dimensional gate-recess structure of these devices, and/or be due to an increased influence of the transition layer as the active layer was thinned, this transition layer being between the active and buffer layers. This is subject to further investigation.

As shown in Table VIII, the μ_0 value deduced from the measured value of R_o using (49) differs from one device to the other in some degree. In accordance with the two-piece approximation of the v-E characteristic, however, μ_0 is assumed to be constant by definition and is equal to v_s/E_c. Substituting the aforementioned best-fit values of v_s and E_c into this yields a fixed value of 4.8×10^3 cm^2/V-sec for μ_0, which is independent of N. This μ_0 value is much greater than that deduced from R_o by (49) and that calculated by (51) as a function of N in the normal range. On the contrary, the fixed values of v_s and E_c have been satisfactory to evaluate I_m for a wide range of N, as seen in the previous sections. Such a conflicting situation must be reconciled. This is also subject to further study. Nevertheless, the following can be applied in practice, for the time being. The fixed values of v_s and E_c are appropriate for the evaluation of I_m for all devices. The μ_0 value provided by (51) is adequate in (49) to calculate R_o for devices with plane gates. However, a suitable correction factor is necessary for μ_0 given by (51) to make μ_0 effective for nonplane gate devices. For example, this factor was 0.5 for the low-noise devices used as samples.

Table VIII—Comparison between the low-field electron mobility value deduced from the measured value of open channel resistance by (49) and that predicted from the most probable value of free-carrier concentration in the active channel by (51)

Parameter	Device										
μ_0 in 10^3 cm^2/V-sec	A	B	C	D	E	F	G	H	I	J	K
Calculated by (49)	3.9	3.7	4.4	3.2	4.2	1.8	2.0	2.1	1.9	1.9	1.9
Calculated by (51)	4.1	4.1	3.7	3.3	3.9	4.5	4.1	3.9	3.8	4.0	3.9

VI. PREDICTION OF MICROWAVE PERFORMANCE

6.1 Maximum output power

As was previously mentioned, devices A through E were originally designed for high-power use. The maximum available output power, $P_{\max}$, of these devices were measured at 4 GHz in a coaxial system. A double-slug tuner was provided in each of the input and output circuits to obtain the conjugate match. The data taken at a drain bias of 12V are shown in Table IX.

The maximum output power delivered from a GaAs MESFET operating at a drain-source voltage of V_{ds} is approximately given by

$$P_{\max} \approx \frac{I_m}{4} (V_{ds} - V_{kf}) \tag{52}$$

if V_{ds} is sufficiently lower than the drain-source breakdown voltage. By substituting the measured values of I_m and V_{kf} into (52), $P_{\max}$ was calculated for $V_{ds} = 12$V. Furthermore, $P_{\max}$ was predicted using the geometrical and material parameters, shown in Tables II and VI, to calculate I_m and V_{kf}. Both predicted values of $P_{\max}$ are shown in Table IX.

As seen in Table IX, there is excellent agreement between the measured and predicted values of $P_{\max}$ in devices C, D, and E at this drain bias voltage. However, the measured values of devices A and B were substantially smaller than the predicted values. This discrepancy could be caused by the saturation effect in output power as V_{ds} increased. This problem of power saturation will be discussed in a separate paper. Without power saturation mechanisms, *the maximum output power capability at microwave frequencies can be predicted by dc parameters as well as by device geometrical and material parameters.*

6.2 Minimum noise figure

Devices F through K were originally designed for low-noise amplifiers. The minimum noise figure, $F_{\min}$, of these devices was measured at 5.92 GHz in a coaxial system with double-slug tuners. The measured values are shown in Table X.

Table IX—Predicted and directly measured values of the maximum output power at 4 GHz and 12V drain bias

Parameter	Device				
$P_{\max}$ @ 12 volts	A	B	C	D	E
Measured directly (W)	0.40	0.40	0.44	0.36	1.00
Predicted from measured dc para's (W)	0.439	0.491	0.437	0.350	1.001
Predicted from geo. and mat. para's (W)	0.439	0.493	0.440	0.353	1.006

Table X—Predicted and directly measured values of the minimum noise figure at 5.92 GHz

Parameter	Device					
$F_{\min}$ @ 5.92 GHz	F	G	H	I	J	K
Measured directly (dB)	2.22	1.51	1.84	1.74	1.75	1.76
Predicted from measured dc para's (dB)	2.21	1.50	1.80	1.73	1.75	1.76
Predicted from geo. and mat. para's (dB)	2.12	1.56	1.79	1.70	1.72	1.80

A simple expression for $F_{\min}$ has been derived as

$$F_{\min} \approx 10 \log [1 + KfL \sqrt{g_m(R_g + R_s)}] \quad \text{(dB)}, \qquad (53)$$

where f is the frequency of interest in GHz and K is the fitting factor.[20] This fitting factor, which represents the channel material properties, ranges from 0.25 to 0.3 in most cases. Substituting (33) into (53) with a typical K-value of 0.27 yields

$$F_{\min} \approx 10 \log \left[1 + 0.27\, fL \sqrt{\frac{g'_m(R_g + R_s)}{1 - g'_m R_s}} \right] \quad \text{(dB)}. \qquad (54)$$

Using the measured values of g'_m, R_g, and R_s, and deduced value of L in (54), $F_{\min}$ was calculated for devices F to K. The minimum noise figure was also predicted from the geometrical and material parameters, using the values shown in Tables II, VI, and VII to calculate g_m, R_s, and R_g.

These predicted values are compared with the directly measured value in Table X. The agreement is excellent between them. This supports the idea that *the dc characterization of a low-noise GaAs MESFET makes it possible to predict $F_{\min}$ at microwave frequencies with a remarkably high accuracy.* Also, *once the geometrical and material parameters are given for a device, its $F_{\min}$ can be calculated as well.* It would be worthwhile to note that, if the operating frequency approaches the cutoff frequency of a device or frequencies where the skin effect on the gate metallization becomes significant, an additional term is required in (54) for an improved accuracy.[20]

VII. SUMMARY OF RELATIONSHIPS

Table XI is a summary of the relationships between the dc and rf performance parameters and the geometrical and material parameters of a GaAs MESFET.

VIII. CONCLUSIONS

This paper complements the recent study of a new model of the GaAs MESFET.[1] A new technique has been introduced in which the basic channel parameters, such as the effective gate length, channel doping, and channel thickness, are determined from the so-called dc

Table XI—Summary of relationships

dc Parameter	Equations	Participating Geometrical and Material Parameters
V_b	(34)	N
W_p	(3)	N, a
V_p	(3, 34)	N, a
I_s	(4)	N, a, Z
β	(6)	N, a, L
I_m	(5, 6, 4) or (7)	N, a, L, Z
R_s, R_d	(44, 45, 46)	$N, N^+, a_1, a_2, a_3, L_2, L_3, Z$
R_g	(48)	Z, z, h, L_g, ρ_g
I_o	(8, 9, 10, 11, 4)	N, a, L, Z, R_s
	or (8, 12, 13, 3, 4)	N, a, Z, R_s, V_c
V_{hf}	(14, 3, 4, 5, 6, 44, 45, 46)	$N, a, L, Z, R_s, R_d, V_c$
Performance Parameter	Equations	Above Parameters plus Bias Parameter
g_m	(36, 3, 4)	N, a, L, Z, I_c
g_m'	(36, 33, 3, 4, 44, 45, 46)	N, a, L, Z, R_s, I_c
$P_{\max}$	(52, 3, 4, 5, 6, 14, 44, 45, 46)	$N, a, L, Z, R_s, R_d, V_{ds}$
$F_{\min}$	(54, 36, 3, 4, 44, 45, 46, 48)	N, a, L, R_s, R_g, f

parameters. Also, a precise technique developed for measuring the dc parameters was shown. Using 11 sample devices chosen from a wide variety of designs, usefulness of the new techniques was demonstrated.

The determined values of the basic channel parameters for the sample devices were used to calculate their dc parameters, such as the maximum channel current, zero-gate-bias channel current, and transconductance in the simple, analytical expressions recently obtained.[1] Their predicted values were then compared with the measured values in excellent agreement for all devices used. Practical expressions, in terms of device geometrical and material parameters, developed for parasitic resistances were verified in good agreement with measured values on all sample devices.

Using the sample devices, it was demonstrated that the maximum output power and minimum noise figure at microwave frequencies can be predicted by dc parameters as well as by device geometrical and material parameters through simple, analytical expressions. In other words, proper dc characterization of a GaAs MESFET makes it possible to predict the microwave power handling capability and minimum noise property.

Finally, the relationships between the dc and rf performance parameters of a GaAs MESFET and its geometrical and material parameters were summarized with the relevant equations. This summary would be very useful as a handy reference for the design and optimization processes of a GaAs MESFET.

IX. ACKNOWLEDGMENTS

The author is grateful to J. V. DiLorenzo, H. M. Cox, L. A. D'Asaro,

B. S. Hewitt, L. C. Luther, W. C. Niehaus, W. Robertson, P. F. Sciortino, J. A. Seman, and J. R. Velebir for fabricating the sample devices used in this paper. He is also thankful to J. P. Beccone, D. E. Iglesias, F. M. Magalhaes, and W. O. Schlosser for preparing the microwave test equipment used for the output power and noise measurements, and to J. E. Kunzler and L. J. Varnerin for their support of the GaAs MESFET project.

REFERENCES

1. H. Fukui, "Channel Current Limitations in GaAs MESFETS," unpublished work.
2. W. Shockley, "A unipolar 'field-effect' transistor," Proc. IRE, *40* (November 1952), pp. 1365–1376.
3. A. B. Grebene and S. K. Ghandhi, "General Theory for Pinched Operation of the Junction Gate FET," Solid-State Elec., *12* (July 1969), pp. 573–585.
4. R. B. Fair, "Graphical Design and Iterative Analysis of the dc Parameters of GaAs FET's," IEEE Trans. Electron Devices, *ED-21* (June 1974), pp. 357–362.
5. R. A. Pucel, H. A. Haus, and H. Statz, "Signal and Noise Properties of Gallium Arsenide Microwave Field-Effect Transistors," *Advances in Electronics and Electron Physics*, vol. 38, New York: Academic Press, 1975, pp. 195–265.
6. S. M. Sze, *Physics of Semiconductor Devices*, New York: Wiley-Interscience, 1969, p. 393.
7. C. R. Crowell, J. C. Sarace, and S. M. Sze, "Tungsten-Semiconductor Schottky-Barrier Diodes," Trans. Metallurgical Soc. AIME, *233*, (March 1965), pp. 478–481.
8. P. L. Hower, W. W. Hooper, B. R. Cairns, R. D. Fairman, and D. A. Tremere, "The GaAs Field-Effect Transistor," *Semiconductors and Semimetals*, vol. 7, New York: Academic Press, 1971, pp. 147–200.
9. W. C. Niehaus, H. M. Cox, B. S. Hewitt, S. H. Wemple, J. V. DiLorenzo, W. O. Schlosser, and F. M. Magalhaes, "GaAs Power MESFETS," *Gallium Arsenide and Related Compounds (St Louis), 1976*, Conf. Series No. 33b, Bristol and London: The Institute of Physics, 1977, pp. 271–280.
10. B. S. Hewitt, H. M. Cox, H. Fukui, J. V. DiLorenzo, W. O. Schlosser, and D. E. Iglesias, "Low-Noise GaAs MESFETS: Fabrication and Performance," *Gallium Arsenide and Related Compounds (Edinburgh), 1976*, Conf. Series No. 33a, Bristol and London: The Institute of Physics, 1977, pp. 246–254.
11. H. M. Cox and J. V. DiLorenzo, "Characteristics of an AsCl$_3$/Ga/H$_2$ Two-Bubbler GaAs CVD System for MESFET Applications," *Gallium Arsenide and Related Compounds (St Louis), 1976*, Conf. Series No. 33b, Bristol and London: The Institute of Physics, 1977, pp. 11–22.
12. G. L. Miller, "A Feedback Method for Investigating Carrier Distributions in Semiconductors," IEEE Trans. Electron Devices, *ED-19* (October 1972), pp. 1103–1108.
13. H. M. Cox, private communication.
14. H. H. Berger, "Contact Resistance on Diffused Resistors," 1969 IEEE ISSCC Digest of Technical Papers, February 1969, pp. 160–161.
15. H. Murrmann and D. Widmann, "Current Crowding on Metal Contacts to Planar Devices," 1969 IEEE ISSCC Digest of Technical Papers, February 1969. pp. 162–163.
16. H. Macksey and R. Adams, "Fabrication Processes for GaAs Power FET's," Proc. Fifth Cornell Conf. on Active Semiconductor Devices for Microwave and Integrated Optics, 1975, pp. 255–264.
17. H. Matino, "A Study of GaAs Microwave Semiconductor Devices," Doctoral Dissertation (Japanese), 1972.
18. K. Heime, U. König, E. Kohn, and A. Wortmann, "Very low resistance Ni-AuGe-Ni contacts to n-GaAs," Solid-State Elec. *17* (1974), pp. 835–837.
19. P. Wolf, "Microwave Properties of Schottky-Barrier Field-Effect Transistors," IBM J. Res. Develop., *14* (March 1970), pp. 125–141.
20. H. Fukui, "Optimal Noise Figure of Microwave GaAs MESFETS," unpublished work.

Contributors to This Issue

Andres Albanese, B.S.E.E., 1970, University of Central Venezuela; M.Sc., 1972, University of Texas at Austin; Ph.D., 1976, Stanford University; Instituto Venezolano de Investigaciones Cientificas, 1969–1970; Bell Laboratories, 1975—. Mr. Albanese's current research interests are systems and components for optical communications.

Ronald E. Crochiere, B.S. (E.E.), 1967, Milwaukee School of Engineering; M.S. (E.E.), 1968, Ph.D. (E.E.), 1974, Massachusetts Institute of Technology; Bell Laboratories, 1974—. At Bell Laboratories, Mr. Crochiere has been involved in research activities in concepts of decimation and interpolation, sub-band and transform coding of speech, and the measurement of digital speech quality. In 1976 he received the IEEE Acoustics, Speech, and Signal Processing (ASSP) paper award for his paper on decimation and interpolation of digital signals. He has been an active member of the ASSP Technical Committee on Digital Signal Processing over the past four years. He has served for two years as a Technical Editor on digital signal processing for the ASSP transactions and was also a member of the conference steering committee which organized the second annual ASSP conference in Hartford, Conn., 1977. He is presently a member and secretary-treasurer of the ADCOM committee of the ASSP society of the IEEE.

John J. Dubnowski, B.Sc. (E.E.), 1968, M.Sc. (E.E.), 1971, New Jersey Institute of Technology; Bell Laboratories, 1967—. Mr. Dubnowski's principal activities in the Acoustics Research Department at Bell Laboratories have been concerned with the research and development of speech processing, coding, and analysis systems.

A. A. Fredericks B.S. (Math) 1962, Fairleigh Dickinson University; M.S., 1965, Ph.D. (Math), 1970, Courant Institute, N.Y. University; Bell Laboratories, 1961—. Mr. Fredericks worked in the military systems area until 1970. More recently he has worked on performance analyses of stored program control switching systems. As supervisor of the system analyses group in the Network Performance Planning Center, he is presently responsible for performance analyses of minicomputer-based operations support systems.

Hatsuaki Fukui, Doctor of Engineering (Electrical Engineering), Osaka University, Japan; Shimada Physical and Chemical Industrial Company, Tokyo, Japan, 1954–55; Tokyo Tsushin Kogyo Kabushiki Kaisha (former name of Sony Corporation), Tokyo, Japan, 1955–1962; Bell Laboratories, 1962—. At Sony, Mr. Fukui was engaged in the early development of transistors for consumer use. In 1960 he was assigned the total responsibility for Esaki tunnel diodes from design theory to sales promotion. Later he was in charge of advanced technology needed for realization of solid-state UHF TV and stereo receivers. At Bell Laboratories, he has been engaged in development of microwave semiconductor devices and subsystems, high-resolution video transmission devices and subsystems, and image pick-up devices and image display devices for future *PICTUREPHONE®* use. For the last five years, he has worked for the GaAs MESFET development project in various aspects. Senior Member, IEEE.

David J. Goodman, B.E.E., 1960, Rensselaer Polytechnic Institute; M.E.E., New York University; Ph.D. (E.E.), 1967, Imperial College, London; Bell Laboratories, 1967—. Mr. Goodman has studied various aspects of digital communications, including analog-to-digital conversion, digital signal processing, assessment of the quality of digitally coded speech, and error mechanisms in digital transmission lines. He is Head, Communications Methods Research Department. In 1974 and 1975, he was a Senior Research Fellow at Imperial College, London, England. Member, IEEE.

Janet S. Goodman, B.A., 1962, Smith College; B.A. (Honours), 1964, and Ph.D., 1968, University College, London; AT&T, 1972–1974; University of Essex, 1974–1976; Bell Laboratories, 1977; Central Headquarters, British Post Office, 1978—. Ms. Goodman is currently head of a section responsible for personnel selection methods used throughout the British Post Office.

Wayne S. Holden, Electronics Technology, 1970, RCA Institutes; Bell Laboratories, 1970—. Mr. Holden has been involved in the evaluation of optical fiber parameters and the design of electronic circuitry for optical fiber communication systems. He is presently engaged in the processing and evaluation of emitters and detectors used in fiber optic communications systems.

Paul J. Kuehn, Dipl.-Ing. (Electrical Engineering), 1967, University of Stuttgart, W.-Germany; Dr.-Ing., 1972, University of Stuttgart; Assistant Professor, 1967–1973, and Head of the Research Group, Stochastic Processes in Computers and Computer-Based Systems, 1973–1977, at the Institute of Switching and Data Technics, University of Stuttgart; Lecturer for Switching Systems, University of Erlangen, W. Germany, 1975–1977; Bell Laboratories, 1977–1978; Professor for Communications Switching and Transmission, the University of Siegen, W. Germany, 1978—. Member IEEE, German Chapter of ACM, German Communications Society (NTG), and German Informatics Society (GI).

Kenneth R. Laker, B.S.E.E., 1968, Manhattan College, M.S.E.E., 1970, Ph.D. (E.E.), 1973, New York University; Captain, USAF, Air Force Cambridge Research Laboratories, 1973–1977; Bell Laboratories, 1977—. Mr. Laker has worked in the areas of surface acoustic wave devices and active networks. Since joining Bell Laboratories he has been engaged in various development and exploratory activities associated with active-RC and active-switched capacitor filters. Member, Eta Kappa Nu, Sigma Xi; Senior Member, IEEE; Administrative Committee, IEEE CAS Society; Co-chairman, Optical, Microwave, and Acoustical Circuits Technical Committee, IEEE CAS Society; IEEE SU Group.

Barbara J. McDermott, B.A. (Psychology), 1949, University of Michigan; M.A. (Psychology), 1963, Columbia University; Haskins Laboratories, 1950–1959; Bell Laboratories, 1959—. Ms. McDermott has worked on speech quality evaluation and multidimensional scaling analysis. Member, Acoustical Society of America.

Carol A. McGonegal, B.S. (cum laude) (mathematics), 1974, Fairleigh Dickinson University; M.S. (computer science), 1977, Stevens Institute of Technology; Bell Laboratories, 1967—. Ms. McGonegal is a member of the Acoustics Research Department, where she has worked on problems in digital filter design, digital speech processing, computer voice response, and speaker verification.

Arun N. Netravali, B. Tech. (Honors), 1967, Indian Institute of Technology, Bombay, India; M.S., 1969, and Ph.D. (E.E.), 1970, Rice University; Optimal Data Corporation, 1970–1972; Bell Laboratories, 1972—. Mr. Netravali has worked on problems related to filtering, guidance, and control for the space shuttle. At Bell Laboratories, he has worked on various aspects of signal processing. He is presently

Head of the Visual Communication Research Department and a Visiting Professor in the Department of Electrical Engineering at Rutgers University. Member, Tau Beta Pi, Sigma Xi; Senior Member, IEEE.

Peter Noll, Dipl.-Ing., 1964, Dr.-Ing. (Electrical Communication Engineering), 1969, Habilitation, 1974, Technical University of Berlin, Germany; Heinrich-Hertz-Institut Berlin-Charlottenburg, 1964–1976. Mr. Noll was initially concerned with the development of electronic telephone exchanges. Since 1970, he has been engaged in research on speech coding and communication theory. During the summers of 1974 to 1977 he was on the technical staff of Bell Laboratories. Since 1976, he has been a member of the University of Bremen, Germany, as a Professor of Electrical Engineering and Statistical Communication Theory. Member, Nachrichtentechnische Gesellschaft (NTG), and Verein Deutscher Elektrotechniker (VDE), Germany. Senior member, IEEE.

Lawrence R. Rabiner, S.B. and S.M., 1964, Ph.D. (electrical engineering), Massachusetts Institute of Technology; Bell Laboratories, 1962–. From 1962 through 1964, Mr. Rabiner participated in the cooperative plan in electrical engineering at Bell Laboratories. He worked on digital circuitry, military communications problems, and problems in binaural hearing. Presently he is engaged in research on speech communications and digital signal processing techniques. He is coauthor of *Theory and Application of Digital Signal Processing* (Prentice-Hall, 1975) and *Digital Processing of Speech Signals* (Prentice-Hall, 1978). Former President, IEEE G-ASSP Ad Com; former Associate Editor, G-ASSP Transactions; former member, Technical Committee on Speech Communication of the Acoustical Society. Member, G-ASSP Technical Committee on Speech Communication, IEEE Proceedings Editorial Board, Eta Kappa Nu, Sigma Xi, Tau Beta Pi. Fellow, Acoustical Society of America and IEEE.

Gerald A. Reisner, B.S. (Applied Mathematics), 1967, New York University, School of Engineering; M.S. (Mathematics), 1970, Adelphi University; Ph.D. (Mathematics), 1974, University of Minnesota; Bell Laboratories, 1974—. Mr. Reisner is a member of the Network Measurements Department. His areas of work include mathematical modeling for network measurements, including the relationship of network service to customer call failures. He is presently responsible for service measurement planning for several local switching systems (No. 1 ESS and No. 5 ESS).

John D. Robbins, B.S.E.E., 1976, Rutgers University; Rutgers University, 1979—; Bell Laboratories, 1976—. Mr. Robbins is a member of the Visual Communication Research Department, where he has worked on various picture coding problems. He is a member of Tau Beta Pi and Eta Kappa Nu.

Carlo Scagliola, Dr. Ing. (Electronic Engineering), 1970, University of Pisa, Italy. Mr. Scagliola has been with CSELT (Centro Studi E Laboratori Telecommunicazioni), Turin, Italy since 1970. He has been engaged in adaptive speech coding, assessment of the quality of digitally coded speech and in studies of automatic synthesis of the Italian language. Mr. Scagliola served as a consultant at Bell Laboratories from January 1977 through January 1978.

Raymond Steele, B.Sc. (E.E.), 1959, University of Durham, England; Ph.D., 1975, Loughborough University of Technology, England. Mr. Steele was a lecturer at Royal Naval College, Greenwich, London from 1965 to 1968 when he became senior lecturer at Loughborough University. He has been engaged in source encoding of speech and picture signals and is the author of a book on delta modulation systems. He was a consultant at Bell Laboratories in the summers of 1975, 1977, and 1978.

Jose M. Tribolet, Engenheiro Electrotécnico, 1972, Instituto Superior Técnico, Lisbon, Portugal; M.S., 1974, E.E., 1975, Sc.D., 1977, Massachusetts Institute of Technology; Assistente Eventual, Instituto Superior Técnico, and Researcher, Centro de Estudos de Electrónica, Instituto de Alta Cultura, 1970–1972; Massachusetts Institute of Technology Research Laboratory of Electronics, 1972–1977; Bell Laboratories, 1977–1978; Professor, Department of Electrical Engineering, Instituto Superior Técnico. At M.I.T., Mr. Tribolet's research activities involved the application of homomorphic signal processing to speech and seismic data analysis. At Bell Laboratories, he worked on adaptive transform coding of speech. Member, Sigma Xi.

Papers by Bell Laboratories Authors

CHEMISTRY

Al Oxidation in Water. C. C. Chang, D. B. Fraser, M. J. Grieco, T. T. Sheng, S. E. Haszko, R. E. Kerwin, K. B. Marcus, and A. K. Sinha, J. Electrochem. Soc., *125* (May 1978), pp. 787–792.

An Examination of the Chemical Staining of Silicon. D. G. Schimmel and M. J. Elkind, J. Electrochem. Soc., *125* (January 1978), pp. 152–155.

Investigation of the Ti-Pt Diffusion Barrier for Gold Beam Lead on Aluminum. S. P. Murarka, H. J. Levinstein, I. Blech, T. T. Sheng, and M. H. Read, J. Electrochem. Soc., *125* (January 1978), pp. 156–162.

Plasma-Grown Oxide on GaAs—Semiquantitative Chemical Depth Profiles Obtained Using Auger Spectroscopy and Neutron Activation Analysis. C. C. Chang, R. P. H. Chang, and S. P. Murarka, J. Electrochem. Soc., *125* (March 1978), pp. 481–487.

Raman Study of the O_2F_2 + VF_5 Reaction: Isolation and Identification of an Unstable Reaction Intermediate. J. E. Griffiths, A. J. Edwards, W. A. Sunder, and W. E. Falconer, J. Fluorine Chem., *11* (1978), pp. 119–142.

Tapered Windows in Phosphorus-Doped SiO_2 by Ion Implantation. J. C. North, T. E. McGahan, D. W. Rige, and A. C. Adams, IEEE Trans. Electron Dev., *ED-25* (July 1978), pp. 809–812.

Urban Kinetic Chemical Calculations with Altered Source Conditions. T. E. Graedel, L. A. Farrow, and T. A. Weber, Atmos. Environ., *12* (1978), pp. 1403–1412.

COMPUTING

A Magnetic Bubble Sub-System for Microcomputer Applications. R. J. Radner, Digest-Intermag Conference (May 9–12, 1978), pp. 1–4.

A Scheduler for Real-Time Task Control in Microcomputers. E. A. Parrish, Jr. and V. K. L. Huang, IEEE Trans. Ind. Electron. Contr. Instrum., *25*, No. 1 (February 1978), pp. 21–26.

ELECTRICAL AND ELECTRONIC ENGINEERING

Analysis of Phosphorous-Diffused Layers in Silicon. R. B. Fay, J. Electrochem. Soc. *125* (February 1978), pp. 323–327.

Effect of Interfacial Reactions on the Electrical Characteristics of Metal Semiconductor Contacts. E. H. Nicollian and A. K. Sinha, Electrochem. Soc. Monograph Thin Films—Interdiffusion and Reactions, New York: John Wiley & Sons (May 1978), pp. 481–531.

Electrical Properties of Si-N Films Deposited on Silicon from Reactive Plasma. A. K. Sinha and R. E. Smith, J. Appl. Phys., *49* (1978), pp. 2756–2760.

Electrochromism in Anodic Iridium Oxide Films. S. Gottesfeld, J. D. E. McIntyre, G. Beni, and J. L. Shay, Appl. Phys. Lett., *33* (July 1978), pp. 208–210.

An Integrated PCM Encoder Using Interpolation. J. L. Henry and B. A. Wooley, IEEE Inter. Solid State Circuits Conf. Tech. Papers, *21* (February 1978), pp. 184–185.

Interatomic Internal Photoemission Peaks in (Auger) Electron Spectra. C. C. Chang, J. Electron. Spectros. Relat. Phenom., *13* (1978), pp. 255–261.

A Low Substrate Leakage Junction Isolated p-n-p-n Crosspoint Array. J. M. Adrian, P. W. Shackle, A. R. Hartman, and R. L. Pritchett, IEEE J. Solid State Circuits, *SC-13*, No. 2 (1978), pp. 210–218.

A Modular High-Speed Serial Pipeline Multiplier for Digital Signal Process-

ing. G. L. Baldwin, B. L. Morris, D. B. Fraser, and A. R. Tretola, IEEE J. Solid State Circuits, *SC-13* (June 1978), pp. 400–408.

Quantified Conditions for Emitter-Mesfet Dislocation Formation in Silicon. R. B. Fair, J. Electrochem. Soc., *125* (June 1978), pp. 923–926.

Reactive Plasma Deposited Si-N Films for MOS-LSI Passivation. A. K. Sinha, H. J. Levinstein, T. E. Smith, D. Quintana, and S. E. Haszko, J. Electrochem. Soc., *125* (April 1978), pp. 601–608.

Technology Dependent Logic Faults. R. W. Wadsack, COMPCON IEEE No. *78 CH 1328-4C* (1978), pp. 124–129.

A UHF Monolithic Operational Amplifier. W. Kruppa and F. O. Waldhauer, IEEE ISGCC Dig. Tech. Papers, *21* (February 1978), pp. 74–75.

Variational Approximations to Current Distribution Problems. II. Rectilinear Electrodes and Baffles. S. H. Glarum, J. Electrochem. Soc., *125* (1978), pp. 84–89.

MATERIALS SCIENCE AND ENGINEERING

Preparation and Electropolishing of Thin Gold Disc Specimens for Transmission-Electron-Microscope Examinations. W. F. Peck, Jr. and S. Nakahara, Metallography, *11* (1978), pp. 347–354.

PHYSICS

Buried Injector Logic: Second Generation I(2)L Performance from Uncompromised SBC Processing. A. A. Yiannoulos, Int. Solid State Circuit Dig. Tech., San Francisco, Ca., *21* (February 1978), pp. 12–13.

Charge Coupled Transversal Triple Filter with Matched Threshold Levels and Binary Outputs. P. I. Suciu, M. F. Tompsett, and J. R. Barner, Int. Symposium on Circuits and Systems, Proceedings *P741-3* (May 1978), pp. 741–743.

A Charge-Sheet Model of the MOSFET. J. R. Brews, Solid State Electron. *21*, No. 2 (1978), pp. 345–355.

Deuteron Quadrupole Coupling in Molecules. L. C. Snyder, J. Chem. Phys., *68*, No. 1 (1978), pp. 291–294.

Differential Studies of Dual-Dielectric, Charge-Storage Cells. K. K. Thornber, D. Kahng, D. M. Boulin, C. T. Neppell, and W. J. Sundburg, J. Appl. Phys., *49* (July 1978), pp. 4047–4063.

Electron-Electron Effects in the Writing and Erasing of Dual-Dielectric, Charge-Storage Cells. K. K. Thornber and D. Kahng, *32*, No. 3 (February 1, 1978), pp. 131–133.

Gettering of Surface and Bulk Impurities in Czochrolski Wafers. G. A. Rozgonyi and C. W. Pearce, Appl. Phys. Lett., *32* (June 1, 1978), pp. 747–749.

Kinetics of the Slow Trapping Instability at the Si/SiO$_2$ Interface. A. K. Sinha and T. E. Smith, J. Electrochem. Soc., *125* (1978), pp. 743–746.

Lorentz-Lorenz Correlation for Reactively Plasma Deposited Si-N Films. A. K. Sinha and E. Luguijo, Appl. Phys. Lett., *32* (February 1978), pp. 245–246.

Optical Absorption as a Control Test for Plasma Silicon Nitride Deposition. Myron J. Rana and David R. Wonsidler, J. Electrochem. Soc., *125* (January 1978), pp. 99-101.

Oxygen Partial Pressure Dependence of the Retrogrowth of Oxidation Induced Stacking Faults in S (100) Silicon. S. P. Murarka, J. Appl. Phys., *49* (1978), pp. 2513–2516.

Plasma Etching of Si and SiO$_2$—The Effect of Oxygen Additions to CF$_4$ Plasmas. C. J. Mogab, A. C. Adams, and D. L. Flamm, J. Appl. Phys., *49* (July 1978), pp. 3796–3803.

Pressure-Broadened Linewidths of the R(9.5)$_{32}$ No Transition. R. E. Richton, Appl. Opt., *17* (May 15, 1978), pp. 1606–1609.

Reliability of Epoxy and Silicone Molded Tape-Carrier Silicon Integrated Circuits with Various Chip-Protection Coatings. N. J. Chaplin and A. J. Masessa, 16th Annual Proceedings of the Reliability Physics Symposium, *78CH1294-8PHY* (1978), pp. 187–193.

Singlet-Triplet Anticrossings Between the Doubly-Excited 3(1)K State and the G(3D)3 Sigma (G$^+$) State of Hydrogen (2). R. S. Freund, T. A. Miller, R. Jost, and M. Lombardi, J. Chem. Phys., *68*, No. 4 (1978), pp. 1683–1688.

The Temperature Dependence of Stresses in Aluminum Films on Oxidized Silicon Substrates. A. K. Sinha and T. T. Sheng, Thin Solid Films, *48* (1978), pp. 117–126.

Thermal Stresses and Cracking Resistance of Dielectric Films (SiN, Si_3N_4, SiO_2) on Si substrates. A. K. Sinha, H. J. Levinstein, and T. E. Smith, J. Appl. Phys., *49* (1978), pp. 2423–2427.