December 1985 Vol. 64 No. 10

AT&T TECHNICAL JOURNAL

A JOURNAL OF THE AT&T COMPANIES

Lightwave Communications

Microwave Radio

Speech Processing

Traffic

Local Area Network

Queueing

Viterbi Algorithm

VCR Information System

Protocols

AT&T TECHNICAL JOURNAL (ISSN 8756-2324) is published ten times each year by AT&T, 550 Madison Avenue, New York, NY 10022; C. L. Brown, Chairman of the Board; L. L. Christensen, Secretary. The Computing Science and Systems section and the special issues are included as they become available. Subscriptions: United States—1 year $35; foreign—1 year $45.

Payment for foreign subscriptions or single copies must be made in United States funds, or by check drawn on a United States bank and made payable to the AT&T Technical Journal, and sent to AT&T Bell Laboratories, Circulation Dept., Room 1E335, 101 J. F. Kennedy Pky, Short Hills, NJ 07078.

Back issues of the special, single-subject supplements may be obtained by writing to the AT&T Customer Information Center, P.O. Box 19901, Indianapolis, Indiana 46219, or by calling (800) 432-6600. Back issues of the general, multisubject issues may be obtained from University Microfilms, 300 N. Zeeb Road, Ann Arbor, Michigan 48106.

Printed in U.S.A. Second-class postage paid at Short Hills, NJ 07078 and additional mailing offices. Postmaster: Send address changes to the AT&T Technical Journal, Room 1E335, 101 J. F. Kennedy Pky, Short Hills, NJ 07078.

AT&T
TECHNICAL
JOURNAL

VOL. 64 DECEMBER 1985 NO. 10

AT&T Technical Journal
Vol. 64, No. 10, December 1985
Printed in U.S.A.

Coherent Lightwave Communications

By J. SALZ*

(Manuscript received March 29, 1985)

The chief objective of this paper is to develop a fundamental understanding of the effects of laser phase noise on the performance of coherent lightwave communication systems. A comprehensive treatment applicable to a wide variety of coherent receiver designs under a broad range of conditions is provided. Our models and analytical tools are developed in sufficient detail to encompass a broad range of applications. Formulas are derived for the bit error rate in homodyne and heterodyne Phase Shift Keying (PSK), Differential Phase Shift Keying (DPSK), Frequency Shift Keying (FSK) and on-off keying. Estimates are provided of the penalties accrued due to phase noise. Based on detailed mathematical analysis and estimates, we made several findings. Near quantum-limited receiver sensitivity can be achieved with PSK using homodyne detection only at signaling rates 3000 times greater than the laser linewidth. A receiver sensitivity 3 to 6 decibels poorer than the quantum limit can be achieved with heterodyne rather than homodyne detection. DPSK, for example, can operate at rates only 300 times greater than the laser linewidth. At lower rates, FSK is an attractive candidate. It can be designed to be extremely tolerant of phase noise by using wide frequency deviations.

I. INTRODUCTION

Despite the rapid advance of lightwave technology over the past decade, the basic operation of an optical fiber communications link has remained essentially unchanged. Direct modulation of the source (on-off keying) and direct detection at the receiver using a pin diode or Avalanche Photodiode (APD) have been the mainstays of lightwave

* AT&T Bell Laboratories.

systems since their infancy. Recent advances in lightwave components, however, now permit significant improvements on this time-honored approach. For example, external modulation of the laser with an electro-optic device has recently produced better long-distance performance at very high bit rates ($\gtrsim$4 Gb/s) than direct modulation of the laser itself. Another promising technique now being pursued in research laboratories around the world is the use of coherent lightwave—the optical analog of superheterodyne radio reception. Here we provide a comprehensive analytical treatment applicable to a wide variety of coherent receiver designs under a broad range of conditions. Recognizing that not all contingencies can be covered explicitly, we have endeavored to develop our model and analytical tools in sufficient detail so they can be applied to other cases of interest.

Unlike direct detection, where the optical signal is converted directly into a demodulated electrical output, the coherent receiver first adds to the signal a locally generated optical wave and then detects the sum. The resulting photocurrent is a replica of the original signal, translated down in frequency from the optical domain ($\sim 10^5$ GHz) to the radio domain ($\lesssim$ few GHz), where conventional electronic techniques can be used for further signal processing, such as filtering and demodulation. This method offers significant improvements in receiver sensitivity and wavelength selectivity compared with direct detection. In the 1.3- to 1.6-μm lightwave band, for example, an ideal coherent receiver requires a signal energy of only 10 to 20 photons per bit to achieve a bit error rate of 10^{-9}—far less than the roughly 1000 photons required by today's APDs. And because of its improved selectivity, a coherent receiver might permit wavelength-division-multiplexed systems with channel spacings of only, say 100 MHz, instead of the 100 GHz required with conventional optical multiplexing technology. A further advantage of coherent reception, not often cited but potentially very important, is that it allows the use of electronic equalization to compensate for the effects of optical pulse dispersion in the fiber.

The possible advantages of coherent optical communications have been explored for numerous applications. Much of the earlier work emphasized space communications, where highly collimated laser beams could be used to span enormous distances.[1,2] More recently, the use of coherent techniques in optical fiber systems has received considerable attention. Especially at bit rates above 2 Gb/s, where APD performance begins to deteriorate, the high sensitivity and potentially broad bandwidth of coherent receivers is a powerful stimulus to further research. As part of this effort, theroretical analyses of several types of coherent receivers have been published in the literature.[3-5] (Since these investigations are generally based on fundamental equations and

hardware designs closely resembling those encountered in the microwave domain, it is perhaps not too surprising that their general conclusions are also similar. In both cases, for example, the most energy-efficient binary system uses phase shift keying and coherent demodulation.) The performance of several experimental receivers has been compared with theory, and the agreement is generally good, especially when HeNe or YAG lasers are used for the optical sources. In practical coherent lightwave systems, however, it is expected that semiconductor injection lasers will be used; when these have been employed in coherent receiver experiments, measured sensitivity is almost always degraded. The effect is most pronounced in angle-modulation experiments, where receiver performance is often so poor that low error rates ($<10^{-9}$) cannot be achieved at all.[6,7] The cause of this degradation has been identified as laser phase noise, an impairment that is particularly severe in semiconductor devices. The effect of this noise mechanism is to impress random phase modulation on the otherwise monochromatic output of the laser, thereby impairing its performance in angle-modulation experiments. A fundamental understanding of this impairment and its effects on performance is the primary objective of this paper.

Laser phase noise is usually characterized in terms of the linewidth of the laser emission spectrum, (a readily measurable quantity that is directly proportional to the spectral density of the underlying phase noise process.) As was implied above, the linewidths available with today's distributed feedback semiconductor lasers, typically 5 to 50 MHz, are too broad to take full advantage of coherent techniques. Consequently the realization of a stable, reliable narrow-linewidth source is an extremely high priority in lightwave research. Several promising techniques have been demonstrated in the laboratory, but their usefulness under actual field conditions has yet to be established. Since reducing laser linewidth appears to be a difficult task, it is important to understand the effects of this impairment in order to establish precisely how much reduction is required.

The paper begins with an executive summary. Section II provides a brief review of direct detection methods and fundamental limits. Properties of phase noise in lasers are reviewed in Section III. Analysis of phase-lock technique are presented in Section IV. Frequency Shift Keying (FSK) is treated in Section V while Differential Phase Shift Keying (DPSK) and on-off-keying are treated in Sections VI and VII.

1.1 Executive summary

A coherent lightwave receiver is the optical analog of a superheterodyne radio set. Instead of detecting photons directly, the coherent receiver first converts the incoming signal from the optical regime

down to the radio regime, and then uses conventional electronic circuitry to perform various signal processing operations, such as amplification and demodulation. In principle, this technique can yield large increases (~20 dB) in receiver sensitivity compared with direct detection using today's avalanche photodiodes. Indeed, in the 1.3- to 1.6-μm lightwave band, coherent receivers offer the only realistic hope of approaching the so-called "quantum limit" of receiver sensitivity: ~10 photons/bit at 10^{-9} error rate. To date, however, the performance of experimental coherent receivers (especially those employing semiconductor lasers) has fallen short of the idealized theoretical predictions. One of the prime causes of this degradation has been identified as laser phase noise, a phenomenon that is known to be particularly serious in semiconductor devices. And since semiconductor lasers are, at present, the preferred candidates for coherent systems applications, it is imperative to develop an understanding of their noise properties. Our goal in this paper is to present a comprehensive treatment of the deleterious effects of laser phase noise in digital lightwave systems, so that the resulting degradation can be understood and predicted.

Laser phase noise is a random process driven by spontaneous emissions within the laser cavity, which cause the phase of the optical output wave to execute a random walk away from the value it would have had in the absence of spontaneous emission. This random phase process manifests itself as a broadening of the laser emission spectrum; it is the cause of the broad linewidth (typically 5 to 100 MHz) of today's InGaAsP Distributed Feedback (DFB) lasers. In communication systems, phase noise degrades performance because unwanted phase fluctuations in the received wave impair the demodulation process, especially when Phase Shift Keying (PSK) is used. At low signaling rates, the accumulated phase "wander" during a signaling interval might be so great that PSK cannot be used at all. In general, however, as the bit rate is increased, the impairment due to phase noise can be made negligibly small.

1.2 The central question

The central question addressed in this paper is, How high must the signaling rate be in order to ensure tolerable system degradation due to phase noise? The answer, not surprisingly, depends on system design constraints. For example, if one requires quantum-limited receiver sensitivity, then PSK with homodyne detection must be used. Based on detailed mathematical analysis and estimates we conclude that the degradation or penalty due to phase noise can be kept small (<1 dB) only if the ratio of signaling rate to laser linewidth, R/B_L, is greater than 3000. For a laser with B_L = 10 MHz, this condition implies a signaling rate of 30 Gb/s—well outside the range of current

technology. To operate at lower rates, one might use any of several techniques available for narrowing laser linewidth, but only at the price of a substantial increase in complexity. With the theoretical guidelines presented in this paper, the design engineer can strike a balance between the cost of linewidth reduction and the value of improved system performance.

If a system design can tolerate a receiver sensitivity 3 to 6 dB poorer than the quantum limit, then considerable robustness against phase noise can be achieved by using heterodyne, rather than homodyne, detection. With heterodyne differential detection of PSK, for example, the phase-noise penalty is less than 1 dB for $R/B_L \gtrsim 300$, an order-of-magnitude improvement over the homodyne case. Finally, we consider the intriguing case of FSK, which can be made extremely tolerant of phase noise by using very wide transmitter frequency deviation. At moderate bit rates ($\lesssim$500 Mb/s), where direct-deviation laser FSK transmitters operate fairly well, this modulation technique appears to be most attractive.

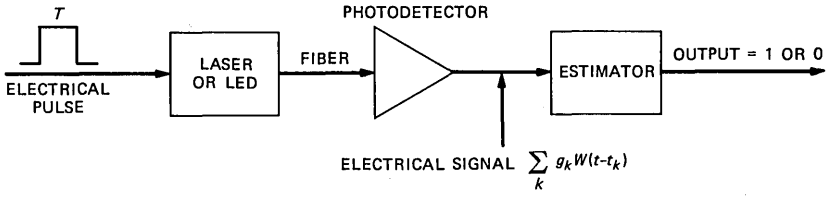
II. REVIEW OF DIRECT DETECTION AND FUNDAMENTAL LIMITS

2.1 Direct detection

Before commencing our principal investigation of coherent techniques, we briefly review some results related to direct detection of light signals.

Direct detection of light pulses implies a photodetector that converts light energy to electrical signals. The detection mechanism is based upon photon counting, which is subject to statistical fluctuations. More specifically, the photon counting process is a time-varying Poisson process whose intensity function $\lambda(t)$ is directly proportional to the information-bearing data wave.

In the case of binary transmission, the choice between a one or a zero is translated into the presence or absence of a burst of optical energy. As an illustration, consider the passage of a single pulse through an ideal transmission model depicted in Fig. 1. In the case of a one being transmitted, a square electrical signal turns on the laser or LED and energy is sent into the fiber. In the photodetector, light will be detected due to the electromagnetic energy present. Exactly when in time the photons register on the detector is random. The actual electrical current at the output of this device caused by a photon is a wideband pulse $g \cdot w(t)$ (which is very narrow compared with the signal duration T), where g (gain) is an integer-valued random variable or $g = 1$, depending on whether an (APD) or a pin diode is used. In practical systems where amplification of weak signals is required, APD's are invariably used.



(t_k) ARE RANDOM ARRIVAL TIMES, POISSON DISTRIBUTED WITH A MEAN ARRIVAL RATE $\lambda(t)$ PHOTONS/s

(g_k): AVALANCHE GAIN OF PHOTODETECTOR

$E(n) = \lambda T$ = EXPECTED NUMBER OF PHOTONS/BIT

$Pe = \frac{e^{-\lambda T}}{2} = \frac{1}{2} e^{-2P}$, P = OPTICAL ENERGY

Fig. 1—On-off direct detection.

Assuming that superposition holds for optical fiber transmission, the single-pulse description may be extended to an entire data wave. If one transmits a sequence of on or off pulses, then the *received signal*, defined as the electrical output of the photodetector on which processing is performed, is written as

$$I(t) = \sum_n g_n w(t - t_n), \tag{1}$$

where the time points $\{t_n\}$ form a Poisson process having intensity function $\lambda(t)$, with

$$\lambda(t) = \sum_n a_n h(t - nT) \tag{2}$$

and $h(t)$ is a square pulse, $\{a_n\} = 0$ or 1 are the data levels, $\{g_n\}$ is avalanche gain, T = signaling interval, and $w(t)$ = output pulse of the photodetector.

In this simple model, to detect the jth bit, one integrates the output of the photodetector over the jth T-second interval and compares the random variable with a threshold. If the output is greater than the threshold, a one is declared; if it is less, a zero is declared.

In the ideal situation, when a pin diode is used, $g_n = 1$ and when the threshold is set at zero, the average output of the integrator will yield $\int_0^T \lambda(t)dt = \lambda T$ when a one is sent and zero output when a zero is sent. Since the number of counts n with intensity λT is Poisson distributed

$$p(n) = \frac{(\lambda T)^n e^{-\lambda T}}{n!}, \tag{3}$$

and the chance of making an error is just $1/2\, p(n = 0)$ or,

$$Pe = \frac{1}{2} e^{-\lambda T}. \tag{4}$$

The average optical energy, photons per bit, is just $P = 1/2\ (\lambda T) + 1/2(0)$ and so (4) is written

$$Pe = \frac{1}{2}\ e^{-2P}. \tag{5}$$

This is a fundamental limit on the bit error rate and is commonly referred to as the "quantum limit."

Equation 5 implies that in order to obtain an error rate of 10^{-9}, about 10 photons/bit are required. This of course is the error rate achieved in the absence of coding. It has been shown recently[8] that by employing coding the number of required photons per bit is on the order of 2 to 3 provided the information rate is less than a characteristic rate called channel capacity.

When an avalanche detector is used to gain optical amplification, the average value of $\{g_n\}$ may be large but the fluctuations are also large causing amplitude jitter. The penalties incurred by avalanche detectors have been extensively studied.[9] Depending on the type of avalanche detectors used, the loss can be anywhere from 10 to 20 dBs from the quantum limit (see, for example, Ref. 10). Thus one of the chief motivations for turning to coherent techniques is to minimize this tremendous loss in detector sensitivity.

2.2 Homodyne direct detection and the super quantum limit

We begin the discussion of coherent techniques and their possible merits by assuming that the electromagnetic wave at the output of a laser can be represented as

$$s(t) = A \cos \omega_0 t, \tag{6}$$

where A^2 is proportional to the optical power. Now suppose that this wave is phase modulated so that a one results in $A \cos \omega_0 t$ and a zero results in $-A \cos \omega_0 t$. An ideal homodyne detector adds to the received wave a local carrier wave of amplitude equal to exactly A. So, the sum is

$$s_0(t) = (A \pm A)\cos \omega_0 t. \tag{7}$$

When the sum is detected by a photodetector (pin diode) and the output integrated for T seconds, one obtains for the average number of counts λT, either $4A^2T$ or 0. The average transmitted optical energy in this case is $P = A^2T$ and so the probability of a bit error is

$$Pe = \frac{1}{2} e^{-4P}.$$

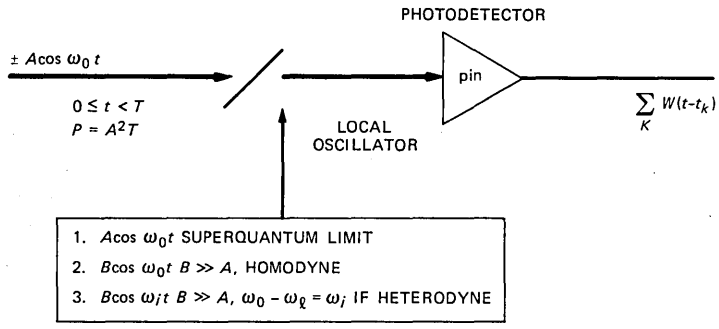


Fig. 2—Ideal homodyne and heterodyne techniques.

This result indicates a 3-dB improvement over the quantum limit, and it is often referred to as the "super quantum" limit.[11] Reviewing briefly, to achieve the super quantum limit, the local laser had to know exactly the frequency, phase, and the magnitude of the transmitter laser—a rather ambitious requirement. This detector is depicted in Fig. 2 with alternative No. 1 used as the input to the photodetector.

Now suppose that we relax the requirements on the local laser and permit its intensity to be any value B, but still requiring knowledge of the transmitted carrier frequency and phase. Now the combined waves become

$$s_0(t) = (B \pm A)\cos \omega_0 t. \tag{8}$$

Again (8) is detected by a photodetector and consequently the average number of counts at the output after integration is now $(B \pm A)^2T$, where B is the amplitude of the local laser and it is assumed that $B \gg A$.

To estimate the resulting bit error rate in this situation we invoke a limit theorem. The theorem has to do with the conditions under which a "shot noise" process—the output current from the photodetector—is well approximated by a "white gaussian" noise process. The main requirement is that the rate of photon arrivals be large. Since B in (8) can be made as large as one desires, the average number of photons is proportional to $\lambda T = (B^2 + A^2 \pm 2AB)T$. If the common bias term $(B^2 + A^2)T$ is subtracted from λT, there is left an antipodal signal pair $\pm 2ABT$ for the net average counts corresponding to reception of binary ones and zeroes. The variance of the resulting Poisson process also equals λT and since $B \gg A$ by hypothesis, the variance is essentially TB^2. Now in the limit of large number of counts due to the addition of the local laser to the incoming signal, the output electrical signal can be modeled by

$$s_0(t) = \pm 2AB + n(t), \qquad 0 \le t \le T, \tag{9}$$

where $n(t)$ is a white Gaussian noise process with double-sided spectral density equal to B^2. Integrating (9) from 0 to T, results in a Gaussian random variable. The resulting bit error rate is then

$$\begin{aligned} P_e &= \frac{1}{2}\operatorname{erfc}\sqrt{2A^2T} \sim e^{-2A^2T} \\ &= e^{-2P}, \end{aligned} \tag{10}$$

which is asymptotically (large P) the quantum limit. We have thus demonstrated that an ideal homodyne detector using a pin photodiode achieves the quantum limit. This is made possible by the availability of large "local" optical power that provides indirect amplification of the incoming weak optical signal. While providing amplification, the procedure also produces additive noise. This mode of detection is depicted in Fig. 2 with alternative No. 2 for the input to the photodetector.

2.3 Ideal heterodyne detection

Finally we consider a detection technique where, instead of translating the incoming optical wave directly to baseband, it might be advantageous in some cases to make a frequency translation to an Intermediate Frequency (IF). This procedure is called heterodyne reception and it is depicted in Fig. 2 with alternative No. 3 for the input to the photodetector.

To understand the consequences of this approach we proceed as follows. Let the local laser frequency be denoted by ω_ℓ and the incoming optical frequency by ω_0 such that the IF frequency is $\omega_i = \omega_0 - \omega_\ell$. The addition of the two waves now results in

$$s(t) = \pm A \cos \omega_0 t + B \cos \omega_\ell t, \qquad 0 \le t \le T, \tag{11}$$

where we denote the phase modulation by ± 1 and again require that $B \gg A$.

Expressing $s(t)$ in terms of the envelope and phase about ω_ℓ results in the representation

$$s(t) = E(t)\cos(\omega_\ell t + \beta(t)), \tag{12}$$

where

$$E(t) = \sqrt{(B \pm A \cos \omega_i t)^2 + A^2 \sin^2 \omega_i t}, \tag{13}$$

and

$$\beta(t) = \tan^{-1} \frac{\pm A \sin \omega_i t}{B \pm A \cos \omega_i t}. \tag{14}$$

Table I—Ideal Performance

1. Super homodyne	e^{-4P}
2. Homodyne	e^{-2P}
3. Heterodyne	e^{-P}

$P = A^2T$ = Energy per bit.

The response of the photodiode to the wave (12) is again a shot-noise process with intensity function, λ_0, equal to the envelope squared.

$$\lambda_0(t) = B^2 + A^2 \pm 2AB \cos \omega_i t \tag{15}$$

Using the same limit arguments as in the previous section, we first subtract $B^2 + A^2$ from $\lambda_0(t)$, which retains the antipodal signal pair

$$\pm 2AB \cos \omega_i t. \tag{16}$$

Because $B \gg A$ the fluctuating noise is white Gaussian with double-sided spectral density $\cong B^2$. Denoting the additive noise by $n(t)$, the equivalent signal-in-noise problem after heterodyning becomes

$$s_0(t) = \pm AB \cos \omega_i t + n(t), \qquad 0 \leq t \leq T. \tag{17}$$

This is a standard elementary detection problem and deciding whether a plus or a minus was sent is accomplished by multiplying (17) by $\cos \omega_i t$, integrating for T seconds, and comparing the result with a threshold set to zero. The decision statistic is

$$\pm ABT + \int_0^T n(t) \cos(\omega_i t) dt, \tag{18}$$

where we have neglected the double frequency term. Since the random variable $\int_0^T n(t) \cos(\omega_i t)\, dt$ has variance equal to $B^2T/2$, the bit error rate in this case is asymptotically

$$P_e \sim e^{-\frac{A^2B^2T^2}{B^2T}} = e^{-A^2T} = e^{-P}. \tag{19}$$

The exponent is seen to be a factor of 2 smaller than in (10) and because of this heterodyne detection is 3 dB inferior to the quantum limit. The asymptotic performance of the ideal frequency translation methods just discussed are summarized in Table I.

With these preliminaries we are now in a position to discuss more realistic detection systems, where laser phase noise must be taken into account. Before doing this however, we briefly review the origins of this noise.

III. PHASE NOISE IN LASERS

Phase or frequency noise in lasers is a well-known and documented phenomenon that sets fundamental limitations on the performance of

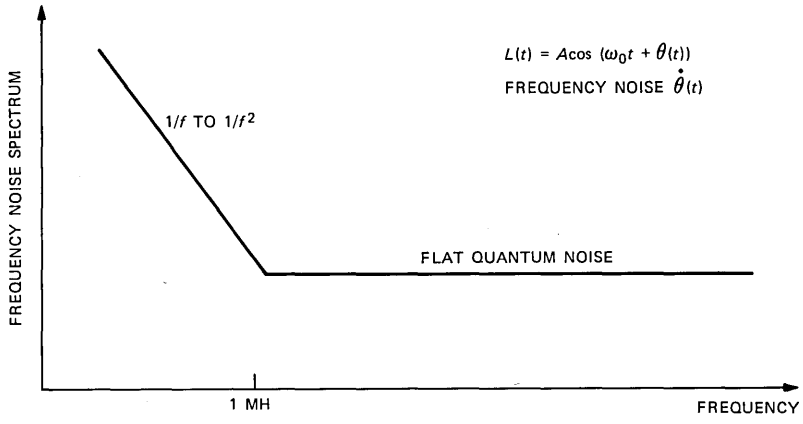


Fig. 3—Laser phase noise.

coherent optical communications.[12-14] It has been observed that the spectral density of this frequency noise has a $1/f$ to $1/f^2$ characteristic up to around 1 MHz, and is flat for frequencies above 1 MHz,[15] as shown in Fig. 3. The flat, or white, component is associated with quantum fluctuations and is the principal cause for line broadening. From a communications theory point of view, the relatively low-frequency components can be easily tracked, and so we shall not dwell on this part of the noise.[16] Our main focus here will be on the white component.

Laser phase noise is caused by randomly occurring spontaneous emission events, which are an inevitable aspect of laser operation. Each event causes a sudden jump (of random magnitude and sign) in the phase of the electromagnetic field generated by the device. As time evolves the phase executes a random walk away from the value it would have had in the absence of spontaneous emission. The mean-squared phase deviation grows approximately linearly with time, and since the average time between steps in the random walk becomes vanishingly small, the random phase $\theta(t)$ becomes in the limit a Wiener process characterized by a zero-mean, white Gaussian frequency noise $\mu(t)$ with two-sided spectral density N_0.[17] Thus, the phase process is represented as

$$\theta(\tau) = 2\pi \int_0^\tau \mu(t)dt, \tag{20}$$

and the mean-squared phase deviation is

$$E\theta^2(\tau) = E\left[2\pi \int_0^\tau \mu(t)dt\right]^2 = (2\pi)^2 N_0 \tau, \tag{21}$$

where E denotes mathematical expectation.

To determine the parameter N_0 (which is a function of both the laser structure and the operating conditions), one can measure the spectral density of the frequency fluctuations in the emitted light and hence determine N_0 directly. Experiments of this sort have shown that the representation in (20) using the white-noise approximation for $\mu(t)$ is reasonably accurate for 0.1 ns $\lesssim \tau \lesssim$ 1 μs, which is adequate for our present purposes. Another technique for measuring N_0 makes use of the fact that phase noise causes an observable broadening of the laser emission spectrum. In effect, the accumulated phase error given by (21) limits the duration of temporal coherence of the laser radiation to an interval of roughly $1/(2\pi)^2 N_0$; the corresponding linewidth is therefore proportional to the noise density N_0. The following discussion makes this relationship more precise.

Consider the sine-wave random process,

$$s(t) = A \cos(2\pi f_0 t + \theta(t) + \phi), \tag{22}$$

where the innocuous inclusion of the uniform phase ϕ renders $s(t)$ a stationary process with correlation function,

$$\begin{aligned} R(\tau) &= Es(t)s(t+\tau) \\ &= \frac{A^2}{2} R_e \left\{ e^{i2\pi f_0 \tau - \frac{(2\pi)^2}{2} N_0 |\tau|} \right\}. \end{aligned} \tag{23}$$

A simple calculation reveals that the Fourier transform of (23), the power spectrum, is

$$G(f) = \frac{A^2}{4\pi^2 N_0} \left\{ \left(1 + \left(\frac{f + f_0}{\pi N_0} \right)^2 \right)^{-1} + \left(1 + \left(\frac{f - f_0}{\pi N_0} \right)^2 \right)^{-1} \right\}, \tag{24}$$

and a quick sanity check yields

$$\int_{-\infty}^{\infty} G(f) df = \frac{A^2}{2}, \tag{25}$$

as it should. A sketch of the baseband spectrum, commonly referred to as *Lorentzian* is shown in Fig. 4. The parameter characterizing $G(f)$, N_0, can be experimentally determined by measuring the 3-dB bandwidth of the spectrum around f_0. Denoting the total (two-sided) 3-dB bandwidth by B_L, it is seen from (24) that

$$N_0 = \frac{B_L}{2\pi}, \tag{26}$$

when $B_L \ll f_0$.

We will see later that seemingly modest amounts of phase noise can seriously degrade coherent system performance; thus it is imperative

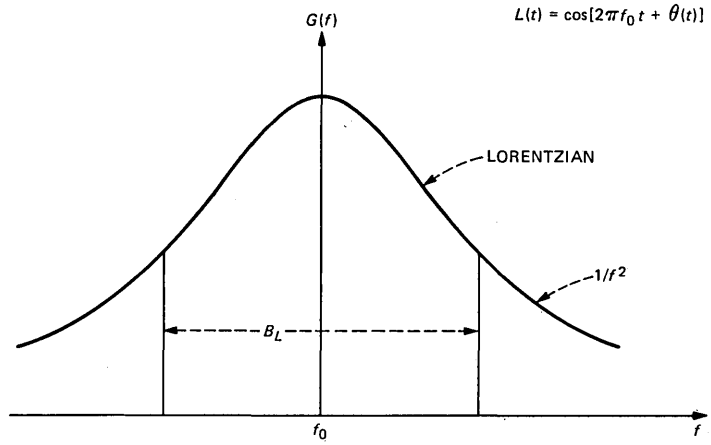


Fig. 4—Power spectrum of laser line.

to make lasers with the narrowest possible linewidth. Unfortunately, the semiconductor injection laser designs likely to be used in the 1.3- to 1.6-μm lightwave band typically have linewidths in the range 5- to 50-MHz, which is too broad for many potentially important coherent lightwave applications.[18-20] (For comparison, the reader should note that microwave oscillators, which are widely used in coherent radio applications, have linewidths on the order of 1 Hz.) To reduce laser linewidth, experimenters have exploited the fact that the noise density N_0 is inversely proportional to P_0Q^2, where P_0 is the laser output power and Q is the quality factor of the "cold" laser cavity resonance; thus high-power, high-Q lasers tend to have narrow linewidths. The most impressive line-narrowing experiments have been performed using a mirror or diffraction grating external to the laser chip to produce a composite cavity of very high Q.[21] Under relatively benign laboratory conditions, linewidths of tens of kilohertz have been obtained: an improvement of three orders of magnitude! Whether this approach will prove practical under harsh field conditions remains to be seen. In any case, it appears that phase noise will be an important consideration for the foreseeable future, so we turn now to developing a clear understanding of its consequences.

IV. PHASE-LOCK TECHNIQUES

We saw in Section III that homodyne detection of an optical PSK data wave makes it possible to achieve the quantum limit. However, to gain the full benefit of this approach the local laser must have perfect knowledge of the transmitted optical center frequency and phase. In this section we explore the possibility of deriving these

crucial parameters from either the optical data wave directly or from a heterodyned version.[22] At microwave frequencies, carrier recovery techniques are well established, while at optical frequencies, the available methodologies are still limited. For example, it is very difficult to directly multiply two optical signals or square an optical wave, which makes it difficult to wipe off the binary modulation. In attempting to derive carrier in the optical frequencies, it is, therefore, necessary to resort to data-aided techniques.[23]

4.1 Optical phase-locked loop

The question we explore here is the possibility of estimating or tracking the phase of the incoming optical wave so that it can be used to coherently demodulate PSK. Making use only of direct intensity detection, a proposed homodyne optical detector is depicted in Fig. 5.

At the input to the detector, the power of the incoming optical data signal is split so that a fraction, A^2k^2, is devoted to the estimation of the phase process $\theta(t)$ by a phase-locked loop, and the remaining portion of the power, $A^2(1-k^2)$, is used for demodulation. The power division that is determined by the choice of the constant $0 \leq k \leq 1$ is a parameter that must be optimized. In the phase-locked loop we presume that the local Voltage Controlled Oscillator (VCO) can be

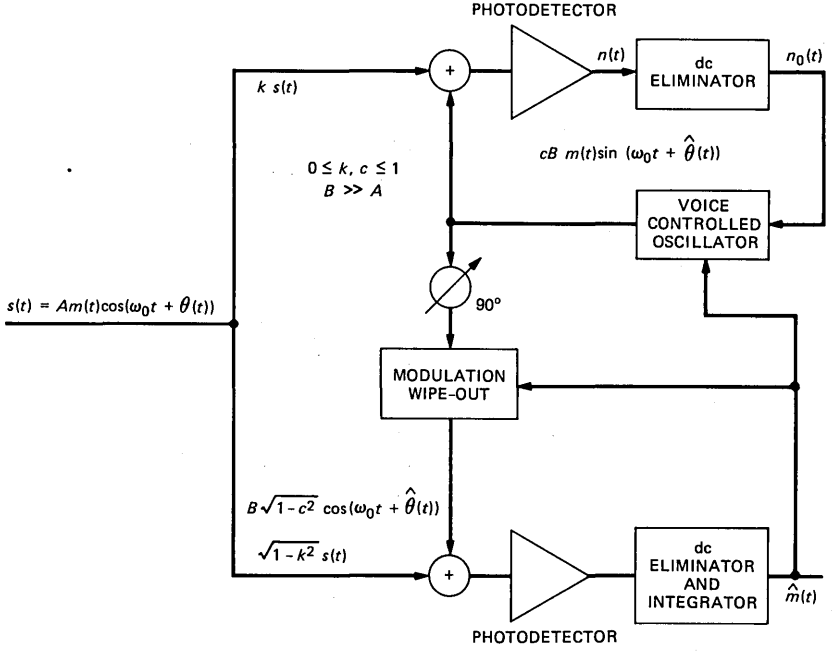


Fig. 5—An optical homodyne detector.

modulated by the estimated data $\hat{m}(t)$ using decision directed techniques, and consequently the modulation can be locally wiped off and a modulation-free carrier thus made available in the lower portion of the figure to perform the homodyne demodulation function. We also assume that the optical center frequency can be identically matched by the local laser. In any event, if this is not the case, an additional phase-locked loop may be needed to track this mismatch.[16] Making these assumptions, it is possible to analyze the performance of this detector and to obtain tight bounds on the degradation from ideal homodyne performance.

For the subsequent analysis we refer to Fig. 5. Adding a fraction c of the VCO output to a fraction of the incoming optical data wave we obtain

$$V(t) = Akm(t)\cos(\omega_o t + \theta(t)) + cB\hat{m}(t)\sin(\omega_o t + \hat{\theta}(t)), \quad (27)$$

where $\hat{m}(t)$ at the output of the VCO is the reconstructed data wave from past decisions, and it is assumed to be devoid of errors. The squared envelope of $V(t)$, which is the average value of the Poisson counting process $n(t)$, at the output of the photodetector is

$$E^2(t) = A^2k^2 + B^2(1 - c^2) + 2ABkc \sin \psi(t), \quad (28)$$

where $\psi(t)$ is the phase-error process

$$\psi(t) = \theta(t) - \hat{\theta}(t), \quad (29)$$

and we used the fact that $\hat{m}(t)m(t) = m^2(t) = 1$.

After dc elimination, the "signal" portion of the "shot noise" process, which in the limit becomes a Gaussian process since B can be made large is

$$2ABk \cdot c \cdot \sin \psi(t), \quad (30)$$

and the resulting zero-mean white Gaussian noise process is denoted by $\nu(t)$ with spectral density equal to B^2c^2.

Thus the equivalent signal-plus-noise process, which controls the frequency of the local laser VCO is

$$u(t) = 2ABkc \sin \psi(t) + \nu(t), \quad (31)$$

and so, because of feedback, we must satisfy the equation

$$\hat{\theta}(t) = \phi(t) + K \int_0^t u(t')dt', \quad (32)$$

where $\phi(t)$ is the phase noise process of the local laser, and K is a proportionality constant to be determined later.

Subtracting from both sides of (32), $\theta(t)$, the phase of the incoming wave, and differentiating, we get the stochastic differential equation

$$\frac{d\psi}{dt} = -\frac{d}{dt}[\theta(t) - \phi(t)] + K[2ABkc \sin \psi(t)] + K\nu(t)$$

$$= K_0 \sin \psi(t) + \nu_0(t), \tag{33}$$

where

$$K_0 = 2ABKkc \tag{34}$$

and $\nu_0(t)$ is now a white Gaussian process with double-sided spectral density,

$$D = B^2K^2c^2 + 2\pi(B_{L1} + B_{L2}), \tag{35}$$

where the Lorentzian bandwidth of the transmitter laser is B_{L1} and the local laser VCO bandwidth is B_{L2}.

It is well known[24,25] that (33) obeys a Fokker-Plank equation yielding the steady-state probability density function (mod2π) for the phase-error process $\psi(t)$,

$$p(\psi) = \frac{\exp[\alpha \cos \psi]}{2\pi I_0(\alpha)}, \qquad -\pi \leq \psi \leq \pi, \tag{36}$$

where $I_0(\cdot)$ is the zeroth-order modified Bessel function, and where

$$\alpha = \frac{2K_0}{D} = \frac{4AkK_e}{K_e^2 + 2\pi B_L}, \tag{37}$$

$$K_e = BK\sqrt{1 - c^2}, \tag{38}$$

and

$$B_L = (B_{L1} + B_{L2}).$$

The probability density (36) is sharply peaked at $\psi = 0$ when α is large and becomes flat, or uniform when α is small. One strives therefore to design the phase-locked loop so that α is as large as needed to obtain minimum degradation from ideal ($\psi = 0$). For fixed A and k, (37) reveals that α cannot be made arbitrarily large because of the finite Lorentzian bandwidth B_L. However, there exist a maximum value of α (when $K_e^2 = 2\pi B_L$) given by,

$$\alpha_0 = \frac{2Ak}{\sqrt{2\pi B_L}} = k\sqrt{\frac{2PR}{\pi B_L}} \tag{39}$$

where $P = A^2T$—the transmitted optical energy in the received signal. Equation (39) reveals that for fixed optical energy and k, α_0 can be made large only by increasing the ratio R/B_L.

4.2 Performance

We now examine the performance of this optical homodyne detector using the phase-locked loop output as the reference carrier wave.

In the lower portion of Fig. 5 the sum signal $W(t)$ is

$$W(t) = A\sqrt{1-k^2}\, m(t)\cos(\omega_0 t + \theta(t)) + B\sqrt{1-c^2}\cos(\omega_0 t + \hat{\theta}(t)). \quad (40)$$

The squared envelope of $W(t)$—the response of the photodetector—is therefore

$$E^2(t) = A^2(1-k^2) + B^2(1-c^2) + 2AB\sqrt{1-k^2}\sqrt{1-c^2}\, m(t)\cos\psi(t). \quad (41)$$

The resulting shot-noise process at the output of the photodetector again becomes in the limit a Gaussian process with average value (after dc elimination) equal to

$$2AB\sqrt{1-k^2}\sqrt{1-c^2}\, m(t)\cos\psi(t), \quad (42)$$

and zero-mean white Gaussian noise $\nu(t)$ with a spectrum equal to

$$B^2(1-c^2).$$

Thus the equivalent signal plus noise prior to integration is

$$S(t) = 2A\sqrt{1-k^2}\, m(t)\cos\psi(t) + \nu_0(t), \quad (43)$$

where we have divided signal plus noise by $B\sqrt{1-c^2}$ thus normalizing the spectrum of $\nu_0(t)$ to unity.

Integrating (43) over a T-second interval results in the decision statistic

$$s_0 = \pm 2A\sqrt{1-k^2}\int_0^T \cos\psi(t)dt + \int_0^T \nu_0(t)dt, \quad (44)$$

or

$$s_0 = \pm\rho\xi + \bar{\nu}_0, \quad (45)$$

where $\bar{\nu}_0$ is now a zero-mean Gaussian random variable with unit variance, the random variable

$$\xi = \frac{1}{T}\int_0^T \cos\psi(t)dt,$$

and $\rho = 2AT^{1/2}\sqrt{1-k^2}$.

Because of symmetry, the probability of error is

$$Pe = \Pr[-\rho\xi + \bar{\nu}_0 \geq 0]. \quad (46)$$

The exact evaluation of this probability is intractable even numer-

ically since it requires knowledge of the nth-order probability distribution of the phase error process $\psi(t)$. The most we have, however, is the first-order distribution, and therefore we must resort to upper bounds using the only information we have.

In Appendix A, an exponential upper bound on (46) is developed with the result

$$Pe \leq \begin{cases} g(\alpha)e^{-\frac{\rho^2}{2}}, & \alpha/\rho^2 > 1 \\ g(\alpha)e^{-\rho^2/2[2\alpha/\rho^2-(\alpha/\rho^2)^2]}, & \alpha/\rho^2 \leq 1, \end{cases} \tag{47}$$

where the coefficient, $g(\alpha)$, is defined in Appendix A, eq. (178).

When $\alpha/\rho^2 = 1$, (47) is reduced to a single bound,

$$Pe \leq g(\alpha)e^{-\frac{\rho^2}{2}}. \tag{48}$$

Recall that

$$\rho^2 = 4P(1 - k^2), \tag{49}$$

and

$$\alpha = k\sqrt{\frac{2PR}{\pi B_L}}, \tag{50}$$

and so the threshold parameter becomes

$$r = \frac{\alpha}{\rho^2} = \frac{1}{4\sqrt{\pi}}\sqrt{\frac{\gamma}{P}}\frac{k}{1-k^2}, \tag{51}$$

where γ is the ratio of the signaling rate to the average Lorentzian laser bandwidth

$$\gamma = \frac{2R}{B_L}. \tag{52}$$

It is seen from (47) that the error exponent assumes two crucially different forms depending on whether $r \geq 0$ or $r < 0$. For fixed γ and k, the exponent is linear in P when $r \geq 1$, while it behaves as the square root of P for $r < 1$. Thus the probability of error decays much more slowly with P when $P >$ constant γ [eq. (51)], indicating a threshold for Pe versus P. For fixed γ and P, however, r is a monotonically increasing function of k, $0 < k < 1$, and so there exists a value $k = k_o$, such that $r(k_o) = 1$, given by

$$k_o = \sqrt{x^2 + 1} - x, \tag{53}$$

where

$$x = \frac{1}{8\sqrt{\pi}} \sqrt{\frac{\gamma}{P}}.$$

At this value of k the exponent is

$$E(k_o, P) = -2P(1 - k_o^2). \tag{54}$$

This value of k is not, however, the optimum value that makes the negative exponent the greatest, or the probability of error the smallest. The optimum value of k, or the power division, is found by setting the derivative of the exponent in (47) for $r < 1$ to zero. From (47) the negative exponent is

$$E(k) = (1 - k^2)[2r(k) - r^2(k)], \tag{55}$$

and

$$\frac{dE}{dk} = 2(1 - k^2)\left[\frac{dr}{dk} - r\frac{dr}{dk}\right] - 2k[2r - r^2]. \tag{56}$$

Note that

$$\left.\frac{dE}{dk}\right|_{k=k_o} = -2k_o,$$

indicating that there exist a $k_{op} \leq k_o$ which increases the exponent. Setting (55) to zero, we get the formula for the optimum k,

$$k_{op}^2 = 1 - r(k_{op}). \tag{57}$$

Substituting (51) into (57) with the definition of x in (53) we obtain explicitly,

$$k_{op}^2 = 1 - 2x \frac{k_{op}}{1 - k_{op}^2}. \tag{58}$$

One can now proceed to solve (58) numerically and use this optimum value to plot the upper bound on the probability of error versus P for different values of γ. A more incisive way to exhibit the behavior of the error rate, however, is to use the suboptimum value of $k_o > k_{op}$ given in (53), which renders $r = 1$, and then define a penalty, which is the reduction of the exponent relative to ideal performances. Since $k_{op} < k_o$, this still provides an upper bound on the error rate. It can be seen from (58) that when the term $1/(1 - k_{op}^2)$ is neglected, the resulting equation is identical to (53) and so, to this degree of approximation, $k_{op} \sim k_o$ (this is a good approximation when k_{op} turns out to be small, which should be the case when the penalty is small). Following this approach, we solve (53) for x and write the negative exponent (54) as

$$-E(\gamma, P) = 2P(1 - k_o^2) = 4Px[\sqrt{x^2 + 1} - x]. \tag{59}$$

It can be checked that when $\gamma \to \infty$ (zero phase noise), $E(\gamma, P) \to 2P$, as it should.

The penalty incurred due to the finite value of γ may now be defined as follows. Equating (59) to $2P_o$, where P_o is chosen to achieve a desired error rate, say 10^{-9}, in which case $P_o \sim 10$, one obtains a function of P_o/P versus γ, and for each value of γ one can then calculate, $-10 \log P_o/P$), which defines the penalty.

Thus, proceeding in this manner we have,

$$2P_o = 4Px\sqrt{x^2 + 1} - x, \tag{60}$$

and when

$$x = \frac{1}{8\sqrt{\pi}} \sqrt{2/P}$$

is substituted into (60), one gets the penalty function

$$b = \frac{P_o}{P} = \frac{\gamma}{\gamma + 16\pi P_o}, \tag{61}$$

as $\gamma \to \infty$, $G = 1$.

In Fig. 6 we plot (61) versus γ for two different values of P_o. One

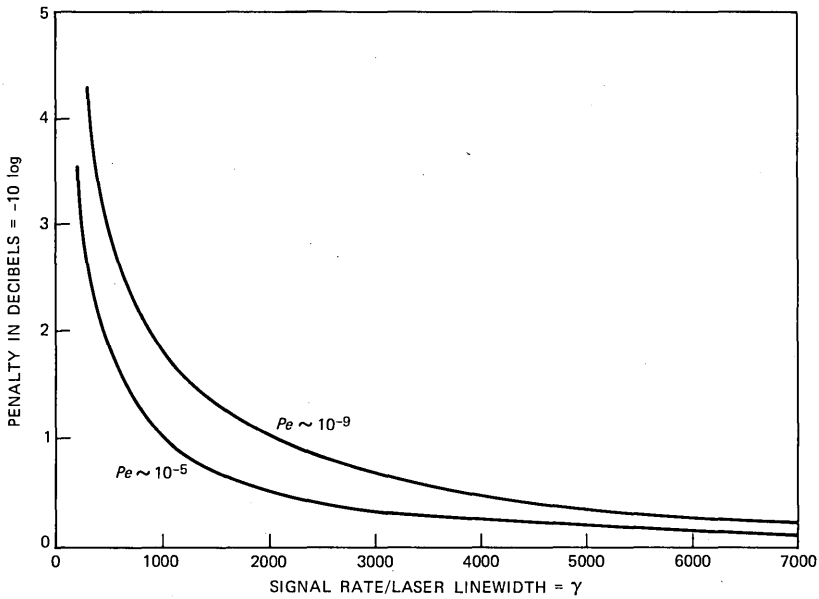


Fig. 6—Penalty versus signaling rate divided by linewidth in homodyne PSK detection at two different ideal error rates for identical laser linewidths.

corresponding to $Pe \sim 10^{-9}$ and the other to $Pe \sim 10^{-5}$. It can be seen that negligible penalty is incurred when $\gamma \geq 3000$ for $Pe \sim 10^{-9}$ and 3 dB is given up at $\gamma \sim 500$. We will see in the next section that a heterodyne Phase-Locked Loop (PLL) also suffers a 3-dB penalty at around $\gamma \sim 500$. This approach is analyzed in the next section.

4.3 Heterodyne phase-locked Loop

Heterodyning the optical data wave down to an IF frequency, and then deriving the carrier from the resulting microwave signal using standard well-known techniques may make it easier to wipe off the modulation, and consequently ease the signal processing burden of the phase-locked loops.[26,27] In the subsequent analysis we presume that the modulation has been eliminated, and as such the phase-locked loop can operate directly on the IF carrier wave. So, after heterodyning the optical signal

$$Am(t)\cos(\omega_o t + \theta(t)) \tag{62}$$

to an IF frequency f_i and wiping off the modulation, we obtain the microwave signal plus noise

$$s(t) = 2A\cos(2\pi f_i t + \theta(t)) + n(t), \tag{63}$$

where A^2 equals optical power, $n(t)$ is again a white Gaussian noise process with unit double sided spectral density, and $\theta(t)$ is now the difference between the transmitting laser's phase noise and the local laser's phase noise. Consequently, the variance of $\theta(t)$ now is

$$\begin{aligned} E\theta^2(t) &= (2\pi)^2(N_{o1} + N_{o2})t \\ &= (2\pi)\bar{B}_L t, \end{aligned} \tag{64}$$

where as before

$$B_L = (B_{L1} + B_{L2}). \tag{65}$$

The signal (63) is now the input to a conventional PLL depicted in Fig. 7. The analysis of the PLL is straightforward. Denote the output of the PLL by

$$S_v(t) = K_1 \sin\theta'(t), \tag{66}$$

where

$$\theta'(t) = 2\pi f_i t + K_2 \int_0^t e(s)ds.$$

The output of the multiplier in Fig. 6 is

$$X(t) = 2A\cos(2\pi f_i t + \theta(t))S_u(t) + S_u(t)n(t), \tag{67}$$

where the function of the Low-Pass Filter (LPF) here is to eliminate

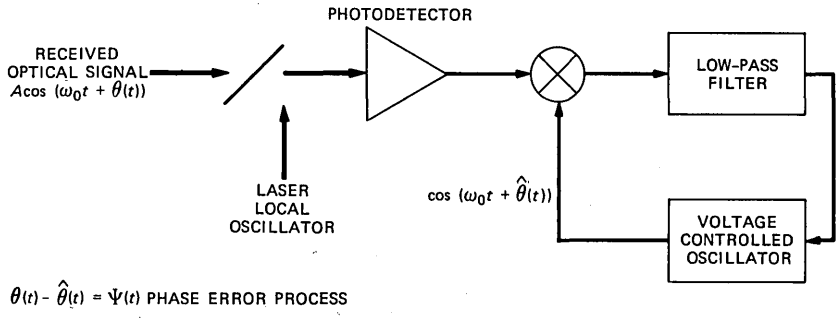


Fig. 7—Phase-lock techniques.

double-frequency terms from its output. Yet, the cut-off frequency is placed high enough so that the output noise can still be regarded as white. Thus, because of feedback, the following equation must be satisfied

$$e(t) = K_1 A \sin(\theta(t) - K_2 \int_0^t e(s)ds) + \nu(t), \tag{68}$$

where now the double-sided spectral density of $\nu(t)$ is equal to $K_1^2/2$. The phase error therefore is

$$\psi(t) = \theta(t) - K_2 \int_0^t e(s)ds, \tag{69}$$

and when this is differentiated one obtains

$$\begin{aligned} \frac{d\psi}{dt} &= \frac{d\theta}{dt} - K_2 e(t) \\ &= 2\pi(\mu_1(t) - \mu_2(t)) - K_2 e(t), \end{aligned} \tag{70}$$

where μ_1 and μ_2 are the frequency noises of the transmitter laser and the laser involved in the local heterodyner, respectively.

When (70) is substituted into (68), one again obtains the well-known stochastic differential equation governing the evolution of the phase-error process,

$$\frac{d\psi(t)}{dt} = -K_1 K_2 A \sin \psi(t), \tag{71}$$

where

$$u(t) = -2\pi(\mu_1 - \mu_2) + \nu(t)K_2 \tag{72}$$

is a white Gaussian noise process with double-sided spectrum equal to

$$D = 2\pi B_L + \frac{K_1^2 K_2^2}{2}. \tag{73}$$

Letting $K_1K_2 = K$, we observe that (71) is identical in structure to (33), which again results in a Fokker-Plank equation yielding the steady-state probability density function (mod 2π) for the phase error

$$p(\psi) = \frac{e^{\alpha}\cos\psi}{2\pi I_o(\alpha)}, \qquad -\pi \le \psi \le \pi. \tag{74}$$

The important system parameter is now given by

$$\alpha = \frac{2AK}{D} = \frac{2AK}{\frac{K^2}{2} + 2\pi B_L}. \tag{75}$$

The algebraic form of the PLL parameter α departs from the conventional form where, by decreasing the loop bandwidth, α can be made as large as possible. Here there is a minimum band resulting from the presence of phase noise. The only way that α can be increased is by increasing the input optical energy, or by decreasing phase noise relative to the signaling rate. Consequently there exists a maximum value of α (when $K^2 = 4\pi B_L$) given by

$$\alpha_{opt} = \sqrt{\frac{RP}{\pi B_L}}, \tag{76}$$

where $R = 1/T$ and $P = A^2T$—optical energy.

4.4 Performance

Now consider the modulated heterodyned wave

$$S_m(t) = \pm 2A\cos(2\pi f_i t + \theta(t)) + n(t), \qquad 0 \le t \le T, \tag{77}$$

and the estimated carrier wave from the PLL

$$\cos(2\pi f_i t + \hat{\theta}(t)). \tag{78}$$

Multiplying (77) by (78), integrating from 0 to T, and eliminating the double-frequency components results in the decision statistic

$$S = \pm A\int_0^T \cos\psi(t)dt + \int_0^T n(t)\cos(2\pi f_i t + \hat{\theta}(t))dt, \tag{79}$$

where $\psi(t)$ is the phase-error process obeying at any instant of time the probability density (74). Rewriting (79) as before,

$$S = \pm AT\xi + \nu, \tag{80}$$

where again

$$\xi = \frac{1}{T}\int_0^T \cos\psi(t)dt,$$

and where now ν is a Gaussian random variable with $E\nu^2 = T/2$.

Dividing (80) by $\sqrt{2/T}$ yields,

$$s_0 = \pm\rho\xi + \nu_0, \tag{81}$$

where $\rho = \sqrt{2T}A$ and ν_0 now has unit variance.

The probability of error is as before,

$$Pe = \Pr[-\rho\xi + \nu_0 \geq 0]. \tag{82}$$

This probability is structurally identical to (46), and we therefore use the same bound from Appendix A.

$$Pe \leq \begin{cases} g(\alpha_o)e^{-\rho^2/2}, & \alpha_o/\rho^2 \geq 1 \\ g(\alpha_o)e^{-\rho^2/2[2\alpha_o/\rho^2+(\alpha_o/\rho)^2]}, & \alpha_o/\rho^2 < 1 \end{cases} \tag{83}$$

With the definition of α_o in eq. (76) we can compute the threshold parameter in this case,

$$r_o = \alpha_o/\rho^2 = \frac{1}{\sqrt{8\pi}} \sqrt{\frac{\gamma}{P}}, \tag{84}$$

where again

$$\gamma = \frac{2R}{B_L}. \tag{85}$$

With these definitions we express (83) as

$$Pe \leq \begin{cases} g(\alpha)e^{-P}, & \gamma \geq 8\pi P \\ g(\alpha)e^{-P[2r_o-(r_o^2)]}, & \gamma \leq 8\pi P. \end{cases} \tag{86}$$

We note that the behavior of this bound is slightly different from the one applicable in the homodyne case. For one, here ideal performance is 3 dB inferior to the quantum limit, which is accounted for by the heterodyning operation, as we have already noted. Next, ideal performance with negligible degradation is attained when

$$\frac{2R}{B_L} > 8\pi P,$$

and for ratios less than this, the degradation is increased gracefully. As an example, when $P = 20$, yielding an ideal error rate $\sim 10^{-9}$, the threshold parameter $\gamma = 500$. For ratios greater than this, no penalty in optical power (from the ideal $P = 20$) is incurred. To assess the penalty at operations at less than this threshold, we use the same definition as before. The probability of error exponent when $r_o \leq 1$ is

$$E(\gamma, P) = P(2r_o - r_o^2) \tag{87}$$

and when this is equated to P_o we get the formula

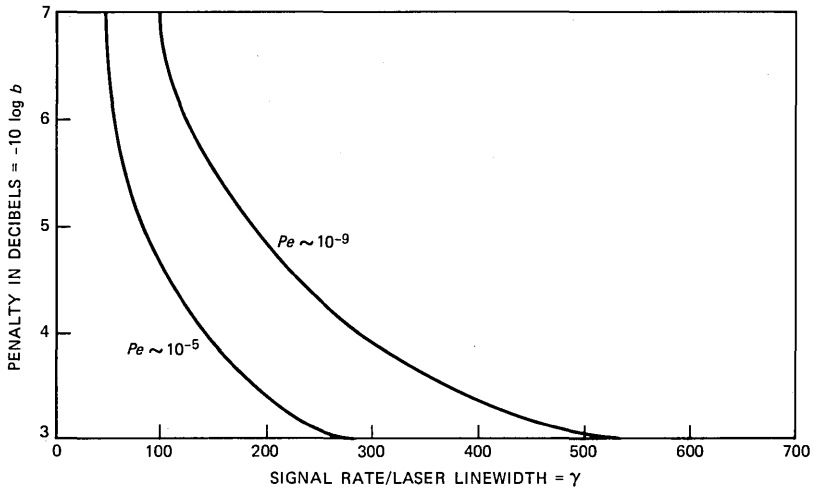


Fig. 8—Penalty versus signaling rate divided by linewidth in heterodyne PSK detection at two different ideal error rates for identical laser linewidths.

$$b = \frac{P_o}{P} = 2r_o - r_o^2 \leq 1. \tag{88}$$

Substituting the definition of r_o [eq. (84)] and solving for γ_o yields

$$\gamma = 8\pi P_o \left[\frac{(1 - \sqrt{1 - b})^2}{b}\right], \tag{89}$$

as can be seen when $b = 1$, $\gamma = 8\pi P_o$—the threshold value.

In Fig. 8 we plot b in decibels versus γ for two different values of P_o. We see that around $\gamma \sim 400$, the asymptotic degradation of 3 dB is approached. Interestingly, at around this ratio the optical PLL also degrades by 3 dB, as we have already mentioned.

In concluding this section we remark that our estimates are only upper bounds, albeit, we feel, tight bounds. The exact evaluation of the error rate is not feasible because of the nonlinear functionals that are involved. If one chooses to ignore the time integrals, than the probability of error can be evaluated numerically as has been done in Refs. 28 and 29.

In the next section we will see that noncoherent techniques such as frequency modulation and differential phase modulation, where knowledge of carrier phase is not essential, yields performance very near to what can be attained with heterodyne phase-lock technique at reasonable signaling rates.

V. BINARY FREQUENCY SHIFT KEYING

Here we discuss and analyze the performance of binary Frequency

Shift Keying (FSK) as one of the modulation options.[30–33] In this modulation method information is conveyed by switching the frequency of the laser between two different values. Thus, during a fixed interval, $T = 1/R$, where R is the signaling rate, the receiver has to decide whether

$$A \cos(\omega_1 t + \theta(t)) \tag{90}$$

or

$$A \cos(\omega_2 t + \theta(t))$$

was transmitted. In (90) A^2 is again proportional to the received optical power and $\theta(t)$ is the phase noise process associated with the laser.

It appears that this modulation method is impervious to the effects of phase noise, since the shifted frequencies ω_1 and ω_2 can be sufficiently separated and if enough bandwidth is available, crosstalk due to the fluctuating phase noise can be minimized. These are the chief reasons, then, for considering FSK. We point out that this is not what is commonly referred to as continuous-phase narrow-band FM. The latter yields the same performance as differential phase modulation treated in the next section.

The first step in the processing of the FSK optical signal is to heterodyne (90) to an IF frequency ω_i. As was already noted, this can be accomplished by adding to (90) a locally generated optical signal and then direct detecting the sum by a photodetector. The sum signal then is

$$S(t) = A \cos(\omega_l t + \theta(t)) + B \cos(\omega_l + \phi(t)), \qquad l = 1, 2, \tag{91}$$

where the IF frequencies are $\omega_i = \omega_l - \omega_f$, and $\phi(t)$ is the phase noise associated with the local laser. The output of the photodetector is again a "shot noise" process

$$I(t) = \sum_n w(t - t_n). \tag{92}$$

The squared envelope of (91) with respect to ω_f is

$$E^2(t) = A^2 + B^2 + 2AB \cos((\omega_l - \omega_f)t + \Delta(t)), \tag{93}$$

where $\Delta(t) = \theta(t) - \phi(t)$.

When the local laser intensity $B \gg A$ (66) approaches a white Gaussian process with average value equal to $\lambda(t)$ and standard deviation also equal to $\lambda(t)$. Thus the dc part of the average value of (93) is

$$S_o(t) = 2AB \cos(\omega_i t + \Delta(t)), \qquad i = 1, 2, \tag{94}$$

while the resultant zero-mean Gaussian noise, $n(t)$, has a double-sided spectral density equal to B^2.

With these preliminaries, we now confront a classical detection

problem. Given the IF signal (94), plus white Gaussian noise of unit spectral density

$$V(t) = 2A\cos(\omega_i t + \Delta(t)) + n(t), \qquad 0 \le t < T, \qquad i = 1, 2, \quad (95)$$

how does one process $V(t)$ so as to attain the least probability of error? While the problem is classical the solution is not tractable in general because of the presence of the phase noise process $\Delta(t)$.

For calibration purposes, let us first review briefly the performance under the assumption that the phase noise is effectively a constant. In the case when $\Delta(t)$ is slowly varying with respect to the rate $R = 1/T$, or when the symbol rate R is much greater than the bandwidth of the laser signal, the optimum detector has a well-known structure depicted schematically in Fig. 9. The results in this case will serve as a benchmark to which the later more general results will be compared. Also assume that the frequency shifted signals are orthogonal, i.e., the two frequencies ω_1 and ω_2 are chosen such that

$$\int_0^T \cos(\omega_1 t + \Delta)\cos(\omega_2 t + \Delta)dt = 0.$$

To proceed with the error rate analysis, assume that ω_1 was sent. In this case, we expect the x output in Fig. 8 to be greater than y, and an error is made when $x - y \le 0$. Because of symmetry, when ω_2 is sent, y is expected to be greater than x and a mistake is now made when $y - x \ge 0$. So, the probability of error is just

$$Pe = \Pr[x - y \le 0]. \quad (96)$$

In order to evaluate this probability, we express the random variables x and y as indicated by the mathematical operations in Fig. 7. It then can be seen that

$$Pe = \Pr[x_1^2 - x_2^2 + x_3^2 - x_4^2 \le 0], \quad (97)$$

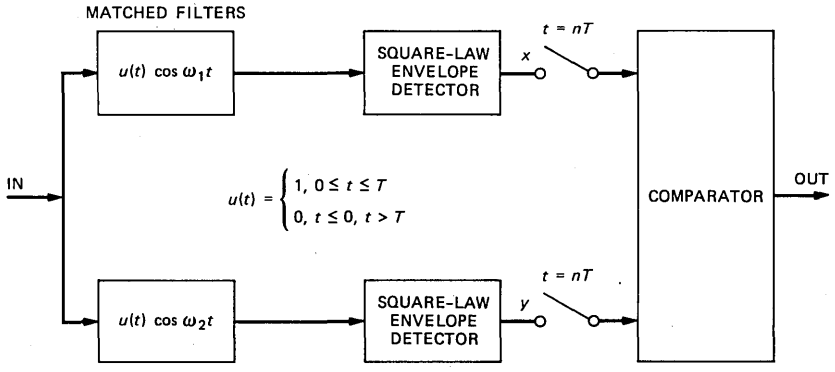


Fig. 9—Structure of the optimum FM detector when linewidth approximates zero.

where

$$x_1 = AT\cos\Delta + \int_0^T n(t)\cos\omega_1 t dt,$$

$$x_2 = \int_0^T n(t)\cos\omega_2 t dt,$$

$$x_3 = AT\sin\Delta + \int_0^T n(t)\sin\omega_1 t dt,$$

and

$$x_4 = \int_0^T n(t)\sin\omega_2 t dt.$$

The desired probability is just the probability that the difference in the lengths of two 2-dimensional Gaussian vectors is less than zero. As can be verified, the x's are independent Gaussian random variables with identical variances, $\sigma^2 = T/2$. For these random variables, (97) can be expressed exactly[34] as

$$Pe = \frac{1}{2}\exp\left[-\frac{A^2T}{2}\right]. \tag{98}$$

Comparing this with the performance of direct detection, we observe that the optical signal (90), after direct detection and integration for T seconds, yields an average photon count equal to A^2T. In this direct detection case, the chance of making an error would be the chance of detecting zero photons in T seconds. From the Poisson distribution, for the number of photons detected, this probability is just $1/2\exp[-A^2T]$, which is 3 dB worse than the quantum limit. Comparing this with (98), however, reveals an additional 3-dB loss due to heterodyning. So, heterodyne FSK detection is 6 dB inferior to the so-called "quantum limit" provided that phase noise can be neglected.

Returning now to the more realistic situation where phase noise is present and must be included in the performance analysis, we recall that the crucial assumption for the previous analysis was that the symbol rate baud $R = 1/T$ be much greater than the linewidth of the laser, so that the phase process $\Delta(t)$ could be regarded as a constant during the integration period. The inclusion of the phase noise process immediately raises fundamental problems concerning the detector structure.

As is well known,[23] the optimal processor in the presence of phase noise first estimates the phase process, and then uses this estimate to coherently demodulate or detect the IF signal. This is precisely what a PLL does, but we have seen that coherent PLL detection becomes

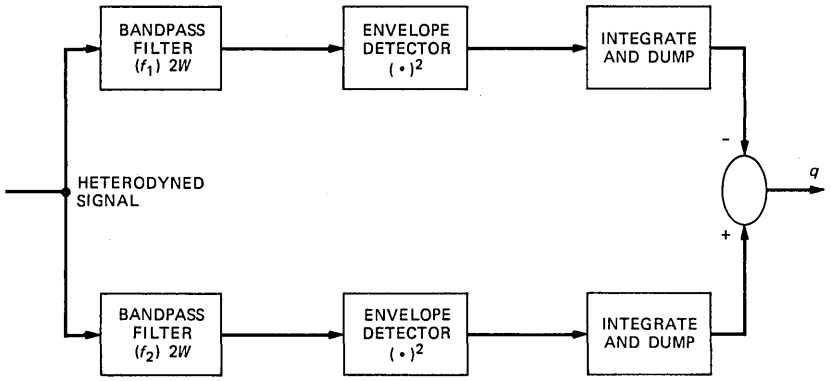


Fig. 10—FSK detector.

feasible only at very high data rates. This situation leads us to postulate a detector that, while not optimum, is reasonable and does not require a phase-locked loop.

The proposed frequency detector structure is shown in Fig. 10. It is essentially an energy detector. It consists of two ideal bandpass filters of total bandwidth equal to $2W$. The purpose of these filters is to limit the added white noise bandwidth as much as possible, while at the same time to retain most of the energy in the information carrying signal. A precise number for the bandwidth that satisfies these two seemingly contradictory requirements is hard to derive because, strictly, the received sine wave with phase noise has infinite bandwidth. The front-end bandwidth must remain as a parameter in our subsequent analysis, and engineering estimates will be attempted later.

Following these bandlimiting filters with square-envelope detectors and an integrator essentially provides an estimate of the energy in each frequency band, and this quantity should be independent of phase noise provided that the front-end bandwidth is sufficiently large. We now proceed to analyze the performance of this structure.

In the representation of signal plus noise, eq. (95), assume that ω_1 was sent. Regarding ω_1 as the center frequency, we represent the signal and noise in terms of in-phase and quadrature components as

$$\begin{aligned} S(t) &= 2A \cos \omega_1 t \cos \Delta(t) - 2A \sin \omega_1 t \sin \Delta(t) \\ &\quad + n_1(t) \cos \omega_1 t + n_2(t) \sin \omega_1 t \\ &= x(t) \cos \omega_1 t + y(t) \sin \omega_1 t, \end{aligned} \tag{99}$$

where

$$x(t) = 2A \cos \Delta(t) + n_1(t),$$

and

$$y(t) = 2A \sin \Delta(t) + n_2(t).$$

At the output of the bandpass filters $x(t)$ and $y(t)$ are bandlimited versions of the input. As we have already stated the signals $x(t)$ and $y(t)$ remain essentially undistorted at the output, and the only effect of the bandpass filters is to limit the noise band. Clearly, as the band W increases, this approximation becomes better.

The output baseband noises with unit input noise intensity now have mean-square values

$$En^2 = En_1^2 = En_2^2 = 4W. \tag{100}$$

Squaring the envelope and integrating as indicated in Fig. 8 yields the quadratic decision statistic q_1,

$$q_1 = \int_0^T [x^2(t) + y^2(t)]dt, \tag{101}$$

where for a given $\Delta(t)$, $x(t)$ and $y(t)$ are independent Gaussian processes with

$$Ex(t) = 2A \cos \Delta(t),$$

and

$$Ey(t) = 2A \sin \Delta(t). \tag{102}$$

The covariance functions of the ideally bandlimited baseband noise processes $n_1(t)$ and $n_2(t)$ are

$$En_1(t)n_1(t + \tau) = En_2(t)n_2(t + \tau) = 4W \frac{\sin 2\pi W\tau}{2\pi W\tau}. \tag{103}$$

As is well known,[35] associated with this covariance kernel are an infinite set of orthonormal eigenfunctions $\{\psi_k(t)\}$ and a set of nonnegative eigenvalues $\{\lambda_k\}$. Using these eigenfunctions, we represent the processes $x(t)$ and $y(t)$ as

$$\begin{pmatrix} x(t) \\ y(t) \end{pmatrix} = \sum_k \psi_k(t) \begin{pmatrix} x_k \\ y_k \end{pmatrix}, \tag{104}$$

where

$$\begin{pmatrix} x_k \\ y_k \end{pmatrix} = \int_0^T \psi_k(t) \begin{pmatrix} x(t) \\ y(t) \end{pmatrix} dt,$$

and

$$Ex_k y_{k'} = \lambda_k \delta_{kk'}.$$

Using these expansions, the quadratic form q_1 in (101) can be put into the form

$$q_1 = \int_0^T [x^2(t) + y^2(t)] = \sum_k (x_k^2 + y_k^2). \tag{105}$$

In the bottom leg of Fig. 10 we assume that only the "noise" goes through (by previous hypothesis), and so the resulting quadratic form q_2 is now comprised only of noise. Of course, this is a mild assumption since ω_1 can be separated from ω_2 as much as one wishes to provide minimum leakage. Thus,

$$\begin{aligned} q_2 &= \int_0^T (\nu_1^2(t) + \nu_2^2(t))dt \\ &= \sum_k (\nu_{1k}^2 + \nu_{2k}^2), \end{aligned} \tag{106}$$

where $\nu_1(t)$ and $\nu_2(t)$ are quadrature and in-phase bandlimited noises in the lower leg, and are independent of the noises in the upper leg since the spectra occupy nonoverlapping frequency bands. For this reason we denote these noises by $\nu_1(t)$ and $\nu_2(t)$ to distinguish them from n_1 and n_2 in the upper leg. The decision statistic is the difference of the quadratic forms (105) and (106) and consequently the probability of error is

$$Pe = \Pr[q = q_1 - q_2 \leq 0]. \tag{107}$$

This can be expressed in terms of the characteristic function of q,

$$C(\omega) = Ee^{i\omega q}, \tag{108}$$

as the integral[34]

$$Pe = -\frac{1}{2\pi i} \int_{-\infty}^{\infty} \frac{C(\omega)}{\omega + i\epsilon}\, d\omega. \tag{109}$$

The $i\epsilon$, $\epsilon > 0$ in the denominator denotes the fact that in the complex ω plane the contour of integration goes above the singularity at $\omega = 0$.

Using (105) and (106) it is straightforward to calculate (108)

$$\begin{aligned} C(\omega) &= Ee^{i\omega\sum_k (x_k^2+y_k^2)} Ee^{-i\omega\sum_k (\nu_{1k}^2+\nu_{2k}^2)} \\ &= E_{\Delta(t)} \left\{ \frac{1}{\prod_k (1 - 2i\omega\lambda_k)} \frac{1}{\prod_k (1 + 2i\omega\lambda_k)} e^{i\omega\sum_k \left(\frac{\bar{x}_k^2 + \bar{y}_k^2}{1 - 2i\omega\lambda_k}\right)} \right\}, \end{aligned} \tag{110}$$

where

$$\begin{pmatrix} \bar{x}_k \\ \bar{y}_k \end{pmatrix} = \int_0^T \psi_k(t) \begin{pmatrix} 2A\cos\Delta(t) \\ 2A\sin\Delta(t) \end{pmatrix} dt,$$

and $E_{\Delta(t)}(\cdot)$ denotes the expectation with respect to the phase process $\Delta(t)$. To proceed further, we invoke an excellent approximation[35] regarding the behavior of the eigenvalues $\{\lambda_n\}$ in this application. Since these are the eigenvalues of the Prolate-Spheroidal wave functions, it is shown in Ref. 35 that

$$\lambda_k \sim \begin{cases} 2, & k \leq n = 2WT \\ 0, & k > n = 2WT, \end{cases} \tag{111}$$

and when this approximation is applied in (110), we get for the characteristic function

$$C(\omega) = \frac{\exp\left(\dfrac{i\omega 4A^2T}{1 - 4i\omega}\right)}{(1 - 4i\omega)^n(1 + 4i\omega)^n}, \tag{112}$$

where $n = 2WT$.

Substituting (112) into (109) we obtain

$$Pe = \frac{-1}{2\pi i}\int_{-\infty}^{\infty} \frac{d\omega}{\omega + i\epsilon} \frac{e^{\frac{Pi\omega}{1-i\omega}}}{(1 - i\omega)^n(1 + i\omega)^n}, \tag{113}$$

where $P = A^2T$ and we set $4\omega \rightarrow \omega$. By letting $i\omega = z$ we write (113) as a contour integral

$$Pe = -\frac{1}{2\pi i}\int_{-i\infty}^{i\infty} \frac{dz}{z} \exp\left(\frac{Pz}{1 - z}\right) \frac{1}{(1 - z)^n(1 + z)^n}, \tag{114}$$

where the indentation is now to the left around the origin.

We note the nth order pole at $z = -1$ and the essential singularity at $z = 1$. When the time-bandwidth product $n = 1$, the contour can be closed in the left-hand plane, and the value of the integral is just the residue of

$$f_1(z) = \frac{e^{P\frac{z}{1-z}}}{(1 - z)(1 + z)}\left(-\frac{1}{z}\right) \tag{115}$$

at $z = -1$. This gives for the probability of bit error

$$Pe(n = 1) = \frac{e^{-\frac{P}{2}}}{2}. \tag{116}$$

This result is identical to (98) and a moments reflection will reveal the reason for the consistency. Note that roughly

$$n = 2TW \sim \frac{1}{R}(R + kB_L) = 1 + \frac{B_L k}{R}, \tag{117}$$

where $R + kB_L = 2W$—the bandpass bandwidth of the incoming signal. This band must equal to or be greater than the signaling rate R, plus kB_L—the bandwidth required to pass the sine-wave signal with the phase noise undistorted (k is a positive integer), and again B_L is the laser linewidth. Clearly, when $B_L/R \ll 1$, no postintegration is required and consequently $n = 1$. Therefore in this case we get the previous result where we regarded the phase noise as a constant—precisely the case when $kB_L/R \ll 1$. When this condition is not necessarily satisfied and so $n \neq 1$, we can still evaluate the contour integral. In this general case, closing the contour in the left-hand plane enclosing the nth order pole gives for (113) the residue and hence the probability of bit error as a function of P and n,

$$Pe(P, n) = -\frac{1}{(n-1)!}\left[\frac{d^{n-1}}{dz^{n-1}}\left(\frac{\exp\left(\frac{Pz}{1-z}\right)}{z(1-z)}\right)\right]_{z=-1}. \tag{118}$$

We show in Appendix B that (118) can be expressed more explicitly as

$$Pe(P, n) = P_n(P/2)\exp\left(-\frac{P}{2}\right), \tag{119}$$

where $P_n(P/2)$ is an nth order polynomial in $(P/2)$ with properties

$$P_1(P/2) = 1/2$$

and

$$\lim_{n\to\infty} P_n(P/2) = \frac{1}{2}\exp\left(\frac{P}{2}\right).$$

As can be seen from (119), the degradation due to excess noise and postintegration manifests itself only in an algebraic coefficient in the bit-error-rate expression and not in the exponent.

From the foregoing analysis we present the curves of Fig. 11 that are plots of Pe versus P in decibels for different values of γ, which is twice the ratio of symbol rate to the sum of laser linewidths. The front-end bandwidths around frequencies f_1 and f_2 in Fig. 9 were selected to be

$$2W = R + 10(B_{L1} + B_{L2}), \tag{120}$$

where B_{L1} and B_{L2} are the laser linewidths, $k = 10$ in (117). The factor

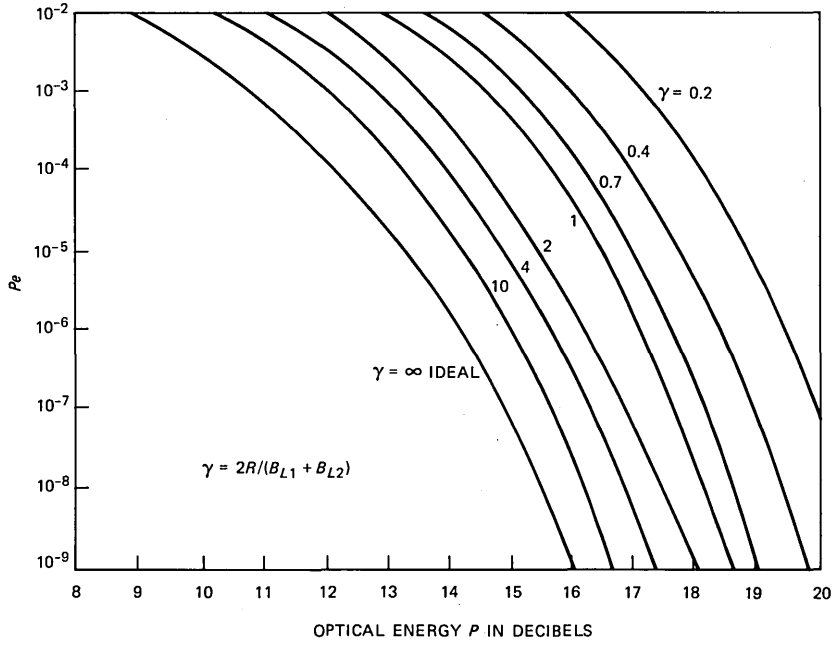


Fig. 11—FSK performance.

of 10 times the laser linewidths is judged to be adequate to pass the incoming FM signal without appreciable distortion.

When $\gamma \rightarrow \infty$, $Pe \rightarrow \exp\{-P/2\}$, which is the ideal binary FM performance. As already mentioned, this ideal performance is still 6 dB worse than the quantum limit, since 3 dB is lost from heterodyning and 3 additional decibels is lost due to the fact that the heterodyned FM signals are orthogonal rather than antipodal.

Fig. 11 can be used to determine the minimum data rate for efficient performance. Suppose the lasers have identical linewidths of 10 MHz, and one wishes the degradation not to exceed 1 dB. What is the minimum admissible data rate R? From Fig. 11, less than 1-dB degradation yields $\gamma \sim 10$, implying $R \geq 100$ Mb/s.

What is the degradation if one desires to transmit at 20 Mb/s? For the same laser linewidths as before, eq. (98) yields $\gamma = 2$, and from the figure we see that this value of γ yields a degradation of 2 dB.

As is well known, FSK can accommodate more than two frequencies to convey digital information without appreciably altering the form of the error rate expression provided again that bandwidth expansion is not an obstacle. For example, consider using 2^m frequences $m \geq 2$. Generalizing the structure of Fig. 10 to 2^m legs yields a probability of

error that is 2^m times the binary error rate but requires only $P/2m$ photons per bit to deliver the same amount of information as in the binary case. To cite an example, using four frequencies, $m = 2$, the information rate is $2/T$ and so T can be doubled to obtain the same bits per second as when $m = 2$. This comes at a moderate increase of bandwidth and a factor-of-4 increase in the error rate.

Let us consider the previous example where the binary rate $R = 100$ Mb/s. This can be achieved with ~40 photons per bit resulting in an error rate $\sim 10^{-9}$. Suppose one had only 20 photons per bit to expand, how can this data rate be accommodated without increasing the error rate? Suppose we half the signaling rate so that the new T equals twice the old T. To maintain the same rate in bits per second we must use four frequencies rather than two. The new signaling rate has now been halved and so we must recalculate γ from eq. (98) corresponding to $R = 50$. We find it to be five. From Fig. 11 we estimate that this value of γ results in a ~1-dB loss. The upshot is that FSK with four frequencies signaling at a data rate of 100 Mb/s can be achieved at a 4-dB increase of optical power over the quantum limit.

Next we analyze differential phase shift keying.

VI. DIFFERENTIAL PHASE SHIFT KEYING

In the presence of additive white Gaussian noise, Differential Phase Shift Keying (DPSK) is known to be very efficient in terms of the s/n required to achieve an acceptable error rate.[36-38] It is only a fraction of a decibel less efficient than coherent phase shift keying—the most efficient known method. Our objective here is to investigate the performance of this modulation method in optical communications and to assess the incurred penalty due to phase noise.

We begin our treatment by first considering detection at optical frequencies and then analyze the heterodyned version. Processing at optical frequencies may be practically inhibited because of the present lack of efficient (noise free) amplifiers, and therefore we also analyze the heterodyned version and compare performance of these two different approaches.

6.1 Optical processing

In DPSK, information is conveyed by the phase differences in two consecutive signaling intervals. Thus when the optical signal in a particular signaling interval is

$$S_0(t) = a_0 A \cos(\omega_0 t + \theta(t)), \qquad 0 \le t \le T, \tag{121}$$

and in the previous interval it was

$$S_{-1}(t) = a_{-1} A \cos(\omega_0 t + \theta(t)), \qquad -T \le t < 0, \tag{122}$$

where a_0, a_{-1} are ± 1 and $\theta(t)$ is again the phase noise process, information then is conveyed by the product $a_0 a_{-1}$.

Ideally one would measure the time average* of the phase difference in (121) and (122) during the interval $[0, T]$ and decide that $a_{-1}a_0 = 1$ if it lies between $-\pi/2$ and $\pi/2$ and $a_{-1}a_0 = -1$ if it lies outside of this phase range. Consequently, in a practically implemented system a lower bound on the bit error rate would be

$$Pe \geq \Pr\left[\frac{1}{T}\int_0^T [\theta(t) - \theta(t-T)]dt \geq \pi/2\right]$$

$$\sim \exp\left\{-\left(\frac{\pi}{2}\right)^2 \frac{1}{2\sigma^2}\right\}, \tag{123}$$

where

$$\sigma^2 = E\left(\frac{1}{T}\int_0^T [\theta(t) - \theta(t-T)]dt\right)^2$$

$$= 2/3(2\pi)^2 N_0 T = \frac{4\pi}{3}\frac{B_L}{R},$$

and B_L is again the full 3-dB linewidth of the laser.

From this lower bound on bit error rate the ratio B_L/R is determined, setting a floor for the minimum admissible rate in terms of B_L. For example, if one desires an error rate $\sim 10^{-9}$, one equates

$$\left(\frac{\pi}{2}\right)^2 \frac{1}{2\sigma^2}$$

to 20 to obtain

$$R \sim 67 B_L.$$

As an example, for a $B_L = 20$ MHz, R would have to be greater than 1.34 Gb/s to achieve $Pe \sim 10^{-9}$.

The problem, however, is that in the optical processing repertory the phase differential cannot yet be obtained directly nor can two optical waves be multiplied directly. Therefore, one must consider a detection method that provides information about the phase difference in an indirect manner. Thus consider the following approach: first, the incoming signal is delayed by T seconds, then half of the power of the delayed signal plus and minus half of the power of the undelayed versions are passed through separate photodetectors. A schematic representation of this phase detector is shown in Fig. 12.

* I am indebted to Leonid G. Kazovsky for pointing out the time average of the phase noise differential provides a better lower bound than just the instantaneous value that appeared in my original manuscript.

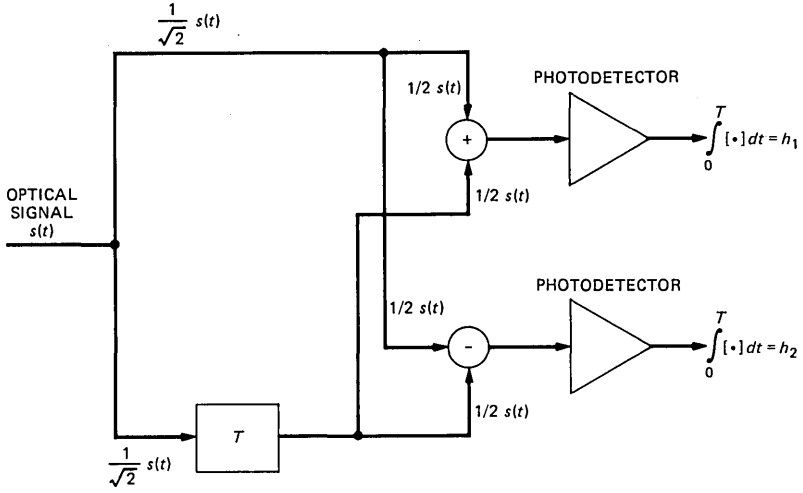


Fig. 12—Optical DPSK detector.

Referring to this figure, we write the sum and difference as

$$V(t) = \frac{a_0 A}{2}\cos(\omega_0 t + \theta(t)) \pm \frac{a_{-1}A}{2}\cos(\omega_0 t + \theta(t-T))$$

$$= \frac{a_0 A}{2}\cos(\omega_0 t + \Delta\theta(t) + \theta(t-T))$$

$$\pm \frac{a_{-1}A}{2}\cos(\omega_0 t + \theta(t-T)$$

$$= \frac{a_0 A}{2}\cos(\Delta\theta(t))\cos(\omega_0 t + \theta(t-T))$$

$$- \frac{a_0 A}{2}\sin(\Delta\theta(t))\sin(\omega_o t + \theta(t-T))$$

$$\pm \frac{a_{-1}A}{2}\cos(\omega_o t + \theta(t-T)), \tag{124}$$

where we set

$$\Delta\theta(t) = \theta(t) - \theta(t-T).$$

The output pulse count from the "sum" photodetector is a doubly stochastic Poisson process n_1 with conditional intensity λ_+ equal to the squared envelope of (124),

$$\lambda_+ = \frac{A^2}{2}(1 + a_0 a_{-1}\cos\Delta\theta(t)), \tag{125}$$

while the "difference" photodetector output is also a Poisson process n_2 independent of n_1 but having intensity

$$\lambda_- = \frac{A^2}{2}(1 - a_0 a_{-1}\cos\,\Delta\theta(t)). \tag{126}$$

Integrating the outputs of the two photodetectors for T seconds yields two average random counts

$$\Lambda_+ = \frac{A^2}{2}\int_0^T (1 + a_0 a_{-1}\cos\,\Delta\theta(t))dt$$

and

$$\Lambda_- = \frac{A^2}{2}\int_0^T (1 - a_0 a_{-1}\cos\,\Delta\theta(t))dt. \tag{127}$$

The detection statistic is the difference between the two counts,

$$n = n_1 - n_2. \tag{128}$$

When $n > 0$, $a_0 a_{-1}$ is taken to be 1, while if it is less than zero, $a_0 a_{-1}$ is taken to be -1.

Because of phase noise, the exact evaluation of the bit error rate is not mathematically tractable, but an exponential upper bound can be obtained from the moment generating function of the differential count, n. The moment generating function of n, conditioned on θ and $a_0 a_{-1}$, is readily calculated from the Poisson distribution

$$M_n(s \mid \theta, a_0 a_{-1}) = e^{\Lambda_+(e^s-1)+\Lambda_-(e^{-s}-1)}, \tag{129}$$

and will be used to upper bound the probability of error.

We begin by writing the probability of error

$$Pe = \frac{1}{2}\Pr(n \le 0 \mid a_0 a_{-1} = 1) + \frac{1}{2}\Pr(n \ge 0 \mid a_0 a_{-1} = -1), \tag{130}$$

and because of symmetry,

$$\begin{aligned} Pe = \Pr[n \le 0, \mid a_o a_{-1} = 1] &\le E_{\Delta\theta(t)} e^{\bar{\nu}(e^{-s}-1)} + \bar{u}(e^s - 1) \\ &\le E_{\Delta\theta} e^{\nu(e^{-s}-1)+u(e^s-1)}, \quad s \ge 0, \end{aligned} \tag{131}$$

where

$$\begin{aligned} \bar{\nu} &= \Lambda_+(A_o A_{-1} = 1) \\ \bar{u} &= \Lambda_-(A_o A_{-1} = 1), \end{aligned} \tag{132}$$

and

$$\nu = \frac{P}{2}(1 + \cos \Delta\theta)$$

$$u = \frac{P}{2}(1 - \cos \Delta\theta). \tag{133}$$

In the last two inequalities we used (129) in a Chernoff bound and made use of the convexity of the exponential function as well as the fact that $\Delta\theta(t)$ is stationary.

The tightest upper bound is obtained by selecting an optimum s for a given ν and u. This value of s can be obtained by setting the derivative of the exponent to zero. The set of random s's optimizing the exponent is then found to be

$$s_o = \tfrac{1}{2} \ln \frac{\nu}{u}, \tag{134}$$

and since s_o has to be positive, we require that $\nu > u$, which from (133) implies that $\cos \Delta\theta(t)$ must be positive. For values of $\Delta\theta(t)$ such that $\cos \Delta\theta \leq 0$, the optimum value of s_o is seen to be zero. Thus, in order that the bound (131) be reasonably tight, the average with respect to $\Delta\theta(t)$ indicated in (131) must be carried out over two sets of $\Delta\theta$

$$\Delta\theta \in R_1, \cos \Delta\theta \leq 0$$

and

$$\Delta\theta \in R_2, \cos \Delta\theta > 0. \tag{135}$$

This yields for (131)

$$Pe \leq E_{\Delta\theta \in R_1} e^{\nu(e^{-s}-1)+u(e^{s}-1)} + E_{\Delta\theta \in R_2} e^{\nu(e^{-s}-1)+u(e^{s}-1)}. \tag{136}$$

The first term above can be upper bounded by setting $s = 0$ and further upperbounding the probability that $\cos \Delta\theta \leq 0$, yields for this term $\Pr[\Delta\theta \geq \pi/2]$. So after some calculations and substitutions, (136) becomes

$$Pe \leq \exp\left\{-\left(\frac{\pi}{2}\right)^2 \frac{1}{2\sigma_{\Delta\theta}^2}\right\} + e^{-P} E_{\Delta\theta \in R_2} e^{P|\sin \Delta\theta|}, \tag{137}$$

where

$$\sigma_{\Delta\theta}^2 = 2\pi \frac{B_L}{R} \tag{138}$$

and again $P = A^2T$.

The penalty incurred due to phase noise depends on the behavior of

the expectation in (137). It does not appear feasible to evaluate this expectation exactly and therefore we must resort to asymptotic analysis valid for large P. We defer this analysis to after the discussion of heterodyne DPSK, since the penalty evaluation there involves a similar calculation.

Before embarking an analysis of heterodyne DPSK, however, we remark that even when $\Delta\theta = 0$, the probability of error in optically processed DPSK is 3 dB inferior to the quantum limit. This is seen from (17) since when $\Delta\theta = 0$,

$$Pe = \frac{1}{2} e^{-A^2T} = \frac{1}{2} e^{-P}. \tag{139}$$

The reason for this loss is inherent in the demodulation process. As can be seen, the optical detector turned the optical signal into an "on-off" signal and the explanation for the inefficiency is that twice as much average optical power has been transmitted to detect an on-off signal.

It should also be pointed out that the chief motivation for using differential phase modulation in the microwave region is that it yields an error probability close to coherent demodulation without having to transmit and recover carrier phase.[39] We therefore turn our attention to heterodyne DPSK detection next.

6.2 Heterodyne DPSK

As has already been pointed out, heterodyning an optical signal results in additive white noise and therefore it is imperative as in the FSK case, to bandlimit the heterodyned signal plus noise to a bandwidth just sufficient to pass the received signal with the impressed phase noise undisturbed. In DPSK, a bandlimited signal is multiplied by a delayed version and the product is post integrated in order to eliminate residual noise. This is a standard comparison detector[39] and is depicted in Fig. 13.

As has been done before, the heterodyned signal is represented in the interval $0 \leq t \leq T$ as,

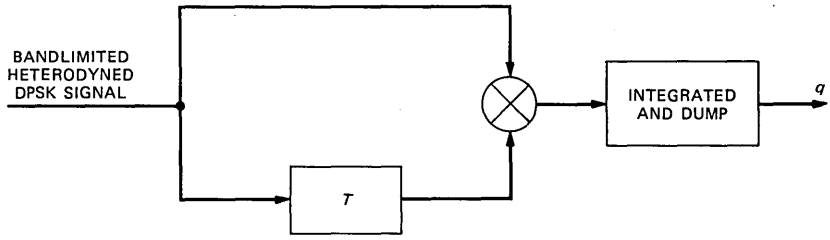


Fig. 13—Heterodyne differential phase shift keying.

$$V(t) = a_0 2A \cos(\omega_i t + \delta(t)) + n(t), \tag{140}$$

$$V_d(t) = a_{-1} 2A \cos(\omega_i t + \delta_d(t)) + n_d(t), \tag{141}$$

where the subscript indicates a delay of T seconds, ω_i is the IF frequency, and $\delta(t) = \theta(t) - \Phi(t)$, ($\theta(t)$ is the transmitter laser phase noise and $\Phi(t)$ is the local laser's phase noise.) Again, $n(t)$ is a white gaussian noise process with unit double sided spectral density.

Expressing (140) and (141) in terms of in-phase and quadrature components, yields

$$\begin{aligned} V(t) &= 2a_0 A \cos \delta(t) \cos \omega_i t + n_1(t) \cos \omega_i t + n_2(t) \sin \omega_i t \\ &\quad - 2a_0 A \sin \delta(t) \sin \omega_i t \\ &= [2a_0 A \cos \delta(t) + n_1(t)] \cos \omega_i t \\ &\quad - [2a_0 A \sin \delta(t) + n_2(t)] \sin \omega_i t, \end{aligned} \tag{142}$$

$$\begin{aligned} V_d(t) &= 2a_{-1} A \cos \delta_d(t) \cos \omega_i t + n_{1d}(t) \cos \omega_i t \\ &\quad + n_{2d} \sin \omega_i t - 2a_{-1} A \cos \delta_d(t) \sin \omega_i t \\ &= [2a_{-1} A \cos \delta_d(t) + n_{1d}(t)] \cos \omega_i t \\ &\quad + [2a_{-1} A \sin \delta_d(t) + n_{2d}(t)] \sin \omega_i t \end{aligned} \tag{143}$$

The baseband noise processes in (142) and (143), $n_1(t)$, $n_{1d}(t)$, $n_2(t)$ and $n_{2d}(t)$ are mutually independent and bandlimited to W hertz. Since the total noise power at the output of the band-pass filter is $4W$, the identical variances of the baseband noises must also equal to $4W$. Now, multiplying $V(t)$ by $V_d(t)$, eliminating double frequency components and integrating, results in the decision statistic q_0,

$$\begin{aligned} q_o &= \int_0^T [2a_o A \cos \delta(t) + n_1(t)][2a_{-1} A \cos \delta_d(t) + n_{1d}(t)] dt \\ &\quad + \int_0^T [2a_o A \sin \delta(t) + n_2(t)] \\ &\quad \cdot [2a_{-1} A \sin \delta_d(t) + n_{2d}(t)] dt \\ &= a_o a_{-1} q, \end{aligned} \tag{144}$$

where

$$\begin{aligned} q &= \int_0^T [2A \cos \delta(t) + n_1(t)][2A \cos \delta_d(t) + n_{1d}(t)] dt \\ &\quad + \int_0^T [2A \sin \delta(t) + n_2(t)][2A \sin \delta_d(t) + n_{2d}(t)] dt. \end{aligned} \tag{145}$$

A detection error is made whenever $a_0a_{-1} = 1$ and $q \leq 0$ or when $a_0a_{-1} = -1$ and $q \leq 0$. So, the bit error rate then is just

$$Pe = \Pr[q \leq 0]. \tag{146}$$

To facilitate the calculation of (146) let,

$$\begin{aligned} x(t) &= 2A \cos \delta(t) + n_1(t) \\ x_d(t) &= 2A \cos \delta_d(t) + n_{1d}(t) \\ y(t) &= 2A \sin \delta(t) + n_2(t) \\ y_d(t) &= 2A \sin \delta_d(t) + n_{2d}(t), \end{aligned}$$

and write

$$\begin{aligned} q = \int_0^T (xx_d + yy_d)dt = \int_0^T \left[\left(\frac{x + x_d}{2}\right)\right]^2 + \left(\frac{y + y_d}{2}\right)^2 \Bigg] dt \\ - \int_0^T \left[\left(\frac{x - x_d}{2}\right)\right]^2 + \left(\frac{y - y_d}{2}\right)^2 \Bigg] dt. \end{aligned} \tag{147}$$

Further define

$$\begin{aligned} u_1 &= \frac{x + x_d}{2}, \quad u_2 = \frac{y + y_d}{2}, \\ v_1 &= \frac{x - x_d}{2}, \quad v_2 = \frac{y - y_d}{2}, \end{aligned} \tag{148}$$

and conditioned on $\delta(t)$, calculate

$$\begin{aligned} Eu_1 &= A(\cos \delta + \cos \delta_d) \\ Eu_2 &= A(\sin \delta + \sin \delta_d) \\ Ev_1 &= A(\cos \delta - \cos \delta_d) \\ Ev_2 &= A(\sin \delta - \sin \delta_d). \end{aligned} \tag{149}$$

Rewriting (147) in terms of (148) we then obtain

$$\begin{aligned} q &= \int_0^T (u_1^2 + u_2^2) - \int_0^T (v_1^2 + v_2^2), \\ &= \sum_k (u_{1k}^2 + u_{2k}^2 - v_{1k}^2 - v_{2k}^2), \end{aligned} \tag{150}$$

where the u_k and v_k are again the coefficients in the expansions of $v(t)$ and $u(t)$ in the eigenfunctions $\{\psi_n(t)\}$.

Structurally (150) is identical to the quadratic forms obtained in the FSK case and therefore, we can express the bit error rate as

$$Pe = \Pr[q \le 0] = \frac{-1}{2\pi i} \int_{-\infty}^{\infty} \frac{Ee^{i\omega q}}{\omega + i\epsilon}\, d\omega. \tag{151}$$

Now however the characteristic function of the quadratic form (150) is

$$Ee^{i\omega q} = E_{\delta,\delta_d}\left[\frac{\exp\left\{ i\omega \sum_k \frac{(\bar{u}_{1k}^2 + \bar{u}_{2k}^2)}{1 - 2i\omega\lambda_k} - i\omega \sum_k \frac{(\bar{v}_{1k}^2 + \bar{v}_{2k}^2)}{1 + 2i\omega\lambda_k} \right\}}{\prod_k (1 - 2i\omega\lambda_k)(1 + 2i\omega\lambda_k)} \right]. \tag{152}$$

The eigenvalues as before are approximately 1 for $k \le n = 2WT$ and zero beyond and consequently (152) is to a good approximation,

$$\begin{aligned} E\{e^{i\omega q} \mid \delta, \delta_d\} &\sim \frac{\exp\left\{ \frac{i\omega}{1 - 2i\omega} \int_0^T [(Eu_1)^2 + (Eu_2)^2]dt \right.}{(1 - 2i\omega)^n (1 + 2i\omega)^n} \\ &\qquad \left. - \frac{i\omega}{1 + 2i\omega} \int_0^T ((Ev_1)^2 + (Ev_2)^2)dt \right\} \\ &= \frac{\exp\left\{ \frac{i\omega 2A^2}{1 - 2i\omega} \int_0^T (1 + \cos(\delta - \delta_d))dt \right.}{(1 - 2i\omega)^n (1 + 2i\omega)^n} \\ &\qquad \left. \cdot \frac{-i\omega 2A^2}{1 + 2i\omega} \int_0^T (1 - \cos(\delta - \delta_d))dt \right\}. \end{aligned} \tag{153}$$

Substituting this formula into (143), we obtain

$$\begin{aligned} Pe = -\frac{1}{2\pi i} \int_{-i\infty}^{i\infty} \frac{dz}{z} E_\phi \\ \cdot \left[\frac{\exp\left\{ \frac{zA^2}{1 - z} \int_0^T (1 + \cos\phi)dt \right.}{(1 - z)^n (1 + z)^n} \right. \\ \left. \left. - \frac{z}{1 + z} A^2 \int_0^T (1 - \cos\phi)dt \right\} \right], \end{aligned} \tag{154}$$

where $\phi = \delta(t) - \delta_d(t)$.

Unfortunately this integral cannot be expressed more explicitly as in FSK because of the essential singularity at $z = -1$. When $\phi = 0$ (no phase noise), the essential singularity disappears and (154) is identical to the previously encountered integral associated with the error rate in FSK. So in this case we obtain exactly

$$Pe = P_{n-1}(P)e^{-P}, \tag{155}$$

where P_{n-1} is again the $(n-1)$th order polynomial defined in Appendix B and $P = A^2T$.

In the important case when $1/T = R \gg B_L$, we obtain from (154), setting $n = 1$,

$$Pe = \frac{e^{-A^2T}}{2} = \frac{e^{-P}}{2}. \tag{156}$$

This result, again as in the optical case, can be seen to be 3 dB inferior to the "quantum limit" that is solely attributed to the 3-dB loss in the heterodyning process.

We now return to the central problem of assessing the penalty incurred by DPSK due to the presence of phase noise. An exact evaluation of the penalty is not mathematically tractable in general, and, as in the previous case, we must resort to upper bounds using the moment generating function of the quadratic form (153) or (154).

From (154) the moment generating function of q is

$$\begin{aligned} M_q(z) = Ee^{zq} &= E_\phi \exp \frac{\left(P\bar{\nu}\dfrac{z}{1-z} - P\bar{u}\dfrac{z}{1+z}\right)}{(1-z)^n(1+z)^n}, \\ &\le E \exp\left\{P\nu\frac{1}{1-z} - Pu\frac{z}{1+z}\right\}, \qquad z \ge 0, \end{aligned} \tag{157}$$

where

$$\bar{\nu} = \frac{1}{T}\int_0^T \nu, \qquad \bar{u} = \frac{1}{T}\int_0^T u$$

and

$$\nu = 1 + \cos\phi, \qquad u = 1 - \cos\phi. \tag{158}$$

The inequality in (157) is valid because of convexity and the stationarity of $\phi(t)$. While the form of this moment generating function is different from (129), similar techniques can be used to bound the probability of error. We proceed as follows:

$$Pe = \Pr[q \le 0] \le \frac{E_\phi e^{-Pf(z)}}{(1-z)^n(1+z)^n}, \tag{159}$$

where

$$f(z) = z\left(\frac{\nu}{1+z} - \frac{u}{1-z}\right), \qquad z \ge 0.$$

Splitting the range of ϕ into two parts, R_1 such that $\phi \epsilon R_1$, $\cos \phi > 0$ and R_2 such that $\phi \epsilon R_2$, $\cos \phi \leq 0$, we write (159) as

$$Pe \leq E_{\phi \epsilon R_1} \frac{e^{-Pf(z)}}{(1-z)^n(1+z)^n} + E_{\phi \epsilon R_2} \frac{e^{-Pf(z)}}{(1-z)^n(1+z)^n} \quad (160)$$

By setting the derivative of $f(z)$ to zero reveals that there exists an optimizing $z > 0$ namely,

$$z_0 = \frac{\sqrt{\nu} - \sqrt{u}}{\sqrt{\nu} + \sqrt{u}}, \qquad \cos \phi > 0$$

$$= 0, \qquad \cos \phi \leq 0 \quad (161)$$

When these values of z_0 are substituted into (160) we get

$$Pe \leq \exp\left(-\left(\frac{\pi}{2}\right)^2 \left(\frac{1}{2\sigma_\phi^2}\right)\right) + E_{\phi \epsilon R_2}\left(\frac{(1 + |\sin \phi|)^n}{2^n |\sin \phi|^n} e^{-P(1-|\sin\phi|)}\right), \quad (162)$$

where

$$\sigma_\phi^2 = 2\pi \frac{(B_{L1} + B_{L2})}{R}, \quad (163)$$

and where B_{L1} and B_{L2} are again the linewidths of the two lasers.

From (162) and (137) we see that the degradation from ideal performance, $\exp(-P)$, is essentially determined by the average value over the range $\phi \epsilon R_2$ of

$$F(\phi) = Ge^{P|\sin\phi|}, \quad (164)$$

where

$$G(\phi) = \left(\frac{1 + |\sin \phi|}{2 \sin \phi}\right)^2,$$

or

$$= 1,$$

depending on whether it is used in eq. (162) or (137), respectively.

Thus we write

$$EF(\phi) = \frac{1}{\sqrt{2\pi}\sigma_\phi} \int_{R_2} G(\phi) e^{P\epsilon(\phi)} d\phi, \quad (165)$$

where

$$\epsilon(\phi) = \sin \phi - \frac{\phi^2}{2C}, \quad (166)$$

and

$$C = \sigma_\phi^2 P. \tag{167}$$

We now evaluate (167) asymptotically valid for $P \to \infty$ and fixed C. Setting the derivative of (166) to zero, we solve the transcendental equation for the saddle point, $0 \le \phi_0 \le \pi/2$.

$$\epsilon'(\phi_0) = \cos \phi_0 - \frac{\phi_0}{C} = 0, \tag{168}$$

and observe that

$$\epsilon''(\phi_0) = -\left(\sin \phi + \frac{1}{C}\right) < 0. \tag{169}$$

With these results we can express (165) as

$$EF(\phi) \sim \frac{1}{\sqrt{2\pi}\sigma_\phi} G(\phi_0) e^{P\epsilon(\phi_0)} \int_{-\infty}^{\infty} e^{\frac{P\epsilon''(\phi_0)}{2}(\phi-\phi_0)^2} d\phi$$

$$\sim \frac{G(\phi_0)}{\sigma_\phi \sqrt{|\epsilon''(\phi_0)|P}} e^{P\epsilon(\phi_0)}. \tag{170}$$

Using (170) we can summarize the error rate estimates developed in the foregoing analysis.

1. Optical DPSK. Substituting (170) into (137) with the definition of $G(\phi)$ in (167), the probability of error is

$$Pe \le \exp\left\{-\left(\frac{\pi}{2}\right)^2 \frac{1}{2\sigma_{\Delta\theta}^2}\right\} + [C \sin \phi_0 + 1]^{-1/2} \cdot \exp\{-P(1 + \phi_0^2/2C - \sin \phi_0)\}, \tag{171}$$

where

$$\sigma_{\Delta\theta}^2 = 2\pi \frac{B_L}{R}.$$

2. Heterodyne DPSK. Now substituting (170) into (162) yields,

$$Pe \le \exp\left\{-\left(\frac{\pi}{2}\right)^2 \frac{1}{2\sigma_\phi^2}\right\} + [C \sin \phi_0 + 1]^{-1/2} \left[\frac{1 + \sin \phi_0}{\sin \phi_0}\right]^n \cdot \exp\left\{-P\left(1 + \frac{\phi_0^2}{2C} - \sin \phi_0\right)\right\}, \tag{172}$$

where

$$\sigma_\phi^2 = 2\pi \frac{B_{L1} + B_{L2}}{R}.$$

Ignoring unimportant coefficients, the probability of error is seen to be dominated by the maximum of the two terms in either (171) or (172). These are seen to be identical expressions barring the coefficients. It is observed from (171) and (172) that the exponents in the second term approach the exponents of the first term as $P \to \infty$. This can be verified from (168) since when $P \to \infty$, $C \to \infty$ for fixed σ_ϕ^2 and so the solution, $\phi_0 \to \pi/2$. As a consequence of this limit, our estimate of the error rate versus P has a threshold, or floor, at

$$\exp\left\{-\left(\frac{\pi}{2}\right)^2\left(\frac{1}{2\sigma_\phi^2}\right)\right\}.$$

As an example, at $Pe \sim 10^{-9}$,

$$\left(\frac{\pi}{2}\right)^2\left(\frac{1}{2\sigma_\phi^2}\right) \sim 20$$

and according to our prediction the floor for optical DPSK is $R/B_L \sim 100$ while in heterodyne DPSK it is $R/B_L \sim 200$, for identical laser linewidths. These floor predictions are slightly pessimistic. The lower bounds in (123) predict a floor of $R/B_L = 67$ for optical DPSK and $R/B_L = 134$ for heterotype DPSK. The discrepancy has to do with our bounding techniques.

It is now possible to define the exponential degradation or penalty, from ideal performance (above the respective floors) in either case by

$$\text{Penalty} = -10 \log\{1 - \sin \phi_0 + \phi_0^2/2C\}. \tag{173}$$

This is seen to be a function of the single parameter C defined in (167) and ϕ_0 the solution to (168).

It is important to observe that (171) and/or both (162) and (137) exhibit an exponential degradation due to phase noise unlike in the case of FSK. In both systems widening the front-end bandwidths of the respective detectors ensures minimal distortion suffered by the received heterodyned signals. However, in DPSK the static phase noise differential manifests itself in an exponential degradation, while in FSK no such degradation occurs.

In Fig. 14 we exhibit the penalty function, (173), as a function of R/B_L for both optical and heterodyne DPSK. We note that the exponential degradation is infinite below these respective floors. The arrows shown on the figures at $R/B_L = 100$ and 200 are aimed to emphasize that according to our estimates, the degradation is infinite at rates less than these values. In other words, no amount of additional optical energy can drive the error rate below these respective floors. Similar curves can be drawn for different Pes and hence different Ps. A striking feature of these curves is that the 3-dB degradation is

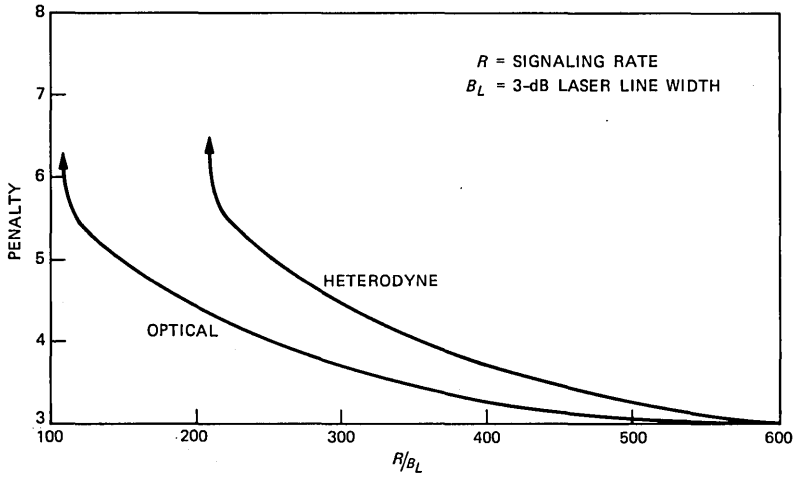


Fig. 14—Penalty in decibels due to phase noise in optical and heterodyne DPSK, $Pe \sim 10^{-9}$, $D = 20$.

approached very rapidly at rates greater than 400 times the laser linewidths (less than 4 dB is given up in either system). Evidently DPSK is very sensitive to phase noise for $R < 300B_L$.

VII. ON-OFF KEYING

We saw that on-off keying of an optical wave using direct detection achieves the quantum limit. Here we wish to analyze the performance of this modulation method when the on-off optical signal is first heterodyned to an IF frequency and then direct detected. Since the microwave version of on-off modulation has the same signal distance properties as FSK, we expect similar performance. There are however some differences and for the sake of completeness we include the following analysis.

Letting the IF frequency be ω_i, the differential phase noise be $\delta(t)$ and the resultant Gaussian noise with unit spectral density be $n(t)$, the observed IF microwave signal is then represented as

$$a2A \cos(\omega_0 t + \delta(t)) + n(t), \qquad 0 \le t \le T, \tag{174}$$

where here $a = 1$ or 0.

A reasonable way to process (121) is to first bandlimit it, then envelope detect the bandlimited version and finally postintegrate the square-envelope to obtain the decision statistic. This processor is depicted in Fig. 15 and the output statistic is

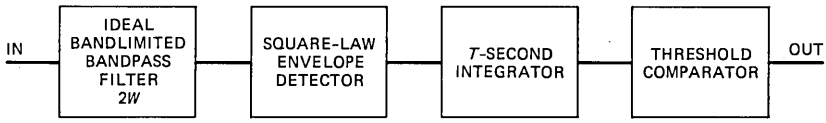


Fig. 15—Envelope detector.

$$q_1 = \int_0^T [2A \cos \delta(t) + n_1(t)]^2 dt + \int_0^T [2A \sin \delta(t) + n_2(t)]^2 dt, \qquad a = 1, \quad (175)$$

and

$$q_0 = \int_0^T [n_1^2(t) + n_2^2(t) dt, \qquad a = 0, \quad (176)$$

where n_1 and n_2 are baseband gaussian noise processes with double-sided spectral densities equal to 2 and are bandlimited to W hertz. This must be so since the bandpass noise process

$$n(t) = n_1(t)\cos \omega_0 t + n_2(t)\sin \omega_0 t,$$

is bandlimited to $2W$ and consequently,

$$En^2 = En_1^2 = En_2^2 = 4W.$$

In writing (122) we assumed as before that W is sufficiently wide to pass the in-phase and quadrature signals in the presence of phase noise undistorted.

Now let

$$x(t) = 2A \cos \delta(t) + n_1(t)$$

and

$$y(t) = 2A \sin \delta(t) + n_2(t), \quad (177)$$

and make an expansion of x and y in terms of the prolate-spheroidal orthonormal set of functions

$$\{\psi_n(t)\}, \qquad 0 \le t \le T, \qquad n = 0, 1, 2, \cdots. \quad (178)$$

Then we can write

$$q_i = \sum_n [x_n^2 + y_n^2]$$

and

$$q_0 = \sum_n [n_{1n}^2 + n_{2n}^2]. \quad (179)$$

We note that $\{x_n\}$, $\{y_n\}$, $\{n_{1n}\}$ and $\{n_{2n}\}$ are mutually independent conditional Gaussian random variables (conditioned on $\delta(t)$) with the following parameters:

$$Ex_n = \int_0^T 2A \cos \delta(t)\psi_n(t)dt, \text{ all } n$$

$$Ey_n = \int_0^T 2A \sin \delta(t)\psi_n(t)dt, \text{ all } n,$$

and

$$En_{1n} = En_{2n} = 0, \text{ all } n.$$

Also

$$\text{Var } x_n = \text{Var } y_n = \text{Var } n_{1n} = \text{Var } n_{2n} = \lambda_n,$$

where $\{\lambda_n\}$ are the eigenvalues associated with the eigenfunctions $\psi_n(t)$.

Our method of estimating the error rate will be based on a Chernof bounding technique requiring the moment generating functions of q_1 and q_0. These functions are readily evaluated for any θ as follows,

$$Mq_1(\theta) = Ee^{\theta q_1} = \prod_\ell Ee^{\theta x_\ell^2} Ee^{\theta y_\ell^2}$$

$$= \frac{\exp\left\{\theta \sum_\ell \frac{(Ex_\ell)^2 + (Ey_\ell)^2}{1 - 2\lambda_\ell\theta}\right\}}{\prod_\ell [1 - 2\lambda_\ell\theta]}, \tag{180}$$

and

$$Mq_0(\theta) = Ee^{\theta q_0} = \prod_\ell Ee^{\theta n_{1\ell}^2} Ee^{\theta n_{2\ell}^2}$$

$$= \prod_\ell (1 - 2\lambda_\ell\theta)^{-1}. \tag{181}$$

To proceed further with the analysis, we invoke again the excellent approximation regarding the behavior of the eigenvalues λ_ℓ. Here it can be verified that

$$\lambda_\ell \sim 2, \qquad \ell \leq n = 2WT$$

$$= 0, \qquad \ell > n = 2WT. \tag{182}$$

and when this approximation is used in (180) and (181) we obtain for the moment generating functions

$$Mq_1(\theta) = \frac{\exp\left\{P\frac{\theta}{1 - \theta)}\right\}}{(1 - \theta)^n}, \tag{183}$$

and

$$Mq_0(\theta) = (1 - \theta)^n, \tag{184}$$

where $P = A^2T$, the optical energy.

The bit error rate is now upper bounded as follows:

$$Pe = \tfrac{1}{2}\Pr[q_1 \le \tau] + \tfrac{1}{2}\Pr[q_0 > \tau], \tag{185}$$

where τ is a threshold that will be optimized later. The probabilities in (185) cannot be evaluated exactly; however, tight exponential upper bounds can be obtained as follows.

$$\Pr[q_1 \le \tau] \le e^{\theta\tau} M_{q1}(-\theta), \qquad 0 \le \theta < 1 \tag{186}$$

and

$$\Pr[q_0 > \tau] \le e^{-\theta\tau} M_{q1}(\theta), \qquad 0 \le \theta < 1. \tag{187}$$

Using (183) in (186) it can be verified that there exists a $\theta = \theta_0$, which makes the bound tightest. By differentiation we find

$$\theta_0 = \sqrt{\frac{P}{\tau}} - 1, \qquad P > \tau, \tag{188}$$

and when this value of θ_0 is substituted into (186) we get

$$\Pr[q_1 \le \tau] \le \frac{\left(\frac{\tau}{P}\right)^n}{2} e^{-P\left(1-\sqrt{\frac{\tau}{P}}\right)^2} \tag{189}$$

Following the same procedure for tightening the bound in (187) we find an optimum θ_0 given by,

$$\theta_0 = 1 - \frac{n}{\tau}, \qquad \tau > n, \tag{190}$$

and when this value of θ is substituted into (187) we obtain,

$$\Pr[q_0 \ge \tau] \le \left(\frac{\tau}{n}\right)^n e^{-n\left(\frac{\tau}{n}-1\right)}. \tag{191}$$

Equating the exponent in (189) to the one in (191) we determine the best threshold τ given by

$$\tau_0 = \frac{1}{4P}(P+n)^2$$

Substituting this value of τ_o in (189) and (191) we get for Pe in (185)

$$Pe \le \left(\frac{1}{2}\right)^{2n+1}\left[\left(\frac{P}{n}\right)^n\left(1+\frac{n}{P}\right)^{2n} + \left(1+\frac{n}{P}\right)^n\right] e^{-\frac{P}{4}\left(1-\frac{n}{P}\right)^2} \tag{192}$$

We see from this upper bound that when $n < P$,

$$Pe \sim e^{\frac{A^2T}{4}} = e^{-\frac{P}{2}} \tag{193}$$

which is identical to the FSK performance since here the average optical energy is half that of FSK. The exponential degradation from ideal is seen to be

$$D_0 = -20 \log \left(1 + \frac{n}{P}\right),$$

where $n = 2WT$ is the total band required to pass the modulated signal with phase noise undistorted. A conservative example might be as follows. $P = 80$ will yield an ideal error rate of $\sim 10^{-9}$, $n = 1/R(1 + 10B_L)$ ($10B_L \sim 400$ MHz should be sufficient to pass the signal with phase noise undistorted.) For these parameters a simple calculation reveals that when $R = 0.5$ GB/s the penalty is about 0.2 dB. At rates below this the degradation starts to be substantial. Note however, that this performance is still 3 dB poorer than DPSK and 6 dB poorer than the quantum limit.

VIII. ACKNOWLEDGMENTS

I wish to thank many of my colleagues at AT&T Bell Laboratories for the many fruitful technical interactions during the course of this research. The contributions of Paul S. Henry have been invaluable. He acquainted me with this subject matter, provided continuous guidance, and made significant and substantial contributions to the evolution of the main results. I would have been honored to have him as a coauthor but he modestly refused.

I also had valuable discussions with the following: A. S. Acampora, N. Amitay, B. Glance, D. J. Goodman, L. J. Greenstein, G. J. Foschini, T. Li, L. A. Linke, B. F. Logan, R. W. Lucky, J. E. Mazo, A. A. M. Saleh, G. Vannucci, and A. D. Wyner. Thanks are also due to Vincent W. S. Chan from MIT Lincoln Laboratory for stimulating discussions.

REFERENCES

1. M. Ross, *Laser Receivers*, New York: Wiley and Sons, 1966.
2. W. K. Pratt, *Laser Communication Systems*, New York: Wiley and Sons, 1969.
3. Y. Yamamoto and T. Kimura, "Coherent Optical Fiber Transmission Systems," IEEE J. Quant. Electron., *QE-17* (June 1981), pp. 919–35.
4. T. Okoshi et al., "Computation of Bit-Error Rate of Various Heterodyne and Coherent-Type Optical Communication Schemes," J. Opt. Commun., *2* (September 1981), pp. 89–96.
5. T. Okoshi, "Heterodyne and Coherent Optical Fiber Communications: Recent Progress," IEEE Trans. Micr. Th. and Tech., *MTT-30* (August 1982), pp. 1138–48.
6. F. Faure and D. LeGuen, "Effect of Semiconductor Laser Phase Noise on BER Performance in an Optical DPSK Heterodyne-Type Experiment," Electron. Lett., *18* (October 28, 1982), pp. 964–5.
7. S. Saito, Y. Yamamoto, and T. Kimura, "S/N and Error Rate Evaluation for an Optical FSK-Heterodyne Detection System Using Semiconductor Lasers," IEEE J. Quant. Electron., *QE-19* (February 1983), pp. 180–93.
8. A. D. Wyner, unpublished work.

9. J. E. Mazo and J. Salz, "On Optical Data Communication via Direct Detection of Light Pulses," B.S.T.J., *55,* No. 3 (March 1976), pp. 347–70.
10. J. C. Campbell et al., "High Performance Avalanche Photodiode with Separate Absorption, Grading and Multiplication Regions," Electron. Lett., *19* (September 29, 1983), pp. 818–9.
11. R. S. Kennedy, private communication.
12. C. H. Henry, "Theory of the Linewidth of Semiconductor Lasers," IEEE J. Quant. Electron., *QE-18* (February 1982), pp. 259–64.
13. M. W. Fleming and A. Mooradian, "Fundamental Line Broadening of Single-Model GaAlAs Diode Lasers," Appl. Phys. Lett., *38* (April 1, 1981), pp. 511–3.
14. C. Harder, K. Vahala, and A. Yariv, "Measurement of the Linewidth Enhancement Factor α of Semiconductor Lasers," Appl. Phys. Lett., *42* (February 15, 1983), pp. 328–30.
15. F. G. Walther and J. E. Kaufmann, "Characterization of GaAlAs Laser Diode Frequency Noise," Sixth Top. Mtg. Opt. Fib. Commun., New Orleans, February–March 1983, Paper TUJ5.
16. J. E. Kaufmann, "Phase and Frequency Tracking Considerations for Heterodyne Optical Communications," Proc. Int. Telemetering Conf., San Diego, Calif., September 1982.
17. S. O. Rice, "Mathematical Analysis of Random Noise, W. Nelson, ed., *Selected Papers on Noise and Stochastic Processes:* Dover Publications Inc., 1954, pp. 145–62.
18. M. Shikada et al., "100 Mb/s ASK Heterodyne Detection Experiment Using 1.3 μm DFB Laser Diodes," Conf. Opt. Fiber Commun., New Orleans, January 1984, Paper TUK6.
19. K. Kikuchi, T. Okoshi, and R. Arata, "Measurements of Linewidth and FM-Noise Spectrum of 1.52 μm InGaAsP Lasers," Electron. Lett., *20* (June 21, 1984), pp. 535–6.
20. T. P. Lee et al., "Measured Spectral Linewidth of Single-Frequency 1.3 and 1.5 μm Injection Lasers," Electon. Lett., *20* (November 22, 1984), pp. 1011–2.
21. R. Wyatt and W. J. Devlin, "10 kHz Linewidth 1.5 μm InGaAsP External Cavity Laser with 55 nm Tuning Range," Electron Lett., *19* (February 3, 1983), pp. 110–2.
22. A. L. Scholtz et al., "Infra-red Homodyne Receiver with Acousto-optically Controlled Local Oscillator," El. Lett., *19* (March 17, 1983), pp. 234–5.
23. D. D. Falconer and J. Salz, "Optimal Reception of Digital Data Over the Gaussian Channel with Unknown Delay and Phase Jitter," IEEE Trans. Inf. Th., *IT-23,* No. 1 (January 1977), pp. 117–26.
24. E. Wong, *Stochastic Processes in Information and Dynamical Systems,* New York: McGraw-Hill, 1971.
25. A. J. Viterbi, *Principles of Coherent Communication,* New York: McGraw-Hill, 1966.
26. R. M. Gagliardi and S. Karp, *Optical Communications,* New York: Wiley and Sons 1976, Chapter 6.
27. R. Wyatt, T. G. Hodgkinson, and D. W. Smith, "1.52 μm PSK Heterodyne Experiment Featuring an External Cavity Diode Laser Local Oscillator," Electron. Lett., *19* (July 7, 1983), pp. 550–2.
28. V. K. Prabhu, "PSK Performance with Imperfect Carrier Phase Recovery," IEEE Trans. Aerospace and Elec. Syst., *AES-12,* No. 2 (March 1976), pp. 275–86.
29. K. Kikuchi et al., "Bit-Error Rate of PSK Heterodyne Optical Communication System and its Degradation Due to Spectral Spread of Transmitter and Local Oscillator," Electron. Lett., *19,* No. 11 (May 26, 1983), pp. 417–8.
30. V. M. S. Chan, L. L. Jeromin, and J. E. Kaufmann, "Heterodyne Lasercom Systems using GaAs Lasers for ISL Applications," IEEE Int. Conf. Commun., Boston, June 1983, Paper E1.5.
31. L. L. Jeromin and V. W. S. Chan, "Performance Estimates for a Coherent Optical Communication System," Conf. Opt. Fiber Commun., New Orleans, January 1984, Paper TUK2.
32. R. Wyatt et al., "140 Mb/s Optical FSK Fibre Heterodyne Experiment at 1.54 μm," Electron. Lett., *20* (October 25, 1984), pp. 912–3.
33. K. Emura et al., "Novel Optical FSK Heterodyne Single Filter Detection System Using a Directly Modulated DFB Laser Diode," Electron. Lett., *20* (November 22, 1984), pp. 1022–3.
34. J. E. Mazo and J. Salz, "Probability of Error for Quadratic Detectors," B.S.T.J., *44* (November 1965), pp. 2165–86.

35. D. Slepian and H. Pollak, "Prolate Sphyroidal Wave Functions—Fourier Analysis and Uncertainty—I," B.S.T.J., *40* (January 1961), pp. 43–63.
36. W. R. Bennett and J. R. Davey, *Data Transmission*, New York: McGraw-Hill, 1965, Chapter 11.
37. M. Shikada, M. Emura, and K. Minemura, "High-Sensitivity Optical PSK Heterodyne Differential Detection Simulation Experiment," Fourth Int. Conf. Int. Opt. and Opt. Fiber Commun., Tokyo, June 1983, Paper 30C3–4.
38. G. Nicholson, "Probability of Error for Optical Heterodyne DPSK System with Quantum Phase Noise," Electron. Lett., *20* (November 22, 1984), pp. 1005–7.
39. R. W. Lucky, J. Salz, and E. J. Weldon, Jr., *Principles of Data Communication*, New York: McGraw-Hill, 1968.

APPENDIX A

An Estimate of Probability of Error in Optical Homodyne Detection

Here we derive an upper bound for the bit error rate which will be used in evaluating performance of optical homodyne as well as heterodyne reception. We need to estimate the following probability (eq. 192),

$$Pe = \Pr[-\rho\xi + \bar{\nu}_0 \geq 0], \tag{194}$$

where ξ and $\bar{\nu}_0$ are independent random variables with joint probability density $p(\xi)p(\bar{\nu}_0)$. By definition, (194) is written,

$$Pe = \int_0^\infty p(x)dx, \tag{195}$$

where $p(x)$ is the density of $\bar{\nu}_0 - \rho\xi$. For any $\lambda \geq 0$, (195) is upper bounded by

$$Pe \leq \int_{-\infty}^{\infty} e^{\lambda x}p(x)dx = Ee^{\lambda x} = Ee^{-\rho\xi\lambda}Ee^{\lambda\bar{\nu}_0} = e^{\frac{\lambda^2}{2}}Ee^{-\rho\xi\lambda} \tag{196}$$

To make progress with (196), we note that since e^{-y} is convex

$$\begin{aligned} Ee^{-\rho\xi\lambda} &= Ee^{-\rho\lambda\frac{1}{T}\int_0^T \cos\psi dt} \\ &\leq \frac{1}{T}\int_0^T Ee^{-\rho\lambda\cos\psi}dt = Ee^{-\lambda\rho\cos\psi}, \end{aligned} \tag{197}$$

since $\psi(t)$ is a stationary process. Using this result, (196) is upper bounded by

$$Pe \leq e^{\frac{\lambda^2}{2}}Ee^{-\lambda\rho\cos\psi} \tag{198}$$

Substituting the stationary probability density of ψ (36) into (198) we get,

$$Pe \leq e^{\frac{\lambda^2}{2}} \frac{1}{2\pi I_0(\alpha)} \int_{-\pi}^{\pi} e^{-\lambda\rho\cos\psi + \alpha\cos\psi} d\psi$$

$$\leq \frac{1}{I_0(\alpha)} e^{\frac{\lambda^2}{2} + |\alpha - \lambda\rho|}. \tag{199}$$

Lower bounding the Bessel function by

$$I_0(\alpha) \geq \frac{e^{\alpha}}{\sqrt{2\pi\alpha}} \left[1 - \frac{1}{\pi} \sqrt{\frac{2}{\alpha}} e^{-\pi^2\alpha/2}\right],$$

we further upper bound (199)

$$Pe \leq \frac{e^{\frac{\lambda^2}{2} - \alpha + |\alpha - \lambda\rho|}}{\left(1 - \frac{1}{\pi}\sqrt{\frac{2}{\alpha}} e^{-\frac{\pi^2\alpha}{2}}\right)(\sqrt{2\pi\alpha})^{-1}}. \tag{200}$$

The exponent in (200) is now optimized with respect to $\lambda > 0$. Let

$$E(\lambda) = \frac{\lambda^2}{2} - \alpha + |\alpha - \lambda\rho|, \tag{201}$$

and note that $E(0) = 0$. Moreover,

$$\frac{d}{d\lambda} E(\lambda)|_{\lambda=0} = (\lambda - \rho\ \mathrm{sgn}(\alpha - \lambda\rho))_{\lambda=0}$$

$$= -\rho. \tag{202}$$

We are thus assured that a $\lambda = \lambda_0$ exists such that $E(\lambda_0) \leq 0$. To find this optimum value of λ, we examine,

$$\frac{dE(\lambda)}{d\lambda} = \lambda - \rho\ \mathrm{sgn}(\alpha - \lambda\rho) = 0, \tag{203}$$

and conclude that when $\alpha/\rho \geq \rho$, $\lambda_0 = \rho$. On the other hand when $\alpha/\rho \leq \rho$, the optimum value of λ_0 is on the boundary $\lambda_0 = \alpha/\rho$ and so the optimum exponent in (201) is

$$E(\lambda_0) = \begin{cases} \frac{\rho^2}{2}, & \alpha > \rho^2 \\ \alpha\left[1 - \frac{\alpha}{2\rho^2}\right], & \alpha \leq \rho^2 \end{cases}. \tag{204}$$

Substituting (204) into (200) we finally get

$$Pe \leq \begin{cases} g(\alpha)e^{-\frac{\rho^2}{2}}, & \frac{\alpha}{\rho^2} > 1 \\ g(\alpha)e^{-\alpha\ 1-\frac{\alpha}{2\rho^2}}, & \frac{\alpha}{\rho^2} < 1 \end{cases}, \tag{205}$$

where

$$g(\alpha) = \left(1 - \frac{1}{\pi}\sqrt{\frac{2}{\alpha}}\, e^{-\frac{\pi^2\alpha}{2}}\right)^{-1} \sqrt{2\pi\alpha}.$$

APPENDIX B

Evaluation of Residues

I am indebted to B. F. Logan, Jr. of Department 11219 for supplying material included in this Appendix.

We require the residue of $f_n(z)/(-z)$ at $z = -1$, where

$$f_n(z) = \frac{\exp\left(\frac{Pz}{1-z}\right)}{(1-z)^{n+1}(1+z)^{n+1}}, \qquad P > 0. \tag{206}$$

Setting $P/2 = \lambda$, $z = t - 1$, the probability of error $Pe(P, n)$ in Section V (eq. 118) is the coefficient of t^n in the Taylor series expansion of the function

$$\frac{\exp\left(P\,\frac{(t-1)}{2-t}\right)}{(1-t)(2-t)^{n+1}} = e^{-\frac{P}{2}}\,\frac{\exp\left(\frac{Pt}{2(2-t)}\right)}{(1-t)(2-t)^{n+1}}. \tag{207}$$

Now consider the function

$$F_n(x, \lambda) = \frac{1}{2^{n+1}}\,\frac{\exp\left(\frac{\lambda x}{1-x}\right)}{(1-2x)(1-x)^{n+1}}$$

$$= \frac{1}{2^{n+1}} \sum_{k=0}^{\infty} p_k(\lambda, n)x^k. \tag{208}$$

Then,

$$Pe(n, \lambda) = \frac{e^{-\lambda}}{2^{2n+1}}\, p_n(\lambda, n).$$

The generator function for the generalized Laguere polynomials, defined as

$$L_m^{(n)}(-\lambda) = \sum_{k=0}^{m} \binom{m+n}{n-k} \frac{\lambda^k}{k!}$$

$$= \sum_{k=0}^{m} \binom{m+n}{k} \frac{\lambda^{m-k}}{(m-k)!} \tag{209}$$

is just the function

$$Gn(x, \lambda) = \frac{\exp\left(\frac{\lambda x}{1-x}\right)}{(1-x)^{n+1}}$$

$$= \sum_{k=0}^{\infty} L_k^{(n)}(-\lambda) x^n. \tag{210}$$

Multiplying Gn by $(1-2x)^{-1}$, we have

$$p_n(\lambda, n) = \sum_{k=0}^{m} 2^{m-k} L_k^{(n)}(-\lambda), \tag{211}$$

and collecting coefficients of $\frac{\lambda^k}{k!}$ we find

$$p_m(\lambda, n) = \sum_{k=0}^{n} a_k(m, n) \frac{\lambda^k}{k!}, \tag{212}$$

where

$$a_k(m, n) = \sum_{j=0}^{m-k} \binom{n+k+j}{j} 2^{m-k-j}$$

$$= \sum_{j=0}^{m-k} \binom{n+m-d}{n+k} 2^d.$$

Now setting $a_k(n, n) = p_k(n)$, we have

$$Pe(\lambda, n) = \frac{e^{-\lambda}}{2^{2n+1}} \sum_{k=0}^{n} p_k(n) \frac{\lambda^k}{k!}, \tag{213}$$

where

$$p_k(n) = \sum_{j=0}^{n} \binom{2n-j}{n+k} 2^j. \tag{214}$$

Since (214) is a finite polynomial, (213) can readily be evaluated by computer.

AUTHOR

Jack Salz, B.S.E.E., 1955, M.S.E., 1956, and Ph.D. (Electrical Engineering), 1961, University of Florida; AT&T Bell Laboratories, 1961—. Mr. Salz first worked on the electronic switching system. Since 1968 he has supervised a group engaged in theoretical studies in data communications and is currently a member of the Network Systems Research Department. During the academic year 1967–68, he was on leave as Professor of Electrical Engineering at the University of Florida. He was a visiting lecturer at Stanford University in Spring 1981 and a visiting MacKay Lecturer at the University of California, at Berkeley, in Spring 1983.

AT&T Technical Journal
Vol. 64, No. 10, December 1985
Printed in U.S.A.

Cross-Polarization Cancellation and Equalization in Digital Transmission Over Dually Polarized Multipath Fading Channels

By M. KAVEHRAD and J. SALZ*

(Manuscript received July 11, 1984)

A theory for data-aided equalization and cancellation in digital data transmission over dually polarized fading radio channels is presented. The present theory generalizes and extends previous work by admitting decision feedback structures with finite-tap transversal filter implementations. Subject to the assumption that some past and/or future data symbols are correctly detected, formulas and algorithms for evaluating the least mean-square error for different structures are presented. In a sequence of curves we evaluate and compare the performance of various structures for a particular propagation model and several fading events. We find improvement in performance for decision feedback over linear equalization. More importantly, we discovered that in this application, as in the single-channel transmission case, decision feedback/canceler structures are much less sensitive to timing phase than linear equalizers.

I. INTRODUCTION

One of the purposes of this article is to call attention to mounting research results pointing the way toward effective methods for combating the deleterious effects of various impairments arising in digital data transmission over dually polarized fading radio channels.

Transmission of M-state Quadrature Amplitude-Modulated (QAM) signals via orthogonally polarized carriers is an effective method for

* Authors are employees of AT&T Bell Laboratories.

reusing existing bandwidth with obvious economic advantages. The main obstacle in the way of realizing these advantages is the unavoidable presence of Cross-Polarization Interference (CPI) between the dually polarized signals that arise due to multipath fading, antenna misalignments, and imperfect waveguide feeds. The chief purpose of our current work is to obtain a fundamental understanding and a solution to this problem.

There is a well established theory of linear and decision feedback equalization/cancellation to mitigate the effects of intersymbol interference (ISI) and noise in the transmission of a single digital signal.[1] However, consideration of data-aided CPI cancellation in addition to ISI equalization in the presence of noise has not been treated before. The work of Amitay and Salz[2] establishes a theoretical base for optimal linear compensation of CPI and ISI in the presence of noise; however, their work is limited strictly to linear techniques and considers only ideal infinite-tap transversal structures.

In this article we generalize previous treatments of this subject in two major respects. Our first contribution is to cast the problem of CPI cancellation and ISI equalization in a general theoretical framework that admits data-aided decision feedback techniques. Secondly, and most importantly, we admit finite-tap transversal structures that in practice can be implemented adaptively.

The receiver configuration is based on a matrix structure suggested by the theory of optimal detection and is shown in Figs. 1 and 2. The optimal structure is comprised of a linear matrix equalizer/canceler and an ISI and CPI estimator, which is used to subtract some of the interference from the received signals. An architecture, previously proposed by Kavehrad,[3] is a special case of this generalized structure.

The dually polarized channel is modeled by a particular 4×4 real matrix impulse response or its Fourier transform followed by additive noise. The 2×2 block-diagonal elements of this matrix represent the copolarized (in line) responses, while the off-diagonal 2×2 block entries represent cross-coupled and cross-polarized interfering responses. Each matrix channel characterizes a snapshot of a multipath fading event, which in the presence of noise limits the achievable error rate of the receiver for a given data rate. We use a propagation model proposed in Ref. 2.

In comparing the performance of various equalizer/cancelers, the Mean-Square Error (MSE) is used. The justification for using this criterion has been amply discussed in the literature.[1,2] But the chief motivation for its use is due to its mathematical tractability. It turns out that it also leads to an exponentially tight upper bound on error rate. In practice, it lends itself to easy estimation and thereby is used to update transversal filter-tap coefficients recursively.

Section II contains the system model and theoretical developments. Computational algorithms are provided in Section III, and our numerical results and associated discussions are given in Section IV. Finally, a summary is presented in the last section.

II. THE MODEL AND THEORETICAL DEVELOPMENTS

2.1 System model

Consider a dually polarized digital radio communications channel supporting two independent QAM data signals. This type of communication channel with an ideal QAM modulator and demodulator is shown in Fig. 1. The four independent synchronous data signals $S_{lv}(t)$, $S_{lh}(t)$, $l = 1, 2$, with the generic representation

$$S_{lv}(t) = \sum_n a_{lvn} g(t - nT), \quad l = 1, 2$$

$$S_{lh}(t) = \sum_n a_{lhn} g(t - nT), \quad l = 1, 2, \tag{1}$$

amplitude modulate two linearly polarized carrier waves in quadrature. The modulated signal,

$$S_v(t) = S_{1v}(t)\cos \omega_0 t + S_{2v}(t)\sin \omega_0 t, \tag{2a}$$

is transmitted over the vertically polarized channel, while

$$S_h(t) = S_{1h}(t)\cos \omega_0 t + S_{2h}(t)\sin \omega_0 t \tag{2b}$$

is transmitted over the horizontal channel. The carrier frequency is ω_0 and the real data symbols

$$\{a_{lvn},\ l = 1, 2\} \quad \text{and} \quad \{a_{lhn},\ l = 1, 2\}, \quad -\infty < n < \infty$$

are assumed to be independently drawn from a lattice of points with odd integer coordinates. The QAM constellations associated with eq. (2) are, therefore, rectangular. The scalar shaping pulse, $g(t)$, is selected by the designer to satisfy limitations on transmitted power and bandwidth.

The individual transmission channels are characterized by bandpass impulse responses or by their respective Fourier transforms,

$$\begin{bmatrix} h_v(t) \\ h_h(t) \end{bmatrix} = \begin{bmatrix} h_{i11}(t) \\ h_{i22}(t) \end{bmatrix} \cos \omega_0 t + \begin{bmatrix} h_{q11}(t) \\ h_{q22}(t) \end{bmatrix} \sin \omega_0 t. \tag{3}$$

The resolution of $h_v(t)$ and $h_h(t)$ into their respective baseband in-phase and quadrature components turns out to be convenient in our application.

To accommodate coupling between the polarized channels, two pairs of impulse responses, one associated with the cochannel and the other

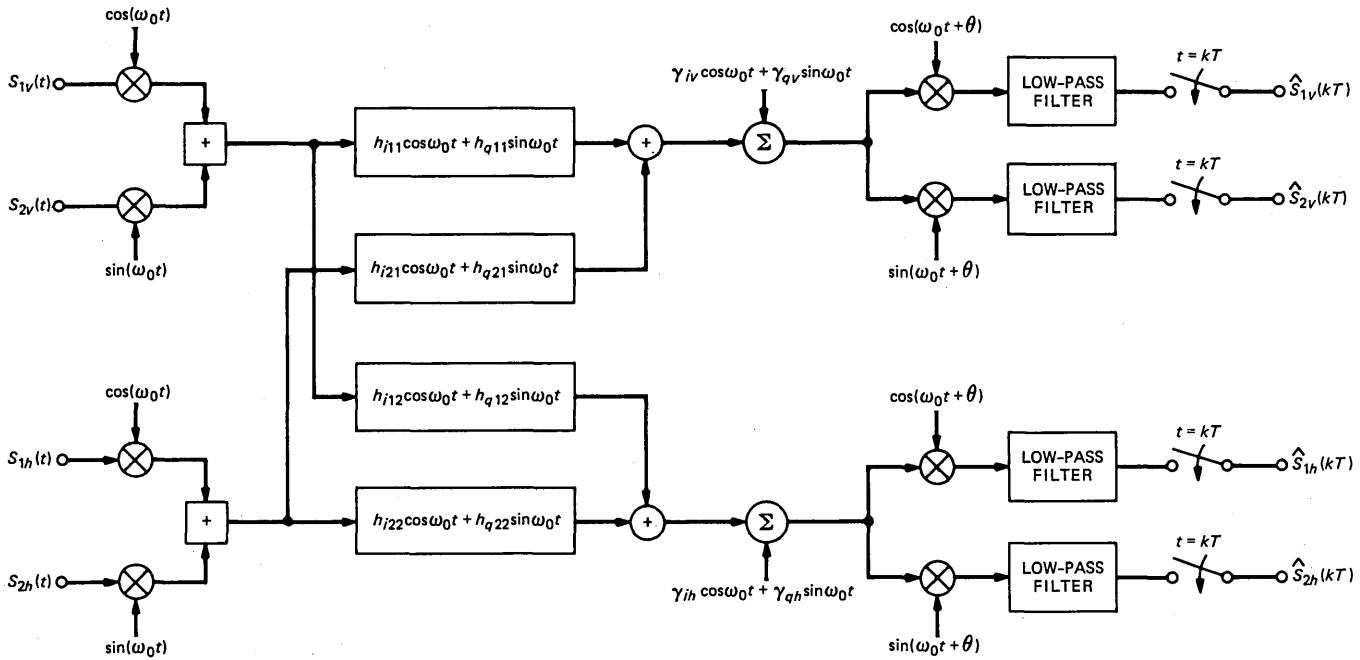


Fig. 1—System block diagram.

associated with the cross-channel, are used to completely characterize the medium.

At the output, two independent noises are added and the signal plus noise is then coherently demodulated. The end-to-end system including the modulators and the demodulators is shown in Fig. 1. It is convenient to view this linear system as a four-input port four-output port network and characterize it by a 4×4 matrix impulse response or its Fourier transform, which is the overall system frequency response.

It is now easy to verify that the I/O relationships can be expressed as follows (see Fig. 1):

$$\begin{aligned}
D_{1v} &= S_{1v}*h_{i11} + S_{2v}*h_{q11} + S_{1h}*h_{i21} + S_{2h}*h_{q21} + \nu_{iv} \\
D_{2v} &= -S_{1v}*h_{q11} + S_{2v}*h_{i11} - S_{1h}*h_{q21} + S_{2h}*h_{i21} + \nu_{qv} \\
D_{1h} &= S_{1v}*h_{i12} + S_{2v}*h_{q12} + S_{1h}*h_{i22} + S_{2h}*h_{q22} + \nu_{ih} \\
D_{2h} &= -S_{1v}*h_{q12} + S_{2v}*h_{i12} - S_{1h}*h_{q22} + S_{2h}*h_{i22} + \nu_{qh}, \qquad (4)
\end{aligned}$$

where $*$ denotes convolution,

$$h*S = \int_{-\infty}^{\infty} h(t-\tau)S(\tau)d\tau.$$

The representation in eq. (4) can be put into a convenient matrix form,

$$D(t) = \int_{-\infty}^{\infty} H(t-\tau)S(\tau)d\tau + \nu(t), \qquad (5)$$

where $H(t)$ is the 4×4 matrix channel impulse response

$$H(t) = \begin{bmatrix} h_{i11}(t) & h_{q11}(t) & h_{i21}(t) & h_{q21}(t) \\ -h_{q11}(t) & h_{i11}(t) & -h_{q21}(t) & h_{i21}(t) \\ h_{i12}(t) & h_{q12}(t) & h_{i22}(t) & h_{q22}(t) \\ -h_{q12}(t) & h_{i12}(t) & -h_{q22}(t) & h_{i22}(t) \end{bmatrix}, \qquad (6)$$

$$S(t) = \begin{bmatrix} S_{1v}(t) \\ S_{2v}(t) \\ S_{1h}(t) \\ S_{2h}(t) \end{bmatrix} \qquad (7)$$

is the input signal vector, and

$$\nu(t) = \begin{bmatrix} \nu_{iv}(t) \\ \nu_{qv}(t) \\ \nu_{ih}(t) \\ \nu_{qh}(t) \end{bmatrix} \qquad (8)$$

is the added noise vector.

Since complex numbers $x + jy$ are isomorphic to matrices of the form

$$x + jy \sim \begin{bmatrix} x & y \\ -y & x \end{bmatrix},$$

the channel model as described by the 4×4 real matrix, eq. (6), can also be represented by a 2×2 complex matrix of the form[2]

$$\begin{bmatrix} h_{i11} + jh_{q11} & h_{i21} + jh_{q21} \\ h_{i12} + jh_{q12} & h_{i22} + jh_{q22} \end{bmatrix}.$$

In our application, however, it turns out to be more convenient to work with the real matrix in eq. (6).

We now return to the I/O relationship in eq. (5) and substitute eq. (1) to obtain in more detail

$$D(t) = \sum_n \int g(\tau - nT)H(t - \tau)d\tau \cdot A_n + \nu(t), \tag{9}$$

where the real data symbol vector A_n is given by

$$A_n = \begin{bmatrix} a_{1vn} \\ a_{2vn} \\ a_{1hn} \\ a_{2hn} \end{bmatrix}. \tag{10}$$

A representative sample of $D(t)$ taken at $t = 0$, without loss of generality, yields

$$D(0) = H_0A_0 + \sum_{\substack{n \\ n \neq 0}} H_nA_n + \nu(0), \tag{11}$$

where

$$H_n = \int_{-\infty}^{\infty} g(\tau - nT) \times H(-\tau)d\tau. \tag{12}$$

In an ideal system, eq. (11) would yield $D(0) = A_0 \times$ constant. This result is obtained when

1. $H_0 =$ constant $\times I$ (I is the identity matrix), which implies that the flat, or nondispersive, CPI vanishes;
2. $H_n = [0]$, ([0] is the zero matrix), implying that CPI, as well as ISI, vanishes; and
3. $\nu(0) = 0$.

Clearly, these requirements cannot be achieved in practice, and the designer of data communications systems must deal with these impairments and find methods that minimize their effects on system performance.

A well-known approach[2] is the use of linear equalization. Our objective here is to investigate a general cancellation technique in conjunction with linear equalization, which could potentially yield better performance than with just the linear equalizer alone. To this end, we begin our analysis by placing a linear matrix filter in cascade with the channel prior to sampling, and we choose its characteristics so as to minimize the total MSE between the actual output sample and the desired output after canceling some CPI and ISI.

Denote the matrix filter impulse response by $W(t)$ and evaluate its output at $t = 0$. This yields the column vector for the overall system response

$$D_0(0) = U_0 A_0 + \sum_{\substack{n \\ n \neq 0}} U_n A_n + \nu_0, \tag{13}$$

where

$$U_n = \int_{-\infty}^{\infty} W(-\tau) H_0(\tau - nT) d\tau,$$

$$H_0(t) = \int_{-\infty}^{\infty} g(\tau) H(t - \tau) d\tau, \tag{14}$$

and

$$\nu_0 = \int_{-\infty}^{\infty} W(-\tau) \nu(\tau) d\tau. \tag{15}$$

2.2 The optimization problem

To describe our approach, we first discuss the following statistical problem. Suppose that one observes the vector $D_0(0)$, eq. (13), and wishes to design the best processing strategy that estimates A_0 in a sense of minimizing the probability of error. The precise solution to this problem remains intractable because of the non-Gaussian nature of ISI and CPI. While the precise mathematical solution is unknown, some qualitative aspects of the solution have been discussed.[4,5] It is easy to argue that the optimal detector structure consists of a matched filter followed by a least-mean-square estimator of the interference, which is then subtracted from the matched filter output. After subtracting the estimate of the interference, the problem reduces to detecting a known signal in additive Gaussian noise, which has a well-known solution. The difficulty with this formulation, while physically appealing, is that the least-mean-square estimator of interference is just as difficult and intractable to evaluate as the detection problem originally posed. One redeeming feature of this approach, however, is that if one does not insist on least-mean-square estimation of inter-

ference, a reasonable detector structure can be determined. We argue that constructing reasonable estimates of CPI and ISI, which are not necessarily optimum, subtracting them from the incoming signal, and then constructing an optimum detector essentially satisfies the spirit of the suggested optimal procedure.

We now formulate our approach more precisely. To start, assume that over a finite set of sampling instants, S, vector data symbols, A_n, $n \in S$, are available at the receiver and before we make a final decision on the current symbol, A_0, a portion of the interference,

$$\sum_{n \in S} U_n A_n,$$

is subtracted from $D_0(0)$. Actually, this is feasible since prior to $n = 0$, symbols have been decoded all along and what is presumed in our proposal is that we use the already-decoded symbols to improve on the current estimate of A_0. Since practical systems are not realizable relative to a large delay, there is a problem in using symbols that have not yet occurred. This can be overcome by introducing a delay, making tentative decisions, and then returning to modify the A_0 decision.

How realistic is this assumption? The answer depends on the system error rate prior to cancellation. For example, when the error rate is 10^{-4} and the cancellation window size is small relative to 10^4, the probability that almost all of the symbols in this window have been correctly detected is fairly large. Thus, after cancellation, the error rate may be much improved. On the other hand, if the error rate prior to cancellation is high, no improvement after cancellation can be expected since the estimation of the interference is not reliable. Evidently, decision-directed cancellation as proposed here is a bootstrapping technique. It is very successful over a certain range of error rates and fails when the error rate is high. Unfortunately, these qualitative statements are extremely difficult to make precise, and it is necessary to rely on simulation results.[6] The assumption that A_n is known in the canceler window will clearly result in optimistic performance predictions, and whether the predicted benefits can be realized must be ascertained experimentally.

We now proceed to include this "genie" in our mathematical analysis. As already stated, the performance criterion we use throughout this work is the least MSE normalized to the transmitted symbols variance, denoted σ_d^2. This is a mathematically tractable criterion to work with, and by minimizing MSE, one also minimizes an exponentially tight upper bound on the error rate. Its use is also practically motivated because it lends itself to easy estimation, and it can be used to update transversal filter-tap coefficients in practical adaptive systems.

Returning to the mathematical problem at hand, we define the error vector ϵ as the difference between $D_0(0)$ minus the canceler output vector, and the desired vector data symbol, A_0,

$$\epsilon = U_0 A_0 + \sum_{\substack{n \\ n \neq 0}} U_n A_n - \sum_{n \in S} C_n A_n + \nu_0 - A_0, \tag{16}$$

where C_n represents canceler-tap values. Total MSE can be expressed as

$$\text{MSE} = \text{tr}[E\{\epsilon\epsilon^\dagger\}], \tag{17}$$

where "tr" stands for trace of a matrix, $E\{\cdot\}$ denotes mathematical expectation with respect to all random variables, and † represents complex conjugate transpose.

The computation of eq. (17) is straightforward and yields

$$\text{MSE} = \sigma_d^2 \text{tr}\left[I - U_0 - U_0^\dagger + \sigma^2 \int_{-\infty}^{\infty} W(t) W^\dagger(t) dt + \sum_{n \in S} (U_n - C_n)(U_n - C_n)^\dagger + \sum_{n \notin S} U_n U_n^\dagger\right], \tag{18}$$

where $I\sigma_d^2 = E\{A_n A_n^\dagger\}$, $\sigma_d^2 = 2(M-1)/3$, and M is the total number of QAM signal states,

$$N_0 I = E\{\nu(t)\nu^\dagger(t)\},$$

and $\sigma^2 = N_0/\sigma_d^2$.

The set of canceler matrices, C_n, $n \in S$, can immediately be determined. If they are not identically set to U_n, they can only increase the value of MSE. Consequently, we set $C_n = U_n$, $n \in S$, and the residual MSE results in a functional of the matrix impulse response, $W(t)$, and the size of the cancellation window.

The minimization of MSE with respect to the matrix $W(t)$ is accomplished by the use of the calculus of variations. After substituting for U_n, defined in eq. (14), we get

$$\frac{\text{MSE}}{\sigma_d^2} = \text{tr}\left[I - 2\int_{-\infty}^{\infty} W(-\tau) H_0(\tau) d\tau + \sigma^2 \int_{-\infty}^{\infty} W(-\tau) W^\dagger(-\tau) d\tau + \sum_{n \notin S} \int W(-\tau) H_0(\tau - nT) d\tau \int H_0^\dagger(\tau - nT) W^\dagger(-\tau) d\tau\right]. \tag{19}$$

To determine the optimum W, we replace the matrix W in eq. (19) by

$$(W_0)_{ij} + (\xi\eta)_{ij}, \quad i, j = 1, \cdots, 4,$$

where η_{ij} is arbitrary, and we set

$$\frac{\partial}{\partial \xi_{ij}} (\text{MSE}) = [0]_{ij} \tag{20}$$

at $\xi_{ij} = 0$, $i, j = 1, 2, 3, 4$. It is easy to verify that

$$\frac{1}{\sigma_d^2} \frac{\partial}{\partial \xi_{ij}} (\text{MSE}) = \text{tr} \Bigg[-2 \int_{-\infty}^{\infty} \eta_{ij}^0(\tau) H_0(\tau) d\tau + 2\sigma^2 \int_{-\infty}^{\infty} W_0(-\tau) \eta_{ji}^0(\tau) d\tau + 2 \sum_{n \notin S} \int_{-\infty}^{\infty} W_0(\tau) H_0(\tau - nT) d\tau \int_{-\infty}^{\infty} H_0^\dagger(\tau - nT) \eta_{ji}^0(\tau) d\tau \Bigg] = 0, \quad (i, j) = 1, \cdots, 4, \tag{21}$$

where the matrices, η_{ij}^0, $i, j = 1, \cdots, 4$ have the entry "l" in the (l_{ij})th position and zero everywhere else. By computing the trace of eq. (21), we obtain

$$- \int_{-\infty}^{\infty} [H_0(\tau)]_{ji} \eta_{ij}^0(\tau) d\tau + \sigma^2 \int_{-\infty}^{\infty} [W_0(-\tau)]_{ij} \eta_{ij}^0(\tau) d\tau + \sum_{n \notin S} \int [H_0(\tau - nT) U_n^\dagger]_{ji} \eta_{ij}^0(\tau) d\tau = 0, \quad i, j = 1, \cdots, 4. \tag{22}$$

Since eq. (22) must hold for all functions of τ and $\eta_{ij}^0(\tau)$, we obtain the matrix integral equation that must be satisfied by the optimum matrix $W_0(\tau)$, namely,

$$\sigma^2 W_0(-\tau) = H^\dagger(\tau) - \sum_{n \notin S} U_n H_0^\dagger(\tau - nT). \tag{23}$$

The structure of $W_0(\tau)$ is practically interesting. It consists of a matched filter followed by a matrix-tapped delay line where the matrix taps are zero for $n \in S$. In other words, the linear transversal filter or equalizer specified in eq. (23) operates over a range of matrix-tap coefficients where the canceler is not operative. This is to avoid interaction between collocated taps and possible unstability problems. The structure is shown schematically in Fig. 2. In practice, this structure can be approximated and implemented by a finite transversal filter whose taps can be adaptively updated.

After post-multiplying eq. (23) by $W^\dagger(-\tau)$, integrating, and then comparing the result with eq. (19), we get an explicit formula for the optimum MSE,

$$\text{MSE}_0 = \sigma_d^2 \text{tr}(I - U_0), \tag{24}$$

where U_0 is obtained by solving a set of infinite linear equations obtained by post-multiplying eq. (23) by $H(\tau - kT)$ and then integrating. Thus,

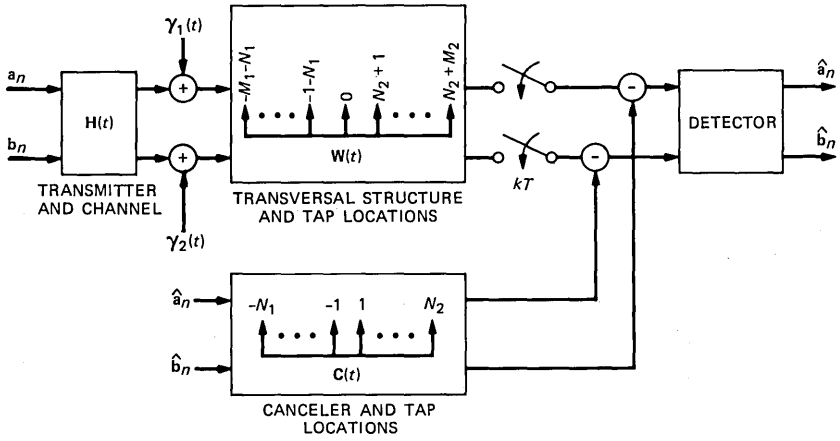


Fig. 2—Cross-polarization interference and intersymbol interference canceler block diagram.

$$\sigma^2 U_k = R_k - \sum_{n \notin S} U_n R_{k-n}, \quad \text{all } k, \tag{25}$$

where

$$R_k = \int_{-\infty}^{\infty} H^\dagger(\tau) H(\tau - kT) d\tau$$

$$= R_{-k}^\dagger. \tag{26}$$

To evaluate the merits of our system, we must have a solution for U_0. The task of solving eq. (25) is rather complicated. It is made difficult by the fact that the matrix equations are not specified over the finite set, S. While the number of unknowns is infinite, the values at the gap window are not specified. A way around this dilemma was found in the scalar case,[4] and with care applied to matrix manipulations, it is possible to adopt the same techniques here.

We proceed by first separating eq. (25) into two equations, one for $k = 0$ and the other for $k \neq 0$. Thus,

$$U_0(I\sigma^2 + R_0) = R_0 - \sum_{n \notin J} U_n R_{-n}, \quad k = 0, \tag{27}$$

and

$$\sum_{n \notin J} U_n M_{k-n} = (I - U_0) R_k, \quad k \notin J, \tag{28}$$

where the set J is defined as

$$\{J: n \in J, \quad n = -N_1, \cdots, 0, \cdots, N_2\} \tag{29}$$

and

$$M_k = R_k + \sigma^2 \delta_{0k} I, \tag{30}$$

where δ_{0k} is the Kronecker delta function. The solution of eq. (28) is facilitated by introducing a set of matrix variables $\{V_n\}_{-\infty}^{\infty}$ and a set of unknown matrices $\{\Lambda_k\}_{-\infty}^{\infty}$. Using these matrices, we write eq. (28) as

$$\sum_{n=-\infty}^{\infty} V_n M_{k-n} = (I - U_0)(R_k - \Lambda_k), \quad \text{all } k. \tag{31}$$

For these doubly infinite sets of matrix equations to identically coincide with eq. (28), the following constraints must hold:

$$\Lambda_n = 0, \quad n \notin J$$

and

$$V_n = 0, \quad n \in J. \tag{32}$$

If these can be satisfied, the solution to eq. (31) will be identical to the solution of eq. (28) with $V_n = U_n$, $n \notin J$, and this is the sole purpose for introducing new variables. Evidently, eq. (31) is easy to solve since it is in a form of a convolutional equation. To this end define the inverse matrix sequences, $\{M_n^{(-1)}\}_{-\infty}^{\infty}$, as

$$\sum_{n=-\infty}^{\infty} M_{k-n} M_n^{(-1)} = I\delta_{k0}, \quad \text{all } k. \tag{33}$$

Now, insert this into eq. (31) to obtain explicitly the desired solution,

$$V_n = (I - U_0) \sum_{k=-\infty}^{\infty} (R_k - \Lambda_k) M_{k-n}^{(-1)}, \quad \text{all } n. \tag{34}$$

From this we can obtain a finite set of equations in the unknown matrices Λ_k, since $V_n = 0$ for $n \in J$,

$$\sum_{k=-\infty}^{\infty} R_k M_{k-n}^{(-1)} = \sum_{k \in J} \Lambda_k M_{k-n}^{(-1)}, \quad n \in J. \tag{35}$$

By substituting the definition of M_k from eq. (30) into the left-hand side of eq. (35) and making use of eq. (33), we obtain the desired equations for the unknown constraint matrices Λ_k, $k \in J$,

$$I\delta_{n0} - \sigma^2 M_{-n}^{(-1)} = \sum_{k \in J} \Lambda_k M_{k-n}^{(-1)}, \quad n \in J. \tag{36}$$

Returning to eq. (31), we get for $k = 0$

$$\sum_{n=-\infty}^{\infty} V_n M_{-n} = (I - U_0)(R_0 - \Lambda_0)$$

$$= \sum_{n \notin J} U_n R_{-n}, \tag{37}$$

where the last equality derives from the fact that $V_n = 0$, $n \in J$; $V_n = U_n$, $n \notin J$; and $R_n = M_n$, $n \in J$. Finally, by substituting eq. (27) into eq. (37), we can write

$$(I - U_0)(R_0 - \Lambda_0) = R_0 - U_0(I\sigma^2 + R_0), \tag{38}$$

and solving for $I - U_0$ yields

$$(I - U_0) = \sigma^2(I\sigma^2 + \Lambda_0)^{-1}. \tag{39}$$

Substituting this into eq. (24) provides an explicit expression for MSE_0 in terms of Λ_0 only,

$$\mathrm{MSE}_0 = \sigma_d^2 \mathrm{tr}\left(I + \frac{\Lambda_0}{\sigma^2}\right)^{-1}. \tag{40}$$

Our effort in the following will be centered on determining Λ_0 as a function of the cancellation window size, or the size of set J.

2.3 The matched filter bound

When the canceler window is doubly infinite in extent, one obtains the very best possible result. In other words, the genie has eliminated all ISI and CPI. In this special case, $N_1 = -\infty$ and $N_2 = \infty$, and eq. (36) is now easy to solve since it reads

$$I\delta_{n0} - \sigma^2 M_{-n}^{(-1)} = \sum_{k=-\infty}^{\infty} \Lambda_k M_{k-n}^{(-1)}. \tag{41}$$

By evaluating the Fourier series of both sides of eq. (41), we obtain

$$I = \sigma^2 M^{(-1)}(\theta) + \Lambda(\theta) M^{(-1)}(\theta), \tag{42}$$

where a generic Fourier series pair representation is

$$X(\theta) = \sum_{l=-\infty}^{\infty} x_l \exp(j\theta l)$$

and

$$x_l = \frac{1}{2\pi} \int_{-\pi}^{\pi} X(\theta) \exp(-j\theta l) d\theta.$$

Since $M(\theta) = \sigma^2 I + R(\theta)$ and $M^{(-1)}(\theta)$ is in fact the inverse, $M^{-1}(\theta)$, we determine from eq. (42) that $\Lambda(\theta) = R(\theta)$. Consequently, the zeroth coefficient of $\Lambda(\theta)$ is $R_0 = 1/(2\pi) \int_{-\pi}^{\pi} R(\theta) d\theta$, and when this is substituted into eq. (40) we get the desired matched filter bound,

$$\mathrm{MSE}_0 = \sigma_d^2 \mathrm{tr}\left(I + \frac{R_0}{\sigma^2}\right)^{-1}. \tag{43}$$

This will serve as a lower bound to attainable performance to which we will compare all other results.

2.4 Linear equalization

In this case, the canceler is absent and so $N_1 = N_2 = 0$. Here, eq. (36) reduces to

$$I - \sigma^2 M_0^{(-1)} = \Lambda_0 M_0^{(-1)}, \tag{44}$$

and solving for $M_0^{(-1)}$, we get

$$\begin{aligned} M_0^{(-1)} &= \frac{1}{\sigma^2}\left(I + \frac{\Lambda_0}{\sigma^2}\right)^{-1} \\ &= \frac{1}{2\pi}\int_{-\pi}^{\pi} M^{-1}(\theta)d\theta = \frac{1}{2\pi\sigma^2}\int_{-\pi}^{\pi}\left[I + \frac{R(\theta)}{\sigma^2}\right]^{-1} d\theta. \end{aligned} \tag{45}$$

It is now immediate that

$$\mathrm{MSE}_0 = \frac{\sigma_d^2}{2\pi}\int_{-\pi}^{\pi} \mathrm{tr}\left(I + \frac{R(\theta)}{\sigma^2}\right)^{-1} d\theta, \tag{46}$$

the well-known formula for linear equalization.[2]

2.5 Decision feedback and finite causal canceler

In this application it is assumed that all the causal terms, which depend only on past decisions, are canceled in addition to a finite number of noncausal terms. This implies that $N_2 = \infty$ and N_1 is finite. When $N_1 = 0$, the canceler becomes a decision feedback equalizer[7] since causal interference can be canceled by a feedback circuit. Here, we will determine MSE for the more general case when N_1 is not necessarily zero.

To treat this case it is more convenient to solve for U_0 directly from eq. (28) rather than through eq. (36). Thus, we rewrite eq. (28) as

$$\sum_{k=-\infty}^{-N_1} U_k M_{m-k} = (I - U_0)R_m, \quad m \le -N_1, \tag{47}$$

which is recognized to be a matrix Wiener-Hopf equation, and its solution depends on being able to factor positive definite Hermitian matrices.[8]

To proceed with the solution of eq. (47), we introduce the following sequence of matrices

$$\begin{aligned} M_n^+ &= [0], \quad n < 0 \\ M_n^- &= [0], \quad n \ge 0, \end{aligned}$$

such that

$$M_m = \sum_{n=0}^{\infty} M_{m-n}^- M_n^+, \quad \text{all } m. \tag{48}$$

The validity of this expression and the existence of M_n^+ and M_n^- were first proved by Wiener and Akutowicz.[9]

Substituting eq. (48) into eq. (47) and rearranging yields two sets of equations

$$\sum_{n=0}^{\infty} y_{m-n} M_n^+ = (I - U_0) R_m, \quad \text{all } m \tag{49}$$

and

$$\sum_{k=-\infty}^{-N_1} U_k M_{m-k}^- = y_m, \quad m \leq -N_1. \tag{50}$$

The procedure for solving these is to first solve for $Y(\theta)$ from eq. (49) in terms of $M^-(\theta)$, an easy task in terms of the Fourier transforms of $\{M_n^-\}$ and $\{y_n\}$. Having obtained $Y(\theta)$, one proceeds to solve eq. (50) for $U(\theta)$ in terms of $M^\dagger(\theta)$. Note that eq. (48) implies

$$M(\theta) = M^-(\theta) M^+(\theta),$$

and since $M(\theta)$ is Hermitian, $M(\theta) = M^\dagger(\theta)$, implying $[M^+(\theta)]^\dagger = M^-(\theta)$, $[M^-(\theta)]^\dagger = M^+(\theta)$, and the factorization problem is reduced to finding a matrix $M^+(\theta)$ such that

$$M(\theta) = [M^+(\theta)]^\dagger M^+(\theta),$$

where the entries in $M^+(\theta)$, $[M^+(\theta)]_{ij}$ are such that $[M^+(\theta)]_{ij}$ has a Fourier series with only positive frequency coefficients. We shall later discuss algorithms for determining $M^+(\theta)$ from $M(\theta)$—a rather complicated task.[10]

We now proceed to determine the sequence y_m. Multiply both sides of eq. (50) by M_{-m}^+ and sum m from $-\infty$ to $-N_1$. This gives the formula

$$\sum_{m=-\infty}^{-N_1} y_m M_{-m}^+ = \sum_{k=-\infty}^{-N_1} U_k M_{-k}. \tag{51}$$

Now, recall that $M_k = R_k + \sigma^2 \delta_{k0} I$, and, therefore, eq. (49) can be put into the form

$$\sum_{n=0}^{\infty} y_{m-n} M_n^+ = (I - U_0) R_m, \quad \text{all } m, \quad m \neq 0. \tag{52}$$

When this is compared with eq. (48), we obtain

$$y_m = (I - U_0) M_m^-, \quad m \neq 0, \tag{53}$$

and when substituted into eq. (51), we get

$$(I - U_0) \sum_{m=-\infty}^{-N_1-1} M_m^- M_{-m}^+ = \sum_{m=-\infty}^{-N_1-1} U_m M_{-m} = (I - U_0) \sum_{m=-\infty}^{-N_1-1} M_m^- (M_m^-)^\dagger. \tag{54}$$

From eq. (37) we have that

$$\sum_{n=-\infty}^{-N_1-1} U_n M_{-n} = (I - U_0)(R_0 - \Lambda_0), \tag{55}$$

and so we conclude that

$$R_0 - \Lambda_0 = \sum_{n=-\infty}^{-N_1-1} M_n^- (M_n^-)^\dagger. \tag{56}$$

Substituting again for $R_0 = M_0 - I\sigma^2$ in eq. (56) and rearranging, we finally obtain

$$I\sigma^2 + \Lambda_0 = \sum_{n=-N_1}^{0} M_n^- (M_n^-)^\dagger, \tag{57}$$

since

$$\begin{aligned} M_0 &= \sum_{n=0}^{\infty} M_{-n}^- M_n^+ \\ &= \sum_{n=0}^{\infty} M_{-n}^- (M_{-n}^-)^\dagger \\ &= \sum_{n=-N_1}^{0} M_n^- (M_n^-)^\dagger; \end{aligned}$$

hence,

$$\begin{aligned} I\sigma^2 + \Lambda_0 &= -\sum_{n=-\infty}^{-N_1-1} M_n^- (M_n^-)^\dagger + \sum_{n=-\infty}^{0} (M_n^-)(M_n^-)^\dagger \\ &= \sum_{n=-N_1}^{0} M_n^- (M_n^-)^\dagger. \end{aligned}$$

Upon substituting eq. (57) into eq. (40), we obtain the desired result,

$$\begin{aligned} \mathrm{MSE}_0 &= \sigma_d^2 \mathrm{tr} \left(I + \frac{\Lambda_0}{\sigma^2} \right)^{-1} \\ &= \sigma_d^2 \mathrm{tr} \left(\frac{1}{\sigma^2} \sum_{n=-N_1}^{0} M_n^- (M_n^-)^\dagger \right)^{-1}. \end{aligned} \tag{58a}$$

Notice that when $N_1 = 0$, that is, no anticausal cancellation,

$$\mathrm{MSE}_0 = \sigma_d^2 \mathrm{tr} \left[\frac{M_o^2}{\sigma^2} \right]^{-1}, \tag{58b}$$

where $M_0 = M_0^- = (M_0^+)^\dagger$. This is the formula for decision feedback equalization derived by Falconer and Foschini for QAM transmission over a single channel, which they cast in a matrix formulation.[11] Evidently, the form of the answer generalizes to arbitrary dimensions.

2.6 *Finite linear equalizer*

The theoretical results we have derived so far apply to an ideal canceler of any window size and an infinite-tap linear equalizer whose matrix taps vanish inside the cancellation window. To assess the penalties incurred by a finite-tap linear equalizer outside the cancellation window, we derive least MSE formulas applicable to this case. With these formulas we will be in a position to evaluate the merits of equalization/cancellation using only a finite number of matrix taps and to gain insight as to how best to deploy the total number of available taps. Also inherent in the theory derived so far is the independence of MSE on sampling phase. This is so since the transversal equalizer/canceler is preceded by a matched filter whose structure presumes knowledge of sampling phase. Here, we shall relax this condition and derive the MSE for a front-end filter matched to the transmitter filter only rather than to the overall channel response and, thereby, bring out the dependence of MSE on timing phase.

We, thus, represent the finite-tap delay line matrix filter by

$$W_{f0}(-\tau) = \sum_{n\in F} g(t - nT)Q_n, \tag{59}$$

where the two sets F and S are disjoint and F now is a finite set,

$$\{F\colon n \in F,\ n = -N_1 - M_1, \cdots, -N_1 - 1, 0, N_2 + 1, \cdots, N_2 + M_2\}.$$

In eq. (59), $g(t)$, as before, is a scaler pulse shape, while $\{Q_n\}_{n\in F}$ is a 4×4 matrix sequence. The objective now is to select the Q_n's that minimize the total MSE, eq. (19),

$$\frac{\text{MSE}}{\sigma_d^2} = \text{tr}\left[I - 2U_0 + \sigma^2 \int_{-\infty}^{\infty} W_f(\tau) W_f^\dagger(\tau) d\tau + \sum_{n\notin S} U_n U_n^\dagger\right]. \tag{60}$$

Substituting eq. (59) into eq. (60) yields

$$\frac{\text{MSE}}{\sigma_d^2} = 2 - 2 \sum_{n\in F} \text{tr}(Q_n H_{-n}) + \sum_{n,m\in F} \text{tr}(Q_n G_{nm} Q_m^\dagger) + \sigma^2 \sum_{n,m\in F} \text{tr}(Q_n \rho_{n-m} Q_m^\dagger), \tag{61}$$

where the H_n's are defined in eq. (12) and

$$\begin{cases} G_{nm} = \sum_{l\notin S} H_{l-n} H_{l-m}^\dagger \\ \rho_n = \int_{-\infty}^{\infty} g(t) g(t - nT) dt. \end{cases} \tag{62}$$

Setting the derivatives of eq. (61) with respect to the elements of the matrices $\{Q_n\}_{n\in F}$ to zero, we get a set of linear matrix equations for the unknowns, $\{Q_n\}_{n\in F}$, namely,

$$H^{\dagger}_{-n} = \sum_{l \in F} Q_l R_{ln}, \quad n \in F, \tag{63}$$

where

$$R_{ln} = G_{ln} + \sigma^2 \rho_{n-l}, \quad n, l \in F. \tag{64}$$

The solution of eq. (63) is straightforward and is discussed in a later section.

For now, label the solution of eq. (63) by Q_l^0—the optimal Q_l's. Premultiply by Q_n^0, sum over $n \in F$, and substitute the result into eq. (63). This yields the desired formula for the least-mean-square error,

$$\mathrm{MSE}_0 = \sigma_d^2 \mathrm{tr} \left(I - \sum_{l \in F} Q_l^0 H_{-l} \right). \tag{65}$$

The next section will present computation algorithms for numerically evaluating the formulas developed here.

III. COMPUTATIONAL ALGORITHMS

An examination of Section III demonstrates that the theoretical analysis of M-QAM signal transmission over dually polarized channels in the presence of multipath fading is a numerically intensive activity. In this section we provide an overview of the major computational issues related to our investigation.

3.1 Infinite linear equalizer/finite canceler

When the linear equalizer in Fig. 2 has a finite-tap window size, the optimum receiver structure comprises a matched filter followed by a matrix transversal filter and a matrix canceler. The most general case under this assumption is when the matrix canceler has a finite number of causal and anticausal taps and the solution of eq. (36) for Λ_0 provides a means of calculating minimum mean-square error by use of eq. (40). To solve for Λ_0, block matrices $M_k^{(-1)}$'s defined in eq. (33) have to be determined first. One way to determine the $M_k^{(-1)}$'s is to solve eq. (33) by a Levinson-type algorithm[12] where the entries are block matrices. Thus, matrix convolution eq. (33) is then represented as

$$\begin{bmatrix} M_0 & M_{-1} & M_{-2} & \cdots & \cdot \\ M_1 & M_0 & M_{-1} & \cdots & \cdot \\ M_2 & M_1 & M_0 & \cdots & \cdot \\ \cdot & \cdot & \cdot & & \cdot \\ \cdot & \cdot & \cdot & & M_{-1} \\ \cdot & \cdot & \cdot & M_1 & M_0 \end{bmatrix} \times \begin{bmatrix} \cdot \\ \cdot \\ M_{-1}^{(-1)} \\ M_0^{(-1)} \\ M_1^{(-1)} \\ \cdot \end{bmatrix} = \begin{bmatrix} 0 \\ \cdot \\ 0 \\ I \\ 0 \\ 0 \end{bmatrix}, \tag{66}$$

where I is the identity matrix. As observed, the block Toeplitz matrix

equation can be solved for $M_k^{(-1)}$'s, with the M_k's given in eq. (30). Having the $M_k^{(-1)}$'s and expressing eq. (36) in the form

$$[\Lambda_{-N_1}\Lambda_{-N_1+1}\cdots\Lambda_{-1}\Lambda_0\Lambda_1\cdots\Lambda_{N_2}] \times \begin{bmatrix} M_0^{(-1)} & M_{-1}^{(-1)} & \cdots\cdots & M_{-(N_1+N_2)}^{(-1)} \\ M_1^{(-1)} & M_0^{(-1)} & \cdots\cdots & \vdots \\ \vdots & & \cdots\cdots & \vdots \\ \vdots & & & \vdots \\ \vdots & & & \vdots \\ \vdots & & & \vdots \\ M_{(N_1+N_2)}^{(-1)} & M_{(N_1+N_2-1)}^{(-1)} & \cdots\cdots & M_0^{(-1)} \end{bmatrix} = [-\sigma^2 M_{N_1}^{(-1)} \cdots (-\sigma^2 M_0^{(-1)} + I) \cdots -\sigma^2 M_{-N_2}^{(-1)}], \tag{67}$$

it is possible to evaluate Λ_0.

3.2 Infinite linear equalizer/decision feedback canceler

When the matrix canceler has knowledge of infinite past data symbols, it becomes a decision feedback equalizer. In addition, it may also employ a finite number of anticausal taps to operate on the future symbols, in which case it becomes a finite window canceler. This can be accomplished by a finite delay. As shown in eq. (47), to determine MSE_0, a matrix Wiener-Hopf equation has to be solved. This involves determination of anticausal factors of the $M(\theta)$ matrix as explained in Section 2.5.

There are at least two computational algorithms available for solving a matrix Wiener-Hopf equation. One method as introduced in Ref. 13 converts the matrix that has to be factored directly into a nonlinear difference equation of a Ricatti type, which converges to a stable solution. Another method, which we adopt in our present work, is a Bauer-type factorization of positive definite polynomial matrices.[14] This algorithm is suited to sampled data applications and takes advantage of the periodic and positive nature of the channel covariance matrix, $M(\theta)$, as in this work. It performs the factorization in the following steps. Suppose one desires to factor the $n \times n$ matrix $M(\theta)$ as follows:

$$M(\theta) = M^{-}(\theta)M^{+}(\theta).$$

This matrix possesses a Fourier series expansion,

$$M(\theta) = \sum_{m=-\infty}^{\infty} A_m \exp(jm\theta), \tag{68}$$

whose $n \times n$ coefficients,

$$A_m = \frac{1}{2\pi}\int_{-\pi}^{\pi} \exp(-jm\theta)M(\theta)d\theta, \tag{69}$$

approach 0_m as $|m|$ becomes infinitely large. One now follows the steps:

Step 1: Form the following variable-size Toeplitz matrix

$$T_m = \begin{bmatrix} A_0 & A_1 & \cdots & A_m \\ A_{-1} & A_0 & \cdots & A_{m-1} \\ \vdots & & & \vdots \\ A_{-m} & A_{-m+1} & \cdots & A_0 \end{bmatrix}, \tag{70}$$

of respective size $(m+1)n \times (m+1)n$, $m = 0, \cdots, \infty$ that is Hermitian and real.

Step 2: For every $m \geq 0$, perform the Cholesky's factorization[15]

$$T_m = L_m^T \cdot L_m, \tag{71}$$

where L_m^T is the transpose of L_m, and L_m is a square, real, and lower triangular matrix with positive diagonal scalar entries,

$$L_m = \begin{bmatrix} L_{00}^{(m)} & & & & & & \\ L_{10}^{(m)} & L_{11}^{(m)} & & & & & \\ \vdots & \vdots & \ddots & & & & 0 \\ \vdots & \vdots & & \ddots & & & \\ \vdots & \vdots & & & \ddots & & \\ L_{m0}^{(m)} & L_{m1}^{(m)} & \cdots & \cdots & \cdots & \cdots & L_{mm}^{(m)} \end{bmatrix}. \tag{72}$$

All blocks in eq. (72) are real $n \times n$, and all $L_{rr}^{(m)}$, $r = 0, \cdots, m$, are lower triangular.

Step 3: It has been proved in Ref. 14 that for every fixed r and k, $r \geq k \geq 0$,

$$\lim_{m \to \infty} L_{rk}^{(m)} = M_{r-k}^{+}. \tag{73}$$

The factorization of eq. (71) and the evaluation of M_{r-k}^{+} in eq. (73) are performed numerically. To find an upper bound on m for stopping the calculations, a convergence point must be established. This can be done by checking the trace of M_{r-k}^{+} in each iteration to determine whether it has reached a level of constancy and, if so, for what value of m. This completes the factorization. Indeed, in Ref. 14 it is shown that a constant trace as a function of m corresponds to a minimum of a quadratic functional:

$$I(P) = \frac{1}{2\pi} \int^{\pi} \mathrm{tr}[P^{\dagger}(\theta) M(\theta) P(\theta)] d\theta, \tag{74}$$

where $P(\theta)$ is a polynomial matrix of the form

$$P(\theta) = \sum_{r=0}^{m} \chi_r \exp(-jr\theta), \tag{75}$$

with the quadratic functional of eq. (74) expressed as

$$I(P) = \mathrm{tr}(X^{\dagger} T_m X), \tag{76}$$

where T_m was defined in eq. (70) and χ_r's represent the elements of X. Hence, there is a theoretical base for establishing the convergence point.

3.3 Finite linear equalizer/finite canceler

Finally, we consider the case where the matrix linear equalizer operates on a finite set of taps that do not overlap with those of the finite-tap matrix canceler. This is a case of great practical interest. Here, the receive filter is assumed to have a square-root-Nyquist transfer function[16] matching the transmit filter. Since it no longer matches the overall channel and transmitter characteristics, MSE_0 is a function of timing phase. Therefore, an optimum timing reference has to be established before the optimum nonstationary covariance matrix can be determined. This is accomplished here by minimizing the mean-square eye closure (MS-EC), which is a measure of the amount of received level perturbation caused by CPI and ISI.[16] In our present work it is assumed that the demodulator removes the channel phase at the optimum sampling time reference.[16] Once an optimal set of samples is found, the covariance matrix, G_{nm}, of eq. (62) is formed as

$$\underset{n,\, m \in F}{G_{nm}} =$$

$$\begin{bmatrix}
G_{-(N_1+M_1),-(N_1+M_1)} & G_{-(N_1+M_1),-(N_1+M_1-1)} & \cdots & G_{-(N_1+M_1),(N_2+M_2)} \\
G_{-(N_1+M_1-1),-(N_1+M_1)} & & & \vdots \\
\vdots & & & \vdots \\
\vdots & & & \vdots \\
\vdots & & & \vdots \\
G_{-(N_1+1),-(N_1+M_1)} & & & \vdots \\
G_{0,-(N_1+M_1)} & & & \vdots \\
G_{(N_2+1),-(N_1+M_1)} & & & \vdots \\
\vdots & & & \vdots \\
\vdots & & & \vdots \\
\vdots & & & \vdots \\
G_{(N_2+M_2),-(N_1+M_1)} & \cdots\cdots\cdots\cdots & \cdots & G_{(N_2+M_2),(N_2+M_2)}
\end{bmatrix} . \tag{77}$$

In terms of the H_n's defined in eq. (12) the covariance matrix can be expressed as

$$G_{nm} = \sum_{l \notin S} \begin{bmatrix} H_{l+N_1+M_1} \\ \vdots \\ H_{l+N_1+1} \\ H_l \\ H_{l-N_2-1} \\ \vdots \\ H_{l-N_2-M_2} \end{bmatrix} \times [H^\dagger_{l+N_1+M_1} \cdots H^\dagger_{l+N_1+1} H^\dagger_l H^\dagger_{l-N_2-1} \cdots H^\dagger_{l-N_2-M_2}]. \tag{78}$$

Hence, by adding σ^2 to the diagonal elements of G_{nm}, the matrix R_{nm} is formed, as expressed in eq. (64). The Q_n's, that is, the coefficients of the finite window equalizer, can be computed as follows:

$$[Q_{-(N_1+M_1)} \cdots Q_{-N_1-1} Q_0 Q_{N_2+1} \cdots Q_{N_2+M_2}] = [H^\dagger_{(N_1+M_1)} H^\dagger_{(N_1+M_1-1)} \cdots H^\dagger_{-(N_2+M_2)}] \times [R_{nm}]^{-1}. \tag{79}$$

These coefficients are used in eq. (65) to determine the optimum MSE.

IV. DISCUSSION OF SIMULATIONS AND NUMERICAL RESULTS

In this section, the minimum mean-square error (MSE_0) is evaluated for the various techniques covered in the previous sections. We will first discuss a channel model, and then we will exhibit and discuss the behavior of MSE_0 as a function of the number of equalizer/canceler taps (M_1, M_2, N_1, N_2) (see Fig. 2).

4.1 Propagation model

The cross-polarization fading propagation model employed is the one that is proposed in Ref. 2 and is briefly reviewed here. The frequency characteristics of the propagation model are presented by the complex matrix

$$C(\omega) = \begin{bmatrix} C_{11}(\omega) & C_{21}(\omega) \\ C_{12}(\omega) & C_{22}(\omega) \end{bmatrix}, \tag{80}$$

where the functional form of $C_{11}(\omega)$ and $C_{22}(\omega)$ is that of a single (in line) fading channel model documented by Rummler[17] with the generic representation

$$C_{11}(\omega) = a[1 - \rho \exp(j\phi)\exp(-j\omega\tau)], \tag{81}$$

where a and ρ are real variables representing flat and dispersive fading levels, ϕ is related to the fade notch offset, and τ is the delay between

direct and reflected paths assumed to be 6.3 ns in this study. Also in the model,

$$C_{22}(\omega) = a[1 - \rho \exp(j\phi)\exp(-j(\omega - \Delta\omega)\tau)], \tag{82}$$

which is in the same form as $C_{11}(\omega)$, except for an additional variable $\Delta\omega$ that allows noncollocated fade notches to occur on the two polarization signal transfer characteristics. From Ref. 2, cross-polarized paths are assumed to behave as

$$C_{21}(\omega) = K_1 C_{11}(\omega) + K_2 C_{22}(\omega) + R_3 \exp(-j\omega D_1) \tag{83}$$

and

$$C_{12}(\omega) = K_4 C_{11}(\omega) + K_5 C_{22}(\omega) + R_6 \exp(-j\omega D_2), \tag{84}$$

where K_1, K_2, K_4, and K_5 are constants that incorporate the nonideal properties of antennas and waveguide feeds at both ends of the channel, typically taking on values varying from one hop to another in the -35 to -20 dB range. The last term in eqs. (83) and (84) represents a nondispersive cross-polarization response contributed by an independent ray. In the present work, R_3, R_6, and $\Delta\omega$ are assumed to be zero and the K_i's are assumed to be -20 dB.

4.2 Channel covariance computation

Computation of the channel covariance matrix is the initial necessary step behind all the MSE_0 calculations. In the case of the infinite window-size equalizer discussed in Sections 3.3 through 3.5, the receive filter is assumed to be a matched filter; hence, no reference timing establishment is necessary. The peak of the correlation function serves as a timing reference. By computing the sampled correlation matrix of eq. (26), we can proceed with the normalized MSE_0 calculations as explained in previous sections.

By applying the finite window equalizer, as discussed in Section 3.6, a set of optimum samples of overall impulse response is found by establishing a timing reference, t_0, for which the MS-EC of the received in-line signal is a minimum, and at this reference, the channel phase is removed.[16] This has to be done for the two polarized signals independently.

The overall transfer function matrix is given by

$$H(\omega) = C(\omega) \times P(\omega), \tag{85}$$

where $C(\omega)$ is the propagation transfer matrix and $P(\omega)$ is the diagonal Nyquist-shaping filter transfer matrix. Now, for instance, if the impulse response of the vertical in-line signal is

$$h_{i11}(t) = a[p(t) - \rho \exp(j\phi)p(t - \tau)], \tag{86}$$

where $p(t)$ is a Nyquist-shaped pulse, the channel phase becomes

$$\theta(t) = \text{Arc tg} \frac{-\rho \sin(\phi)p(t-\tau)}{p(t) - \rho \cos(\phi)p(t-\tau)}, \tag{87}$$

and the upper row block matrices of the overall impulse response matrix have to be multiplied by $\exp(-j\theta(t_0)) \times I$ (I being the unity matrix) in order to remove the channel phase at t_0. With this background, we now present the numerical results in the following subsection.

4.3 Numerical results

To provide a single set of curves for MSE_0, independent of the number of transmit states in M-QAM signal space, we normalize MSE_0 as defined in eqs. (40), (43), (46), (58a), (58b), and (65) by dividing the formulas by σ_d^2, that is, the transmitted symbols variance. In addition, we only compute the normalized MSE_0 for one of the M-QAM signals that comprise the dually polarized signals, namely, $S_v(t)$.

If one defines the unfaded signal-to-noise ratio (s/n) by Γ, it can be verified that in the case of a matched filter receiver, the normalized MSE_0 in the absence of any cross-polarization interference (K_1, K_2, K_4, K_5, R_3, $R_6 = 0$) is simply

$$\frac{1}{\sigma_d^2}\text{MSE}_0 = \frac{1}{1+\Gamma} \tag{88}$$

and, consequently, for a large unfaded s/n, it becomes Γ^{-1}. Hence, eq. (88) establishes an ultimate performance bound that can only be achieved in a utopian environment. This reference will be our baseline in the following evaluations. In a dually polarized system with a finite amount of nondispersive coupling (K_1, K_2, K_4, $K_5 > 0$), the matched filter bound is degraded somewhat. For $K_1 = K_2 = K_4 = K_5 = -20$ dB we found a small amount of degradation in the ideal MSE_0, which is not a function of the dispersive fade depth and only diminishes when there is no cross-coupling, that is, in a completely orthogonal system.

In all that follows it is assumed that the transmit filter is square-root Nyquist shaped,[16] and the receive filter either matches the overall transmitter and channel or the transmitter only. A Nyquist roll-off of 45 percent, both a 40- and a 22-MHz channel bandwidth, and an s/n of 63 dB are used in our numerical evaluations.

Figure 3 depicts the normalized MSE_0 as a function of the number of canceler taps, Q, when a 40-dB centered fade over a 22-MHz channel band is applied to both polarized signals. The linear equalizer in this case possesses an infinite number of taps. The case of pure linear equalization ($N_1 = N_2 = 0$), no cancellation, exhibits the largest MSE_0

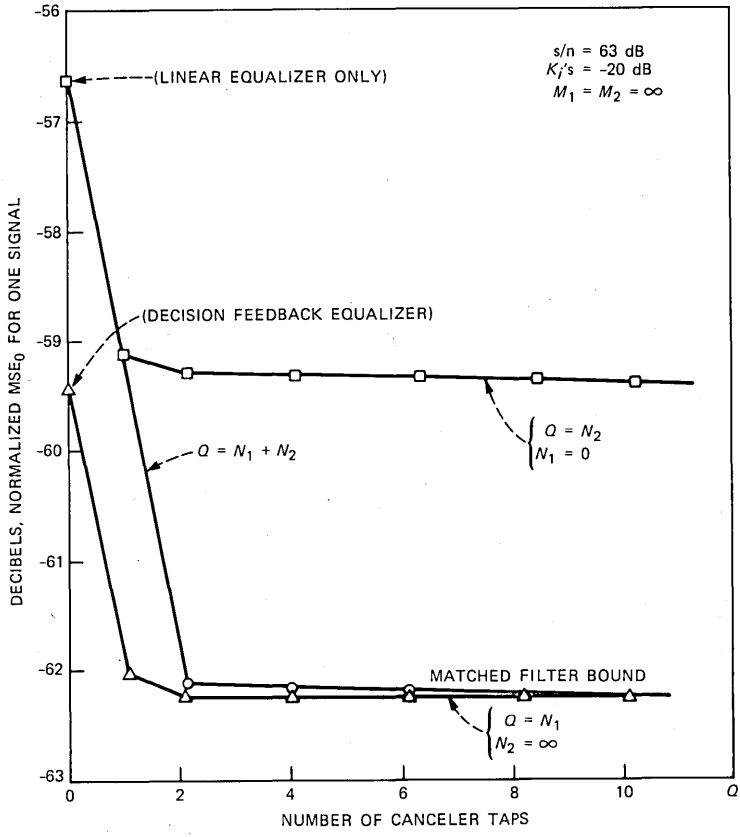


Fig. 3—Optimum normalized MSE versus number of canceler taps for a 40-dB centered fade over a 22-MHz channel.

degradation relative to the asymptotic matched filter bound. This is due to the noise enhancement experienced by the linear equalizer during deep fades. When both causal and anticausal canceler taps are present, all the curves rapidly approach the matched filter bound for a finite constant coupling ($K_i = -20$ dB, $i = 1, 2, 4, 5$). The curve for a decision feedback type canceler starts at an ideal decision feedback equalizer normalized MSE_0 and approaches the asymptotic value with two anticausal taps. The finite window size canceler curve starts at the linear equalizer case ($N_1 = N_2 = 0$) and reaches the matched filter bound asymptotic value with a total of four causal/anticausal taps. Finally, when no anticausal taps are employed the curve asymptotically approaches the ideal decision feedback case with only two causal taps.

In Fig. 4, we depict results similar to Fig. 3 for the case when the centered fade notch depth is reduced to 20 dB over a 22-MHz channel.

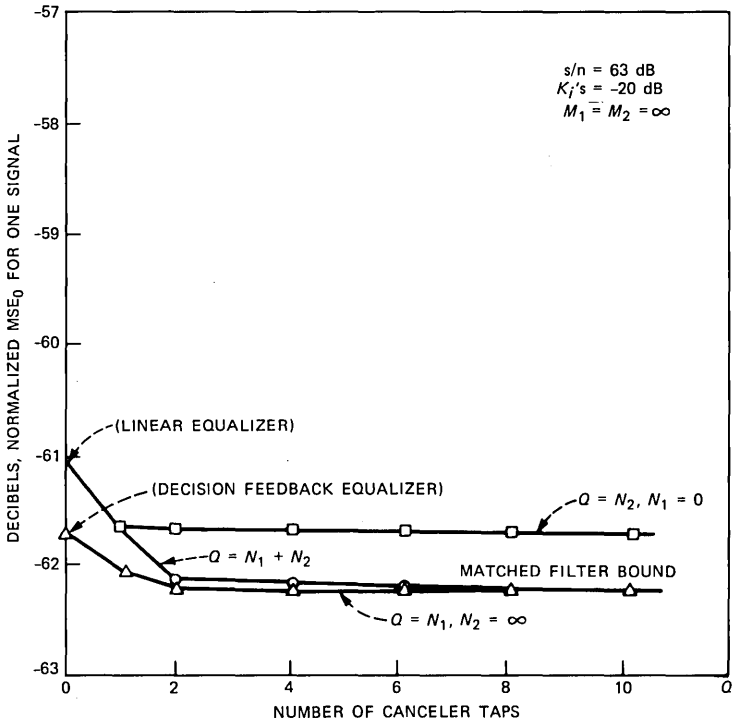


Fig. 4—Optimum normalized MSE versus number of canceler taps for a 20-dB centered fade over a 22-MHz channel.

As can be observed, the linear equalizer ($N_1 = N_2 = 0$) performance is improved. In both figures the fade notch is located at the band center; however, since in both cases the receive filter matches the overall channel and transmitter, an offset fade notch does not have a serious impact on the results for the same fade notch depth.[18]

In Figs. 5 and 6 we depict the achievable MSE_0 when the linear equalizer has a finite number of taps. The fade notch in Fig. 5 is centered, but in Fig. 6 it is offset from the band center. For ease of presenting the results in our work, fade notch offset from the band center is expressed in terms of the ratio of the fade notch distance from the band center to the channel equivalent baseband bandwidth in percentage. In Fig. 6, the fade notch is offset by 69 percent over a 22-MHz channel, that is, an offset of 7.6 MHz from the band center. As observed from Fig. 5, a total of nine taps (including the center tap) are required to achieve the asymptotic matched filter bound when decision feedback taps are present. It is interesting to note that the same asymptotic performance can be achieved no matter how the nine synchronously spaced taps are deployed between the linear equalizer

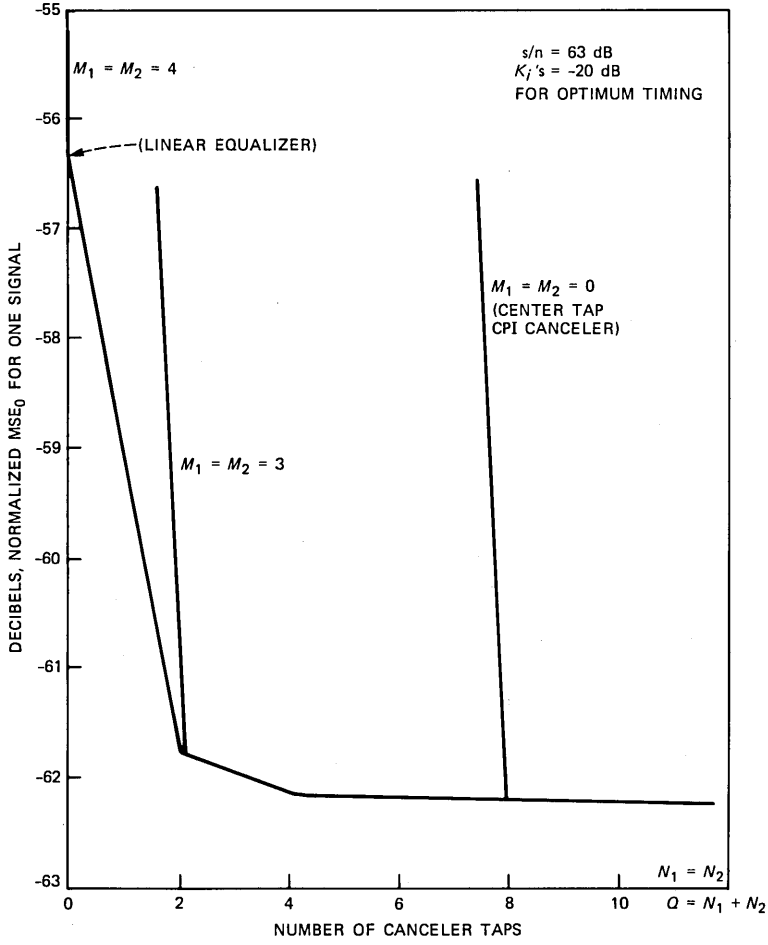


Fig. 5—Optimum normalized MSE versus number of canceler taps for a 40-dB centered fade over a 22-MHz channel.

and the canceler as long as the canceler operates in a decision feedback mode. This is because the equalizer and canceler-tap windows complement one another; therefore, since the taps do not overlap, for the same number of taps, the performance remains almost the same in the decision feedback cases. An important configuration is when the linear equalizer operates only on the main lobe of CPI by means of its center matrix taps ($M_1 = M_2 = 0$). This is a single tap linear equalizer structure as opposed to the single tap decision feedback CPI canceler proposed by Kavehrad.[3] It is clear that as long as the canceler window is sufficiently wide, a main lobe CPI canceler can achieve the asymptotic matched filter bound. The curves again indicate that deep fades degrade the linear equalizer ($N_1 = N_2 = 0$) performance significantly.

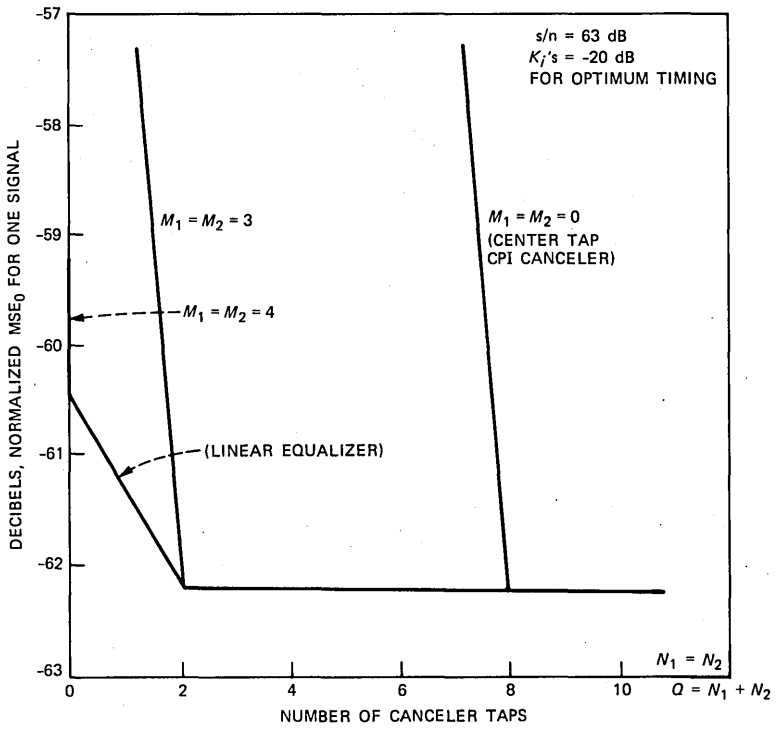


Fig. 6—Optimum normalized MSE versus number of canceler taps for a 40-dB, 69-percent offset fade over a 22-MHz channel.

Previous studies[1,2] showed that every 3-dB degradation in MSE_0 translates into a loss of 1 bit/s/Hz of data rate efficiency. Hence, linear equalization may not provide adequate rate efficiency in deep fades.

In Fig. 6 we depict curves similar to Fig. 5, except for a 40-dB offset fade with the notch frequency offset by 69 percent. Improved performance turns out to be due to the particular notch position as is brought out in the discussion of Fig. 8.

Figure 7 illustrates a similar set of curves for a 20-dB centered fading of dually polarized signals over both a 22- and a 40-MHz channel. As can be observed, the linear equalizer ($N_1 = N_2 = 0$) performance improves because of the decreased fade depth; however, over the wider channel band the degradation over decision feedback is more, as expected. This is due to the wider channel band over which the same fade notch depth causes more dispersion. The degradation amounts to 2.2-dB loss of MSE_0 comparing to a matched filter bound, that is, roughly 1 bit/s/Hz loss of data rate efficiency, and the loss can even be more for offset fades, as will be seen in Fig. 8. Hence, even

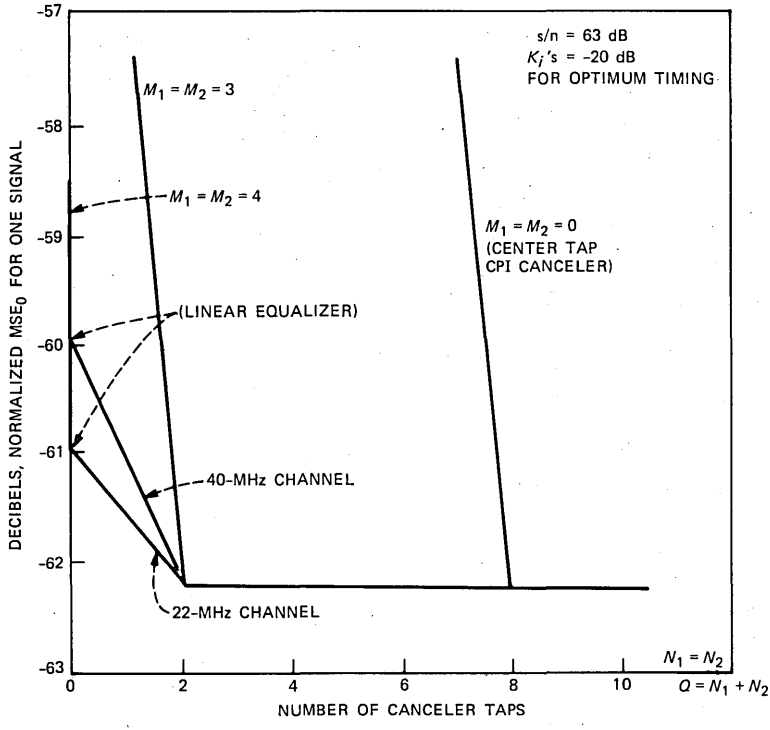


Fig. 7—Optimum normalized MSE versus number of canceler taps for a 20-dB centered fade over a 22-MHz and a 40-MHz channel.

with more typical fades the use of the linear equalizer can be troublesome over a 40-MHz channel.

Finally, to compare some of the techniques described earlier in terms of their sensitivity to fade notch offset, we plot, in Fig. 8, the normalized MSE_0 as a function of fade notch position which, as explained earlier, is expressed here in terms of the ratio of the fade notch distance from the band center to the channel equivalent baseband bandwidth. We consider the following structures:

1. A linear equalizer with

$$M_1 = M_2 = 4$$

$$N_1 = N_2 = 0 \text{ (no cancellation).}$$

2. Center tap only linear equalizer/finite window canceler with

$$M_1 = M_2 = 0$$

$$N_1 = N_2 = 4.$$

The sensitivity of the linear equalizer to offset fades is quite pro-

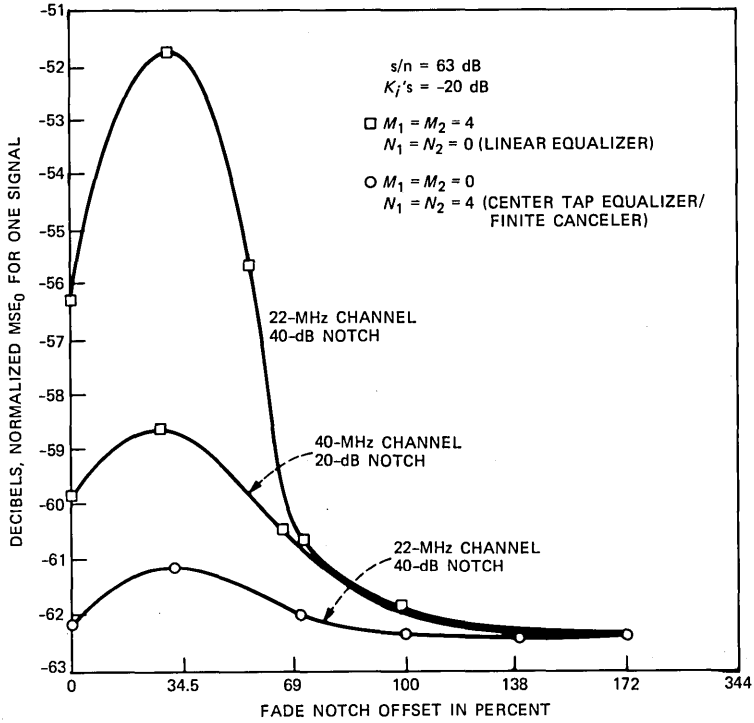


Fig. 8—Normalized MSE_0 versus fade notch offset.

nounced. The center tap equalizer with a finite window canceler exhibits a very small sensitivity to offset fades. The degradation of MSE_0 for some offset fades can be explained considering the fact that these fades cause cross-coupling of the imaginary part of a complex QAM signal into its real part, and for a particular notch offset frequency within the band, the coupling reaches its maximum. Therefore, the MSE_0 versus fade notch offset curves exhibit this phenomenon. In dually polarized systems, as in the case of the problem at hand, this is even more pronounced than in single signal transmission, because in the 4 × 4 system under offset fading there is coupling of three interfering data streams into the fourth one. A decision-feedback-type canceler structure, by canceling the major contributors to CPI and ISI and with a lesser noise enhancement, exhibits an improved performance compared with the linear equalizer. Note that all the curves in Fig. 8 have been obtained under optimum timing conditions.

To investigate the sensitivity of the two structures to timing phase, we plot in Fig. 9 the normalized MSE_0 and superimpose the normalized MS-EC of the received signal before equalization/cancellation as a function of sample timing offset from the optimum timing reference.

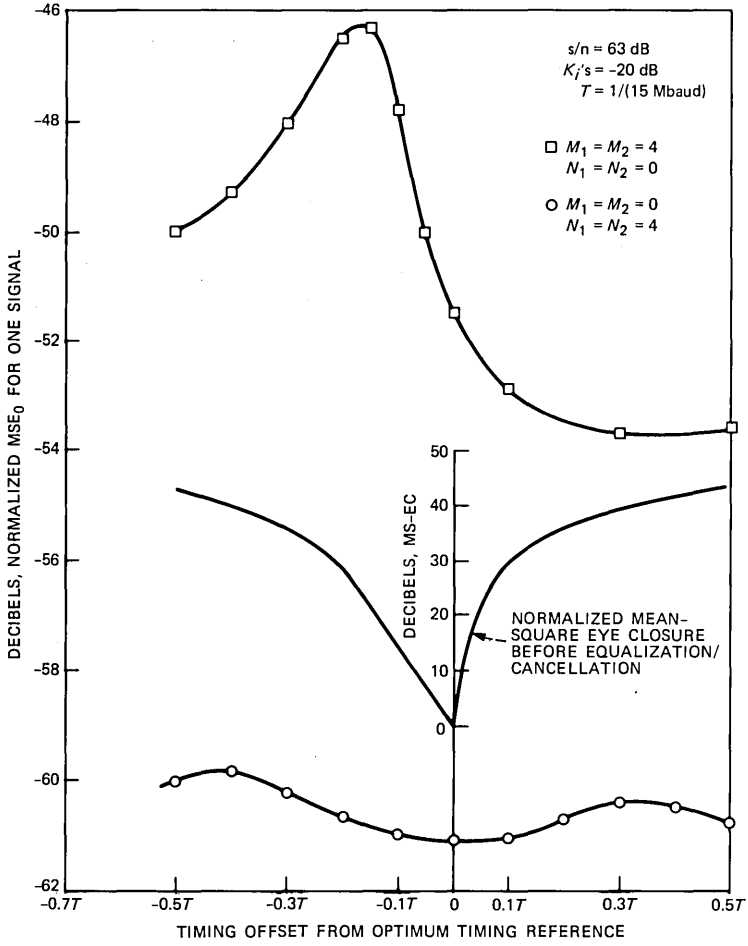


Fig. 9—Sensitivity to timing phase for a 40-dB fade notch, offset by 34.5 percent, over a 22-MHz channel.

This is done for a severe fade, namely, a 40-dB fade with a notch frequency offset by 34.5 percent over a 22-MHz channel. It is clear that the finite linear equalizer is much more sensitive to timing phase than the decision feedback type. This was previously shown in a paper by J. Salz[19] for infinite window structures in single-signal transmission. We demonstrated the concept here for dual-polarization transmission and finite window architectures. Notice that in Fig. 9 the optimum timing reference is established based on minimizing the MS-EC of the received signal in presence of fading, before CPI and ISI cancellation; hence, after cancellation occurs this timing reference may not be the one that minimizes the canceler output MSE, and

indeed the linear equalizer curve on Fig. 9 indicates this fact. As seen, the MS-EC curve has a minimum at the optimum timing reference. The sensitivity of the matrix linear equalizer to timing phase can be reduced by applying half-a-baud spaced taps, that is, by deploying fractionally spaced taps.[20] Also, a decision feedback timing method may prove more robust[21] than the minimum MS-EC timing that has been adopted in our work. Needless to say, that decision feedback timing method is more complex in terms of implementation than the minimum MS-EC timing, which can essentially be implemented at intermediate frequency. The degradation in MSE_0 seen in Fig. 9 is partly attributed to the asymmetric amplitude and delay responses of the fading channel that in the presence of a nonzero roll-off shaping filter cause a destructive addition of aliases.[16]

V. SUMMARY AND CONCLUSIONS

Current work generalizes and extends previous results[2,4] in the following respects. Data-aided decision feedback and canceler structures, known to be effective in single-channel data transmission, are adopted and included in our class of receiver structures. As a practical feature, we admit transversal filter realizations with a finite number of matrix taps and pay attention to timing phase recovery.

Because of the departure from ideal linear infinite structures considered previously, we encountered extremely difficult numerical problems, which we addressed and solved.

The dually polarized digital radio channel is modeled as a four-input port, four-output port linear network followed by additive noise. We determine the optimum admissible receiver structures when the transmitted signals are two independent M-state QAM digital data signals.

The mathematically tractable criterion, the MSE, is used throughout our work. This figure of merit has several redeeming features in addition to its being mathematically tractable. For one, it can be used to determine a sharp upper bound on error rate. More importantly, it is the quantity which is estimated in practice to provide information for updating tap coefficients in adaptive systems.

The receiver structure that minimizes the MSE consists of a matrix matched filter in cascade with a transversal filter combined with an intersymbol interference as well as a cross-polarization interference canceler. The canceler uses the detected data symbols to estimate the interference to be canceled. This is a major assumption on which our results rest. Since data-aided operations presume correct knowledge of detected data symbols and since wrong decisions will be occasionally accepted, our proposal is necessarily a boot-strapping approach. Thus, cancellation is only feasible when tentative decisions are correct most

of the time, and yet the error rate is not sufficiently low to meet system specifications. Our approach makes possible the reduction of the final error rate to an acceptable level. In circumstances where the initial error rate is very high ($\geq 10^{-2}$), our proposal will not work, and in order to ensure availability of reliable data symbols, one option is to dedicate a small fraction of the main data frame to a sequence a priori known to both transmitter and receiver for proper acquisition of data.

We use the assumption of the availability of correct data symbols to derive our main results. These are expressions for minimum attainable MSE as a function of various system parameters and numerical algorithms for evaluating the mathematical formulas—a rather intensive activity because of the large number of matrix equations that has to be solved. Inclusion of the effects of errors in the feedback/canceler loops has proved so far to be mathematically intractable.

From our extensive numerical work, which is exhibited in a sequence of graphs, we draw these major conclusions:

1. For a reasonable copolarized and cross-polarized propagation model[2] and a severe centered fade—40-dB notch depth with a secondary ray delay of 6.3 ns over an approximately 22-MHz channel bandwidth—the performance of transversal filters with a finite number of taps deployed in a decision feedback/canceler structure is substantially (6 dB) better than linear equalization, and the difference can be up to 10 dB for offset fades. It can be shown that a 3-dB increase in MSE translates into about 1 bit/s/Hz decrease in data rate at a fixed error rate or an order of magnitude increase in outage probability. Hence, linear equalization may not be adequate in deep fades. Whether this gain can be realized in practice depends on the degree of error propagation. This is difficult to assess mathematically and must be studied by computer simulation and/or by experimentation.

2. Decision feedback/canceler structures achieve the ultimate matched filter bound with only nine matrix taps provided that error propagation is neglected.

3. Nine linear equalizer taps essentially achieve the performance of the infinite-tap linear equalizer. This method is, of course, free of error propagation.

4. For milder centered fades—20-dB depth with a secondary ray delay of 6.3 ns—the linear equalizer configuration with nine taps is only 1-dB inferior to the decision feedback structure over a 22-MHz channel. However, if the channel bandwidth is increased to 40 MHz, the performance of the linear equalizer is worse than that of the decision feedback structure by 2.2 dB, and the difference can be up to 3 dB for offset fades.

5. Decision feedback/canceler configurations are less sensitive to timing phase than linear structures.

VI. ACKNOWLEDGMENTS

Discussions with N. Amitay; G. J. Foschini; L. J. Greenstein; D. J. Goodman; T. Kailath and Hanoch Lev-Ari of Stanford University; N. Kazanjian; A. M. Saleh; and D. C. Youla of Polytechnic Institute of New York were extremely valuable during the course of this work.

REFERENCES

1. G. J. Foschini and J. Salz, "Digital Communication Over Fading Radio Channels," B.S.T.J., *62,* No. 2, Part I (February 1983), pp. 429–59.
2. N. Amitay and J. Salz, "Linear Equalization Theory in Digital Data Transmission Over Dually Polarized Fading Radio Channels," AT&T Bell Lab. Tech. J., *63,* No. 10 (December 1984), pp. 2215–59.
3. M. Kavehrad, "Baseband Cross-Polarization Interference Cancellation for M-Quadrature Amplitude-Modulated Signals Over Multipath Fading Radio Channels," AT&T Tech. J., *64,* No. 8 (October 1985), pp. 1913–26.
4. M. S. Mueller and J. Salz, "A Unified Theory of Data-Aided Equalization," B.S.T.J., *60,* No. 11 (November 1981), pp. 2023–38.
5. T. Kailath, "A General Likelihood-Ratio Formula for Random Signals in Gaussian Noise," IEEE Trans. Inform. Theory, *IT-15,* No. 3 (May 1969), pp. 350–61.
6. A. Gersho and T. L. Lim, "Adaptive Cancellation of Intersymbol Interference for Data Transmission," B.S.T.J., *60,* No. 11 (November 1981), pp. 1997–2020.
7. J. Salz, "Optimum Mean-Square Decision Feedback Equalization," B.S.T.J., *52,* No. 8 (October 1973), pp. 1341–73.
8. D. C. Youla, "On the Factorization of Rational Matrices," IRE Trans. Inform. Theory, *IT-7* (July 1961), pp. 172–89.
9. N. Wiener and E. J. Akutowicz, "A Factorization of Positive Hermitian Matrices," J. Math. Mech., *8* (August 1959), pp. 111–20.
10. M. C. Davis, "Factoring the Spectral Matrix," IEEE Trans. Automat. Contr., *AC-8* (October 1963), pp. 296–305.
11. D. D. Falconer and G. J. Foschini, "Theory of Minimum Mean-Square-Error QAM Systems Employing Decision Feedback Equalization," B.S.T.J., *52,* No. 10 (December 1973), pp. 1821–49.
12. N. Levinson, "The Wiener RMS (Root Mean Square) Error Criterion in Filter Design and Prediction," J. Math. Phys., *25* (February 1946), pp. 261–78.
13. W. G. Tuel, "Computer Algorithm for Spectral Factorization of Rational Matrices," IBM J. Res. Develop., *12* (March 1968), pp. 163–70.
14. D. C. Youla and N. Kazanjian, "Bauer-Type Factorization of Positive Matrices and the Theory of Matrix Polynomials Orthogonal on the Unit Circle," IEEE Trans. Circuits Syst., *CAS-25,* No. 2 (February 1978), pp. 57–69.
15. G. H. Golub and J. H. Welsch, "Calculations of Gauss Quadrature Rules," Math. Comp., *26,* No. 106 (April 1969), pp. 221–30.
16. R. W. Lucky, J. Salz, and E. J. Weldon, *Principles of Data Communication,* New York: McGraw-Hill, 1968.
17. W. D. Rummler, "A New Selective Fading Model: Application to Propagation Data," B.S.T.J., *58,* No. 7 (May–June 1979), pp. 1037–71.
18. M. Kavehrad, "Adaptive Decision Feedback Cancellation of Intersymbol Interference Over Multipath Fading Radio Channels," Proc. ICC (June 1983), pp. 869–71.
19. J. Salz, "On Mean-Square Decision Feedback Equalization and Timing Phase," IEEE Trans. Commun., *COM-25,* No. 12, pp. 1471–76, December 1977.
20. G. Ungerboeck, "Fractional Tap-Spacing Equalizer and Consequences for Clock Recovery in Data Modems," IEEE Trans. Commun., *COM-24,* No. 8 (August 1976), pp. 856–64.
21. G. R. McMillen, M. Shafi, and D. P. Taylor, "Simultaneous Adaptive Estimation of Carrier Phase, Symbol Timing, and Data for a 49-QPRS DFE Radio Receiver," IEEE Trans. Commun., *COM-32,* No. 4 (April 1984), pp. 429–43.

AUTHORS

Mohsen Kavehrad, B.S. (Electrical Engineering), 1973, Tehran Polytechnic Institute; M.S. (Electrical Engineering), 1975, Worcester Polytechnic Institute; Ph.D. (Electrical Engineering), 1977, Polytechnic Institute of New York; Fairchild Industries, 1977–1978; GTE, 1978–1981; AT&T Bell Laboratories, 1981—. At AT&T Bell Laboratories Mr. Kavehrad is a member of the Communications Methods Research Department at Crawford Hill Laboratory. His research interests are digital communications and computer networks. He has organized and chaired sessions for IEEE sponsored conferences. Technical Editor, IEEE Communications Magazine; Chairman, IEEE Communications Chapter of New Hampshire, 1984. Member, IEEE, Sigma Xi.

Jack Salz, B.S.E.E., 1955, M.S.E., 1956, and Ph.D., 1961, University of Florida; AT&T Bell Laboratories, 1961—. Mr. Salz first worked on the electronic switching system. Since 1968 he has supervised a group engaged in theoretical studies in data communications and is currently a member of the Communications Methods Research Department. During the academic year 1967–1968, he was on leave as Professor of Electrical Engineering at the University of Florida. He was a visiting lecturer at Stanford University in Spring 1981 and a visiting MacKay Lecturer at the University of California, at Berkeley, in Spring 1983.

AT&T Technical Journal
Vol. 64, No. 10, December 1985
Printed in U.S.A.

Cross-Polarization Interference Cancellation and Nonminimum Phase Fades

By M. KAVEHRAD*

(Manuscript received January 14, 1985)

In this paper we examine the effects of nonminimum phase fades on dual-polarized M-state quadrature amplitude-modulated signals. Performance is evaluated in the presence of a baseband cross-polarization interference canceler described in another paper. Results indicate that the canceler performance is practically transparent to the fade type.

I. INTRODUCTION

This paper reports on a continuation of the work presented in Ref. 1, that is, cancellation of cross-polarization interference by a transversal structure at baseband when two M-state quadrature amplitude-modulated (M-QAM) signals are transmitted over the same channel using orthogonal field polarizations. In this paper we extend the scope of the previous study, which only dealt with minimum phase fades, to include evaluation of nonminimum phase fade effects encountered in transmission over multipath fading channels such as those experienced in line-of-sight terrestrial radio applications.

Recent studies[2-4] have examined IF and baseband equalizers in the transmission of a single QAM signal under nonminimum phase fades. For example, Ref. 4 examines transition distortions when the fade phase changes from minimum to nonminimum. In particular, Ref. 2 concludes that, in mitigating intersymbol interference, baseband equalizers outperform IF structures used for this purpose in the

* AT&T Bell Laboratories.

presence of nonminimum phase fades. Since, as pointed out in Ref. 2, on the average 50 percent of deep selective fades take on a nonminimum phase, we examine the proposed baseband interference canceler performance[1] in nonminimum phase fades. We use the static modeling of the dual-polarized channels introduced in Ref. 5 to emulate snapshots of fadings of the main (reference) and the cross-coupled paths. This simple model follows the results of recent measurements cited in Ref. 5.

In this study, we first show that some of the nonminimum phase fade channel models available in the literature that were meant to be used in single-signal transmission may not be appropriate for application in multicarrier transmission systems such as dual-polarized or space-diversity systems. Then, employing what seems to be the proper static model for nonminimum phase fades, we evaluate dual-polarized system performance with and without the proposed canceler.[1] It is demonstrated that, for a proper model, the type of fading has no impact on the performance of the baseband cross-polarized interference canceler, as long as the fadings of the reference (main) copolarized path signal and the cross-coupled path signal are of the same type, i.e., both minimum or both nonminimum phase. As in Ref. 1, dual-polarized system performance signature (M-curve) is used as a measure in the performance evaluation.

In the following section, we describe and compare the nonminimum phase-fading static models and select what seems to be the appropriate one to carry on the numerical evaluation in Section III.

II. ANALYTICAL CHANNEL MODEL

The underlying assumptions in modeling and cross-polarization cancellation are described.

2.1 Nonminimum phase fading model

In this section we compare three commonly used static models of nonminimum phase fades and demonstrate that only one of them seems to be appropriate for application in dual-polarized systems. The first model, which has been used in Ref. 4 and some other studies, states that if for minimum phase fades we employ a two-ray transfer function shown by

$$H(\omega) = a[1 - b \exp(-j(\omega - \omega_0)\tau)], \qquad 0 \le b \le 1, \tag{1}$$

in which b represents the fade notch depth and ω_0 is its displacement from the midband frequency, τ is the delay between the arrival times of the two rays, and a is the flat fade level, then for nonminimum phase fades we have to use

$$H(\omega) = a[1 - b' \exp(-j(\omega - \omega_0)\tau)], \qquad 1 \le b' \le 2, \tag{2}$$

in which b' is the fade notch depth. The second model suggests using

$$H(\omega) = a[b - \exp(-j(\omega - \omega_0)\tau)], \qquad 0 \le b \le 1 \tag{3}$$

for nonminimum phase fades, i.e., replacing the magnitude of the reference by that of the delayed ray. Finally, the third model, introduced in Ref. 6, prescribes the following transfer function under nonminimum phase fades:

$$H(\omega) = a[1 - b \exp(+j(\omega - \omega_0)\tau)], \qquad 0 \le b \le 1. \tag{4}$$

Comparing equations (1) through (4), it is obvious that the magnitudes of the transfer functions are the same at the fade notch frequency in all three different models and are all equal to

$$|H(\omega)| = a[1 + b^2 - 2b \cos(\omega - \omega_0)\tau]^{1/2}. \tag{5}$$

For the magnitude to be the same at the fade notch frequency in eq. (2), the value of b' has to be set to $(2 - b)$ so one can compare minimum and nonminimum phase fades of equal magnitude. This stems from the fact that the minimum phase fade depth here is defined as

$$B = 10 \log(1 - b)^2, \qquad 0 \le b \le 1,$$

and for the nonminimum phase fades of eq. (2) to have comparable magnitudes with minimum phase fades, b' has to be equal to $(2 - b)$. Although the fade magnitudes are all the same, the envelope delay responses of the different transfer functions cited above are not. That is exactly why one has to be careful in choosing a model to form static nonminimum phase fades in multicarrier channels. The envelope delay response as a function of frequency for the minimum phase transfer characteristic in eq. (1) can be found by using

$$T(\omega) = -\frac{d}{d\omega}\Phi(\omega),$$

where $\Phi(\omega)$ is the transfer function phase response; hence,

$$T_{\text{MPF}}(\omega) = \frac{b\tau[b - \cos(\omega - \omega_0)\tau]}{1 + b^2 - 2b \cos(\omega - \omega_0)\tau}, \qquad 0 \le b \le 1, \tag{6}$$

where the subscript MPF stands for Minimum Phase Fade. In the following, for simplicity we use midband fades to prove our point, i.e., when ω in eq. (6) is equal to ω_0. The conclusions arrived at, however, are felt to be general. Hence, the envelope delay of a midband minimum phase fade from eq. (1) is

$$T_{\text{MPF}}(\omega_0) = \frac{b\tau}{b - 1}, \qquad 0 \le b \le 1. \tag{7}$$

For the first, second, and third nonminimum phase fade transfer functions the envelope delay response is

$$T_{\mathrm{NMPF}}(\omega_0) = -T_{\mathrm{MPF}}(\omega_0) + 2\tau, \tag{8}$$

$$T_{\mathrm{NMPF}}(\omega_0) = -T_{\mathrm{MPF}}(\omega_0) + \tau, \tag{9}$$

and

$$T_{\mathrm{NMPF}}(\omega_0) = -T_{\mathrm{MPF}}(\omega_0), \tag{10}$$

respectively, where the subscript NMPF stands for nonminimum phase fade. Notice that the first and the second models introduce constant delays of 2τ and τ in the envelope delay response, respectively, and these delays have nothing to do with the nonminimum phase fade nature. To prove the point, in Fig. 1 we show the overall impulse response of a single minimum phase faded channel using a 7.5-dB midband fade notch. As seen, the optimum timing reference, t_{opt}, is to the left of the unfaded signal optimum timing reference, i.e., the time origin. This is of course due to the negative delay characteristic. Using a 7.5-dB midband nonminimum phase faded transfer function of the third model, the overall impulse response of the system for 45-percent roll-off Nyquist-shaped transmit/receive filters is shown in Fig. 2. Note that the optimum time reference is now to the right of the time origin owing to the positive delay character, and the impulse response is time reversed about the timing reference when compared to the corresponding one for the minimum phase fade of Fig. 1. The first or the second model will translate to the right the

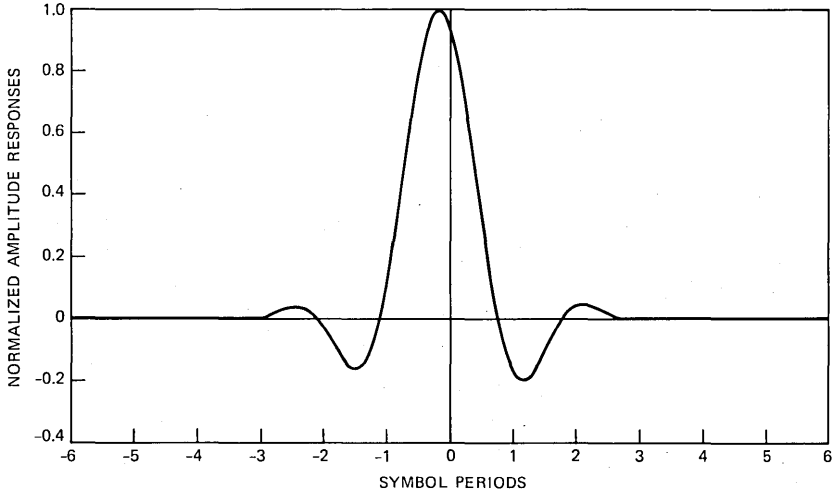


Fig. 1—In-phase received impulse response of a single-carrier system for a minimum phase fade.

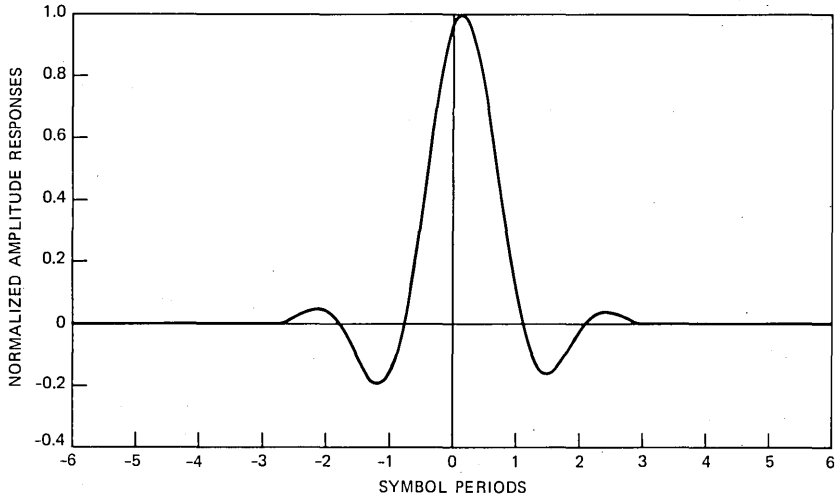


Fig. 2—In-phase received impulse response of a single-carrier system for a nonminimum phase fade.

position of the optimum timing reference of the nonminimum phase faded impulse response of Fig. 2 by 2τ or τ, respectively. Such a translation is harmless in single-carrier systems because once the optimum timing reference is established, taking samples of the impulse response at every baud period results in the same set of samples for minimum and nonminimum phase fades regardless of the actual position of the timing reference. However, in dual-polarized, or generally in multicarrier systems, the position of the timing reference can be as important as the sampling points in time. To demonstrate this, we use the dual-polarized channel model shown in Fig. 1 of Ref. 1, and illustrate the received in-phase impulse responses formed by the main copolarized and the cross-coupled paths in Fig. 3. These responses are composed of the real part of the main-path signal $U_{i,\mathrm{I}}$, cross-coupling of the main-path quadrature phase part $U_{q,\mathrm{I}}$, cross-coupled in-phase part of the cross-coupled path $U_{i,\mathrm{II}}$, and cross-coupled quadrature phase part of the cross-coupled path $U_{q,\mathrm{II}}$. Using notations of Refs. 1 and 5, the in-phase received waveforms shown in Fig. 3 correspond to 7.5-dB midband minimum phase fading of the main-polarization path and a cross-coupled path fading of (−20., 0., 0.),* that is, an interfering cross-coupled signal with a flat fading level of 20 dB below the main-polarized signal level. Hence, the main-path impulse response timing

* This triplet describes the fading status of the cross-coupled path. The first number shows the flat fading level of the interfering signal compared to that of the main-path signal. The second number is the dispersive fade notch depth, and the third one is the fade notch position.

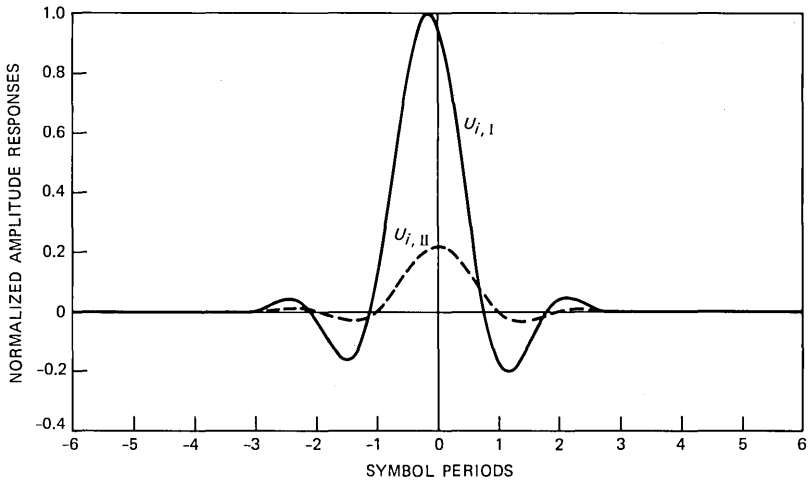


Fig. 3—In-phase received impulse responses of dual-polarized system for a midband minimum phase fade on the main path and a flat fade on the cross-coupled path.

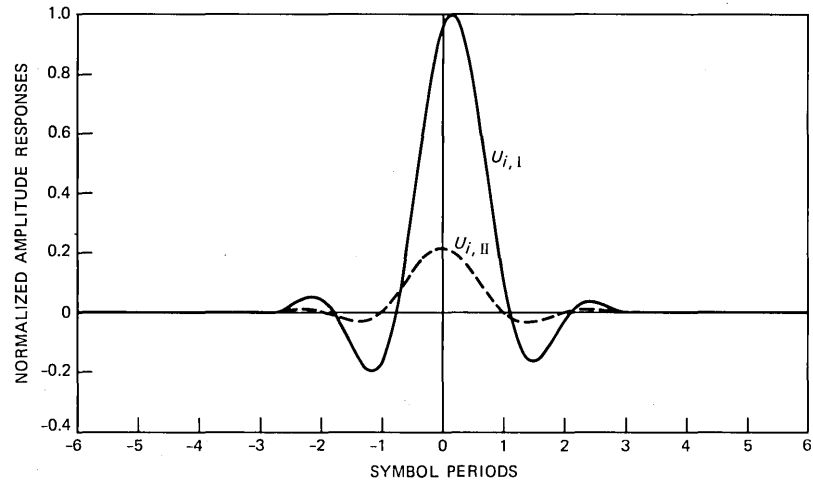


Fig. 4—In-phase received impulse responses of dual-polarized system for a midband nonminimum phase fade on the main path and a flat fade on the cross-coupled path.

reference is to the left of the time origin because of its minimum phase fading, and the cross-polarized path impulse response is an attenuated Nyquist pulse centered at the origin because of its flat fading, as expected. The corresponding shapes for an equivalent nonminimum phase fade of the main path, using the third model, are shown in Fig. 4. As seen, the interferer impulse-response shape does not change; however, the main-path impulse-response timing reference moves to the right of the time origin, and the impulse response shape itself is

time reversed. Now in the dual-polarized system, once the receiver timing recovery circuit locates the optimum time reference by minimum-peak-distortion or minimum-mean-square error criterion, the main and the interfering impulse responses are sampled using the same reference timing, as explained in detail in Ref. 5; hence, the unnecessary time shifts inherent to the first and the second nonminimum phase fade models will result in establishing an incorrect reference point and a false set of samples of the cross-coupled interfering impulse responses. For further illustration, the displacement of the optimum timing reference from the origin as a function of the fade notch offset for different fade notch depths is shown in Fig. 5. Clearly, the upper half-plane represents nonminimum phase and the lower half corresponds to minimum phase fades. We employed the second nonminimum phase fade transfer function model in this case. As seen, the two sets of curves are vertically symmetrical except for a constant time shift of 6.3 ns which happens to be the value of τ used in this example. The constant time shift is what the transfer function model

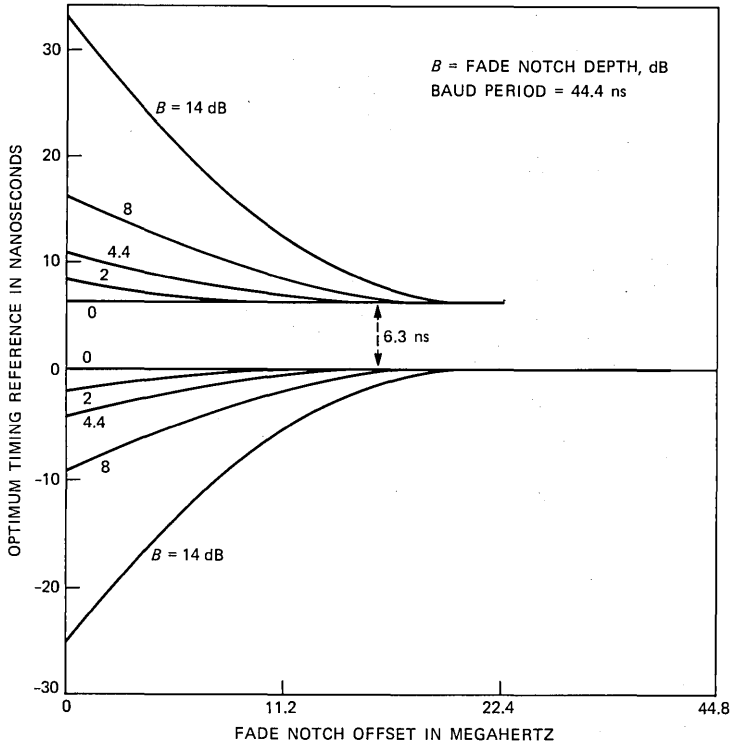


Fig. 5—Constant time shift in optimum timing reference by the second nonminimum phase fade model.

imposes on the overall impulse response. The method of establishing the optimum reference timing point in all the figures shown is the minimum-peak-distortion, trial-and-error method, described in Ref. 5. To avoid the unnecessary time shifts introduced by the first and second nonminimum phase transfer function models, we will use the third model in the canceler performance evaluation of Section III.

2.2 System model

The system model and cross-coupled interference canceler are the same as those introduced in Figs. 1 and 6 of Ref. 1, respectively. The channel model allows for a single frequency-selective fade notch, both in the reference copolarized and the cross-coupled paths; therefore, it requires the two-ray model parameters of both paths.

The canceler structure shown in Fig. 6 of Ref. 1 operates on the baseband received digitized signals. The main-lobe estimators and decision circuits together make preliminary estimates of the main lobe of the impulse response of the reference copolarized path. These estimators can be made of two to three tap transversal equalizers, and decisions made by detection circuits are provided to the cross-polarization canceler transversal filter to form and then eliminate the interfering main lobe from the opposite polarization received signal that has been properly delayed. Then, the cross-coupled interference canceled signals are equalized for mitigating intersymbol interference and cross-rail interference. The tap coefficients of the main-lobe estimators can either be derived from the preliminary decision circuit error signals or, to get a better performance, they can be set by using the error signals of the final decision circuits, as shown in Fig. 6 of Ref. 1. The slow channel time variations allow the use of the final error signals in estimating the tap coefficients of the estimators.

III. NUMERICAL CALCULATIONS OF CANCELER PERFORMANCE IN NONMINIMUM PHASE FADES

In this section we assume two 16-QAM signals are dual-polarized and transmitted over the channel model shown in Fig. 1 of Ref. 1 assuming *nonminimum phase fading of the main path.* The cross-coupled path is assumed to be flat *or minimum phase faded.* For nonminimum phase fades we use the third model described in Section II. The transmit/receive filters are the Nyquist type of 45-percent roll-off. The symbol rate is chosen to be 22.5 Mbaud and a signal-to-noise ratio of 60 dB is assumed. Performance signature (M-curve) for a 10^{-3} symbol error rate is used in the numerical evaluation, and the Gauss quadrature method[5] is used to derive the probability of error, accurately.

Before presenting the calculated results, it might be helpful to

consider the following two examples. In the first example the main path is nonminimum phase faded with a fade notch depth of 7.5 dB and a notch offset of 11.2 MHz from the band center. The cross-coupled path in this case is assumed to be flat faded by 20 dB. Clearly, the cross-coupled path signal has a perfect Nyquist shape and is centered at the time origin. The main-path signal timing reference is to the left or the right of the origin depending on whether the main-path fade is minimum or nonminimum phase. However, since the amount by which the timing reference is translated has the same absolute value, as seen in Figs. 6 and 7, once the timing reference is found, the main and the cross-coupled signals are sampled using the same reference; hence, the same set of interferer samples results, regardless of the type of the fade. Therefore, the resulting end performance should not change. However, this is no longer true when the cross-coupled path is faded dispersively because the interfering impulse response is no longer symmetrical about the vertical axis going through the origin. This is shown in Figs. 8 and 9 where, in addition to a 7.5-dB depth and an 11.2-MHz offset fading of the main path, the cross-coupled path is faded by a 5-dB midband fade. Hence, depending on whether the main-path fading is a minimum or nonminimum phase, different sets of samples of the interfering impulse response are involved and the end performance may not necessarily remain the same.

The system performance signatures for two different fading conditions of the cross-coupled path with and without the canceler of Ref. 1 are shown in Figs. 10 and 11. In Fig. 10 it is assumed that the cross-

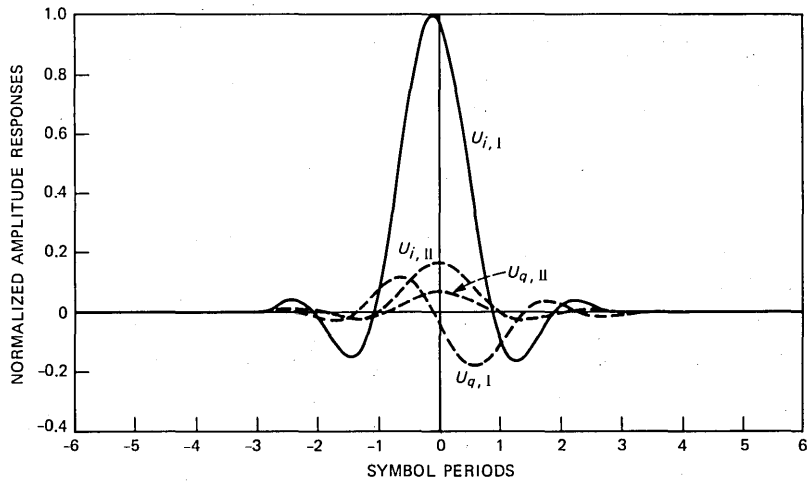


Fig. 6—In-phase received impulse responses of dual-polarized system for an offset minimum phase fade on the main path and a flat fade on the cross-coupled path.

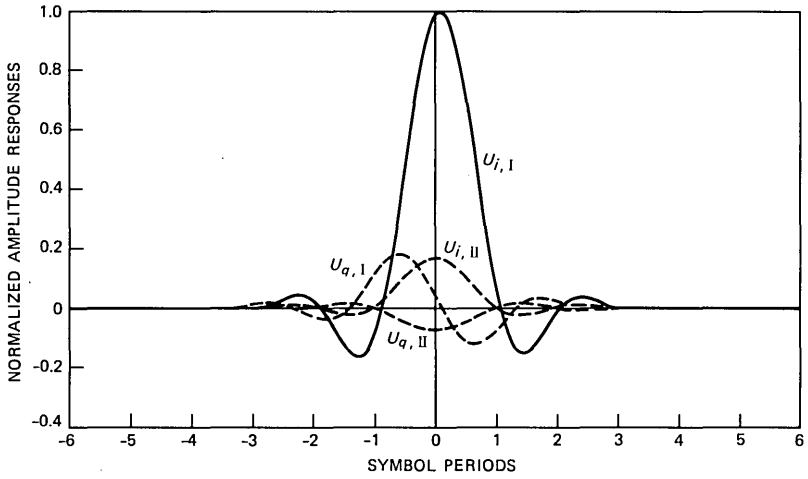


Fig. 7—In-phase impulse responses of dual-polarized system for an offset nonminimum phase fade on the main path and a flat fade on the cross-coupled path.

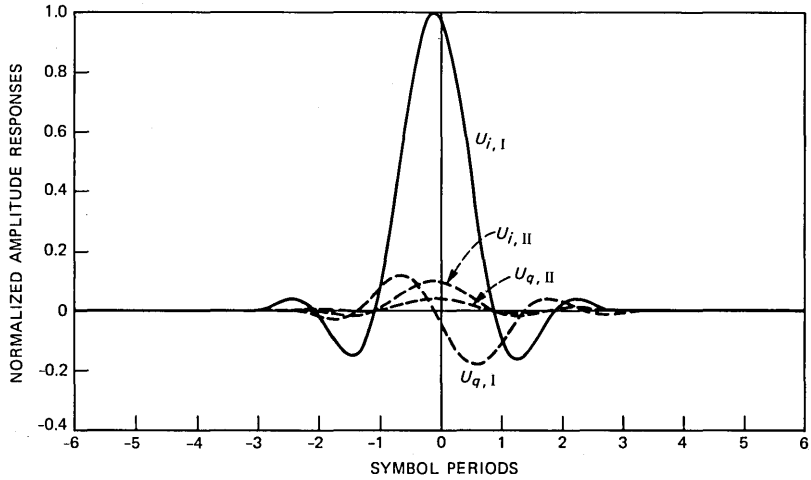


Fig. 8—In-phase impulse responses of dual-polarized system for an offset minimum phase fade on the main path and a midband minimum phase fade on the cross-coupled path.

coupled path is only flat faded; hence, the nonminimum phase performance signatures with and without the canceler are identical to the minimum phase faded signatures of Fig. 7 in Ref. 1. As seen in Fig. 10, the single-tap canceler[1] improves the dual-pol system performance to nearly that of a single-pol system shown by a dotted M-curve. For midband fades, as observed in Fig. 10, the timing reference of the main-path impulse response is offset from the peak of the main lobe

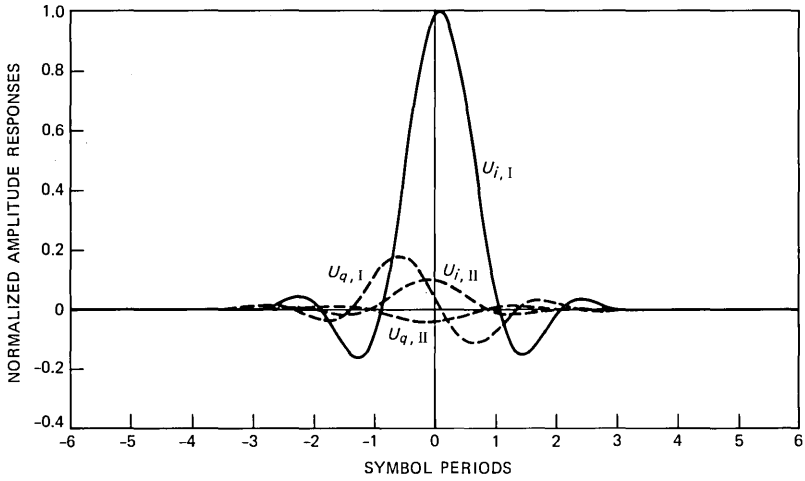


Fig. 9—In-phase impulse responses of dual-polarized system for an offset nonminimum phase fade on the main path and a midband minimum phase fade on the cross-coupled path.

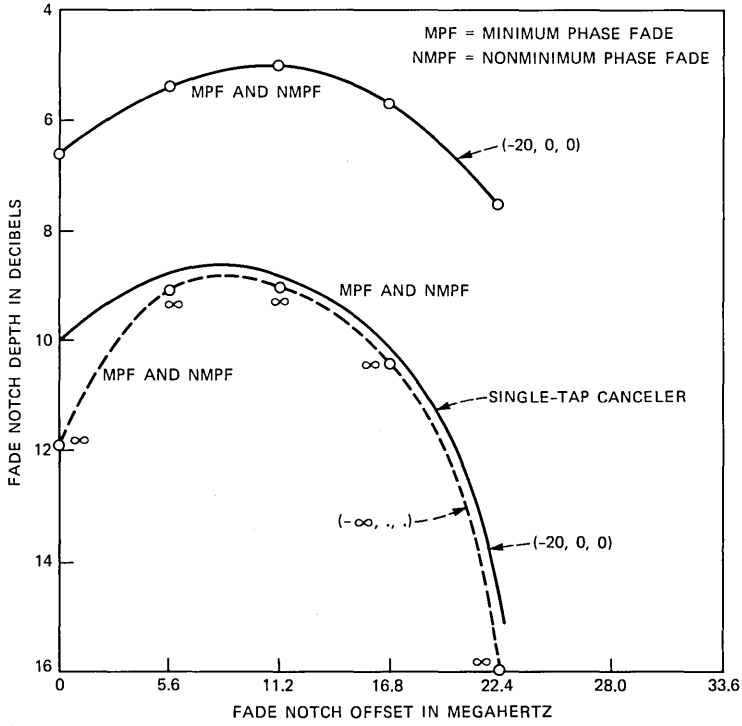


Fig. 10—Canceler performance in dual-polarized 16-QAM radio for a flat fade on the cross-coupled path.

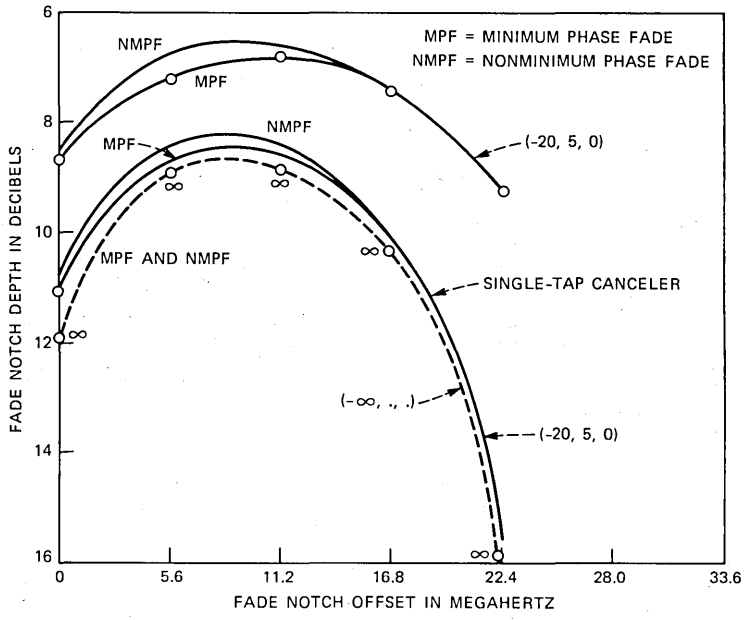


Fig. 11—Canceler performance in dual-polarized 16-QAM radio for a midband dispersive fade on the cross-coupled path.

of the interfering impulse response shape; hence, the canceler does not perform as well for midband fades as it would for offset fades of the main path because as the main-path fade notch moves toward the band edge, the timing reference of the overall impulse response moves toward the origin; hence, the two peaks tend to align. Of course, the remedy for this situation is to increase the number of canceler transversal filter taps. In Fig. 11, we consider the case where the *cross-coupled path is dispersively faded by a minimum phase midband fade of 5-dB notch depth.* In this case, since by asymmetry two different sets of samples of the interfering impulse responses are involved, the resulting signatures under minimum or nonminimum phase fades are not the same, and the nonminimum phase fade increases the outage slightly for the same cross-coupled path fading conditions. As the fade notch moves away from the band center, and since the reference timing of the main-path impulse response moves toward the time origin for both minimum and nonminimum phase fades, the two sets of samples taken of the interfering impulse responses become similar; hence, the minimum and nonminimum phase fade signatures approach one another and become approximately the same around the band edge.

A comparison of the signatures for minimum and nonminimum phase fadings of the main path indicates that the baseband canceler

performance is relatively insensitive to the type of the fade as anticipated in Ref. 1.

Notice that the difference in the signatures for minimum and nonminimum phase fading of the copolarized channel is because we have assumed a minimum phase fade for the cross-coupled channel in both cases. Given that the copolarized and the cross-coupled channel fading have both minimum or both nonminimum phase, the signatures are indeed identical in both cases.

IV. CONCLUSIONS

In this paper performance of a previously proposed canceler[1] is evaluated under nonminimum phase fades, and it is shown that the canceler performance is practically transparent to the type of the fade. An investigation of different static models for nonminimum phase fades is also performed, and it is shown that some of the known fading models for single signal transmission may not be appropriate for multicarrier systems such as dual-polarized channels.

V. ACKNOWLEDGMENT

The writer is deeply indebted to W. C. Jakes, Jr. for discussions during this work.

REFERENCES

1. M. Kavehrad, "Baseband Cross-Polarization Interference Cancellation for M-Quadrature Amplitude-Modulated Signals Over Multipath Fading Radio Channels," AT&T Tech. J., *64*, No. 8 (November 1985), pp. 1913–26.
2. L. Leclert and P. Vandamme, "Non-Minimum Phase Fadings Effects on Equalization Techniques in Digital Radio Systems," GLOBECOM Conf., November 1983, San Diego.
3. H. Sari, "Baseband Equalizer Performance in the Presence of Selective Fading," GLOBECOM Conf., November 1983, San Diego.
4. G. Niezgoda et al., "Effects of Non-Minimum Phase Fades on the Performance of a 49 QPRS 90 Mb/s Digital Radio Using Decision Feedback Equalizations," GLOBECOM Conf., November 1983, San Diego.
5. M. Kavehrad and C. A. Siller, Jr., "Performance Signatures for Dual-Polarized Transmission of M-QAM Signals Over Fading Multipath Channels," AT&T Tech. J., this issue.
6. W. D. Rummler, "A New Selective Fading Model: Application to Propagation Data," B.S.T.J., *58*, No. 5 (May–June 1979), pp. 1037–71.

AUTHOR

Mohsen Kavehrad, B.S. (Electrical Engineering), 1973, Tehran Polytechnic Institute; M.S. (Electrical Engineering), 1975, Worcester Polytechnic Institute; Ph.D. (Electrical Engineering), 1977, Polytechnic Institute of New York; Fairchild Industries, 1977–1978; GTE, 1978–1981; AT&T Bell Laboratories, 1981—. At AT&T Bell Laboratories Mr. Kavehrad is a member of the Communications Methods Research Department at Crawford Hill Laboratory. His research interests are digital communications and computer networks. He has organized and chaired sessions for IEEE sponsored conferences. Technical Editor, IEEE Communications Magazine; Chairman, IEEE Communications Chapter of New Hampshire, 1984; Member, IEEE, Sigma Xi.

AT&T Technical Journal
Vol. 64, No. 10, December 1985
Printed in U.S.A.

Analysis/Simulation Study of Cross-Polarization Cancellation in Dual-Polarization Digital Radio

By L. J. GREENSTEIN*

(Manuscript received November 12, 1984)

This paper analyzes cross-polarization cancellation in dual-polarization digital radio links transmitting M-ary quadrature amplitude-modulation (M-QAM) signals. We consider the use of three options in the receiver: (1) no cancellation ; (2) ideal (i.e., total) cancellation; and (3) optimal nondispersive cancellation. For every option, we assume the canceler to be followed, in each polarization branch, by an ideal minimum mean-square error equalizer. By postulating a statistical model for the co-polarization and cross-polarization responses of a digital radio channel, and then simulating thousands of sets of these responses, we obtain curves that relate outage probability to the number of modulation levels. We show graphically that the no-canceler case is unthinkable; that total cancellation permits results close to those for single-polarization transmission; and that optimal nondispersive cancellation can have a limited range of application. We also examine the effects of key system parameters and the various modeling assumptions.

I. INTRODUCTION

This paper reports on a theoretical study of cross-polarization cancellation in dual-polarization microwave digital radio receivers. In another recent theoretical study, Amitay and Salz derived and evaluated the optimal linear receiver response to cross-pol coupling and multipath fading in combination.[1] By contrast, this paper analyzes suboptimal structures wherein the cross-pol canceler and multipath

* AT&T Bell Laboratories.

equalizer stages are in cascade. We do this because, in the design of practical adaptive receivers, it may be desirable to deal with cross-pol interference and co-pol dispersion separately. For example, cross-pol cancellation may be simpler to perform at IF, while multipath equalization is best done at baseband. More important, *modularity* in providing the cross-pol cancellation and multipath equalization functions may permit more economy and flexibility in designing and using the radio system.

Our primary aim is to show that, in using cross-pol cancellation followed by multipath equalization, overall outage performance can be made nearly as good as that for an equalized single-pol radio link. Our secondary aim is to quantify the differences in attainable outage performance for three distinct approaches to cancellation, namely, (1) no cancellation; (2) ideal (i.e., total) cancellation; and (3) optimal nondispersive cancellation. Finally, we aim to show how these results are influenced by key parameters of the system and various assumptions regarding the radio propagation.

To obtain these quantitative results, we use a combination of receiver analysis and Monte Carlo simulation of the co-pol and cross-pol responses of the propagation channel. The simulations require specifying a model for the channel responses. We have done so, despite the sparsity of published work in this area, by using data, tentative theories and speculations by the author and others. Model uncertainties are dealt with, in part, by means of sensitivity studies.

Our assumptions regarding the system, canceler/equalizer structure, and channel are given in Section II, while the methods of analysis and simulation are described in Section III. In Section IV, we present a number of simulation results in graphical form, and our major findings are summarized in Section V. The reader may wish to review this summary before plunging into the details of the study.

II. STUDY ASSUMPTIONS

2.1 The system

We consider digital radio links transmitting independent random data streams on two nominally identical cofrequency channels using nominally orthogonal polarizations. For reasons of both practicality and convenience, we assume the polarizations to be vertical (V-pol) and horizontal (H-pol).

The cofrequency channels are in the microwave common carrier bands at 4, 6, and 11 GHz, corresponding to channel bandwidths, respectively, of 20, 30, and 40 MHz. The modulation is M-ary Quadrature Amplitude Modulation (M-QAM), with M = 4, 16, 64, 256, etc. The end-to-end spectral shaping (excluding channel dispersion and

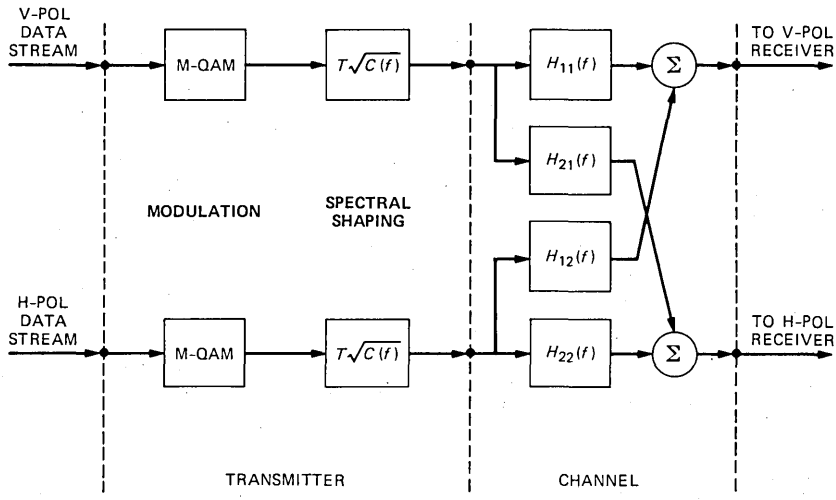


Fig. 1—Schematic representation of a dual-pol M-QAM transmitter with root-cosine-roll-off spectral shaping and a microwave channel with co-pol dispersion and cross-pol coupling. The H-functions are slowly time varying during periods of multipath activity.

adaptive filtering) is cosine roll-off, with roll-off factor α. This shaping is divided evenly between transmitter and receiver. In the receiver, the H-pol and V-pol signals are applied to a cross-pol canceler that linearly combines the two input branches, followed by fixed filtering and adaptive equalization in each output branch.

The above description is reflected in Figs. 1 and 2.* Figure 1 depicts the transmitter and channel, the latter being characterizable by a pair of co-pol responses [$H_{11}(f)$ for the V-pol signal and $H_{22}(f)$ for the H-pol signal] and a pair of cross-coupling responses [$H_{12}(f)$ and $H_{21}(f)$]. Ideally (and in fact under normal conditions), $H_{11}(f)$ and $H_{22}(f)$ are identical nondispersive responses. During multipath fading, however, they become dispersive and possibly smaller as well, as recounted in numerous papers on statistical models.[2-5]

The cross-pol responses, on the other hand, are ideally zero. In fact, however, they are generally nonzero, even under normal conditions. During such times, they are typically nondispersive and small compared to the co-pol responses; but, during multipath fading, they tend to be dispersive and can become relatively large as well,[6] giving rise to potentially serious impairments in detection.

Figure 2 shows the structure of the receiver to be analyzed. All the G-functions are adaptive, and we assume they can be controlled in practice to satisfy the various criteria specified below. For convenience

* Note that f is referred to the center frequency of the radio channel, and that $C(f)$ represents the raised cosine function (dimensionless).

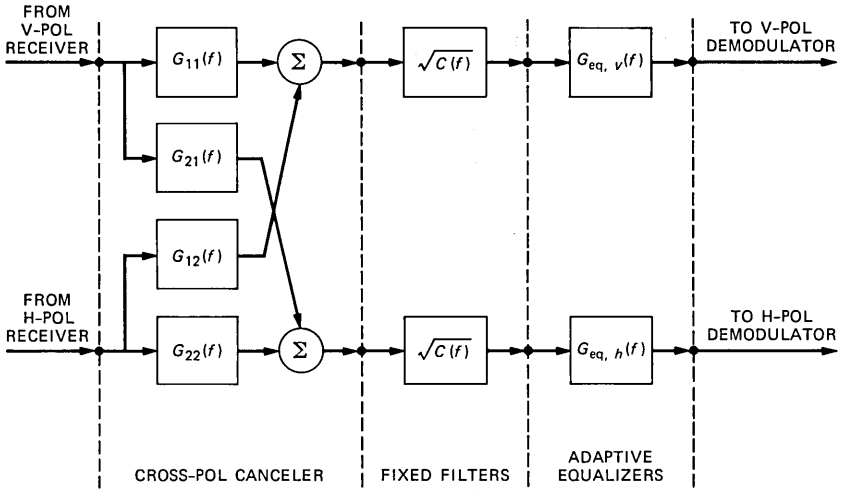


Fig. 2—Schematic representation of dual-pol receiver processing consisting of cross-pol cancellation, root-cosine-roll-off filtering, and adaptive equalization. The G-functions are assumed to be adapted to the prevailing channel response functions in Fig. 1. In principle, the location of the fixed filtering is immaterial, and each of the various linear stages can be at RF,IF or baseband, as appropriate.

only, all the processing stages are shown as passband (rather than baseband) circuits.

Finally, we assume in this study that the RF carriers and the symbol timings are nominally identical but asynchronous. This assumption enables the cross-pol interference to be treated as a noise-like process, which simplifies the analysis in some respects. The results should differ little from those for synchronized dual-pol transmissions.

2.2 Canceler/equalizer structure

To evaluate the canceler/equalizer structure, we need some analytical relationships. Let $S_V(f)$ be the Fourier transform of a long random sequence of (complex) data into the V-pol transmit filtering (Fig. 1), and let $S_H(f)$ be the same for the H-pol branch. Because of cross-coupling in both the propagation medium and the receiver, each branch will show a mixture of V-pol and H-pol data streams at its equalizer output (Fig. 2). Because of symmetry, however, we need only consider the output of the V-pol branch, and will follow that custom throughout.

The Fourier transform of the equivalent baseband signal at the V-pol output is

$$S'_V = S_V(f)C(f)\{[H_{11}(f)G_{11}(f) + H_{21}(f)G_{12}(f)]G_{eq,v}(f)\} + S_H(f)C(f)\{[H_{12}(f)G_{11}(f) + H_{22}(f)G_{12}(f)]G_{eq,v}(f)\}. \quad (1)$$

The second term clearly represents the *cross-pol interference* (hereafter

abbreviated CPI); the bracketed quantity of the first term represents the changes in the desired signal due to the propagation medium, canceler, and equalizer. If $H_{11}(f) = H_{22}(f) = 1$ and $H_{12}(f) = H_{21}(f) = 0$, $S_V'(f)$ reduces to the case of an undistorted single-pol transmission.

We now state the three conditions on $G_{11}(f)$ and $G_{12}(f)$ that will be analyzed here:

1. No cancellation, i.e.,

$$G_{11}(f) = 1 + j0; \qquad G_{12}(f) = 0. \tag{2}$$

2. Ideal cancellation, i.e., from (1),

$$\frac{G_{12}(f)}{G_{11}(f)} = -\frac{H_{12}(f)}{H_{22}(f)}. \tag{3}$$

3. Optimal nondispersive cancellation, defined here by

$$G_{11}(f) = 1 + j0; \qquad G_{12}(f) = g_{\text{opt}}, \tag{4}$$

where g_{opt} is that complex value for which the mean square CPI at the canceler output is minimized. There are other possible criteria for defining g_{opt} (e.g., that value that minimizes the mean-square CPI at the equalizer output), but the simple criterion used here is good enough for our purposes.

Finally, we assume that the equalizer associated with each of the above three conditions is an ideal Minimum Mean-Square Error (MMSE) equalizer. That is, it maximizes the ratio of the signal half-distance to the total root-mean-square (rms) distortion (thermal noise, CPI, and intersymbol interference) as sampled at the baseband detector. Previous studies for single-pol channels have shown that the performance of an ideal MMSE equalizer can be closely approached using fractionally spaced tapped-delay-line filters with just a few taps.[7]

2.3 Channel model

Obtaining an empirical statistical model for the four channel response functions—$H_{11}(f)$, $H_{12}(f)$, $H_{21}(f)$ and $H_{22}(f)$—is a very difficult business. The model to be used here is based only partly on measured data, the other parts being theories on the underlying physics of the propagation medium, and speculations on what mathematical artifacts to include so that important issues are not overlooked. The model draws on the published data, models or ideas of N. Amitay,[1] W. D. Rummler,[2,3] K. T. Wu,[6] S. Lin,[8] M. L. Steinberger[9] and M. Liniger,[10] as well as on private discussions with these investigators and others. The author's own ideas are included in the mixture, and he accepts sole responsibility for flaws in the speculative model presented next.

2.3.1 The co-pol functions, $H_{11}(f)$ and $H_{22}(f)$

The co-pol responses can be represented using the three-path function introduced by Rummler.[2] For most of the year, $H_{11}(f) = H_{22}(f) = 1$; but, during T_0 seconds per hop per year, the two functions become

$$H_{ii}(f) = \begin{cases} a_{ii}[1 - b_{ii}e^{j\phi_{ii}}e^{-j\omega\tau}]; & \text{minimum phase response} \\ a_{ii}[b_{ii} - e^{j\phi_{ii}}e^{-j\omega\tau}]; & \text{nonminimum phase response} \end{cases} \quad (5)$$

$$i = 1, 2,$$

where T_0 is a function of the radio path and band; τ is a fixed parameter of 6.3 ns; $(a_{11}, a_{22}, b_{11}, b_{22}\phi_{11}, \phi_{22})$ are slowly varying "fitting" parameters that collectively provide accurate approximations to the true responses; $b_{11} \leqslant 1$ and $b_{22} \leqslant 1$ at all times; and these representations apply over bandwidths of 40 MHz or less.

Rummler[3] has published a widely used joint probability density function (pdf) for the a-, b-, and ϕ-parameters of a co-pol function such as (5). We will use that pdf to characterize the statistics of both $H_{11}(f)$ and $H_{22}(f)$. Moreover, we assume each function to be minimum phase over a fraction p of all fade events, with p a parameter lying between 0 and 1.

In evaluating the dual-pol receiver, it will be necessary to assume something about the similarity *in time* between $H_{11}(f)$ and $H_{22}(f)$. According to a limited body of data, these functions tend to track quite closely, i.e., $H_{12}(f) \approx H_{11}(f)$ during most fade events. On the other hand, significant impairments could accrue during those events where they are dissimilar. To bracket the range of possibilities, we have obtained simulation results under two distinct assumptions: $H_{22}(f) = H_{11}(f)$ in every fade event; and $H_{11}(f)$ and $H_{22}(f)$ are *statistically* identical but mutually independent over the ensemble of fade events.

2.3.2 Cross-pol functions, $H_{12}(f)$ and $H_{21}(f)$

The model to be used here assumes that

$$H_{ij}(f) = \{0.5k_{ij}[H_{11}(f) + H_{22}(f)] + \epsilon_{ij}e^{j\psi_{ij}}\}e^{-j\omega\delta T}, \quad i \neq j \quad (6)$$

where k_{ij} is a fixed complex constant related to the residual cross-pol coupling in the radio antennas and waveguide runs (typically, 20 $\log|k_{ij}| \leqslant -25$ dB); ψ_{ij} is a uniformly random phase over the ensemble of fade events; ϵ_{ij} is a Rayleigh variate over that ensemble (typically, $10 \log|\epsilon_{ij}|^2 \leqslant -35$ dB); and δT is a fixed time "misalignment" between the co-pol and cross-pol paths to the receiver. We further assume that

ψ_{12}, ψ_{21}, ϵ_{12}, and ϵ_{21} are jointly independent, and that $k_{12} = k_{21}$, both real.*

This model for the cross-pol responses is more or less consistent with data and theories reported previously.[6,8–10] The delay parameter δT is new and is included to add richness to the model; we will see if δT has any important effects on performance as it ranges from 0 to 4 ns. This way of including delay effects may, in fact, be too mild. Amitay[1] has suggested that δT might be larger than 4 ns, and that the factor $e^{-j\omega\delta T}$ might more properly be attached to just $\epsilon_{ij}e^{j\phi_{ij}}$. Such an approach would make the cross-pol responses more dispersive, consistent with some reported measurements. For now, however, we will use the model for cross-pol responses described above.

III. METHODS OF ANALYSIS AND SIMULATION

3.1 Preliminaries

To begin, we note that an M-QAM signal contains two $\sqrt{M}$-level AM signals in phase quadrature, and that the possible data values in each quadrature rail are $\pm 1, \pm 3, \cdots \pm (\sqrt{M} - 1)$. We also note that, after cross-pol cancellation, fixed filtering, adaptive equalization, and coherent demodulation (in whatever order), two baseband pulse streams are sampled every T seconds at each of two data detectors in the V-pol receiver branch, and similarly for the H-pol receiver branch. Our analytical goal is to derive a signal-to-distortion ratio that characterizes performance at either baseband detector of either receiver branch. To the maximum extent possible, we invoke known relationships and any simplifying assumptions that do not compromise the generality of the results.

One simplifying assumption, noted and explained in Section 2.1, is that the cross-pol interference can be treated as a noise-like process. Another simplification accrues by assuming a zero-percent roll-off factor ($\alpha = 0$) for the end-to-end spectral shaping, in which case $C(f)$ is a unit rectangle on the f-interval $[-1/2T, 1/2T]$. This simplifying assumption is justified by earlier findings that, over the practical range $\alpha \leq 0.5$, the roll-off factor has a very mild effect on performance results.[11]

Finally, for convenience we define a carrier-to-noise ratio as follows: let P_0 be the average received power in an M-QAM signal in the absence of fading; and let N_0 be the power spectrum density of the

* The effects of the phases of k_{12} and k_{21}, and of any dissimilarities between these two constants, should be small. We thus remove them from the study to simplify matters.

noise input to the V-pol (or H-pol) receiver, including the contribution of the receiver noise figure. Then

$$\text{CNR} \triangleq P_0 T/N_0. \tag{7}$$

This is the unfaded Carrier-to-Noise Ratio (CNR) in the Nyquist bandwidth, and is typically on the order of 10^6 (60 dB).

3.2 Signal and distortions at the canceler output

From previous discussions and Figs. 1 and 2, we can show that the data pulse at the canceler output for a data value of unity would have a Fourier transform

$$S(f) = \sqrt{\frac{3}{M-1}}\, P_0\, (T\sqrt{C(f)})A(f), \tag{8}$$

where [see (1)]

$$A(f) = [H_{11}(f)G_{11}(f) + H_{21}(f)G_{12}(f)]. \tag{9}$$

Similarly, the thermal noise at that output has a power spectrum density

$$N(f) = N_0 B_n(f), \tag{10}$$

where (see Fig. 2)

$$B_n(f) = [|\, G_{11}(f)\,|^2 + |\, G_{12}(f)\,|^2]. \tag{11}$$

The cross-pol interference has a power spectrum density

$$X(f) = P_0 T C(f) B_c(f), \tag{12}$$

where [see (1)]

$$B_c(f) = |\, H_{12}(f)G_{11}(f) + H_{22}(f)G_{12}(f)\,|^2. \tag{13}$$

The composite noise power spectrum density* can then be written as

$$\begin{aligned} N'(f) &= N(f) + X(f) \\ &= N_0[B_n(f) + \text{CNR}\ C(f)B_c(f)]. \end{aligned} \tag{14}$$

Henceforth, we will make use of our assumption that $\alpha = 0$, i.e., that $C(f)$ is a unit rectangle on $[-1/2T, 1/2T]$.

3.3 The MMSE equalizer

The MMSE equalizer is the linear filter that maximizes, at the

* This spectrum density is for the sum of CPI and thermal noise, which can be treated as noise-like but is clearly not Gaussian except when the CPI vanishes.

baseband detector, the ratio of sampled-signal half-distance to rms distortion, the latter consisting of thermal noise, CPI, and Intersymbol Interference (ISI). We need not concern ourselves with how such an equalizer is realized, as this is a well-developed art; we merely need to derive and analyze its frequency response.

The derivation is accomplished in two steps. First, we assume a noise-whitening input filter, i.e., one having a real transfer function proportional to $1/\sqrt{N'(f)}$, and then we invoke the result in Section 5.1 of Ref. 12 for MMSE equalizers with white input noise. The equalizer response for $\alpha = 0$ turns out to be

$$G_{eq}(f) = \frac{A^*(f)}{B_n(f) + \mathrm{CNR}[B_c(f) + |A(f)|^2]}; \qquad |f| \leq \frac{1}{2}T$$
$$= 0; \qquad \text{elsewhere,} \tag{15}$$

where, for convenience, we have suppressed the subscript v in $G_{eq,v}(f)$, Fig. 2.

The data pulse following an equalizer with this response has a real, nonnegative Fourier transform. Assuming optimal carrier and timing recovery, we can show that the squared half distance between adjacent signal samples (constellation points) at the detector is therefore

$$P_s = [\int S(f)G_{eq}(f)df]^2, \tag{16}$$

where the integration limits, both here and in subsequent expressions, are $\pm 1/2T$.

The mean-square intersymbol interference at the sample times can likewise be shown to be

$$P_i = \frac{M-1}{3}\left[\frac{1}{T}\int |S(f)G_{eq}(f)|^2 df - P_s\right] \tag{17}$$

and the mean-square composite noise power is

$$P_{n'} = \int N'(f)|G_{eq}(f)|^2 df. \tag{18}$$

3.4 The signal-to-distortion parameter, Γ

The signal-to-distortion ratio (ρ) at either baseband detector is now defined to be

$$\rho \triangleq P_s/(P_i + P_{n'}). \tag{19}$$

The significance of ρ is that the Bit Error Rate (BER) can be tightly upperbounded by[1,13]

$$\mathrm{BER} \leq 2\exp(-\rho/2). \tag{20}$$

By using the various relationships in Sections 3.1 through 3.3, we can express ρ in the form

$$\rho = \frac{3}{M-1}\left(\frac{X_s}{X_i + X_{n'}/\text{CNR}}\right) \equiv \frac{3}{M-1}\,\Gamma, \tag{21}$$

where

$$X_s \triangleq \left\{\overline{\left[\frac{Y(f)}{1+\text{CNR}\ Y(f)}\right]}\right\}^2 \tag{22}$$

$$X_i \triangleq \overline{\left\{\frac{Y(f)}{1+\text{CNR}\ Y(f)}\right\}^2} - X_s \tag{23}$$

$$X_{n'} \triangleq \overline{\left\{\frac{Y(f)}{[1+\text{CNR}\ Y(f)]^2}\right\}} \tag{24}$$

$$Y(f) \triangleq |A(f)|^2/[B_n(f) + \text{CNR}\ B_c(f)], \tag{25}$$

and the overbar denotes a frequency average over $[-1/2T, 1/2T]$, i.e.,

$$\overline{Z(f)} \triangleq \int_{-1/2T}^{1/2T} Z(f)df T. \tag{26}$$

The focus of our investigation is the signal-to-distortion parameter Γ introduced by (21). To "calibrate" this quantity, let Γ_0 denote the value of Γ that must be exceeded if BER is to lie below some threshold value BER_0. We can find a tight upperbound for Γ_0 given BER_0 and M by invoking the equality in (20) and combining it with (21). The result is

$$\Gamma_0 = \frac{2}{3}(M-1)\ell n(2/\text{BER}_0). \tag{27}$$

Some values of Γ_0, expressed in decibles, are given in Table I for various BER_0 and M of interest.

3.5 Two special cases

Although the above results are applicable to any assumptions we make about the canceler, two special cases are worthy of note. The

Table I—Values of Γ_0 for various M and specified bit error rates

	M		
BER_0	16	64	256
10^{-3}	18.81 dB	25.04 dB	31.11 dB
10^{-4}	19.96 dB	26.19 dB	32.26 dB
10^{-5}	20.87 dB	27.10 dB	33.17 dB
10^{-6}	21.62 dB	27.85 dB	33.92 dB

first is the case of total cancellation [see (3)]. Under this condition, $X(f)$ in (12) vanishes entirely and so $N'(f)$ reduces to $N(f)$. Also, as the mathematics of this situation makes clear, the choices of $G_{11}(f)$ and $G_{12}(f)$ individually are immaterial, just so their ratio satisfies (3). This is a consequence of the (presumed) fact that the dominant thermal noise is introduced before the canceler.

The second special case is that of nondispersive cancellation [see (4)]. We assume that g_{opt} is chosen so as to minimize $\overline{X(f)}$ [see (12) and (13)]. For a zero-percent roll-off factor, it is easy to show that

$$g_{\text{opt}} = -\frac{\overline{H_{22}^*(f)H_{12}(f)}}{\overline{|H_{22}(f)|^2}}. \tag{28}$$

We have used this relationship in our computations.

3.6 *Analysis/simulation program*

For convenience in what follows, we now redefine two parameters of the channel model, namely,

$$K \triangleq 20 \log k, \tag{29}$$

where $k = k_{12} = k_{21}$ (real) in (6) and

$$E \triangleq 10 \log\overline{|\epsilon_{12}|^2} = 10 \log\overline{|\epsilon_{21}|^2}. \tag{30}$$

We can say that the channel is statistically specified once we (1) assign numerical values to p, δT, K and E; and (2) declare $H_{11}(f)$ and $H_{22}(f)$ to be either identical or statistically independent. Similarly, we can say that the radio system is design-specified once we (1) assign numerical values to α, CNR, and $1/T$; and (2) declare the type of cross-pol cancellation to be used.

A computer program has been written that obtains, for any joint specification of the channel and system, the yearly probability distribution of Γ. To accomplish this, the program combines Monte Carlo simulation methods with the channel model of Section 2.3 to generate a large statistical "ensemble" of channel response functions. That is, each member of the ensemble is a set of functions, $\{H_{11}(f), H_{12}(f), H_{21}(f), H_{22}(f)\}$, generated by deriving the various parameter values (a_{11}, b_{11}, a_{22}, b_{22}, ϵ_{12}, ϵ_{21}, etc.) in accordance with the statistics of the model. For each set of H-functions thus generated, the program computes Γ using the formulas of Section 3.4. After generating and evaluating the prescribed number of sets (20,000 in our study), the program computes a cumulative probability function, $P(\Gamma)$, for the population of Γ-values thus obtained.

We present results for the following channel/system parameter values or conditions, where those in boldface are the ones used the most extensively:

Minimum phase probability, p: 0, **0.5** and 1.0.
Delay parameter, δT: 0, **2 ns** and 4 ns.
Proportional coupling parameter, K: **−25 dB,** −30 dB, and −35 dB.
Additive coupling parameter, E: −35 dB, **−40 dB,** and −45 dB.
Statistical dependence between co-pol functions: **Totally dependent ($\mathbf{H_{11}(f) = H_{22}(f)}$)** and totally independent.
Type of cancellation: No cancellation, **total cancellation,** and **nondispersive cancellation.**
Symbol rate, $1/T$: 15 Mbaud, **22.5 Mbaud,** and 30 Mbaud (typical values for digital radio systems in the 4-, 6-, and 11-GHz radio bands, respectively).

Throughout our simulations and computations, we have assumed a 0-percent roll-off factor ($\alpha = 0$) and a 63-dB unfaded carrier-to-noise ratio (CNR = 2×10^6). However, we will also discuss how the computed results would vary with these parameters.

IV. RESULTS

The results of this study are given by Figs. 3 through 6. Each figure shows curves of $P(\Gamma)$ versus Γ for a number of different channel/system specifications. These curves can be interpreted as conditional outage probabilities and can be used to estimate yearly outage seconds on a radio hop, as follows: Given a threshold bit error rate (BER_0) and the number of modulation levels (M), the minimum acceptable value of Γ can be obtained using (27) or Table I. (Assuming $BER_0 = 10^{-4}$, Γ_0 is roughly 20 dB for $M = 16$ and 26 dB for $M = 64$.) The resulting $P(\Gamma_0)$ is the probability of outage on a given hop *conditioned* on the occurrence of fading. To estimate yearly outage seconds on the hop, one would need to estimate, measure, or obtain from available models the expected number of yearly fading seconds, T_0. (A representative value is 16,000 seconds.) Multiplying T_0 by $P(\Gamma_0)$ would yield the expected number of outage seconds per hop per year. This utilitarian aspect of the curves in Figs. 3 through 6 should be kept in mind as we make some relative comparisons.

4.1 Influence of symbol rate and type of cancellation

Figure 3 shows $P(\Gamma)$ for each of three symbol rates and each of three cancellation options. The "common conditions" listed on the figure are assumed to be the most representative for an actual dual-pol channel.

The top curve shows what can be expected if no cancellation whatsoever is employed. The heavy line used here contains the results for all symbol rates from 15 Mbaud to 30 Mbaud. It is clear that the absence of some sort of cancellation is unthinkable for the dual-pol

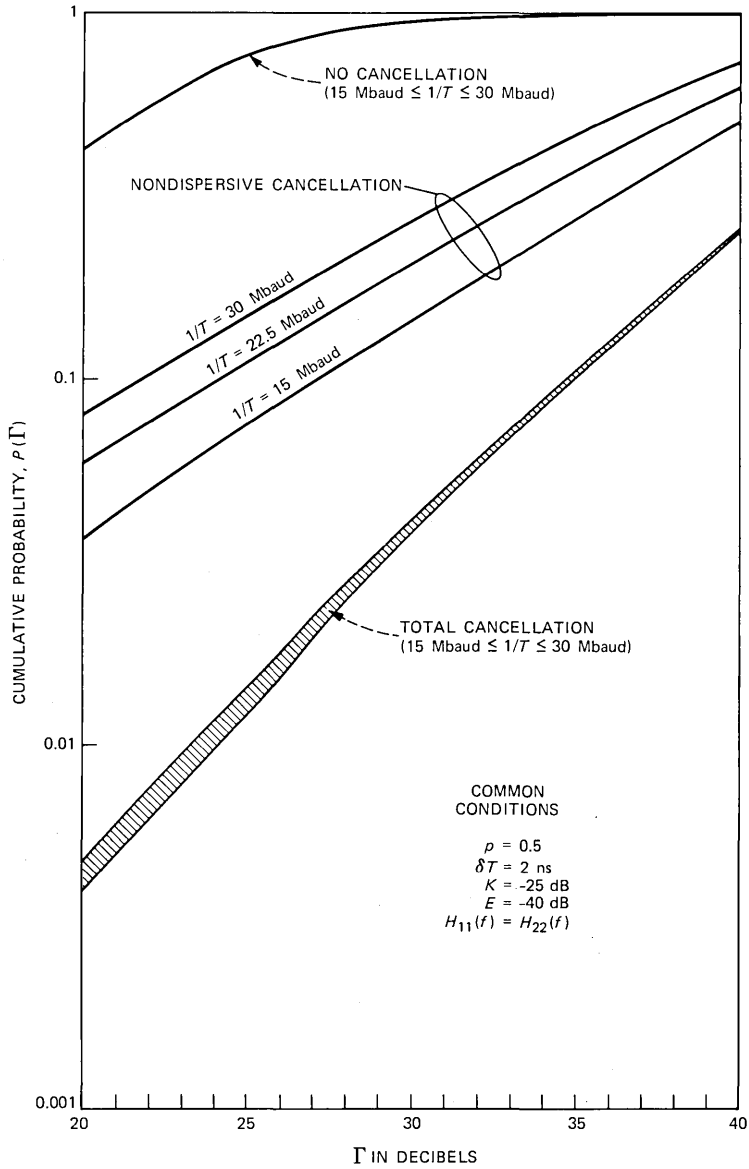


Fig. 3—Cumulative probability functions for Γ. Results are shown for each of the three canceller options considered and for symbol rates of 15, 22.5, and 30 Mbaud. In addition to the "common conditions" shown, $\alpha = 0$ and CNR $= 2 \times 10^6$ (63 dB).

systems of interest, and we dismiss this option from further consideration.

The results for nondispersive cancellation reveal an order-of-magnitude improvement (for the lower values of Γ) and a fairly strong

dependence on symbol rate. The latter is not surprising since larger symbol rates involve larger bandwidths and, hence, greater dispersion in the channel H-functions. In microwave common carrier channels with narrow bandwidths (e.g., 3.6 MHz in the 2-GHz band), nondispersive cancellation might thus be quite adequate. This would depend, of course, on the number of modulation levels and the outage probability requirements.

The narrow band at the bottom, for the case of total cancellation, contains results for all symbol rates between 15 Mbaud and 30 Mbaud. This approach provides, at the lower values of Γ, another order-of-magnitude improvement beyond nondispersive cancellation. Moreover, the dependence on symbol rate (or bandwidth) is seen to be small. Later, we will compare these results with those for single-pol transmission using ideal MMSE equalization.

4.2 *Influence of channel parameters p and δT*

Figure 4 shows results for both nondispersive and total cancellation as p and δT range over the sets of values we have specified for them. The "common conditions" assumed here are typical ones; alternative realistic assumptions would not alter the trends revealed by these curves.

The main conclusion we can draw here is that outage performance for nondispersive cancellation is sensitive to the details of the channel model, while that for total cancellation is not. This is reflected in the wideness and narrowness, respectively, of the bands for these two cases. Since total cancellation is the obvious design choice for a high-quality system, this is good news. It implies that certain hard-to-determine fine details of the channel model need not be accurately specified to obtain reliable performance estimates.

4.3 *Influence of the dependence between co-pol responses*

Figure 5 illustrates, for both nondispersive and total cancellation, how performance is affected by the statistical dependence between $H_{11}(f)$ and $H_{22}(f)$. The performance variation over the range between total dependence $[H_{11}(f) = H_{22}(f)]$ and total independence is seen to be fairly small. Nevertheless, it would be clearly beneficial if the co-pol responses were more independent than they apparently are.

A simple explanation can be given for the improvement shown when $H_{11}(f)$ and $H_{22}(f)$ are independent. Let us consider the case of total cancellation and assume, for simplicity, that all H-functions are essentially flat with frequency. Combining (10) and (11) with (3) and (1) under these circumstances, we can show that the receiver output

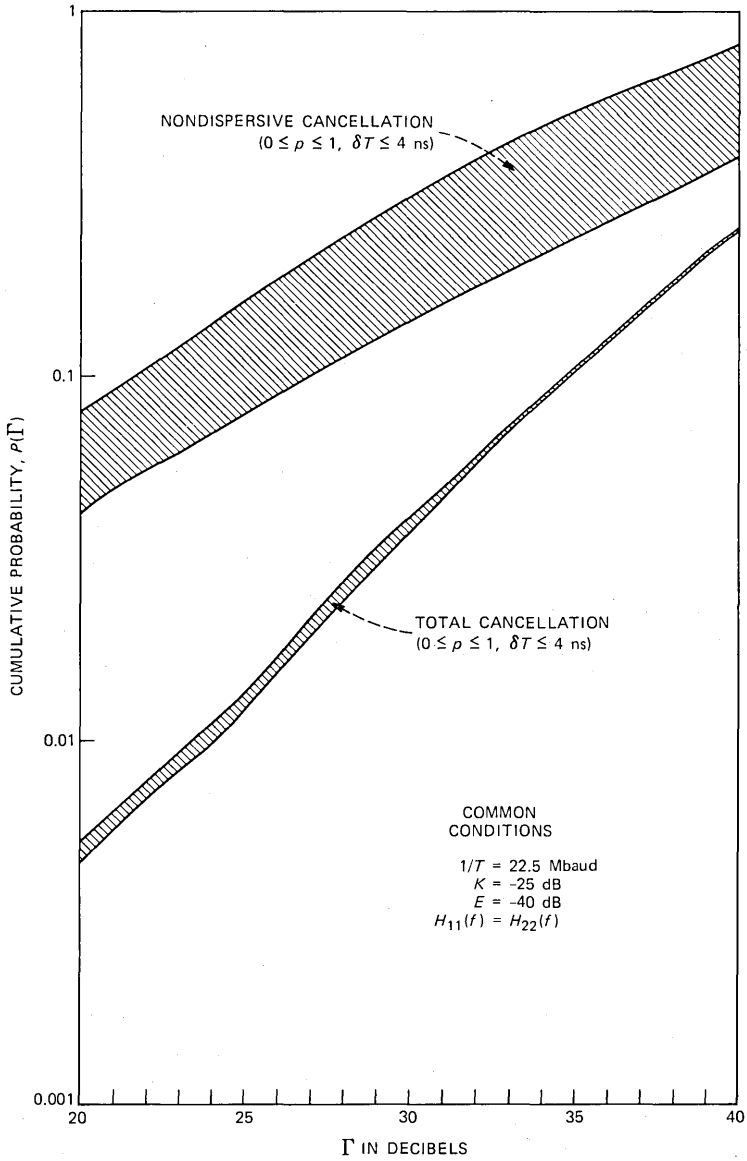


Fig. 4—Cumulative probability functions for Γ. Results are shown for nondispersive and total cancellation, with the symbol rate fixed at 22.5 Mbaud. The parameters here are the minimum phase probability (p) and the delay parameter (δT). All else is the same as in Fig. 3.

signal-to-noise ratio would be proportional to $|H_{11}H_{22} - H_{12}H_{21}|^2/[|H_{22}|^2 + |H_{12}|^2]$. The potential benefit of statistical independence arises when $|H_{11}|$ is weak; at such times, the numerator can be quite small if $|H_{22}|$ is similarly weak, via near cancellation of $H_{11}H_{22}$ by

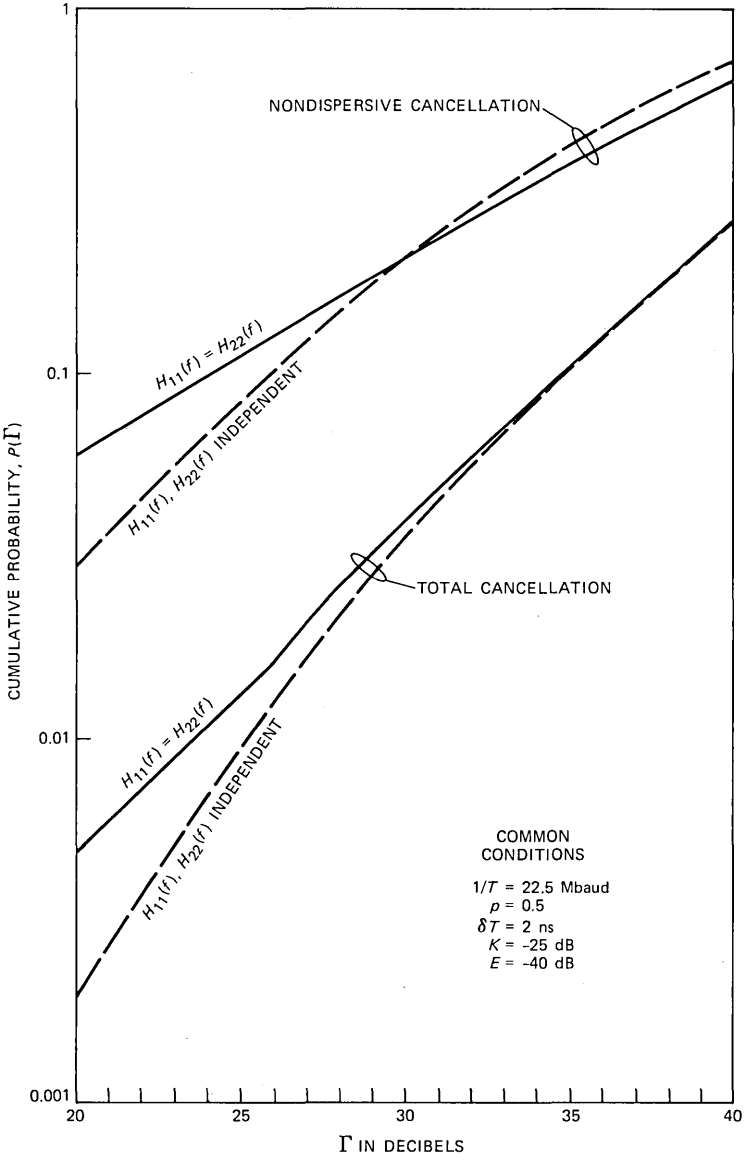


Fig. 5—Cumulative probability functions for Γ. Same conditions as in Fig. 4, except that p and δT are fixed at 0.5 and 2 ns, and results are shown for both totally dependent and totally independent co-pol response functions.

$H_{12}H_{21}$. If H_{11} and H_{22} are independent, however, there is a chance that H_{22} will be strong enough at such times to avoid this near-cancellation. The statistical effect of all this is reflected in the comparative results of Fig. 5.

4.4 *Influence of channel coupling parameters K and E*

Figure 6 shows results for both nondispersive and total cancellation as K and E range over the sets of values we have specified for them. Note that these particular results are for $1/T = 22.5$ Mbaud and $H_{11}(f) = H_{22}(f)$.

For nondispersive cancellation, each heavy line contains results for a particular E and all K below -25 dB. The negligible influence of K over this range is apparent. However, E is another matter. As this quantity decreases in 5-dB steps from -35 dB, the probability curves shift to the right by nearly 5 dB. These features reflect the fact that nondispersive cancellation virtually eliminates the effect of the cross-coupling gain component $kH_{11}(f)$ in (6) [remember that $H_{22}(f) = H_{11}(f)$ in these calculations] but is less effective against the second component, whose mean-square value in decibels is E. This reveals yet another way in which precision in the channel model is needed to estimate nondispersive canceller performance.

For the case of total cancellation, on the other hand, the lower band in Fig. 6 shows how much that need is attenuated. Even so, this band widens measurably as E decreases below -45 dB. The point is made clear by the dotted curve, for $K = E = -\infty$, which is equivalent to the case of single-pol transmission (i.e., no input CPI) and ideal MMSE equalization.

We now see how closely dual-pol outage performance comes to that for single-pol operation when total cross-pol cancellation is used. We can also make a limited comparison between the cascaded approach (i.e., canceller and equalizer in tandem) and the optimum linear receiver (i.e., jointly optimizing $\{G_{11}(f), G_{12}(f), G_{21}(f), G_{22}(f)\}$ against both CPI and multipath dispersion, thereby eliminating the separate equalizer). The latter case is treated comprehensively by Amitay and Salz in Ref. 1. In that paper, the probability functions are plotted against *spectral efficiency*, in b/s/Hz, and so the following correspondences apply: the abscissa values of 4 and 6 b/s/Hz in Ref. 1 correspond closely to $\Gamma = 20$ and 26 dB, respectively, in the present paper. Now assuming that $E = -35$ dB and $1/T = 30$ Mbaud, Fig. 3 of Ref. 1 shows the outage probability for the optimal linear receiver to be roughly four times higher than for single-pol transmission when the spectral efficiency is 4 b/s/Hz, and roughly two times higher when the spectral efficiency is 6 b/s/Hz. The corresponding results in the present study for $\Gamma = 20$ and 26 dB are quite similar and, for higher abscissa values, the similarities are even greater. While recognizing that the two studies used somewhat different models and methods of analysis, and entirely different random numbers in their Monte Carlo

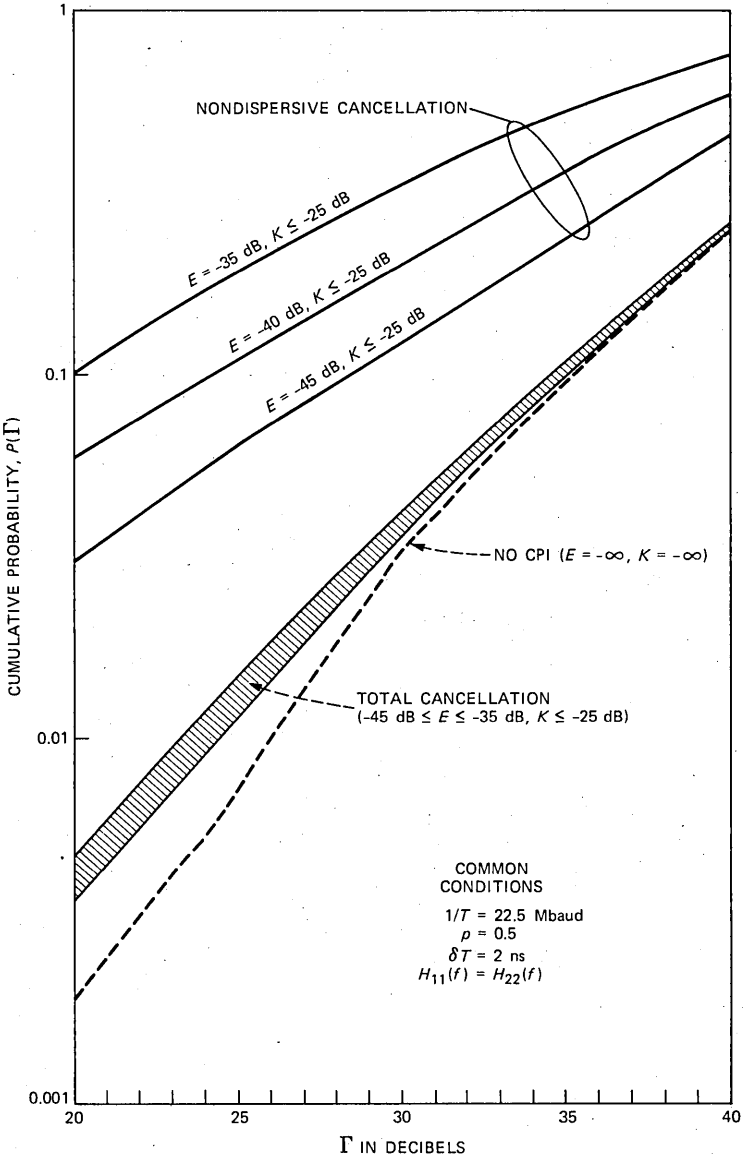


Fig. 6—Cumulative probability functions for Γ. Same conditions as in Fig. 4, except that p and δT are fixed at 0.5 and 2 ns, and K and E are parameters. Note the dotted curve "No CPI," which corresponds to an ideally equalized single-pol channel.

simulations, we perceive a general truth in this comparison: specifically, the cascaded approach, in which cancellation and equalization are functionally separate, leads to outage statistics very close to those for optimal linear reception.

4.5 *Influence of system design parameters α and CNR*

All of the results in Figs. 3 through 6 are for $\alpha = 0$ and CNR $= 2 \times 10^6$ (63 dB). Nevertheless, we can say something about the influences of these parameters. From previous studies,[11] for example, we know that α has a very small effect on outage probability over the practical range $0 < \alpha \leq 0.5$. Also, the curves for total cancellation would essentially shift X dB to the right (or left) for every X-dB increase (or decrease) in CNR. This is because the residual distortion in the case of total cancellation followed by MMSE equalization is almost entirely thermal noise. In the case of nondispersive cancellation, wherein the dominant residual distortion is uncancelled CPI, this sensitivity of the results to CNR would be sharply reduced.

V. SUMMARY AND CONCLUSION

We have used analysis and Monte Carlo simulation to estimate conditional outage probabilities in dual-pol digital radio as functions of a detection measure (Γ) that can be related to the number of modulation levels and the bit error rate. In performing the simulations, we have resorted to a statistical model for the dual-pol channel that lacks a firm empirical basis. This results unavoidably from the current incomplete status of dual-pol channel measurement and modeling. We have dealt with this limitation, in part, by treating various uncertain aspects of the model parametrically.

The main findings of this study can be summarized as follows:

- Obtaining reliable estimates of outage probability for the case of nondispersive cancellation requires accurate, detailed descriptions of the underlying channel model. The outage performance of this cancellation approach is also quite sensitive to bandwidth (or symbol rate).
- In the case of total cancellation, by contrast, outage performance is insensitive to many details of the channel model as well as to symbol rate.
- While far superior to no cancellation, nondispersive cancellation leads to outage probabilities an order-of-magnitude greater than does total cancellation. Nonetheless, it may find applications where symbol rates are low (less than 5 Mbaud) and the outage requirements are liberal.
- The outage statistics for total cancellation followed by ideal equalization are fairly close to those for single-pol transmission, wherein there is no cross-pol interference to cancel.
- More significantly, total cancellation in cascade with ideal equalization appears to produce outage statistics very close to those for optimal linear reception, wherein the effects of cross-pol interfer-

ence and multipath dispersion are jointly minimized in the same receiver stage. Use of the cascade approach, therefore, may permit such benefits as design simplicity, manufacturing economy, and operational flexibility with no serious loss in performance.

VI. ACKNOWLEDGMENT

I am grateful to Lisa J. (Domenico) Case for her help in executing the computer simulation/analysis programs.

REFERENCES

1. N. Amitay and J. Salz, "Linear Equalization Theory in Digital Data Transmission Over Dually Polarized Fading Radio Channels," AT&T Bell Lab. Tech. J., *63,* No. 10, Part 1 (December 1984), pp. 2215–59.
2. W. D. Rummler, "A New Selective Fading Model: Application to Propagation Data," B.S.T.J., *58,* No. 7 (May–June 1979), pp. 1037–71.
3. W. D. Rummler, "More on the Multipath Fading Channel Model," IEEE Trans. Commun., *COM-29,* No. 3 (March 1981), pp. 346–52.
4. L. J. Greenstein and B. A. Czekaj, "A Polynomial Model for Multipath Fading Channel Responses," B.S.T.J., *59,* No. 7 (September 1980), pp. 1197–225.
5. J. C. Campbell and R. P. Coutts, "Outage Prediction of Digital Radio Systems," Electron. Lett., *18,* No. 25/26 (December 1982), pp. 1071–2.
6. K. T. Wu, "Measured Statistics of Multipath Dispersion of Cross Polarization Interference," Paper 46.3, Int. Conf. Commun., May 14–17, 1984, Amsterdam.
7. N. Amitay and L. J. Greenstein, "Multipath Outage Performance of Digital Radio Receivers Using Finite-Tap Adaptive Equalizers," IEEE Trans. Commun., *COM-32,* No. 5 (May 1984), pp. 597–608.
8. S. H. Lin, "Impact of Microwave Depolarization During Multipath Fading on Digital Radio Performance," B.S.T.J., *56,* No. 5 (May 1977), pp. 645–74.
9. M. L. Steinberger, "Design of a Terrestrial Cross Pol Canceller," Paper 2B.6, Int. Conf. Commun., June 13–17, 1982, Philadelphia, Pa.
10. M. Liniger, "Sweep Measurements of Multipath Effects on Cross-Polarized RF-Channels Including Space Diversity," Paper 45.7, GLOBECOM '84, November 26–29, 1984, Atlanta, Ga.
11. W. C. Wong and L. J. Greenstein, "Multipath Fading Models and Adoptive Equalizers in Microwave Digital Radio," IEEE Trans. Commun., *COM-32,* No. 8 (August 1984), pp. 928–34.
12. R. W. Lucky, J. Salz, and E. J. Weldon, Jr., *Principles of Data Communication,* New York: McGraw-Hill, 1968.
13. G. J. Foschini and J. Salz, "Digital Communications Over Fading Radio Channels," B.S.T.J., *62,* No. 2, Part 1 (February 1983), pp. 429–56.

AUTHOR

Larry J. Greenstein, B.S.E.E., 1958, M.S.E.E., 1961, and Ph.D. (Electrical Engineering), 1967, Illinois Institute of Technology; AT&T Bell Laboratories, 1970—. Mr. Greenstein currently heads the Radio Systems Research Department at Crawford Hill in Holmdel, N.J. His most recent work has dealt with communications satellites, mobile telephony, microwave digital radio and optical communications. His previous work was on digital encoding, digital filtering, and, at IIT Research Institute before 1970, airborne radar. Member, Eta Kappa Nu, Tau Beta Pi, and Sigma Xi; Senior Member, IEEE; Senior Technical Editor, IEEE Communications Magazine; co-recipient, IEEE Communications Society's 1984 Prize Paper Award in Communications Systems.

AT&T Technical Journal
Vol. 64, No. 10, December 1985
Printed in U.S.A.

A Laboratory Simulation Facility for Multipath Fading Microwave Radio Channels

By A. J. RUSTAKO, JR., C. B. WOODWORTH, R. S. ROMAN, and H. H. HOFFMAN*

(Manuscript received April 29, 1985)

This paper describes a laboratory facility capable of simulating time-varying radio multipath channel responses in real time under computer control. Four independent fading channels are available that can be used for single-polarization nondiversity, combined in pairs for single-polarization dual diversity, or cross-coupled to simulate the two outputs of a dual-polarization nondiversity channel. Each channel contains a controllable variable network capable of producing a narrowband intermediate frequency response that resembles the "three-path" function of Rummler. A wide range of models can be accommodated by altering the computer-stored sequences used to control each variable channel network. The only assumption implicit in the choice of model is that the channel response can be fitted to the generic function over bandwidths up to 40 MHz. The channel responses are controlled by either entering fixed parameters from a keyboard, or by reading time-varying parameters stored in disk memory. This description includes the architecture, hardware design, software implementation, and performance of the simulation facility.

I. INTRODUCTION

1.1 Motivation

The primary impediment to the operation of digital radio on microwave line-of-sight paths is multipath fading. In dual-polarization

* Authors are employees of AT&T Bell Laboratories.

(dual-pol) systems, this problem is augmented by cross-polarization (cross-pol) coupling. Numerous radio measurements and data analyses over the past several years have served to characterize multipath and cross-pol responses, and to translate these characterizations into statistical models.[1-11] Many of these models have been eagerly engaged by radio systems analysts to estimate, using analysis and/or Monte Carlo simulation, the expected link outage times for different modulations, link parameters and receiver techniques.[12-22] Over the same period, a number of outage measurements have been reported for specific hardware designs, based on either field trials conducted over radio paths or laboratory "signature" measurements coupled with assumed radio channel models.[23-31]

If one stands back from this myriad of activities, important limitations in all of the above approaches become apparent. The analysis/simulation of statistical models applied to specified designs permits rapid comparisons among contending radio schemes, but relies on idealized models of the hardware behavior. Moreover, such study methods do not readily take account of the time dynamics of the channel responses. The coupling of statistical models with lab-measured hardware "signatures" likewise omits channel time dynamics and their effects on system techniques. This deficiency is absent only in the case of field measurements, but these are both costly and time-consuming. More important, system qualities inferred from this approach are subject to the particular responses that nature provides during the test interval. Meaningful comparisons between systems, therefore, require measurements conducted in parallel under identical conditions of path and time.

The Channel Simulation Facility (CSF) reported here is intended to fill the gaps between these various study approaches. It simulates time-varying radio channel responses in the laboratory in real time, and it plays the channel responses into actual hardware realizations rather than idealized system models. The channel responses are dictated by computer-generated control signals, and so the twin benefits of *model selectability* (software-controllable changes in the history of the channel response functions) and *repeatability* (ability to replicate the channel response history for different systems at different times) are realized. And, finally, the ability to simulate channels in a laboratory can sharply contract the time needed to cover a "fading year" and permit considerable reductions in cost.

1.2 Features of the CSF

The heart of the CSF is a group of four identical, variable channel networks, each of which produces, under computer control, a narrow-

band Intermediate Frequency (IF) response that resembles the "three-path" function of Rummler.[1] That function is commonly expressed in the literature as

$$H(f) = a[1 - be^{j\phi}e^{-j\omega\tau}]; \qquad \tau = 6.3 \text{ ns}, \tag{1}$$

where a, b and ϕ are slowly varying random "fade parameters," and $\omega(= 2\pi f)$ is measured from the selected intermediate frequency. For later convenience, the following parameters are defined:

$$A = -20 \log_{10} a \tag{2}$$

$$B = -20 \log_{10}(1 - b). \tag{3}$$

Thus each variable network has a response about the IF (either 70 MHz or 1.070 GHz, as selected) given by

$$H(f) = a + ce^{j\theta}e^{-j\omega\tau}, \tag{4}$$

where a and c are computer-controlled attenuations and θ is a computer-controlled phase shift. In terms of (1), the hardware variables c and θ are ab and $\phi + \pi$, respectively. For later convenience, we define

$$C = -20 \log_{10} c. \tag{5}$$

The four variable channel networks are physically paired, i.e., two such networks with a common input but separate outputs are contained in each of two identical Dual Channel Units (DCUs). The four networks can be used in three distinct ways: (1) A single network (any of the four) can be used to produce the output of a single-pol nondiversity channel, e.g., see Fig. 1a; (2) two networks within the same DCU can be used to produce the two outputs of a single-pol dual diversity channel, e.g., see Fig. 1b; or (3) all four networks can be combined, using a Cross-Coupling Unit (CCU), to produce the two outputs of a dual-pol nondiversity channel, e.g., see Fig. 2.

We thus see that a total of three units comprise the CSF, permitting either of three modes of use at either of two IFs. These and other features of the CSF are summarized in Table I.

Two important facts about the CSF are important to emphasize. One is that the responses of the four networks are completely and separately controllable, and can be driven either via specified hardware parameters typed into a computer terminal (dial-up responses, which are fixed until the input parameters are changed); or via software control, whereby the electronically adjustable hardware parameters are time varied by reading stored sequences from a disk and applying Digital-To-Analog (D/A) conversion and low-pass filtering. In the latter case, the sequences are produced by software routines tailored to a specified statistical channel model. A wide range of models can

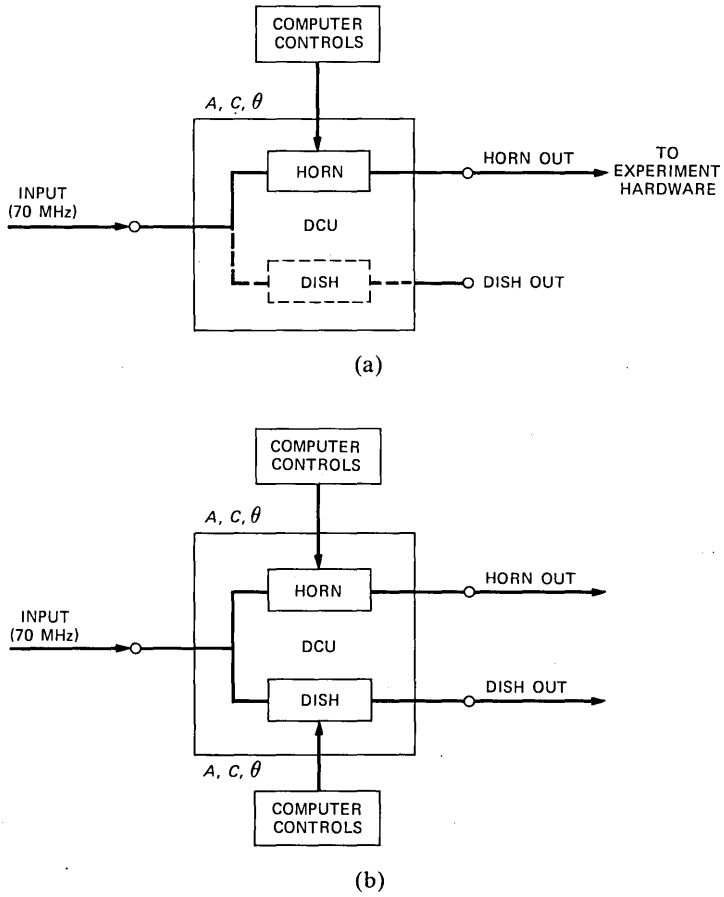


Fig. 1—Simulation modes of (a) a single-pol channel nondiversity and (b) a dual space diversity.

thus be accommodated by suitably altering the software routines. This feature is the key to the flexibility and simplicity of use of the CSF.

The second important fact is that designing the variable networks about the "three-path" function is *not* restrictive with respect to permissible channel models. The *only* assumption implicit in this approach is that all channel response functions of interest can be well fitted by this generic function. For the vast majority of radio paths that have been measured, the evidence supports this assumption for channel bandwidths up to 40 MHz. This means that if a given user of the CSF wishes to consider a model with a different fitting function (e.g., the first-degree polynomial[3] or the "two-path" function[10]), software can be written to interface this model with the simulator hardware.

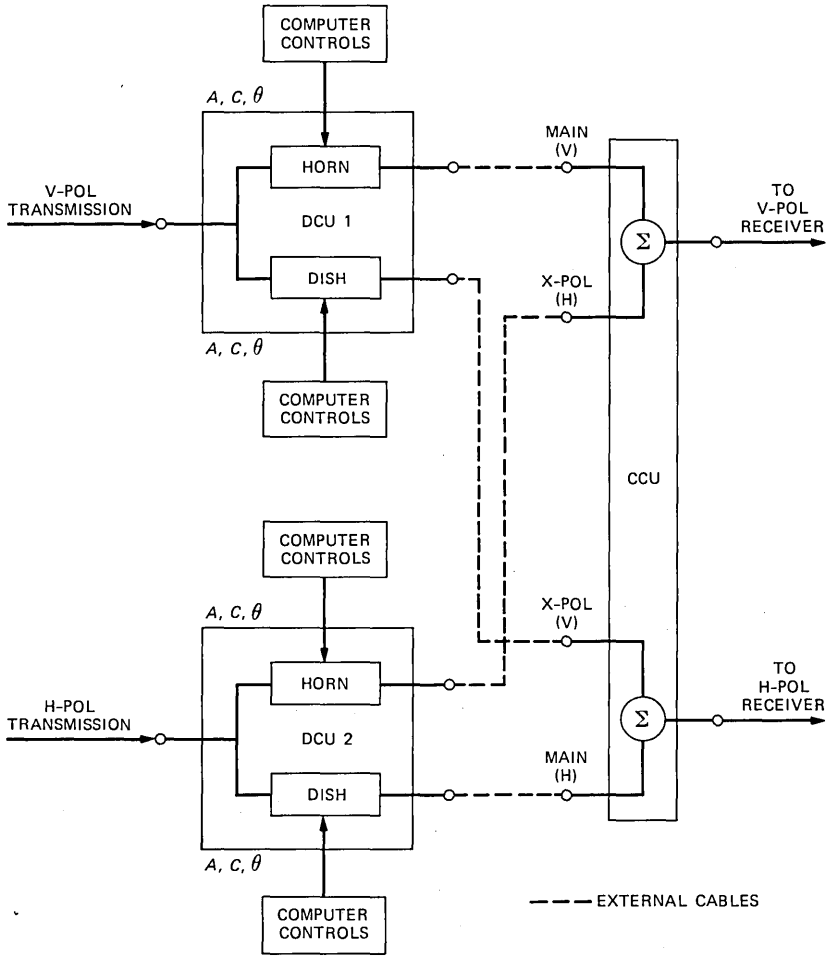


Fig. 2—Simulation mode of a dual-pol nondiversity channel.

1.3 Outline of the remaining material

Section II describes the architecture of the CSF, showing block diagrams of the three main units and discussing how they are used to provide the various possible modes of operation. Section III gives a more detailed description of the electronic circuits and components, while Section IV gives details on the construction. Section V discusses the software associated with the CSF. Specific topics include the generation of the sequences that drive the variable hardware parameters, and the calibration and measurement of the variable networks. The existing software for generating the sequences is tailored to a specific dual-diversity channel model. Section VI presents perform-

Table I—Definitions and some features of the channel simulation facility

Capabilities	Simulates either a single-pol nondiversity channel (1 input, 1 output); a single-pol diversity channel (1 input, 2 outputs); or a dual-pol nondiversity channel (2 inputs, 2 outputs).
Physical configuration	Consists of two identical dual channel units (DCUs) and one cross-connecting unit (CCU), plus connecting cables.
DCU (Multipath fade simulator)	Contains two parallel networks (common input, separate outputs), each producing a separate, variable "three-path" frequency response.
CCU ("Cross-pol coupler")	Connects the four outputs of two DCUs so as to simulate a dual-pol nondiversity channel.
Variability of the network responses	Computer-controlled; either keyboard entry fixed responses or program-generated time-varying responses.
Input IF	70 MHz
Input power level	0 dBm
Output IF	70 MHz or 1.070 GHz, as selected.
Output power	0 dBm for each variable channel network at 70 MHz. −15 dBm for each variable channel network at 1.070 GHz.
Bandwidth	Each network provides a "three-path" response over ±20 MHz about the intermediate frequency.

ance results. This includes specific data on bandwidths, power levels and calibration stability; assessments of how well the computer-generated sequences satisfy the underlying statistical model; and assessments of how well the hardware response variations match the intended ones, i.e., those dictated by the computer-generated outputs.

II. ARCHITECTURE

2.1 Dual channel units

Figure 3 shows, in simplified form, the block diagram of a DCU. A DCU contains two parallel networks with a common IF input, the top network being labeled *Horn Channel* and the bottom are being labeled *Dish Channel.* These labels are particularly apt when the two networks simulate the two paths of a dual space diversity link. For convenience, we will use these designations throughout our discussions to distinguish the top and bottom networks.

An actual DCU contains circuitry not depicted in Fig. 3, including bandlimiting IF filtering, upconversions from 70 MHz to 1.070 GHz, and the provision of the variable phase shifts (θ) via local oscillators and mixers. This circuitry will be discussed in Section III.

A control computer is used to control, through D/A conversion and low-pass filtering, each of the circuit parameter a, c and θ. It is thus

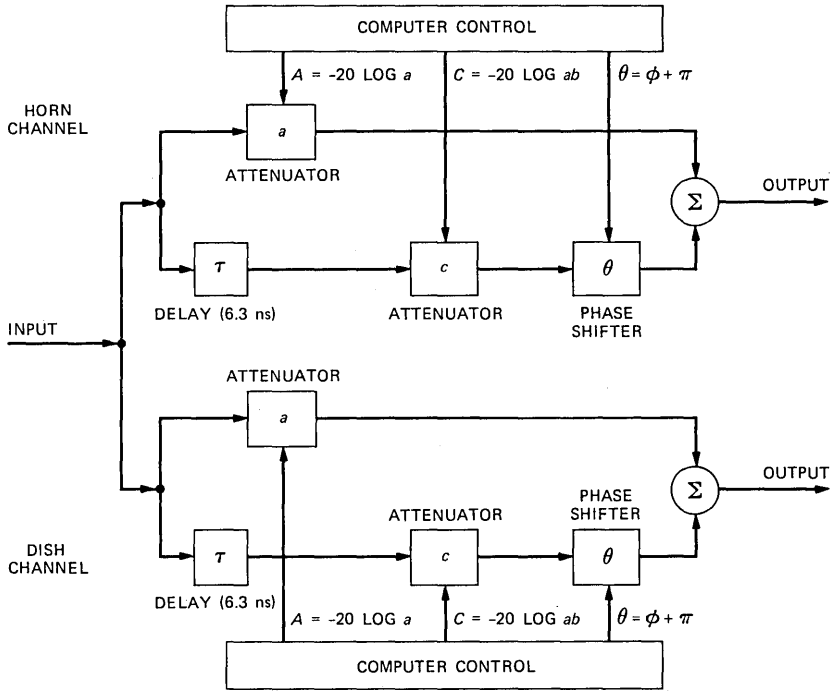


Fig. 3—Simplified block diagram of a dual channel unit.

clear that each of the two networks provides a frequency response identical in form to (4).

Finally, Fig. 3 shows the manner in which the intermediate frequency of the network outputs is selected. When the external connecting cables (shown dashed) are absent, the two outputs are at 1.070 GHz. When the cables are connected as shown, each 1.070-GHz output is applied to a separate downconverter, with a shared 1.0-GHz Local Oscillator (LO) providing the other input. In this case, the outputs will be at 70 MHz, the IF of the DCU input.

2.2 Modes of operation

Figure 1 shows two obvious ways to use a single DCU. In Fig. 1a, the control computer delivers fade parameters (A, C and θ) to just the Horn Channel network, and the output of just that network is applied to the follow-on equipment. Alternatively, one could control, and use the output from, just the Dish Channel. In either case, the simulator is used in this way to represent a single-pol nondiversity channel.

In Fig. 1b, the control computer delivers fade parameters to both the Horn *and* Dish Channel networks, and both outputs are used. In this case, the simulator can be configured as a dual space diversity

channel. On a typical space diversity link, the primary (upper) and secondary (lower) receiving antennas on the radio tower would be of the horn and dish type, respectively, thus leading to the designations used here.

Finally, Fig. 2 shows how two DCUs, combined with a CCU (described shortly) can be used to represent a dual-pol nondiversity channel: The path from the V-pol transmitter to the V-pol receiver passes through the Horn Channel of DCU 1. The interfering path from that transmitter into the H-pol receiver passes through the Dish Channel of the same DCU 1. Similarly, the H-pol path to the H-pol and V-pol receivers passes through the Dish and Horn Channels, respectively, of DCU 2.

It should be mentioned that the two IF inputs in Fig. 2 need not be frequency synchronous. In most practical dual-pol systems, the dually polarized signals have the same *nominal* carrier, but the carriers are not identical unless the same Radio Frequency (RF) source is used for each. The arrangement of Fig. 2 is amenable to either possibility. To complete the picture, we now describe the internal pathways of the CCU.

2.3 The cross-coupling unit

Figure 4 shows how the CCU combines four inputs to produce the simulated V-pol and H-pol outputs of a dual-pol channel: the co-pol V and cross-pol H signals are combined in a 10-dB directional coupler, following a manually set attenuation and phase shift of the latter, to produce the V-pol receiver input. Similar combining produces the H-pol receiver input.

The manual attenuation adjustments in the CCU, achieved via panel-mounted controls, extend the dynamic range over which the cross-pol coupling levels can be varied. Each attenuator can be incremented in 1-dB steps from 0 dB to 43 dB.

The manual phase shift adjustments, also achieved via panel-mounted controls, provide flexibility in how the co-pol and cross-pol signals from the same original transmission are relatively phased. The phase adjustment is continuous over a range of nearly 360 degrees.

III. CIRCUIT DESIGN

3.1 General

The CSF could have been designed in any number of ways. We have taken a particular approach that combines practical circuitry, wide bandwidth, accurate performance and operational flexibility. We will elaborate here on some of the circuitry that comprises a DCU. The circuitry of the CCU was adequately described in the previous section.

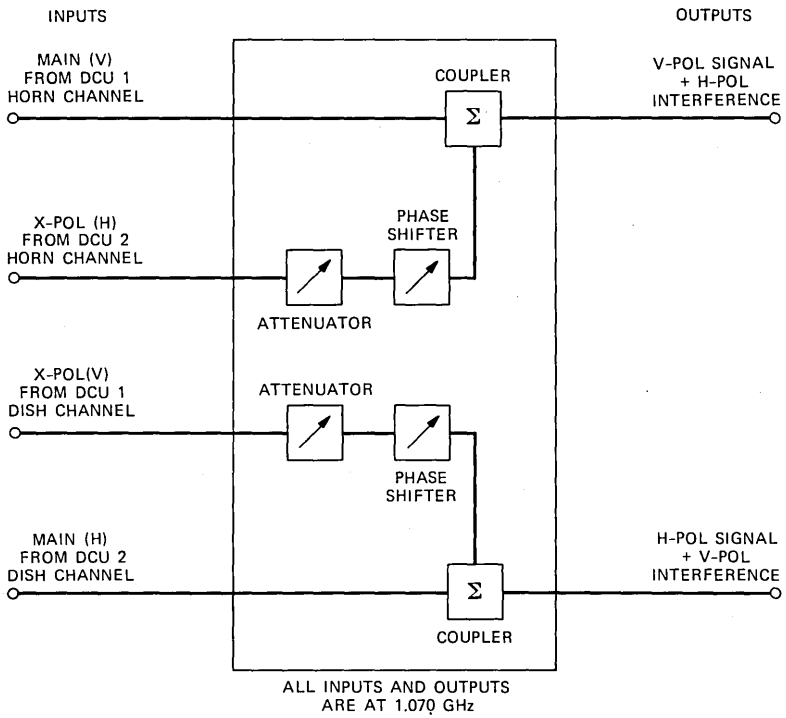


Fig. 4—Block diagram of the cross-coupling unit.

Each variable channel network of a DCU contains a direct path with a variable (analog voltage controlled) attenuator, and a parallel path with a 6.3-ns delay, a variable attenuator *and* a variable (analog voltage controlled) phase shifter. The summed outputs of these paths yield a network response akin to the "three path" function of Rummler. Each of the parallel paths must be nondispersive, in fact, over a bandwidth of 40 MHz or more. Further, it should be possible to vary each attenuator without introducing phase changes and to vary the phase shifter without introducing attenuation changes.

An input frequency of 70 MHz and a channel bandwidth of 40 MHz were chosen for the CSF. The variable channel networks could be built at this or any other frequency if ideal components were available. However, given the limitations of practical components, we chose the approach of converting the input signal frequency to a higher frequency, where narrowband circuit techniques can be used. This approach, moreover, simplifies the task of providing the variable phase shift, as we shall see.

A network frequency of 1.070 GHz was chosen. This choice trades off the availability of "phase shift free" narrowband attenuators, and

ease of filtering of unwanted mixing products and local oscillator leakage through the mixer. The network output at 1.070 GHz can be either downconverted back to 70 MHz, used directly, or easily upconverted to, say, 4, 6, or 11 GHz for radio equipment tests.

3.2 The variable channel network

A simplified block diagram of a single variable channel network is shown in Fig. 5. The input signal at 70 MHz (Point 1) is power divided into two paths. One component is upconverted in a double-balanced diode mixer to 1.070 GHz; the other is delayed and then similarly upconverted to 1.070 GHz by a second double-balanced diode mixer. The output from the first mixer is controlled in amplitude by a variable attenuator and provides one input to a signal combiner. The output from the second mixer (in the delayed path) is controlled in amplitude by a second attenuator and in relative phase by the local oscillator phase shifter. This delayed signal sums with the direct input in the output signal combiner.

The 6.3-ns delay is provided by adding additional coaxial cable in the 70-MHz signal path between the input power divider and the delayed path upconverter. The exact length of this cable is determined empirically by measuring the signal delay difference at the output signal combiner (points 2 and 3). This is most easily done in the frequency domain by sweeping the network input signal, setting the

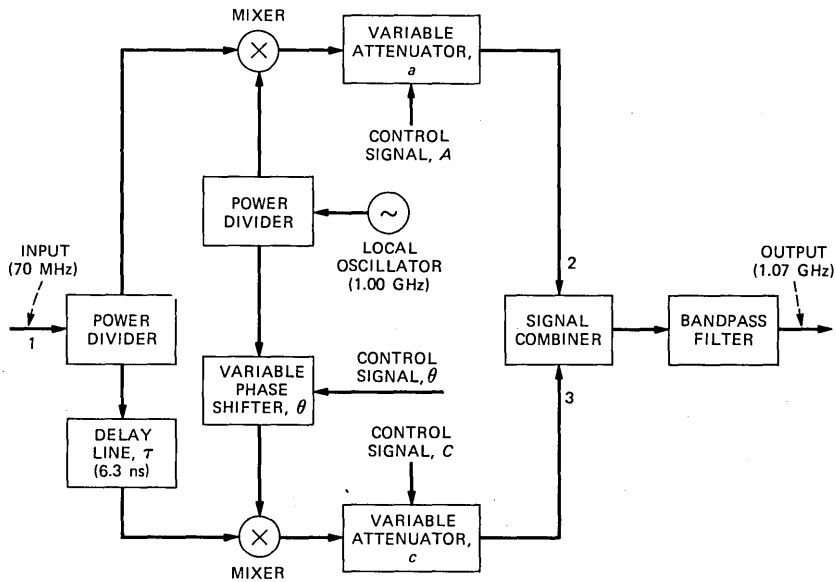


Fig. 5—Simplified block diagram of a single variable channel network.

direct path and delayed path attenuations to be about equal (to form periodic deep fades), and then measuring the frequency spacing between successive nulls. When the path difference is 6.3 ns, the null spacing is 158.73 MHz, which is the reciprocal of the delay.

The output of the variable channel network contains both the upper and lower mixing products at 1.070 GHz and 0.930 GHz. If the 1.070-GHz signal is to be mixed back to 70 MHz, the contribution from the 0.930-GHz signal will distort the desired channel response and must be effectively eliminated by filtering.

3.3 Detailed description of a DCU

A detailed block diagram of a DCU is shown in Fig. 6. This diagram shows the two variable channel networks driven by a common 70-MHz input signal and the two downconverters for mixing the simulator outputs back to the input frequency. A crystal-controlled phase-locked Continuous Wave (CW) source at 1 GHz acts as a common local oscillator to all upconverters and downconverters, to ensure frequency coherence between the input signal and both output signals. When both DCUs are used in the complete CSF, they are virtually identical except for the provision to drive all four variable channel networks from a common local oscillator residing in one DCU (master). This higher power source in the master DCU can drive the second DCU (slave). The result is four frequency-coherent outputs, assuming the input signals to the two DCUs are themselves frequency coherent. (As described in Section 2.2, the two input signals to the DCUs may or may not be frequency coherent, depending on the experiment.)

Each variable network uses two local oscillator signals for upconversion. One is derived through power dividers from the 1-GHz source; the other passes first through a voltage-controlled phase shifter. The phase shifter varies the phase of the 1-GHz CW signal from 0 degrees to over 360 degrees, when the θ control signal varies between 0 and +10 volts. This phase is imparted to the signal in the delayed path of the variable network by the mixing process of the upconverter.

The signals are scaled in the direct and delayed paths of each variable channel network by voltage-controlled attenuators. The attenuations are variable from 0 dB to over 36 dB when the A and C control signals vary between 0 and +10 volts. The outputs from the attenuators are added in a reactive signal combiner. The combined signal is bandpass filtered to eliminate the lower mixing product at 0.930 GHz. The desired output from the filter at 1.070 GHz is amplified to provide the final fading channel output.

The output filter, a fifth-order Butterworth filter with a 3-dB bandwidth of 70 MHz, is the most narrowband circuit of each variable network. This filter adequately attenuates the unwanted mixing prod-

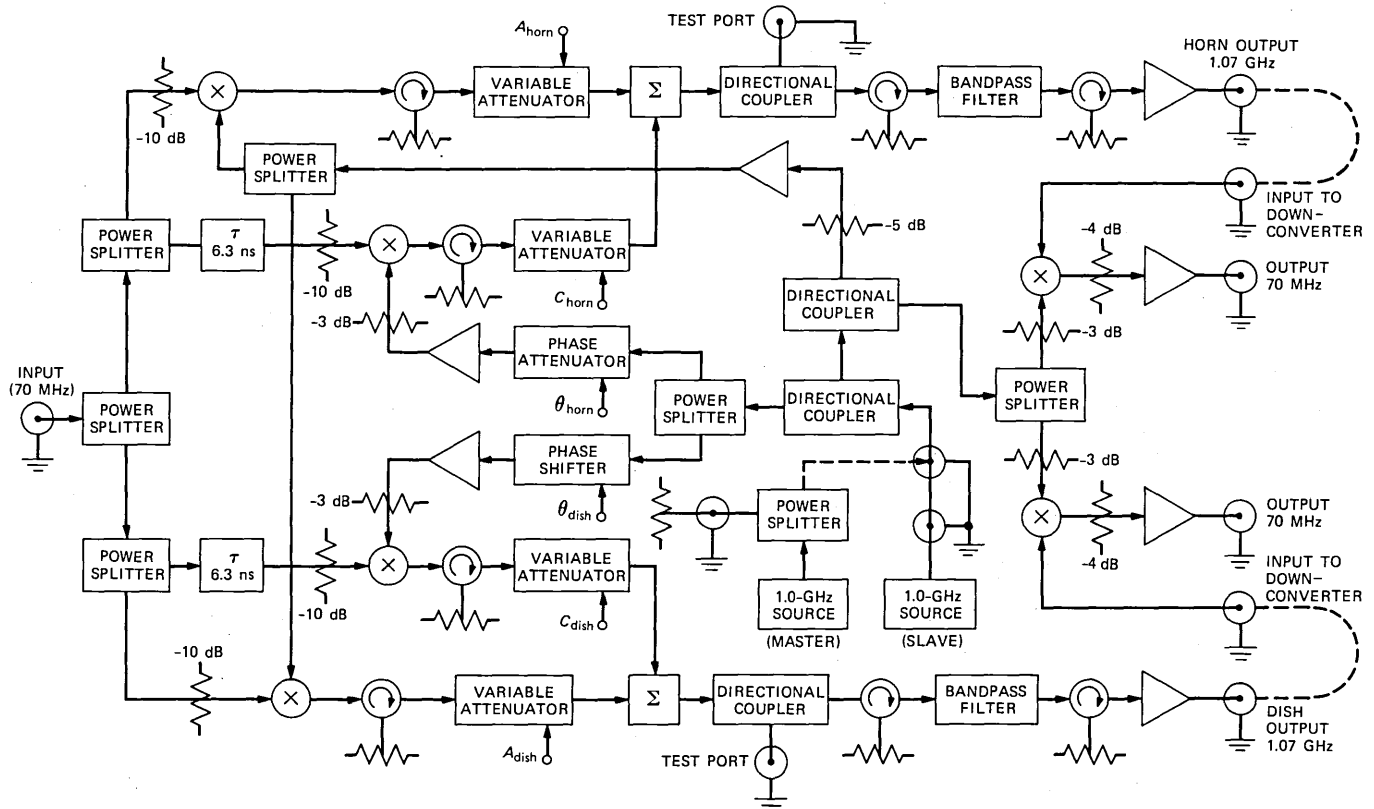


Fig. 6—Detailed block diagram of a dual channel unit.

uct while maintaining a flat amplitude response with small time delay variation (±1.5 ns) over the design bandwidth of 40 MHz.

Care was taken to isolate components in the various signal paths, to prevent reflections from mismatches that produce unwanted ripples in the channel response. Fixed coaxial attenuators and ferrite isolators were used where necessary to minimize reflections.

3.4 Interface circuits

As we have seen, the four attenuators and two phase shifters within each DCU are controlled by time-varying analog voltages, the latter being derived from digital sequences supplied by the control computer. To accomplish this, two D/A converters mounted within the computer provide six control voltage sequences to each DCU. Each sequence consists of 12-bit words delivered at a 1-Hz rate. Each sequence is converted into a smooth analog variation by a fourth-order Butterworth state variable low-pass filter with a 3-dB bandwidth of 0.5 Hz.

IV. CONSTRUCTION

4.1 Construction of the DCU

Figure 7 shows a top view of the inside of a DCU. Semirigid coaxial

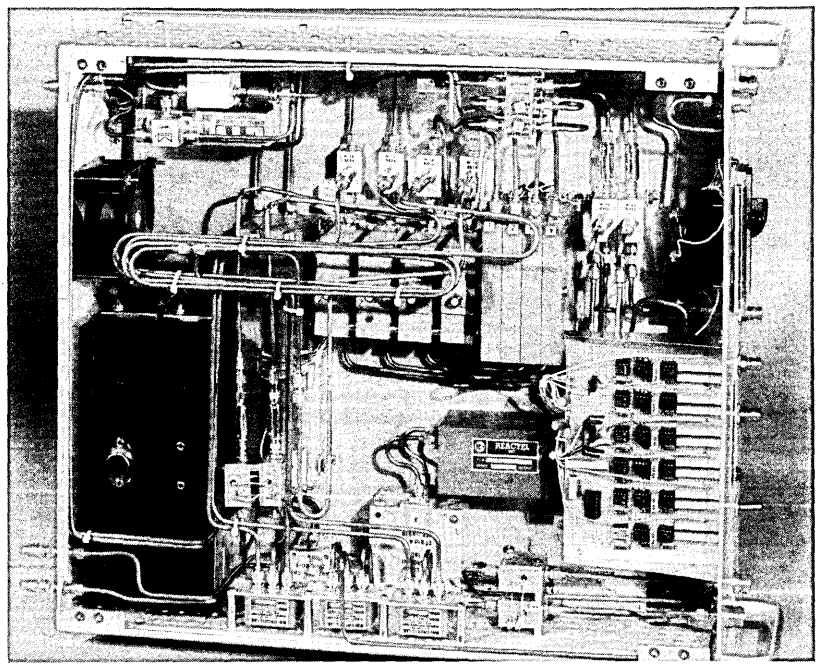


Fig. 7—Top view of the inside of a dual channel unit.

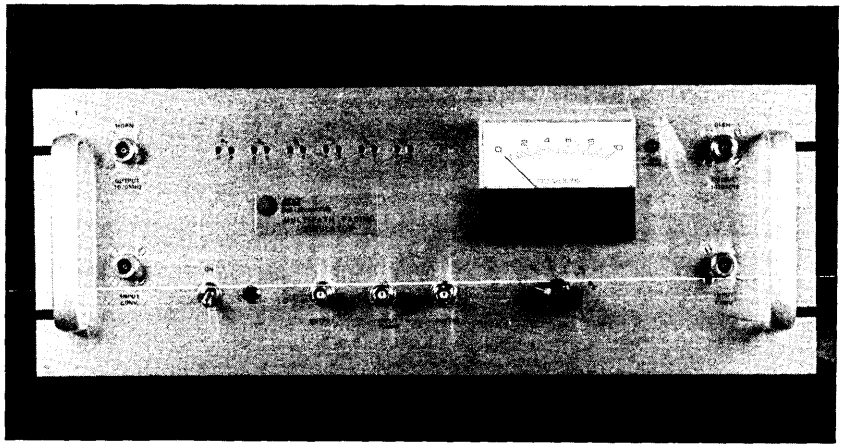


Fig. 8—Front view of a dual channel unit.

line (.141 inch) was used to interconnect components to improve phase stability. Line lengths in both the direct and delay paths were closely matched in each variable channel network of each DCU, so as to obtain nearly identical characteristics. The elongated coils in the center of the figure are the additional line lengths required to produce the 6.3-ns path delays.

A view of the front panel of the dual channel unit is shown in Fig. 8. The input and output signal ports are available on the panel for interconnection convenience. The 70-MHz input port and the 70-MHz downconverter output ports are the three BNC-type coaxial connectors in the lower center of the panel. The 1.070-GHz outputs from the Horn and Dish variable channel networks are brought out on N-type coaxial connectors located at the upper left and upper right sides of the panel. The inputs to the two downconverters associated with the Horn and Dish channels are located directly below these N-type connectors.

Screw-trimmer adjustments, paired in 12 holes in the upper center left of the panel, are provided for shifting and scaling the control voltages to the four variable attenuators and two phase shifters. The range of these control voltages is approximately preset by adjustments on the D/A converter boards within the control computer. The front panel adjustments allow for convenient trimming at the DCU. The panel meter can be selectively switched to monitor each of these voltages.

4.2 Construction of the CCU

Figures 9 and 10 show a top view of the inside and the front panel of the cross-coupling unit. This relatively simple unit contains the two

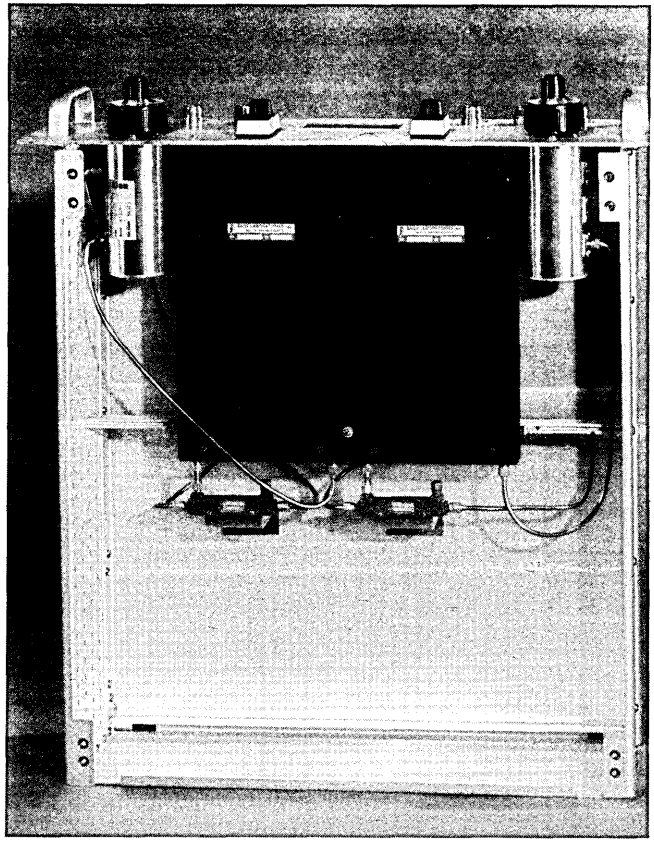


Fig. 9—Top view of the inside of the cross-coupling unit.

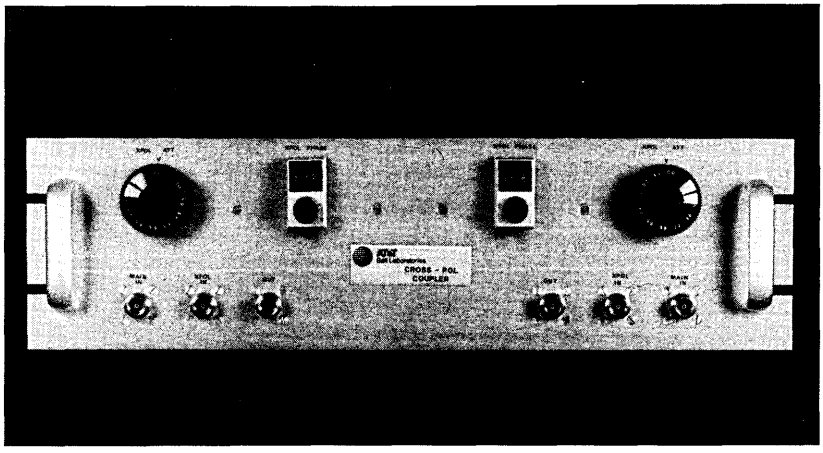


Fig. 10—Front view of the cross-coupling unit.

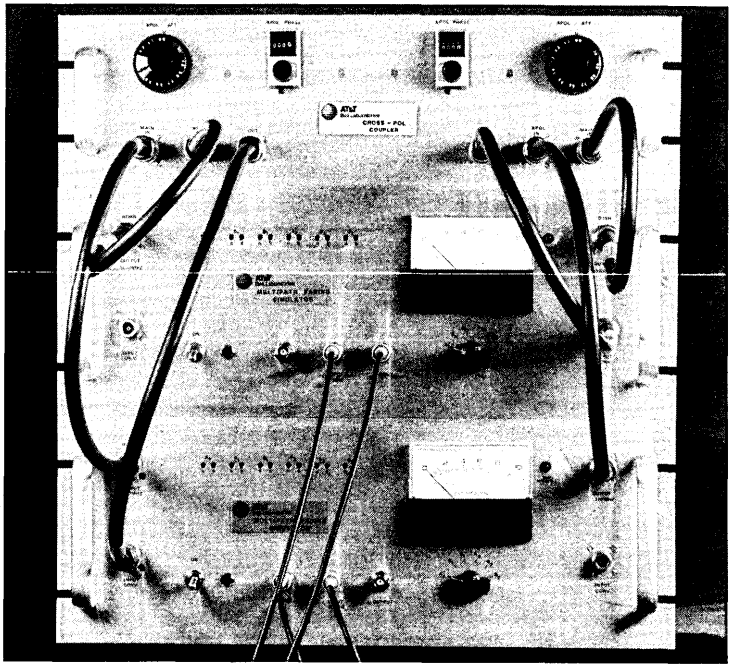


Fig. 11—Front view of the complete channel simulation facility.

cross-pol phase shifters (large rectangular devices in the center), the two cross-pol attenuators (cylindrical devices adjacent to the phase shifters) and two 10-dB directional couplers.

4.3 Interconnections for the CSF

The front view of the complete CSF is shown in Fig. 11. The cable interconnections shown represent the cross-coupled case described in Section 2.2. The lower DCU can be arbitrarily called the V-pol channel and the upper DCU, the H-pol channel. The V-pol Horn Channel output is connected to the left cross-pol coupler Main input. The H-pol Horn Channel output is connected to the left cross-pol coupler X-Pol input, to be attenuated and phase-shifted before summing to the main V-pol signal. The dish and horn outputs from the H-pol and V-pol channels, respectively, are similarly combined in the right cross-pol coupler. The cross-pol coupler outputs are shown returned to the respective downconverters in the DCUs to provide 70-MHz outputs. If desired, instead, the 1.070-GHz outputs from the coupler can be used directly.

V. SOFTWARE DESIGN

This section deals with two distinct aspects of the CSF software.

One is the computer routines that generate random time sequences for parameters of the specified channel models; the other is the software that runs on the control computer to effect real-time operation of the CSF hardware.

The sequence generation software was developed on the *UNIX*™ operating system with the option to move it to the control computer. In the initial arrangement, however, the fade parameter sequences were generated on a *UNIX* system minicomputer and down loaded to the control computer. As described earlier, these sequences were converted into continuous analog signals and played into the simulator hardware to achieve time-varying channel responses. Special software was written to test the accuracy of the generated sequences and of the resulting hardware responses. These issues are dealt with in Section VI.

5.1 Generating the random fade parameter sequences

We describe here how to generate the random time-varying parameters for a fade model. We will first summarize the generation procedure employed by our software to generate random variations with any given Probability Density Function (PDF) and Autocorrelation Function (ACF). Then we will describe the techniques used to generate random variations with PDFs and ACFs specified by Rummler.[7] In this subsection, we show how to generate the random variations A, B, and ϕ, as defined in (2), for both the Horn and Dish Channels. The transformation of these variations into A, C, and θ is performed by the control computer and is described briefly in a later section on control computer software.

5.1.1 Iterative random variation generation method

The problem of generating a random variation with arbitrary PDF and ACF is difficult in general and sometimes impossible.[32] In our case, however, we wish to match the statistics of fade models with well-behaved PDFs and ACFs. Further, while accurate matching of the PDF is essential, particularly when the simulator is used for outage measurements, slight variations in the ACF shape are acceptable since the ACF is either unknown or only sparsely measured for many models. These considerations ease the variation generation problem considerably. It should be noted that the second-order statistics are not uniquely determined by the PDF and ACF and that we will accept whatever second-order statistics are generated when matching the PDF and ACF.

This method is general in that it generates random variations with any specified PDF and with an ACF close to the specified ACF. The method starts by generating a Gaussian random process with an initial

ACF that we normally take as the desired ACF. From the Gaussian variation, a new variation with the required PDF can be formed by a nonlinear transformation. It is easier to think of this transformation in two steps—a Gaussian-to-uniform transformation followed by a transformation to the required PDF. The first transformation involves using the Q function, while the second transformation involves using the inverse of the required distribution function. While this gives variations with the correct PDF, the nonlinear transformations will make the output ACF different from the initial ACF.

To make the output ACF match the desired ACF, we use the following iterative procedure: The initial ACF is applied to the underlying Gaussian variation. Based on the output ACF and an educated guess, we select a new ACF for the underlying Gaussian variation. This procedure stops when the desired ACF is obtained. In our case, the iterative procedure converges quickly. We found that the ACFs of two of our parameters remained essentially unchanged after the nonlinear transformations.

We describe the first part of this procedure in detail here and in the following subsection, and continue the detailed discussion by way of an example for clarity. We begin the procedure by generating a Gaussian variation. Many techniques can be used here and are well known, although we chose one similar to that used by Vannucci and Teich.[33] The ACF for the Gaussian variation generation is initialized, usually to the desired ACF.

The nonlinear transformation required to convert the Gaussian variations into variations with the desired PDF is broken up into a transformation to uniform followed by a transformation to the desired PDF. The transformation to uniform can be accomplished through the Q function, i.e.,

$$u = Q(x) = \frac{1}{\sqrt{2}} \int_{-\infty}^{x} e^{-\frac{y^2}{2}} dy, \tag{6}$$

where x is the original Gaussian variation and u is the uniform variation. This function can also be used as a test of the quality of the Gaussian variation by computing the new probability distribution function and comparing the result with the uniform probability distribution function. This test was used to verify the uniform-to-Gaussian random variation conversions.

5.1.2 Implementation of a space diversity fade model

We chose to use the Rummler dual-channel space diversity fade model,[7] although Rummler's rationalized model[8] or any other dual channel model could have been employed as well. Familiarity with Rummler models is assumed in the following paragraphs.

Since ACF measurements are not available for this model but are available for the single-channel model,[34] we use the single-channel model ACFs for both channels of the diversity model. These ACFs, measured by Rummler, are not true ACFs; they were rescaled so that $R(0) = 1$. The scaling factor was determined by extrapolating the measured ACF to the ordinate and finding the intercept. The ACF for each of the fade parameters closely approximates an exponential. For this reason, we applied an exponential ACF to each underlying Gaussian variation and adjusted its time constant to form different ACFs. Matching the time constants at the $1/e$ points of the rescaled Rummler ACFs gives time constants of 129, 53, and 32 seconds for the respective A, B, and ϕ parameters. These ACFs were used in generating the underlying Gaussian variation discussed above.

The easiest parameters to generate are the ϕ's because they are independent of each other and of all the other parameters. For each ϕ, we begin with $U(0, 1)$ variation, u, generated as discussed in the previous subsection. The ϕ parameter, with a pedestal-type PDF, is generated from the u variation by the simple rescaling

$$\phi(u) = \begin{cases} 90(1 + \alpha)(2u - 1)/\alpha; & |2u - 1)| \leq \alpha/(1 + \alpha) \\ 90(1 + \alpha)(2u - 1) + 90(1 - \alpha)\text{sgn}(2u - 1); & \text{otherwise,} \end{cases} \quad (7)$$

where α is the pedestal parameter (5 for the Horn Channel and 8 for the Dish Channel) and ϕ is in degrees.

The B parameters are generated next because, in both the Horn and Dish Channels, the A parameter's mean value is dependent on the B parameter. The general technique for converting a $U(0, 1)$ random variation to a particular distribution can be accomplished by passing each sample through the inverse of the distribution function. For the B parameters, finding the inverse function algebraically is rather complicated, so this was done numerically.

Finally, the A parameters are generated. For this case, the final distribution is Gaussian so no transformations are need to obtain the correct PDF. However, the mean of each A is dependent on the associated value of B. Further, A_{horn} is correlated with A_{dish}. To generate the A's, two independent $N(0, 1)$ random variations (x_1 and x_2) with identical ACFs are generated. Next, they are linearly combined to produce the desired correlation and, finally, the variances are scaled and appropriate mean values are added. In short,

$$A_{\text{horn}} = \sigma_{\text{horn}} x_1 + \mu_{\text{horn}}(B_{\text{horn}}) \quad (8)$$

$$A_{\text{dish}} = \rho\sigma_{\text{dish}} x_1 + \sigma_{\text{dish}} \sqrt{1 - \rho^2} x_2 + \mu_{\text{dish}}(B_{\text{dish}}), \quad (9)$$

where the Greek constants and the $\mu(B)$ functions are defined in Ref. 7.

5.2 Control computer software

The control computer software falls into two categories, namely software to produce the fades and software to calibrate and test the hardware. The purpose of the fade software is (1) to translate fade model variations (e.g., A, B, and ϕ) into the A, C, and θ variations supported by the hardware; (2) to perform calibration table lookup and quantization to the closest binary output sample value; and (3) to play the binary variations through the analog hardware. The remaining software generates the calibration tables, tests the overall accuracy of the fade control and calibration software, as well as the analog hardware, and allows manual control over the fade simulator. These operations are described in more detail in the following subsections.

5.2.1 Control computer—fade software

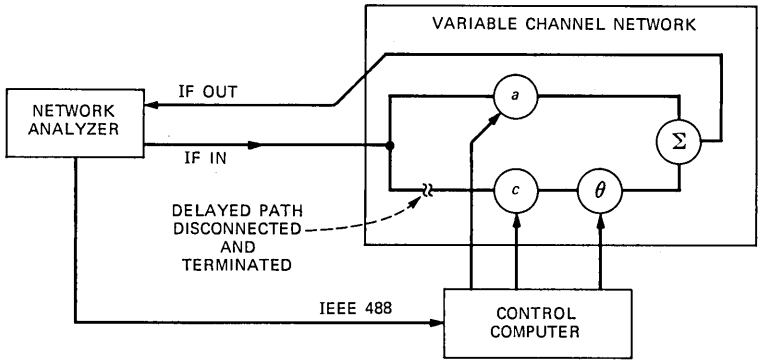
The analog hardware implements the transfer function given by (4), (see Fig. 5), where $c = ab$ and $\theta = \phi + \pi$. The voltage-controlled attenuators have control signals proportional to 10 log (power attenuation). These final log conversions are handled by a lookup table (described shortly), but the software-generated parameters A and B are in decibel form and must first be converted to linear form before computing c. The phase offset of π is required in the hardware to ensure continuity into the D/A low-pass filter.

To ensure accuracy, calibration lookup tables are used to map and quantize the analog floating point values generated from the fade model software into one of the integer output values of the D/A converter. Software automatically finds the closest table entry to the floating point value and outputs the corresponding integer value. Separate software outputs these integer variations from RAM memory or disk to the D/A converters at a 1-Hz rate.

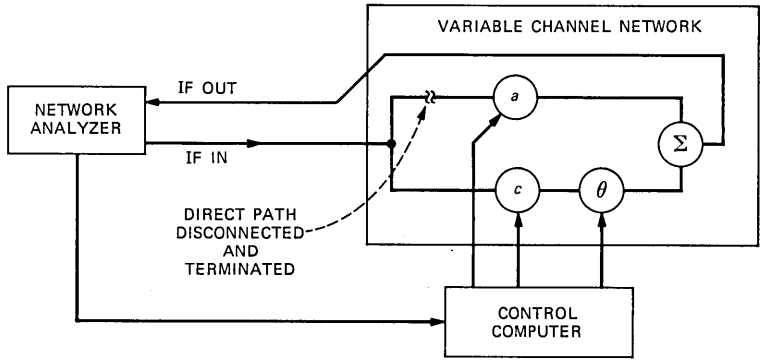
5.2.2 Control computer—calibration and test software

The calibration lookup tables discussed above are automatically generated under software control via hardware measurements taken with a network analyzer connected to a computer. The a and c parameter calibrations are similar as shown in Figs. 12a and b and require the disconnection and termination of the delayed path (direct path for the c parameter). The control computer then measures the end-to-end attenuation of the connected path for each of the 4096 values and generates a table. These values are smoothed by averaging within a window centered around each value. This is required to obtain a monotonic table, especially for large attenuations where measurements become noisy.

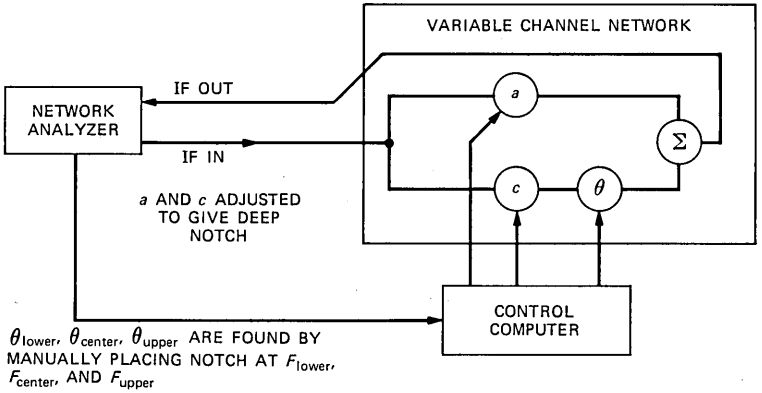
To simplify the software, the θ calibration was made interactive. With both paths connected as in Fig. 12c, a spectral dip or notch is



(a) CALIBRATION OF a



(b) CALIBRATION OF c



(c) CALIBRATION OF θ

Fig. 12—Block diagrams of parameter calibrations: (a) calibration of a, (b) calibration of c, and (c) calibration of θ.

formed by manually adjusting the relative values of a and c. This notch is placed at the upper and lower band edges as well as the center of the channel and the binary control values are stored. This information is then used to generate the lookup table via interpolation and extrapolation. The extrapolation is needed since notch position measurements can only be made in the bandlimited channel.

Additional software allows for manual control over each of the parameters, A, C, and θ, in real time. This provides for quick checking of the hardware and the lookup tables. This software also provides a useful mode of operation for the simulator, allowing selected responses to be easily "dialed" into the hardware.

Finally, there is software for playing a selected response and comparing it with the actual response measured by the network analyzer to obtain an error measure. More details about this are contained in the performance section that follows.

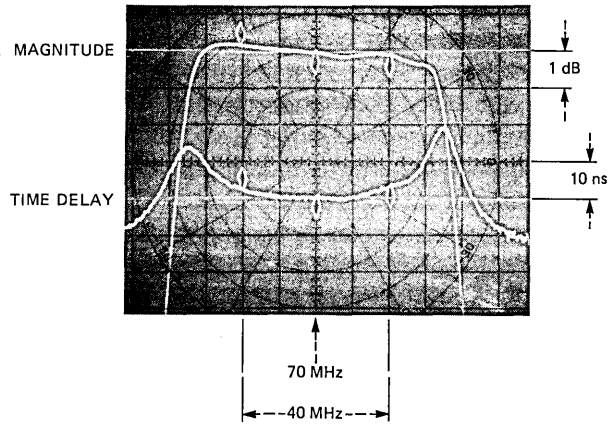
VI. PERFORMANCE

We will evaluate the performance of the CSF in several ways here. First, we illustrate some "three-path" responses produced by the variable channel networks over the 40-MHz bandwidth of interest. Next, we demonstrate that the joint statistics of the computer-generated sequences of fade variables (A, B and ϕ) conform to the model from which they are derived. Finally, we demonstrate that the frequency responses of the hardware, over a computer-generated population of A, C, θ values, are indeed the responses these fade variables are intended to produce. With these demonstrations, we affirm that the CSF can produce laboratory fading environments similar to those on actual digital radio links.

6.1 Channel frequency responses

The static and dynamic transmission performance of each DCU was measured using a Hewlett-Packard Model 8505A Network Analyzer. Here we discuss frequency response measurements on one or both variable channel networks of a DCU, and show sample results.

In a typical measurement, fixed control voltages are set by keyboard inputs on the control computer, and applied to the attenuators and phase shifters of the variable channel networks. The values of the control voltages produce a particular transmission function in each network. The Network Analyzer provides a frequency-swept input signal, in the vicinity of 70 MHz, to the DCU. The two 70-MHz outputs from the DCU are returned to the analyzer, where the frequency response magnitudes of the two networks are simultaneously observed. Alternatively, only one network output from the DCU is returned to the analyzer, along with a reference sample of the input



DIRECT BRANCH ATTENUATION 3 dB
DELAYED BRANCH ATTENUATION > 36 dB

Fig. 13—Frequency response of one channel of a dual channel unit.

signal, to permit measurement of both the transmission magnitude *and* phase (or group delay) of that network.

In our initial experiment, we set the control voltage of the delay path attenuator so as to give maximum attenuation (>36 dB). This enabled us to measure the transmission performance of the direct path alone, with the output filter and downconverter included. Figure 13 shows the corresponding frequency response for one channel of a DCU. The upper trace is the magnitude and the lower trace is the group delay. Over a 40-MHz bandwidth centered at 70 MHz, the magnitude is flat within ±0.12 dB and the group delay is flat within ±1.5 ns. This response is primarily determined by the output Butterworth filter described in Section 3.3.

In another experiment, we set the control voltage to the delay path attenuator so that the signal level at the variable channel network combiner was 0.92 dB weaker than that of the direct path. A minimum-phase fade was thus created, with a maximum fade depth 20 dB below the direct path gain. The upper trace in Fig. 14 shows the magnitude of the resulting response. The delay path phase shifter was set to place the maximum fade in the center of the channel passband. The lower trace shows the channel group delay, wherein the time delay distortion can be clearly seen: Frequency components at the edges of the channel are delayed by more than 60 ns relative to components in the center of the channel.

When the signal levels in the direct and delay paths are interchanged so that the delayed signal to the network combiner is 0.92 dB *stronger*,

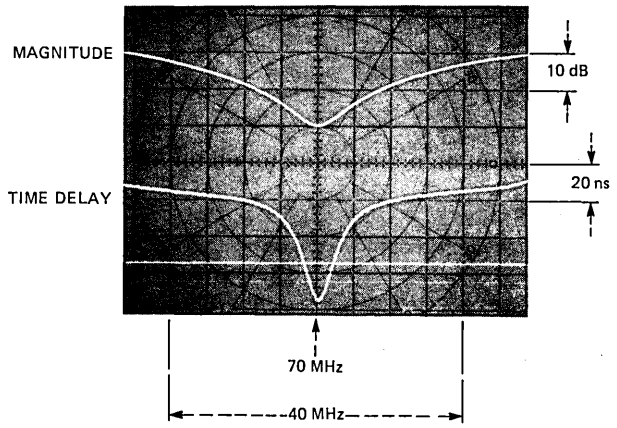


Fig. 14—A single-channel response for a 20-dB minimum phase fade.

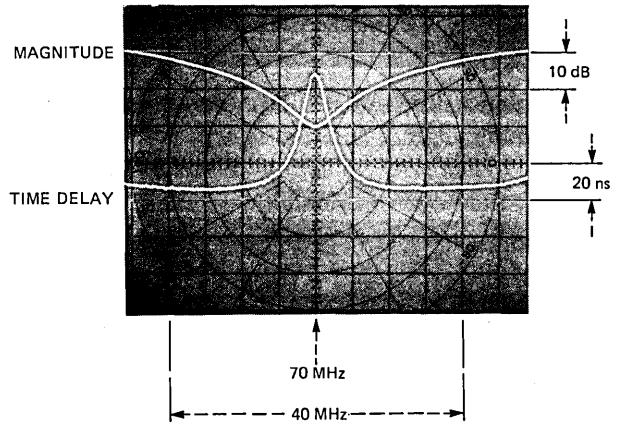


Fig. 15—A single-channel response for a 20-dB nonminimum phase fade.

a nonminimum-phase fade occurs, with the same maximum fade depth of 20 dB. The upper trace in Fig. 15 shows the transmission magnitude of the resulting channel response, with the maximum fade placed once more in the center of the channel passband. The time delay distortion is shown in the lower trace. Under these conditions, signal components at the center of the channel are delayed relative to components at the edges of the channel.

In general, by appropriately setting the relative signal levels in the direct and delay paths of a variable channel network, the depth of fade in the channel response can be selected. This fade can then be positioned in frequency by appropriately setting the phase shift in the delay path. Figure 16a shows the transmission magnitude of the

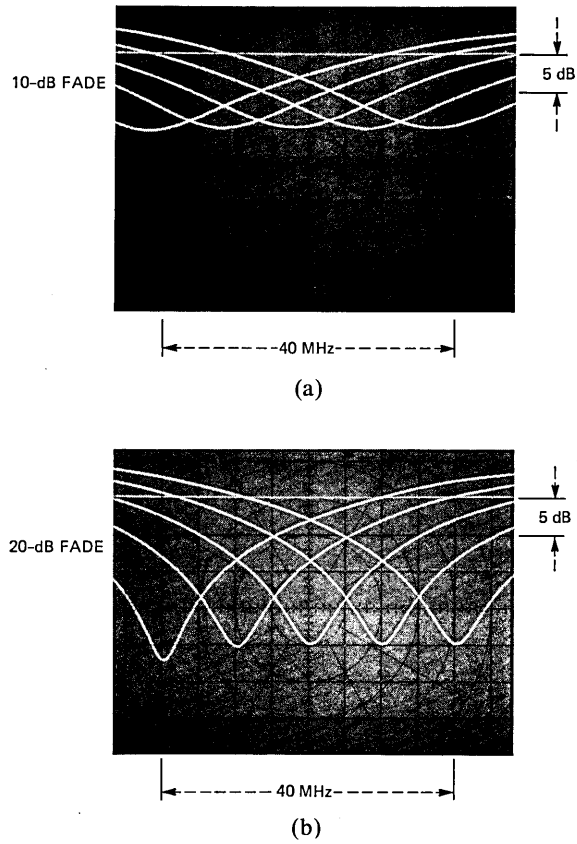


Fig. 16—Transmission magnitude response as a fade is moved in frequency across channel.

channel response as a 10-dB fade is moved in frequency across the channel. The fade depth remains constant within better than ±0.25 dB over the 40-MHz channel bandwidth. If the delay path signal level is increased in magnitude to produce a 20-dB fade depth, the variation in fade depth across the channel increases to ±1.0 dB, as shown in Fig. 16b. These variations are due to minor amplitude response imbalances between the direct and delay paths. The ±1.0-dB variation for a 20-dB fade depth, for example, is caused by imbalances of about ±0.1 dB.

6.2 Signal levels

The DCU was designed to be a general-purpose laboratory instrument. A nominal input signal level of 0 dBm was chosen for the 70-MHz DCU input. This level is normally available from the modulator

portion of typical radio modems. The unfaded 70-MHz output from each network of the DCU is designed to have a level of 0 dBm when there is 3-dB attenuation in the direct path. This 3 dB can be removed, under software control, to provide some up-fade capability. The unfaded outputs at the alternative frequency of 1.070 GHz are designed for a level of −15 dBm.

Finally, test ports are provided to permit frequency response measurements of each variable network prior to the 1.070-GHz bandpass filter. The signal level at each of these ports is nominally −52 dBm.

6.3 Software accuracy: matching the model statistics

Using the methods described in Section 5.1, random variations for each parameter of the Rummler space diversity model were generated. Figures 17 and 18 show the distributions of the uniform variation, and the Rayleigh plus exponential variation representing B_{horn}. B_{horn} was nonlinearly transformed to produce, in theory, a linear distribution function. This check on the accuracy was performed and the result, not shown, was very nearly linear.

The distributions were obtained from a single variation containing 50,000 samples, which corresponds to about 1666 independent samples for the ACF used. Similar results were obtained for the other fade parameters but are not shown for brevity.

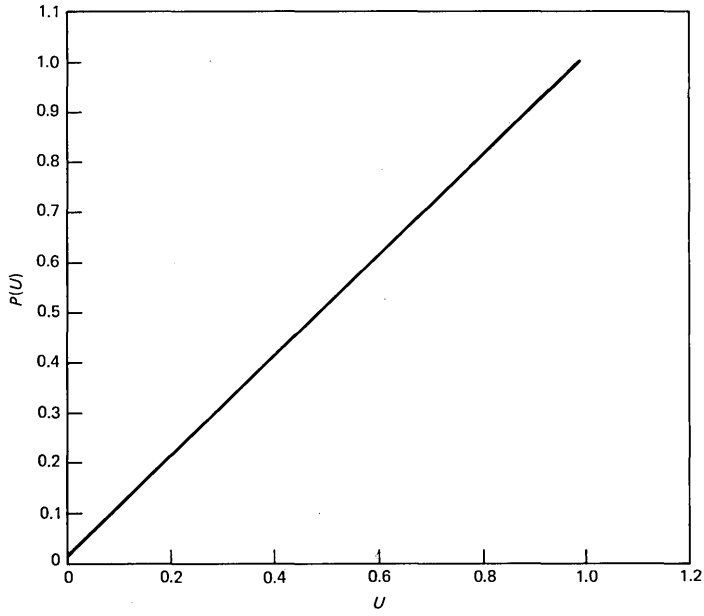


Fig. 17—Distribution of the underlining uniform for B_{horn}.

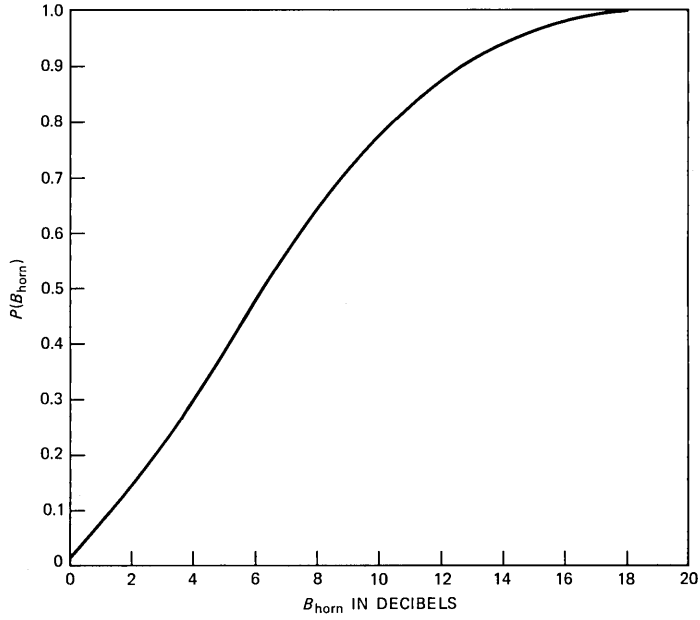


Fig. 18—Final distribution of the B_{horn} parameter.

For the pedestal-type density functions for ϕ, accuracy was checked directly by measuring the slopes of the final distribution functions. This yielded the desired values of 5 and 8 for the density function pedestal ratios for the Horn and Dish Channels. The A parameter distributions were checked using the same Gaussian-to-uniform transformation as in the generation procedures for B and ϕ. Again, a linear plot should result and was obtained.

The ACFs for B_{dish} are shown in Fig. 19. The time separation among ACFs for the underlying Gaussian, uniform, and final variations is about 2 seconds for this parameter. Results for B_{horn} and the ϕ parameters are similar but show less variation. The underlying Gaussians variations were adjusted so that their ACFs agree with the fade model ACFs at the 1/e points.

The ACF for the Gaussian variations of A_{horn} and A_{dish} must be the same so that the resulting distributions both follow the required ACF. Therefore, the ACF time constants for the Horn and Dish Channels are set equal and adjusted together. It was found that changes in ACF resulted when the A parameters were correlated and then each was biased by a mean related to B, as specified by the model. This is shown in Fig. 20.

Figures 21 through 23 show the differences between a smoothed version of the prescribed Rummler single-channel ACFs and the

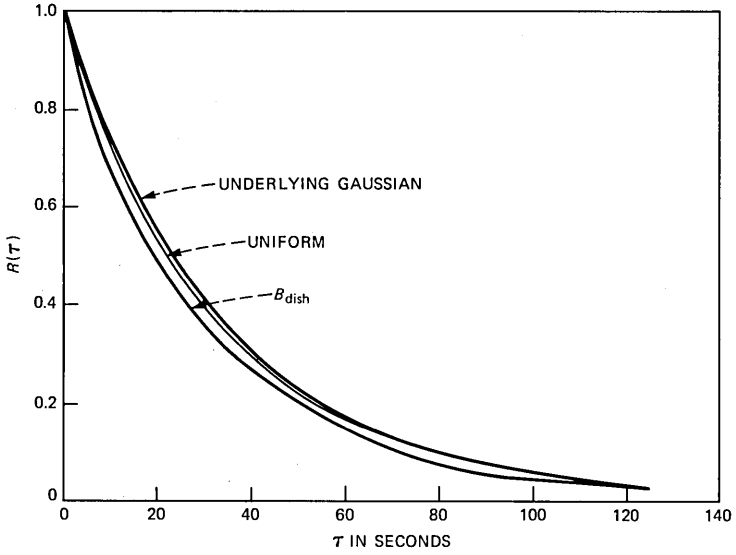


Fig. 19—ACF for the underlying Gaussian, uniform and B_{dish}.

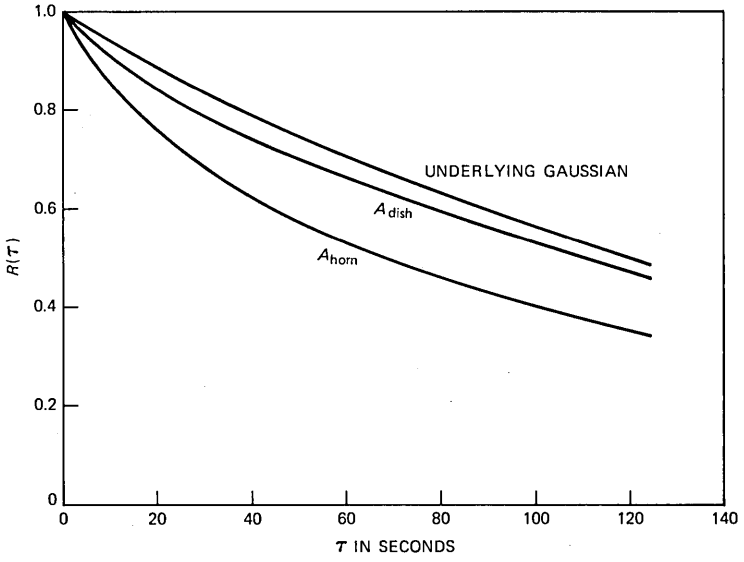


Fig. 20—ACF for the underlying Gaussian, A_{dish} and A_{horn}.

resulting ACFs for the Horn and Dish Channels for each of the A, B and ϕ parameters, respectively. The agreements are quite good.

Thus, we have carried out our parameter generation method for the Rummler dual channel space diversity model and have verified its

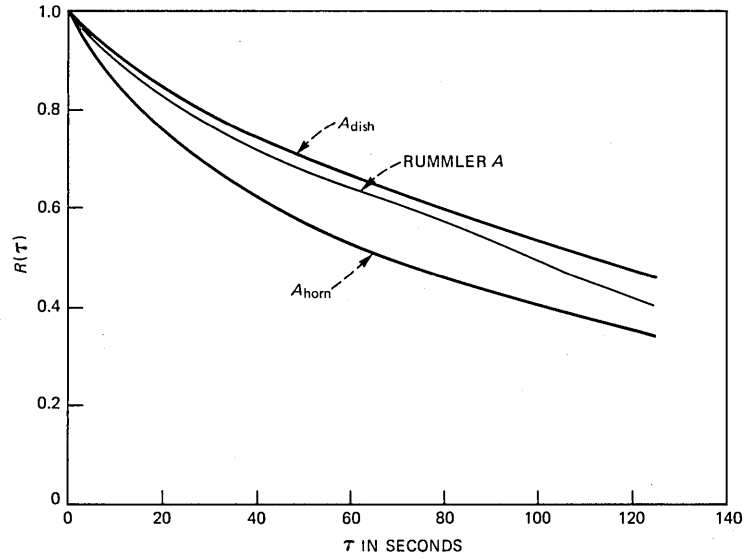


Fig. 21—Comparison of desired Rummler ACF with measured ACF for A_{horn} and A_{dish}.

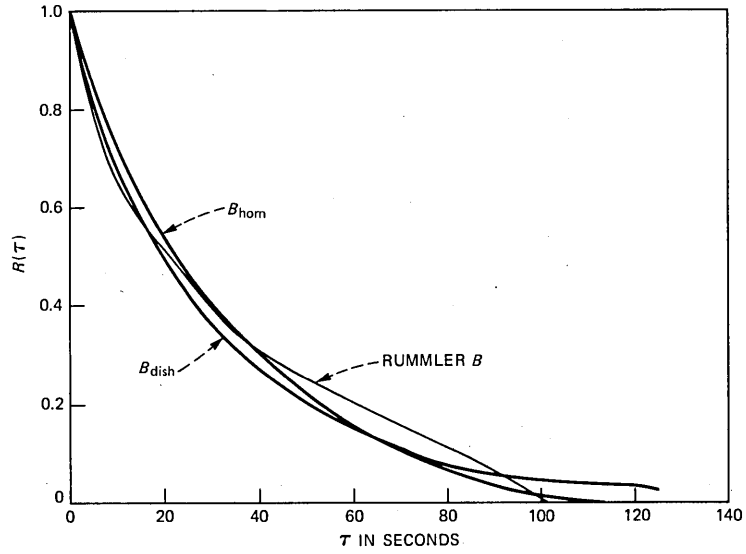


Fig. 22—Comparison of desired Rummler ACF with measured ACF for B_{horn} and B_{dish}.

accuracy. For reasonable PDFs, our iterative procedure yields a random variation whose ACF is close to the desired ACF when the latter is applied to the underlining Gaussian variation. The B and ϕ parameters showed little variation or no variation after the nonlinear func-

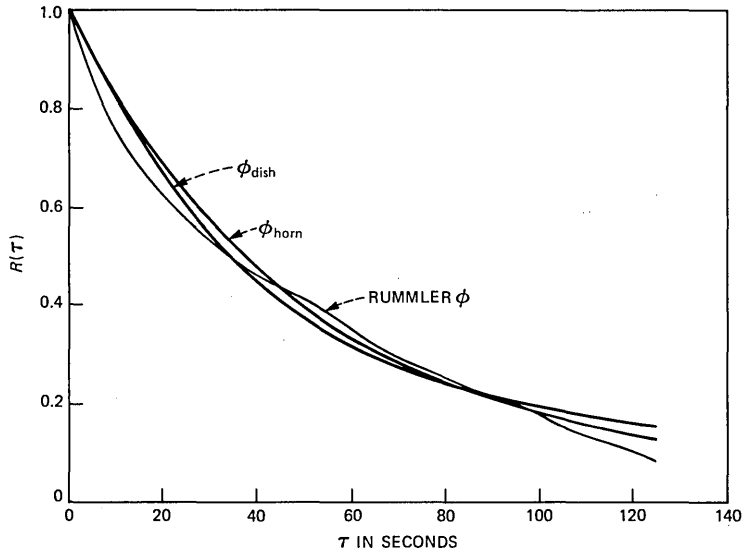


Fig. 23—Comparison of desired Rummler ACF with measured ACF for ϕ_{horn} and ϕ_{dish}.

tions were applied. The A parameter (which does not use the iterative generation method) showed a modest variation due to the addition of the mean which is related to the B parameter. Finally, the PDFs of the generated parameters have been shown to agree with the PDFs specified by the model.

6.4 Hardware accuracy: matching the intended responses

In this subsection, we assess the accuracy with which the simulation hardware produces the responses called for by the software. We begin by discussing appropriate error measures and end with experimental results.

6.4.1 General approach

Consider any one of the variable channel networks, i.e., the Horn Channel or Dish Channel of a given DCU. The input control variables (A, C, and θ) to this network at any instant are intended to produce a short-term frequency response [see (2), (4), and (5)] given by

$$H(f) = 10^{A/20} + 10^{C/20} e^{j\theta} e^{-j\omega\tau}. \tag{10}$$

As the control variables change slowly, $H(f)$ is meant to change slowly, too, in accordance with this equation.

Now suppose that, for given A, C, and θ, we measure the *actual* response of the network, $H_0(f)$, and compute some index of its depar-

ture from the *intended* response, (10), over some bandwidth. For example, we could compute

$$\epsilon = \overline{|H(f) - H_0(f)|^2} / \overline{|H(f)|^2}, \tag{11}$$

where the overbar denotes a frequency integration from, say, −20 MHz to +20 MHz. Next, suppose that ϵ were computed at fixed intervals in time as A, C, and θ moved slowly through their computer-generated variations. We could then obtain a population of ϵ-values and compute a cumulative distribution, $P(\epsilon)$. If the joint variations of A, C and θ for this experiment were representative of a particular statistical fading model, we could say that $P(\epsilon)$ is the error-measure distribution, over the *fade response ensemble* of that model, of the variable network. Finally, a set of such distributions for all four variable networks could be used to characterize—for that particular model—the accuracy of the simulator hardware.

6.4.2 Specific error measures

We devised an experimental procedure similar to the one just described, and used it to evaluate each of the variable channel networks of the CSF. In so doing, we used the dual diversity channel model of Rummler[7] to derive the fade parameter variations. At the same time, we made two important changes in the error measure, which we now discuss.

First, we chose, for the sake of simplicity, to avoid measuring the complex response $H_0(f)$; instead, we measured its squared magnitude (amplitude response) only. This function can be usefully compared with the squared magnitude of $H(f)$, (10), by means of the following error measure:

$$\epsilon(A, C, \theta) = \overline{[(G(f) - G_0(f))/G(f)]^2} \,|\, \{A, C, \theta\}, \tag{12}*$$

where

$$G(f) \triangleq |H(f)|^2; \qquad G_0(f) \triangleq |H_0(f)|^2, \tag{13}$$

and the frequency averaging is over the specified bandwidth. Given the uncomplicated nature of the frequency-selective networks, we assert that little is lost by omitting phase response data from the error measures; amplitude response accuracy alone is a reliable indicator of how well $H_0(f)$ matches $H(f)$.

It can be shown that the above error measure can be used to approximate quite closely the *root mean square decibel error* between

* We show the control variables A, C and θ here to make their pertinence explicit. Henceforth, we shall suppress them.

$G(f)$ and $G_0(f)$, as averaged over the specified bandwidth. Specifically, this quantity is approximated by $(4.34\sqrt{\epsilon})$ dB for all $\epsilon \leq 0.2$.

Our second change was to separate the error measure into a *scale error* and a *shape error*.* This approach acknowledges the fact that a small *scale* difference between $G(f)$ and $G_0(f)$ (i.e., a small *level* difference in their decibel variations) would be relatively unimportant given the vagaries of transmitter power, free space path loss, receiver noise figure, and other link power quantities. We estimate that pure level differences lying within ±1 dB would be quite acceptable in view of these factors.

Accordingly, we modify ϵ in (12) so as to permit a scaling of $G_0(f)$ by an "optimal" factor, r_{opt}, to be defined shortly. The value of ϵ with this scaling incorporated is then a measure solely of the *shape* difference between the intended and measured responses. The mathematics follows.

We define ϵ to be

$$\epsilon = \min_r \{\overline{[(G(f) - rG_0(f))/G(f)]^2}\}. \tag{14}$$

The minimizing r, which we define to be optimal and call r_{opt}, is easily found to be

$$r_{opt} = \frac{\overline{(G_0(f)/G(f))}}{\overline{(G_0(f)/G(f))^2}} \tag{15}$$

and the resulting ϵ is

$$\epsilon = 1 - \frac{\overline{[(G_0(f)/G(f))]^2}}{\overline{(G_0(f)/G(f))^2}}. \tag{16}$$

For a given A, C and θ, the quantity

$$R = 10 \log_{10}(r_{opt}) \tag{17}$$

is the *scale error* of the variables network, in decibels, for these control variables; and the quantity

$$\sigma \cong (10/\ell n\ 10)\sqrt{\epsilon} \tag{18}$$

estimates the root-mean square decibel shape error (or just *shape error*) of the variable network for these control variables. A conservative accuracy requirement is that these quantities lie within ±1 dB and below 0.5 dB, respectively, over all likely combinations of the control variables.

---

* If the power responses $G(f)$ and $G_0(f)$ are replaced by their decibel equivalents, this separation is akin to isolating the "dc" and "ac" error variations with respect to f.

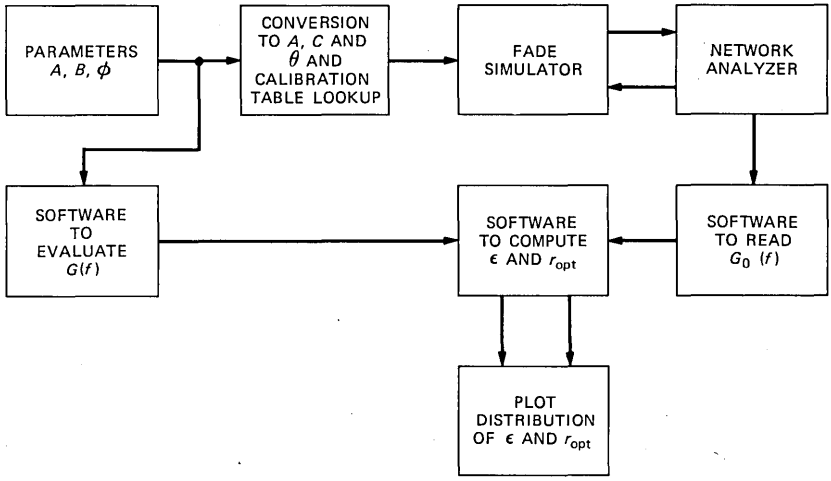


Fig. 24—Block diagram of error measurement.

6.4.3 Results

The error measures defined above were obtained for each of the four variable channel networks of the CSF. The results can be regarded as more than a check on the hardware alone. They also reflect on the accuracy of the software translations into electrical signals, the lookup routines and tables, and the software that generates those tables.

The hardware and software test configuration is shown in Fig. 24. The control computer first calculates the magnitude of the desired responses $G(f)$, from the down loaded files containing A, B, and ϕ parameters. These files are also used to generate the binary A, C, and θ values for the D/A converters by transformations and lookup tables as previously discussed. The D/A converters are loaded with the binary values and after a delay to allow the lowpass filters on the D/A control lines to settle, the response of the channel, $G_0(f)$, is sampled at 40 frequencies spaced at 1-MHz intervals over the specified bandwidth. These frequency samples are compared to the calculated samples, and error measures are computed and saved on a disk. Specifically, for each time sample of $G(f)$ and $G_0(f)$, we computed ϵ and r_{opt} using (16) and (15). In doing so, we used numerical-sum approximations to the continuous-frequency integrations called for in these equations. Given the smooth frequency responses produced by the variable channel networks, we found the 1-MHz spacings between measured values of G and G_0 to be quite adequate.

The final step in the procedure was to obtain, for each network, a cumulative distribution from the 7200 values of ϵ, and another for r_{opt}. Results for r_{opt} are given in Fig. 25 for the Horn and Dish Channels

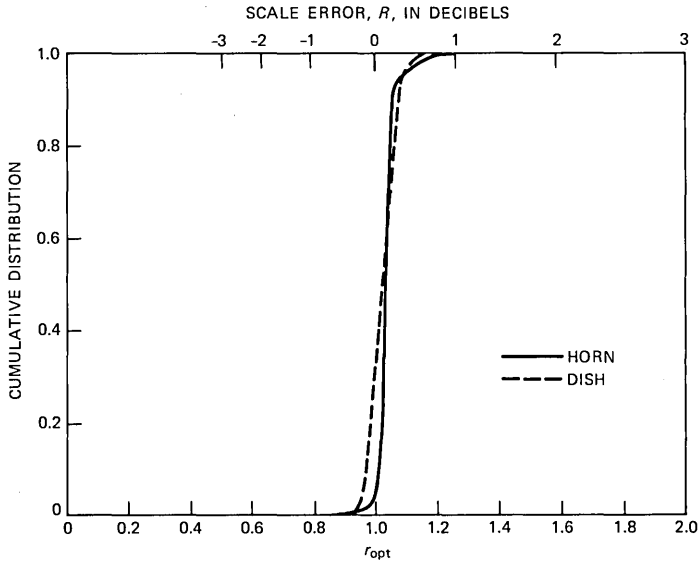


Fig. 25—Cumulative distributions of scale error, R, for each channel of a dual channel unit.

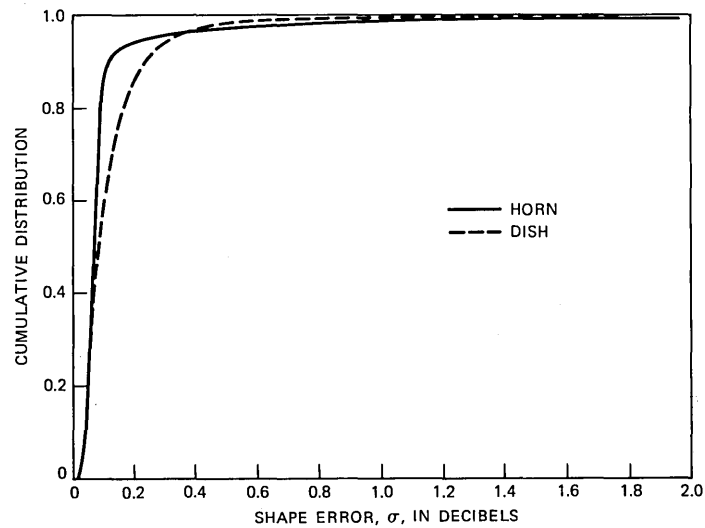


Fig. 26—Cumulative distributions of shape error, σ, for each channel of a dual channel unit.

of one of the DCUs; the corresponding results for ϵ [converted to σ via (18)] are given in Fig. 26. Very similar results were obtained for the other DCU. There are small but discernible differences between the distributions for the Horn and Dish Channels. Mostly, these are due

to the slightly different multipath fading histories generated for these two channels by the software.

The important findings from Figs. 25 and 26 are that the scale error (R) lies well within ±1 dB over all fades and that the root-mean-square decibel error (or shape error, σ) lies below 0.5 dB for over 98 percent of all fades. This is true for all four networks of the CSF and is the result we wanted. The accuracy of the hardware is thus confirmed.

VII. CONCLUSION

We have described a CSF capable of providing the radio system designer with means of measuring and comparing performance of new or existing radio hardware over multipath channels without field installation on a real path. The ability to replay a fade ensemble repeatedly in a test instead of waiting for the perversity of nature makes this simulation technique very attractive.

VIII. ACKNOWLEDGMENT

The authors want to thank M. F. Wazowicz for his help in hardware assembly and E. C. Cox, P. Knoetgen and V. R. Dillard for their help in the development of control software. We also want to especially thank L. J. Greenstein and Y. S. Yeh for their generous advice and support in both design and documentation of the simulation facility.

REFERENCES

1. W. T. Barnett, "Multipath Propagation at 4, 6 and 11 GHz," B.S.T.J., *51,* No. 2 (February 1972), pp. 321–61.
2. W. D. Rummler, "A New Selective Fading Model: Application to Propagation Data," B.S.T.J., *58,* No. 5 (May–June 1979), pp. 1037–71.
3. L. J. Greenstein and B. A. Czekaj, "A Polynomial Model for Multipath Fading Channel Responses," B.S.T.J., *59,* No. 9 (September 1980), pp. 1197–225.
4. W. D. Rummler, "More on the Multipath Fading Channel Model," IEEE Trans. Commun., *COM-29,* No. 3 (March 1981), pp. 346–52.
5. M. Liniger, "Sweep Measurements of the Transferfunction of an RF-Channel and Their Representation by Polynomials," Intl. Conf. Commun., Philadelphia, Penn., June 13–17, 1982, paper 7B3.
6. L. Martin, "Statistical Results on Selective Fading," Intl. Conf. Commun., Philadelphia, Penn., June 13–7, 1982, paper 7B5.
7. W. D. Rummler, "A Statistical Model of Multipath Fading on a Space Diversity Radio Channel," B.S.T.J., *61,* No. 9, Part 1 (November 1982), pp. 2185–219.
8. W. D. Rummler, "A Rationalized Model for Space and Frequency Diversity Line-of-Sight Radio Channels," Int. Conf. Commun., June 19–22, 1983, Boston, Mass., paper E27.
9. K. T. Wu, "Measured Statistics on Multipath Dispersion of Cross Polarization Interference," Intl. Conf. Commun., May 14–7, 1984, Amsterdam, paper 46.3.
10. M. H. Meyers, "Multipath Fading Characteristics of Broadband Radio Channels," GLOBECOM '84, Atlanta, Ga., November 26–29, 1984, paper 45.1.
11. M. Liniger, "Sweep Measurements of Multipath Effects on Cross-Polarized RF-Channels Including Space Diversity," GLOBECOM '84, Atlanta, Ga., November 26–9, 2984, paper 45.
12. L. J. Greenstein and V. K. Prabhu, "Analysis of Multipath Outage With Applications

to 90 Mbit/s PSK Systems at 6 and 11 GHz," IEEE Trans. Commun., *COM-27*, No. 1 (January 1979), pp. 68–75.
13. W. C. Jakes, Jr., "An Approximate Method to Estimate an Upper Bound on the Effect of Multipath Delay Distortion on Digital Transmission," IEEE Trans. Commun., *COM-27*, No. 1 (January 1979), pp. 76–81.
14. R. P. Coutts and J. C. Campbell, "Mean Square Error Analysis of QAM Digital Radio Systems Subject to Frequency Selective Fading," A.T.R., *16*, No. 1, 1982, pp. 23–38.
15. L. J. Greenstein and B. A. Czekaj-Augun, "Performance Comparisons Among Digital Radio Techniques Subjected to Multipath Fading," IEEE Trans. Commun., *COM-30*, No. 5 (May 1982), pp. 1184–97.
16. G. J. Foschini and J. Salz, "Digital Communications Over Fading Radio Channels," B.S.T.J., *62*, No. 2, Part 1 (February 1983), pp. 429–56.
17. D. P. Taylor and M. Shafi, "Fade Margin and Outage Computation of 4ϕ-QPRS Radio Employing Decision Feedback Equalization," Intl. Conf. Commun., Boston, Mass., June 19–22, 1983, paper F2.1.
18. O. Andrisano, "The Combined Effects of Noise and Multipath Propagation in Multilevel PSK Radio Links," IEEE Trans. Commun., *COM-32*, No. 4 (April 1984), pp. 411–8.
19. N. Amitay and L. J. Greenstein, "Multipath Outage Performance of Digital Radio Receivers Using Finite-Tap Adaptive Equalizers," IEEE Trans. Commun., *COM-32*, No. 5 (May 1984), pp. 597.
20. W. C. Wong and L. J. Greenstein, "Multipath Fading Models and Adaptive Equalizers in Microwave Digital Radio," IEEE Trans. Commun., *COM-32*, No. 8 (August 1984), pp. 928–34.
21. N. Amitay and J. Salz, "Linear Equalization Theory in Digital Data Transmission Over Dually Polarized Fading Radio Channels," AT&T Bell Lab. Tech. J., *63*, No. 10, Part 1 (December 1984), pp. 2215–59.
22. L. J. Greenstein and Y. S. Yeh, "A Simulation Study of Space Diversity and Adaptive Equalization in Microwave Digital Radio," AT&T Tech. J., *64*, No. 4 (April 1985), pp. 885–905.
23. M. Emshwiller, "Characterization of the Performance of PSK Digital Radio Transmission in the Presence of Multipath Fading," ICC78, June 1978, Paper 47.3.
24. C. W. Lundgren and W. D. Rummler, "Digital Radio Outage Due to Selective Fading-Observation vs. Prediction from Laboratory Simulation" B.S.T.J., *58*, No. 5 (May–June 1979), pp. 1073–1100.
25. Y. Y. Wang, "Simulation and Measured Performance of a Space Diversity Combiner for 6 GHz Digital Radio," IEEE Trans. Commun., *COM-27*, No. 12 (December 1979), pp. 1896–1907.
26. S. T. Matsuura, "Estimated Performance of a QPR Digital Microwave Radio in the Presence of Frequency Selective Fading," Intl. Conf. Commun., Philadelphia, Penn., June 13–7, 1982, paper 7B2.
27. J. C. Campbell and R. P. Coutts, "Outage Prediction of Digital Radio Systems," Electron. Lett., *18*, No. 25/26, December 1982, pp. 1071–2.
28. A. L. Martin, R. P. Coutts and J. C. Campbell, "Results of a 16 QAM 140 Mbit/s Digital Radio Field Experiment," Intl. Conf. Commun., Boston, Mass., June 19–22, 1983, paper F2.2.
29. C. P. Bates and M. A Skinner, "Impact of Technology on High Capacity Digital Radio Systems," Intl. Conf. Commun., Boston, MA, June 19–22, 1983, paper F2.3.
30. S. Barber, "Cofrequency Cross-Polarized Operation of a 91 Mb/s Digital Radio," IEEE Trans. Commun., Vol. COM-32, No. 1, January 1984, pp. 87–91.
31. M. H. Meyers, "Multipath Fading Outage Estimates Incorporating Path and Equipment Characteristics," GLOBECOM '84, Atlanta, Ga., November 26–9, 1984, paper 45.2.
32. M. M. Sondhi, "Random Processes With Specified Spectral Density and First Order Probability Density," B.S.T.J., *62*, No. 3 (March 1983), pp. 679–702.
33. G. Vannucci and M. C. Teich, "Computer Simulation of Superposed Coherent and Cohaotic Radiation," Applied Optics, *19*, No. 4 (February 15, 1980), pp. 548–53.
34. W. D. Rummler, "Advances in Multipath Channel Modeling," Intl. Conf. Commun., Seattle, Wash., June 8–12, 1980, 52.3.1.

AUTHORS

Harold H. Hoffman, New York University; AT&T Bell Laboratories. Mr. Hoffman, a member of the Radio Systems Research Department, has worked

on microwave radio relay systems, cordless telephone, satellite orientation, mobile radio, millimeter wave propagation, and a 7-meter radio astronomy receiver and antenna. He holds patents for IF amplifier design and signal processing. He is presently concerned with digital radio fading simulators and mobile radio propagation studies.

R. S. Roman, AT&T Bell Laboratories, 1970—. Mr. Roman, a member of the Radio Systems Research Department, has been concerned with digital radio system studies. He is currently a student at Brookdale Community College.

A. J. Rustako, Jr., B.S.E.E. 1965, M.S.E.E. 1969, New Jersey Institute of Technology; AT&T Bell Laboratories, 1957—. Mr. Rustako, a member of the Radio Systems Research Department, has been engaged in radio system and propagation studies. He contributed to early work in UHF and microwave mobile radio communications, in particular, propagation measurements of the multipath medium, and the use of diversity techniques. He has also made contributions in the field of satellite communications. These include earth-space propagation measurements of rain attenuation and depolarization at 12 GHz, and the use of phased array techniques for rapid scanning spot beam satellite antennas. He is presently concerned with high-speed digital radio transmission techniques.

Clark B. Woodworth, B.S.E.E., 1977, Monmouth College; M.S.E.E., 1980, Rutgers University; AT&T Bell Laboratories, 1977—. Mr. Woodworth worked in the Satellite Systems Research Department and the Radio Systems Research Department. Work in the latter department included digital radio studies and multipath channel simulation. He is currently working in the Network Systems Research Department. Member, Eta Kappa Nu, Sigma Pi Sigma, IEEE.

AT&T Technical Journal
Vol. 64, No. 10, December 1985
Printed in U.S.A.

Single-Frame Vowel Recognition Using Vector Quantization With Several Distance Measures

By L. R. RABINER and F. K. SOONG*

(Manuscript received June 18, 1985)

One of the most fundamental concepts used in the standard pattern recognition model for speech recognition is that of distance between pairs of frames of speech. Several distance measures have been proposed and studied in the context of an overall speech recognizer. The purpose of this investigation was to isolate the effects of different distance measures in a recognizer from the other types of processing typically used in recognition. The way in which this isolation was achieved was to use a recognizer based on single-frame distance scores, using a vector quantization approach to give the single-frame reference patterns required by the recognizer. The vocabulary for recognition was the set of continuant vowels extracted from carrier words. A speaker-dependent vowel recognition experiment was carried out using seven talkers (four male, three female) and five distance measures. Results indicated that there were differences in performance for the different distance measures when the number of code-book patterns per vowel was one or two; however, when the number of code-book patterns was four or more, these differences in performance became insignificant.

I. INTRODUCTION

In the past several years, interest has focused on defining and studying distance measures for speech recognition that reflect meaningful differences between pairs of speech spectra.[1-7] Although several different distance measures have been proposed,[1-4] and they have been studied in a variety of recognition systems,[5-7] as yet there is little

* Authors are employees of AT&T Bell Laboratories.

consistency in the reported performance of different recognizers using different distance measures. For example, although Shikano and Sugiyama[3] found consistent recognition performance improvements using the weighted likelihood ratio distance measure (as opposed to an unweighted likelihood ratio distance measure) for a Japanese speech recognition system, Nocerino et al.[7] were not able to match these results in English alpha-digit recognition experiments. Similarly, although Davis and Mermelstein[6] achieved the best performance among several distance measures with a mel-based cepstral distance measure, this result has not been confirmed in other recognition tests.

There are several possible explanations for the discrepancies in results obtained in the various investigations of distance measure performance cited above. One explanation is that the basic feature measurements of each of the recognizers were different in all cases, for example, filter bank analysis versus Linear Predictive Coding (LPC), different recording conditions and bandwidths, etc. These differences in recognizer front ends could account for the differences in performance, but if this were the case then the robustness of the distance measure would become a major issue. A second explanation is the difference in vocabulary, talkers, and transmission conditions (e.g., telephone line versus microphone input). Again these differences could be important, but they should not be factors for a robust distance measure. A third explanation is that the experimental results did not just reflect differences in distance measures but were affected by the interaction between the components of the recognizer and the distance measure. Thus, for example, improved performance for a distance measure might be overshadowed by the power of dynamic time warping, which could compensate for a distance measure with poorer performance.

It is the purpose of this paper to investigate the last possibility discussed above—namely to isolate the effects of different distance measures from all the other temporal alignment processing used in recognition. The way in which we accomplish this goal is to design a recognizer that makes its decisions based on single frames of speech. In this manner any real differences in distance measures will manifest themselves as differences in recognition scores.

The implication of using single-frame distance scores for recognition is that the only vocabulary that can be considered is the set of continuant (steady) vowel sounds. We have considered ten such vowel sounds and they are listed in Table I, along with carrier words in which the vowels occur. One side benefit of the experiments to be reported here is that a range of performance scores for single-frame recognition of vowel sounds will be established and can be used to assess future recognition algorithms in much the same way as digit

Table I—List of vowel sounds and typical carrier words

Vowel	Carrier Word
ee	beet
I	bit
e	bet
ae	bat
a	father
uh	butt
ow	bought
oo	boot
er	Bert
U	foot

and alphabet scores have become standardized for isolated word recognition.[8]

Based on the above discussion a series of speaker-dependent recognition tests was performed in the following manner. Each of seven talkers (four male, three female) spoke the carrier words in Table I ten times each, in two separate recording sessions, over a dialed-up telephone line. Each talker also created, in a separate recording session, a single robust pattern for each of the ten carrier words. For diagnostic purposes, an isolated word recognition test was performed on the 100 isolated word tokens for each talker. All words which were misrecognized were manually checked to make sure that no recording errors (by either the talker or the automatic recording system) were made.

The way in which the vowel frames, of each of the ten recordings of each carrier word, were selected was as follows. The energy contour of the word was measured, and the vowel portion was defined as the set of frames whose log energies were contiguous to and within 6 dB of the global energy peak of the word. The first five replications of each carrier word were used as a training set, and a series of LPC Vector Quantization (VQ) code books were designed from the vowel frames for each vowel and for each talker. The second five replications were used as an independent test set for recognition purposes.

Five distance measures were used in the evaluations, namely the likelihood ratio;[1,2] the weighted likelihood ratio;[3] the cepstral distance;[5] a weighted cepstral distance;[9] and a bandpass liftered, weighted cepstral distance.[10]

The overall results of the single-frame recognition tests show that for speaker-trained code books with moderate size—that is, either four or eight vectors per vowel—there were no significant differences in performance for the five distance measures. For code books with one or two vectors per vowel, the two weighted cepstral distances per-

formed best; the likelihood ratio was third; the (unweighted) cepstral distance was fourth; the weighted likelihood ratio was last.

The outline of this paper is as follows. In Section II we discuss the speech analysis system, show how we extracted the vowel frames from each carrier word, review the process of creating VQ code books, and present the five distance measures used in our experiments. In Section III we summarize the experimental conditions and present the results of the word recognition tests, the code-book design, and the single-frame recognition tests. In Section IV we discuss the results and give general conclusions.

II. SINGLE-FRAME, VQ-BASED RECOGNITION SYSTEM

A block diagram of the single-frame, VQ-based recognition system is given in Fig. 1. For each vowel frame, an LPC analysis is performed to give either an LPC vector or an LPC derived cepstral vector. We denote the resulting vector as a. This vector is then passed to a series of ten vector quantizers (VQ's), one for each of the ten vowels, and the minimum VQ distance, ϵ^i, from the VQ for the ith vowel is computed as

$$\epsilon^i = \min_{1 \le m \le M} [d(\mathbf{a}, \mathbf{b}_m^i)], \tag{1}$$

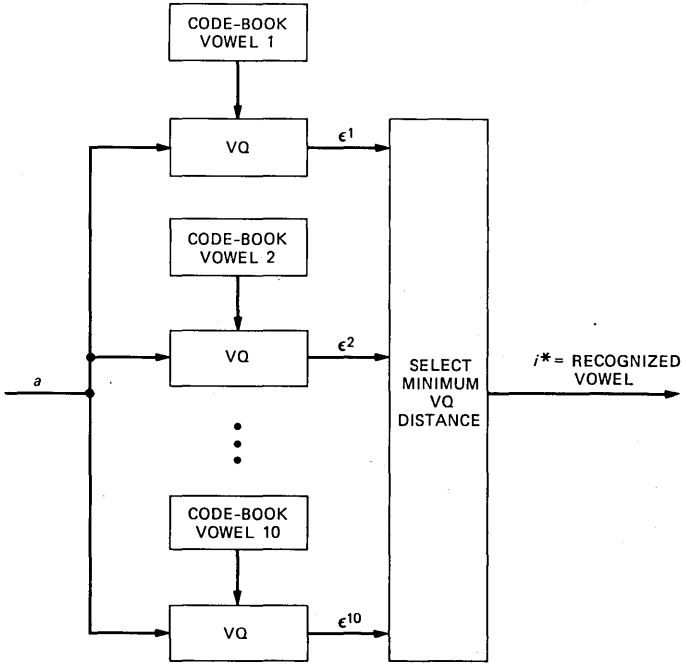


Fig. 1—VQ-based single-frame vowel recognizer.

where we assume that the ith vowel code book consists of the set of M vectors $\mathbf{b}_m^i$, $1 \leq m \leq M$. The local distance measure of eq. (1) can be any of five measures, namely the likelihood ratio; the weighted likelihood ratio; the (unweighted) cepstral distance; the weighted cepstral distance; and the bandpass liftered, weighted cepstral distance. The recognized vowel, i^*, is chosen as the one whose VQ distance ϵ^{i^*} is minimum, that is,

$$i^* = \underset{1 \leq i \leq 10}{\operatorname{argmin}} [\epsilon^i]. \tag{2}$$

In the following sections we briefly review the LPC analysis conditions, the method of extraction of vowel frames from carrier words, the procedure for VQ code-book formation, and the five distance measures used in this study.

2.1 LPC analysis conditions

The speech signal, $s(n)$, was recorded off a dialed-up, local, telephone line. We used a sampling rate of 6.67 kHz. The speech signal is digitized and then preemphasized using a first-order digital network with transfer function $H(z) = 1 - 0.95z^{-1}$. The signal is then blocked into frames of size $N = 300$ samples (45 ms), with consecutive frames spaced by L samples (15 ms). A Hamming window is applied to each speech frame and an eighth-order ($p = 8$) autocorrelation analysis is performed. The zeroth-order autocorrelation is the energy for the frame, and it is used as the basis for word detection[11] and energy normalization. An eighth-order LPC analysis is done on each frame, using the autocorrelation method of LPC,[12] to give the LPC vector for that frame. If a cepstral representation is required, a simple transformation of the LPC vector is performed.[12]

2.2 Extraction of vowel regions from carrier words

The way in which the vowel frames were extracted from the isolated word tokens is illustrated in Fig. 2. Basically we used the log energy contour of the word to find the vowel region—which was arbitrarily defined as the set of frames—in the vicinity of the maximum energy vowel frame, such that the log energy of each frame was within E_{DIF} (dB) of the vowel maximum energy, $E_{\max}$. After some preliminary experimentation, a value of $E_{\text{DIF}} = 6$ dB was used. Thus, for a typical carrier word, as illustrated in Fig. 2a, we first located the frame of maximum energy, t_v, and then, by searching in the local region around t_v, found the beginning, t_B, and ending, t_E, frames of the vowel. Although this procedure worked well, in general, there were some specific cases in which it failed. One such example is illustrated in Fig. 2b, in which the carrier word had a stop release at the end (e.g., boot) whose energy exceeded the maximum vowel energy. The simple strat-

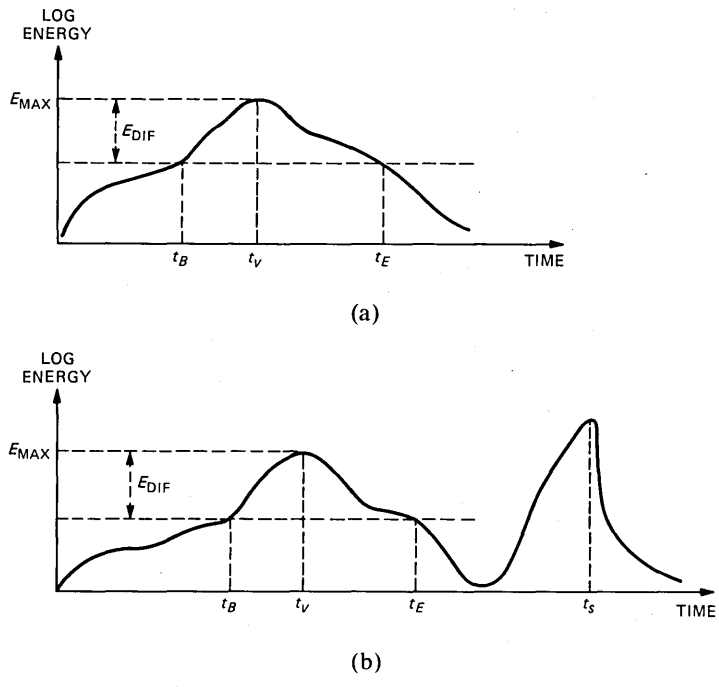


Fig. 2—Illustrations of how vowel frames were extracted from the carrier word.

egy of finding the frame with the maximum energy across the word would fail in this case. Hence a check was made to ensure that all strong local maxima of the energy contour were found, and that the correct vowel maximum was located.

2.3 VQ code-book design

The code-book training set for each vowel (and for each talker) consisted of all the "vowel frames" that occurred in five occurrences of each carrier word. In fact, there were between 35 and 90 training vectors for each vowel. From these training vectors a series of VQ code books were designed with 1, 2, 4, and 8 vectors per vowel, using a standard VQ code-book design algorithm.[13,14] The distance measure used in the code-book design was the same one used in the single-frame recognizer—that is, each of the five distance metrics was used. The centroid of the vectors in each cluster was chosen to represent the whole cluster. In our VQ design algorithm the centroid was chosen to minimize the average distortion of the whole cluster.[13]

2.4 Distance measures used in the recognizer

The five distance measures used in the recognizer included the following:

1. Likelihood ratio distance—$d_{\mathrm{LR}}(\mathbf{a}, \mathbf{b})$
2. Weighted likelihood ratio distance—$d_{\mathrm{WLR}}(\mathbf{a}, \mathbf{b})$
3. (Unweighted) cepstral distance—$d_{\mathrm{CEP}}(\mathbf{a}, \mathbf{b})$
4. Weighted cepstral distance—$d_{\mathrm{WCEP}}(\mathbf{a}, \mathbf{b})$
5. Bandpass liftered, weighted cepstral distance—$d_{\mathrm{BPCEP}}(\mathbf{a}, \mathbf{b})$.

The form for computation of the likelihood ratio is

$$d_{\mathrm{LR}}(\mathbf{a}, \mathbf{b}) = 2 \sum_{i=1}^{p} R_b(i)\hat{R}_{x_a}(i) + R_b(0)\hat{R}_{x_a}(0) - 1, \tag{3}$$

where **a** and **b** are the LPC vectors being compared, and

$$R_b(i) = \sum_{j=0}^{p-i} b(j)b(j+i), \qquad 0 \le i \le p \tag{4}$$

$$R_{x_a}(i) = \sum_{n=0}^{N-1-i} x_a(n)x_a(n+i), \qquad 0 \le i \le p \tag{5}$$

$$\hat{R}_{x_a}(i) = \frac{R_{x_a}(i)}{\alpha}, \tag{6}$$

where α is the residual error of the LPC analysis of the frame with autocorrelation $R_{x_a}(i)$.

The form for computation of the weighted likelihood ratio[3] is

$$d_{\mathrm{WLR}}(\mathbf{a}, \mathbf{b}) = \sum_{i=1}^{q} \left[\frac{R_{x_a}(i)}{R_{x_a}(0)} - \frac{R_{x_b}(i)}{R_{x_b}(0)}\right] [c_a(i) - c_b(i)], \tag{7}$$

where $R_{x_a}(i)$ and $R_{x_b}(i)$ refer to the signal autocorrelations of the frames corresponding to vectors **a** and **b**, and $c_a(i)$ and $c_b(i)$ are the corresponding LPC-derived cepstral vectors. It should be noted that we use $q > p$ terms, in the summation of eq. (7), to approximate the infinite summation of the true weighted likelihood ratio distance. In particular we have used $q = 2p$ (16), where the "extended" autocorrelations and cepstra were derived from the so-called "maximum entropy" extension of the first $(p + 1)$ terms.[15]

The form for the (unweighted) cepstral distance is

$$d_{\mathrm{CEP}}(\mathbf{a}, \mathbf{b}) = \sum_{i=1}^{q} [c_a(i) - c_b(i)]^2, \tag{8}$$

where we have again used the cepstrum extended to $q = 2p$ terms. The form for the weighted cepstral distance[9] is

$$d_{\mathrm{WCEP}}(\mathbf{a}, \mathbf{b}) = \sum_{i=1}^{p} w_i[c_a(i) - c_b(i)]^2, \tag{9}$$

where

$$w_i = \sigma_1^2/\sigma_i^2 \tag{10}$$

and σ_i^2 is the sample variance of the ith cepstral coefficient, where the averaging is over the individual vowel sounds, that is,

$$\sigma_i^2 = \frac{\sum_{v=1}^{10} [\sigma_i^2]_v \cdot n_v}{\sum_{v=1}^{10} n_v} \tag{11}$$

with $[\sigma_i^2]_v$ being the variance of $c(i)$ over the n_v frames in the training set for vowel v. Typically the weighting function w_i increases monotonically with the index i.

Finally, the form for the bandpass liftered, weighted cepstral distance[10] is

$$d_{\text{BPCEP}}(\mathbf{a}, \mathbf{b}) = \sum_{i=1}^{q} w_i'[c_a(i) - c_b(i)]^2, \tag{12}$$

where q was set to 12, and w_i' had the form of a bandpass lifter, that is, a raised sinewave of the form

$$w_i' = 1 + 6 \sin\left(\frac{\pi i}{12}\right). \tag{13}$$

III. EXPERIMENTAL EVALUATIONS AND RESULTS

A series of recognition tests was run in which each of seven talkers (four male, three female) first created robust training tokens of each carrier word[16] and then, in separate recording sessions, spoke each carrier word ten times each. The first five such recordings were used as a training set for the VQ code books; the second five recordings were used as an independent test set. The robust tokens were used in an isolated word recognition test to check the validity of the recorded carrier words. The results of the isolated word recognition test are given in Table II. It can be seen that for three of the talkers (1, 4, and

Table II—Word recognition errors for carrier words for each talker (100 recognition trials per talker)

	Talker						
Word	1(M)	2(M)	3(M)	4(F)	5(M)	6(F)	7(F)
beet							
bit			6		2		
bet		2	1		3		
bat			2		1		1
father							
butt			1		1		1
bought					1		
boot							
Bert							
foot							
TOTALS	0	2	10	0	8	0	2

6) there were no word errors; for talkers 2 and 7 there were 2 word errors (out of 100 trials each); for talkers 3 and 5 there were 10 and 8 word errors. The overall isolated word recognition accuracy for the seven talkers is 96.9 percent. The word "bit," which accounted for 8 of the 22 recognition errors, was confused with the word "bet" in all such cases.

The results given in Table II indicate that there is a lot of variability in the recognition performance on the isolated words across both talkers and vocabulary words.

3.1 Single-frame vowel recognition results

The results of the single-frame vowel recognition tests are given in Table III and are shown plotted in Fig. 3. The data in Table III are the average vowel error rates in percent averaged over the ten vowels and the seven talkers as a function of VQ code-book size and distance measure for both the training and testing sets, that is, there were about 4000 recognitions per set. Figure 3 shows these same data in graphical form. Several observations can be made from these results, including the following:

1. There are significant degradations in performance, for all distance measures and for all code-book sizes, between the training and testing sets of data. Thus for the VQ code-book size of one we see degradations of 3 to 4 percent, whereas for the VQ code-book size of eight we see degradations of from 9 to 10 percent in averaged vowel error rates.

2. The effects of different distance measures can be seen primarily for code-book sizes of one and two vectors per vowel, in which case the two weighted cepstral distances consistently outperformed the other three metrics, and the weighted likelihood ratio consistently performed the worst of the five measures. For code-book sizes of four and eight vectors per vowel, there were no significant performance differences among the five distance measures.

3. For the independent test set there was an average vowel error

Table III—Average word error rate (%) as a function of VQ code-book size and distance measure for both the training and testing sets

Distance Measure	Results on Training Set				Results on Testing Set			
	VQ Code-Book Size				VQ Code-Book Size			
	1	2	4	8	1	2	4	8
d_{LR}	17.6	12.2	7.0	3.4	21.6	16.9	14.1	12.9
d_{WLR}	18.8	13.6	7.2	3.8	22.4	18.9	14.2	12.5
d_{CEP}	18.6	11.8	7.1	3.5	21.6	17.4	13.7	13.4
d_{WCEP}	16.5	11.0	6.7	3.8	20.0	16.5	14.2	13.3
d_{BPCEP}	16.7	10.5	6.4	3.2	19.4	15.5	13.4	12.4

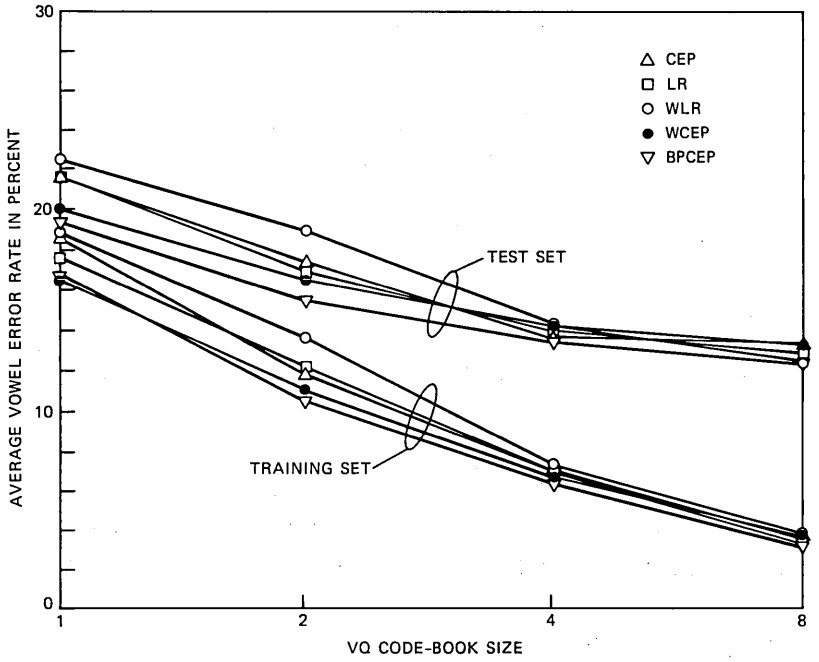


Fig. 3—Average vowel error rate (%) versus code-book size for each of the four distance measures and for the testing and training sets of data.

rate of about 20 to 22 percent for a single code-book vector per vowel, and the error rate dropped to about 13 percent for eight code-book vectors per vowel. Thus we conclude that vowel recognition (among the ten vowels in Table I) cannot be performed reliably using any of the distance measures we have considered, in the framework of a single-frame VQ code book-based recognizer.

IV. DISCUSSION AND CONCLUSIONS

The results presented in Section III can be interpreted as follows. In the case where we have a good representation of the patterns to be recognized, the effects of different distance measures on recognition performance are small. Such was the case when we used four or eight code-book vectors to characterize each vowel in the vocabulary. However, when the representation of the patterns to be recognized becomes more coarse, then the effects of different distance measures start to become important. In these cases a better characterization of speech sound differences, as obtained from a good distance measure, should give better recognition scores. Such was the case when we used one or two code-book vectors to characterize each vowel.

There is another important observation that can be made from the

results presented in Table III. We see a big difference in average vowel error rates between comparable test conditions (distance measure, VQ code-book size) for the training and testing sets, especially when we have four or eight vectors per vowel code book. Thus, in a sense, the effects of different distance measures are small when the code-book vectors begin to characterize well the seemingly insignificant details of the training set, and are larger when the code-book vectors characterize mainly the gross spectral behavior of the vowels. For real-world recognition systems it is most probably the latter case that is the more important one in that the reference patterns would be expected to characterize the gross behavior of spectral variations with time. In general there is not enough training data to reach the point where we have characterized the fine spectral variations of words reliably.

The conclusion we reach from the above discussion is that the results for small code-book sizes, in which there were significant effects of different distance measures, are probably more representative of real recognition systems than the results for large code-book sizes. In these cases—as is evidenced by recent investigations by Tokhura,[9] Juang et al.,[10] and Nocerino et al.,[7]—the weighted cepstral distances and the likelihood ratio would be expected to give better recognition performance than the unweighted cepstral distance or the weighted likelihood ratio measures.

V. SUMMARY

We have presented results on speaker-dependent, single-frame, VQ-based, vowel recognition for five different distance measures and for four different size VQ code books. Our results indicate that for small code-book sizes (one or two vectors per vowel) there is improved recognition performance using a weighted cepstral distance rather than the likelihood ratio, the unweighted cepstral distance, or the weighted likelihood ratio measures. For larger code-book sizes (four or eight vectors per vowel) the performance differences among the five distance measures decrease. For practical recognizers, the weighted cepstral distances appear to have advantages for application to speaker-independent systems and for large vocabulary recognizers. These advantages include increased efficiency of representation, reduced complexity of computation, and improved performance.

REFERENCES

1. F. Itakura and S. Saito, "Analysis-Synthesis Telephony Based on the Maximum Likelihood Method," Proc. Int. Cong. Acoustics, Tokyo, Japan, Paper C5-6, 1968.
2. F. Itakura, "Minimum Prediction Residual Principle Applied to Speech Recognition," IEEE Trans. Acoust., Speech, Signal Process., *ASSP-23,* No. 1, (February 1975), pp. 67–72.

3. K. Shikano and M. Sugiyama, "Evaluation of LPC Spectral Matching Measures for Spoken Word Recognition," Trans. IECE, J65-D, No. 5 (May 1982), pp. 535–41.
4. D. H. Klatt, "Prediction of Perceived Phonetic Distance From Critical Band Spectra: A First Step," Proc. ICASSP '82, (May 1982), pp. 1278–81.
5. A. H. Gray Jr. and J. D. Markel, "Distance Measures for Speech Processing," IEEE Trans. Acoust., Speech, Signal Process., *ASSP-24,* No. 5, (October 1976), pp. 380–91.
6. S. Davis and P. Mermelstein, "Comparison of Parametric Representations for Monosyllabic Word Recognition in Continuously Spoken Sentences," IEEE Trans. Acoust., Speech, Signal Process., *ASSP-28,* No. 4, (August 1980), pp. 357–66.
7. N. Nocerino et al., "Comparative Study of Several Distortion Measures for Speech Recognition," Speech Commun., *4,* No. 4 (November 1985).
8. G. R. Doddington and T. B. Schalk, "Speech Recognition-Turning Theory to Practice," IEEE Spectrum, *18* (September 1981), pp. 26–32.
9. Y. Tokhura, "Speaker Independent Recognition of Isolated Digits Using a Weighted Cepstral Distance," J. Acoust. Soc. Am., Suppl. 1, *77,* Paper E13 (Spring 1985), p. S11.
10. B. H. Juang, L. R. Rabiner, and J. G. Wilpon, "On the Use of Bandpass Liftering in Speech Recognition," unpublished work.
11. L. F. Lamel et al., "An Improved Endpoint Detector for Isolated Word Recognition," IEEE Trans. Acoust., Speech, Signal Process., *ASSP-29,* No. 4 (August 1981), pp. 777–85.
12. J. D. Markel and A. H. Gray Jr., *Linear Prediction of Speech,* Springer-Verlag, 1976.
13. Y. Linde, A. Buzo, and R. M. Gray, "An Algorithm for Vector Quantization," IEEE Trans. Comm., *COM-28,* No. 1 (January 1980), pp. 84–95.
14. B. H. Juang, D. Wang, and A. H. Gray, Jr., "Distortion Performance of Vector Quantization for LPC Voice Coding," IEEE Trans. Acoust., Speech, Signal Process., *ASSP-30,* No. 2 (April 1982), pp. 294–303.
15. J. P. Burg, "Maximum Entropy Spectral Analysis," PhD. Thesis, Stanford Univ., 1975.
16. L. R. Rabiner and J. G. Wilpon, "A Simplified, Robust Training Procedure for Speaker Trained, Isolated Word Recognition Systems," J. Acoust. Soc. Am., *68,* No. 5 (November 1980), pp. 1271–6.

AUTHORS

Lawrence R. Rabiner, S.B. and S.M., 1964, Ph.D., 1967 (Electrical Engineering), The Massachusetts Institute of Technology; AT&T Bell Laboratories, 1962—. From 1962 through 1964 Mr. Rabiner participated in the cooperative plan in electrical engineering at AT&T Bell Laboratories, in Whippany and Murray Hill, New Jersey. He worked on digital circuitry, military communications problems, and problems in binaural hearing. Presently, Mr. Rabiner 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), *Digital Processing of Speech Signals* (Prentice Hall, 1978), and *Multirate Digital Signal Processing* (Prentice-Hall, 1983), Member, National Academy of Engineering, Eta Kappa Nu, Sigma Xi, Tau Beta Pi, Fellow, Acoustical Society of America, IEEE.

Frank K. Soong, B.S., 1973, National Taiwan University, M.S., 1977, University of Rhode Island, Ph.D., 1983, Stanford University, all in Electrical Engineering; AT&T Bell Laboratories, 1982—. From 1972 to 1975 Mr. Soong served as a teacher at the Chinese Naval Engineering School at Tsoying, Taiwan. In 1982 he joined the technical staff at AT&T Bell Laboratories, where he engaged in research in speech, coding, and speaker recognition. Member, IEEE.

AT&T Technical Journal
Vol. 64, No. 10, December 1985
Printed in U.S.A.

Traffic Capabilities of Two Rearrangeably Nonblocking Photonic Switching Modules

By R. A. THOMPSON*

(Manuscript received June 10, 1985)

The architectures of two small switching networks are compared as potential implementations of a 4 × 4 photonic switching module. Such a module would be made by interconnecting several 2 × 2 photonic directional couplers on a single $LiNbO_3$ substrate. While both networks are rearrangeably nonblocking, we investigate whether one network requires significantly more rearrangements than the other. The analysis includes transient, Monte Carlo simulation, and Markov steady-state techniques. We conclude that the traffic capabilities of the two structures are not significantly different, and that selection of an architecture can be based on other criteria, like loss, crosstalk, or ease of manufacture.

I. INTRODUCTION

The percentage of the world's voice and data communications that is carried by optical fibers increases daily. The importance of research in photonic switching increases with it. A promising element for the implementation of photonic switching systems is the photonic directional coupler, made from titanium-diffused lithium niobate. The current state of this technology allows a level of integration of tens of devices on a single substrate. We investigate two competing architectures for a 4 × 4 switching module, fabricated in this technology, that could be useful in building photonic systems.

In the next two sections of the paper, we review some basic concepts

*AT&T Bell Laboratories.

of switching networks and describe the two candidate photonic switching modules. In Sections IV and V, we present the results of transient analysis and Monte Carlo simulation, respectively, as applied to the two switching modules. In Sections VI through VIII, we present steady-state Markov analysis as applied to a generalized module and to each candidate switching module, respectively. In Section IX, we enumerate the network configurations in each of the candidate switching modules.

II. SWITCHING NETWORKS

The traffic-handling performance of a switching network depends on two entities:

- The topology of the internal elementary switching stages.
- The rule by which paths through the network are selected.

2.1 Topology

One topological classification applied to switching networks is the hierarchy of "blockingness." Networks are classified by their ability to establish connections, especially sequences of connections with intermediate disconnections. An excellent tutorial and summary of the state-of-the-art in switching network topologies is found in Ref. 1.

2.1.1 Strict- and wide-sense nonblocking networks

A switching network is *nonblocking* if any desired connection between two idle ports can be completed immediately without interference from connections that may be already established in the network. If this property is independent of any switching rule used to select paths through the network, then the topology is *nonblocking in the strict sense.* If the property is true, provided the paths assigned to the established connections were selected under some switching rule, then the topology is *nonblocking in the wide sense.*

Such topologies are *serial nonblocking* in that arbitrary sequences of input-output pairs can be connected and disconnected without blocking. Neither switching module described in this paper is nonblocking in either sense.

2.1.2 Rearrangeably nonblocking and blocking networks

A switching network is *rearrangeably nonblocking* if any desired connection between two idle ports, which might be temporarily blocked by connections already established in the network, can be completed possibly after some established connections are moved to different paths. Such topologies are *parallel nonblocking* in that any set of I/O pairs can be connected without blocking if the network is initially idle and the set is known in advance.

A switching network is *blocking* if there exist network configurations of established connections from which some connection between two idle ports can not be completed, even with rearrangement of the established connections. Both switching modules under consideration in this paper are rearrangeably nonblocking.

2.2 Rule

A switching network generally allows more than one path by which ports on either end of the network may be connected. The *switching rule* is the means of selecting one path. For certain special cases, such as the strict-sense nonblocking networks, the performance is independent of the switching rule. In general, however, the switching rule is an important factor in the performance of a switching network.

The optimal switching rule follows: Route a new connection through the network in a way that least affects the routing of any future connections.

Depending on the network topology, such a switching rule may be difficult to implement because of tedious look-ahead iterations. Therefore, rules that are simple to implement and are optimal, or almost optimal, are sought. Beneš describes several switching rules:

> ... route a call through the most heavily loaded part of the network that will still take the call.
>
> Do not use a *fresh* middle switch (in a 3-stage network) unless you have to.[2]

A similar switching rule, used in the simulation program described in Section V, follows: Select the network path that minimizes the count of additional switches whose transmission state must be set.

Such switching rules are called *packing rules* because of their analogy to similar rules used in the problem of packing spheres into boxes. For one module under consideration here, the four switching rules above appear equivalent, and the rule is called *prudent* in the paper. For the other module under consideration here, a case will be demonstrated that represents a counter example to the general adoption of switching rules that recommend close packing as the primary consideration.

2.3 Performance measures

Three measures of the quality of switching networks are transmission, topology, and traffic capability.

2.3.1 Transmission

Two measures of transmission quality are insertion loss and crosstalk. Either measure may be in terms of average or worse-case value. Under a uniform wiring scheme, insertion loss and crosstalk worsen

as a network becomes deeper. In electrical implementations, crosstalk usually worsens as interchip, interboard, and especially interframe wiring increase. This should be much less noticeable in photonic systems. Loss and crosstalk measures will not necessarily recommend the same switching networks.

2.3.2 Topology

Topological complexity is not so much a performance issue as it is a manufacturing and economic issue. Quantitative measures that correlate with manufacturing cost are not known, nor is the importance of this issue relative to transmission or traffic measures. It is an important open question.

2.3.3 Traffic capability

Traffic capability is a well-known analytical measure applied to blocking networks.[2] The probability of blocking is that weighted proportion of ⟨new connection, network configurations⟩ in which the given new connection cannot be completed through the network in the given configuration.

A corresponding measure for rearrangeably nonblocking networks is developed here. The *probability of requiring rearrangement* P_{rr} is that weighted proportion of ⟨new connection, network configurations⟩ in which the given new connection can be completed through the network in the given configuration, but only after some established connections in the network are first rearranged.

A dichotomy appears. If the malevolence is the CPU real time used in effecting the rearrangement, then the demerit is that any rearrangement is required, regardless of the count of switches or connections involved, because they are probably all rearranged in parallel. If the malevolence is the count of established connections disturbed by the rearrangement, then the demerit may not be binary and one choice of rearrangement may cost less than another. We will show that the count of established connections disturbed by a rearrangement is different for the two modules under consideration, so the dichotomy is relevant. We will derive expressions for how many established connections must be rearranged, on the average, for each of the competing architectures.

2.4 Traffic analysis techniques

In transient analysis, a sequence of connections and disconnections is applied to an idle module. Over a set of such sequences, P_{rr} is the proportion of those sequences in the set in which a rearrangement was required to complete a connection. The inadequacy of this analysis is

that it does not produce a single number or expression, because many sets of sequences must be investigated. But the advantages over the classical analyses are that the results are not dependent on potentially unrealistic assumptions about traffic statistics, and that this analysis entails a level of detail (and corresponding tedium) not required in the other analyses. This detail uncovered an optimal switching rule for one module that probably would have been overlooked had the analysis been confined to traditional steady-state techniques.

In Monte Carlo simulation, a sequence of randomly generated events is applied to an initially idle module. The inadequacies of this analysis are that repeated simulations with identical event statistics yield different results and that no closed-form solution is obtained, giving P_{rr} as a function of those statistics. The benefit of this analysis is that both transient and steady-state behavior may be observed, and in fine detail. If properly recorded, the events leading up to anomalies may be studied.

In steady-state Markov analysis, the module is assumed to be in some random network configuration and then a new random connection or disconnection occurs. P_{rr} is a weighted sum over all network configurations, and all possible new connections from those network configurations, of those cases in which the module requires rearrangement to complete the given connection from the given configuration. The weighting is the steady-state probability distribution over the set of network configurations or states of equivalence classes. Both the steady-state probability distribution and the interstate transition probabilities are dependent on the traffic load. This analysis yields a closed-form result, but it is only valid in the steady state and it is dependent on potentially unrealistic statistical assumptions.

III. PHOTONIC SWITCHING MODULES

We review the technology of photonic switching and present the topologies of two proposed implementations of a 4 × 4 switching module.

3.1 Photonic directional coupler

A photonic directional coupler is a two-input two-output device whose transmission state depends on the magnitude of an applied external voltage.[3] (See Fig. 1.) With nominal 0-volt applied, the transmission is such that the signals cross over from input to output (called the *crossed* state) and with a positive voltage applied, the transmission is straight through from input to output (called the *bar* state). This device is a photonic realization of the generic 2 × 2 *beta switching element*,[4] where the control signal is electronic and the switched data is photonic.

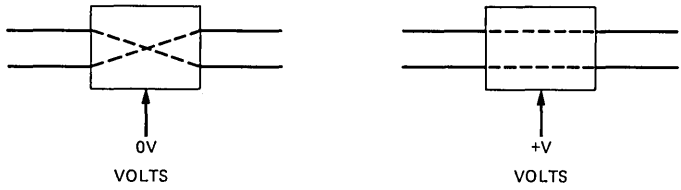


Fig. 1—Functional states of a photonic directional coupler.

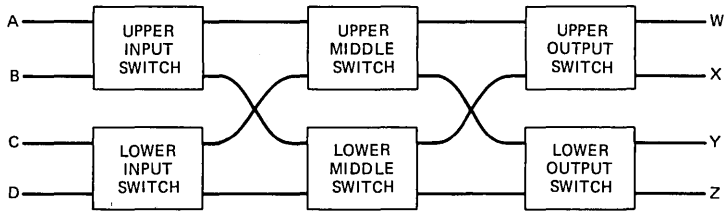


Fig. 2—Configuration and terminology of the 222 Module.

Switching speeds in the tens of picoseconds have been reported,[5] but speeds in the nanoseconds are more common.[6] The attainable switching speed is highly dependent on device packaging, on the quality of the electronic driver circuit that switches the applied voltage to the device, and on the magnitude of this required voltage. A 1:16 multiplexer/demultiplexer has been built as an integrated circuit module.[7] It is a simple tree topology of photonic directional couplers with one unused port at the input to each photonic directional coupler. Another integrated circuit is a 4×4 switching module built as a square array.[8,9] These circuits show the level of integration available today in this technology. Because each photonic directional coupler has length in the order of a centimeter, the expectation of increasing the level of integration by several orders of magnitude is not high, unless some major breakthrough changes the photonic coupling length by orders of magnitude.

Two waveguides, made by diffusing titanium in a lithium-niobate substrate, can be made to intersect or cross over. Crosstalk increases as the angle of intersection decreases, and becomes unpredictable at very small angles.[10] Thus a configuration, like that of Fig. 2, is feasible if the crossovers are truly at large angles, as the figure suggests. However, the photonic directional couplers have lengths in the order of a centimeter and widths measured in microns, so Fig. 2 is not drawn to true scale. Furthermore, the loss in a waveguide increases rapidly as its radius of curvature decreases. So, a configuration like that of

Fig. 2 would require a physically large chip to allow geometries with high crossover angles and high radii of curvature.

A major technological issue with today's photonic directional couplers is that they have significant insertion loss. While most of this loss is due to coupling between fiber and chip and would be alleviated with integrated circuits, the device loss is still high enough that multidevice systems, like large switching networks, will require signal amplification. Photonic gain can be simulated, expensively and with imposing a bit-rate constraint, by a circuit with a photodetector, an electronic amplifier, and a laser. However, direct photonic amplification is believed to be coming available soon.[11] The devices described in the reference are fabricated from gallium arsenide, while the photonic directional couplers are fabricated from lithium niobate. Thus, integration on the same chip is currently impossible. However, when such devices become practical, they could be integrated onto the same chip carrier as the chip containing the photonic directional couplers. If this form of integration is truly practical in the future, it allows small radius bends and alleviates many of the problems described in the preceding paragraph.

3.2 The 222 Module

We call one candidate 4 × 4 module the *222 Module* because it is a three-stage module, where each stage has two switches. (See Fig. 2.) This topology is in the class of networks called *Clos networks*, and this exact configuration is a frequently used example in the literature. The following terminology is used to identify symbolic inputs and outputs in the 222 Module:

- A and W represent an arbitrary input and output, respectively, each shown arbitrarily as an upper port on an upper switch.
- B represents the *other* port on the same input switch as A and X represents the *other* port on the same output switch as W.
- C represents either port on the input switch that A and B do *not* share, and Y represents either port on the output switch that W and X do *not* share, each shown arbitrarily as an upper port on a lower switch.
- D represents the *other* port on the same input switch as C, and Z represents the *other* port on the same output switch as Y.

Because of the symmetry of the 222 Module, there is no topological relationship between A and W.

3.3 The 2121 Module

We call the other candidate 4 × 4 module the *2121 Module* because the topology has four stages, where the input and middle stages have two switches and the central and output stages have one switch. (See

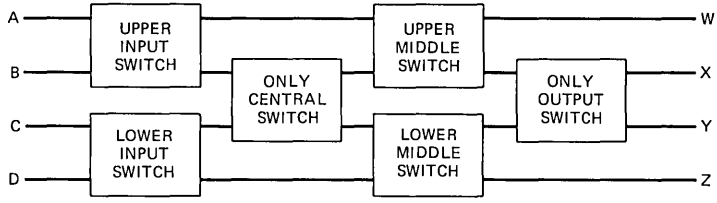


Fig. 3—Configuration and terminology of the 2121 Module.

Fig. 3.) This topology is also a special case of a general network structure.[12] Throughout the paper we will carefully distinguish the terms *middle* and *central* when applied to switching stages. The following terminology is used to identify symbolic inputs and outputs in the 2121 Module:

- A represents an arbitrary input and B represents the *other* port on the same input switch as A, both shown arbitrarily on the upper input switch with A above B.
- W represents the only output that A or B can reach through two stages (necessarily on the same horizontal level as A and B).
- C represents either port on the input switch that A and B do *not* share, and D represents the *other* port on the same input switch as C, both shown arbitrarily on the lower input switch with C above D.
- Z represents the only output that C or D can reach through two stages (necessarily on the same horizontal level as C and D).
- X represents either port on the only switch with two outputs on it, and Y represents the *other* port on the output switch with X, shown arbitrarily with X above Y.

Because the 2121 Module is not symmetric, there is a topological relationship between the inputs and the outputs. Accordingly, W or Z is on the upper or lower middle switches depending on whether A and B or C and D are on the upper or lower input switches, respectively.

3.4 A switching rule for the 2121 Module

An interesting and nonintuitive switching rule was discovered for the 2121 Module. Consider the following sequence of events applied to an idle 2121 Module. Let A to W be the first connection, and let the path that avoids the central switch be selected (intuitively, the best choice), setting the states of A's input switch appropriately, and W's middle switch to the bar state.

If the second connection is B to X, representing two of the nine second connections, the intuitively preferred path is not available because the A to W connection has already set the switches, and is using that junctor that B can reach that skips the central switch. Two

other paths are available and the choice is extremely interesting. One path shares the state of A's input switch and W's middle switch, and assigns the central switch to the bar state and the output switch, appropriately. This path requires the assignment of *two* switches in addition to the ones already assigned, and it is intuitively appealing. After A to W is disconnected, A to Y, A to Z, and C to W, representing four of the nine third connections, require rearrangement, and the other five do not.

The other path for B to X still shares the state of A's input switch, but it assigns the central switch to the *crossed* state, the unused middle switch to the bar state, and the output switch appropriately. This path requires the assignment of three switches in addition to the ones already assigned, and is, therefore, intuitively less appealing than the previous path. However, after A to W is disconnected, only A to Z, requiring some of the junctors and switch connections used by B to X, requires a rearrangement of the network configuration.

I have two general interpretations that cover the essence of this switching rule: (1) If you must use the central switch, the crossed state is preferred, and (2) Minimize the count of switches that are shared with any other path. It is not clear yet whether either or both of these statements is applicable to a generalization of the 2121 Module to an $n \times n$ architecture. Even in the 2121 Module, I was afraid that the overall use of this switching rule, while improving the performance with some sequences of events, would degrade the performance with other sequences. In all the sequences investigated and all the simulations that were run, no such case arose.

3.5 Established connections disturbed in the 2121 Module

We give an example of a transient sequence of events applied to the 2121 Module in which the final connection in the sequence requires the disturbance of two established connections. We then argue that the worst-case number could not be greater than two.

The example begins, with an idle module, by connecting A to X. The optimal path avoids the central switch by assigning W's middle switch to the crossed state and A's input switch and the output switch, appropriately. Let the second connection be from B to Y. Since two of the paths are blocked, the remaining path must be used, by assigning the central switch to the crossed state and Z's middle switch to the bar state. Now disconnect the first connection, the one from A to X, freeing the state of W's middle switch.

Let a new second connection be from C to X. Since the output switch and Z's middle switch are in the wrong states to use two of the paths, the remaining path must be used, by assigning W's middle switch to the bar state, and using the idle link through the central

switch. The states of all six switches are set, and two of four new connections will require rearrangement.

Consider a third connection from A to Z. Its only path requires the reconfiguration of the input switch shared by A and B and Z's middle switch, and also requires the link through the central switch now used by B to Y. So the B-to-Y connection must be moved to its other (optimal) path. But it can't be moved directly because W's middle switch and the output switch are in the wrong state. So, the C-to-X connection must also be moved to its other (optimal) path.

Having established, by example, that the worst-case number is at least two, the question remains whether it could be higher. Since there can only be four established connections, the only other number to consider is three. However, in any network configuration with three established connections, the unused fourth connection is always available. While this fact may not be immediately obvious, it can be proven exhaustively for the 2121 Module. A more elegant proof, applicable to the general $n \times n$ module, is desirable.

3.6 Comparing the modules

3.6.1 Blocking characteristic

The generalized Clos Network has the topology of Fig. 2, but with n inputs and m junctors on each of r rectangular input switches, with n outputs and m junctors on each of r rectangular output switches, and with m square middle switches each connecting to r junctors on each side. It is known[2] that such a switch is nonblocking if $m \geq 2n - 1$ and is rearrangeably nonblocking if $m \geq n$. The latter is satisfied in the 222 Module, in which $n = m = r = 2$.

The 2121 Module is also rearrangeably nonblocking. This result can be proven easily by exhaustion for the 2121 Module and is obvious by inspection of the state diagram in Section VIII. A general proof for a generalized $n \times n$ network is being developed.[12]

3.6.2 Symmetry and uniformity

In the 222 Module each I/O pair has exactly two connection paths. That is, there are exactly two paths through the module from any input to any output. The 2121 Module does not have this symmetry. The I/O pairs at opposite corners of the module (A to Z, B to Z, C to W, and D to W) have only one path through the module, requiring the central and the appropriate middle switches in the crossed state. The I/O pairs straight across the module (A to W, B to W, C to Z, and D to Z) have two paths through the module: one avoiding the central switch and the other using the central switch in the bar state. Any connection to an output on the only output switch (A to X, B to X, A to Y, B to Y, C to X, D to X, C to Y, and D to Y), has three paths

through the module: one avoiding the central switch, the second using the central switch in the bar state, and the third using the central switch in the crossed state.

In the 222 Module each path through the module passes through exactly three photonic directional couplers. The 2121 Module does not have this symmetry, either. In the 2121 Module some paths pass through only two photonic directional couplers, some through three, and some through four. This variation will make deterministic gain difficult. If crossovers in the 222 Module require photonic directional couplers, or have similar loss and crosstalk as photonic directional couplers, then this module would not be symmetric in this sense either.

If crossovers in the 222 Module are insignificant, its inputs and outputs are uniform. That is, all inputs and outputs have the same properties of the following: count of paths per I/O pair, count of switches per path, and equal access to any port on the other side of the module. The last two properties do not hold true if crossovers are significant in the 222 Module, but none hold true in the 2121 Module.

3.6.3 Crossovers

The interconnection topology of the 222 Module has two crossovers: one between the input and central stages and one between the central and output stages. The interconnection topology of the 2121 Module has no crossovers.

An implementation is illustrated in Fig. 4 in which the two crossovers in the 222 Module are eliminated by off-chip fibers. Its practicality depends on the difficulty of off-chip fibering, the impracticality of crossovers on the photonic substrate, and the magnitude of the advantage of the 222 Module over the 2121 Module (if any).

3.6.4 Transmission

It is difficult to predict the difference in the crosstalk characteristics

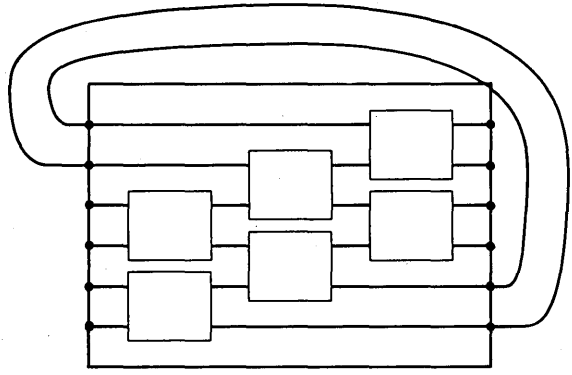


Fig. 4—Off-chip fibering to avoid crossovers in the 222 Module.

of the two modules. The crossovers in the 222 Module suggest that it may be worse than the 2121 Module, but the extra stage in the latter suggests otherwise.

Insertion loss is easier to predict. The worse-case count of switches in a network path is four in the 2121 Module and is three in the 222 Module. Thus, the 2121 Module would appear to have a poorer insertion loss characteristic. However, if crossovers must be implemented by photonic directional couplers, or if they have equivalent loss, then the worse-case count of (equivalent) switches in a network path is five in the 222 Module and it would have worse insertion loss than the 2121 Module.

3.6.5 Switching rule

The switching rule used in the 222 Module is the classical packing rule discussed in Section 2.2. The switching rule used in the 2121 Module is the unusual rule discussed in Section 3.4. Two implementations of the switching rule are discussed in Section IX and the network configurations for each module are enumerated.

3.6.6 The number of established connections disturbed

If some network configuration of the 222 Module must be rearranged before a new connection can be completed, only one established connection need ever be disturbed.[2] It was shown in Section 3.5 that there exist network configurations in the 2121 Module in which two established connections may need to be disturbed. Consequently, we not only compute P_{rr} for each module in the sections below, but we also compute, N_{rr}, how many established connections must be disturbed, on the average.

3.6.7 Familiarity

The 222 Module is intuitive, in so far as the theory of switching networks is intuitive. Its transient behavior is not surprising, except for one set of connect/disconnect sequences, as described in Section IV. The application of switching rules is logical and classical and the plethora of network configurations is easily partitioned into only 10 equivalence classes, or Markov states.

By contrast, the 2121 Module has been counter-intuitive. While its transient behavior is smooth, it degrades with simple connect/disconnect sequences. The switching rule and count of established connections disturbed has been startling to those familiar with such things. The plethora of network configurations is partitioned, with great difficulty, into 50 equivalence classes, making Markov analysis tedious.

IV. TRANSIENT ANALYSIS

In transient analysis, we assume the module is idle and study its

behavior under different sets of connect/disconnect sequences. The network quality, the average probability of requiring rearrangement for a specific set of sequences, is the proportion of those sequences in the set that require rearrangement. Four sets are applied to each of the two modules, giving eight results. Computer programs applied all sequences in the four sets to each module and tabulated whether rearrangement was required. The program's output was studied and generalized into theorems whose formal proofs are exhaustive and tedious and not presented in this paper.[13] The proofs are available to the interested reader.

4.1 Sequences of switching events

A *template* is a set of fixed-length sequences of events applied to the switching module, where an *event* is a connection or a disconnection of an input-output pair. The nomenclature for templates is $x_1 \ldots x_n$, where x_i represents the ith event in a sequence; x_i has value C or D, depending on whether this ith event is a connection or disconnection, respectively; and n is the length of a sequence.

The following rules govern the determination of significant templates:

- Since the module is assumed to be idle before any sequence is applied, no sequence would begin with a D.
- Similarly, no sequence, nor initial subsequence, would have more Ds than Cs.
- Since a module can only support four connections, no sequence, nor interior subsequence, would have four more Cs than Ds.
- No significant sequences would have, nor would contain an initial subsequence that has, as many Cs as Ds (e.g., no significant sequence would begin with CD) because it would have the effect of restarting from an idle module.
- Since we expect no problems honoring disconnects in these modules, no significant sequence would end with a D.
- In no significant sequence would a D apply to an immediately preceding C because it would have the effect that the C never occurred.

Templates with length 3 or less exhibit no anomalous behavior, and templates with length 6 or greater are too complex to examine exhaustively. Fortunately, enough templates with lengths 4 and 5 are significant. Combining these length and content constraints, the significant templates are CCCC, CCDC, CCDCC, and CCCDC.

4.1.1 CCCC sequences

For sequences in a CCCC template, beginning with all switches idle, the assumed order of events is

1. One of four inputs connects to one of four outputs.
2. One of three idle inputs connects to one of three idle outputs.
3. One of two idle inputs connects to one of two idle outputs.
4. The last idle input connects to the last idle output.

With $4 \times 4 = 16$ cases of first connection, $3 \times 3 = 9$ cases of second connection, $2 \times 2 = 4$ cases of third connection, and $1 \times 1 = 1$ case of final connection, the CCCC template contains $16 \times 9 \times 4 \times 1 = 576$ sequences.

4.1.2 CCDC(C) sequences

For sequences in a CCDC or CCDCC template, beginning with all switches idle, the assumed order of events is

1. One of four inputs connects to one of four outputs.
2. One of three idle inputs connects to one of three idle outputs.
3. The first input-output pair is disconnected.
4. One of three idle inputs connects to one of three idle outputs.
5. In a CCDCC sequence only, one of two idle inputs connects to one of two idle outputs.

With $4 \times 4 = 16$ cases of first connection, $3 \times 3 = 9$ cases of second connection, 1 case of disconnection, $3 \times 3 = 9$ cases of third connection, and, in the CCDCC sequences only, $2 \times 2 = 4$ cases of fourth connection, the CCDC template contains $16 \times 9 \times 1 \times 9 = 1296$ sequences and the CCDCC template contains $16 \times 9 \times 1 \times 9 \times 4 = 5184$ sequences.

4.1.3 CCCDC sequences

For sequences in a CCCDC template, beginning with all switches idle, the assumed order of events is

1. One of four inputs connects to one of four outputs.
2. One of three idle inputs connects to one of three idle outputs.
3. One of two idle inputs connects to one of two idle outputs.
4. Either of the first two input-output pairs is disconnected.
5. One of two idle inputs connects to one ot two idle outputs.

With $4 \times 4 = 16$ cases of first connection, $3 \times 3 = 9$ cases of second connection, $2 \times 2 = 4$ cases of third connection, 2 cases of disconnection (either the first or second connection), and $2 \times 2 = 4$ cases of final connection. The CCCDC template contains $16 \times 9 \times 4 \times 2 \times 4 = 4608$ sequences.

4.2 Results

In either module, with a prudent switching rule, no CCCC sequences require rearrangement. Since all combinations of idle I/O pairs can be simultaneously interconnected in both modules, they are both at least rearrangeably nonblocking.

The 222 Module outperforms the 2121 Module under CCDC and CCDCC sequences. In the 222 Module, with a prudent switching rule, no CCDC sequences, nor CCDCC sequences, require rearrangement. However, in the 2121 Module with its unusual switching rule, 2 percent of the CCDC sequences require rearrangement and 6 percent of the CCDCC sequences require rearrangement.

Conversely, the 2121 Module outperforms the 222 Module under CCCDC sequences. In the 222 Module, with a prudent switching rule, 8 percent of all CCCDC sequences require rearrangement. However, in the 2121 Module, with its unusual switching rule, only 6 percent of all CCCDC sequences require rearrangement. Since neither module is generally nonblocking but both are at least rearrangeably nonblocking, they are identically rearrangeably nonblocking.

Summarizing transient analysis, Fig. 5 illustrates module performance versus template complexity for the two modules. The scale of the X axis has no mathematical nor physical meaning. The 222 Module outperforms the 2121 Module under all tested templates up to and including CCDCC sequences, but is outperformed by the 2121 Module under CCCDC sequences. I expected the 222 Module to be consistently better, so I was surprised by this turn of events. A qualitative explanation of this unusual behavior is elusive, but I propose a conjecture.

The 2121 Module is distorted from its optimal connectivity by simple sequences of events, because of its asymmetric junctor pattern and nonuniform switch-count in alternate paths. These distortions usually

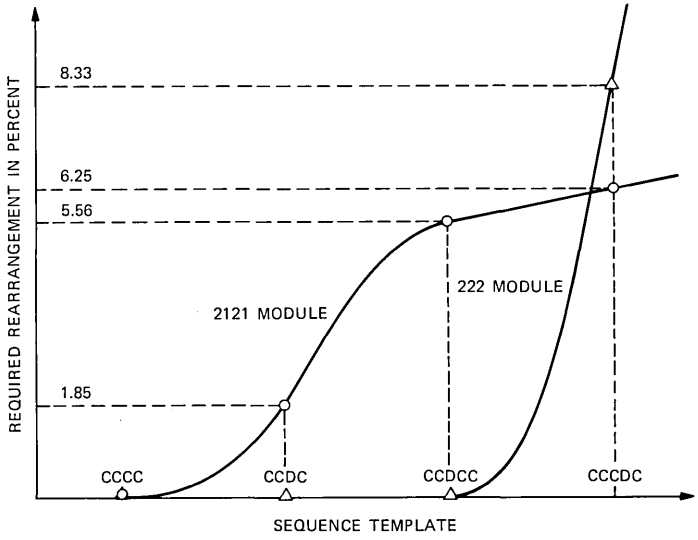


Fig. 5—Percentage of sequences requiring rearrangement versus template.

involve connections, like A to Z, that go diagonally across the module and have only one path. The severity of these distortions is softened by the presence of I/O pairs that have three paths, and by the opportunity to be clever in the choice of switching rule. The 222 Module, however, is not distorted from optimal connectivity until the event sequences are more complex. When it finally becomes distorted, it is more severely distorted than the 2121 Module.

V. MONTE CARLO SIMULATION

In a simulation, a sequence of randomly generated events is applied to an initially idle module. In this section, the results of simulation program are plotted in a scatter diagram and interpreted.

5.1 Program description

Two distinct programs were written—one for each module. The programs are, however, similar at a high level of description. Program variables record switch states, paths assigned to connections, interconnected parties, and other parameters and outputs.

The program begins by initializing a random number generator and reading module connectivity information from files. The program then enters a loop on traffic intensity: varying from 1.1 to 3.8 in increments of 0.3. Since the double exponential model of traffic is simulated, the average holding time and the average quiet time of each input are computed from the traffic intensity.

Rather than simulate the Poisson environment, by stepping through small intervals of time and simulating events, the program skips through time from one event to the next. Corresponding to each input i, is the time to the next event $tne[i]$, associated with that input. These values are initialized to an exponentially distributed random number whose mean is the quiet time. The number of rearrangements is initialized to zero and an interior loop, on the number of events in the simulation, is entered.

Time is advanced to the minimum value of the four $tne[i]$ and all $tne[i]$ are reduced by that value. The event that timed out is simulated. If it was a holding time that expired, the connection associated with the relevant input is disconnected and $tne[i]$ for that input is set to a random quiet time. If it was a quiet time that expired, a connection is made with the relevant input.

The connection is established by first locating a random idle input. A subroutine that implements the switching rule determines the best available path through the module. If there is no available path, the count of rearrangements is incremented, and established connections are disconnected and reconnected by their best paths. The connection

associated with the event is established and *tne*[*i*] for that input is set to a random holding time.

The loop on events terminates, and after printing out results as a function of traffic intensity, the loop on traffic intensity terminates.

5.2 Results

Simulations were run with sequences of 1000 and 5000 events, for the 222 and 2121 Modules, under varying traffic load. These numbers of events should give statistically significant results, and should be large enough that the transient effects from starting idle would be damped out. The independent parameter is the mean of *traffic intensity*, the product of global rate-of-origination by per-call holding time, how many simultaneous connections exist in a module at any time. The dependent variable is the percentage of new connections requiring a rearrangement. Traffic intensity is varied from slightly over 1.0 to slightly under 4.0, and the result of each simulation is plotted as a scatter diagram in Fig. 6. Simulation results for the 222 Module are shown with × and for the 2121 Module with +. Results of simulations with 5000 events are circled and with 1000 events are not.

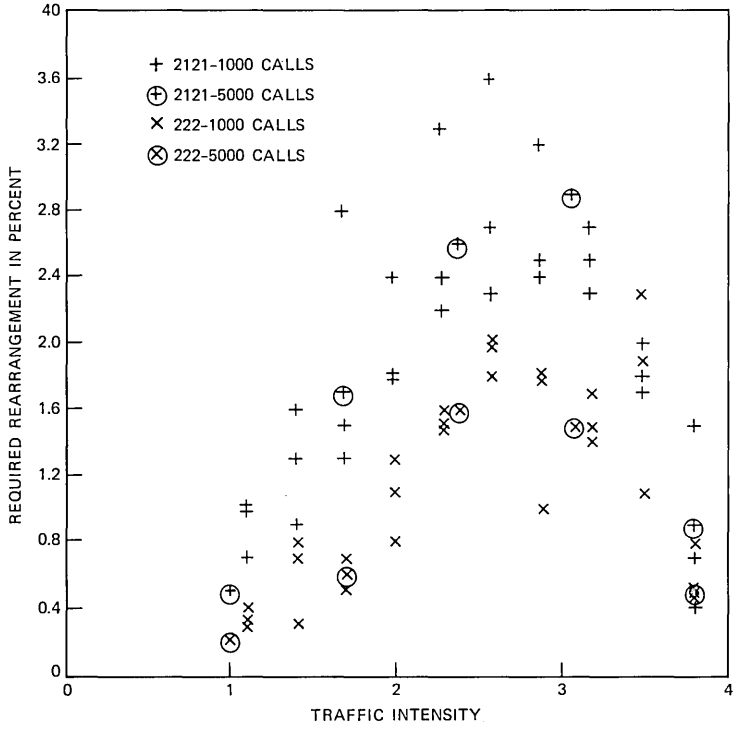


Fig. 6—Scatter diagram of P_{rr} versus traffic.

5.3 Interpretation

At low and moderate traffic intensities, it is noted that P_{rr} gets worse as the traffic intensity increases and that P_{rr} is uniformly, but slightly, worse in the 2121 Module than in 222 Module. In neither case is P_{rr} particularly bad, even at their maxima. That there is a maximum is somewhat surprising. A monotonically increasing P_{rr} might have been expected.

At high traffic load, there will be times when a 4×4 module is completely connected. The next event would have to be the disconnection of one I/O pair, and the event after that would be likely to be the reconnection of the same pair. It would be likely to be a connection because the traffic load is high, and it would have to be the same pair because they are the only idle input and output, the only connection that could be made. Such a connection would never require rearrangement because it was just disconnected. This behavior is consistent, whether the module is isolated as a simple 4×4 network or part of a large network.

A familiar result in classical statistics comes from applying Chebyshev's inequality to Bernoulli trials. This Bernoulli law of large numbers is used, for example in determining sample sizes of public opinion polls:

$$P(|f_n - p| > \epsilon) \leq p(1-p)/n\epsilon^2,$$

where f_n is the observed frequency after n trials, p is the given or assumed probability, and ϵ is an arbitrary tolerance. For simulations with $n = 1000$ events and $p = 0.03$, the inequality states that the variation in the outcome should be within ± 3 percent for 97 percent of the simulations. The tolerance is even less with smaller values of p. A casual glance at Fig. 6 shows that the variation is much greater than this. A conjectured explanation is that law of large numbers is derived from the assumption that the Bernoulli trials are independent. Since P_{rr} in the ith event of a network simulation is highly dependent on previous events, this fundamental assumption is invalidated.

VI. MARKOV ANALYSIS OF A GENERALIZED MODULE

Three analyses were performed. The first, presented in the remainder of this section, is of a generalized nature and pertains to both modules. It establishes a general model to be used as a check for the models of the modules under investigation. The second analysis, in Section VII, is based on a Markov model of the operation of the 222 Module. The third analysis, in Section VIII, is based on a Markov model of the operation of the 2121 Module.

6.1 The generalized model

In this oversimplified model, any network configurations in which the same count of connections are established are deemed to be equivalent. There are five equivalence classes of network configurations, corresponding to zero to four established connections, inclusive. Hence, there are five states in the corresponding Markov process, shown in Fig. 7. The model is general enough to cover both modules.

The stochastic behavior of the model is based on the classical traffic assumptions: Poisson-distributed service arrivals and exponentially

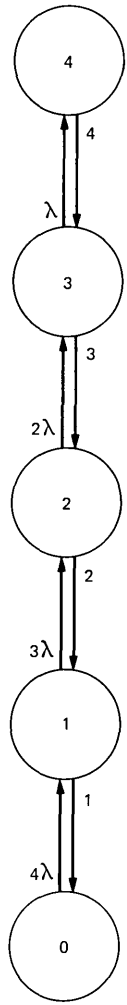


Fig. 7—Markov-queuing model.

distributed holding times. Since the Poisson and exponential distributions are mutual inverses, this model is equivalent to having Poisson-distributed arrivals of connections and disconnections or exponentially distributed off-hook and on-hook times (as used in the simulation program). Either way of looking at it, there are two random variables and there are two underlying parameters: the busyness and the true time scale. If results are considered per unit time or limited to steady-state behavior, then true time scale is irrelevant, and one random variable may be arbitrary and the other specifies busyness per unit time. We will use the double-Poisson process, for this and the two later models, and arbitrarily set the per-line disconnect rate to 1.

The state transitions are computed from the individual connect and disconnect rates of the four inputs (or outputs, equivalently), which are assumed to be statistically identical. Let λ be the rate at which each individual input requests a connection through the module, and let 1.0 be the rate at which each individual established connection is disconnected. In other words, λ is the ratio of the connect rate to the disconnect rate of each individual input (or output). The global rate of new connections from state i is $(4 - i)\lambda$, because there are $4 - i$ idle inputs that could make such a request. The global rate of disconnections from state i is i, because there are i established connections, any of which could request disconnection.

The analysis is a special case of a queue with dependence on the state of the system,[14] but can also be viewed as a queue with finite customer population and infinite servers.[15]

6.2 *Steady-state probabilities*

The steady-state probabilities are calculated from a conservation law. In the steady state, the mean exit rate from state i is the sum of the rates on all exit arcs from state i times the steady-state probability of being in state i. The mean entry rate into state i is the sum of the products of a rate on each entry arc times the steady-state probability of being in the state from which the arc comes. Conservation of calls dictates that the mean exit rate must equal the mean entry rate in the steady-state for each state. The resulting simultaneous equations have the general solution:[14]

$$g_n = (\Lambda_0 \cdots \Lambda_{n-1})g_0/(\mu_1 \cdots \mu_n),$$

where $n > 0$, Λ_i is the global connect rate on the arc from state i to state $i + 1$, μ_i is the global disconnect rate on the arc from state i to state $i - 1$, and g_i is the steady-state probability of being in state i. Setting $\Lambda_i = (4 - i)\lambda$ and $\mu_i = i$ gives

$$g_1 = (4\lambda)g_0/1 = 4\lambda g_0$$

$$g_2 = (4\lambda)(3\lambda)g_0/2 = 6\lambda^2 g_0$$

$$g_3 = (4\lambda)(3\lambda)(2\lambda)g_0/(2 \times 3) = 4\lambda^3 g_0$$

$$g_4 = (4\lambda)(3\lambda)(2\lambda)\lambda g_0/(2 \times 3 \times 4) = \lambda^4 g_0.$$

Note that each g_i is expressed as a function of g_0. The probability g_0 that there are no connections is found by setting the sum of all steady-state probabilities to 1. This gives the neat result

$$\sum g_i = (1 + 4\lambda + 6\lambda^2 + 4\lambda^3 + \lambda^4) \times g_0 = 1$$

$$g_0 = 1/(1 + \lambda)^4.$$

The steady-state probability mass function is

$$G(\lambda) = [1, 4\lambda, 6\lambda^2, 4\lambda^3, \lambda^4]/(1 + \lambda)^4.$$

6.3 Traffic intensity

Traffic intensity is a random variable giving the count of established connections at any time. In the classical traffic model, having Poisson arrivals with rate r and exponential holding time with mean h, traffic intensity is known to be Poisson distributed with mean, $\tau = rh$.

One way of looking at the model of Fig. 7 is that the system has four sources, each generating one calling cycle per unit time. The calling cycle consists of an exponentially distributed off-hook interval with mean $\lambda/(1 + \lambda)$ and an exponentially distributed on-hook interval with mean $1/(1 + \lambda)$. Setting the arrival rate to 4 and the holding time to the mean off-hook interval, the Poisson-distributed traffic intensity has mean $\tau = 4 \times \lambda/(1 + \lambda)$. This intuitive argument is verified by computing the mean count of established connections in the steady state

$$\tau = \sum (i \times g_i) = (0 \times 1 + 1 \times 4\lambda + 2 \times 6\lambda^2 + 3 \times 4\lambda^3 + 4 \times \lambda^4)/(1 + \lambda)^4$$

$$= 4\lambda(1 + 3\lambda + 3\lambda^2 + \lambda^3)/(1 + \lambda)^4$$

$$= 4\lambda/(1 + \lambda).$$

The inverse of this expression will prove useful later on:

$$\lambda = \tau/(4 - \tau).$$

6.4 Examples

Consider the two extremes. If $\lambda = 0$, the steady-state probability mass function is

$$G(0) = [1, 0, 0, 0, 0],$$

and the mean traffic intensity $\tau = 0$. If $\lambda \rightarrow \infty$, the limit of the steady-state state probability mass function is

$$G(\infty) \rightarrow [0, 0, 0, 0, 1],$$

and the limit of the mean traffic intensity $\tau \rightarrow 4$. As a better example, let $\lambda = 1$, which means that the individual arrival rate equals the departure rate, or that each input is on-hook for the same average time as off-hook. The steady-state probability mass function is

$$G(1.0) = [1, 4, 6, 4, 1]/16,$$

showing a trend toward state 2, with decreasing probability in either direction away from state 2. The symmetry about state 2 suggests an average of two connections in the steady state and setting $\lambda = 1$ in the equation for traffic intensity gives $\tau = 2$.

As another example, let $\lambda = 2.0$, meaning that each input is off-hook twice the time that it is on-hook, that is, two-thirds of the off-hook/on-hook cycle. The steady-state probability mass function is

$$G(0.5) = [1, 8, 24, 32, 16]/81,$$

showing a trend slightly under state 3 and a lack of the symmetry observed in the case where $\lambda = 1$. If $\lambda = 0.5$, the steady-state state probabilities are reversed from the case where $\lambda = 2$. The mean traffic intensity in these cases is 8/3 and 4/3, respectively, representing the center of mass for each distribution.

VII. MARKOV ANALYSIS OF THE 222 MODULE

7.1 The model

The state model of the 222 Module is illustrated in Fig. 8.[16] Each bubble in the figure represents a state and contains the state's name and a representative network configuration from the state's equivalence class. State I represents the idle network configuration and is equivalent to state 0 in the generalized model. State J represents the 16 network configurations with one connection established. These 16 network configurations are all equivalent, for purposes of determining P_{rr}, and this state is equivalent to state 1 in the generalized model.

States S through V represent four equivalence classes of network configurations in which two connections are established. This set of states is equivalent to state 2 in the generalized model. In state S, the two established connections terminate on the same input switch and on the same output switch. In states T and U, the two established connections terminate on opposite input switches and on opposite output switches. In state T, both middle switches are used and in state

U, one middle switch is shared by both established connections and the other is idle. State T is the only malevolent state in the model and the only transition in which rearrangement is required is the one from state T to state X. In state V, either the two established connections terminate on the same input switch and on opposite output switches, or the two established connections terminate on opposite input switches and on the same output switch. These conditions are equivalent for purposes of computing P_{rr}.

States W and X represent two equivalence classes of network configurations in which three connections are established. These two states are equivalent to state 3 in the generalized model. In state W, two of the established connections terminate on the same input switch and on the same output switch, and the third established connection terminates on the other input switch and the other output switch. In state X, the two established connections that terminate on the same input switch terminate on opposite output switches, and the two established connections that terminate on the same output switch terminate on opposite input switches. These are the only distinctions that need be made among all network configurations with three connections established.

States Y and Z represent two equivalence classes of network configurations in which four connections are established. These two states are equivalent to state 4 in the generalized model. In state Y, two established connections that terminate on the same input switch terminate on the same output switch. In state Z, two established connections that terminate on the same input switch terminate on opposite output switches. These are the only distinctions that need be made among all network configurations with four connections established.

7.2 State transitions

Corresponding to the transition from state I to J is a connection through the module in which two paths are possible. The choice of path is arbitrary and in no way affects the results. Corresponding to the transitions from state J to S or V is a second connection that has only one available path. Corresponding to the transition from state J to U, is a connection in which the selected path shares a middle switch with the established connection. Selecting the other path would correspond to an imprudent transition to state T.[16] Corresponding to transitions from states T, U, and V to states W and X are third connections that have only one available path, even with the rearrangement required before the connection corresponding to the transition from state T to X. Corresponding to the transition from state S to W, is a third connection in which two paths are possible. The

choice is arbitrary, not affecting the results, but the choice determines that established connection, which if later disconnected, would correspond to the undesired transition to state T. Corresponding to the transitions from states W or X to states Y or Z, respectively, are fourth connections in which only one path is available. The transition from state J to U represents the only connection where the switching rule is relevant.

The rates on the arcs connecting the states in Fig. 8 are similar to those in the generalized model, but several need further explanation.

- The sum of the connect rates on transitions from state J to states S through V must be 3λ, corresponding to the connect rate on the transition from state 1 to 2 in the generalized model. With one established connection, there are nine possible second connections: one that terminates on the same input and output switches as the established connection, four that terminate on the opposite input and output switches as the established connection, and four that terminate on one same switch and one opposite switch as the established connection. Thus the rates on the transitions to states S, U, and V are $3\lambda \times 1/9$, $3\lambda \times 4/9$, and $3\lambda \times 4/9$, respectively. The switching rule would prevent a direct transition from state J to state T; state U always being preferred over state T.
- The sum of the connect rates on transitions from each of states S through V must be 2λ, corresponding to the connect rate on the transition from state 2 to 3 in the generalized model. In states S and V, all third connections lead to the same next state, state W or X, respectively, and so each single transition is labeled with 2λ. In states T and U, however, half the third connections terminate on the same input and output switches as an existing connection, and half terminate on the same input switch as one connection and the same output switch as the other connection. Thus, states T and U each transit to both states W and X, and the rates on those transitions are all λ. All third connections associated with transitions from state T to state X require rearrangement, and these are the only connections requiring rearrangement in the entire model.[16] Either of the two established connections may be rearranged so that one middle switch is shared by both connections and one is idle—an intermediate network configuration that conforms to state U.
- The sum of the disconnect rates on transitions from each of states W and X must be 3, corresponding to the disconnect rate on the transition from state 3 to 2 in the generalized model. In any network configuration belonging to state X, two of the three single disconnections results in a configuration belonging to state V, so that arc is labeled with a rate of 2. The other single disconnection results in a configuration belonging to state U, so that arc is labeled with a

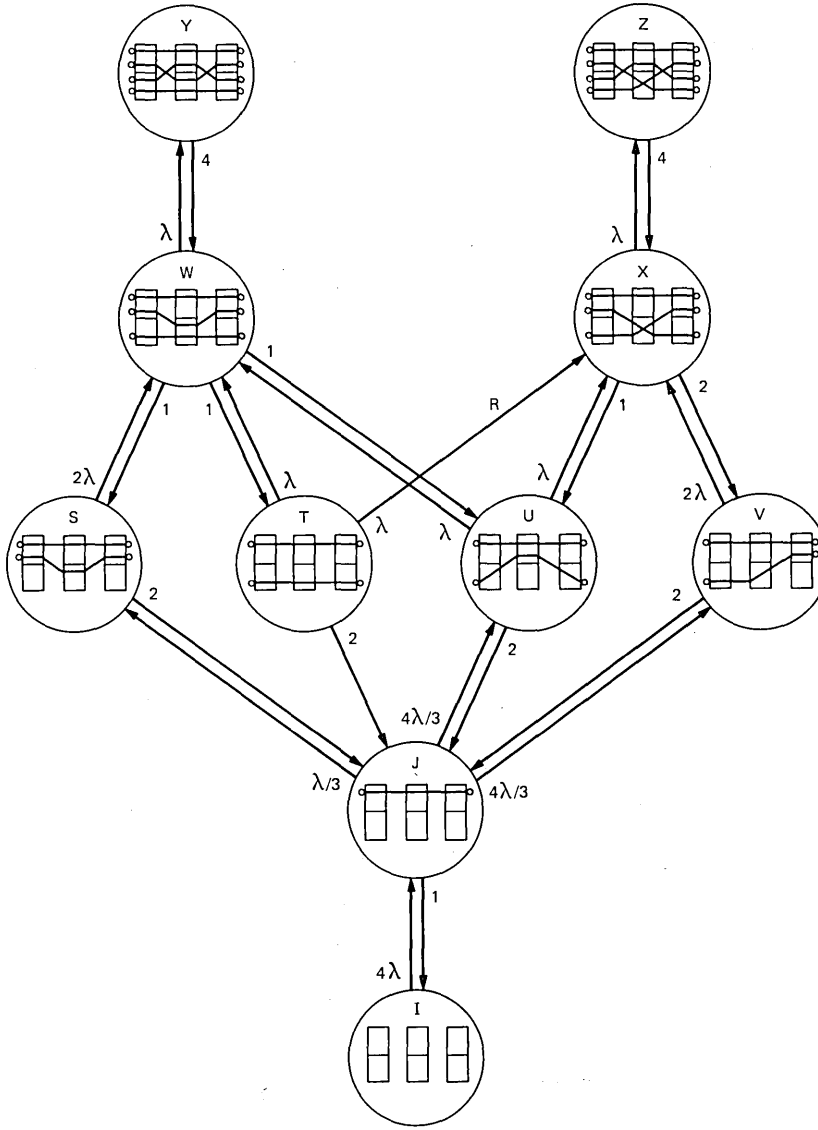


Fig. 8—Markov model for the 222 Module.

rate of 1. In any network configuration belonging to state W, a transition to a configuration belonging to state V is impossible, but each of the three single disconnections results in a configuration belonging to state S, T, or U, respectively. Thus, each of these transitions is labeled with a rate of 1. The transition from state W to T is the only entry into that malevolent state, and it is unavoidable under any prudent switching rule.

The rates on the state transitions and the notion of λ in this development are slightly different from that of the reference.[16] In that paper, λ represented the rate of origination of connections between specific inlet and outlet pairs. In this development, traffic is assumed to originate at an inlet, and terminate on any random idle outlet.

7.3 Steady-state probabilities

The rate equations for each of the ten states are

$$
\begin{aligned}
4\lambda p_I &= p_J \\
(3\lambda + 1)p_J &= 4\lambda p_I + 2(p_S + p_T + p_U + p_V) \\
(2\lambda + 2)p_S &= (\lambda/3)p_J + p_W \\
(2\lambda + 2)p_T &= p_W \\
(2\lambda + 2)p_U &= (4\lambda/3)p_J + p_W + p_X \\
(2\lambda + 2)p_V &= (4\lambda/3)p_J + 2p_X \\
(\lambda + 3)p_W &= 2\lambda p_S + \lambda p_T + \lambda p_U + 4p_Y \\
(\lambda + 3)p_X &= \lambda p_T + \lambda p_U + 2\lambda p_V + 4p_Z \\
4p_Y &= \lambda p_W \\
4p_Z &= \lambda p_X.
\end{aligned}
$$

As is customary with such processes, only $n - 1$ of the n equations are independent. Setting the sum of the n probabilities to 1 provides the nth independent equation. The solution, after several hours of manual algebra is

$$
\begin{aligned}
p_I &= 1/(1 + \lambda)^4 \\
p_J &= 4\lambda/(1 + \lambda)^4 \\
p_S &= 2\lambda^2/3(1 + \lambda)^4 \\
p_T &= 2\lambda^3/3(1 + \lambda)^5 \\
p_U &= \lambda^2(8 + 6\lambda)/3(1 + \lambda)^5 \\
p_V &= 8\lambda^2/3(1 + \lambda)^4 \\
p_W &= 4\lambda^3/3(1 + \lambda)^4 \\
p_X &= 8\lambda^3/3(1 + \lambda)^4 \\
p_Y &= \lambda^4/3(1 + \lambda)^4 \\
p_Z &= 2\lambda^4/3(1 + \lambda)^4.
\end{aligned}
$$

As a check, it is verified that p_I above equals g_0 from the generalized model, $p_J = g_1$, $p_S + p_T + p_U + p_V = g_2$, $p_W + p_X = g_3$, and $p_Y + p_Z = g_4$. Of particular interest, of course, is the steady-state probability of the malevolent state, p_T.

7.4 Probability of requiring a rearrangment

P_{rr} is the proportion of those new connections that require that an (one) established connection be rearranged before a new connection may be completed. The numerator is the sum over all states of ⟨the average count of new connections requiring rearrangement from state i⟩ × ⟨the steady-state probability of state i⟩. For this network, it is simply $1 \times p_T$. The denominator is the weighted average count of possible new connections, similar to the calculation of τ in the previous section

$$\sum(4 - i) \times g_i = (4 \times 1 + 3 \times 4\lambda + 2 \times 6\lambda^2 + 1 \times 4\lambda^3 + 0 \times \lambda^4)/(1 + \lambda)^4$$
$$= 4(1 + 3\lambda + 3\lambda^2 + \lambda^3)/(1 + \lambda)^4$$
$$= 4/(1 + \lambda).$$

The ratio is then

$$P_{rr} = [2\lambda^3/3(1 + \lambda)^5] \div [4/(1 + \lambda)] = \lambda^3/6(1 + \lambda)^4,$$

or, as a function of traffic intensity,

$$P_{rr} = \tau^3(4 - \tau)/1536.$$

The curve for this expression is plotted in Fig. 9 through the data

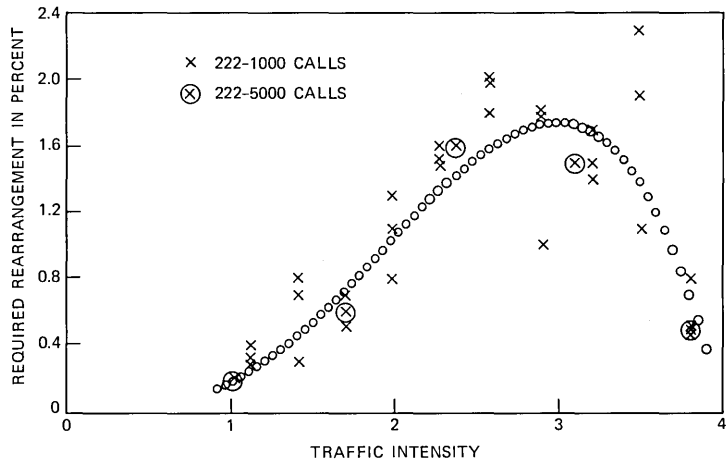


Fig. 9—Simulation results and derived curve for the 222 Module.

points for the 222 Module, taken from the scatter diagram of Fig. 6. Setting the derivative of the expression above to zero yields an extremum at $\lambda = \tau = 3$, and the value at that maximum is 1.76 percent.

Related to this figure is N_{rr}, the average count of established connections that must be rearranged when a new connection is completed. Since exactly one established connection is rearranged when the 222 Module requires rearrangement,

$$N_{rr} = P_{rr} = \lambda^3/6(1 + \lambda)^4 = \tau^3(4 - \tau)/1536.$$

VIII. MARKOV ANALYSIS OF THE 2121 MODULE

8.1 The model

Since the Markov model for the 2121 Module has 50 states, a natural nomenclature is a mapping to a familiar set with 50 elements, also called "states." The mapping is shown in Fig. 10, where each U. S. state represents a set of equivalent network configurations from the 2121 Module. A representative configuration is shown with each state in Fig. 10. The states are grouped according to the number of established connections, or their relationship to states in the generalized model. FL represents the idle Markov state and the six New England states represent six Markov states in which all four inputs connect to all four outputs. States in three intermediate east-to-west bands across the U. S. A. represent Markov states with one, two, and three established connections, respectively. To avoid clutter in Fig. 10, the transitions among the states are shown in later figures.

8.2 State transitions

The upward transitions in the model for the 2121 Module, in which no rearrangements are required, are shown in Fig. 11. Consider the three upward transitions from FL. The transition from FL to GA represents establishing, in an originally idle module, an A-to-Z (or C-to-W) connection diagonally across the module. This connection has only one path through the module, and that path requires the central switch in the crossed state. By contrast, A-to-W (or C-to-Z) connections have two paths through the module, represented by NM and AZ. A direct upward transition from FL to AZ, and none from FL to NM, demonstrates the preference for the path that avoids the central switch. Similarly, the direct upward transition from FL to AL, and lack of same to LA or TX, demonstrates the choice of the path for an A-to-X connection that avoids the central switch over the other two paths. Similar logic governs the other upward transitions on the graph. For simplicity, the weights on the arcs are not shown, but they are similar to those of the 222 Module.

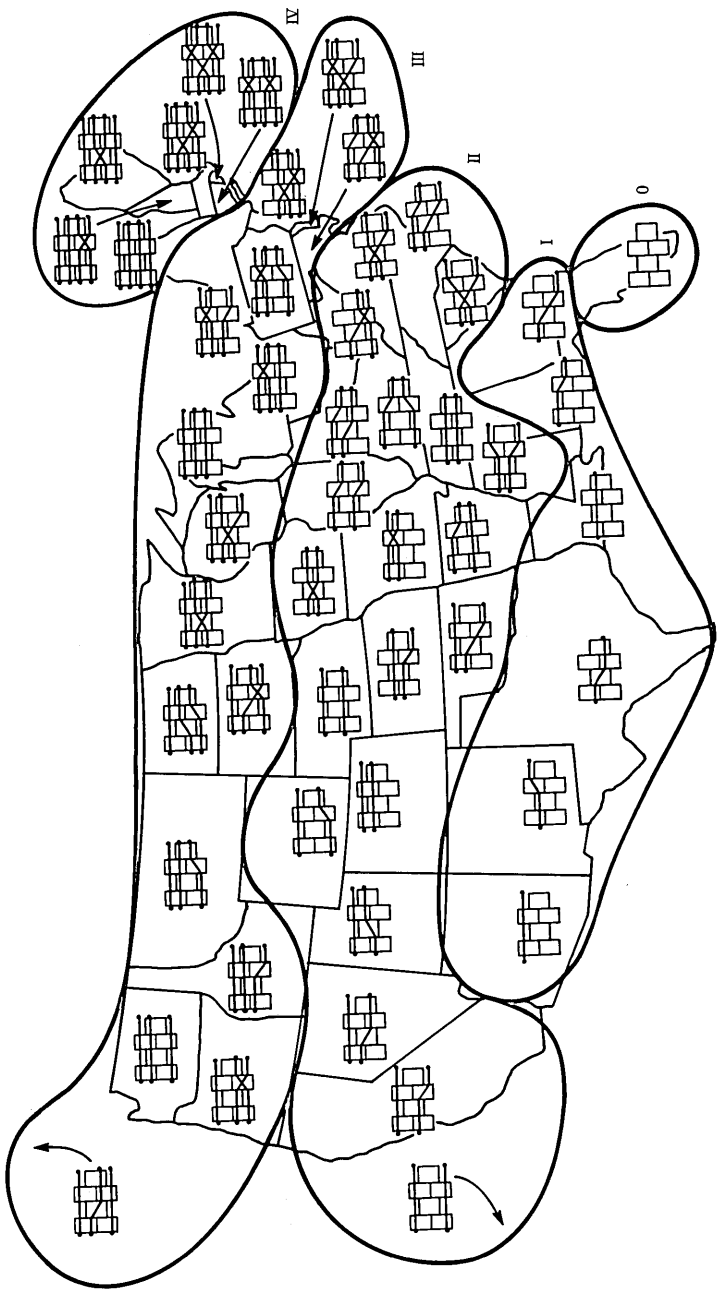


Fig. 10—Markov states and nomenclature for the 2121 Module.

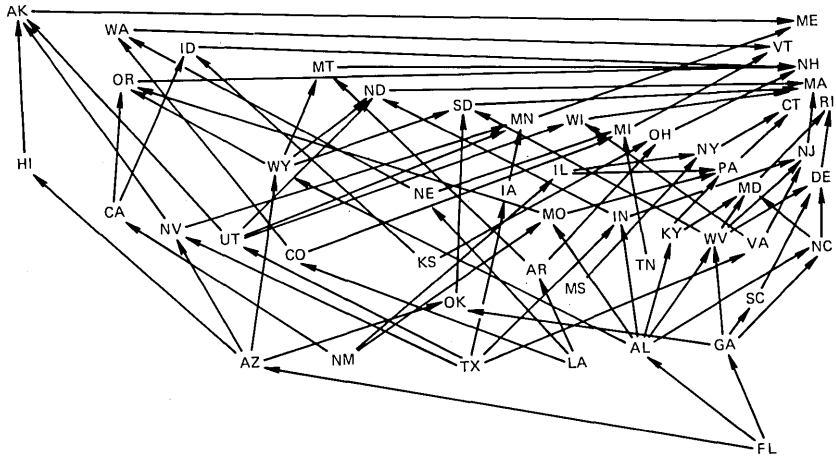


Fig. 11—Direct upward transitions in the 2121 Model.

The upward transitions, in which one or two established connections require rearrangement, are shown in Fig. 12. The three states with one established connection from which rearrangements may be required—LA, TX, and NM—have no direct upward transitions from FL, while the only three states that can be reached directly from FL—GA, AL, and AZ—have no upward transitions requiring rearrangement. Two of the states with two established connections from which any rearrangements may be required—TN and KS—have no upward transitions from any state with one established connection. The other five states with two established connections from which any rearrangements may be required—VA, IL, AR, IA, and NE—have upward transitions from states with one established connection, but only from LA, TX, and NM, the states that cannot be reached directly from FL. The transitions from IA to MD, KS to SD, and KS to ND are particularly interesting because both established connections require rearrangement before the new connection, corresponding to the transition, can be completed. Since the states from which rearrangement is required are reached only through complex sequences of connection and disconnection, we expect P_{rr} to be small.

The downward transitions in the model for the 2121 Module are shown in Fig. 13. The rates on all the arcs connecting the states, as in the two previous figures, are not shown.

8.3 Steady-state probabilities

The rate equations for each of the 50 states are given in Appendix A. A closed-form solution to this set of equations has not been found, but approximations were derived by a simple, and tedious, iteration.

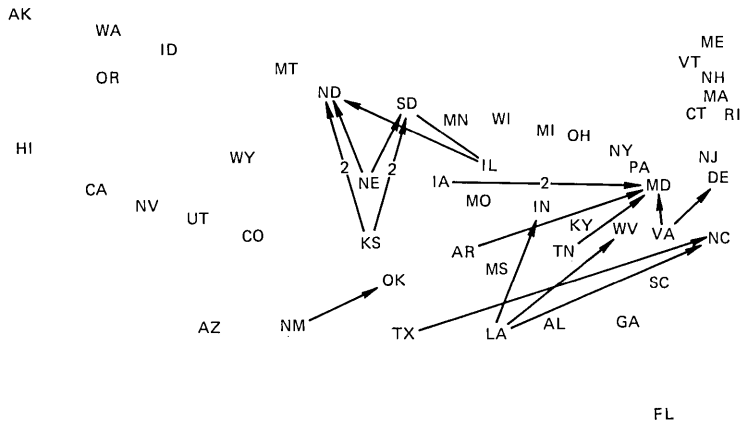


Fig. 12—Transitions requiring rearrangement in the 2121 Model.

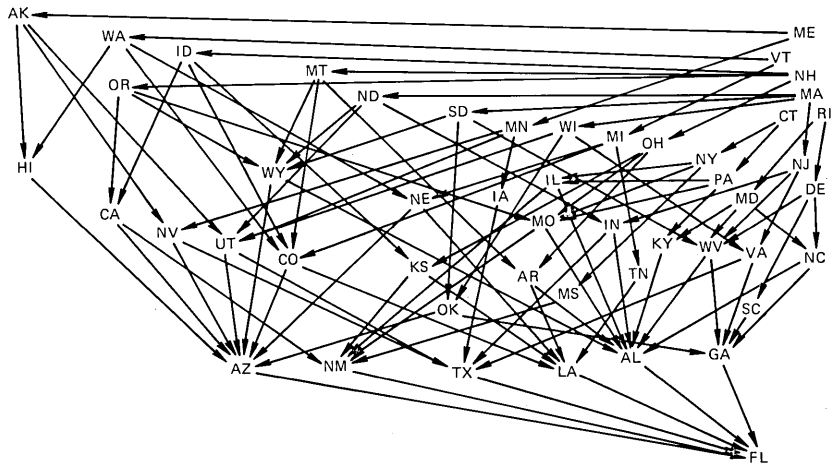


Fig. 13—Downward transitions in the 2121 Model.

The states are initially distributed equilikely within their equivalence class such that the sum of the state probabilities within each class equals the corresponding state probability in the generalized model. Thus,

$$q_{FL} = 1/(1 + \lambda)^4,$$

$$q_x = 4\lambda/6(1 + \lambda)^4 \text{ for the 6 states: GA to AZ,}$$

$$q_x = 6\lambda^2/21(1 + \lambda)^4 \text{ for the 21 states: NC to HI,}$$

$$q_x = 4\lambda^3/16(1 + \lambda)^4 \text{ for the 16 states: DE to AK,}$$

$$q_x = \lambda^4/6(1 + \lambda)^4 \text{ for the 6 New England states.}$$

Then, temporarily deleting $(1 + \lambda)^4$ from every denominator, these expressions were substituted into the right side of the rate equations in Appendix A. The resulting intermediate expressions for each q_x are polynomials in λ divided by the $(a + b\lambda)$ term on the left side of the corresponding rate equation. The long division was effected and the quotient was truncated to a simple polynomial in λ, a new expression for each q_x. Most of these new expressions had coefficients between double and half their analogs in the original expressions.

These new expressions were then substituted into the rate equations, and a similar simplification and approximation was effected. By the third iteration, the expressions were surprisingly close to those of the second iteration, and rapid convergence was observed. These approximations to the steady-state probabilities, without the long division and quotient truncation in the final iteration, are given in Appendix B. Of particular interest, again, are the steady-state probabilities of the ten malevolent states.

8.4 Probability of requiring a rearrangement

P_{rr} is the proportion of those new connections that require rearrangement of an (at least one) established connection before the new connection may be completed. The numerator is the sum over all states of ⟨the average count of new connections requiring rearrangement from state i⟩ × ⟨the steady-state probability of state i⟩. For the 2121 Module, the numerator is

$$(4/3) \times q_{LA} + (1/3) \times q_{TX} + (1/3) \times q_{NM} + 1 \times q_{VA} + 1 \times q_{TN} + 1 \times q_{IL} + 1 \times q_{AR} + 1 \times q_{IA} + 1 \times q_{KS} + 1 \times q_{NE} = (2.1\lambda^4 + 4.8\lambda^3 + 1.0\lambda^2 - 1.1\lambda)/(1 + 3\lambda)(1 + \lambda)^5,$$

using the approximate expressions from Appendix B. The denominator, $4/(1 + \lambda)$, is the weighted average count of possible new connections, as in the calculation from the previous section. The ratio is then

$$P_{rr} = (0.5\lambda^4 + 1.2\lambda^3 + 0.2\lambda^2 - 0.3\lambda)/(1 + 3\lambda)(1 + \lambda)^4$$

or, as a function of traffic intensity,

$$P_{rr} = (0.5\tau^4(4 - \tau) + 1.2\tau^3(4 - \tau)^2 + 0.2\tau^2(4 - \tau)^3 - 0.3\tau(4 - \tau)^4)/512(2 + \tau) = \tau(\tau - 4)(\tau^3 + 2\tau^2 - 88\tau + 96)/2560(2 + \tau).$$

The curve for this expression is plotted in Fig. 14 through the data points for the 2121 Module, taken from the scatter diagram of Fig. 6. Setting the derivative of the expression above to zero yields an extremum at $\tau = 2.6$, and the value at that maximum is 3.1 percent.

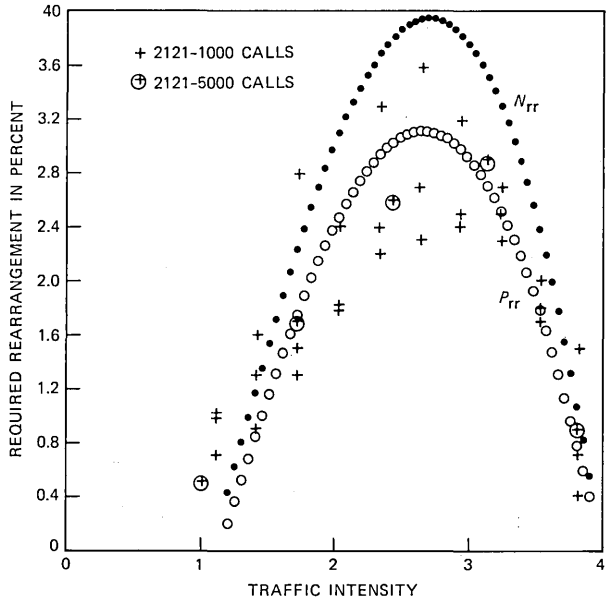


Fig. 14—Simulation results and derived curve for the 2121 Module.

Related to this figure is, N_{rr}, the average count of established connections that require rearrangement when a new connection is completed. Since this count is two on three of the state transitions, this figure is not equal to P_{rr}:

$$N_{rr} = [(4/3) \times q_{LA} + (1/3) \times q_{TX} + (1/3) \times q_{NM} + 1 \times q_{VA} + 1 \times q_{TN} + 1 \times q_{IL} + 1 \times q_{AR} + 2 \times q_{IA} + 2 \times q_{KS} + 1 \times q_{NE}]/[4/(1 + \lambda)]$$

$$\cdot N_{rr} = \tau(\tau - 4)(\tau^3 - 2\tau^2 - 88\tau + 96)/2560(2 + \tau).$$

This expression is also plotted in Fig. 14. We observe that the Markov analysis of such an innocent-looking network is surprisingly complex.

IX. NETWORK CONFIGURATIONS

Two algorithms for a switching rule are by direct calculation or by table look-up. The direct calculation of the switching rule for either module would be time-consuming, with the switching rule for the 2121 Module significantly more complicated than that for the 222 Module. The count of connection combinations in either module is also large, with little difference between the two modules, so a table look-up implementation of the switching rules would require a large ROM.

Two distinct realizations of such an algorithm are by a microcontroller per module or by a common controller governing all modules

in a network. Direct calculation in a common controller of a large network suggests a real-time bottleneck and table look-up in a per-module algorithm suggests considerable replication of a large ROM. Therefore, the logical choices are distributed control by direct calculation in a per-module microcontroller, or centralized control by table look-up. The count of network configurations in a table look-up algorithm is discussed in this section.

9.1 Count of 222 Module configurations

The Markov diagram of the 222 Module is repeated in Fig. 15, except that an additional number is placed in each bubble, the count of unique network configurations that are represented by the Markov state. State I represents the only idle network configuration. Considering configurations with a single established connection, since any of four inputs can be connected to any of four outputs, and each connection has two paths through the network, state J represents $4 \times 4 \times 2 = 32$ network configurations.

States S through V represent all configurations with two established connections. A configuration in state S derives from a configuration in state J by connecting the only other input on the same input switch as the established connection to the only other output on the same output switch as the established connection by the only available path. Thus state S also represents 32 configurations. A configuration in state T derives from a configuration in state J by connecting either input on the opposite input switch as the established connection to either output on the opposite output switch as the established connection by the path using the unused middle switch. Thus state T represents $32 \times 2 \times 2 = 128$ configurations. A configuration in state U derives from a configuration in state J by connecting either input on the opposite input switch as the established connection to either output on the opposite output switch as the established connection by the path sharing the used middle switch. Thus state U represents $32 \times 2 \times 2 = 128$ configurations. A configuration in state V derives from a configuration in state J by connecting the only other input on the same input switch as the established connection to either output on the opposite output switch as the established connection, or either input on the opposite input switch as the established connection to the only other output on the same output switch as the established connection by the only available path. Thus state V represents $32 \times (2 + 2) = 128$ configurations.

States W and X represent all configurations with three established connections. A configuration in state W derives from a configuration in state S by connecting either input on the unused input switch to either output on the unused output switch by either available path, or

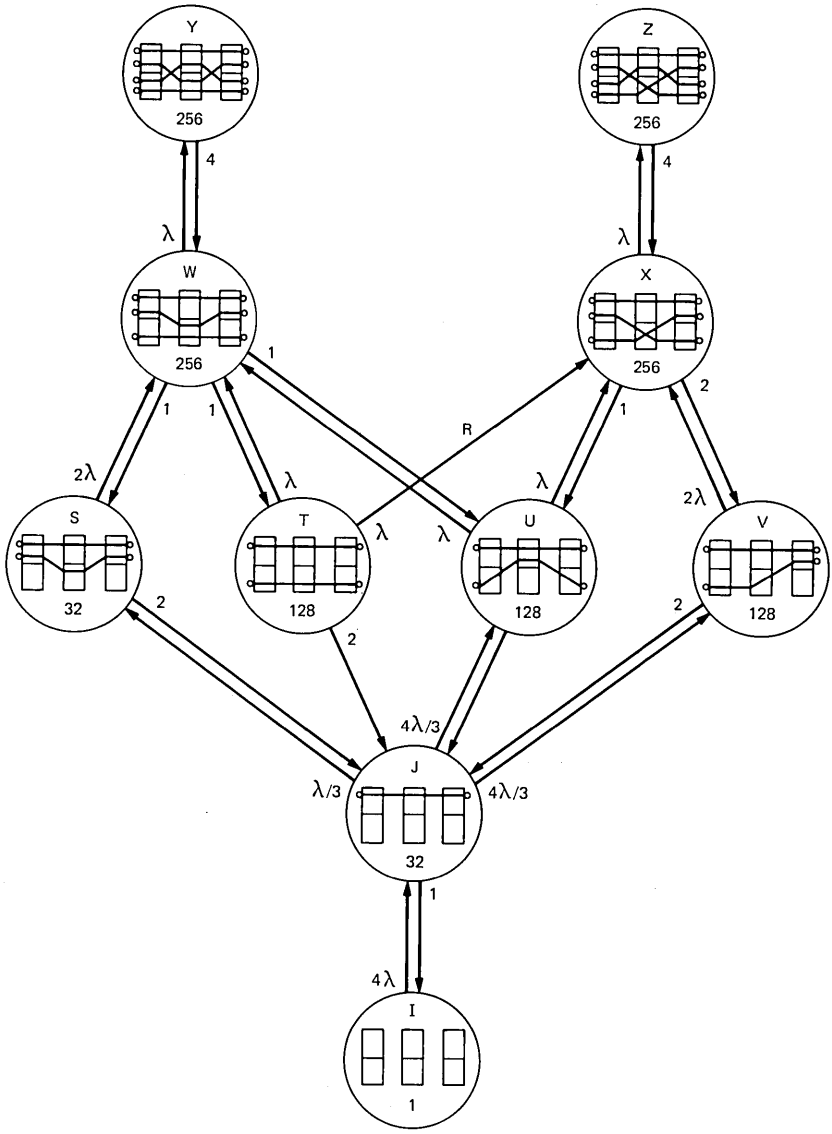


Fig. 15—Configurations per state in the 222 Module.

it derives from a configuration in state T or U by interconnecting either of two specific I/O pairs by the only available path. The specific pair share both an input switch and an output switch with either established connection. Thus state W represents $32 \times 2 \times 2 \times 2 = 128 \times 2 = 256$ configurations. A configuration in state X derives from a configuration in state U by connecting either of two specific I/O pairs,

those that share an input switch with one established connection and an output switch with the other one, by the only available path. Each also derives from configurations in state V by connecting either idle input to either idle output by the only available path, but each derives from two different configurations in state V. Thus state X represents $128 \times 2 = 128 \times 2 \times 2/2 = 256$ configurations.

States Y and Z represent all configurations with four established connections. Each configuration in state Y derives from a configuration in state W by connecting the only idle I/O pair by the only available path. Each configuration in state Z derives from a configuration in state X by connecting the only idle I/O pair by the only available path. Thus state Y represents 256 configurations and state Z represents 256 configurations.

Summing, we count a total of

$$1 + 32 + 32 + 128 + 128 + 128 + 256 + 256 + 256 + 256 = 1473$$

distinct network configurations in the 222 Module.

9.2 Count of 2121 Module configurations

The Markov diagram of the 2121 Module, showing downward transitions, is repeated in Fig. 16, except that a number replaces the state name, the count of unique network configurations that are represented by the corresponding Markov state.

State FL represents the only idle network configuration. States GA through AZ represent all configurations with one established connection. In the four configurations in GA, any of the four inputs connects to the output on a middle switch diagonally across the module by the only path. In the configurations in AL, LA, and TX, any of the four inputs connects to either output on the output switch. Each state represents $4 \times 2 = 8$ configurations and the states are distinguished by which of three paths is used for the connection. In the configurations in NM and AZ, any of the four inputs connects to the output on a middle switch directly across the module. Each state represents four configurations and the states are distinguished by which of two paths is used for the connection. Summing it up, states GA through AZ represent $3 \times 4 + 3 \times 8 = 36$ configurations.

States NC through HI represent all configurations with two established connections. In one subset of these 21 states, the inputs, one from each input switch, connect to the two outputs on the middle switches. Each state in this subset—SC, MS, or HI—represents four configurations, and the states are distinguished by whether the connections are diagonally across the module or directly across by either of two similar paths. In OK, either output on a middle switch connects to either input directly across the module by the best path, and the

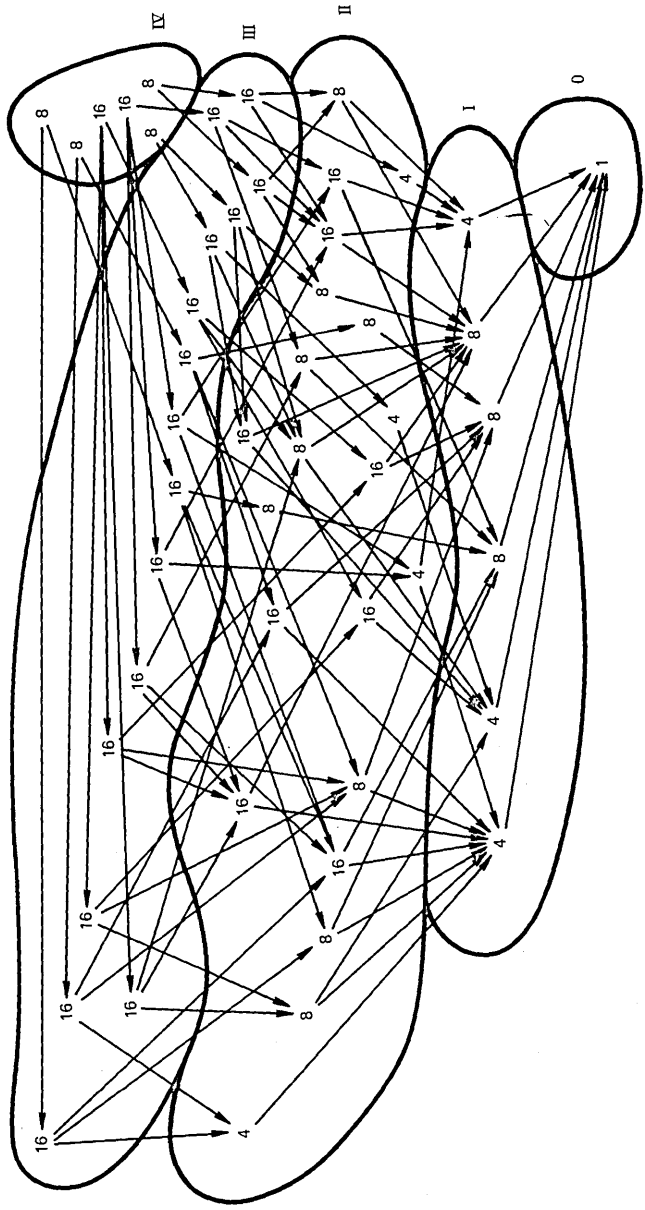


Fig. 16—Configurations per state for the 2121 Module.

other input on that same input switch connects diagonally across the module to the other output on a middle switch. In CA, either output on a middle switch connects to either input directly across the module by the best path and either input on the other input switch connects directly across the module to the other output on a middle switch by the worst path. OK represents $2 \times 2 = 4$ configurations and CA represents $2 \times 2 \times 2 = 8$ configurations.

In another subset of these 21 states, any of the four inputs connects to either output on the output switch and the other input on the same switch connects to any other output by the best remaining path. Each state in this subset—NC, IN, MO, CO, or NV—represents $4 \times 2 = 8$ configurations, and the states are distinguished by the path of the first connection and/or the output of the second connection. In another subset of these 21 states, either input on the upper input switch connects to either output on the output switch, and either input on the lower input switch connects to the other output on the output switch, and both connections use similar paths. Each state in this subset—KY, TN, or IA—represents $2 \times 2 \times 2 = 8$ configurations, and the states are distinguished by the three possible paths. In the final subset of these 21 states, any of the four inputs connects to either output on the output switch and either input on the other input switch connects to any other output. Excluding the cases, covered in the previous subset, where the paths are symmetric, each state in this subset—VA, WV, IL, AR, NE, KS, WY, and UT—represents $4 \times 2 \times 2 = 16$ configurations, and the states are distinguished by the path of the first connection and/or the output and/or the path of the second connection. Summing it up, states NC through HI represent $4 \times 4 + 9 \times 8 + 8 \times 16 = 216$ configurations.

States DE through AK represent all configurations with three established connections. In all 16 states, any of four inputs connects to either output on the output switch, and either input on the other input switch connects to some other output. Each state represents $4 \times 2 \times 2 = 16$ configurations, and the states are distinguished by the path of the first connection and/or the path and/or output of the second connection and/or the input and/or output and/or path of the third connection. Summing then, states DE through AK represent $16 \times 16 = 256$ configurations.

States RI through VT represent all configurations with four established connections. In the subset containing RI, CT, ME, and VT, either input on the upper input switch connects to either output on the output switch, either input on the lower input switch connects to the other output on the output switch, and both connections use similar paths. The remaining two inputs connect to the remaining two outputs, by the best remaining similar paths. Each state represents

$2 \times 2 \times 2 = 8$ configurations, and the states are distinguished by the path of the first two connections and/or the parties interconnected by the last two connections. In MA, both inputs on one input switch connect to the outputs on the middle switches by the best paths, and both inputs on the other input switch connect to the outputs on the output switch. In NH, both inputs on one input switch connect, one to the output straight across on a middle switch and the other to one of the outputs on the output switch, using paths like those in VT. Both inputs on the other input switch connect to similar outputs, but using paths like those in CT. Both MA and NH represent $2 \times 4 \times 2 = 16$ configurations. Summing them up, states RI through VT represent $4 \times 8 + 2 \times 16 = 64$ configurations.

Summing them up, we count a total of

$$1 + 36 + 216 + 256 + 64 = 573$$

distinct network configurations in the 2121 Module. The module supports more configurations, but no others are reached using the prudent rule. We see that the 222 Module has more than 2.5 times as many reachable configurations as the 2121 Module and, hence, a table-look-up implementation of a control algorithm would be more complex for the 222 Module than for the 2121 Module.

X. CONCLUSION

Two architectures for a 4×4 photonic switching network were compared by their traffic-handling capacity. Both networks are rearrangeably nonblocking. The percentage of sequences requiring rearrangement was found to be tolerable for both modules. Thus, both modules are judged acceptable, insofar as rearrangeably nonblocking modules are acceptable, and practically indistinguishable in their traffic performance. The selection of one module over the other may proceed based on criteria other than traffic capacity, like loss, crosstalk, or cost of manufacture.

XI. ACKNOWLEDGMENTS

Acknowledgments are extended to Vic Beneš, Nick Maxemchuk, and Krishnan Padmanabhan for their time, encouragement, and useful discussion and to Eric Grosse for his help with SMP.[17]

REFERENCES

1. G. Broomell and J. R. Heath, "Classification Categories and Historical Development of Circuit Switching Topologies," Comput. Surv., *15,* No. 2 (June 1983), pp. 95–133.
2. V. E. Beneš, *Mathematical Theory of Connecting Networks,* New York: Academic Press, 1965.

3. R. C. Alferness, R. V. Schmidt, and E. H. Turner, "Characteristics of Ti-Diffused Lithium Niobate Optical Directional Couplers," Appl. Opt., *18,* No. 23 (December 1979), pp. 4012–16.
4. A. E. Joel, "On Permutation Switching Networks, B.S.T.J., *47,* No. 5 (May 1968), pp. 813–22.
5. S. K. Korotky et al., "Fully Connectorized High-Speed Ti:$LiNbO_3$ Switch/Modulator for Time-Division Multiplexing and Data Encoding," J. Lightwave Technol., *LT-3,* No. 1 (February 1985).
6. L. McCaughan, "Design and Performance Limitation of Integrated Electro-Optic Cross Points," Proc. IEEE Globecom, Alanta, November 1984, pp. 878–9.
7. J. E. Watson, "Polarization-Independent 1 × 16 Optical Switch Using Ti:$LiNbO_3$ Waveguides," Conf. Optical Fiber Commun., San Diego, February 1985, p. 110.
8. H. S. Hinton, "A Nonblocking Optical Interconnection Network Using Directional Couplers," Proc. of IEEE Globecom, Atlanta, November 1984, pp. 885–9.
9. L. McCaughan and G. A. Bogert, "4 × 4 Strictly Nonblocking Integrated Ti:$LiNbO_3$ Switch Array," Conf. Optical Fiber Commun., San Diego, February 1985, p. 76.
10. E. E. Bergman, L. McCaughan, and J. E. Watson, "Coupling of Intersecting Ti:$LiNbO_3$ Diffused Waveguides," Appl. Opt., *23,* No. 17 (September 1984), pp. 3192–5.
11. S. Kobayashi and T. Kimura, "Semiconductor Optical Amplifiers," IEEE Spectrum, *21,* No. 5 (May 1984), pp. 26–33.
12. R. A. Spanke and V. E. Beneš, private communication.
13. R. A. Thompson, private communication.
14. M. Schwartz, Computer-Communication Network Design and Analysis, Englewood Cliffs, N.J.: Prentice-Hall, 1977.
15. L. Kleinrock, Queueing Systems, New York: Wiley, 1975.
16. V. E. Beneš, "Programming and Control Problems Arising from Optimal Routing in Telephone Networks," B.S.T. J., *45,* No. 9 (November 1966), pp. 1373–438.
17. SMP, A Symbolic Manipulation Program, Primer and Summary, Inference Corp., 1983.

APPENDIX A

Rate Equations for the 2121 Module

$$(4\lambda)\ q_{FL} = q_{AZ} + q_{NM} + q_{TX} + q_{LA} + q_{AL} + q_{GA}$$

$$(3\lambda + 1)\ q_{AZ} = \lambda q_{FL} + 2q_{HI} + q_{CA} + q_{NV} + q_{UT} + q_{WY} + q_{CO} + q_{NE} + q_{OK}$$

$$(3\lambda + 1)\ q_{NM} = q_{CA} + q_{KS} + q_{IL} + q_{MO} + 2q_{MS}$$

$$(3\lambda + 1)\ q_{TX} = q_{NV} + q_{UT} + 2q_{IA} + q_{IN} + q_{VA}$$

$$(3\lambda + 1)\ q_{LA} = q_{CO} + q_{KS} + q_{NE} + q_{AR} + 2q_{TN}$$

$$(3\lambda + 1)\ q_{AL} = 2\lambda q_{FL} + q_{WY} + q_{AR} + q_{MO} + q_{IL} + q_{IN} + 2q_{KY} + q_{WV} + q_{NC}$$

$$(3\lambda + 1)\ q_{GA} = \lambda q_{FL} + q_{OK} + q_{WV} + q_{VA} + 2q_{SC} + q_{NC}$$

$$(2\lambda + 2)\ q_{HI} = (2\lambda/3)\ q_{AZ} + q_{AK} + q_{WA}$$

$$(2\lambda + 2)\ q_{CA} = (2\lambda/3)\ q_{NM} + q_{OR} + q_{ID}$$

$$(2\lambda + 2)\ q_{NV} = (\lambda/3)[2q_{AZ} + q_{TX}] + q_{AK} + q_{MN}$$

$$(2\lambda + 2)\ q_{UT} = (2\lambda/3)\ q_{TX} + q_{AK} + q_{ND} + q_{MN} + q_{WI}$$

$$(2\lambda + 2)\ q_{WY} = (\lambda/3)[4q_{AZ} + 2q_{AL}] + q_{OR} + q_{MT} + q_{ND} + q_{SD}$$

$$(2\lambda + 2)\ q_{CO} = (\lambda/3)\ q_{LA} + q_{WA} + q_{ID} + q_{MT} + q_{MI}$$

$$(2\lambda + 2)\ q_{NE} = (2\lambda/3)\ q_{LA} + q_{WA} + q_{MI}$$

$$(2\lambda + 2)\ q_{KS} = q_{ID} + q_{OH}$$

$$(2\lambda + 2)\ q_{OK} = (\lambda/3)[q_{AZ} + q_{NM} + q_{GA}] + q_{SD} + q_{WI}$$

$$
\begin{aligned}
(2\lambda+2)\,q_{IA} &= (2\lambda/3)\,q_{TX} + && q_{MN}\\
(2\lambda+2)\,q_{AR} &= (2\lambda/3)\,q_{LA} + && q_{MT} + q_{OH}\\
(2\lambda+2)\,q_{MO} &= (\lambda/3)[2q_{NM} + q_{AL}] + && q_{OR} + q_{OH} + q_{NY} + q_{PA}\\
(2\lambda+2)\,q_{IL} &= (4\lambda/3)\,q_{NM} + && q_{NY} + q_{PA}\\
(2\lambda+2)\,q_{MS} &= && q_{NY}\\
(2\lambda+2)\,q_{IN} &= (\lambda/3)[q_{TX} + q_{LA} + q_{AL}] + && q_{ND} + q_{NJ}\\
(2\lambda+2)\,q_{TN} &= && q_{MI}\\
(2\lambda+2)\,q_{KY} &= (2\lambda/3)\,q_{AL} + && q_{PA} + q_{MD}\\
(2\lambda+2)\,q_{WV} &= (\lambda/3)[2q_{LA} + 2q_{AL} + 4q_{GA}] + && q_{SD} + q_{MD} + q_{NJ} + q_{DE}\\
(2\lambda+2)\,q_{VA} &= (2\lambda/3)\,q_{TX} + && q_{WI} + q_{NJ}\\
(2\lambda+2)\,q_{SC} &= (2\lambda/3)\,q_{GA} + && q_{DE}\\
(2\lambda+2)\,q_{NC} &= (\lambda/3)[q_{AL} + q_{TX} + q_{LA} + 2q_{GA}] + && q_{MD} + q_{DE}
\end{aligned}
$$

$$
\begin{aligned}
(\lambda+3)\,q_{AK} &= (\lambda/2)[4q_{HI} + 2q_{NV} + q_{UT}] + && 2q_{ME}\\
(\lambda+3)\,q_{OR} &= (\lambda/2)[2q_{CA} + q_{WY} + 2q_{MO}] + && q_{NH}\\
(\lambda+3)\,q_{WA} &= (\lambda/2)[2q_{CO} + q_{NE}] + && 2q_{VT}\\
(\lambda+3)\,q_{ID} &= (\lambda/2)[2q_{CA} + q_{KS}] + && q_{NH}\\
(\lambda+3)\,q_{MT} &= (\lambda/2)[q_{WY} + q_{AR}] + && q_{NH}\\
(\lambda+3)\,q_{ND} &= (\lambda/2)[q_{WY} + q_{UT} + q_{KS} + q_{NE} + 2q_{IN} + q_{IL}] + && q_{MA}\\
(\lambda+3)\,q_{SD} &= (\lambda/2)[q_{WY} + q_{NE} + q_{KS} + 4q_{OK} + q_{IL} + q_{WV}] + && q_{MA}\\
(\lambda+3)\,q_{MN} &= (\lambda/2)[2q_{NV} + q_{UT} + 2q_{IA}] + && 2q_{ME}\\
(\lambda+3)\,q_{WI} &= (\lambda/2)[q_{UT} + q_{VA}] + && q_{MA}\\
(\lambda+3)\,q_{MI} &= (\lambda/2)[q_{NE} + 2q_{CO} + 2q_{TN}] + && 2q_{VT}\\
(\lambda+3)\,q_{OH} &= (\lambda/2)[q_{KS} + q_{AR}] + && q_{NH}\\
(\lambda+3)\,q_{NY} &= (\lambda/2)[q_{IL} + 4q_{MS}] + && 2q_{CT}\\
(\lambda+3)\,q_{PA} &= (\lambda/2)[q_{IL} + 2q_{MO} + 2q_{KY}] + && 2q_{CT}\\
(\lambda+3)\,q_{MD} &= (\lambda/2)[2q_{IA} + 2q_{AR} + 2q_{KY} + 2q_{TN} + q_{WV} + q_{VA} + 2q_{NC}] + && 2q_{RI}\\
(\lambda+3)\,q_{NJ} &= (\lambda/2)[2q_{IN} + q_{WV} + q_{VA}] + && q_{MA}\\
(\lambda+3)\,q_{DE} &= (\lambda/2)[q_{WV} + q_{VA} + 4q_{SC} + 2q_{NC}] + && 2q_{RI}
\end{aligned}
$$

$$
\begin{aligned}
4q_{ME} &= \lambda[q_{AK} + q_{MN}]\\
4q_{VT} &= \lambda[q_{WA} + q_{MI}]\\
4q_{NH} &= \lambda[q_{ID} + q_{MT} + q_{OR} + q_{OH}]\\
4q_{MA} &= \lambda[q_{ND} + q_{SD} + q_{WI} + q_{NJ}]\\
4q_{CT} &= \lambda[q_{NY} + q_{PA}]\\
4q_{RI} &= \lambda[q_{MD} + q_{DE}]
\end{aligned}
$$

APPENDIX B

Approximate Probabilities for the 2121 Module

$$q_{FL} = 1/(1+\lambda)^4$$

$$
\begin{aligned}
q_{AZ} &= (2.8\lambda^2 + 1.1\lambda)/(3\lambda+1)(1+\lambda)^4\\
q_{NM} &= (1.2\lambda^2 - 0.3\lambda)/(3\lambda+1)(1+\lambda)^4
\end{aligned}
$$

$$q_{TX} = (1.4\lambda^2) /(3\lambda + 1)(1 + \lambda)^4$$
$$q_{LA} = (0.9\lambda^2 - 0.5\lambda) /(3\lambda + 1)(1 + \lambda)^4$$
$$q_{AL} = (3.6\lambda^2 + 2.4\lambda) /(3\lambda + 1)(1 + \lambda)^4$$
$$q_{GA} = (2.1\lambda^2 + 1.3\lambda) /(3\lambda + 1)(1 + \lambda)^4$$

$$q_{HI} = (0.5\lambda^3 + 0.6\lambda^2) /(2\lambda + 2)(1 + \lambda)^4$$
$$q_{CA} = (0.4\lambda^3 + 0.3\lambda^2 - 0.1\lambda)/(2\lambda + 2)(1 + \lambda)^4$$
$$q_{NV} = (0.6\lambda^3 + 0.8\lambda^2) /(2\lambda + 2)(1 + \lambda)^4$$
$$q_{UT} = (1.0\lambda^3 + 0.4\lambda^2 - 0.1\lambda)/(2\lambda + 2)(1 + \lambda)^4$$
$$q_{WY} = (1.4\lambda^3 + 2.0\lambda^2 + 0.3\lambda)/(2\lambda + 2)(1 + \lambda)^4$$
$$q_{CO} = (0.6\lambda^3 + 0.1\lambda^2) /(2\lambda + 2)(1 + \lambda)^4$$
$$q_{NE} = (0.3\lambda^3 + 0.2\lambda^2 - 0.1\lambda)/(2\lambda + 2)(1 + \lambda)^4$$
$$q_{KS} = (0.1\lambda^3) /(2\lambda + 2)(1 + \lambda)^4$$
$$q_{OK} = (0.6\lambda^3 + 0.7\lambda^2) /(2\lambda + 2)(1 + \lambda)^4$$
$$q_{IA} = (0.2\lambda^3 + 0.3\lambda^2 - 0.1\lambda)/(2\lambda + 2)(1 + \lambda)^4$$
$$q_{AR} = (0.2\lambda^3 + 0.2\lambda^2 - 0.1\lambda)/(2\lambda + 2)(1 + \lambda)^4$$
$$q_{MO} = (0.8\lambda^3 + 0.7\lambda^2) /(2\lambda + 2)(1 + \lambda)^4$$
$$q_{IL} = (0.3\lambda^3 + 0.5\lambda^2 - 0.2\lambda)/(2\lambda + 2)(1 + \lambda)^4$$
$$q_{MS} = (0.1\lambda^3) /(2\lambda + 2)(1 + \lambda)^4$$
$$q_{IN} = (0.6\lambda^3 + 0.7\lambda^2) /(2\lambda + 2)(1 + \lambda)^4$$
$$q_{TN} = (0.1\lambda^3) /(2\lambda + 2)(1 + \lambda)^4$$
$$q_{KY} = (0.8\lambda^3 + 0.8\lambda^2 + 0.2\lambda)/(2\lambda + 2)(1 + \lambda)^4$$
$$q_{WV} = (1.7\lambda^3 + 1.9\lambda^2 + 0.3\lambda)/(2\lambda + 2)(1 + \lambda)^4$$
$$q_{VA} = (0.3\lambda^3 + 0.3\lambda^2 - 0.1\lambda)/(2\lambda + 2)(1 + \lambda)^4$$
$$q_{SC} = (0.4\lambda^3 + 0.5\lambda^2 + 0.1\lambda)/(2\lambda + 2)(1 + \lambda)^4$$
$$q_{NC} = (0.9\lambda^3 + 1.1\lambda^2 + 0.1\lambda)/(2\lambda + 2)(1 + \lambda)^4$$

$$q_{AK} = (0.3\lambda^4 + 1.0\lambda^3)/(\lambda + 3)(1 + \lambda)^4$$
$$q_{OR} = (0.1\lambda^4 + 0.9\lambda^3)/(\lambda + 3)(1 + \lambda)^4$$
$$q_{WA} = (0.2\lambda^4 + 0.4\lambda^3)/(\lambda + 3)(1 + \lambda)^4$$
$$q_{ID} = (0.1\lambda^4 + 0.2\lambda^3)/(\lambda + 3)(1 + \lambda)^4$$
$$q_{MT} = (0.1\lambda^4 + 0.3\lambda^3)/(\lambda + 3)(1 + \lambda)^4$$
$$q_{ND} = (0.3\lambda^4 + 1.1\lambda^3)/(\lambda + 3)(1 + \lambda)^4$$
$$q_{SD} = (0.3\lambda^4 + 1.5\lambda^3)/(\lambda + 3)(1 + \lambda)^4$$
$$q_{MN} = (0.3\lambda^4 + 0.6\lambda^3)/(\lambda + 3)(1 + \lambda)^4$$
$$q_{WI} = (0.3\lambda^4 + 0.3\lambda^3)/(\lambda + 3)(1 + \lambda)^4$$
$$q_{MI} = (0.2\lambda^4 + 0.5\lambda^3)/(\lambda + 3)(1 + \lambda)^4$$
$$q_{OH} = (0.1\lambda^4 + 0.1\lambda^3)/(\lambda + 3)(1 + \lambda)^4$$
$$q_{NY} = (0.2\lambda^4 + 0.4\lambda^3)/(\lambda + 3)(1 + \lambda)^4$$
$$q_{PA} = (0.2\lambda^4 + 0.9\lambda^3)/(\lambda + 3)(1 + \lambda)^4$$
$$q_{ND} = (0.5\lambda^4 + 1.7\lambda^3)/(\lambda + 3)(1 + \lambda)^4$$
$$q_{NJ} = (0.3\lambda^4 + 0.7\lambda^3)/(\lambda + 3)(1 + \lambda)^4$$
$$q_{DE} = (0.5\lambda^4 + 1.3\lambda^3)/(\lambda + 3)(1 + \lambda)^4$$

$$q_{ME} = (0.1\lambda^4)/(1+\lambda)^4$$
$$q_{VT} = (0.1\lambda^4)/(1+\lambda)^4$$
$$q_{NH} = (0.2\lambda^4)/(1+\lambda)^4$$
$$q_{MA} = (0.3\lambda^4)/(1+\lambda)^4$$
$$q_{CT} = (0.1\lambda^4)/(1+\lambda)^4$$
$$q_{RI} = (0.2\lambda^4)/(1+\lambda)^4$$

AUTHOR

Richard A. Thompson, B. S. (Electrical Engineering), 1964, Lafayette College; M. S. (Electrical Engineering), 1966, Columbia University; Ph.D. (Computer Science), 1971, University of Connecticut; AT&T Bell Laboratories, 1963–1968, 1977—. From 1963 to 1968 Mr. Thompson was involved in switching systems development at AT&T Bell Laboratories. He was a member of the Electrical Engineering Department at Virginia Polytechnic Institute from 1971 to 1977, achieving the rank of Associate Professor. Since 1977 Mr. Thompson has been a member of the Digital Systems Research Department at AT&T Bell Laboratories, with a brief stint in ABI/ATTCP in 1983. His research interests are probabilistic formal languages, fault tolerance and cellular automata, terminals and the human-machine interface, communications switching systems, and, most recently, photonic switching. Mr. Thompson is an active participant in IEEE Computer and Communications Societies. Senior Member, IEEE.

AT&T Technical Journal
Vol. 64, No. 10, December 1985
Printed in U.S.A.

Union Bounds on Viterbi Algorithm Performance

By W. TURIN*

(Manuscript received January 30, 1985)

In the present paper we use transform methods (characteristic function techniques) and contour integrals to derive a closed-form expression for the performance union bound of a general discrete-time system. We show that previously published results may be derived as particular cases of the general formulation developed in this paper. It is well known that the maximum-likelihood Viterbi algorithm may be employed not only for decoding of convolutional codes but also for optimal detection in other situations. Examples include bandwidth-efficient demodulation, optimal accommodation for intersymbol interference and cross-channel coupling, text recognition, simultaneous carrier phase recovery and data demodulation, digital magnetic recording, nonlinear estimation and smoothing. The union bound is a useful measure of the performance of the Viterbi algorithm. Past closed-form expressions for the union bound have usually involved considerable approximation.

I. INTRODUCTION

A Maximum-Likelihood Receiver (MLR) is optimal when the input signal is distorted not only by noise but also by some deterministic factors. The MLR compares the received signal with all possible signals distorted by the same deterministic factors but not by the noise. The latter comparison signals must be available at the receiver. Possible deterministic impairments include intersymbol interference, cross-channel coupling, modem-implementation errors, channel misequalization, signal distortion by the channel nonlinearities, etc. Guided by some metric, the MLR searches for the comparison signal

* AT&T Bell Laboratories.

that is closest to the signal actually received and asserts that this signal, as it existed before being subject to the deterministic impairment, was the transmitted message.

The MLR performance depends on how well we can model the deterministic distortions of the signal and also on distances between signals. The signal separation may be increased by coding.

Technical problems arise in MLR implementation. Strictly speaking, we need to store the whole transmission history and generate all possible comparison sequences. However, if the system may be modeled by a Markov process, then the Viterbi algorithm may be used to realize a recursive MLR.[1]

In designing the MLR it is very important to be able to accurately evaluate the receiver performance. Because of a very large number of possible comparison signals, it is difficult to find the exact formula for an MLR performance characteristic. The performance characteristic upper bound (so-called union bound) is more easily found. Viterbi introduced transfer function techniques to evaluate the union bound for some performance characteristics of binary convolutional codes.[1] These methods have been extended to obtain performance bounds of the general finite-state system.[2] However, the original union bound was loosened to simplify a series summation. Using the transform methods developed in this paper, the original union bound is expressed in closed form.

II. SYSTEM MODEL

Consider a discrete-time system[2]

$$\begin{aligned} x_k &= f(w_k), \\ s_{k+1} &= g(w_k), \\ w_k &= (u_k, s_k), \quad -\infty < k < \infty, \end{aligned} \tag{1}$$

where u_k is a source symbol, x_k is the transmitted channel symbol, and s_k is the corresponding system (transmitter) state. Symbols x_k are transmitted over a noisy memoryless channel that outputs symbols

$$y_k = h(x_k, n_k), \tag{2}$$

where n_k are independent identically distributed variables. The receiver outputs symbols $\hat{w}_k = (\hat{s}_k, \hat{u}_k)$ which minimize the sum

$$M(y, w) = \sum_k m(y_k, w_k),$$

where $m(y_k, w_k)$ is the so-called branch metric and may be treated as a cost function for making a decision that the transmitter superstate

was w_k if y_k was received. This metric is usually a measure of the signal degradation due to noise. For example,

$$m(y_k, w_k) = \ln \Pr\{y_k/f(w_k)\}$$
for the maximum-likelihood receiver,

$$m(y_k, w_k) = \ln \Pr\{u_k\}\Pr\{y_k/f(w_k)\}$$
for the maximum a posteriori receiver,

$$m(y_k, w_k) = -\|y_k - f(w_k)\|^2$$
for the minimum mean-square receiver.

The optimal solution may be found using the Viterbi algorithm. A sequence $\hat{w}_0, \hat{w}_1, \cdots, \hat{w}_{j-1}, w_j$, which terminates at time j, is called a survivor if it minimizes the metric accumulated to this time:

$$\sum_{k=0}^{j} m(y_k, \hat{w}_k) = \min_{(w_0,\cdots,w_{j-1})} \sum_{k=0}^{j} m(y_k, w_k),$$

where $w_j = \hat{w}_j$. It is obvious that the sequence that minimizes the total sum $M(y, w)$ must begin with one of the survivors. The survivors may be determined recursively via the Viterbi algorithm: $\hat{w}_0, \hat{w}_1, \cdots, \hat{w}_j, w_{j+1}$ is a survivor if and only if $\hat{w}_0, \hat{w}_1, \cdots, \hat{w}_{j-1}, w_j$ is a survivor and $\hat{w}_j$ satisfies the equation:

$$\sum_{k=0}^{j+1} m(y_k, \hat{w}_k) = \min_{wj} \sum_{k=0}^{j+1} m(y_k, w_k),$$

where $w_0 = \hat{w}_0, \cdots, w_{j-1} = \hat{w}_{j-1}$.

If all the survivors have a common part, then this part also belongs to the sequence that minimizes the total metric $M(y, w)$ and the corresponding symbols are output by the Viterbi receiver.

Depending on the application, we may want to evaluate the Viterbi receiver performance using some distortion measure $\bar{d} = \mathbf{E}\{d(w_k, \hat{w}_k)\}$, where $d(w_k, \hat{w}_k)$ is the distortion characteristic of the symbol w_k which the receiver identifies as $\hat{w}_k$. For example, if we wish to find a symbol error probability, then the distortion characteristic $d(w_k, \hat{w}_k)$ is equal to zero if there are no errors in the symbol ($u_k = \hat{u}_k$) and is equal to one otherwise. If we wish to evaluate a bit error probability, then $d(w_k, \hat{w}_k) = m_k/b$, where m_k is the number of bit errors in the symbol $\hat{u}_k$ (the Hamming distance between u_k and $\hat{u}_k$), and b is the total number of bits in the symbol.

III. PERFORMANCE UNION BOUND

In order to find the distortion measure union bound, consider an error event of length L (see Ref. 1) that is a pair of the correct path

$\sigma_L = \{w_k\}$ and an incorrect path $\hat{\sigma}_L = \{\hat{w}_k\}$ such that $s_k \neq \hat{s}_k$ for $k = 1, 2, \cdots, L-1$ and $s_k = \hat{s}_k$ otherwise.

Then the distortion measure is upper bounded by the union bound:[2]

$$\bar{d} \leq \mathbf{E}_u \sum_{L=1}^{\infty} \sum_{\hat{\sigma}_L} P_L(\hat{\sigma}_L/\sigma_L) d_L(\sigma_L, \hat{\sigma}_L),$$

where the average is taken over all source sequences, $P_L(\hat{\sigma}_L/\sigma_L)$ is the conditional probability that the incorrect path $\hat{\sigma}_L$ has a larger metric than the correct path σ_L and 0.5 the probability of equality of metrics if a tie is resolved randomly:

$$P_L(\hat{\sigma}_L/\sigma_L) = \Pr\left\{\sum_{k=0}^{L-1} \Delta m(y_k, w_k, \hat{w}_k) > 0\right\} + 0.5 \Pr\left\{\sum_{k=0}^{L-1} \Delta m(y_k, w_k, \hat{w}_k) = 0\right\}, \tag{3}$$

$$\Delta m(y_k, w_k, \hat{w}_k) = m(y_k, \hat{w}_k) - m(y_k, w_k),$$

$d_L(\sigma_L, \hat{\sigma}_L)$ is the total distortion along the incorrect path:

$$d_L(\sigma_L, \hat{\sigma}_L) = \sum_{k=0}^{L-1} d(w_k, \hat{w}_k).$$

Consider first a discrete memoryless channel. Define a generating function of a variable $\Delta m(y_k, w_k, \hat{w}_k)$:

$$D(z; w_k, \hat{w}_k) = \sum_{y_k} \Pr\{y_k/w_k\} z^{\Delta m(y_k, w_k, \hat{w}_k)}. \tag{4}$$

The generating function of the sum of variables $\Delta m(y_k, w_k, \hat{w}_k)$ is equal to the product of the generating functions (4), and the probability $P_L(\hat{\sigma}_L/\sigma_L)$ is equal to the sum of the coefficients of the positive power terms of the product and one half of the zero power term. Using contour integrals, we may express the sum as

$$P_L(\hat{\sigma}_L/\sigma_L) = \frac{1}{2\pi j} \int_{|s|=\rho} \left(\frac{1}{z-1} - \frac{1}{2z}\right) \prod_{k=0}^{L-1} D(z; w_k, \hat{w}_k) dz, \tag{5}$$

where $\rho > 1$. The total distortion along the incorrect path $\hat{\sigma}_L$ may also be expressed using generating functions:

$$\frac{d}{dz} \prod_{k=0}^{L-1} z^{d(w_k, \hat{w}_k)}\Big|_{z=1}, \tag{6}$$

and therefore the average distortion is bounded by

$$\bar{d} \leq \frac{1}{2\pi j} \frac{d}{dz} \left\{\int_C \left(\frac{1}{v-1} - \frac{1}{2v}\right) G(z, v) dv\right\}\Big|_{s=1}, \tag{7}$$

where

$$G(z, v) = \mathbf{E}_u \sum_{L=1}^{\infty} \sum_{\hat{\sigma}_L} \prod_{k=0}^{L-1} D(v; w_k, \hat{w}_k) z^{d(w_k, \hat{w}_k)}, \tag{8}$$

$C \in \{v: R_G \cap |v| > 1\}$, R_G is the region of convergence of (8). The right-hand side of the inequality (7) is a new expression of the union bound of the average distortion $\bar{d}$.

The generating function $G(z, v)$ may be found from the system state transition graph with branch weights

$$D(z; w_k, \hat{w}_k) z^{d(w_k, \hat{w}_k)} \tag{9}$$

or the corresponding matrix equation.[1-3] The system symmetry simplifies the construction of the generating function.

In the case of a continuous channel we may also use a transform (characteristic function) technique to obtain a union bound similar to (7). For example, if $h(x, n) = x + n$, where n is a zero-mean Gaussian variable with one-sided spectral density N_0 (AWGN) and Euclidean metric is used by a Viterbi algorithm, then (3) takes on the form

$$P_L(\hat{\sigma}_L/\sigma_L) = 0.5 \operatorname{erfc}(\beta\delta), \tag{10}$$

where $\beta^2 = E_s/N_0$ is the signal-to-noise ratio,

$$\delta^2 = \sum_{k=0}^{L-1} \| f(\hat{w}_k) - f(w_k) \|^2. \tag{11}$$

Using Laplace transform,[4]

$$\int_a^{\infty} \frac{e^{-pt} dt}{t\sqrt{t-a}} = \frac{\pi}{\sqrt{a}} \operatorname{erfc}(\sqrt{ap}),$$

which after substitutions $\sqrt{a} = \beta$, $\sqrt{p} = \delta$, $\sqrt{t-a} = v$, takes on the form

$$\operatorname{erfc}(\beta\delta) = \frac{2\beta}{\pi} \int_0^{\infty} \frac{1}{(\beta^2 + v^2)} e^{-\delta^2(\beta^2+v^2)} dv.$$

We obtain from (10) and (11)

$$P_L(\hat{\sigma}_L/\sigma_L) = \frac{\beta}{\pi} \int_0^{\infty} \frac{1}{(\beta^2 + v^2)} \prod_{k=0}^{L-1} D_1(v; w_k, \hat{w}_k) dv, \tag{12}$$

where

$$D_1(v; w_k, \hat{w}_k) = e^{-\| f(\hat{w}_k) - f(w_k) \|^2 (\beta^2 + v^2)}. \tag{13}$$

Equation (12) is similar to (5), and therefore we may derive a new union bound for the analog case that is similar to (7):

$$\bar{d} \le \frac{\beta}{\pi} \frac{d}{dz} \left\{ \int_0^\infty \frac{1}{(\beta^2 + v^2)} G_1(z, v) dv \right\} \Bigg|_{z=1}, \tag{14}$$

where

$$G_1(z, v) = \mathbf{E}_u \sum_{L=1}^{\infty} \sum_{\hat{\sigma}_L} \prod_{k=0}^{L-1} D_1(v; w_k, \hat{w}_k) z^{d(w_k, \hat{w}_k)}.$$

IV. EXAMPLES

4.1 Hard-decision convolutional code

Consider the binary convolutional code shown in Fig. 1 with generator polynomials $\{1 + D + D^2, 1 + D^2\}$ (Ref. 1), and the binary symmetric channel with the bit error probability $p = 1 - q < 0.5$. For each input bit u_k the encoder generates two channel bits $x_k = (x_k', x_k'')$ and the encoder state is defined by the two input bits $s_k = (u_{k-1}, u_{k-2})$ and therefore $w_k = (u_k, u_{k-1}, u_{k-2})$ is simply the contents of the shift registers. According to Fig. 1 the equation $x_k = f(w_k)$ is equivalent to the following two equations:

$$x_k' = u_k + u_{k-1} + u_{k-2}$$

$$x_k'' = u_k + u_{k-2},$$

where "+" denotes *mod*2 addition (exclusive-or).

The second equation of the system (1) $s_{k+1} = g(w_k)$ may also be expressed as the system

$$s_{k+1}' = u_k$$

$$s_{k+1}'' = u_{k-1}.$$

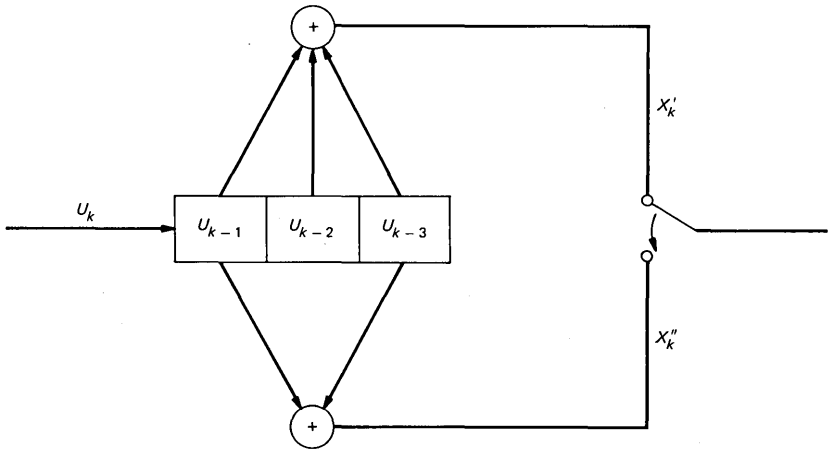


Fig. 1—Convolutional encoder.

However, usually it is represented by the system state graph whose nodes correspond to the system states, and the symbols along the transition lines indicate the encoder input symbols (see Fig. 2).

The next step is to define the algorithm branch metric. Suppose that we received a sequence $y_0, y_1, \cdots, y_N$. Then the MLR will output the sequence $\hat{u}_0, \hat{u}_1, \cdots, \hat{u}_N$, which maximizes the likelihood probability

$$\Pr\{y_0, \cdots, y_N/u_0, \cdots, u_N\} = q^{2N}(p/q)^z,$$

where z is the number of bit errors in the sequence $x_0, x_1, \cdots, x_N$ or, in other words, the Hamming distance (HD) between the sequences: $z = \mathrm{HD}\{(y_0, \cdots, y_N); (x_0, \cdots, x_N)\}$. Since $0 < (p/q) < 1$, maximization of the likelihood is equivalent to minimization of the HD. Therefore we may define the algorithm metric as $m(\hat{w}, w) = \mathrm{HD}\{y; x\}$. This metric depends only on the signal difference.

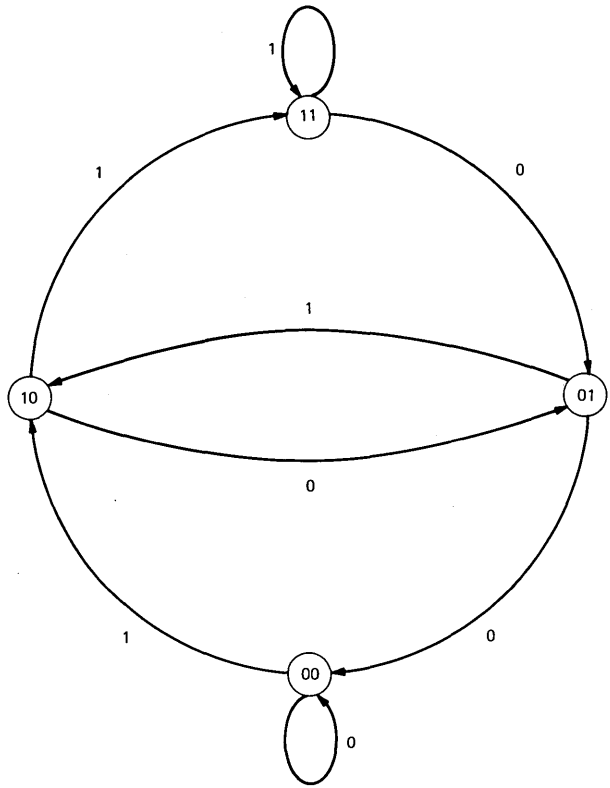


Fig. 2—Encoder state graph.

The channel model may be expressed by eq. (2), which takes the form $y_k = x_k + n_k$, where $n_k = (n_k', n_k'')$ has the following distribution:

$$\Pr\{n_k = (0, 0)\} = p^2, \quad \Pr\{n_k = (1, 0)\} = pq,$$

$$\Pr\{n_k = (0, 1)\} = pq, \quad \Pr\{n_k = (1, 1)\} = q^2.$$

We wish to evaluate the decoder output error probability. Therefore the distortion characteristic is

$$d(\hat{w}, w) = \begin{cases} 0 & \text{if } u = \hat{u} \\ 1 & \text{if } u \neq \hat{u} \end{cases}.$$

This distortion characteristic depends only on the signal difference.

We see that the convolutional code is described in terms of the discrete-time system defined above, and therefore the bit error probability union bound may be found from (7). Because the code is linear, the branch metric and the distortion characteristic depend only on the signal difference. Therefore the generating function (8) is independent of the encoder input sequence and the averaging in (7) is not needed.

We assume that the all-zero sequence is transmitted. The generating function (4) now takes on the form

$$D(v; w, \hat{w}) = (pv + qv^{-1})^{\mu(\hat{w})},$$

where $w = (0, 0, 0)$, $\mu(\hat{w}) = \mathrm{HD}\{(0, 0); \hat{x}\}$ is the Hamming weight of $\hat{x} = f(\hat{w})$. The distortion measure $d(\hat{w}, w) = \hat{u}$; therefore $z^{d(w,\hat{w})} = z^{\hat{u}}$ and

$$D(z; w, \hat{w}) z^{d(w,\hat{w})} = z^{\hat{u}}(pv + qv^{-1})^{\mu(\hat{w})}.$$

If we use these values as branch weights on the system state graph, where the all-zero state is split into an initial and final state[1] as shown in Fig. 3, we obtain

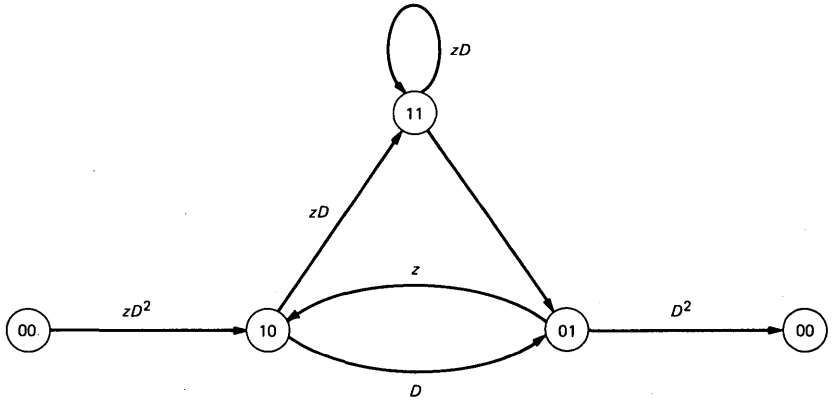


Fig. 3—Signal flow graph of the encoder. $D = (pv + qv^{-1})$.

$$G(z, v) = \frac{z(pv + qv^{-1})^5}{1 - 2z(pv + qv^{-1})},$$

$$|2z(pv + qv^{-1})| < 1$$

as the transition from the initial state to the final state.

The union bound for the bit error probability is found from (7):

$$p_b \le \frac{1}{2\pi j} \int_C \left(\frac{1}{v - 1} - \frac{1}{2v}\right) \frac{(pv + qv^{-1})^5}{[1 - 2(pv + qv^{-1})]^2}\, dv$$

$$C \in \{v: |2(pv + qv^{-1})| < 1 \cap |v| > 1\}.$$

Applying the residue theorem to the previous integral, we obtain

$$p_b \le 2^{-5}[5 - (q - p)(5 - 96pq)(1 - 16pq)^{-1.5} + 14p + 12p^2 - 8p^3],$$

which coincides with the bound derived in Ref. 5.

4.2 Soft-decision convolutional code

In this example we find the union bound on the bit error probability for the same convolutional code but with soft-decision decoding. Suppose that symbols x_k are BPSK modulated. The receive demodulator is followed by a soft-decision sampler with infinite precision.

The likelihood density of receiving $(y_0, \cdots, y_N)$ if $(x_0, \cdots, x_N)$ was transmitted is

$$\psi(y_0, \cdots, y_N/x_0, \cdots, x_N) = \left(\frac{\beta}{\sqrt{\pi}}\right)^{2N} e^{-\delta^2\beta^2},$$

where

$$\delta = \sqrt{\sum_{k=0}^{N} (y_k' - x_k')^2 + (y_k'' - x_k'')^2}$$

is the Euclidean distance (11) between the signals and $\beta^2 = E_s/N_0$ is the signal-to-noise ratio.

It is clear that the maximum likelihood corresponds to the minimum of the Euclidean distance and vice versa. Therefore we may define the MLR metric as $m(\hat{w}, w) = \|x - \hat{x}\|^2 = (x' - x'')^2 + (x'' - \hat{x}')^2$. This metric depends only on the signal difference.

Using the same arguments as in the previous example, we may assume that the all-zero sequence was transmitted. According to (10) and (13)

$$D_1(v; w, \hat{w}) = e^{-(\beta^2 + v^2)\mu(\hat{w})}.$$

The union bound for p_b is found from (14):

$$p_b \le \frac{\beta}{\pi} \int_0^\infty \frac{e^{-5(\beta^2 + v^2)} dv}{(\beta^2 + v^2)[1 - 2e^{-(\beta^2 + v^2)}]^2}.$$

If we use the inequality

$$\frac{1}{1 - 2e^{-(\beta^2+v^2)}} \leq \frac{1}{1 - 2e^{-\beta^2}},$$

we obtain

$$p_b < 0.5(1 - 2e^{-\beta^2})^{-2}\text{erfc}(\beta\sqrt{5}),$$

which coincides with the well-known result.[1] Using the more general inequality

$$\frac{1}{(a+q)^2} \leq \frac{1}{a^2} \sum_{k=0}^{2n} (k+1)\left(\frac{-q}{a}\right)^k,$$

where $a = 1 - 2e^{-\beta^2}$, $(\beta \geq \sqrt{\ln 2})$, and $q = 2e^{-\beta^2}(1 - e^{-v^2})$, we may obtain better approximation. For $n = 2$

$$p_b < 0.5a^{-2}[A\ \text{erfc}(\beta\sqrt{5}) + B\ \text{erfc}(\beta\sqrt{6}) + C\ \text{erfc}(\beta\sqrt{7})],$$

where $A = 1 - 2a^{-1}e^{-\beta^2} + 4a^{-2}e^{-2\beta^2}$, $B = 2a^{-1} - 8a^{-2}e^{-\beta^2}$, and $C = 4a^{-2}$.

V. SUMMARY

Using contour integrals we have derived a closed-form expression of the union bound on Viterbi algorithm performance. The transform methods developed in this paper may be used for some other applications. For example, using eq. (12), a sum

$$S = \sum_k a_k\ \text{erfc}(\beta\sqrt{k})$$

may be expressed as

$$S = \frac{2\beta}{\pi} \int_0^\infty \frac{F(e^{-(\beta^2+v^2)})}{(\beta^2 + v^2)}\, dv,$$

where $F(z) = \sum_k a_k z^k$ is the sequence generating function.

The union bound on the performance characteristic of hard- and soft-decision codes was found as an illustration.

VI. ACKNOWLEDGMENTS

The author wishes to thank D. Leed and the referees for their valuable suggestions and comments.

REFERENCES

1. A. J. Viterbi, "Convolutional Codes and Their Performance in Communications Systems," IEEE Trans. Commun. Tech., *COM-19* (October 1971), pp. 751–72.
2. J. K. Omura, "Performance Bounds for Viterbi Algorithms," ICC'81 Conf. Record, Denver, June 1981, pp. 2.2.1–.5.

3. G. D. Forney, Jr., "Maximum-Likelihood Sequence Estimation of Digital Sequences in the Presence of Intersymbol Interference," IEEE Trans. Inform. Theory., *IT-18* (May 1972), pp. 363–78.
4. F. Oberhettinger and L. Badii, *Tables of Laplace Transform*, New York: Springer-Verlag, 1973, p. 16, eq. 2.34.
5. K. A. Post, "Explicit Evaluation of Viterbi's Union Bounds on Convolutional Code Performance for the Binary Symmetric Channel," IEEE Trans. Inform. Theory., *IT-23* (May 1977), pp. 403–4.

AUTHOR

William Turin, M.S. (Mathematics), 1958, Odessa State University; Ph.D., (Mathematics), 1966, Institute for Problems of Mechanics, Academy of Sciences, USSR; AT&T Bell Laboratories, 1981—. Mr. Turin has worked at New York University and was an Associate Professor at the Moscow Institute of Electrical Engineering and Telecommunications. His research activities include modeling satellite communication channels, simulation, design and analysis of error-correcting codes, and analysis of Markov processes. He is the author of three books. Senior member, IEEE.

AT&T Technical Journal
Vol. 64, No. 10, December 1985
Printed in U.S.A.

A New File Transfer Protocol

By S. AGGARWAL,* K. SABNANI,* and B. GOPINATH†

(Manuscript received November 15, 1984)

We describe a new File Transfer Protocol (FTP) that provides a simple and efficient way of transferring files between heterogeneous systems, such as the *UNIX*™ operating system, Duplex Multi-Environment Real-Time (DMERT), and IBM/MVS. This FTP has been adopted as an AT&T standard. In this protocol, global functions requiring close coordination are separated from local functions. Functions that require global coordination are mandatory parts of the protocol and must be implemented uniformly. Local functions, such as file management and user interface, can be adjusted to local needs and might even be optional. This results in a flexible protocol that can be implemented at various levels of complexity. Thus, FTP implementations can range from very simple ones that provide basic file transfer service to highly complex ones that provide extensive security checking and allow a variety of file management services.

I. INTRODUCTION

File transfers typically represent a large percentage of the traffic volume on data networks. In one internal AT&T network, for example, over 50 percent of the data volume is file transfer traffic. Thus, it is important that file transfers be implemented through an efficient, flexible, and reliable protocol.

In the past, several customized protocols have been developed by various groups within AT&T. Each protocol was designed with only one application in mind, limiting its applicability and functionality. In this paper, we describe a new file transfer protocol that provides

* AT&T Bell Laboratories. † Bell Communications Research, Inc.

the functionality of all file transfer protocols previously used within AT&T. This protocol, called the BX.25 File Transfer Protocol (or simply the FTP), provides a simple and efficient way to transfer files between heterogeneous systems such as the *UNIX* operating system, Duplex Multi-Environment Real-Time (DMERT),[1] IBM/MVS, and UNIVAC/EXEC. This FTP has been adopted as an AT&T standard.[2] Several groups in the company have implemented or are implementing this protocol.[3]

Our approach in designing the FTP was to separate global functions requiring close coordination (such as parameter negotiation) from functions that could be done locally at each individual node. Functions that require global coordination are mandatory parts of the protocol and must be implemented uniformly. Local functions, such as file management and user interface, can be adjusted to local needs and might even be optional. This results in a flexible protocol that can be implemented at various levels of complexity. One can view the FTP as essentially an intelligent bulk data transfer facility.

The FTP is a three-party protocol in which a user at a remote node (the initiator) can transfer files between a source and a destination node. The FTP offers three types of services: Copy, Cancel, and Status. The Copy service allows the transfer of a set of files between the source and destination nodes. The Cancel and Status services, respectively, allow the user to cancel a transfer and request information about a transfer. All services can be done in the background, requiring minimal user supervision.

Our approach in specifying the FTP is consistent with the one currently recommended by the protocol community. This approach recommends that one first describe the services offered by the protocol to the upper layer, as well as the services required by the protocol from the lower layer. Then, it defines the peer-level protocol that fills the gap between the upper and lower layers. Typical protocol specifications, such as levels 2 and 3 of X.25,[4] cover only peer-level message exchange rules and formats.

The application program resident at each node that performs file transfers will be called a File Transfer System (FTS). Functions performed by the FTS cover the application and presentation layer functions as defined in the Open System Interconnection (OSI) model.[5]

As shown in Fig. 1, an FTS has several interfaces: an interface with the user, a data transfer facility interface, and interfaces with the file system and with the operating system. In addition, an FTS communicates with the remote FTSs using a peer-level protocol. The user is a person or a computer program that wants to use the file transfer service. The data transfer facility is the lower level of protocol, such

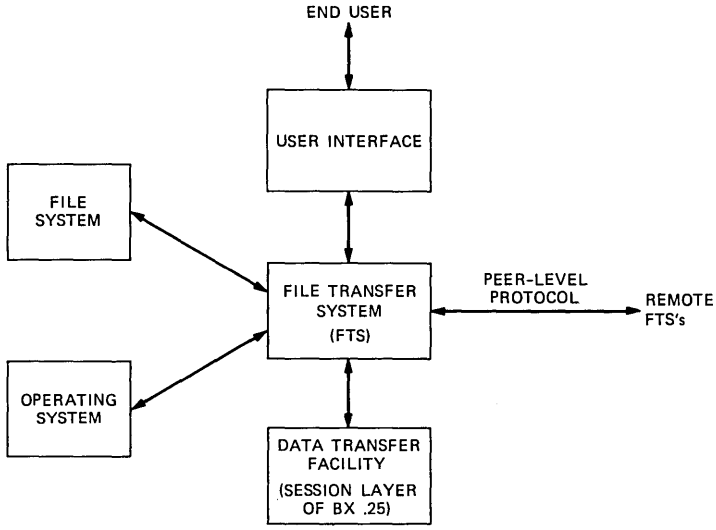


Fig. 1—Interfaces for the file transfer system in any node.

as the BX.25 session layer. The file system is the local storage system for files, and the operating system manages the local computing resources.

The user issues a command to the local FTS to perform a file transfer. The local FTS checks the command for its syntactic correctness. It reads or writes the files being transferred using the local file system. It sends or receives file data to/from a remote FTS through the data transfer facility. The FTS invokes local processing (see pre- and postprocessors below) on the file contents using the operating system interface.

In Section II, we describe our overall design approach in greater detail. We then describe services offered by the FTP in Section III, and identify the services required from the data transport layer in Section IV. The peer-level protocol is described in Section V. Section VI describes the negotiation procedure during which the source FTS and the destination FTS agree on the options for file transfers. Some examples are given in Section VII. We also describe a formal specification of the FTP using the selection/resolution model in Section VIII.[6,7] This specification gives a more complete and precise description of the protocol than an English language specification alone can provide. In Section IX, we compare the FTP with others reported in the literature. Finally, we make some concluding comments in Section X.

II. DESIGN APPROACH

In our design, we have clearly separated functions requiring global

coordination from those that are local to a node. Every implementation would be required to implement the global coordination functions that would form the core of the protocol. Implementors would be free to implement the local functions as needed. Implementations would thus range from very simple ones that provide basic file transfer services to highly complex ones that provide extensive security checking, allow a variety of file management capabilities, and provide a sophisticated user interface. Furthermore, the flexibility in implementing local functions results in a file transfer protocol that works in a heterogeneous environment, is adaptable to diverse local needs, and can evolve as requirements change.

The core of the protocol implements the coordination functions to transfer a byte stream efficiently. Translation between the file formats at the source and the destination is done outside the core of the protocol by local functions called preprocessors and postprocessors.

A preprocessor is a local function at the source that translates the structure and the contents of the file being transferred to a byte stream. Similarly, a postprocessor at the destination translates the byte stream back to the file at the destination. The use of these local functions provides a great deal of flexibility and allows implementations to expand gracefully.

For example, if the initial requirements are file transfers between a *UNIX* system and an IBM/MVS system, we need only write pre- and postprocessors for translation between their file formats. As additional systems, such as a Honeywell computer, are added, we will then add new pre- and postprocessors.

The FTP does not have a networkwide standard for file naming. A user must supply all the naming information required for accessing a remote file. This includes the information about the device name on which the file is stored, as well as the path name of the file. This approach allows us to keep the FTP core small in size.

The FTP uses checkpointing to transfer a byte stream efficiently. During a file transfer, checkpoints are set along the way. After recovery from a transmission failure, the file transfer is resumed from a checkpoint up to which the destination has received the file data correctly. The file transfer does not have to be restarted from the beginning. This feature is useful when transferring very large files such as those stored on several magnetic tapes.

Our security policy makes it difficult for a user to break into the security mechanism of a remote file system. To access a remote file, the user must provide access information required at the remote file system. The user without the proper access information is not allowed to access a remote file. In the following sections, we give further details of the protocol.

III. FTP SERVICE SPECIFICATION

In this section, we describe the services offered by the FTP: Copy, Cancel, and Status. This set of services is sufficient for most file transfer applications. For each service, all the information described here must be provided to the local FTS. We describe here the interaction between a user and an FTS while providing each one of these services.

To invoke a service, the user must issue a command to the local FTS. The syntax of the command is a local decision. However, the information contained in the command is required for global coordination. Thus, in any implementation of the FTP, each command must contain the information given below.

3.1 Copy service

For the Copy service, the command must contain the following information:

COPY ⟨User id⟩ ⟨Source address⟩
⟨File descriptor at the source⟩ ⟨Preprocessors⟩
⟨Destination address⟩
⟨File descriptor at the destination⟩ ⟨Postprocessors⟩
⟨Priority⟩

where

⟨User id⟩ identifies the user initiating the file transfer,

⟨Source address⟩ identifies the computer on which the files to be transferred reside,

⟨File descriptor at the source⟩ is the list of file names to be transferred,

⟨Pre-processors⟩ is a list of programs executed on the files before transfer,

⟨Destination address⟩ identifies the computer to which the files are to be transferred,

⟨File descriptor at the destination⟩ is the list of file names at the destination. The number of files in this list must be the same as that in the list ⟨File descriptor at the source⟩,

⟨Post-processors⟩ is a list of programs executed at the destination on the files transferred,

⟨Priority⟩ specifies the priority level assigned to the copy command. This field specifies how the FTS should treat the present job compared to the other file transfers waiting to be done at the local FTS.

The initiator FTS checks the syntactic correctness of the copy command. The number of file names in the list ⟨File descriptor at the source⟩ must be the same as that in ⟨File descriptor at the destination⟩. If there is an error in the copy command, then the FTS returns

an error message to the user. Otherwise, the FTS returns a job number to the user. The job number identifies a copy command throughout the network and the command's lifetime. The format for a job number ensures that it is unique throughout the network. The job number has the following format:

⟨Initiator address⟩ ⟨Unique number assigned by the initiator⟩

where

⟨Initiator address⟩ specifies the address of the initiator,

⟨Unique number assigned by the initiator FTS⟩ is a number assigned to the command; it uniquely identifies the command at the initiator.

The local FTS must notify the user upon successful or unsuccessful completion of the file transfer.

3.2 Status service

A user at the initiator FTS can inquire about the status of a file transfer request using the following command:

STATUS ⟨Job number⟩.

The status of a file transfer request must provide the user with the following information:

- Whether preprocessing, file transfer, and postprocessing have started, and, if they have, whether they have been completed successfully or unsuccessfully;
- If unsuccessful, the cause of the error.

Additional information, such as percentage of transfer completed, could also be provided to the user. At the initiator, the FTS returns the status described above to the user.

3.3 Cancel service

In a similar manner, a user at the initiator can cancel a file transfer in progress. For such a service, a user issues the following command:

CANCEL ⟨Job number⟩.

The initiator FTS must authenticate the user's identity and check the user's privileges. If the user is not allowed to cancel, then the initiator FTS must return an error message to the user. Otherwise, the file transfer must be canceled. In addition, the state of files modified as a result of the file transfer request must be restored to what it was before the carrying out of the file transfer request. The services described are provided using the peer-level protocol in Section V.

IV. DATA TRANSPORT SERVICES REQUIRED

The FTP requires certain data transport services from the lower

layer. More specifically, it requires a connection-oriented service. In addition, a connectionless service is desirable for short message exchanges. Both data transport services are provided by the BX.25 Session Layer.[4] In this section, we describe these two data-transport services.

4.1 Connection-oriented service

The connection-oriented service must transfer messages of variable length from a transmitter FTS* to a receiver FTS in error-free form and in the same sequence as originally delivered to the transmitter FTS. If it is unable to deliver messages in sequence because of a network failure, it must inform the transmitter FTS. Once a connection is set up between two FTSs, it can be used to transport the files from several file transfer requests. The following service primitives, or their equivalent, must be provided to the FTS:

1. *Connect* (*Destination Address, FTS*). Using the *Connect* primitive, the FTS in the local computer establishes a connection with an FTS resident in the computer with the address *Destination Address.* If the connection is successfully established, the data transport entity returns to the FTS *Connection id,* a unique identifier. The FTS uses *Connection id* later to reference the connection just established. If the data transport entity is unable to establish the connection, it notifies the local FTS about the failure.

2. *Send* (*Connection id, message*). This primitive is used to send a block of data called *message* on the connection *Connection id.*

3. *Receive* (*FTS, message, Connection id*). The local FTS uses the service primitive *Receive* to receive a block of data *message* from the connection *Connection id.*

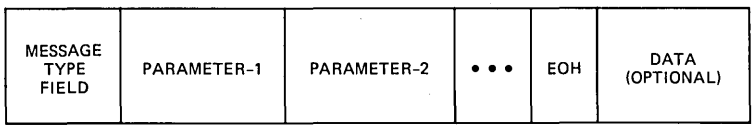
4. *Disconnect* (*Connection id*). The FTS uses the *Disconnect* service primitive to break the connection *Connection id.* The data transport facility delivers any messages in transit before the breakup.

5. *Abort* (*Connection id*). This primitive is used to abnormally break up the connection *Connection id.* The data transport facility does not ensure delivery of the messages in transit before the breakup.

4.2 Connectionless service

A Connectionless service is useful for the transport of single messages between FTSs. It ensures that such message exchanges are done with minimal network resources. Establishment and termination of a connection, such as a BX.25 session, require a significant amount of

* A transmitter FTS is a program that establishes connection with a receiver FTS to transfer a sequence of messages. A transmitter FTS can be either the initiator or the source. Similarly, the receiver FTS can be either the source or the destination.



EOH – END OF HEADER

Fig. 2—Format of a peer-level message.

network resources, which is not justified for short message exchanges. A Connectionless service must provide the following service primitives:

1. *Send* (*Destination Address*, *message*). This primitive is used to send a block of data called *message* to the FTS in the computer with the address *Destination Address*.
2. *Receive* (*message*). The local FTS uses this primitive to receive a block of data called *message* from the data transport entity.

The data transport services described here are used to transport the peer-level messages described in the next section.

V. PEER-LEVEL PROTOCOL

File transfer systems resident in different computers interact with each other using the peer-level protocol described in this section. The peer-level protocol consists of the procedures for message exchange and message formats. The messages are transported using the data transport services described in the previous section.

Since the peer-level protocol gives the rules for interaction between any pair of FTSs, it is the core of the global coordination functions. The rules and the formats specified in this section must be uniformly implemented in *all* FTSs.

5.1 Message formats

All fields in the peer-level messages have been encoded using the International Organization for Standardization (ISO) communication heading format standard.[4,8] Figure 2 shows a general message format. The first field, the message type field, identifies the message type; it is one byte long. Encodings for various message types are given in Table I. The first field is followed by one or more parameter fields, each of which carries a parameter, such as a file descriptor. The first byte uniquely identifies the parameter type, such as file descriptor. Codes for various types of parameters are given in Ref. 2. Each parameter field can be of variable length, with a maximum of 255 bytes. The list of parameters is followed by an end of heading field, H'80'.* The last field contains data for a file data message.

* H stands for hexadecimal. H 'xx' is used to represent a two-digit hexadecimal number.

Table I—Message type codes

Message Type	Message Name	Message Type Code
Command	Authenticated COPY command, CAC	H'24'
	Negotiation package, CNEG	H'31'
	Checkpoint command, CCM	H'32'
	Inquiry about checkpoint command, CIC	H'33'
	Positive confirmation message, CPC	H'34'
	Negative confirmation message, CNC	H'35'
	Inquiry command, CIO	H'36'
	Cancel command, CCC	H'37'
	Interrupt command, CIN	H'38'
	Restart command, CRS	H'39'
Responses	Acknowledgment, RAC	H'41'
	Reply negotiation package, RRN	H'42'
	Checkpoint response, RCH	H'43'
	Status Response, RSR	H'44'
	Cancel Response, RCR	H'45'
	Notification message, RNM	H'46'
Data Messages	File Header Message, DHM	H'01'
	File Data Message, DDM	H'03'

5.2 Rules for message exchanges

The rules for message exchanges for the Copy and Status services are given in Sections 5.2.1 and 5.2.2, respectively. The rules for the Cancel service are similar to those for the Status service and are not given here.

5.2.1 Copy service

For the Copy service, FTSs in the initiator, the source, and the destination go through the following phases:

- Transfer of an authenticated copy command from the initiator FTS to the source FTS (if the initiator FTS is not the same as the source FTS),
- Negotiation phase,
- File transfer phase,
- Notification phase.

After the user issues a copy command, the initiator FTS checks the access privileges of the user initiating the file transfer and the syntactic correctness of the copy command. If the user does not have the

necessary access privileges and the syntax of the copy command is incorrect, the initiator FTS returns an error message to the user and the file transfer is abandoned. Otherwise, the initiator ships the copy command to the source only if the initiator is different from the source in a peer-level message called the Authenticated Copy Command (CAC). After sending the CAC, the initiator FTS waits for an acknowledgment from the source FTS. If it does not receive an acknowledgment within a time-out period FT1, the initiator will again send the message CAC to the initiator. The initiator makes at most A1 attempts at sending the authenticated copy command.

After receiving an authenticated copy command, the source FTS sends an acknowledgment to the initiator FTS and then enters the negotiation phase. During the negotiation phase, the source and destination FTSs negotiate about options for the file transfer, such as selection of pre- and postprocessors, the values of the intercheckpoint interval and the window size for checkpointing. Intercheckpoint interval and window size are defined later in this section. Further details about the negotiation phase are given in Section VI.

If the negotiation phase is successful, the source and destination FTSs enter the file transfer phase. Each file is transferred completely before the transfer of the next file is initiated. The source FTS sends the following sequence of peer-level messages to the destination FTS for each file transmitted:

- File header message: the file header message carries the name of the file in which the file to be transferred will be stored, the file's attributes, such as code set type and file length,
- One or more file data messages (a file data message is a peer-level message that contains a part of the file contents),
- One or more checkpoint commands (a checkpoint command is a peer-level message which marks a checkpoint during a file transfer).

Checkpoint commands are sent between file data messages at every intercheckpoint interval in the file contents. The intercheckpoint interval is the amount of file data sent between two checkpoints. After sending the checkpoint command, the source FTS starts a timer with the time-out period FT3. It keeps on sending additional file data until the number of outstanding checkpoint commands is equal to the window size for checkpointing, defined as the maximum allowed number of outstanding commands. On receiving a checkpoint command, the destination sends an acknowledgment called the checkpoint response to the source.

The checkpoint commands are assigned sequence numbers. Since the sequence numbers cannot be infinitely large, a cyclically reusable sequence numbering scheme is used. We take the sequence range to

be 0 to $(2^{32} - 1)$, where 2^{32} is the size of the sequence space. The number of outstanding checkpoint commands is limited to w, the window size for checkpointing. The checkpoint command and the checkpoint response carry the sequence number of the checkpoint. The source's reception of a checkpoint response with the sequence number x means that the destination FTS has received everything correctly up to the checkpoint assigned the sequence number x.

End of File (EOF) is marked by sending a checkpoint command. The source FTS must receive the checkpoint response to the checkpoint command indicating EOF before it initiates the transfer of the next file. The checkpoint command indicating EOF for the last element in the file transfer request must also indicate whether additional file data will be transferred during the current session.

5.2.2 Status service

Now, we give the rules for the Status service. A user can issue a status command at the initiator FTS to inquire about the status of a file transfer identified by a job number. The initiator FTS checks if it has sent out the authenticated copy command corresponding to the job number given in the status command. If it has not, the initiator FTS informs the user that preprocessing, file transfer, and postprocessing have not started. Otherwise, the initiator FTS sends an inquiry command to the source FTS. The source FTS checks to see if it has completed preprocessing and file transfer. If not, it sends a status message to the initiator FTS, stating the steps it has already taken and the step presently in progress. If the source FTS has already completed preprocessing and file transfer, then it sends an inquiry command to the destination FTS. On receiving the inquiry command, the destination FTS sends a status message indicating whether postprocessing has been completed successfully or unsuccessfully. The source FTS combines this information from the destination FTS with the status information it has about preprocessing and file transfer. The source FTS sends a status message to the initiator FTS. The status message contains the following information:

- Which stages of the copy command (preprocessing, file transfer, and postprocessing) have been completed,
- The nature of the error if one of the stages described above ended in error.

The initiator FTS reports the status information received to the user.

VI. NEGOTIATION PHASE

A file transfer using the FTP has available a choice of several options. For example, checkpointing may or may not be used during

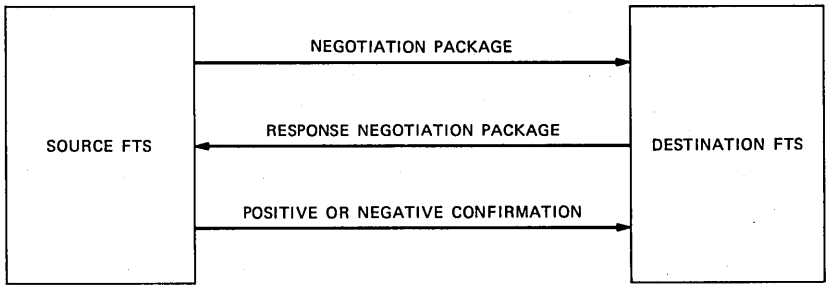


Fig. 3—Message exchange during the negotiation phase.

file transfer; if checkpointing is used during a file transfer, then several parameters, such as the intercheckpoint interval and the window size, must be agreed upon between the source and the destination. These parameters will determine the buffer requirements during the file transfer at the source and the destination. Another parameter to be chosen is the list of postprocessors to be used at the destination.

Before the file transfer phase starts, the source FTS and the destination FTS must negotiate regarding the parameters to be used during the file transfer and come to an agreement. For this negotiation, a three-way message exchange between the source FTS and the destination FTS is used, as shown in Fig. 3.

In the first step, the source FTS sends a negotiation package to the destination and then waits for a reply negotiation package. The negotiation package contains the following information about the parameters to be used during the file transfer:

- Block size. This field specifies the size of the file data transported in one file data message.
- List of postprocessors.
- Whether or not checkpointing is to be used.
- Intercheckpoint interval. This field specifies in units of bytes the amount of file data transmitted between two checkpoints. This field is required only if the checkpointing option is chosen.
- Window size for checkpointing. This field contains the maximum number of outstanding checkpoint intervals at the source FTS. If this field is blank, then the default value is 1.

In the second step, the destination FTS examines the options for file transfer after receiving a negotiation package from the source FTS. The destination FTS decides whether it can accept them. If the destination FTS cannot accept some or all options, it selects alternatives that it can accept. It puts acceptances, rejections, and alternatives in a response negotiation package and sends them to the source FTS. A reply negotiation package contains the following information:

- Specification of rejection or acceptance of each option identified in the negotiation package,
- Suggested alternatives to the rejected options.

In the third step, on receiving a reply negotiation package, the source FTS replies with a positive confirmation message if the destination FTS accepted all options or if the source FTS can accept the alternatives given in the reply negotiation package. Otherwise, the source FTS will send a negative confirmation message to the destination FTS.

If the source FTS does not receive a reply negotiation package from the destination FTS within a time-our period FT2 after sending a negotiation package, it sends the negotiation package again. If the source FTS does not succeed within A2 attempts, it informs the initiator FTS about the failure and aborts the file transfer.

One parameter that the source and destination FTSs might negotiate on is the time when the file transfer should be started. The FTP described here does not negotiate on the starting time of the file transfer, but it can be easily extended to do so. If the underlying network only carries file transfer traffic, then such negotiation can be very useful.[9] Otherwise, we feel that it should not be used for the following reason. Such negotiation requires that the FTSs should be allowed to schedule the future allocation of connections, but this is not consistent with the OSI reference model approach.[5] Several application layer protocols must share the same network resources. Scheduling of the connection allocation, if any, must be done by the lower layers, not by an application layer protocol, such as the FTP.

For networks that carry file transfers exclusively, negotiation about the starting time can be very useful. File transfers typically require a heavy commitment from the network. A connection used for a file transfer typically utilizes nearly 100 percent of its maximum transfer rate as defined by its throughput class. On the other hand, a connection carrying interactive data utilizes only 1 to 2 percent of its maximum transfer rate. As a result, the number of outgoing and incoming connections used for file transfers is very small. Scheduling of connections and file transfers to optimize a cost function, such as network utilization, or minimizing delay can be very useful.

VII. EXAMPLES

We give two examples to illustrate the use of the FTP: one for the Copy service and one for the Status service.

In the first example (Fig. 4), a user issues a copy command to the file transfer system in Computer-C to copy a file FSOU from Computer-A to the file FDEST in Computer-B. The file transfer system in Computer-C (henceforth, referred to as FTS-C) checks the com-

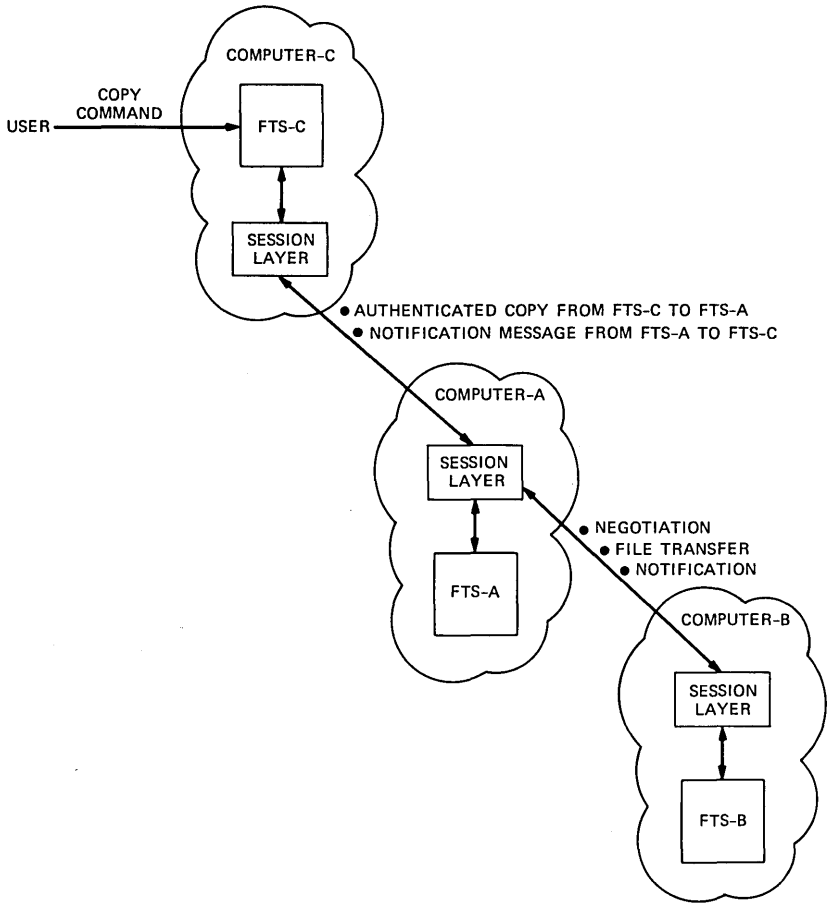


Fig. 4—An example of the Copy service.

mand for its syntactic correctness and checks the access privileges of the user to find out whether the user is allowed to initiate this file transfer.

If the copy command has a syntax error, then FTS-C returns an error message to the user. If the user has the proper access privileges, then FTS-C returns a job number to the user. The job number is a unique identifier assigned to the copy command. It uniquely identifies the command throughout the network. The format for the job number is given earlier in Section III.

FTS-C sends an authenticated copy command to the FTS in Computer-A (henceforth, referred to as FTS-A). Then, it waits for an acknowledgment from FTS-A. If FTS-C does not receive an acknowledgment within the time-out period FT1, then it retransmits the

authenticated copy command. It makes at most A1 transmission attempts.

On receiving the authenticated copy command from FTS-C, FTS-A negotiates with FTS-B for the options of file transfer such as choice of pre- and postprocessors, the checkpointing option, and the parameters for checkpointing. If the negotiation is successful, then the actions described below take place. Otherwise, FTS-A sends an error message to FTS-C and no further operation takes place.

FTS-A sends a file header message to FTS-B. A file header message contains the information such as the file length and the file name. Then, FTS-A processes the file data using the preprocessors and divides it into blocks. Each block is packed into a separate file data message. Then, FTS-A sends these file data messages to FTS-B. It also sends the checkpoint commands at every intercheckpoint interval. If the session being used for the file transfer breaks down, then FTS-A can resume file transfer from an intermediate checkpoint up to which FTS-B has received the data correctly. FTS-B unpacks the file data in the messages received. Then, it processes the file data received using the postprocessors agreed upon during the negotiation phase. FTS-B then stores the file data in the file FDEST.

After the successful file transfer, FTS-B sends a notification message to FTS-A informing it about the successful completion. FTS-A, in turn, sends a notification message to FTS-C informing it about the successful completion. Finally, FTS-C informs the user about the successful completion of the file transfer.

In the second example, we illustrate use of the Status service (Fig. 5). A user issues a status command at the local FTS, FTS-C, to inquire about the status of the copy command described above. The status of a copy command consists of at least the following information:

- Whether preprocessing has begun. If yes, whether it has been completed successfully.
- Similar information about file transfer and postprocessing. The status can further include additional information, such as how far the file transfer has progressed.

FTS-C checks the status command for its syntactic correctness. If the command has a syntax error, then FTS-C returns an error message to the user. Then, FTS-C checks the access privileges of the user to determine whether the user can inquire about the file transfer. If the user has the necessary privileges, then FTS-C checks the local records to find out whether the file transfer has been completed. Otherwise, FTS-C sends a status command to the destination FTS, FTS-B. FTS-B checks the local records to locate the status of the file transfer. It checks whether it was ever involved in that file transfer. If it was involved in the file transfer, FTS-B checks whether the file transfer

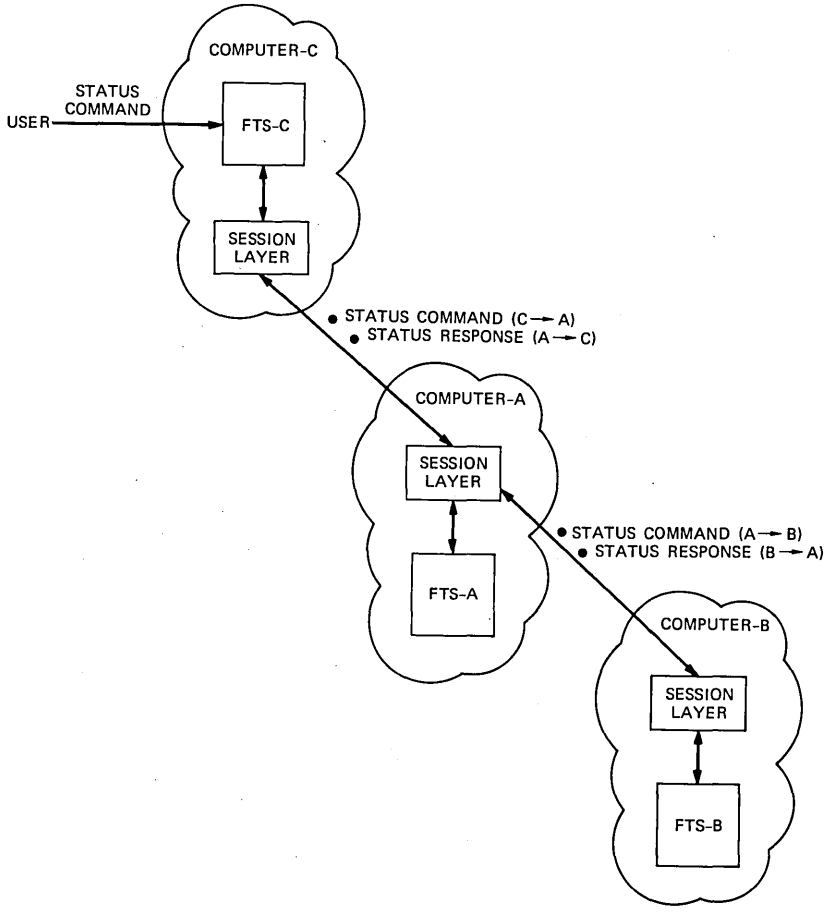


Fig. 5—An example of the Status service.

has been completed or if it is still in progress. FTS-B sends the status information in a status response message to FTS-A. FTS-A also collects the local information about the file transfer and combines it with the information received from the destination FTS, FTS-B. Then, it sends the status information to FTS-C. Finally, FTS-C delivers the status information to the user.

VIII. FORMAL SPECIFICATION OF THE FTP

We have specified the FTP[8] in a mathematically precise notation based on the *selection/resolution* (s/r) model of coordination described in Refs. 6, 7, 10, and 11. Compared with the English language specification, the formal specification provides a more precise and complete

description of the protocol. In this section, we first present an informal discussion of the s/r model; the reader is referred to the references for more details. We then discuss how the FTP was formally specified using this approach.

8.1 The selection/resolution model

The selection/resolution model is a method in which a complex system such as a protocol can be represented as a set of coordinating finite state machines. Each component finite state machine, called a *process*, is described by an appropriately labeled directed graph. For instance, suppose we wished to describe the coordination of two processes A and B that can be executing segments of code in either a noncritical section or a critical section. We wish the coordination to be such that both processes are not simultaneously in their critical sections. Figure 6 has two labeled graphs representing the processes.

In the labeled graphs of Fig. 6, the vertices represent *states* of the process, and the edges represent possible single-step transitions. Thus, process A (also B) can be in the states NCS (Noncritical Section), TRY (trying to enter the critical section), and CS (Critical Section). A state encapsulates the relevant past of the process needed to determine future behavior and is known only to that process. The processes coordinate by exchanging information about their future intentions. That is, they communicate their *selections* (intentions) to all the other processes. For example, in state NCS, process A can only choose the selection *ok*, while in state TRY it can nondeterministically choose between *head* and *tail*. Note that a process can make only one selection at any time. Selections of a process are given in curly braces next to the state.

Each edge has a label next to it. The edge labels are Boolean conditions based on the selections of all the processes, and determine the possible transitions of a process. In conditions, + is the same as a logical *or* and * is the same as a logical *and*. For instance, the condition for process A to go from CS to NCS is simply (*A*:*nok*), that is, it really doesn't depend on B, and is in fact always true since the only selection of A in state CS is *nok*. However, the condition for A to enter the CS state from the state TRY is [(*B*:*ok*) + (*A*:*head*)*(*B*:*head*) + (*A*:*tail*)*(*B*:*tail*)]. In simple words, the condition says that A can go to the CS state if B selects *ok*, or if A and B both select *head*, or if A and B both select *tail*. The transition of process A from state TRY to its next state clearly depends on what B is selecting. Both processes essentially toss coins to determine who wins if they are both trying to enter the critical section. The transition of a process to one of the possible next states (there may be more than one transition enabled) is called *resolution*.

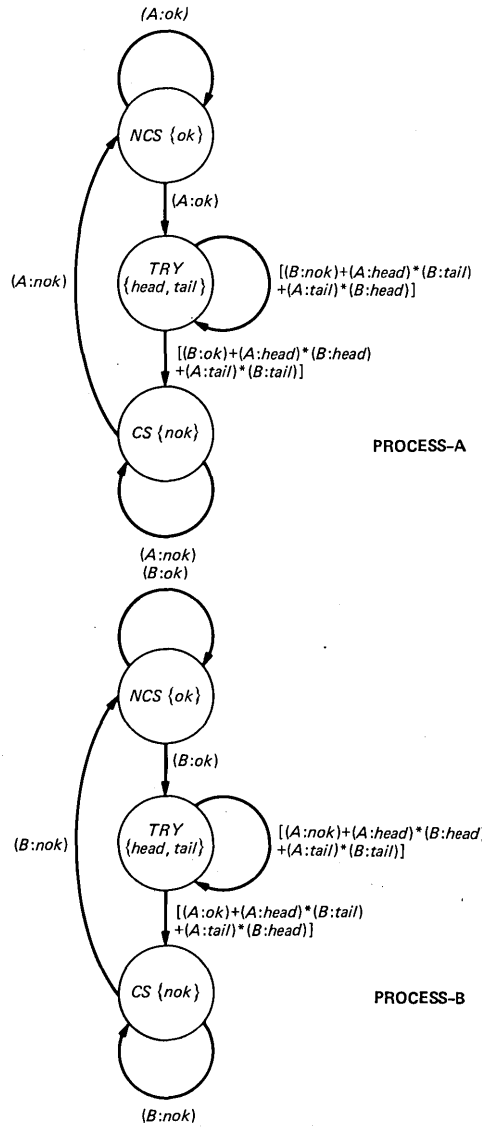


Fig. 6—Mutual exclusion example.

In the above example, developed in more detail in Ref. 11, both processes initially start in the state NCS at time-step 0. Then, in each unit of time, the processes first make a selection based on the state they are in, and then they resolve (transition to a new state) based on the selection of all the processes. It can be seen that the processes will never both be in the states CS at any time step.

The s/r model gives precise algebraic descriptions of processes that can be combined to give a precise description of the entire system. Furthermore, many computational aspects of this model can be automated. In fact, the formal specification can be used to test the correctness of the FTP by conducting a reachability analysis using the procedure given in Refs. 6 and 7.

8.2 Specification of the FTP

The formal specification of the FTP consists of descriptions of 45 processes divided for convenience into 9 clusters (see Fig. 7). A cluster

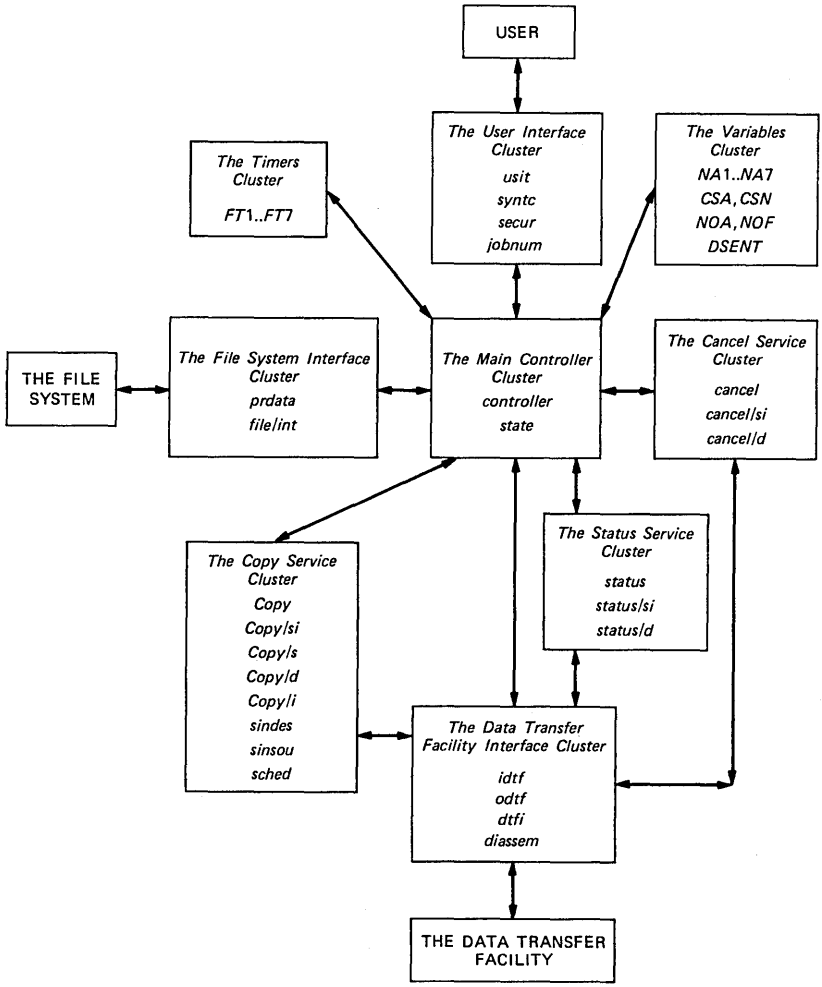


Fig. 7—Block diagram of the formal specification.

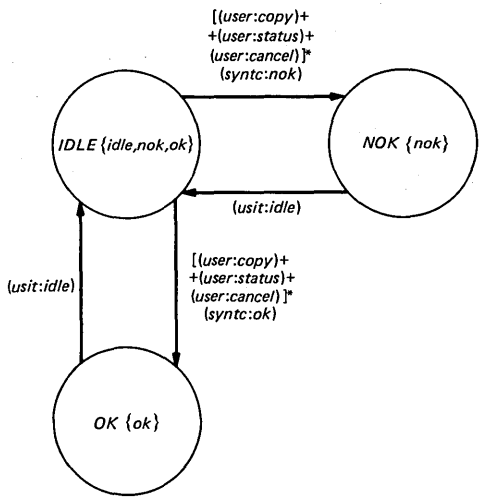


Fig. 8—The Process *syntc*.

is a convenient grouping of a set of processes. We have grouped processes into clusters based on their functions. Thus, there are clusters that correspond to the basic service functions such as Copy service, Status service, and Cancel service. There are also clusters that correspond to interface functions such as interfacing with the user, the data transfer facility, and the file system. Finally, there is a main controller cluster that acts as an overall coordinator, and there are clusters for the timers and the variables.

Each process is defined in a language that essentially provides the same information as the labeled graph corresponding to that process. For example, we can define a process that does a syntax check on the user command. We model this as the process *syntc* shown in Fig. 8. As discussed in Section III, the exact choice of the command syntax is a local decision. Notice that we model only the portion of the process behavior that describes global coordination with other processes. The process *syntc* makes the selection *ok*, only if the command contains all the required information and is syntactically correct. Otherwise, it selects *nok*.

In this figure we have not indicated the conditions for the self-loops.* We shall assume that the self-loop condition is like an *otherwise*; that is, it is the *negation* of the *or* of the conditions on the other edges. The process *syntc* is defined using the specification language in Fig. 9. Notice the close correspondence with the graphical specification. We use $ as shorthand to signify the current state. In this

* A self-loop is a transition from a state back to the same state.

```
    process     syntc/* A process to do syntax check of the user commands */

selvar: valr 0..2
       valnm    idle, ok, nok

stvar: valr 0..2
      valnm     IDLE, OK, NOK

initial condition: IDLE/(syntc:idle)

trans:

IDLE                {idle, nok, ok}

  > OK              :((user:copy) + (user:status) +
                       (user:cancel)) * (syntc:ok))
  > NOK             :((user:copy) + (user:status) +
                      (user:cancel))*(syntc:nok))
  > $                 :;

OK                 {ok}

  > IDLE            :(usit:idle)
  > $                 :;

NOK                {nok}

  > IDLE            :(usit:idle)
  > $                 :;
```

Fig. 9—Specification of Process *syntc*.

example, the empty condition on the self-loop is equivalent to *otherwise*.

Process *syntc* is one of the processes in the User Interface Cluster. This cluster consists of four processes: *usit*, *syntc*, *secur*, and *jobnum*. Each process's description is on the average about as complex as that of *syntc*. The process *usit* gets into one of the three states—COPY, CANCEL, and STATUS—if the user command is syntactically correct and if the user has the permission to execute the command. The process *secur* is similar to *syntc*; it checks the security privileges of the user. It makes the selection (*secur:ok*) if the user can execute the command. Otherwise it makes the selection (*secur:nok*). The process *jobnum* generates a job number for each copy command. The job number uniquely identifies a copy command throughout the network.

For the processes *secur*, *syntc*, and *jobnum*, we model only their synchronization and interaction with other processes. An FTS developer must design the appropriate procedures used in these processes. The interaction, however, should be exactly as described. The English language specification of our FTP does not define what procedures to use to do syntax checking or security checking, etc. Thus, the formal specification must not define the procedures to be used either.

The full formal specification (about 50 pages long) consists of similar descriptions of all the clusters. The descriptions of the processes, of course, vary in complexity. The formal specification can be used as a precise guideline by FTP implementors. In fact, the high-level design

of the FTP implementation may be readily derived from the formal description. A development group at AT&T Communications used the formal specification in their design process. They found it to be extremely useful in developing a high-level C-like language design of FTP.

For implementation, each user command can be viewed as invoking an independent copy of the 45 processes. If an FTS must handle several file transfer commands at the same time, then an independent copy of the processes is invoked for each command. Since our FTP is a three-party protocol, a file transfer at an FTS can be initiated from a remote FTS. For a file transfer initiated from a remote FTS, an independent copy of the processes is invoked at the local FTS. The FTS can thus be implemented as a reentrant program that can be used independently by each command.

IX. COMPARISONS WITH OTHER FILE TRANSFER PROTOCOLS

As discussed previously, our approach here is to separate the file transfer issues from local service functions. This allows great flexibility in local matters and avoids proscribing rigid formats for the user interface, for file management and naming, and for presentation functions such as code sets. Our FTP differs in this regard from other file transfer protocols that generally are rather inflexible and less adaptable. For example, none of the other file transfer protocols has a general mechanism such as pre- and post-processors that allow additional capabilities to be incorporated in a standard way. In this section, we compare the FTP with four file-transfer protocols: Arpanet FTP,[12] Network Independent (NI) FTP,[13-15] Autodin II FTP,[16] and the International Organization for Standardization (ISO) File Service Protocol.[17]

The Arpanet FTP requires the user (initiator) to establish separate command and data connections with both the source and the destination of the file transfer. The user must keep track of these connections. The user is also involved at too detailed a level with the file transfer operation; it is not possible to simply copy a set of files from the source to the destination with a single macro command. It is expected that the user remain on-line at each phase of the transfer, and the user must be aware of details such as the socket number of the data connection. The protocol does provide for both copy and append services, as well as status notification. File management services such as creating and deleting files are also available. Security measures are fixed and encompass only user name, password, and account number. There is no provision for adding other local security options.

The NI FTP is a two-party protocol (no provision is made for third party initiation of a file transfer) that divides the transfer operation into three phases. The protocol handles only transfers of single sequential files; multiple files must be sent as separate file transfers. The NI FTP protocol provides for a high degree of flexibility in the actual data transfer operation. This is accomplished through the negotiation of a large number of possible alternatives. The possible alternatives are defined in any particular implementation, and cannot be readily modified. There is a very limited file management capability in NI FTP. The protocol assumes only minimal support from the lower layers, and specifies presentation services. It allows for modular implementation, but does not have provisions for easily adapting to new requirements. The protocol provides for only foreground file transfers.

The Autodin II FTP provides high-level service primitives and allows background file transfers. However, it goes too deeply into defining the structure of a file at the network level; this results in the negotiation of a large number of parameters for the actual data transfer, rather than allowing this to be handled through local processing. Furthermore, each of the three parties involved in the file transfer must keep an elaborate list of parameters that define all the options that have been agreed upon. There is an extensive security mechanism, but it is not modifiable. Similarly, the user interface is very specific, and cannot be adjusted to local preferences. Autodin II FTP is definitely comprehensive but all the detail is defined at the network level. This results in a protocol that requires a complex implementation at each node.

A committee of the ISO is working on the File Access, File Transfer, and File Management (FTAM) services and on a corresponding protocol. The FTAM services allow a user to access and transfer a remote file. They also allow a user to manipulate a file in a remote file system. An example of a service is a command to open a file in a remote file system. These are micro services. In contrast, our FTP provides macro services, such as the copy service. Each service of our FTP can be provided by a long sequence of the ISO service calls. We have carried out a study to define these sequences of the ISO service calls.[18]

The ISO file service uses the concept of the Virtual File Store. The Virtual File Store is a common model for describing file names and their attributes. Different file systems have a wide range of styles for describing the storage of data and the means by which it can be accessed. The Virtual File Store allows the differences in style and specification to be absorbed in a local mapping function. The Virtual File Store attempts to encompass all possible variations of file defi-

nition, resulting in an excessively detailed description of a file that cannot be easily modified in the future.

X. CONCLUSIONS

In the FTP presented here, we separated global functions requiring close coordination (such as parameter negotiation) from functions that could be done locally at each individual node. In an FTP implementation, the global functions *must* be implemented. Local functions can be implemented depending on user needs. As a result, a minimal implementation of the protocol is fairly simple. Furthermore, it requires only incremental efforts to extend the minimal implementation.

In our approach, we first defined the service specification and the services required by the protocol from the lower layer. Then, we designed the peer-level protocol that fills the gap between the upper and lower layers. Adding service specification leads to a clearer specification of the entire protocol and is useful to implementors. We have also described a formal specification of the protocol that can be used to resolve any ambiguities in the English-language document.

We feel that the FTP is flexible, adaptable, and powerful enough to meet most file transfer requirements. Furthermore, it has been specified precisely enough so as to make implementation a relatively straightforward task.

REFERENCES

1. M. E. Grzelakowski, J. H. Campbell, and M. R. Dubman, "DMERT Operating System," B.S.T.J. *62*, No. 1, Part 2 (January 1983), pp. 303–22.
2. *Operations Systems Network Protocol Specification: BX.25 Issue 3 Addendum-A*, AT&T Bell Laboratories, Holmdel, NJ, August 1983.
3. H. R. Patel and G. S. Lohr, unpublished work.
4. *Operations Systems Network Protocol Specification: BX.25 Issue 3*, AT&T Bell Laboratories, Holmdel, NJ, June 1982.
5. A. S. Tanenbaum, *Computer Networks*, Englewood Cliffs, N.J.: Prentice-Hall, 1981, Chapter 1, pages 10–21.
6. R. P. Kurshan and B. Gopinath, Unpublished work.
7. S. Aggarwal, R. P. Kurshan, and K. Sabnani, "A Calculus for Protocol Specification and Validation." In *Protocol Specification, Testing, and Verification, III*, edited by H. Rudin and C. H. West, Amsterdam: North-Holland, 1983.
8. S. Aggawal and K. Sabnani, Unpublished work.
9. E. G. Coffman, Jr. et al., "Scheduling File Transfers in a Distributed Network," Second Annual Symp. on Principles of Distributed Computing, 1983, pp 254–6.
10. S. Aggarwal, R. P. Kurshan, and D. K. Sharma, "A Language for the Specification and Analysis of Protocols," In *Protocol Specification, Testing, and Verification, III*, edited by H. Rudin and C. H. West, Amsterdam: North-Holland, 1983.
11. S. Aggarwal and C. Courcoubetis, "Distributed Implementation of a Model of Communication and Computation," Proc. 18th Hawaii Int. Conf. on System Sciences, January 1985, pp 206–218.
12. N. J. Neigus, "File Transfer Protocol for the ARPA Network." In *ARPANET Protocol Handbook*, edited by E. Feinler and J. Postel, Defense Communications Agency, 1978.
13. C. J. Bennet and D. N. Frost, "Network Independent File Transfer," Tech. Report, University College, London.
14. R. W. S. Hale, "File Transfer Protocols—Comparison and Critique," NPL Report DNACS 48/81, Teddington, United Kingdom, 1981.

15. High Level Protocols Group, "A Network Independent File Transfer Protocol," HLP/CP(78), NPL, Teddington, United Kingdom, 1977.
16. H. C. Forsdick, "Autodin II File Transfer Protocol," Bolt Beranek and Newman Inc., Report No. 4246, Boston, 1980.
17. *Second Draft Proposal on File Transfer, Access, and Management,* ISO/TC97/SC21 No. 8571, 1985. (Available from ANSI)
18. D. Lewan and K. Sabnani, unpublished work.

AUTHORS

Sudhir Aggarwal, B.S. (Mathematics), 1969, Stanford University; M.S. (Mathematics), 1971, and Ph.D. (Computer and Communication Sciences), 1975, University of Michigan; Assistant Professor of Computer Science, University of Oregon, 1975–1976; Research Scientist, Lawrence Livermore Laboratory, 1976–1977; Assistant and Associate Professor of Mathematics, University of California, Riverside, 1977–1982; AT&T Bell Laboratories, 1982—. Mr. Aggarwal is a member of the Mathematical Sciences Research Center. His research interests are computer communication protocols, local area networks, and modelling and simulation.

B. Gopinath, M.Sc. (Mathematics), 1964, University of Bombay; Ph.D. (E.E.), 1968, Stanford University; Research Associate, Stanford University, 1967–1968; Alexander von Humboldt Research Fellow, University of Gottingen, 1971–1972; Gordon McKay Professor, University of California, Berkeley, 1980–1981; AT&T Bell Laboratories, 1968–1983; Bell Communications Research, 1983—. Mr. Gopinath is Division Manager for Systems Principles Research, and is engaged in research in communications and computer science.

Krishan Sabnani, B. Tech. (E.E.), Indian Institute of Technology, New Delhi; Ph.D. (E.E.), Columbia University, New York, 1982; fellowship, School of Engineering and Applied Sciences, Columbia University, 1977; research assistant, Columbia University, 1978 to 1979; he was employed by RCA, Princeton, 1979–1981, AT&T Bell Laboratories, 1981—. Mr Sabnani's current interests are computer communication protocols and fault-tolerant computing. Member, Sigma Xi, Eta Kappa Nu, and Epsilon Pi Upsilon.

AT&T Technical Journal
Vol. 64, No. 10, December 1985
Printed in U.S.A.

Tracing Protocols

By G. J. HOLZMANN*

(Manuscript received June 12, 1985)

Automated protocol validation tools are by necessity often based on some form of symbolic execution. The complexity of the analysis problem however imposes restrictions on the scope of these tools. The paper studies the nature of these restrictions and explicitly addresses the problem of finding errors in data communication protocols of which the size precludes analysis by traditional means. The protocol tracing method described here allows one to locate design errors in protocols relatively quickly by probing a partial state space. This scatter searching method was implemented in a portable program called Trace. Specifications for the tracer are written in a higher-level language and are compiled into a minimized finite state machine model, which is then used to perform either partial or exhaustive symbolic executions. The user of the tracer can control the scope of each search. The tracer can be used as a fast debugging tool but also, depending on the complexity of the protocol being analyzed, as a slower and rather naive correctness prover. The specifications define the control flow of the protocol and may formalize correctness criteria in assertion primitives.

I. INTRODUCTION

Protocol validation by symbolic execution is inherently a time- and space-consuming task. For lack of better methods, though, many automated protocol validation tools do use symbolic execution algorithms,[1-5] and even methods based on validation algebras such as CCS[6] or PVA[7,8] still implicitly formalize symbolic executions.† Unfortu-

* AT&T Bell Laboratories.

† Cf. the expansion theorem in CCS and the shuffle operator in PVA.

nately, the assumption that a computer will always be able to take over when the complexity of a complete analysis surpasses our ability to perform algebraic expansions by hand is decidedly wrong.[9,10]

A protocol of a realistic size can generate a state space of in the order of 10^9 system states and up. As little as adding one single message type, one protocol variable, or one slot to the message queues can further expand the number of reachable system states by orders of magnitude. For a protocol of this size a symbolic execution algorithm can at best analyze in the order of 10 to 100 system states per second of CPU time if the state space is built in core.[11] To analyze 10^9 states exhaustively would then take at least 115 days of computation. Furthermore, assuming that each state can be encoded in no more than 10 to 100 bytes, storing a state space of this size would still require a machine with several gigabytes of main memory.

So, if this appears to be infeasible, what is the best that can be done? In the design phase one would like to have tools that can trace the most glaring bugs in a protocol in no more than a few seconds of real time. The completeness of an analysis is not really at issue here; *speed* is. To find more subtle design errors of a completed protocol one may be willing to spend more time, but not much more than perhaps 10 hours or in the order of 10^5 seconds of CPU time. For symbolic execution algorithms, this requirement sets an upper limit to the number of states that can be searched at roughly 10^7 states. At 10 to 100 bytes per state, however, we cannot expect to do anything useful with a state space of more than in the order of 10^6 states. Therefore, it is preferable that the tracer be able to perform complete analyses in small state spaces holding just a fraction of the total number of states. In the remainder of this paper we concentrate on these two issues: the effectiveness of partial searches and the possibility of performing complete searches in partial state spaces.

In the following discussion we assume that the protocol submitted to a tracer is likely to contain errors and that a designer is interested in seeing any nonempty subset of these. A protocol tracer may, for instance, scan the state space in an effort to quickly discover typical violations of user-specified correctness requirements. It is important to note that the objective of such a partial analysis, or *scatter search* as we shall call it, is to establish the presence rather than the absence of errors.

What we are aiming for is a protocol tracing method that allows us to spend a small fraction of the time required by an exhaustive analysis to find a substantial portion of all design errors. The emphasis is on speed, not on completeness. If a protocol contains an error, an exhaustive search would meticulously report every possible circumstance

under which the error could make the protocol fail. For our purposes, tracing a single variant of the error in a partial search will suffice.

Section II explains how a general symbolic execution algorithm based on depth-first search can be organized. It discusses a variant of symbolic execution called scatter searching and compares its performance with exhaustive searching. Section III discusses in more detail heuristics that can be used to perform a partial search, and Section IV shows how depth-first searches can be organized in incomplete state spaces. Section V shows how protocol specific correctness criteria can be verified with a standard symbolic execution algorithm. Section VI gives a small example of the use of correctness assertions in tracing bugs in a protocol. A larger example is presented in the Appendix. Section VII summarizes the main results.

II. SCATTER SEARCHING

In this section we discuss some experiments with a program called Trace, which performs a simple depth-first search in a partial state space generated by a set of interacting finite state machines, where the state space is maintained as a tree of system states. To determine the effectiveness of partial searches, the performance of exhaustive searches and scatter searches was compared, using a search depth restriction as a parameter. But, first let us consider the working of the tracer in a little more detail.

With the exhaustive tracing method a state space tree is searched starting from the initial system state, exploring every possible execution path until an end state, a previous state, or an error state is reached, or until the search depth limit is encountered. A return to a previously analyzed state terminates the search under two conditions:

1. If the previously analyzed state is in the execution path that leads from the root of the state space tree to the current state, or
2. If the previously analyzed state was encountered elsewhere in the state space tree either at the same depth or closer to the root of the tree than the current state.

In the first case the tracer has discovered an execution loop in the protocol. The loop could be checked further on liveness, but to save time the program Trace simply checks that its repetition does not violate the user-specified correctness criteria and continues. In the second case the subtree that would be explored by continuing the search down to the search depth restriction would be contained in the subtree of the previously analyzed state, and cannot lead to new results. The tracer can therefore ignore the subtree and continue exploring new leaves in the tree.

Though the state matching method is more general than the one

described in, for instance, Ref. 1, the design of the experimental tracer so far is fairly standard. The exhaustive trace method, however, can be considered to be a special case of the scatter search. In a scatter search not every possible execution sequence is explored. The tracer makes an estimate of the likelihood that exploring a new sequence can lead to the discovery of a new error, and will search only those sequences that optimize its chances of finding the largest set of unique errors in the smallest amount of time. The tracer's estimate will be based on a heuristic that should be general enough to work on any type of protocol. One straightforward way to do this, for instance, is to restrict the amount of nondeterminism that will be taken into account by the tracer. These and other techniques will be discussed in more detail in Section III.

2.1 Test results

To test the performance of a partial search, we want to compare its coverage or "scope" with that of an exhaustive search. The test protocol chosen for these comparisons was large enough to show the necessity of partial searches and also to give some room for experimenting with different flavors of partial searches. However, the size of the test case precluded, by the nature of the problem, the compilation of a definitive list of "all" errors for reference. As a measure of the scope of the scatter search we will therefore take the number of errors traced and compare it with the number of errors traced by an exhaustive search method.

Figure 1 shows results of tracing an experimental data switch control

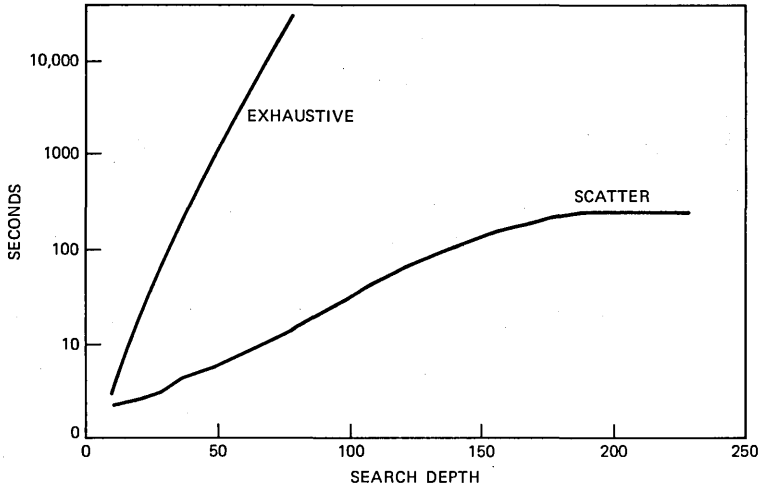


Fig. 1—Run time.

protocol, generating a state space in the order of 10^9 system states.[11] The protocol was analyzed several times, both for the exhaustive search and the scatter search, with a search depth restriction that was incremented in steps of 10 levels for each new analysis run.

An exhaustive search for this protocol became infeasible beyond a depth of 80 levels, that is, numbers of states down from the root of the state space tree. The tree scanned by the scatter search method had a maximum depth of 189 steps. Setting the search depth restriction beyond 189, therefore, no longer affects the scope of the analysis. To illustrate this, the curve for the scatter search was continued in Fig. 1 up to a depth of 230. The longest scatter search required less than 4 minutes of CPU time to complete. The run time of the exhaustive search tends to be exponential in the search depth. Using Fig. 1, it can be estimated that searching the state space tree down to the same depth (189 steps) with the exhaustive search would take some 3000 years of CPU time.

Fortunately, the test protocol analyzed contained a generous number of design errors. No attempt was made to classify them. In Fig. 2a the number of deadlocks reported by the tracer is shown as a function of the search depth, and in Fig. 2b the number of deadlocks versus the time it took to find them is plotted on logarithmic scales.

No deadlocks are found at search depths 10, 20, and 30. The first error is reported with the scatter search for a search depth of 40 steps, requiring 4 seconds of CPU time. For the same search depth restriction the exhaustive search reports the first 3 errors in 6 minutes. By repeating the analyses for intermediate levels between 30 and 40, we found that the first error is reported both in the scatter and the exhaustive search mode at level 35, requiring 4 seconds for the former and 3 minutes for the latter search. The two intermediate tests were included in the results shown in Fig. 2.

Very probably, no protocol designer would be interested in tracing this protocol beyond the first 100 error sequences generated. For the given test case this would mean that with an exhaustive search the

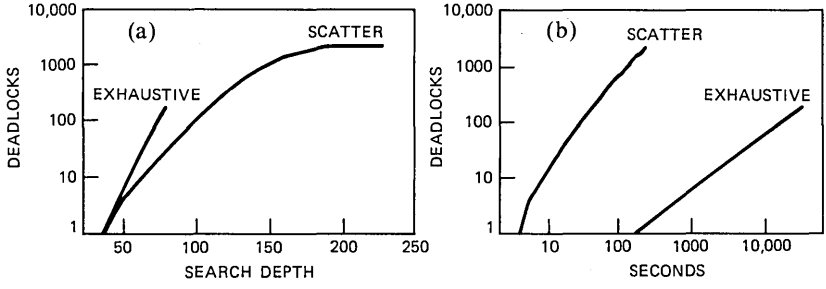


Fig. 2—Deadlocks.

first 70 steps in state space can be searched in roughly 3 hours of CPU time. Alternatively, the first 100 steps can be traced with a scatter search in only 30 seconds of CPU time.

Note that the time required to find the first error, the minimum search depth required to trace it, and the relation between search depth and the number of errors reported are favorable for the scatter search method.

2.2 State space

The protocol used for these tests requires roughly 40 bytes in the state space per system state. A total of 332,527 system states is generated in the longest exhaustive search analysis performed. As a result, for every new state generated, in the exhaustive search a data base of up to 15 megabytes must be probed for a state match. Even with the best hashing methods, this is bound to slow down the analysis noticeably. In the scatter search the largest number of states seen is 172,402 at a depth of 189 in the tree, corresponding to a database of 8 megabytes. The scatter search therefore should slow down less rapidly. This effect is illustrated in Fig. 3. The time efficiency is expressed in the average number of states analyzed per second for each analysis run.

The steep left-hand side of the curves can be attributed to the overhead involved in the setup of a state space, which is felt more if the number of states explored is small. With the current tracer, the optimum speed for both search methods is reached when the state space contains roughly 1000 states.

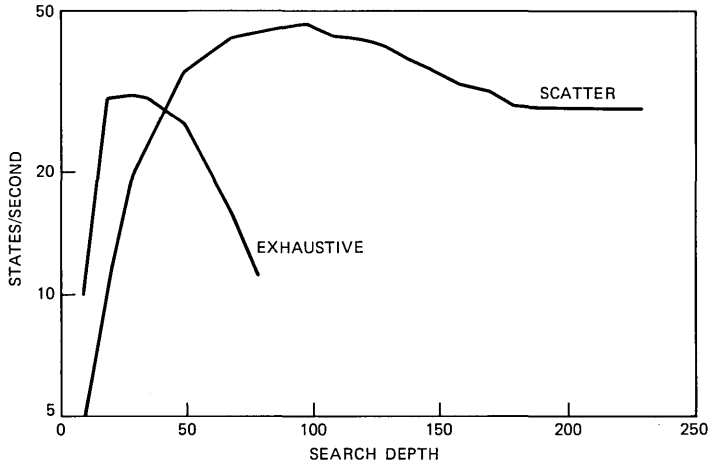


Fig. 3—Time efficiency.

III. SEARCH HEURISTICS

It is relatively straightforward to give preference to the shortest complete execution sequences and to defer analysis for longer sequences. We have already used this method in the preparation of the figures above by bounding the depth of the tree explored during a search. In this section we consider some other partial search heuristics.

3.1 Fewer interleavings

A method for reducing the run time of an analysis effectively is to restrict the amount of nondeterminism in the protocol model. In an exhaustive search, each node in the state space tree is root to one subtree for each executable option in each finite state machine in the protocol. Not all interleavings of these actions are necessarily relevant. Consider two executable actions: one action `a` local to machine $M1$, for example, an assignment to a local variable, and the other an external action `b` in machine $M2$, for example, a send or a receive. There are two possible orders in which these two actions could be executed, corresponding to the two sequences:

```
a; b and b; a
```

Each of the two sequences leads into a new state that forms the root of an entire subtree in the state space. The question is of whether or not the two subtrees are equivalent with respect to the errors to be traced. Note that the execution of internal action `a` will not change the environment for the remote machine $M2$, so neither the executability nor the result of `b` can be any different when `a` is executed first or last. Similarly, the execution of `a` is independent of the environment affected by $M2$ and also its executability and result is independent of whether `b` preceded it or not. In this case, then, it suffices to search one of the two possible interleavings and to ignore the other. Unfortunately, there are not many cases where a complete subtree can be ignored without restricting the scope of an analysis. In some cases, though, we can predict in what way the scope will be affected. It would be unwise to restrict the nondeterminism that is local to a finite state machine, as shown in the following Argos fragment:[12]

```
if
:: A?one -> P( )
:: B?two -> Q( )
fi
```

Argos is a CSP-like[13] guarded command language[14] defined on buffered message channels. A detailed discussion of the language itself can be found in Ref. 12. In the above example `A` and `B` are channel names (bounded buffers declared elsewhere), `one` and `two` are message names,

and P and Q are procedure names. If message one is the first message in A and message two is the first message in B, both input statements are executable and the process executing the above fragment can make a nondeterministic choice between the two alternatives, and then proceed with the execution of either P( ) or Q( ). The protocol tracer cannot foresee which of the two alternatives may produce an error without executing them. Note that ignoring one of the two alternatives in an analysis implies ignoring a potentially important code fragment, that is, either P( ) or Q( ), without having reason to assume that this code would be error free. In this case then both alternatives will have to be explored. The situation is different for the nondeterminism that results from concurrency, as illustrated by the following Argos fragment:

```
proc P1 { A?one -> P( ) }
proc P2 { B!two -> Q( ) }
```

It defines two processes P1 and P2. Assuming that both initial actions are executable, it must be decided in what order they will be executed by the tracer. This time it may, but it will not always make a difference in what order these two I/O statements are executed. In an exhaustive search both orders are always analyzed. Ignoring one of the two possible orders, however, can halve the amount of work to be done for this node in return for the chance that it will cause the tracer to miss error sequences. No code fragments are ignored here, only a potentially erroneous timing of executions. Fortunately, not all orderings have the same probability of leading into error states. For instance, if we are primarily interested in finding deadlock states, that is, states in which all message channels are empty and not all processes have reached their end states, we may choose to explore the sequence starting with a receive action and ignore the other. In practice this heuristic performs remarkably well, as illustrated by the results discussed earlier.

3.2 Tracing priorities

If at some node in the state space tree there are N concurrent processes, all executable, the tracer can decide to ignore any $M \leq N$ of the processes to reduce the search. In the tests reported in Figs. 1 to 3 we set $M = 1$ for the scatter search and $M = N$ for the exhaustive search. In the case where $M = 1$ the search heuristic is implemented as a priority scheme that determines which process should be executed next. Highest priority is given to internal actions. At the next level we place receive actions, since these tend to bring the system closer to a deadlock state with empty channels. A lower priority is given to send actions, and a lower priority still to channel time-outs. Time-outs are

given lowest priority in the partial searches since they tend to create many spurious error reports. In partial search mode the correct working of the time-out mechanism is assumed, that is, a time-out is only considered to be enabled when there is no other option to continue the protocol. Though this definitely reduces the scope of an analysis, it does allow us to trace for another class of errors first and defer the costly tracing of timing errors.

3.3 Queue sizes

The capacity of a communication channel for holding messages can also have an important effect on the size of a state space. In the specification language Argos the channels are modeled by finite queues. A channel then can be in only a finite number of states

$$\sum_{i=0}^{N} |S|^i,$$

where N is the number of slots in the channel (i.e., the queue size), and S is the size of the channel *sort*, that is, the set of all messages that can be recognized by the channel. Reducing the number of slots N by 1 can reduce the size of the state space and speed up the analysis by a factor of up to

$$\sum_{i=0}^{N} |S|^i - \sum_{i=0}^{N-1} |S|^i = |S|^N.$$

In the scatter searches of Figs. 1 to 3 the queue sizes were restricted to two slots. To study the effect of a variation of the queue size the tests were repeated for a small range of sizes. Figure 4 shows the effect

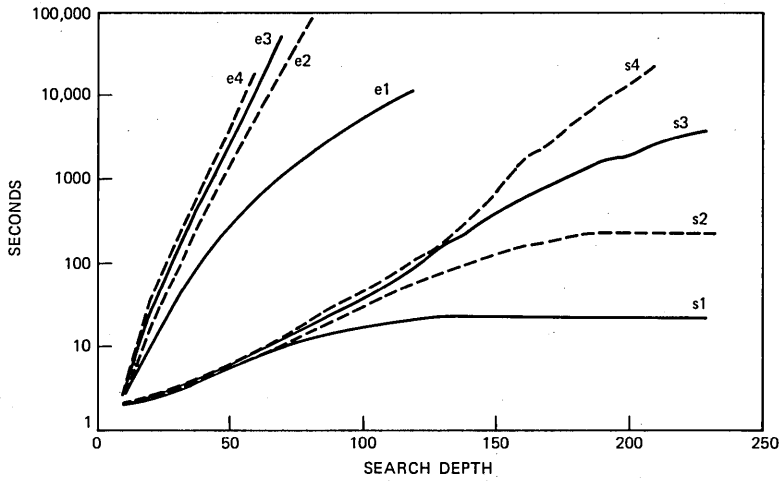


Fig. 4—Effect of queue sizes on run time.

of a variation in the number of slots between one and four for both the exhaustive and the scatter search.

IV. RESTRICTING THE STATE SPACE

In the Introduction we mentioned that the tracer should be able to perform searches in even incomplete state spaces since the size of a complete state space generally precludes its storage or even its usage during the search. In this section we show how this can be accomplished.

First it should be noted that in a depth-first search, at each execution step only those states that lead from initial state to the current state are indispensable in the state space. The presence of these states is necessary for the detection of system execution loops. Not every system state, though, can be found at the start of such an execution loop, and therefore it is not necessary to remember each state along a single execution path. The only states that must be remembered are those in which at least one of the interacting finite state machines is at the start of a local execution loop. Figure 5a shows a small but consistent reduction in the numbers of states if we restrict the state space to such "loop states."

4.1 Minimization

Since the analysis is performed on finite state machines we can try to minimize the machines in an effort to reduce time or space complexity without affecting the scope of an analysis. The machines cannot be reduced under the standard notion of language equivalence, since that will change the behavior or the protocol. A stronger notion of state equivalence,[12] similar to that defined in CCS[6] can be used.

Figure 5b compares the analyses of minimized state machines and nonminimized state machines. The protocol tested defines 4 processes, 34 message types, 6 message channels, and 3 local variables. The state machines generated for the processes contain 69, 47, 7, and 5 states, respectively. The strongly equivalent minimized machines contain 35,

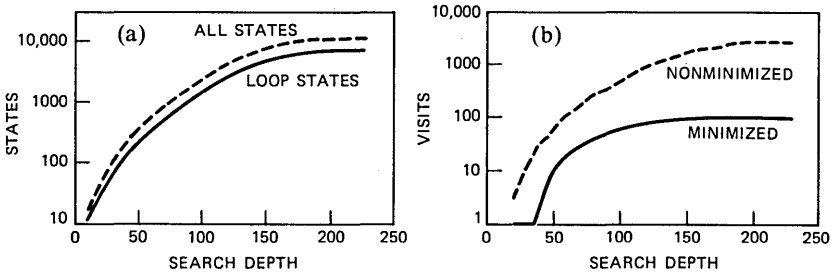


Fig. 5—Reducing numbers of states stored (scatter searches).

31, 7, and 5 states. As it turns out, the number of states generated in the state space is roughly the same in both cases. The connectivity of the state space tree, however, is different, causing the same states to be visited more frequently for the nonminimized machines, resulting in a small increase in run times.

The effort to minimize the amount of work to be done in the search algorithm is concentrated on minimizing the theoretical maximum number of states in the product space of the individual finite state machines. We can do this by reducing the number of states per state machine (e.g., by masking a variable or a message queue) or, less straightforwardly, by reducing the number of state machines as such. The last thing we would like to do is, of course, to extend the number of state machines that we begin an analysis with.

Somewhat paradoxically, this approach seems to conflict with the more conventional structured approach to program design that tells us to identify functions and to separate these in a relatively large number of logical entities. For protocol design this approach was most recently suggested in Ref. 15, which describes a method where each logical entity is formalized in a small finite state machine that interacts with the others. Dividing a single automaton of 16 states into 2 state machines of 8 states each, however, quadruples the number of states in the product space. Similarly, dividing it into 4 even simpler state machines of 4 states each expands the product state space to 16 times its original size. In general, increasing the number of state machines leads to an exponential growth of the product state space and is counterproductive in analyses.

4.2 Cache size

We noted above that, unlike the more commonly used *breadth*-first search (see Refs. 5, 9, and 12), the state space in a *depth*-first search need only contain the states in a single execution path from the root to the current state. Storing other states can avoid double work, but does not affect the scope of the analysis. This property of the depth-first search method gives us greater flexibility in controlling the state space size during analysis runs. If more states can be stored, though, the search will be more time efficient. Figure 6 shows the effect of the size of the state space on the time and space requirements of a search, for a state space cache of 150,000 states that is reduced in steps of 1,000 to a cache of 50,000 states. "Double work" is measured here as the total number of states created or recreated while searching. Note that the number of states stored in the cache could roughly be halfed without noticeable effect on the runtime or the total number of states created. With a partial state space cache it has to be decided which state will be deleted from a full cache when a new state must be

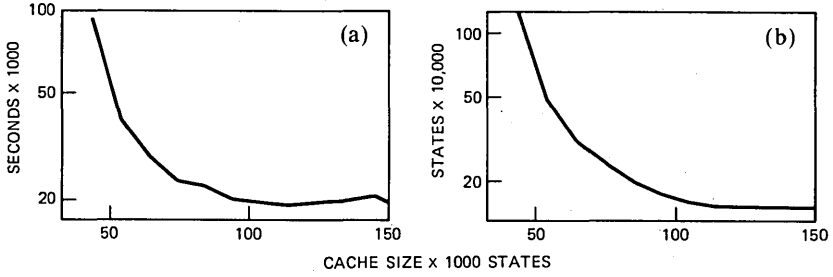


Fig. 6—Size of state space cache.

created. A simple blind round robin selection of states was found to outperform a series of other, more subtle, schemes.[11,12] It is the strategy used in the test of Fig. 6.

V. ASSERTION CHECKING BY SYMBOLIC EXECUTION

By default a protocol tracer can check a protocol for the observance of general correctness requirements such as absence of deadlock and completeness. The validation language Argos allows for the specification of assertions to check on the observance of other correctness requirements. Assertions are defined as a restricted class of processes. They specify global system behavior in terms of external actions. For example, the specification

```
assert
{       do
        :: large!mesg; small!mesg
        od
}
```

is a requirement on the order in which messages of the type `mesg` are sent to the two channels `large` and `small`. The assertion is that in each execution sequence a message on channel `large` must precede a message on channel `small`, and that these two actions will be executed repeatedly (they are enclosed in a `do` loop) in precisely this order.

The main restriction to assertion specifications is that they can only refer to external actions, that is, sends and receives, and not to variables. Assignments and Boolean conditions are only allowed in process definitions, not in assertions. The control flow constructs are the same as those for process specifications: concatenations, selections, iterations, jumps, procedure calls, and macros. In other words, the assertions specify *global* constraints on the execution of the system as a whole in terms of message exchanges only. The scope of the assertion, that is the set of external actions that is traced to verify or to violate an assertion, is implicitly defined by the set of external actions it

refers to. If an external action occurs at least once in an assertion body, all its occurrences in an execution of the protocol are required to comply with it. Compliance with the assertion then means that the execution of these actions should match the context specified in the assertion.

Since we define assertions as restricted processes, the assertion primitives can be compiled into a restricted class of state machines and minimized with the same algorithm that is used for the compilation of the protocol processes. The protocol tracer uses the assertion state machines to *monitor* the external actions on which they are defined. Alternatively, though our tracer does not exploit this possibility, it may be possible to develop a heuristic that allows the tracer to select those executions in a partial search that have the best chance of violating the correctness requirements expressed in the assertions.

If an action is within the scope of an assertion, the state of the corresponding state machine will be updated as a side effect of the execution of that action, as if the assertion machine itself generated it. Since the assertion primitives cannot access variables or channels, the "state" of an assertion machine is uniquely defined by its control-flow state. The "execution" of an assertion machine then costs very little in the tracing algorithm. When the protocol system reaches an end state, compliance with the assertion can be established by verifying that the assertion machine can reach an end state, too. If this is not true, the assertion is violated and the current execution sequence can be listed as a counter example. Similarly, if the assertion machine cannot be executed for an action that is within its scope, the assertion has been violated and a counter example can be produced. With little overhead or added complexity, the finite state machine model can thus be exploited to combine the depth-first search with assertion checking capabilities.

VI. AN EXAMPLE

A small example can illustrate how the experimental protocol tracer described in this paper is typically used. More elaborate examples can be found in the Appendix and in Ref. 12. A protocol is defined in the language Argos. The example below shows three processes `a`, `b`, and `c`, three message queues of one slot each named `A`, `B`, and `C`, and one assertion labeled `assert` (a keyword in the language).

```
assert { C!a; C!b }

proc a
{       queue A[1];
        C!a; A?c
}
```

```
proc b
{       queue B[1];
        C!b; B?c
}
proc c
{       queue C[1];
        if
        :: C?a -> A!c; C?b -> B!c
        :: C?b -> A!c; C?a -> B!c
        fi
}
```

The assertion states that the two messages a and b will be appended precisely once to queue C when the three processes are executed, and that they can be sent in that order only. Process a starts by sending message a to queue C and then waits for a response a to arrive in queue A. Similarly, process b first sends b to queue C and then waits for a message c. The third process waits for a message to arrive in queue C, which is assumed to be either an a or a b, anything else would be an error. Process c then responds by sending a c message to queue C and waits for a second message to arrive: a b if the first received message was an a, or an a if the first message was a b. It will complete by sending a c into queue B.

The protocol is compiled into four finite state machines of three states each for a and b, three states for the `assert` primitive, and seven states for process c. The protocol tracer then takes over and completes an exhaustive search in 1.35 seconds, reporting the obvious assertion violation for the execution sequence that starts with C!b. The violation is reported by the tracer in the following format:

```
queue:  A  B  C
event:
   1           b
   2     c
   3           a
   4        c
```

Each column corresponds to a queue and each line to a time step. The first event is the sending of message b to queue C, which already violates the assertion. Then message c is sent to A by process c, message a is sent to C by process a, and finally a c message is sent to queue B by process c.

Changing the assertion to a more reasonable statement such as `assert { A!c; B!c }` will avoid the problem. The exhaustive search

for this assertion completes in 1.32 seconds. Omitting the assertion completely will trigger a default search for deadlocks and incompleteness (e.g., unspecified receptions), which completes in 1.18 seconds. Note that it is relatively straightforward to formalize liveness criteria in assert statements. In this case, as for many protocols generating up to 10^5 system states, exhaustive analyses are quite feasible. The real problems of partial searching only occur for the larger protocols comparable in size to the experimental protocol used for the tests reported earlier in this paper.

VII. CONCLUSIONS

The main assumption we make in this paper is that in a design phase a protocol is typically known to contain errors and there is a need for a protocol tracing tool that can quickly find a representative subset of these errors. The user of such a protocol tracer is not so much interested in completeness but is very much interested in speed. With these assumptions important reductions in the time and space requirements of a tracer become feasible.

The protocol tracer described here consumes only a small fraction of the time and space required by an exhaustive analysis algorithm to find a relatively large fraction of the errors present. The run time of a state space search is reduced by several orders of magnitude by restricting the number of interleavings, by using search depth and queue size restrictions, and by using compile time minimizations (Figs. 1, 2, 4, and 5(b)). A more general method of reducing run time would be the definition of equivalence classes, or *state space foldings*, as described in Ref. 8. The experimental protocol debugger Trace does allow for the definition of such foldings, but too little experience with this technique has yet been gained to report any results.

The number of states stored in a state space can be reduced by carefully selecting the states that may be revisited (Fig. 5a). More importantly, though, the depth-first search technique used allows one to perform searches with an incomplete state space cache. For the protocol tested, the cache could be reduced to less than 50 percent of the state space size (Fig. 6).

VIII. ACKNOWLEDGMENTS

The experiments with assert primitives and the state space cache were inspired by discussions with Bob Kurshan and Sudhir Aggerwal. I am also grateful to Doug McIlroy, Lee McMahon, Rob Pike, Ed Sitar, and Ken Thompson for discussions, support, and inspiration during the development of the protocol tracer.

REFERENCES

1. T. P. Blumer and R. L. Tenney, "A Formal Specification Technique and Implementation Method for Protocols," Comput. Networks, *6,* No. 3 (1982), pp. 201–19.
2. D. Brand and W. H. Joyner Jr., "Verification of Protocols Using Symbolic Execution," Comput. Networks, 2 (1978), pp. 351–60.
3. J. Hajek, "Automatically Verified Data Transfer Protocols," Proc. 4th ICCC, Kyoto, September 1978, pp. 749–56.
4. C. West, "Applications and Limitations of Automated Protocol Validation," Proc. 2nd IFIP WG 6.1 Int. Workshop on Protocol Specification, Testing, and Verification, USC/ISI, Idyllwild, Calif., May 1982, pp. 361–73.
5. P. Zafiropulo et al., "Toward Analyzing and Synthesizing Protocols," IEEE Trans. Commun., *COM-28,* No. 4 (1980), pp. 651–61.
6. R. Milner, "A Calculus for Communicating Systems," Lecture Notes in Computer Science, *92* (1980).
7. G. J. Holzmann, "A Theory for Protocol Validation," IEEE Trans. Comput., *C-31,* No. 8 (August 1982), pp. 730–38.
8. G. J. Holzmann, "The Pandora System—An Interactive System for the Design of Data Communication Protocols," Comput. Networks, *8,* No. 2 (1984), pp. 71–81.
9. D. Brand and P. Zafiropulo, "Synthesis of Protocols for an Unlimited Number of Processes," Proc. Comput. Network Protocols Conf., IEEE 1980, pp. 29–40.
10. P. R. F. Cunha and T. S. E. Maibaum, "A Synchronization Calculus for Message Oriented Programming," Proc. Int. Conf. on Distributed Systems, IEEE 1981, pp. 433–45.
11. G. J. Holzmann, "Trace—Performance Measurements," AT&T Bell Laboratories, internal report, Jan. 1, 1985.
12. G. J. Holzmann, "Automated Protocol Validation in 'Argos,' Assertion Proving and Scatter Searching," 1984, available from the author.
13. C. A. R. Hoare, "Communicating Sequential Processes," Comm. ACM, *21,* No. 8 (August 1978), pp. 666–77.
14. E. W. Dijkstra, "Guarded Commands, Nondeterminacy and Formal Derivation of Programs," Comm. ACM, *18,* No. 8 (August 1975), pp. 453–57.
15. R. P. Kurshan, "Proposed Specification of BX.25 Link Layer Protocol," AT&T Tech. J. *64,* No. 2 (February 1985), pp. 559–96.
16. National Bureau of Standards, Specification of a Transport Protocol for Computer Communications, *4,* "Service Specifications," June 1984.
17. C. S. Crall and D. P. Sidhu, unpublished work.

APPENDIX

The following specification describes a transport protocol defined by the National Bureau of Standards.[16] The specification is based on the model given in Ref. 17. Four processes are defined: a local user process AU connected to a server process A, and a remote user BU connected to server process B. The control flow constructs and the I/O statements in Argos are based on CSP, using buffered message channels instead of rendezvous. Process A, for instance, receives messages via two channels: one is named `ua` and is used by the user process to request services, the other is named `ca` and is used here to receive control messages from the remote server. Messages from server to user are sent through channel UA. The communication between the two servers is modeled with control messages `m1` to `m7`, as defined in Ref. 17. The analysis discussed here uses no *assert* primitives and is thus a general one for completeness and absence of deadlocks. The arrow and the semicolon are syntactically equivalent statement separators. A double colon flags the start of an option in a repetitive

construct (`do ··· od`) or in an alternative construct (`if ··· fi`). In this case, the state transition diagram defining the protocol is most conveniently, though not most elegantly, modeled by assigning a label to every state and including a goto-jump for every transition. Processes `A` and `B` are symmetrical. Null transitions from the original protocol were deleted from the model.

```
proc A
{       queue ca[8], ua[8];

closed:
        do
        :: ca?m1 → UA!conn_ind → goto rcvd
        :: ua?conn_req → cb!m1 → goto crsent
        :: ua?abort → cb!m4
        :: ca?m4 → UA!d
        od;
crsent:
        if
        :: ca?m2 → UA!conn_conf → goto estab
        :: ca?m7 → UA!disconn → goto closed
        :: ua?abort → cb!m4 → goto closed
        :: ca?m4 → UA!d → goto closed
        fi;
  rcvd:
        if
        :: ua?conn_resp → cb!m2 → goto estab
        :: ca?m7 → UA!disconn → goto closed
        :: ua?abort → cb!m4 → goto closed
        :: ca?m4 → UA!d → goto closed
        fi;
 estab:
        do
        :: ua?close_req → cb!m3 → goto Aclose
        :: ca?m3 → UA!close_ind → goto Pclose
        :: ua?data_req → cb!m5
        :: ca?m5 → UA!data_ind
        :: ua?expid_req → cb!m6
        :: ca?m6 → UA!expid_ind
        :: ca?m7 → UA!disconn → goto closed
        :: ua?abort → cb!m4 → goto closed
        :: ca?m4 → UA!d → goto closed
        od;
```

```
Aclose:
        do
        :: ca?m3 → UA!close_ind → goto closed
        :: ca?m5 → UA!data_ind
        :: ca?m6 → UA!expid_ind
        :: ca?m7 → UA!disconn → goto closed
        :: ua?abort → cb!m4 → goto closed
        :: ca?m4 → UA!d → goto closed
        od;
Pclose:
        do
        :: ua?data_req → cb!m5
        :: ua?expid_req → cb!m6
        :: ua?abort → cb!m4 → goto closed
        :: ca?m4 → UA!d → goto closed
        :: ua?close_req → cb!m3 → goto closed
        :: ca?m7 → UA!disconn → goto closed
        od
}

proc B
{       queue cb[8], ub[8];

closed:
        do
        :: cb?m1 → UB!conn_ind → goto rcvd
        :: ub?conn_req → ca!m1 → goto crsent
        :: ub?abort → ca!m4
        :: cb?m4 → UB!d
        od;
crsent:
        if
        :: cb?m2 → UB!conn_conf → goto estab
        :: cb?m7 → UB!disconn → goto closed
        :: ub?abort → ca!m4 → goto closed
        :: cb?m4 → UB!d → goto closed
        fi;
  rcvd:
        if
        :: ub?conn_resp → ca!m2 → goto estab
        :: cb?m7 → UB!disconn → goto closed
        :: ub?abort → ca!m4 → goto closed
        :: cb?m4 → UB!d → goto closed
        fi;
```

```
 estab:
        do
        :: ub?close_req → ca!m3 → goto Aclose
        :: cb?m3 → UB!close_ind → goto Pclose
        :: ub?data_req → ca!m5
        :: cb?m5 → UB!data_ind
        :: ub?expid_req → ca!m6
        :: cb?m6 → UB!expid_ind
        :: cb?m7 → UB!disconn → goto closed
        :: ub?abort → ca!m4 → goto closed
        :: cb?m4 → UB!d → goto closed
        od;
Aclose:
        do
        :: cb?m3 → UB!close_ind → goto closed
        :: cb?m5 → UB!data_ind
        :: cb?m6 → UB!expid_ind
        :: cb?m7 → UB!disconn → goto closed
        :: ub?abort → ca!m4 → goto closed
        :: cb?m4 → UB!d → goto closed
        od;
Pclose:
        do
        :: ub?data_req → ca!m5
        :: ub?expid_req → ca!m6
        :: ub?abort → ca!m4 → goto closed
        :: cb?m4 → UB!d → goto closed
        :: ub?close_req → ca!m3 → goto closed
        :: cb?m7 → UB!disconn → goto closed
        od
}

proc AU
{       queue UA[8];
        pvar m = 0;

        do
        :: UA?conn_conf → ua!close_req; UA?close_ind
        :: UA?close_ind → ua!close_req
        :: UA?conn_ind → ua!conn_resp
        :: (m == 0) → m = 1; ua!conn_req
        :: (m == 1) → m = 2; ua!abort
        :: UA?default
        od
}
```

```
proc BU
{       queue UB[8];

        do
        :: UB?conn_conf → ub!close_req; UB?close_ind
        :: UB?close_ind → ub!close_req
        :: UB?conn_ind → ub!conn_resp
        :: UB?default
        od
}
```

The queue sizes were arbitrarily set to 8 slots per channel. The protocol tested is defined by the behavior of the two server machines, as visible to the users. The user behavior is no part of the formal protocol. An arbitrary set of user processes was defined specifically for the test. The local user `AU` will open the connection by sending a `conn_req` message to `ua` and some arbitrary time later it will close it with an `abort` message. The remote user `BU` is considered to be passive, responding only to close messages and accepting, but ignoring all others. `Default` is a keyword for receptions that match any input from the queue specified.

The protocol as specified above is compiled—in 23 seconds of CPU time on a VAX-11/750*—into four finite state machines of 27, 27, 10, and 6 states, respectively. The compiler flags a series of incompleteness errors, noting for instance that control message `m7` can be received but is never sent. Ignoring those warnings, an exhaustive analysis with `trace` takes just under 3 seconds of CPU time and reports 4 error sequences that reduce to two types of errors. The first one is an unspecified reception of the message `conn_resp` in state `closed`, for instance, after the following message exchange:

```
queue:  ca    ua      UA   cb      ub        UB
event:
 1          conn_req,
 2                       m1,
 3                                        conn_ind,
 4          abort,
 5                       m4,
 6                                        d,
 7                           [conn_resp],
```

Each column corresponds to a queue and each line to a time step. The first event recorded is the sending of a message `conn_req` into queue `ua`, followed by an `m1` into queue `cb`, etc. The last message sent is

*Trademark of Digital Equipment Corporation.

enclosed in square brackets to indicate that it was sent but could not be received. Comparing the event sequence with the program shows that server process `B` is in state `closed` at the time.

The second problem is an unspecified reception of `m2` also in state `closed`:

```
queue:  ca     ua       UA  cb      ub        UB
event:
 1             conn_req,
 2                         m1,
 3                                           conn_ind,
 4             abort,
 5                               conn_resp,
 6                         m4,
 7      [m2],
 8                                           d,
```

It is now straightforward to study the behavior of the protocol for different user behaviors, which can reveal, for instance the possibility of the unspecified reception of a message `conn_resp` in state `Pclose`, or the more obvious deadlock after a simultaneous `conn_req` message from both users.

AUTHOR

Gerard J. Holzmann, Kand. Ir. (B.S.), 1973, Ir. (M.S.) (Electrical Engineering), 1976, Ph.D. (Technical Sciences), 1979, Delft University of Technology, The Netherlands; Mr. Holzmann obtained a Fullbright Fellowship in 1979; with the University of Southern California in Los Angeles, 1979–1980; AT&T Bell Laboratories, 1980–1981, 1983—. Before returning to AT&T Bell Laboratories, Mr. Holzmann was an Assistant Professor at the Delft University of Technology. In 1981 he was awarded the Prof. Bähler prize of the Royal Dutch Institute of Engineers (KIVI) for his work on telecommunication systems. His current research is in distributed systems and computer graphics in the computing science research center.

AT&T Technical Journal
Vol. 64, No. 10, December 1985
Printed in U.S.A.

A VCR-Based Access System for Large Pictorial Databases

By K. Y. ENG, O. YUE, B. G. HASKELL, and C. GRIMES*

(Manuscript received July 11, 1985)

We describe a Videocassette Recorder (VCR)-based information system whereby we can distribute frequently updated large pictorial databases to individual users and provide a variety of interactive video services. The four key advantages of this system are: (1) economics, (2) good picture quality, (3) capability to reach nationwide users, and (4) ability to update the database frequently (say, daily, preferably in early morning hours when many transmission facilities are unused). An experimental home terminal consisting of a VCR driven by a personal computer for random-access searches was constructed to demonstrate this concept. The pictorial database used in the demonstration includes real estate listings, vacation guides, autos and Sears-type merchandise catalogs. We also make comparisons of this system to other video services and conclude that the present approach has potential advantages in many applications.

I. INTRODUCTION

There are presently many systems under development for providing interactive visual displays.[1] For example, videotex uses the switched telephone network to send and receive digital data, which are then used by a microprocessor terminal to construct color graphics on a TV screen. Teletext is another technique whereby digital data for the same purpose are imbedded in the vertical blanking period of a video signal broadcasted to the end users. The graphics capability of these two

* Authors are employees of AT&T Bell Laboratories.

methods is limited in representing those real objects, e.g., clothing and furniture, for which visual attractiveness is more important than functional appeal.

Limitations in the computer-graphic representation of real objects can be overcome by storing the pictures on an interactive videodisc (as in the "electronic book" application).[2] In such a case, hard copies of the disc have to be distributed by mail or through stores to the end users. If real-time access to a central database is really required, then the so-called frame-grabber approach is frequently suggested. In this latter system, single frames are sent to individual users, time multiplexed on a dedicated video channel.[3] The user terminal must then store the received frame so that it can be examined by the human viewer. Digital frame stores for this application are expensive, although their cost should decline eventually. Nevertheless, a multiuser computer system at a central location must manage the data requests and send different video frames to different users. Such a system can be overloaded easily, and practical solutions for giving nationwide service to thousands of users simultaneously have yet to be worked out.

Here we suggest an alternative arrangement which appears to be more economical than the videotex or frame-grabber systems. We propose that the home terminal consist of a home Videocassette Recorder (VCR) connected to a personal computer, and that a "frame-search" capability be provided whereby the home computer can specify which frame of the videocassette is displayed (in still frame) at any one time. The VCR must of course have good still-frame performance. The overall system for distributing pictorial databases from a central station to end users with such home terminals is outlined in Section II. Then we describe an experimental home terminal for demonstrating the feasibility of this idea (Section III). Finally, we make comparisons with other systems (Section IV) and conclude that our present proposal has potential applications in both the business and consumer markets.

II. DIRECT DISTRIBUTION OF PICTORIAL DATABASES TO USER VCRS

We propose the direct distribution of pictorial information to the end users' VCRs via a TV broadcasting channel, as illustrated in Fig. 1. The pictorial database is assembled in one central location, where individual color pictures (35 mm photographs or slides) are recorded onto a master videotape (one-inch type) in a frame-by-frame manner, i.e., a single color picture on a single video frame. A two-hour videotape can then store up to 216,000 single-frame pictures. In the vertical blanking period of these video pictures, we insert a frame number for identification as analogous to the page number in a book. This technique of numbering the video frames can be implemented easily with the conventional Vertical Interval Time Code (VITC). In addition,

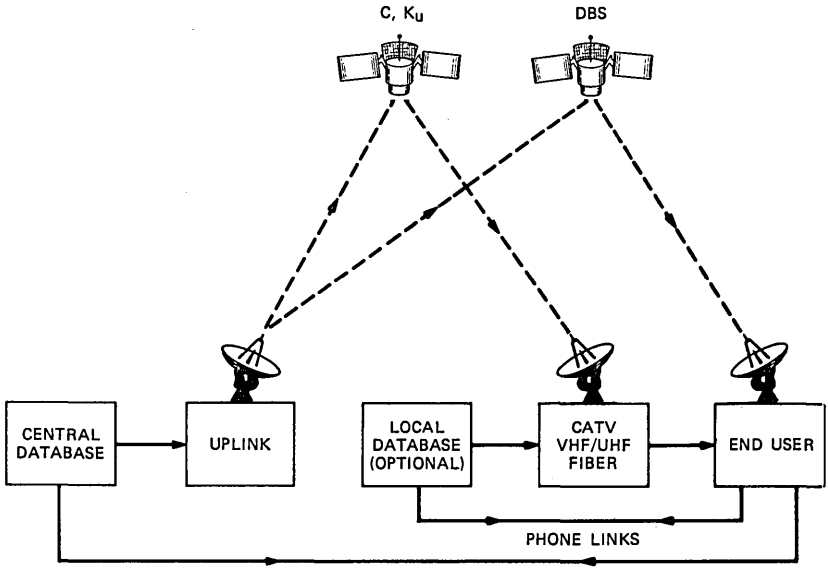


Fig. 1—Direct distribution of pictorial databases to home users.

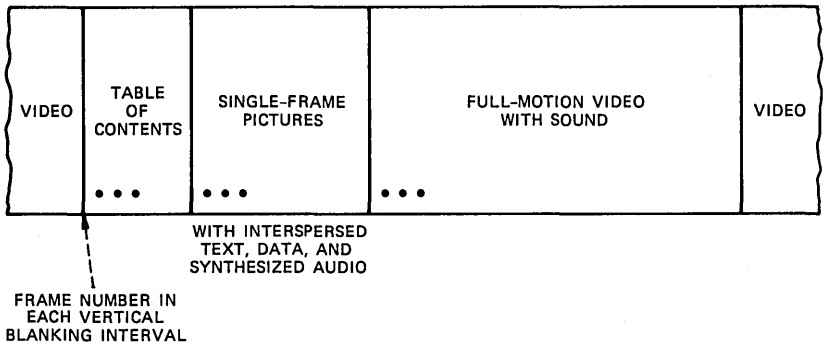


Fig. 2—A video segment showing a sample database layout.

part of the database, i.e., parts or the whole of a video frame, can be devoted to nonpictorial information such as table of contents, text, software instructions, etc. These data, although encoded digitally, can easily be incorporated in place of the normal active video. As an example, we show in Fig. 2 the layout of a database consisting of table of contents, text descriptions, synthesized audio, single-frame pictures, and full-motion video with sound.

As suggested in Fig. 1, the database from the central station can be transmitted through a satellite broadcasting system (C-, K_u-band, or DBS) whereby nationwide users can either receive the information directly or through a local broadcaster. In the latter case, the local

distributor simply takes the received video signal from the satellite and retransmits it to the users via his own broadcasting system (e.g., cable TV, off-the-air VHF/UHF channels, optical fiber, etc.). Attaching additional information from a local database is optional. Since many transmission facilities and TV channels are idle in early morning hours, this operation can most conveniently be done in the middle of the night with the VCRs programmed or pretimed for unattended recording. Once the database is recorded on a videocassette, the users can assess the information at their home terminals at their leisure. Direct distribution in this manner avoids the cost of recording and shipping thousands of tapes (or discs). Moreover, the database is always up-to-date, depending on how often it is sent (once a day should be adequate for most applications). No special equipment is required at the TV station or CATV head-end for sending out the databases. Indeed with satellite transmission, nationwide distribution is possible with transmission systems already in place. For example, the Sears catalog could be sent in 4.5 minutes, assuming 1600 pages and 5 frames per page. One thousand real-estate listings could be sent in 5.5 minutes, assuming 10 frames per listing. Custom orders for still-picture or full-motion information (e.g., instruction manuals) could even be served if more transmission time were available.

With the database recorded on a videocassette, we can use the interactive home terminal suggested earlier to browse through the pictures in a random-access manner. We constructed an experimental terminal to demonstrate this idea, which is discussed in the next section. But first, we should point out that the assembly of large databases at the master station is no easy task. In fact, the production cost could be an important consideration in systems of this type. The need for automation in database production is mandatory and indeed manageable with modern production equipment.

III. AN EXPERIMENTAL VCR-BASED INTERACTIVE TERMINAL

We show in Fig. 3 the block diagram of a VCR-based interactive terminal suitable for examining stored video data from a videocassette. The solid-line portion represents what was implemented in our laboratory as an experiment to demonstrate the proposed concept, while the dashed-line part stands for an alternate approach using a remote computer (via telephone hook-up) to control the operation. They are functionally the same, but offer different kinds of user flexibility. More is discussed about their differences after we explain the terminal itself.

The terminal consists of three major components: the computer, which serves as a controller for the entire system; the VCR/computer

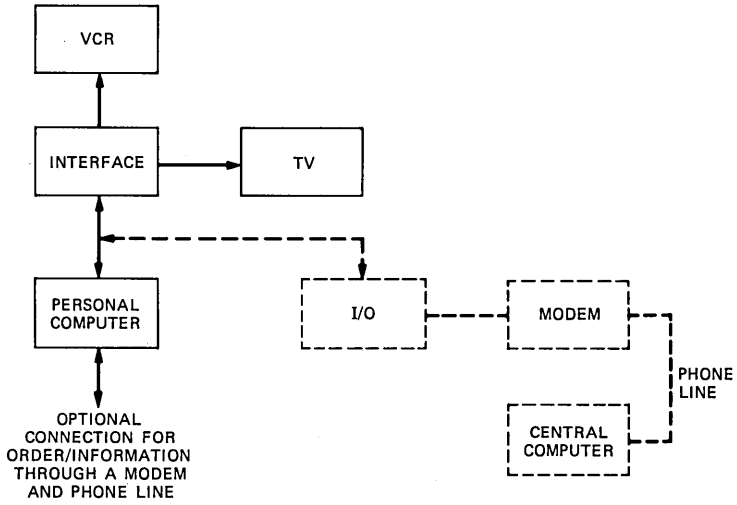


Fig. 3—A VCR-based user terminal.

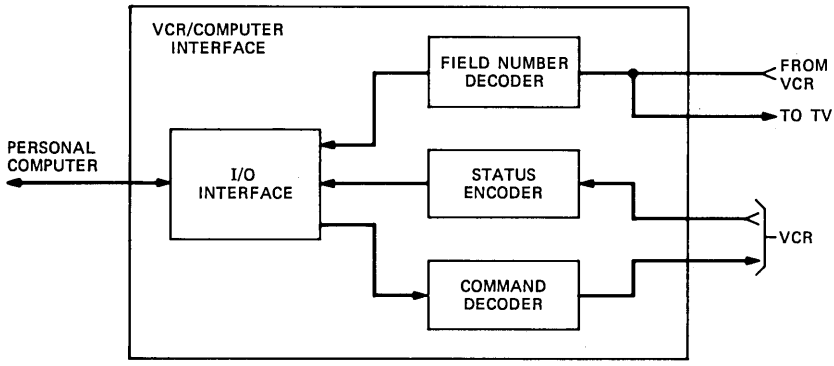


Fig. 4—VCR/computer interface.

interface; and the VCR* (with TV) itself. The video signal from the VCR is passed through the VCR/computer interface before being displayed on the TV. Inside the interface (Fig. 4), a field number decoder is used to examine the video frame identification number (VITC) recorded in the vertical blanking interval. This information should be made available to the computer at all times, i.e. when the tape is in still frame or in motion. But for simplicity, it is sufficient to provide valid frame numbers for tape motions from still-frame to play speed (30 frames/second). For higher tape speeds, the computer can

* Both the VHS and the Betamax formats would work as long as the VCR has good still-frame performance. We used VHS in our terminal.

use calculated estimates based on the initial known tape location, the current known tape speed, and the measured elapsed time plus some precalibrated adjustments. As an adjunct, the status encoder monitors the current operational status of the VCR (such as stop, fast forward, play, etc.) and conveys it to the computer. The command decoder, on the other hand, receives commands from the computer and translates them into actual instructions to the recorder, i.e., emulating a human pushing buttons to control the VCR. This was done simply by wiring up computer-driven electronic switches over the contact points of the pushbuttons on the remote control unit of the recorder.

Although the field number decoder is designed to detect VITC for frame identification, it is easy to extend the idea for decoding digital data recorded in all or part of an active TV field. Thus, part of the database can be devoted to digital information such as table of contents, programming or search instructions, synthesized audio, etc. They can be copied directly to the computer memory for use.

A sample video database was put together for the experimental demonstration of the terminal. It consists of four sections: (a) real estate listings; (b) automobiles; (c) a vacation package; and (d) merchandise catalogs. The interactive access to these data is done via a touch-sensitive screen on which menus are printed to prompt the user. The choices are all self-explanatory and are available to the user as touch-sensitive buttons on the personal computer. Instructions have been kept to a minimum, and the operation is so user friendly that a user manual is seldom needed. The majority of the material in the demonstration database is still-frame pictures. However, the full-motion video segments (with sound too) in the Hawaii Vacation Guide and also in a merchandise catalog on how to use the riding lawn mower consume much more recording time than the still-frame pictures. Note that our use of the personal computer with a touch-sensitive screen was merely a choice for experimental convenience. Any other user interface may be substituted to serve the same purpose in an actual system.

In addition to the user-prompting menu, 16 additional touch-sensitive buttons are always available in the bottom of the computer screen (see Table I). As an example, if one selects the button for real estate listings, then a table of choices for different towns appears on the touch-sensitive screen (Fig. 5). Touching one of these choices would bring up the next menu for specific price ranges. After that we have one more menu selection for specific house types, e.g., two-story colonial, log home, etc. The computer then fetches the text listing of the house selected from its memory and displays it on the touch-sensitive screen. Meanwhile, the search routine to locate the picture for the house is executed to the VCR. Therefore, the user sees the

Table I—List of 16 touch-sensitive buttons always available to a user

Name	Description
Autos	Brings up a new car catalog.
Vacation Package	Brings up the Hawaii Vacation Guide consisting of "Map of the Hawaii Islands," "Scenes From Hawaii," "The Polynesian Culture Center," and "The Sea Life Park," all of which (except the map) are full-motion video with sound.
Real Estate	Brings up the real estate listings: still pictures of the houses from the VCR and their corresponding text descriptions on the touch-sensitive screen.
Sears Catalog J. C. Penney Montgomery Ward	Brings up the merchandise catalogs: still pictures of individual items plus full-motion-and-sound segments of merchandise demonstrations.
Menu	Returns to the previous menu.
Order/Contact	Allows user to place an order or obtain order information.
Help	Brings up function definitions.
Compare	Compares prices of same item from different catalogs, if available.
Single Frame≫	Single-frame advance in the forward direction.
Single Frame≪	Single-frame advance in the reverse direction.
Browse≫	Causes video to move forward with momentary pause at each still-frame picture for browsing.
Browse≪	Causes video to move backward with momentary pause at each still-frame picture for browsing.
Pause	Pauses for still-frame viewing and brings up text description of current item.
Exit	Terminates the viewing session and returns to beginning.

color picture of the house on the TV and has the text listing simultaneously from the computer. The 16 buttons at the bottom of the screen provide plenty of choices and flexibility for the next step.

The search algorithm implemented in our software is designed to perform random access on the VCR in the most expedient manner. It

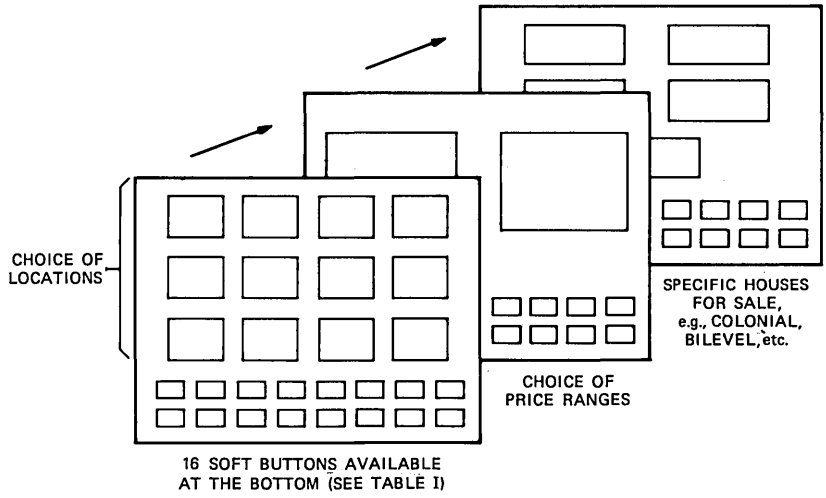


Fig. 5—Three successive menus for real estate selections.

should not be a surprise, however, that the VCR access time is considerably slow compared to that of a device truly capable of random access, e.g., a videodisc player. It takes a VCR typically 3 to 5 minutes to rewind a standard T120 (2/4/6-hour) cassette, and this would be the worst-case waiting time for getting to a picture from one end of the tape to another. Such a long access time is probably unsuitable for many applications. To circumvent this inconvenience, we propose that the whole database be divided into segments of, say, 1000 pictures each. Then the search within each segment is quite fast. As an illustration, if a 10 times play (10X) speed is used for the search, i.e., 300 frames/second, the maximum access time within such a segment is only 3.3 seconds. In our experimentation, we found that the speed search feature of the VCR was the most appropriate function to use for random access (i.e., fast visual search with 10X, 5X, 2X, etc.) because the video heads did not have to be disengaged and then reengaged during the process, and the continuous display of fast motions on the TV helped to make the waiting time less objectionable. In the event that the desired TV frame is very far away from the current position, the normal fast forward or rewind might have to be used. The optimization process is based on information previously obtained by an automatic calibration program which sizes the start and stop times, the video head engage/disengage intervals, and so on for each specific VCR. In short, we have developed an automatic calibration process that can characterize the access performance of a VCR, and the random-access software is intelligent enough to use this information for optimal control.

Laying out the database in segments of 1000 frames each requires intelligent partition of the overall source into groups of "humanly correlated" materials. For example, we don't want to put the listings of all different tires in one segment as in a conventional merchandise catalog. Instead we want to group all the accessories of a specific car model together. The situation is analogous to that of a large library where the books are carefully categorized and authored in such a way that most of the information desired for a subject is confined to a book of 1000 pages. The user can browse through each book very quickly, but probably does not mind spending more time in looking for another book. The assumption here is that most readers would spend some time reading a specific book of interest rather than reading pages of uncorrelated materials from different books in a random manner. We recognize that such an idea needs much more research before it can be put into practice. Nevertheless, recent experience from the videotex experiments seems to suggest evidence supporting the validity of this concept.

It is clear from the foregoing discussion that software plays a key role in this system. In our experimental setup, all the software resides in the personal computer (Fig. 3), and the user has complete control. The dashed-line portion in the figure indicates the other alternative of having the controlling software in a remote computer, and the user interacts with the system via a key pad connected to a modem. This latter approach has the advantage that more powerful software can be used at the expense of less user control as well as less privacy. The basic idea remains, however, that very intelligent software is needed in managing the enormous database made possible by a simple consumer-type VCR.

Finally, let us point out that the VCR system can be used as a high-density digital storage unit, which can provide digital high-fidelity music as well as synthesized voice and text data.

IV. APPLICATIONS AND COMPARISONS

The VCR-based information system has the same applications as other interactive video services. Some of the generic examples include real estate listings, vacation/entertainment guides, merchandise catalogs, product demonstrations, and service/instruction manuals.

We summarize in Table II a comparison with videotex and the videodisc-based system. It should be emphasized before we discuss these results that there is no single criterion possible for judging the relative merits or shortcomings of any approach. Instead, most systems tend to be application oriented. In other words, each individual video service tends to appeal more for the application it is intended for, and there is probably no single system that is universally "better" than all

Table II—Comparison of pictorial information systems

	VCR Based	Disc Based	Videotex
Video quality	Good	Good	Graphic
Database creation	Automatic photo-to-tape	Automatic photo-to-tape	Manual photo-to-graphic
Distribution	VHF/UHF, CATV, DBS, fiber, etc.	By mail or through stores	Telephone
Number of hard copies	1	100,000+	100+
Frequent updates	Yes	No	Yes
Real-time interaction with data suppliers	Limited	Limited	Yes
Response time	Slow	Fast	Depends on number of users

others. With this in mind, let us proceed with the comparison item by item.

4.1 Video quality

The picture performance of VCRs is usually designed to be compatible with the characteristics of other devices they are connected to in most home use, e.g. resolution is similar to that of a popular consumer TV (without comb filtering) and signal-to-noise ratio is comparable to most cable TV systems. In any event, their picture quality in our subjective viewing was found to be remarkably close to that of cable TV, while the videotex picture tends to be cartoon-like. The videodisc is potentially capable of noticeably better quality than the VCR although this is usually not so in practice.

4.2 Database creation

The database creation process for the VCR and the videodisc is almost the same. In both systems, the original material (e.g., photos or slides) is recorded on a one-inch video tape serving as the master, and the difference between the two cases is that the videodisc requires further processing in transferring the tape material onto a master disc before mass duplication. As for videotex, the original material has to be recreated in the computer (with computer-aided tools) as a graphic representation of the real object.

4.3 Distribution

The VCR system uses the TV broadcasting for distribution. Videotex uses the phone lines to connect customers to a central computer, while videodiscs have to be distributed via mailing or store sale.

4.4 Number of hard copies

Only one master copy of the database needs to be maintained nationwide in the VCR system. Videotex service requires a hard copy at each computer center, and hundreds are thus needed nationwide. The videodiscs are, of course, hard copies of the master, and a national market would require hundreds of thousands of them.

4.5 Frequent updates

The VCR database can be updated as often as possible because only one master copy is involved, and the user copies can be updated as often as transmission time permits. Similarly, videotex data can also be kept up to date. On the contrary, videodiscs cannot be changed easily, and current rewritable disc systems are prohibitively expensive for most applications.

4.6 Real-time interaction with data supplier

Because the database resides in the user terminal, real-time interaction with the data supplier tends to be limited in both the VCR and videodisc systems. On the other hand, videotex users enjoy continuous real-time interactive picture transmissions with the central computer.

4.7 Response time

The VCR access time is considerably slow compared to the videodisc, as discussed earlier. The videotex response time depends mainly on the number of simultaneous users and is probably slow (tens of seconds) for a fair size of simultaneous accesses (say thousands).

4.8 Cost

The VCR approach could become extremely economical if there was a mass availability of VCRs and personal computers, both of which have tremendous appeal in their own right and are gaining popularity among businesses and consumers. The custom interface necessary to connect the VCR to the personal computer is so simple that it can easily be incorporated into either the VCR or the computer. In any case, its cost should only constitute a small fraction of that for the total user terminal. Videodisc players capable of true random-access search are fairly expensive, and their popularity with consumers seems to be on the decline. Videotex terminals are also quite expensive, but their cost could decrease dramatically if more large-scale integration were employed.

V. CONCLUSIONS

We have proposed a system for distributing large pictorial databases to home videocassette recorders (VCRs). Distribution is done by

broadcasting from a master station where the picture information has been assembled in a frame-by-frame manner, i.e., one picture per video frame, resulting in 30 independent pictures in a 1-second video segment. The broadcasting medium could be a combination of direct broadcast satellites, cable TV, conventional VHF/UHF TV channels, or custom fiber systems. The main idea is that this can take place in the middle of the night when many transmission facilities become vacant, and the home VCRs can easily be pretimed for unattended recording. In fact, distribution or updating is possible as often as transmission time permits, but once a day is probably adequate for most applications.

With the complete database stored in a videocassette, an end user can retrieve the data of his particular interest at his leisure with the aid of a simple home terminal. We constructed such a terminal comprising a VCR capable of good still-frame performance, a personal computer serving as a controller for random access, and a custom interface connecting them together. The VCR/computer interface translates digital commands from the computer into actual operational instructions to the VCR, i.e., emulating a human pushing the control buttons on the recorder. It also feeds back the operational status of the VCR to the computer. Most important of all, it examines the video signal and decodes a frame number previously recorded in the vertical blanking interval during data assembly. This frame number is the "page number" of the electronic book and is supplied to the computer so that it knows which video frame or picture is being displayed on the TV. Software on the computer was implemented to do random-access search through the database, and the interface to the user is in the form of a touch-sensitive screen with menu-driven selections. The capability of this experimental terminal was demonstrated with a sample database consisting of real estate listings, new car models, vacation packages, and merchandise catalogs.

The main attraction of our proposal is its potential economics. That is, it takes advantage of other potentially low-cost and widely available terminal equipments, namely, the personal computer and the VCR, both of which have consumer appeal in their own right. Distribution requires only a single database plus transmission facilities that are already in place. Thus, a service supplier need only provide a hardware interface plus software, a networker need only supply a satellite or a TV station plus a video production unit, and purveyors of information need only furnish color photos and text.

VI. ACKNOWLEDGMENT

The authors wish to thank R. F. Weihs for his work on the experimental terminal.

REFERENCES

1. IEEE Journal on Selected Areas in Communications, *1*, No. 1, 1983.
2. R. D. Gordon, "An Intelligent Electronic Book System and Publishing Facility," Conference Record of Outlook For Optical and Video Disc Systems and Application, February 20, 1985, The Institute for Graphic Communications, Miami, Florida.
3. H. Ando and H. Yamine, "Still-Picture Broadcasting—A New Informational and Instructional Broadcasting System," IEEE Trans. Broadcasting, *BC-19* (September 1973), pp. 68–76.

AUTHORS

Kai Y. Eng, B.S.E.E. (summa cum laude), 1974, Newark College of Engineering; M.S. (Electrical Engineering), 1976, Dr. Engr. Sc. (Electrical Engineering), 1979, Columbia University; RCA Astro-Electronics, 1974–1979; AT&T Bell Laboratories, 1979—. Mr. Eng has worked on various areas of microwave transmission, spacecraft antenna analysis, and TV transmission in communications satellites. He is presently a member of the Network Systems Research Department, studying metropolitan area networks, switching theory, and real-time software. Member, IEEE, Sigma Xi, Tau Beta Pi, Eta Kappa Nu, Phi Eta Sigma.

Catherine R. Grimes, B.S. (cum laude) (Physics and Chemistry), 1982, Georgian Court College; AT&T Bell Laboratories, 1982—. Ms. Grimes is currently pursuing a graduate degree in computer science at Stevens Institute of Technology. She is a member of the Network Systems Research Department. Previous work has been in the area of process control software and user-friendly interfaces. Current work deals with the software aspects of metropolitan area networks.

Barry G. Haskell, B.S., 1964, M.S., 1965, and Ph.D., 1968 (Electrical Engineering), University of California, Berkeley; AT&T Bell Laboratories 1968—. From 1964 to 1968 Mr. Haskell was a Research Assistant in the University of California Electronics Research Laboratory, with one summer spent at the Lawrence Livermore Laboratory. Since he joined AT&T Bell Laboratories he has worked as a consultant in the Computer and Robotics Systems Research Laboratory. In 1977 and 1979 he was a part-time Faculty Member of the Department of Electrical Engineering at Rutgers University, and in 1983–84 he was similarly associated with The City College of New York. His research interests include digital transmission and coding of images, videotelephone, satellite television transmission, medical imaging, as well as most other applications of digital image processing. He has published over 30 papers on these subjects and has 15 patents either granted or pending. Mr. Haskell is a member of Phi Beta Kappa and Sigma Xi, and is a Senior Member of the IEEE.

On-Ching Yue, B.E.E., 1968, Cooper Union; M.S.E.E., 1971, Rochester Institute of Technology; Ph.D. (Information Sciences), 1977, University of California at San Diego; General Dynamics/Electronics Division, 1968–1977; AT&T Bell Laboratories, 1977—. Mr. Yue has worked in the areas of underwater acoustics and microwave imaging. Since joining AT&T Bell Laboratories, he has been a member of the Radio Research Laboratory, studying the effect of interference on digital communications systems, including intersymbol, adjacent satellite, and multiuser interferences. He is currently Supervisor of the Performance Analysis Tools Group in the Systems Analysis Center. Member, Tau Beta Pi, Eta Kappa Nu, Phi Kappa Phi; Senior Member, IEEE.

AT&T Technical Journal
Vol. 64, No. 10 December 1985
Printed in U.S.A.

A Broadband Local Area Network

By A. N. NETRAVALI* and Z. L. BUDRIKIS†

(Manuscript received April 18, 1985)

The IEEE 802 standard for local area network based on Carrier Sense Multiple-Access with Collision Detection (CSMA/CD) operates at a peak rate of 10 Mb/s on a cable of maximum length 2.5 km using baseband signaling. In many situations, larger channel rates are required over a much larger area. However, the efficiency of the CSMA/CD access method decreases rapidly if either the length of the cable is increased for a fixed bit rate or if the bit rate is increased for a fixed cable length. In this paper, we propose a broadband network for computer communications containing several CSMA/CD-type systems, each operating in a different frequency band. In addition, in order to have a wide area access, while minimizing the loss of performance associated with large collision delays, terminals in a small given geographical area are given one of the frequency bands for transmission. Two access protocols are developed. Using these schemes, it is possible to increase the channel throughput and the access area and to reduce the collision delay. We present a simplified analysis to quantify the improvement in performance using our schemes.

I. INTRODUCTION

Local Area Networks (LANs) share computing and other resources among many users and, if properly designed, increase the reliability by reducing the dependence of a user on one processing unit or a peripheral. Unlike long-haul networks, where channel utilization has to be optimized owing to high cost of communication over long

* AT&T Bell Laboratories.
† Visiting from the Department of Electrical and Electronic Engineering, University of Western Australia, Nedlands, Australia.

distance, local networks use bandwidth somewhat extravagantly to reduce the switching costs. Several network topologies, such as rings, buses, and trees, have been proposed along with access methods such as carrier sense, token passing, etc.[1,2]

The IEEE 802 standard for local area networks uses CSMA/CD (Carrier Sense Multiple-Access with Collision Detection) as one of its access methods.[3] It uses baseband transmission on coaxial cables (although other media are possible) at a peak rate of 10 Mb/s. For a variety of reasons, length of the cable (and therefore length of each segment of the network) is limited to 2.5 km. Within the limitations of the above parameters, the CSMA/CD-based access method provides an efficient means of computer communication for low loads on the channel. However, if the channel loading is increased, or if the requirements dictate either higher bit rates or longer cable lengths—for example, to serve a metropolitan area—there is considerable loss of efficiency. Much of this inefficiency comes from the use of the CSMA/CD protocol. In CSMA/CD, a source transmits a packet when the channel is sensed as idle, but this injection of the packet can be known to the other sources only after it has propagated throughout the length of the cable, during which time another source may attempt to transmit on the channel. Thus, the number of bits wasted due to collision is proportional to the propagation delay and the peak bit rate. Also, the need to detect collisions makes it necessary that each packet have a duration equal to at least the round-trip delay. With very large nets and high bit rates, that may represent an unreasonably large minimum number of bits.

Baseband CSMA/CD has been extended to broadband CSMA/CD by several CSMA/CD networks, each in a different frequency band put on the same cable (see, for example, Ref. 4). However, each of these networks operates almost independently, connected usually by a signaling channel. Also, the cable length limitation still applies, making it difficult to use for a metropolitan area. In this paper, we propose schemes that extend the capabilities of both the baseband and the broadband CSMA/CD networks by allowing higher bit rates on a cable, larger cable segments, and at the same time smaller collision delays. We do this by dividing the available bandwidth of the cable into several frequency bands and operating a network (or channel) in each frequency band. Since coaxial cables can easily carry up to 400 MHz, several networks can be accommodated on one cable rather easily. By modulating the baseband data from devices connected to the network to a high-frequency band, total channel bit rates of higher than 50 Mb/s can be obtained easily. However, since the bit rate of each of the nets is kept low, channel inefficiency due to the use of CSMA/CD protocol is not increased. To increase the length of the

cable segment, and at the same time limit the collision delay, we divide the users into communities based on their location and give each user community a network (i.e., one band of frequencies) to transmit most of the time. Thus the "effective" end-to-end delay is reduced although the cable length is increased. The principal characteristics and advantages of our system are the following:

1. Larger channel throughput by using multiple frequency bands.
2. Larger cable length but smaller collision delay by dividing the cable into several parts and operating a network in a given frequency band for each part to be used by a user community, while retaining the listening ability on the full cable length.
3. Complete connection of users between any network.
4. Different grade of service, depending on the complexity of network interface.
5. Restricting most of the high-speed processing to the analog domain and baseband processing to digital domain. Thus, although the network may have channel throughput over 50 Mb/s, individual networks may carry at much lower bit rates.

We describe the system in more detail in the next section, and we develop two protocols for access. A crude analysis is presented at the end to bring out some of the trade-offs for our system.

II. SYSTEM DESCRIPTION

In this section, we describe one possible implementation of our system. A block diagram is shown in Fig. 1. Each terminal has a frequency-agile Radio Frequency (RF) modem that can modulate binary data for transmission on the cable and demodulate the signal from the cable to extract the transmitted binary data. Unlike long distance transmission, since the intent is not to maximize the data transmission rate, simple inexpensive modulation schemes can be chosen with enough separation between the various frequency bands to keep the filtering simple and to reduce the crosstalk. As an example, if modems based on Frequency Shift Keying (FSK) with 1/4 bit/Hz are used, then a cable of bandwidth 300 MHz can support six CSMA/CD networks of peak rate 10 Mb/s each with a guard band of 10 MHz to separate each of them.

Our block diagram in Fig. 1 shows bidirectional transmission, that is, signals injected on the cable at each tap travel in both directions and the amplifiers are bidirectional. It is necessary for the taps to be bidirectional so that they can receive signals from either direction. Although this is a straightforward extension of the baseband CSMA/CD network, bidirectional amplification and taps may present engineering difficulties, particularly at high frequencies. Alternative de-

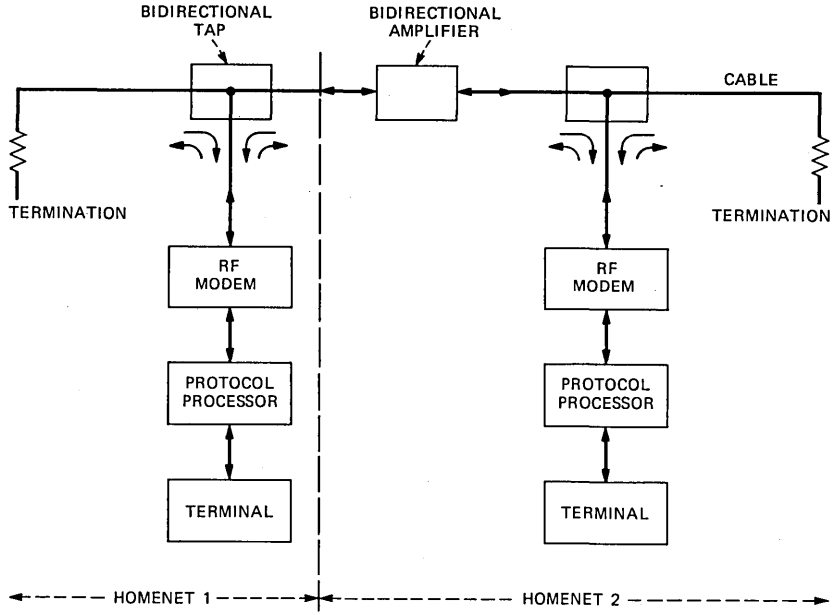


Fig. 1—Single cable system for multiple CSMA/CD networks.

signs using cable-television-type systems and technology are described in Ref. 5.

In Fig. 1, there are two networks (called Homenets) and therefore two frequency bands. The transmitter can, by the agility of the modem, transmit on any of the two frequency bands, and the receiver can receive and demodulate data from both the frequency bands. Terminals attached to homenet 1 transmit mostly on frequency band f1 and those attached to homenet 2 transmit mostly on frequency band f2. If several simultaneous conversations with terminals on different homenets are required (as in the case of a host computer), then a terminal may need multiple transmitters and receivers. Details of the protocols for the access are given in the next section.

III. THE ACCESS PROTOCOLS

Below we give two types of access protocols; the first does not require synchronization of the different terminals, whereas the second does. Some desirable characteristics of any protocol should be noted first. The access delay should be decreased by scheduling the transmission on a net that is either least busy or has the least chance of collision. The load on the different networks should be distributed such that a situation does not arise in which many terminals are trying to transmit on a network and are unable to do so, while the rest of the

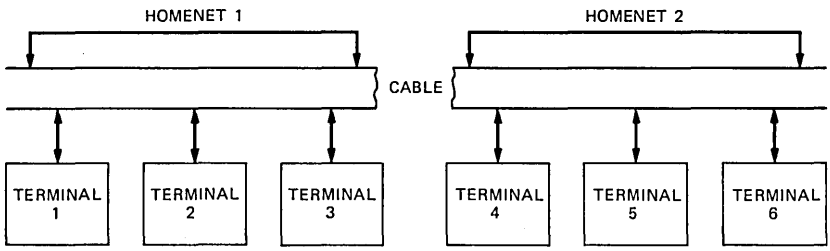


Fig. 2—Homenet assignments based on spatial distance.

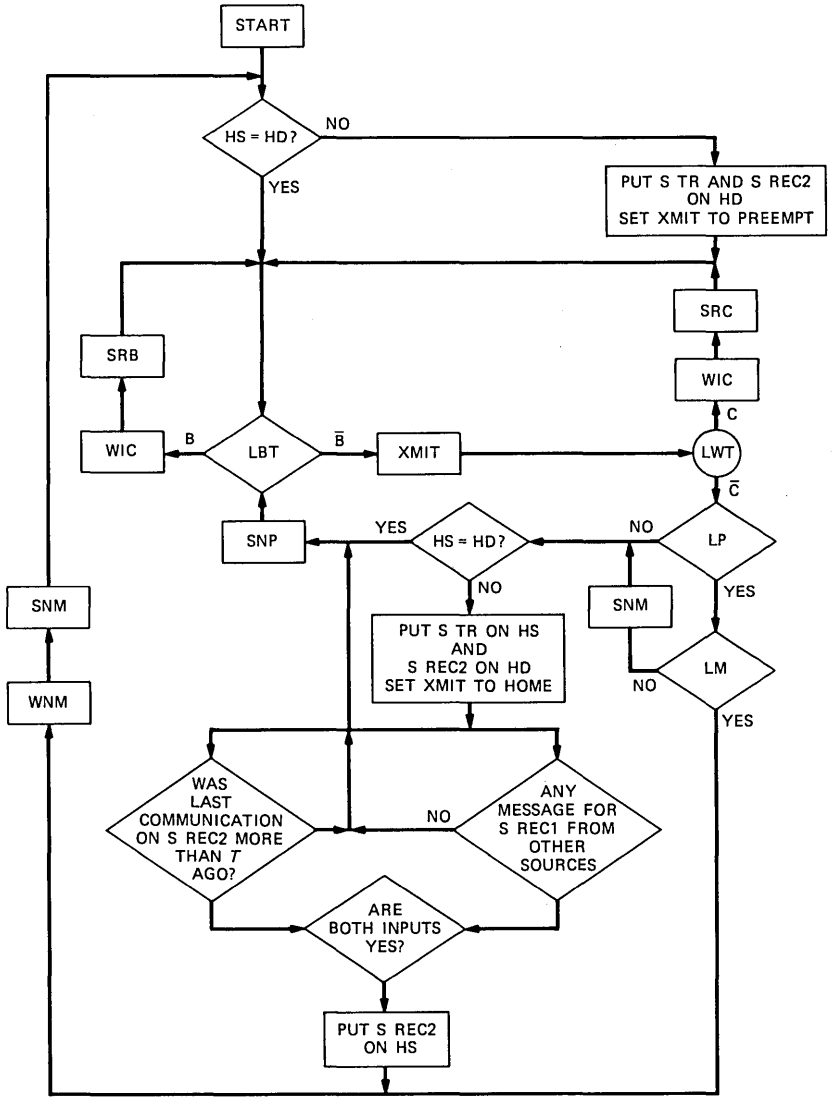
networks are carrying a very light load. In carrier sense multiple-access systems, collisions are a result of the terminals knowing the transmission by other terminals only after the propagation delay. Therefore, by requiring all the terminals that are close to each other to initiate their transmission on a particular network, the collision delay can be reduced considerably. Thus, in both the protocols, each terminal is assigned to a particular network. This network is called the Homenet of the terminal. Homenet assignment is included as a part of the address of the terminal. Each terminal maintains a list of the homenet assignments of the other terminals. The homenet assignment, although made primarily by geographical location, may also take into account the desired connectivity, traffic patterns, etc.

Figure 2 shows a system in which there are two homenets and three terminals per homenet. The transmission initiated by any of terminals 1, 2, and 3 is mostly on homenet 1, whereas the transmission by terminals 4, 5, and 6 is on net 2. Since net 1 and net 2 are on two different bands of frequency, the collisions are now localized. That is, data from terminal 1 can only collide with data from terminals 2 and 3. Since the distance between the taps on the cable of terminals 1, 2 and 3 is much shorter compared with distance between the taps of terminals 1 and 6, the probability of collision and, therefore, of data wasted due to collision is significantly reduced. This increases the channel utilization and decreases the delay. Of course, the protocols must and do allow communication between the terminals on different homenets.

3.1 Protocol 1

The flow diagram for this protocol is shown in Fig. 3. The details follow.

1. Every terminal has at least two receivers and, therefore, is capable of listening to at least two networks. One of these receivers always listens to the homenet. The other receiver is free to listen to any net.
2. An inactive terminal has one of its receivers listening to its



B – CHANNEL BUSY
C – COLLISION DURING TRANSMITTING
HD – HOME NET OF DESTINATION
HS – HOME NET OF SOURCE
LBT – LISTEN BEFORE TRANSMITTING
LM – TEST IF TRANSMITTED MESSAGE WAS LAST
LP – TEST IF TRANSMITTED PACKET WAS LAST
LWT – LISTEN WHILE TRANSMITTING
REC1 – RECEIVER 1
REC2 – RECEIVER 2
S – SOURCE
SNM – SCHEDULE TRANSMISSION OF NEXT MESSAGE
SNP – SCHEDULE TRANSMISSION OF NEXT PACKET
SRB – SCHEDULE RETRY AFTER BUSY CHANNEL
SRC – SCHEDULE RETRY AFTER COLLISION
TR – TRANSMITTER
WIC – WAIT FOR IDLE CHANNEL
WNM – WAIT FOR NEXT MESSAGE
XMIT – TRANSMIT

Fig. 3—Flow diagram for a transmitter working on protocol of Section 3.1.

homenet. The other receiver of that terminal becomes active only after the first receiver starts listening to a network other than its homenet.

3. Any terminal A desiring to transmit to terminal B goes through the following sequence.

a. Determine the homenet of B, say it is net_b.

b. Listen to net_b before transmitting.

c. If net_b is idle (i.e., absence of carrier), transmit net_b carrier for a period T, the two-way propagation delay through the total network. This amounts to a priority preempt on net_b.[6*]

d. If during the second half of the period T there is collision on net_b, then it implies a preemptive transmission from another terminal, not on net_b. In that case, terminal A backs off and attempts a transmission on net_b with reduced probability at the next time slot of T. If there is no collision, then terminal A follows its preempt with a message to terminal B.

e. Terminal B always has one receiver listening to net_b; therefore, it receives information from every collision-free packet on net_b.

f. If packet communications is to be continued, then terminal A starts listening on net_b, terminal B on net_a, and both terminals transmit on their homenets. Thus, if a message has several packets, only the first packet may be transmitted on a homenet different from the homenet of the source; all the subsequent packets are transmitted on their own homenet with standard CSMA/CD protocol with retrial period equal to round-trip delay of the homenet.

g. If at step e terminal B is already in communication with some other terminal on a different network, then it still has a receiver on net_b. If terminal B's transmitter is on net_b (as it normally is, except when it is trying to set up an initial connection with a terminal on a network other than net_b), even if it is in communication with some other terminal, it can send an acknowledgment back to terminal A on net_b. If, however, Terminal B's transmitter is transmitting on a different channel, there may be delay in sending the acknowledgment.[†]

h. If after successful connection there is no communication for a given amount of time and if the receiver on the homenet receives a message for communication from another source, then the other receiver of both the home terminals go back to their respective homenets.

---

* Alternatively, there could be a signaling channel in a different frequency band accessed by all the terminals, and the first packet could be transmitted on the signaling channel. This alternative is attractive for a large network, since it confines collision to a common homenet and, hence, does not constrain minimum packet length. With the utilization very low and the network large, the most appropriate protocol on the signaling channel would be ALOHA.

† If several simultaneous conversations are required, then a terminal may need multiple transmitters and receivers.

i. Many different broadcast modes are possible. If the broadcast to only the terminals in the particular homenet is desired, then data is transmitted on that network only. However, if broadcast is required to all the terminals, then the transmitter has to successfully transmit on each network.

3.2 Protocol 2

The above protocol is reasonable in that it reduces collisions and works well when the traffic is quite bursty, with many terminals trying to transmit messages containing large numbers of small packets frequently. However, when there are large file transfers, use of the homenet by a terminal prevents other terminals with the same homenet from using the channel even though the other networks may be idle. Thus, a reasonable protocol is needed that will share the channels more evenly in the presence of large file transfers by one of the terminals. Protocol 2 attempts to accomplish this at the cost of slightly larger average delay in establishing a connection. In this protocol, networks on which a given group of terminals begin transmitting are switched on a periodic basis. The period is of the order of several packets long (or tens of milliseconds). It thus requires a clock at every terminal, which may be provided from a central clock on a different band of frequencies. Details of the protocol are

1. As in protocol 1, terminals in a given geographical area are grouped together. This grouping is made known to all the terminals (similar to homenets).

2. A group has a homenet and an assigned transmission network. The homenet is fixed, whereas the transmission net changes cyclically. A terminal may, at any time, initiate a transmission only on the transmission network to which its group is then assigned. Once initiated, the transmission may spill beyond this fixed interval, since the packet size is not fixed.* As an example, Fig. 4 shows the case of three groups and three sectors of time.

3. A terminal has at least two receivers. When the terminal is idle, both of them listen to the homenet. After establishing a connection, however, one of the receivers switches to the network on which it has established connection with the other terminal (and, therefore, the network to which it listens changes cyclically) and the other receiver remains at the homenet.

---

* In CSMA/CD the entire packet must be received and CRC checked before the destination address is verified. Thus, if the entire packet is not received before the period ends, the receiver may miss it. To overcome this problem, a separate CRC is provided for the header information and a source terminal starts a transmission sufficiently before the end of a period such that the destination is able to receive the header information before the period ends.

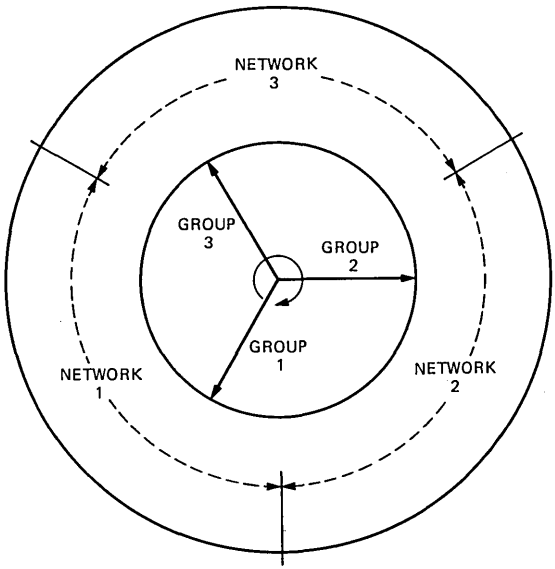


Fig. 4—Protocol of Section 3.2. The assignment of group j terminals to net_k is made by rotating the inner circle at a given speed. The terminals in group j may initiate transmission on net_k if the group j pointer is in the sector corresponding to net_k.

4. Terminal A, desiring to transmit to terminal B, goes through the following sequence:

a. Determine the homenet of terminal B; let it be net_b.

b. Wait for the time period when terminal A is allowed to transmit on net_b.

c. At that time, listen to net_b before transmitting.

d. If net_b is idle (during the assigned time slot), then transmit to terminal B on net_b.

e. If there is no collision, a packet is assumed to have been received by its intended receiver. If there is a collision, terminal A ceases transmission immediately and tries again, using a standard retry strategy, but the additional constraint that its starting time must be when terminal A is allowed to transmit on net_b.

f. If communication is to be continued, then terminal B switches one of its receivers to the net on which terminal A will be transmitting (this will periodically switch), and terminal A will set its receiver on the net on which terminal B will be transmitting.

g. If at step f terminal B is already in communication, then its acknowledgment to terminal A will so indicate.

h. If after a successful connection there is no communication for a given amount of time, receivers of both the terminals go back to their respective homenets.

IV. ANALYSIS

In this section, we present a simplified analysis of the access protocol described in Section 3.1. The assumptions follow those made by Metcalfe and Boggs[7] for a single CSMA/CD network. The symbols used are

P = Number of bits in a packet.
C = Peak channel capacity.
T = Round-trip propagation delay.
Q = Number of stations continuously queued to transmit a packet; represents the total offered load.
N = Number of networks on the frequency-multiplexed cable.
Π_m = Probability that a message from a source contains m packets.

The analysis is used first to calculate the delay versus the channel throughput characteristics of a single CSMA/CD network operating at a peak channel capacity of C. This uses the formulas given by Metcalfe and Boggs. The extension is then made to the case of multiple networks whose total capacity is $(NC)/t$, but is divided equally into N networks. Number t is assumed to be larger than 1. If $t = N$, then the total capacity C is divided equally in N networks, each with capacity C/N. If $t = N/2$, then each net has capacity of $2C/N$, adding to a total of $2C$.

4.1 Single network

The average delay in sending a packet (including transmission time) when Q stations are continuously queued to transmit a packet, is given by

$$D_1 = \frac{P}{C} + T[(1 - 1/Q)^{1-Q} - 1]. \tag{1}$$

This assumes an optimum retry strategy. Since the Offered Load (OL) is Q packets, in terms of bits it is given by

$$\mathrm{OL} = PQ. \tag{2}$$

4.2 Multiple networks

Here if a message from a source contains m packets, then the first packet may be transmitted on a different network, but the subsequent $(m - 1)$ packets will be transmitted on the homenet of the source. Thus, the total transmission time is divided into two parts: time to transmit the first packet and time to transmit the remaining packets. It is assumed for simplicity that the length of each homenet is the same and it is $1/N$ times the total cable length. The following analysis can be easily modified for other configurations.

4.2.1 Time to transmit first packet

If the first packet of a source on homenet j is transmitted on homenet k, then

$$\text{Time for a packet} = \frac{tP}{C} + \text{contention time.} \tag{3}$$

Assume that a source on the jth homenet has probability ρ_{jk} of wanting to communicate with a station on the kth homenet. Further, assume that the number of stations is the same for all nets; distribution, Π_m, of packets is uniform for all messages; and $\{\rho_{jk}\}$ are a constant* independent of j, k. Then the total traffic offered to the kth homenet is

$$q_k = \frac{Q}{N}, \tag{4}$$

of which the offered load from out-of-net is

$$q_k^1 = \frac{N-1}{N} Q\Pi_1 \tag{5}$$

and from within homenet

$$q_k^r = \frac{Q}{N}\left(1 - \frac{N-1}{N}\Pi_1\right). \tag{6}$$

Therefore the time per out-of-net packet is

$$D_{21} = \frac{tP}{C} + T[(1 - 1/q_k^1)^{1-q_k^1} - 1]. \tag{7}$$

4.2.2 Time for subsequent packets

This transmission is on the homenet itself. Therefore, the time for each packet is simply obtained by

$$= \frac{tP}{C} + \frac{T}{N}[(1 - q_k^r)^{1-q_k^r} - 1]. \tag{8}$$

This neglects the traffic generated by first packets of terminals from other homenets. It is assumed that the first packet is a small fraction of the total message and does not result in any significant traffic.

4.2.3 Total time

Since the probability that a message contains k packets is Π_k, the average time per packet is given by

* We have made no measurements of traffic on real systems to justify this assumption. It is made only so that a closed-form expression can be derived for the delay. If other values of ρ_{jk} are more realistic, they can be substituted easily in the equations that follow.

$$D_2 = \frac{N}{Q}(q_k^1 D_{21} + q_k^r D_{2r}) = \frac{tP}{C} + (N-1)\Pi_1 T[(1 - 1/q_k^1)^{1-q_k^1} - 1] + \left(1 - \frac{N-1}{N}\Pi_1\right)\frac{T}{N}[(1 - 1/q_k^r)^{1-q_k^r} - 1]. \quad (9)$$

Thus the delay versus the offered load characteristic will be given by D_2 versus OL.

4.2.4 *Total time with Protocol 2*

With Protocol 2, all packets are sent on homenet, and the delay for optimum strategy is given as

$$D_3 = \frac{tP}{C} + \frac{T}{N}\left[\left(1 - \frac{N}{Q}\right)^{1-Q/N} - 1\right].$$

When Π_1 is small, that is, messages consist on average of many packets, then D_3 and D_2 differ very little from each other.

4.3 *Optimum N* for multiple networks

It is possible to compute the optimum number of nets based on the above expressions for average delay per packet. This can be done for the case when Q (and Q/N) is large and the messages contain a large number of packets, implying that the average delay per packet is dominated not by the first packet, but by the subsequent packets. From eq. (1), for single network, the delay is given by

$$D_1 = \frac{P}{C} + T[(1 - 1/Q)^{1-Q} - 1]$$

since

$$\lim_{k\to\infty}(1 + x/k)^k = e^x \lim_{Q\to\infty}[D_1] = \frac{P}{C} + T\left[\frac{1}{e^{-1}} - 1\right] = \frac{P}{C} + T(e-1). \quad (10)$$

For multiple networks, the delay is approximated by

$$D_2 = \frac{tP}{C} + \frac{T}{N}[(1 - N/Q)^{1-Q/N} - 1]. \quad (11)$$

If $t = N$, then the total capacity C is divided among N nets equally. However, in most cases, the individual N nets may have a capacity such that the capacities add up to more than C. Thus let $t = N/s$. In

this case, each net has a capacity $(sC)/N$ and the total capacity is given by sC. The delay then becomes

$$D_2 = \frac{NP}{sC} + \frac{T}{N}\left[(1 - N/Q)^{1-Q/N} - 1\right]. \tag{12}$$

For large (Q/N) (or small N/Q) we can expand D_2 as a function of N/Q in Taylor's series

$$D_2 \simeq \frac{NP}{sC} + \frac{T}{N}\left[e - \frac{e}{2}\frac{N}{Q} - 1\right] = \frac{NP}{sC} + \frac{T}{N}(e - 1) - \frac{Te}{2Q}. \tag{13}$$

Minimum D_2 with respect to N is achieved when

$$N = \sqrt{\frac{sCT(e - 1)}{P}}. \tag{14}$$

For some typical cases, the optimum N can be worked out as follows:

$s = 1$, $C = 10$ Mb/s, T = propagation delay in seconds on a cable of length 2.5 km $\cong \dfrac{(2.5 \times 10^{-5})}{2}$ sec.

$P = 1000$ bits;

then

$$N \leq 1 \rightarrow N = 1.$$

Thus with standard Ethernet* parameters, from the point of view of average delay per packet, N should be 1.

$s = 1$, $C = 10$ Mb/s, T = propagation delay in seconds on a cable of length 20 km = $(20 \times 10^{-5})/2$ sec.

$P = 500$ bits;

then

$$N \cong 2.$$

This implies that as the length of the cable increases, more networks are required.

---

* Trademark of Xerox Corporation.

$s = 5$, $C = 10$ Mb/s, T = propagation delay in seconds on a cable of length 20 km = $\frac{(20 \times 10^{-5})}{2}$ sec.

$P = 500$ bits;

then

$$N \cong 4.$$

This implies that if each net is operated at 12.5 Mb/s, adding up to a total capacity of $12.5 \times N$ Mb/s, average delay is minimized when $N = 4$.

4.4 Delay versus offered load plots

The average time per packet derived in Section 4.1 was evaluated for a variety of cases and is plotted in Figs. 5 and 6. In all the cases, a

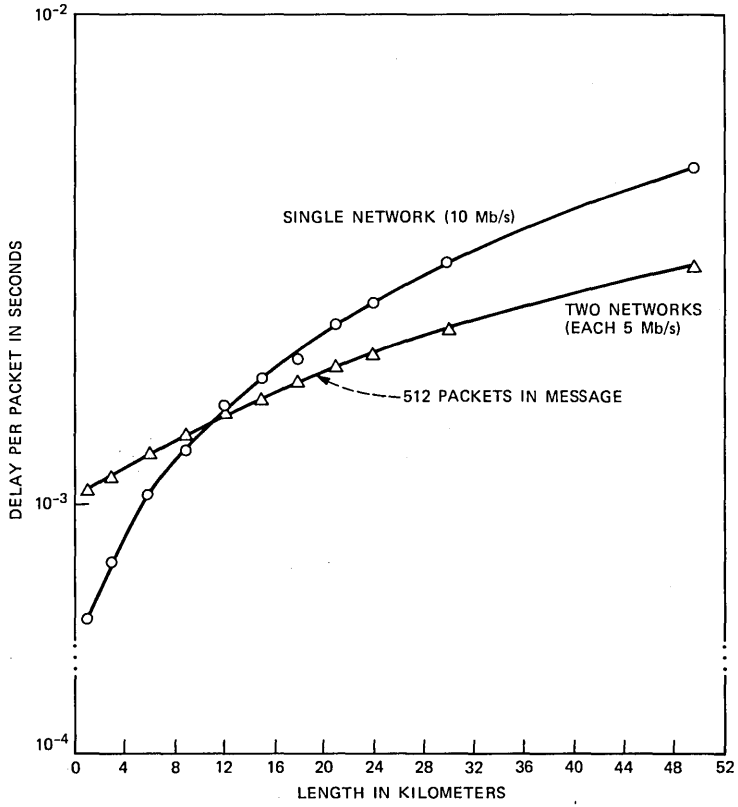


Fig. 5—Average delay per packet versus length of the cable. Performance comparisons are made between two networks on a cable and a single net.

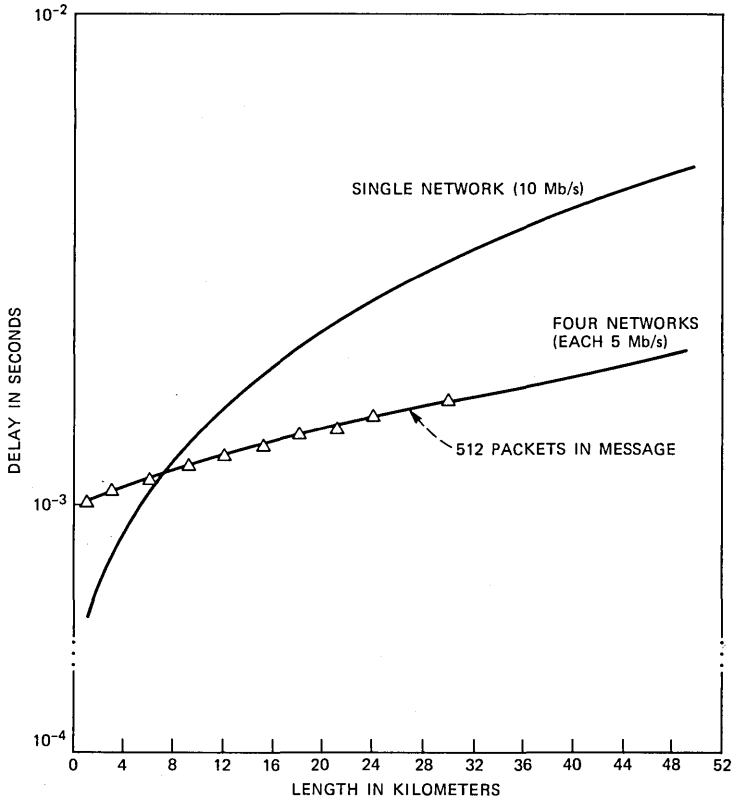


Fig. 6—Average delay per packet versus length of cable. Comparison is made between four networks on a cable and single network.

packet size of 500 bits was used. The curves in Figs. 5 and 6 are for $Q = 200$, that is, 200 packets are continuously queued. It is assumed that a message has 512 packets, that is,

$$\Pi_1 = \Pi_2 = \cdots, = \Pi_{512} = 1/512; \qquad \Pi_k = 0, \qquad k = 513, \cdots.$$

As expected, the delay increases with length, and depending upon the other parameters of the network, the delay corresponding to multiple network becomes smaller than that corresponding to single network if the length is increased beyond a certain value. Figure 7 shows the variation of the average delay with respect to number of networks. The capacity of each of the nets is equal and is such that the total capacity of all the nets adds up to capacity of the single network. We find that, as expected, for the parameters chosen in Fig. 7, the average delay does show a minimum around $N = 4$. This verifies our approximations of the previous section.

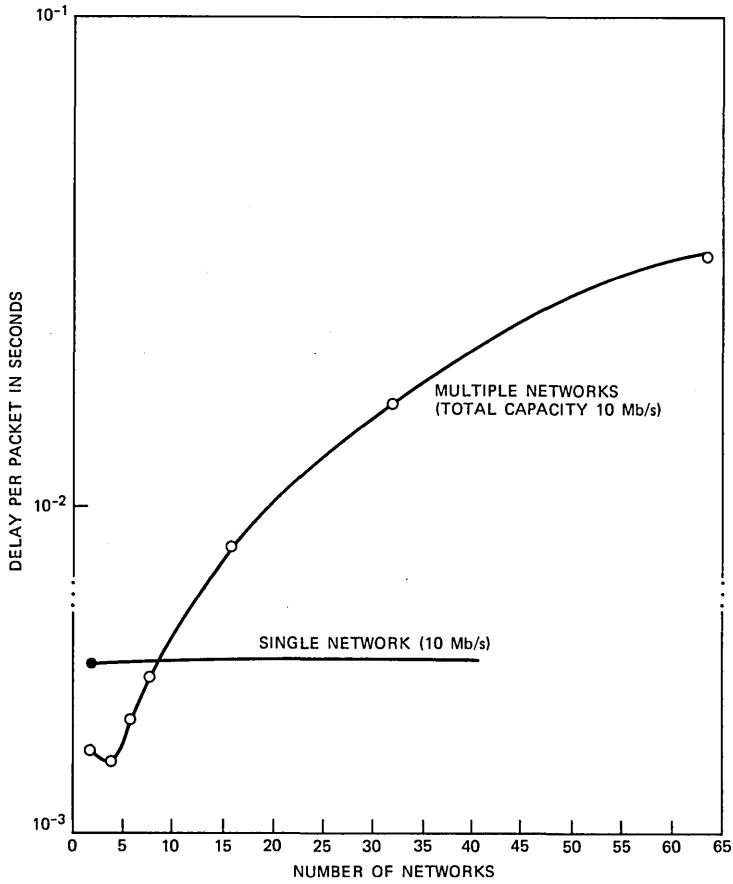


Fig. 7—Average delay per packet versus number of networks. Each network has a capacity of C/N, where N is the number of nets and C is the capacity of the single network.

V. CONCLUSIONS

We have described a broadband local area computer network. It consists of several local area networks whose data is frequency multiplexed on a single cable. The entire cable length is divided into parts; each part is assigned a network and a frequency band for transmission. We have also described two protocols that overcome some of the limitations of the present baseband as well as broadband CSMA/CD networks. Using our schemes, it is possible to increase the channel throughput and the length of the cable network, reduce the delay due to collisions, and at the same time allow complete connectivity among all the terminals and devices logged into any network. Approximate analytical results are also presented to substantiate these claims.

VI. ACKNOWLEDGMENTS

We are grateful to Jerry Foschini and Nick Maxemchuck for many helpful discussions.

REFERENCES

1. A. S. Tanenbaum, *Computer Networks*, New York: Prentice-Hall, 1981, Chap. 7.
2. W. R. Franta and I. Chlamtac, *Local Networks*, Lexington, MA: Lexington Books, 1981.
3. The IEEE Project 802, Local Area Network Standards, CSMA/CD Access Method and Physical Layer Specification, IEEE P802-3-82/0.1 10, December 1982.
4. M. Stahlman, "Inside Wang's Local Net Architecture," Data Commun., (January 1982), pp. 85–90.
5. N. F. Maxemchuck and A. N. Netravali, "A Multifrequency Multiaccess System for Local Access," Proc. ICC 83, Boston.
6. N. F. Maxemchuck, "A Variation on CSMA/CD That Yields Movable TDM Slots in Integrated Voice/Data Local Networks," B.S.T.J., 61, No. 7 (September 1982), pp. 1527–50.
7. R. M. Metcalfe and D. R. Boggs, "Ethernet: Distributed Packet Switching for Local Computer Networks," Commun. ACM, *19*, No. 7 (July 1976), pp. 395–404.

AUTHORS

Zigmantas L. Budrikis, B.Sc., 1955, and B.E. (Hons I, Electrical Engineering), 1957, University of Sydney; Ph.D., 1970, University of Western Australia; P.M.G. (now Telecom Australia) Research Laboratories, 1958–1960; Aeronautical Research Laboratories, Fishermen's Bend, 1961; Electrical Engineering Faculty at University of Western Australia, 1962—. Mr. Budrikis has had a number of visiting appointments: University of California at Berkeley, 1968; AT&T Bell Laboratories, 1972, 1973, 1981, 1983, 1984; TU Munich, 1977. He is interested in problems in communications, man-machine interfaces, and foundations of electromagnetism. Fellow, IE Australia; member, IEEE, Optical Society of America, New York Academy of Science.

Arun N. Netravali, B. Tech. (Honors), 1967, Indian Institute of Technology, Bombay, India; M.S., 1969, Ph.D. (Electrical Engineering), 1970, Rice University; Optimal Data Corporation, 1970–1972; AT&T Bell Laboratories, 1972—. Mr. Netravali has worked on problems related to filtering, guidance, and control for the space shuttle. At AT&T Bell Laboratories, he has worked on various aspects of digital processing and computing. He was a Visiting Professor in the Department of Electrical Engineering at Rutgers University and the City College. He is presently Director of the Computer Technology Research Laboratory. Mr. Netravali holds over 20 patents and has had more than 60 papers published. He was the recipient of the Donald Fink Prize Award for the best review paper published in the Proceedings of the IEEE and the journal award for the best paper from the SMPTE. Editorial board, Proceedings of the IEEE; Editor, IEEE Transactions on Communications; senior member, IEEE; member, Tau Beta Pi, Sigma Xi.

AT&T Technical Journal
Vol. 64, No. 10, December 1985
Printed in U.S.A.

On Binary Differential Detection for Coherent Lightwave Communication

By J. E. MAZO*

(Manuscript received July 31, 1985)

Motivated by the communication problems caused by phase noise in those semiconductor lasers that may be used for fiber-optic data transmission, we consider heterodyned binary Differential Phase-Shift Keying (DPSK) in conjunction with high-rate (short time chip) redundancy as provided by repetition or by more complex coding techniques. In surprising contrast to repetitive coherent phase-shift keying where only a loss of a $2/\pi$ (2 db) in power is incurred in the limit of infinitely many infinitesimal time chips, we show that DPSK requires, in this limit, an infinite number of photons per bit. This is true regardless of the coding scheme used with the DPSK modulation. Next we find the bandwidth expansion that minimizes the number of received photons per bit required to hold the error rate at 10^{-9} for two situations: first for a simple repetition code, and then for a repeated (24, 12) Golay code with maximum likelihood detection. The performance of the latter is assumed to be representative of other optimally detected codes of the same rate, such as convolutional codes with Viterbi decoding. Explicit curves relating required photons per bit to the bandwidth expansion are given for B/R ratios of 0.01 to 10, where B is the laser linewidth and R is the data rate. An example of the results is that for $B/R = 0.1$ and a bandwidth expansion of 10, about 23 photons per bit are required for the repeated Golay code to perform as well as uncoded DPSK without phase noise (which requires 20 photons per bit for $Pe = 10^{-9}$). If $B/R = 0.01$ the bandwidth expansion is reduced to 2, and 12 photons per bit are required, thus outperforming the phase-stable, but uncoded, situation.

* AT&T Information Systems.

I. INTRODUCTION

Semiconductor lasers that may be used for coherent data transmission over optical fibers* have severe phase instabilities. Attempts to transmit a sine wave of frequency f_c result in outputs that are modeled as

$$\cos(2\pi f_c t + \phi + w(t)), \qquad 0 \leqslant \phi \leqslant 2\pi, \tag{1}$$

where $w(t)$ (measured in radians) is a random process representing the phase instability. The process $w(t)$ is usually taken to be a Wiener process and that then implies a Lorentzian [see eq. (4)] line shape for the power spectrum of (1). Such spectra are indeed observed and 3-db bandwidths as large as 10 to 20 MHz have been measured.† These bandwidths imply that the standard deviation of the change in $w(t)$ over a μs can be as large as 4π. Severe problems would be encountered with any conventional coherent detection scheme if one is transmitting data at ten-megabit rates (or lower) rather than gigabit rates. Nevertheless, one may wish to do precisely that, and our purpose here is to mathematically explore one very natural approach, Differentially coherent Phase-Shift Keying (DPSK) in combination with code symbols that have short transmission time. The purpose of using code chips of short duration is to mitigate the effects of phase wander between adjacent chips.

We emphasize that the scheme we are about to investigate is not the only possible one. One could use on-off keying of the optical carrier (1) with photon counting for detection. Theoretically this outperforms DPSK by 3 db, even with a stable transmitting carrier assumed for the latter. However, photon counting is not easy to implement, and practical avalanche photodiodes can introduce 20 db of loss. Thus other techniques, which involve heterodyning, are of interest, in hopes that their implementations can be closer to their own theoretical ideals. For a general survey of lightwave communications we recommend Ref. 1. In Ref. 2 a large number of modulation schemes for coherent optics are evaluated with the main purpose being that of determining the range of B/R values for which coding is not required. We choose here to examine DPSK in detail, but the general behavior of its performance with coding is expected to be representative of modulation formats that do not involve tracking the phase $w(t)$ with a phase-locked loop. The latter was one of the methods considered in Ref. 2 and was shown to be feasible only if $B/R < 0.003$.

Returning to the repetition-DPSK scheme, we note that it might be expected that in the limit of an infinite number of infinitely rapid

* In optical-fiber work, coherent transmission refers to any modulation format where an optical oscillator is required at the receiver.

† The carrier wavelength of interest is 1.55 μm, or f_c is roughly 2×10^{14} Hz.

code chips, the performance would approach something like that of full interval DPSK with a stable oscillator. One of the surprises that we have uncovered (and perhaps the major point of interest of this work) is that this is far from the truth. We show that in this limit an infinite number of incident photons per bit are required for fixed error rate.

Decreasing the number of repetitions while holding the error rate fixed also will ultimately require the photons per bit to become unbounded. This occurs when one approaches (from above) the number of repetitions required to achieve the given error rate, with phase noise being the sole impairment. Consequently, one expects that there will be an optimum bandwidth expansion. This is in fact true, and it is discussed in Section IV, while the similar problem for more sophisticated coding is treated in V.

In Section II we begin by presenting the mathematical details of the model, while the two limiting cases of large and minimum bandwidth expansion are investigated for the repetition code in Section III.

II. MODEL DETAILS

In the representation (1) we take

$$w(t) = \int_0^t n(t')dt' \tag{2}$$

and set the (two-sided) spectral density of the white noise $n(t)$ to be $N/2$. The variance $\sigma_w^2(t)$ of $w(t)$ at time t is then

$$\sigma_w^2(t) = \frac{Nt}{2} \text{ (radians)}^2. \tag{3}$$

The power spectrum of (1) can be calculated in terms of these quantities and is given by[3]

$$\frac{1}{8}G(f - f_c) + \frac{1}{8}G(f + f_c)$$

with

$$G(f) = \frac{N}{\omega^2 + (N/4)^2}, \tag{4}$$

where, as usual, $\omega = 2\pi f$. From (4) the 3-db bandwidth B of the spectrum is

$$B = N/4\pi, \tag{5}$$

and thus from (3) and (5)

$$\sigma_w^2(t) = 2\pi Bt. \tag{6}$$

Often in coherent optics one converts (1) to microwave frequencies (GHz) where conventional signal processing techniques are available. This heterodyning is accomplished by mixing (1) with a locally generated optical wave. The local oscillator also has phase instabilities that add to those of the received signal, and thus in this paper the effective bandwidth at microwave is taken to be double that at optical frequencies. Furthermore, shot noise fluctuations in photon counts during the heterodyning causes a white noise background to be added to the microwave signal. In our model we assume heterodyning to be done, and thus our received unmodulated carrier is modeled as

$$A \cos(\omega_o t + \phi + \phi(t)) + n(t), \tag{7}$$

where $\phi(t)$ is the Wiener process phase noise of variance

$$\sigma_\phi^2(t) = 4\pi B t \tag{8}$$

and $n(t)$ is a Gaussian white noise process of spectral density $N_0/2$.

Assume, momentarily, that $\phi = 0$ and $\phi(t) = 0$ over a bit interval T, and we wish to coherently detect the modulation $\pm A$. We simply multiply by $\cos \omega_o t$, integrate the result for T seconds, and observe the sign of the output. The chance of making an error, Pe, is then

$$Pe = \frac{1}{2} \operatorname{erfc} \sqrt{\frac{E_b}{N_0}}, \tag{9}$$

where

$$\operatorname{erfc} x = 1 - \frac{2}{\sqrt{x}} \int_0^x \exp(-t^2) dt.$$

In (9), $E_b = A^2 T/2$ is the energy per bit in the transmitting signal. When (7) arises, as it does in our case of interest, from heterodyning of an optical wave, E_b/N_0 is not an arbitrary parameter but is (see Ref. 1) numerically equal to the average number of photons per bit in the optical wave at the receiver. An E_b/N_0 corresponding to 18 photons per bit yields an error rate of 10^{-9} for coherent detection.

The quantity E_b/N_0 is also numerically equal to the signal-to-noise ratio (s/n) if the noise power N is measured in a bandwidth equal to $1/T$.

To motivate a later discussion, consider the coherent case further and instead of integrating the received signal over $(0, T)$ and making a decision (what we might unconventionally call soft-decision decoding), we make $n = 2m + 1$ hard decisions based on time chips of length T/n, and then use a majority vote to decide the sign of the transmitted bit. If p_c is the chip error rate, the bit error rate, $Pe(n)$, would be

$$Pe(n) = \sum_{k=m+1}^{2m+1} \binom{2m+1}{k} p_c^k (1-p_c)^{2m+1-k}. \tag{10}$$

Since we want to keep E_b constant, the energy in each chip decreases as E_b/n. For n large, then, we have from (8) and (9)

$$p_c = \frac{1}{2} - \sqrt{\frac{\gamma}{n\pi}}, \tag{11}$$

where

$$\gamma = E_b/N_0. \tag{12}$$

For the binomial distribution represented by (10) and (11), we have that

$$(2m+1)p_c \approx m - \sqrt{\frac{2\gamma}{\pi}}\sqrt{m}$$

is the average number of errors; the variance of this number is

$$(2m+1)p_c(1-p_c) \approx \frac{m}{2}.$$

Since the lower limit on the sum in (10) is $(m+1)$, or $2\sqrt{\gamma/\pi}$ standard deviations above the mean, we have

$$\lim_{n\to\infty} Pe(n) = \frac{1}{\sqrt{2\pi}} \int_{2\sqrt{\frac{\gamma}{\pi}}}^{\infty} \exp(-x^2/2)dx = \frac{1}{2}\operatorname{erfc}\sqrt{\frac{2\gamma}{\pi}}. \tag{13}$$

In (13) we see the ubiquitous $2/\pi$ penalty in s/n for using hard decisions. Equation (13) was derived for coherent detection. When we later consider repetition-DPSK to overcome phase noise, we will be concerned with a corresponding limit for differential detection. Then the fortunate limiting behavior we have just observed will not occur, because, with DPSK, the chip error rate approaches 1/2 more rapidly with n than it does in the coherent case exemplified by (11).

To make a tractable model for DPSK, we assume that the received waveform is

$$A\left(\sum_{-\infty}^{\infty} a_n g(t - nT_c)\right)\cos(\omega_o t + \theta + \phi(t)) + n_x(t)\cos\,\omega_o t - n_y(t)\sin\,\omega_o t. \tag{14}$$

The pulse $g(t)$ is assumed brick-wall Nyquist with $g(0) = 1$, and energy T_c. The Gaussian noise processes $n_x(t)$ and $n_y(t)$ are independent, flat spectrum, and of equal variance $\sigma^2 = N_o/T_c$. As stated earlier, the chip time, T_c, for the repetition code is related to the bit time T via $T_c =$

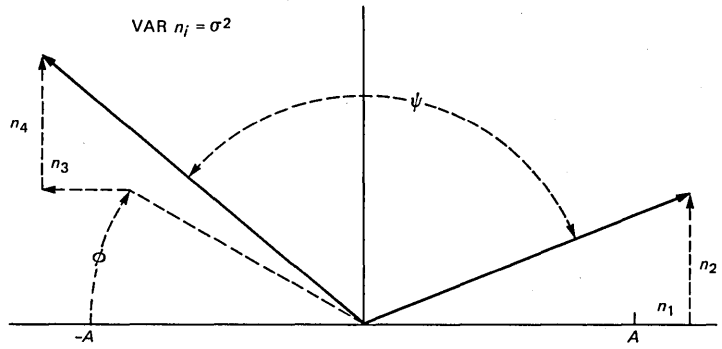


Fig. 1—Geometry of variables for differentially coherent phase-shift keying.

$T/(2m + 1)$, and $(2m + 1)$ repetitions* of the bit are differently encoded into the chips $a_n = \pm 1$.

We imagine recovering the differentially encoded chips a_n by demodulating (14) with $\cos \omega_o t$ and $-\sin \omega_o t$ and sampling the demodulation outputs at the appropriate T_c second intervals. At each sampling instant, then, we obtain a pair of real numbers, which we regard as a point in the plane. In the absence of phase and additive noise, this point would simply be $(\pm A, 0)$, relative to a coordinate system fixed by the unknown phase θ. With noise included, the geometry for two consecutive samples is equivalent to that shown in Fig. 1, drawn for the case of different consecutive a_n. All the components of the two vectors $(A, 0)$ and $(-A, 0)$ are perturbed by additive independent and identically distributed zero-mean Gaussian noise of variance σ^2. Owing to the phase noise, one of the perturbed vectors is also rotated by an angle ϕ, where ϕ itself is a zero-mean Gaussian variable of variance

$$\sigma_\phi^2 = 4\pi B T_c, \tag{15}$$

where B is the laser linewidth. If the angle ψ between the resulting vectors is less than $\pi/2$, we would declare that the two consecutive chips were the same and, in the present case, an error would be made. Of course, for $\psi > \pi/2$ we decide the chips were different.

From (15) we see that as the chip interval decreases, σ_ϕ^2 approaches zero and phase noise will make a negligible contribution to the chip error rate p_c. Note also that the additive noise variance σ^2 increases as T_c decreases.

Before we proceed, a few comments about the model are in order. An equivalent detection procedure is to delay the signal represented

* More generally, our results for the repetition code are unchanged if the bits are mapped into any two complementary patterns of n chips.

in (14) by a chip time, T_c, and sample the product of the delayed and undelayed signals. The resulting decision statistic is identical to the angle that we consider, being a representation of it as a quadratic form equal to the inner product of two noisy vectors. Secondly, a more realistic description of the signal uses a pulse $g(t)$ that has unit value over the interval T_c and is zero elsewhere. The bandwidth of the signal then is not precisely defined, but if one estimates the bandwidth of a flat front-end filter required for noise filtering to be $1/T_c$ (and ignores any intersymbol interference), then the numerical results are unchanged. Finally, we note that Salz[2] integrates the product of the signal and the delayed signal, rather than simply sampling. This complicates the analysis considerably. We have not attempted to investigate the difference in detail over the full range of B/R values, but we note the following. Salz estimates the B/R value required in order that DPSK detection suffers only 1-db degradation compared with $B/R = 0$ and finds $B/R < 0.003$ is sufficient. We calculate this precisely for our model and find $B/R < 0.002$. This suggests that the postdetection processing provided by the integrator might not be significant.

We use (10) later to calculate the bit error rate for repetitive DPSK (with an appropriate p_c). It may be objected that the use of this formula for the repetitive code is not rigorously justified, since samples used for detecting consecutive chips have one noise sample in common and thus the chip detection probabilities are not independent. This objection is easily overcome by assuming that the chips for two successive bits are interleaved in the manner *abab* $\cdots$.

III. LIMITING CASES

In this section we treat two limiting cases of repetitive DPSK. One case is concerned with a very large number of rapid repetitions. Here since the chip interval T_c becomes small, the phase noise is neglected, and for fixed E_b the chip s/n is small. The other case examines the minimum number of repetitions required to achieve a fixed error rate if phase noise were the only impairment. To approach this limit would require a large E_b provided that more than one repetition is required.

We begin with the high repetition rate limit. It is well known that the error probability for DPSK (see Ref. 4) is

$$\frac{1}{2}\exp(-\rho), \tag{16}$$

where ρ is the s/n. In terms of the parameters that apply to Fig. 1,

$$\rho = \frac{A^2}{2\sigma^2} = \frac{A^2}{2}\frac{1}{N_0(1/T_c)} = \frac{E_c}{N_0}, \tag{17}$$

where $E_c = E_b/n$ is the energy per chip. For small E_c (large n) (16) yields for the chip error probability, p_c,

$$p_c = \frac{1}{2}\left(1 - \frac{\gamma}{n}\right). \tag{18}$$

Note the difference in behavior for the differentially coherent problem [see (18)] versus the coherent one [see (11)]. In (18) the chip error rate approaches 1/2 as $1/n$, whereas in (11) the behavior is as $1/\sqrt{n}$. Thus, for DPSK the lower limit of (10) is $O(1/\sqrt{n})$ standard deviations above the mean, and for large n we have that the bit error probability $Pe(n)$ obeys

$$\lim_{n\to\infty} Pe(n) = \frac{1}{2}. \tag{19}$$

In essence, then, we have that if, in Fig. 1, only one of the vectors is noisy and ϕ is set to zero (coherent phase-shift keying), then $Pe(n)$ is small in the limit of many repetitions with constant E_b, but if both are noisy (DPSK), $Pe(n)$ limits to 1/2.

For the second limiting case, when the only perturbing influence to the transmission is the Gaussian phase noise variable ϕ, the chip error rate is

$$p_c = \frac{2}{\sqrt{2\pi}\sigma_\phi} \sum_{k=0}^{\infty} \int_{(1+4k)\pi/2}^{(3+4k)\pi/2} \exp(-\phi^2/2\sigma_\phi^2(n))d\phi$$

$$= \sum_{k=0}^{\infty} \left[\operatorname{erf} \frac{(3+4k)\pi}{2\sqrt{2}\sigma_\phi(n)} - \operatorname{erf} \frac{(1+4k)\pi}{2\sqrt{2}\sigma_\phi(n)}\right], \tag{20}$$

where $\sigma_\phi^2(n)$ is given by

$$\sigma_\phi^2(n) = 4\pi BT_c = \frac{4\pi BT}{(2m+1)} = \frac{4\pi B}{nR}. \tag{21}$$

In (20), p_c is explicitly a decreasing function of the number of repetitions via the phase noise variance σ_ϕ^2. Further, if $Pe(n)$ is fixed in (10), that expression implicitly determines p_c as an increasing function of n. Requiring that both (10) and (20) determine the same value of p_c fixes the number of repetitions required to achieve the bit error probability Pe. Including additive noise in the calculations can only increase the required repetition rate for fixed Pe. Setting $Pe = 10^{-9}$, we have computed the number of repetitions, $\bar{n}$, required when phase noise is the sole impairment. The probability $\bar{p}_c$ is the chip error probability (20) for $\bar{n}$ repetitions. Both are displayed in Table I for several values of B/R. Although our main focus will not be on numerical values of p_c, it is worth emphasizing that throughout this paper values of p_c above 0.1, and even approaching 1/2, are possible for the larger values of B/R.

Table I—Minimum number of repetitions $\bar{n}$ required so that bit error rate does not exceed 10^{-9} with phase noise as sole impairment

B/R	$\bar{n}$	$\bar{p}_c$
0.01	3	2×10^{-5}
0.1	5	0.0017
1	21	0.042
10	83	0.20

The ratio signal-bandwidth/laser-linewidth equals nR/B. From Table I we see that this number is sufficiently high so that the implicit neglect of the wideband filtering on the phase noise that was made when writing the model [see (14)] seems justified for the parameters of interest.

IV. OPTIMUM REPETITION RATE

In this section we do a general investigation of DPSK detection with repetitions, including both phase noise and additive noise as impairments. Our main interest will be to determine the optimum repetition rate and the corresponding bits per photon required to achieve a bit error probability of 10^{-9}.

There are probably several useful general expressions for the chip error rate p_c when both phase noise and Gaussian noise are present. We shall work with the one given in (22), namely,

$$p_c = \frac{1}{2} - \frac{\rho \exp(-\rho)}{2} \sum_{s=0}^{\infty} \frac{(-1)^s}{(2s+1)} \cdot \left[I_s\left(\frac{\rho}{2}\right) + I_{s+1}\left(\frac{\rho}{2}\right) \right]^2 \exp\left(- \frac{(2s+1)^2 \sigma_\phi^2(n)}{2} \right), \quad (22)$$

where ρ is the chip (s/n) [see (17)], $\sigma_\phi^2(n)$ is the phase noise variance with n repetitions [see (21)], and $I_s(\rho/2)$ are the modified Bessel functions given by

$$I_s(x) = \left(\frac{x}{2}\right)^s \sum_{j=0}^{\infty} \frac{(x^2/4)^j}{j!(j+s)!}. \quad (23)$$

An expression similar to (22) for constant phase error was first derived by Blachman[5] and is given in eq. (62) of Ref. 4. Performing a simple average when this angle has Gaussian statistics yields (22). An independent derivation is given in the Appendix.

The successive terms of the sum in (22) decrease in magnitude, and hence, as for any such alternating series, the first neglected term

bounds the error. We find that 15 terms in double precision (single precision on a Cray I) is enough to duplicate (16) when $\sigma_\phi^2(n) = 0$, and $\rho \leq 20$ ($p_c \geq 10^{-9}$).

We first use (22) to calculate the deterioration in p_c as B/R increases. Table II shows some results for $\rho = 20$, $n = 1$. We note that B/R = 0.002 yields about a 1-db degradation.

Next in Figs. 2 through 5 we plot as a function of the number of repetitions, n, the number of photons per bit, γ, which are required at the receiver to maintain a bit error rate of 10^{-9} for B/R values of 0.01 to 10. This is done using (10) and (22) in the following way. First n is chosen and the required p_c is determined from (10). For fixed B/R and known n, σ_ϕ in (21) is known, and (22) is then used to compute the chip s/n ρ that will achieve that p_c. Finally, $\gamma = n\rho$. These figures show quantitatively how the optimum value of n (and the correspond-

Table II—p_c vs. B/R for $\rho = 20$, $n = 1$

B/R	p_c
0	10^{-9}
0.001	6×10^{-9}
0.002	4×10^{-8}
0.01	2×10^{-4}
0.1	0.17
1	0.498841
10	0.49999...

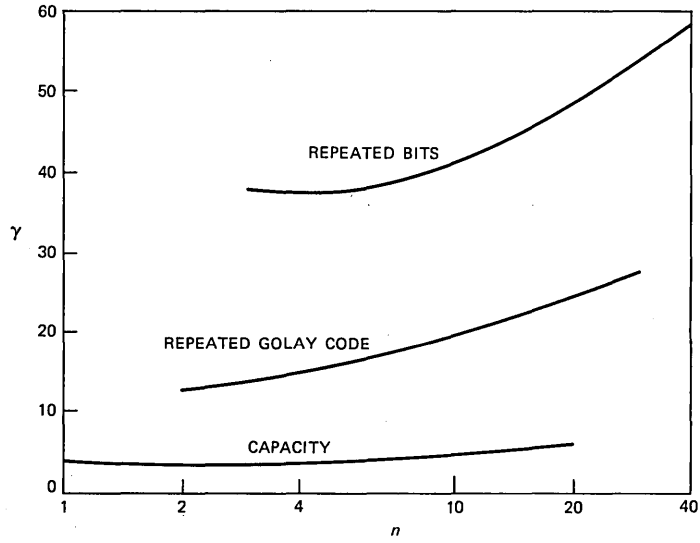


Fig. 2—Bits per photon versus bandwidth expansion for $B/R = 0.01$.

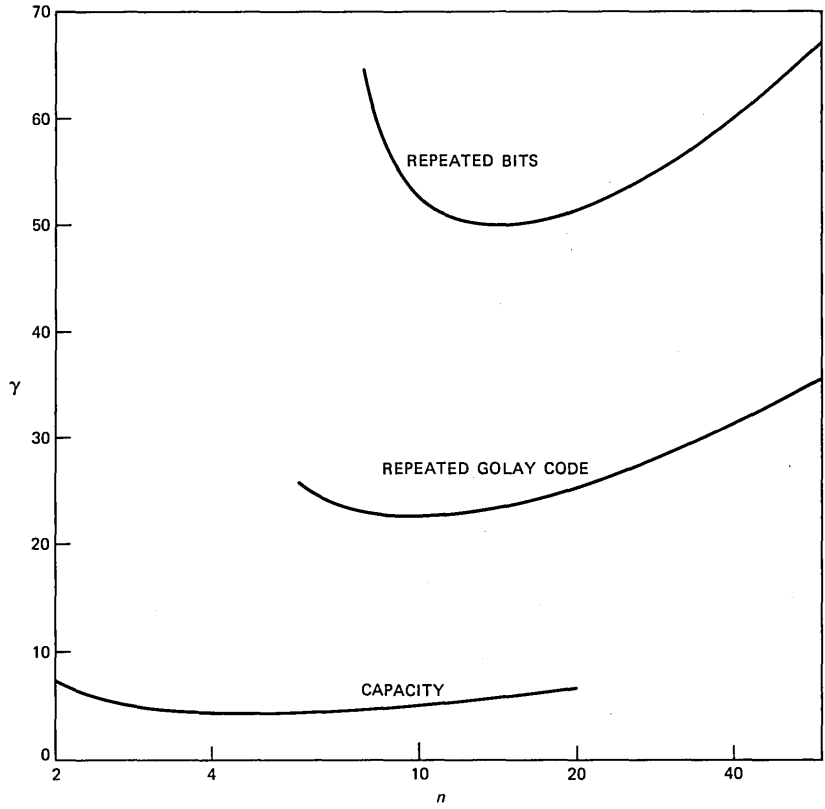


Fig. 3—Bits per photon versus bandwidth expansion for $B/R = 0.1$.

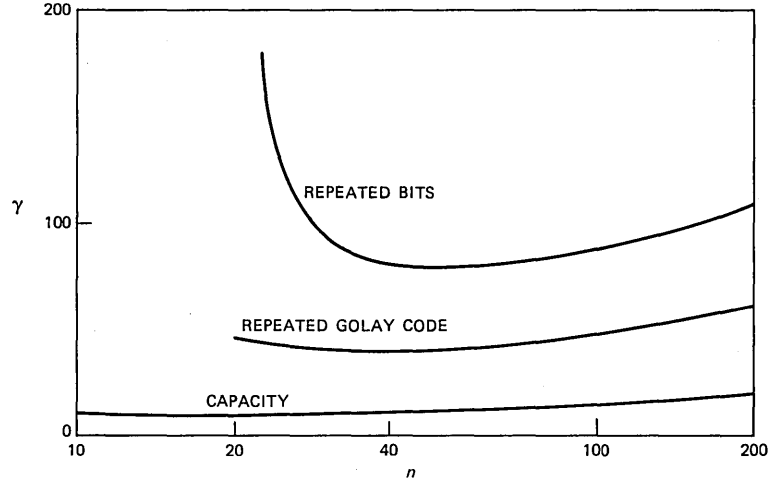


Fig. 4—Bits per photon versus bandwidth expansion for $B/R = 1$.

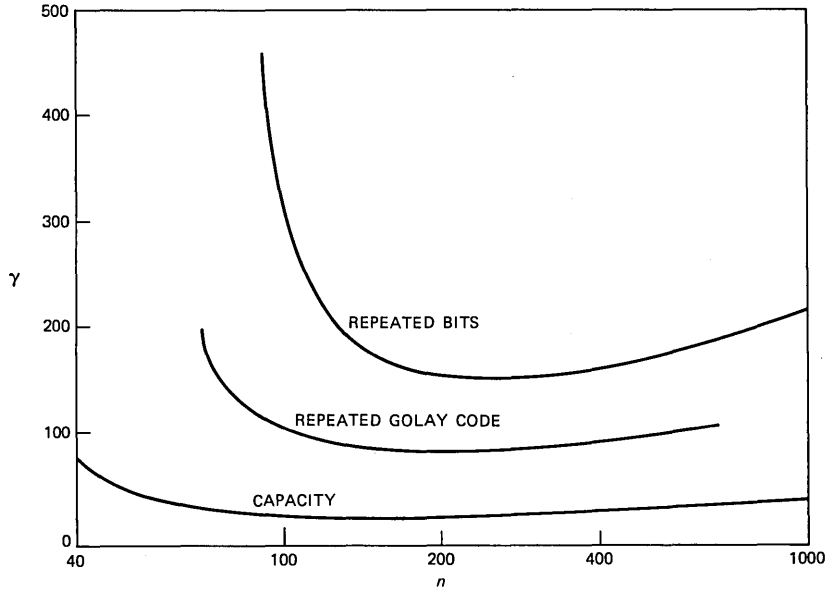


Fig. 5—Bits per photon versus bandwidth expansion for $B/R = 10$.

Table III—Optimum bandwidth expansion for repetition code ($Pe = 10^{-9}$)

B/R	n^*	p_c^*	γ^*	$\gamma^*/20$ (db)
0.01	5	4.6×10^{-4}	37	2.7 db
0.1	15	0.0255	50	4.0
1	49	0.141	79	6.0
10	239	0.313	157	8.9

ing value of γ) increase with B/R. From these figures, we derive Table III, which lists n^*, the optimum repetition rate, and γ^*, the minimum number of photons per bit required to hold the error rate at 10^{-9}. The last column in Table III compares γ^* with 20, the number of photons per bit required at this error rate for DPSK with no repetitions and a stable phase.

We note that the curves are often fairly flat around the minimum, and hence less bandwidth may be used without significantly increasing the required number of photons per bit.

V. CHANNEL CAPACITY AND GOLAY CODES FOR HETERODYNED DPSK

The repetition (or complementary) code for transmitting at rate $R = 1/T$ is but one way to use the discrete time Binary Symmetric

Table IV—Minimum and optimum bandwidth expansion for capacity and repeated Golay code

B/C	$\bar{n}$	$\bar{p}_c$ capacity	n^*	γ^*	p_c^*
0.01	<1.0001	~10^{-5}	2.2	3.192	0.126
0.1	1.7	0.0828	4.3	4.29	0.224
1	6.3	0.265	20	8.83	0.369
10	34.5	0.397	146	24	0.451
B/R	Repeated (24, 12) Golay Code $P_w = 1.67 \times 10^{-10}$				
0.01	—	—	2	12.5	0.00237
0.1	4	0.00507	10	22.8	0.0691
1	14	0.0973	36	38	0.221
10	60	0.277	194	85	0.372

Channel (BSC) that we have to work with.* Therefore, to see what is theoretically possible, we next consider the required number of photons per bit that would be required to transmit at channel capacity. If C is capacity in bits per second, we will fix B/C and plot γ versus n where the chip time is $T_c = 1/(nC)$. That is, the binary code using the chips should have a capacity $\bar{C} = 1/n$ bits per chip. Here n is the bandwidth expansion and need not be an integer.

The required chip error probability is found by equating the channel capacity for the BSC to $1/n$, that is,

$$\bar{C} \equiv 1 + p_c \log_2 p_c + (1 - p_c)\log_2(1 - p_c) = \frac{1}{n}. \tag{24}$$

The chip s/n ρ required to yield this p_c is then computed from (22), and $\gamma = n\rho$.

Curves for $B/C = 0.01$ to 10 are presented in Figs. 2 through 5, and summarized in the first half of Table IV. Once again, a feature is the existence of an optimum chip rate. The necessity of this is easily argued. The lowest value of n possible is determined by the phase noise alone, which causes p_c to increase as n decreases. Eventually, the capacity $\bar{C}$ drops below $1/n$, and this fixes the minimum $n = \bar{n}$. But to approach the pure phase noise situation, γ must increase indefinitely as $\bar{n}$ is approached from above. To see why γ must increase as n is very large, consider plotting $\bar{C}$ versus n with γ fixed. Then, as we have seen [see (18)], $p_c = (1/2) - O(1/n)$ owing to Gaussian (shot) noise. Setting $p_c = (1/2) - \epsilon$ and expanding the left side of (24) in powers of ϵ, we have, to lowest order,

$$\bar{C} = \frac{2\epsilon^2}{ln2} \text{ bits/chip.} \tag{25}$$

* In fact, interleaving (explained at the end of Section II) creates two parallel BSCs. The total capacity is the sum of the capacity of each and is the same as the capacity calculated here as if all chip errors were independent.

Since $\epsilon = O(1/n)$ and we have $O(n)$ chips per second, the capacity measured in bits per second vanishes like $1/n$. To avoid this, γ must increase.

Finally, we consider specific coding schemes that are more involved than repetition. We explicitly consider the extended binary (24, 12) Golay code. Thus for block length 24, 12 information bits are specified, yielding a rate 1/2 code. This is a linear code and any code word has 759 nearest neighbors at minimum distance $d_{\min} = 8$. The coding scheme that we consider is simply to construct low-rate code words by repeating a given Golay code word J times, make hard decisions on the chips at the receiver (assuming appropriately interleaved DPSK), and then use maximum likelihood decoding on the resulting binary code word of length $24J$. Note that the new code has $d_{\min} = 8J$ and that the code rate has decreased to $1/(2J)$.

Assuming that a word error results, on the average, in six bit errors, we set the word error rate, P_w, to be $(1/6)\ 10^{-9}$. If p_c is the chip error rate, then the union bound yields

$$P_w \leqslant 759 \sum_{i=4J}^{8J} \binom{8J}{i} p_c^i (1 - p_c)^{8J-i} + \binom{\text{higher}}{\text{terms}}, \tag{26}$$

where "higher terms" represents the probability of decoding into words further away than the minimum distance. These terms are neglected for the low word error probability of interest here.

We have in mind that actual attempts at coding would use repeated convolutional codes and Viterbi decoding. Our introduction of Golay codes is simply to make the analysis easier, but overall performance gains for the same repeated code rate are expected to be the same. In fact, Chase[6] finds that a (properly chosen) repeated 16-state convolutional code performs slightly better than a repeated Golay code. The effectiveness of repeating an appropriately chosen code to obtain a good low-rate code was, in fact, proposed by Chase,[6] who was concerned with codes when p_c is large, as is often the case for the present problem.* Perhaps better rate $1/(2J)$ convolutional codes exist than can be generated by repeating the symbols of a given one, but Chase shows that, at least from a minimum distance point of view, a repeated rate 1/2 convolutional code (suitably chosen) is close to optimum for code rates at least as small as 1/128.

Returning to the Golay code, note that J repeats (of any rate 1/2 code, in fact) corresponds to a bandwidth expansion of $n = 2J$.

Calculations of required photons per bit versus bandwidth expansion

* A point emphasized in Ref. 6 is that for $p_c > 0.25$ and asymptotically large block length, bounded distance algebraic decoders cannot operate ($Pe \to 1$) and maximum likelihood decoders must be used.

are done in a similar manner, as earlier. The bandwidth expansion $n = 2J$ is picked, p_c is found from (26) with $P_w = 1.67 \times 10^{-10}$, and then (22) is used to solve for ρ. The improvements obtained over simple repetition are displayed in Figs. 2 through 5, and essential features of the results are given in the second half of Table IV. In general, we see that the optimum bandwidth expansion is less than with repetition, and, of course, so is the required number of photons per bit. An important feature is that for $B/R = 0.01$ and 0.1, the required number of photons per bit is less, or comparable to, that required when no phase noise is present and no coding is used. In these cases the bandwidth expansion is relatively modest as well.

VI. CONCLUSION

To reduce the harmful effects of phase noise, coding schemes that use DPSK detection of code symbols having short duration were examined. We first considered in detail a simple repetition code and determined the optimum bandwidth expansion that minimized the number of received photons per bit required for an error probability of 10^{-9}. If $B/R = 0.01$, we found $n^* = 5$ and $\gamma^* = 37$. The corresponding numbers for a repeated Golay code were $n^* = 2$ and $\gamma^* = 12.5$. By contrast, 20 photons per bit are required for phase stable but uncoded DPSK transmission.

The performance of the repeated Golay code is intended to be typical of that obtained with other moderate coding efforts using maximum likelihood detection. In particular, it should be comparable to a repeated 16-state convolutional code (of the same overall rate) with Viterbi decoding. It is our understanding that the fastest commercially available Viterbi decoders operate at about 20 Mb/s (with 64 states).

VII. ACKNOWLEDGMENT

My interest in, and exposure to, this subset of communication problems in fiber optics is due to several enjoyable and informative discussions with Paul Henry and Jack Salz.

REFERENCES

1. Paul S. Henry, "Lightwave Primer," IEEE J. Quantum Electron., *QE-21* (December 1985).
2. J. Salz, "Coherent Lightwave Communications," AT&T Bell Lab. Tech. J., this issue.
3. J. E. Mazo and J. Salz, "Spectra of Frequency Modulation With Random Waveforms," Inform. Contr., *9*, No. 4 (August 1966), p. 419.
4. R. F. Pawula, S. O. Rice, and J. H. Roberts, "Distribution of the Phase Angle Between Two Vectors Perturbed by Gaussian Noise," IEEE Trans. Commun., *COM-30* (August 1982), pp. 1828–41.
5. N. M. Blachman, "The Effect of Phase Error on DPSK Error Probability," IEEE Trans. Commun., *COM-29* (March 1981), pp. 364–5.

6. David Chase, "Code Combining," IEEE Trans. Commun., *COM-33* (May 1985), pp. 385–93.
7. I. S. Gradshteyn and I. M. Ryzhik, *Tables of Integrals, Series, and Products*, Fourth ed. New York: Academic Press, 1965.

APPENDIX

Derivation of (22)

We begin by deriving a Fourier series for the angular distribution $g(\theta_1)$ of a vector $(A, 0)$ perturbed in each component by additive, zero-mean, independent and identically distributed Gaussian noise of variance σ^2. In what follows,

$$\rho = \frac{A^2}{2\sigma^2} \tag{27}$$

and

$$I_k(z) = I_{-k}(z) = \left(\frac{z}{2}\right)^k \sum_{j=0}^{\infty} \frac{(z^2/4)^j}{j!(j+k)!}, \qquad k = 0, 1, \ldots \tag{28}$$

are the modified Bessel functions. We have

$$\exp(z \cos t) = \sum_{k=-\infty}^{\infty} I_k(z) \cos kt. \tag{29}$$

Expressing the joint density of the components of $(A + n_1, n_2)$ in polar coordinates (r, θ_1) and integrating over the r variable after applying (29), we obtain, after a variable change,

$$\begin{aligned} g(\theta_1) &= \frac{\exp(-\rho)}{2\pi} \sum_{k=-\infty}^{\infty} \cos k\theta_1 \int_0^{\infty} \exp(-x) I_k(2\sqrt{\rho x}) dx \\ &= \frac{1}{2\pi} + \frac{\exp(-\rho)}{\pi} \sum_{k=1}^{\infty} \cos k\theta_1 \int_0^{\infty} \exp(-x) I_k(2\sqrt{\rho x}) dx, \\ &\qquad -\pi \leqslant \theta_1 \leqslant \pi. \end{aligned} \tag{30}$$

In our problem, we have a second noisy vector that is also perturbed by rotation through ϕ, ϕ being zero-mean Gaussian. The modulo 2π angle θ_2 of this vector has density $h(\theta_2)$, where

$$h(\theta_2) = E_\phi g(\theta_2 - \phi), \qquad -\pi \leqslant \theta_2 \leqslant \pi, \tag{31}$$

E_ϕ being expectation with respect to ϕ. In (31), $g(\theta_2 - \phi)$ is evaluated by the periodic extension of $g(\theta)$. We are assuming here that the two consecutive chips are identical, but the error probability is the same when two consecutive chips have opposite signs.

Since θ_1 and θ_2 are independent, the difference angle $\psi = (\theta_2 - \theta_1) \bmod 2\pi$ has density $p(\psi)$ [see eq. [6] of Ref. 4], where

$$p(\psi) = E_\phi \int_{-\pi}^{\pi} g(\theta_1)g(\theta_1 + \psi - \phi)d\theta_1. \tag{32}$$

Finally, using the symmetry $p(\psi) = p(-\psi)$, the chip error probability is

$$p_c = 2 \int_{\pi/2}^{\pi} p(\psi)d\psi. \tag{33}$$

Performing the θ_1 and then the ψ integrations gives

$$p_c = \frac{1}{2} - \frac{2}{\pi} \exp(-2\rho) \sum_{s=0}^{\infty} \frac{(-1)^s}{2s + 1} \cdot \left(\int_0^{\infty} dx \exp(-x) I_{2s+1}(2\sqrt{\rho x}) \right)^2 \exp\left(- \frac{(2s + 1)^2 \sigma_\phi^2}{2} \right). \tag{34}$$

A final use of eq. 6.614 (1) from Ref. 7 to evaluate the integral in (34) results in (22).

AUTHOR

James E. Mazo, B.S. (Physics), 1958, The Massachusetts Institute of Technology; M.S. (Physics), 1960, and Ph.D. (Physics), 1963, Syracuse University; Research Associate, Department of Physics, University of Indiana, 1963–1964; AT&T Bell Laboratories, 1964–1985; AT&T Information Systems, 1985—. At AT&T Bell Laboratories, Mr. Mazo had been concerned with theoretical problems in data transmission. He now supervises the Data Theory group at AT&T Information Systems.

AT&T Technical Journal
Vol. 64, No. 10, December 1985
Printed in U.S.A.

Performance Signatures for Dual-Polarized Transmission of M-QAM Signals Over Fading Multipath Channels

By M. KAVEHRAD and C. A. SILLER, JR.*

(Manuscript received August 6, 1985)

Performance signatures for dual-polarized transmission of M-state quadrature amplitude-modulated signals over dispersive multipath digital radio channels are theoretically derived in this work. We report on two major findings. Firstly, we show that for the assumed propagation model, a cross-coupled interferer exhibits noiselike behavior and impacts on digital radio outage time in direct relation to its power level. Secondly, our theoretical finding is based on a new application of performance signature curves for two cross-coupled multipath channels. This treatment permits the prediction of multipath-induced digital radio outage for specified ratios of power in the copolarized and cross-coupled signals. Theoretical findings are qualitatively supported by measured performance signatures obtained from a laboratory simulation of the model.

I. INTRODUCTION

The last decade has witnessed a surge of interest in terrestrial digital radio transmission, with the newer high-capacity systems relying almost exclusively on single-polarization microwave transmission of M-state Quadrature Amplitude-Modulation (M-QAM) signals. In these digital radio systems, the performance degradation associated with multipath propagation has been of paramount importance and

*Authors are employees of AT&T Bell Laboratories.

the subject of considerable prior investigations. For single-polarization systems, the effects of multipath are well understood;[1,2] suitable countermeasures—particularly adaptive equalization—have been studied;[2-4] and these countermeasures are now widely deployed in a variety of digital radio systems.[5,6]

When compared with single-polarization systems, dual-polarized operation obviously engenders economic and efficient spectrum-utilization advantages. Unlike single-polarization transmission, however, an understanding of the effects of multipath propagation on this latter mode of operation is still in its infancy. For single-polarization operation, countermeasures to frequency-selective fading mitigate intersymbol interference (ISI) in the presence of Gaussian noise; the transmission of orthogonally polarized signals over the same bandlimited facility is similarly vulnerable to the effects of ISI and noise, but now Cross-Polarization Interference (CPI), normally suppressed by the polarization selectivity of the receiving antenna, is also present. Consequently, an optimal receiver must recover the transmitted signal in the presence of ISI, CPI, and noise.

Kavehrad[7] has previously studied dual-polarized M-QAM transmission over *nondispersive* media. He concluded that satisfactory transmission is not feasible without some form of cross-polarization interference cancellation. Furthermore, the work showed that in an optimal detection process, the total noise and CPI power must be adaptively minimized. In this paper we extended the scope of the previous work by considering dual-polarized transmission of M-QAM signals over *dispersive* fading channels like those experienced in line-of-sight applications.

The dual-polarized channel is modeled using Rummler-like[8] multipath transfer functions to describe both the copolarized and cross-coupled paths. The transfer functions emulate snapshots of independent multipath fading events on the copolarized and cross-coupled paths, which, in the presence of noise, limit the achievable error rate at the receiver. Our results are predicted on the major assumption that the two simultaneous fading events are statistically independent. Performance "signatures," defined by a locus of fade notch depths and frequency positions corresponding to a 10^{-3} error rate,[9] are used as a system performance measure since they are conveniently related to digital radio outage.[10]

For the assumed propagation model, we find that the cross-coupled interfering signal exhibits a noiselike behavior. That is, the power loss associated with a cross-coupled signal subject to flat or dispersive fading brings about an actual reduction in system outage time. Apparently, the deleterious effect of ISI contributed by dispersive fading of the cross-polarized interference is more than offset by the power loss

of the same associated with the fading event, consequently improving net system outage. Furthermore, this finding is supported by laboratory measurements in which the fading of two independent 16-QAM signals is simulated.

The theoretical and experimental finding cited above is based on a novel application of the aforementioned system performance signature curves for the propagation model adopted in this work. In particular, it permits an immediate comparison of multipath-induced dual-polarized digital radio outage for specified ratios of power in the copolarized signal and cross-coupled interferer.

In the following section we begin by providing a complete model of the dually polarized communication channel, including the influence of frequency-selective fading. In Section III the computational methods and performance measurers are discussed. Numerical results and laboratory measurements that support our finding are presented in Section IV. Our conclusions are summarized in Section V.

II. ANALYTICAL MODEL

2.1 Channel description

The system model for dual-polarized operation is illustrated in Fig. 1. Two independent data sources (one for each of the dual-polarized channels) are assumed to generate complex-valued symbols at the baud rate, $1/T_s$, where T_s is a symbol period. We denote these complex

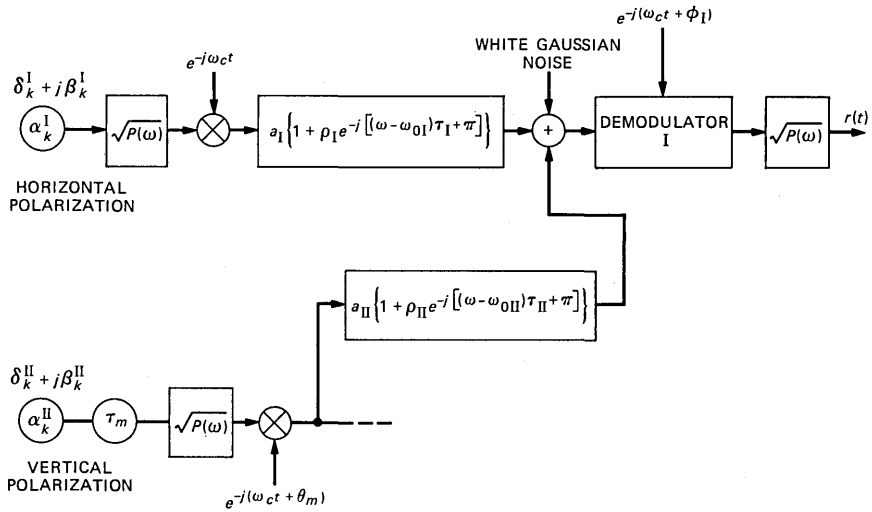


Fig. 1—Channel model for dual-polarized radio system.

symbols as

$$\alpha_k^i = \delta_k^i + j\beta_k^i \quad i = \text{I, II} \tag{1}$$

at consecutive instants kT_s, $k = 1, 2, 3 \cdots$. The index i denotes symbol sets transmitted on the main ($i = \text{I}$) and cross-coupled ($i = \text{II}$) channels. The real and imaginary parts $\{\delta_k^i, \beta_k^i\}$ take on discrete levels $\pm\Omega, \pm 3\Omega, \cdots, \pm(L-1)\Omega$ with equal probability. Parameter $L = \sqrt{M}$ is selected in accord with the number of states in the M-ary signal; the symbols δ and β are independent and identically distributed; and Ω is a measure of signal-point distance from the nearest decision boundary in signal space.

We assume each of the Pulse-Amplitude-Modulated (PAM) signals has a raised-cosine (Nyquist) spectral shaping with a roll-off factor Γ. The corresponding time-domain impulse shape is therefore[11]

$$p(t) = \frac{\sin(\pi t/T_s)}{\pi t/T_s} \cdot \frac{\cos(\Gamma \pi t/T_s)}{1 - (2\Gamma t/T_s)^2}, \tag{2}$$

and the frequency-domain transfer function is designated $P(\omega)$. On each of the orthogonally polarized channels, the shaped signals are modulated by quadrature carrier signals of frequency f_c. Furthermore, the two independent carrier local oscillators for $i = \text{I}$ and $i = \text{II}$ at the transmitter are out of phase by θ_m, where this phase difference is a uniformly distributed random variable over the range $0 \leq \theta_m \leq 2\pi$. The complex baseband symbol sequences are assumed misaligned by τ_m, where τ_m is also a uniformly distributed random variable over the range $0 \leq \tau_m \leq T_s$.

As previously stated, we follow Rummler's prescription for modeling multipath fading. The model is applied to both the copolarized ($i = \text{I}$) and the cross-coupled ($i = \text{II}$) signal paths and assumes the presence of a single inband notch in each of the main and cross-coupled path transfer functions. This latter assumption of notched fading in the cross-coupled signal band and notched fading in the main polarized signal band agrees with recent measurements that indicate the possibility of a shallow fade notch in the interfering cross-coupled path.[12]

For our analysis we use "static" fading models to emulate "snapshots" of multipath fading events.[13] The passband transfer function for the two-ray propagation model can be expressed as

$$F(\omega) = a\{1 + \rho e^{-j[(\omega-\omega_0)\tau+\pi]}\}, \tag{3}$$

where a represents the flat fade level and all other parameters are related to dispersive fading as follows: the fade notch depth is $-20 \log |1 - \rho|$, where ρ is the relative amplitude of the secondary ray;* the

* For minimum phase fades with $\rho < 1$.

relative delay between rays is τ and ω_0 is the fade notch frequency. For a fade notch centered in the passband,

$$\tau = \frac{2n\pi}{\omega_c - \omega_0}, \tag{4}$$

where ω_c is the radian carrier frequency.

In the absence of a cross-coupled interfering signal, the main polarization baseband waveform after demodulation at the receiver, as shown in Fig. 1, can be expressed as

$$x_{\mathrm{I}}(t) = \sum_k \alpha_k^{\mathrm{I}} h_{\mathrm{I}}(t - kT_s)e^{-j\phi_{\mathrm{I}}} + n_{\mathrm{I}}(t), \tag{5}$$

where

$$h_{\mathrm{I}}(t) = a_{\mathrm{I}}\{p(t) + \rho_{\mathrm{I}}p(t - \tau_{\mathrm{I}})e^{-j[(\omega_c - \omega_{0\mathrm{I}})\tau_{\mathrm{I}} + \pi]}\} \tag{6}$$

is the low-pass equivalent of the channel impulse response. The index I in eqs. (5) and (6) explicitly denotes reference to the main polarization channel; $n_{\mathrm{I}}(t)$ is filtered Gaussian noise with variance $\sigma_{n_{\mathrm{I}}}^2$; and ϕ_{I} represents the relative phase difference between the modulator and demodulator oscillators. In terms of a specific sampling epoch, t_0, distortion in the overall channel impulse response is minimized by an optimum relative carrier phase[3]

$$\phi_{\mathrm{I,opt}}(t_0) = -\operatorname{Arc\,tg}\frac{\rho_{\mathrm{I}}p(t_0 - \tau_{\mathrm{I}})\sin[(\omega_c - \omega_{0\mathrm{I}})\tau_{\mathrm{I}} + \pi]}{p(t_0) + \rho_{\mathrm{I}}p(t_0 - \tau_{\mathrm{I}})\cos[(\omega_c - \omega_{0\mathrm{I}})\tau_{\mathrm{I}} + \pi]}. \tag{7}$$

Extending the above discussion to dual-polarized operation, we give the received baseband signal in the main polarization channel as

$$r_{\mathrm{I}}(t) = \sum_k \alpha_k^{\mathrm{I}} h_{\mathrm{I}}(t - kT_s)e^{-j\phi_{\mathrm{I}}} + \sum_k \alpha_k^{\mathrm{II}} h_{\mathrm{II}}(t - kT_s - \tau_m)e^{-j(\phi_{\mathrm{I}} + \theta_m)} + n_{\mathrm{I}}(t), \tag{8}$$

where h_{I} and h_{II} represent the low-pass equivalents of the main and cross-coupled paths impulse responses, respectively. As long as the main signal [the first term in eq. (8)] is much stronger than the cross-coupled interferer, the carrier phase tracked by the demodulator of the main polarization receiver is the relative phase difference between local oscillators of the main polarization modulator and demodulator, $\phi_{\mathrm{I,opt}}(t_0)$. In its expanded form, eq. (8), representing the demodulated composite signal at the main polarization receiver, can be expressed as

$$r_{\mathrm{I}}(t) = \sum_k (\delta_k^{\mathrm{I}} + j\beta_k^{\mathrm{I}})$$
$$\times \{a_{\mathrm{I}} p(t - kT_s) + a_{\mathrm{I}}\rho_{\mathrm{I}} p(t - kT_s - \tau_{\mathrm{I}}) e^{-j[(\omega_c - \omega_{0\mathrm{I}})\tau_{\mathrm{I}} + \pi]} e^{-j\phi_{\mathrm{I}}}\}$$
$$+ \sum_k (\delta_k^{\mathrm{II}} + j\beta_k^{\mathrm{II}})$$
$$\times \{a_{\mathrm{II}} p(t - kT_s - \tau_m) + a_{\mathrm{II}}\rho_{\mathrm{II}} p(t - kT_s - \tau_{\mathrm{II}} - \tau_m)$$
$$\cdot e^{-j[(\omega_c - \omega_{0\mathrm{II}})\tau_{\mathrm{II}} + \pi]} e^{-j(\phi_{\mathrm{I}} + \theta_m)} + n_{\mathrm{I}}(t). \tag{9}$$

Notice that the parameters of the two independent fading events can be varied independently.

To establish the relationship of eq. (9) to probability of error, and ultimately derive the performance signature (M curve)[9] that describes outage performance for a 10^{-3} symbol error rate, we can focus on the in-phase or quadrature rail signal on the received main polarization. Because the signal constellations are symmetrical, the associated probability of error for each of those two rails is identical and the *total* symbol error rate can be determined. From eq. (9) the in-phase component of the received signal is clearly $\mathrm{Re}[r_{\mathrm{I}}(t)] \equiv r_{i,\mathrm{I}}(t)$:

$$r_{i,\mathrm{I}}(t) = a_{\mathrm{I}}\delta_0^{\mathrm{I}}\{\cos(\phi_{\mathrm{I}})p(t) + \rho_{\mathrm{I}}\cos[(\omega_c - \omega_{0\mathrm{I}})\tau_{\mathrm{I}} + \pi + \phi_{\mathrm{I}}]p(t - \tau_{\mathrm{I}})\}$$
$$+ a_{\mathrm{I}} \sum_{k\neq 0} \delta_k^{\mathrm{I}}\{\cos(\phi_{\mathrm{I}})p(t - kT_s)$$
$$+ \rho_{\mathrm{I}}\cos[(\omega_c - \omega_{0\mathrm{I}})\tau_{\mathrm{I}} + \pi + \phi_{\mathrm{I}}]p(t - kT_s - \tau_{\mathrm{I}})\}$$
$$+ a_{\mathrm{I}} \sum_k \beta_k^{\mathrm{I}}\{\sin(\phi_{\mathrm{I}})p(t - kT_s)$$
$$+ \rho_{\mathrm{I}}\sin[(\omega_c - \omega_{0\mathrm{I}})\tau_{\mathrm{I}} + \pi + \phi_{\mathrm{I}}]p(t - kT_s - \tau_{\mathrm{I}})\}$$
$$+ a_{\mathrm{II}} \sum_k \delta_k^{\mathrm{II}}\{\cos(\phi_{\mathrm{I}} + \theta_m)p(t - kT_s - \tau_m)$$
$$+ \rho_{\mathrm{II}}\cos[(\omega_c - \omega_{0\mathrm{II}})\tau_{\mathrm{II}} + \pi + \phi_{\mathrm{I}} + \theta_m]$$
$$\cdot p(t - kT_s - \tau_{\mathrm{II}} - \tau_m)\}$$
$$+ a_{\mathrm{II}} \sum_k \beta_k^{\mathrm{II}}\{\sin(\phi_{\mathrm{I}} + \theta_m)p(t - kT_s - \tau_m)$$
$$+ \rho_{\mathrm{II}}\sin[(\omega_c - \omega_{0\mathrm{II}})\tau_{\mathrm{II}} + \pi + \phi_{\mathrm{I}} + \theta_m]$$
$$\cdot p(t - kT_s - \tau_{\mathrm{II}} - \tau_m)\} + \mathrm{Re}[n_{\mathrm{I}}(t)], \tag{10}$$

and we denote the real part of the thermal noise by $n_{i,\mathrm{I}}(t)$. The preceding remarks are germane to modeling the channel. In the following subsection we relate the received in-phase baseband signal,

given by eq. (10), to the associated probability-of-error performance of the dual-polarized QAM radio system.

2.2 *Probability-of-error considerations*

Denoting the coefficient of the desired detected in-phase state as Z_0, we have

$$Z_0 = a_I\{\cos(\phi_I)p(t_0) + \rho_I\cos[(\omega_c - \omega_{0I})\tau_I + \pi + \phi_I]p(t_0 - \tau_I)\}, \quad (11)$$

where t_0, the sampling time, is optimized by mimizing the distortion contributed by the second and third terms in eq. (10). The set of slicing levels at the in-phase detector of the main polarization receiver can be set to

$$\{-(L-2)\Omega Z_0, \cdots, -2\Omega Z_0, 0, 2\Omega Z_0, \cdots, (L-2)\Omega Z_0\},$$

and, with no loss of generality, we set $\Omega = 1$.

Designating the sum of ISI contributed by dispersion in the main polarization and cross-coupled interference at the optimum sampling time t_0 by $\chi(\delta_k^I, \delta_k^{II}, \beta_k^I, \beta_k^{II})$, eq. (10) reduces to

$$r_{i,I}(t_0) = \delta_0^I Z_0 + \chi(\delta_k^I, \delta_k^{II}, \beta_k^I, \beta_k^{II}) + n_{i,I}(t_0). \quad (12)$$

Considering the automatic gain control operation at the receiver, the associated probability of error is then

$$P_{e,I_i} = 2\left[\frac{L-1}{L}\right]P_r\{(n_{i,I} + \chi) > Z_0\}. \quad (13)$$

Also, note that the noise variance $\sigma_{n_{i,I}}^2$ is equal to the total filtered noise variance, subsequently denoted by σ_n^2. Additionally,

$$P_r\{(\chi + n_{i,I}) > Z_0\} = E_{\chi,n_I}\{P_r[n_{i,I} > (Z_0 - \chi) \mid \chi = x]\}, \quad (14)$$

where $E\{\cdot\}$ denotes statistical averaging and x is the conditioned value of random variable χ. First, taking the average over the Gaussian noise, we obtain

$$P_r\{(\chi + n_{i,I}) > Z_0\} = \frac{1}{2}E_\chi\left\{\text{erfc}\left(\frac{Z_0 - x}{\sqrt{2}\,\sigma_n}\right)\right\}, \quad (15)$$

where the complementary error function, erfc(ϵ), is defined by

$$\text{erfc}(\epsilon) = \frac{2}{\sqrt{\pi}}\int_\epsilon^\infty e^{-\eta^2}d\eta\,. \quad (16)$$

By symmetry of the constellation, the total symbol error rate can therefore be expressed as

$$P_e \approx 2P_{e,I_i} = 2\,\frac{L-1}{L}\int_x \text{erfc}\left(\frac{Z_0 - x}{\sqrt{2}\,\sigma_n}\right)dF(x), \quad (17)$$

where $F(x)$ is the cumulative distribution function for the random variable χ.

The integration in eq. (17) can be evaluated using Gauss Quadrature Rules (GQR).[14] Equation (17) is thus expressed as

$$P_e \approx 2\frac{L-1}{L}\sum_{j=1}^{N} w_j \operatorname{erfc}\left[\frac{Z_0 - \xi_j}{\sqrt{2}\,\sigma_n}\right]. \tag{18}$$

In this equation, N is the number of terms in the finite series, and w_j and ξ_j are weights and nodes in the GQR approximation. At this point, P_e in eq. (18) is calculated from the $N_0 = 2N + 1$ moments of χ. Because χ is functionally dependent upon the independent transmitted symbol states, we have adopted Prabhu's algorithm[15] to determine the moments. Note that the moments obtained in this manner are conditioned on the values of the two random variables τ_m and θ_m. Hence, the resulting moments must be averaged over $\{\tau_m, \theta_m\}$ before they can be used in the GQR method.

To derive the probability of symbol error as a function of signal-to-noise ratio (s/n), we define

$$\gamma = \frac{L^2 - 1}{3}\frac{\Omega^2}{\sigma_n^2} \tag{19}$$

as the s/n. Using this relationship, we rewrite eq. (18) as

$$P_e = 2\frac{L-1}{L}\sum_{j=1}^{N} w_j \operatorname{erfc}\left[(Z_0 - \xi_j)\sqrt{3\gamma/2(L^2-1)}\right], \tag{20}$$

where we have normalized Ω to one. In the next section we expand further on the computational aspects of the GQR method.

As previously stated, the relationships above are all dependent upon the timing phase. This parameter can be selected a posteriori to minimize P_e, thus making probability-of-error computations a formidable, numerically intensive task. In this work we choose a priori sampling epoch by minimizing the peak distortion of the received signal prior to equalization. The timing phase is thus dependent on both the in-phase and quadrature rails of the main polarization M-QAM signal.

2.3 Signal-to-interference ratio

In addition to performance signature, we have also evaluated at select points along the M curve a corresponding signal-to-interference ratio (SIR). This ratio is defined as the relative power in the main polarization to that of the cross-coupled interfering signal. Using parameters previously defined, this ratio is simply

SIR

$$= \frac{a_{\mathrm{I}}^2 \int_0^{(1+\Gamma)\pi/T_s} |P(\omega)|^2\{1 + \rho_{\mathrm{I}}^2 - 2\rho_{\mathrm{I}}\cos[(\omega - \omega_{0\mathrm{I}})\tau_{\mathrm{I}}]\}d\omega}{a_{\mathrm{II}}^2 \int_0^{(1+\Gamma)\pi/T_s} |P(\omega)|^2\{1 + \rho_{\mathrm{II}}^2 - 2\rho_{\mathrm{II}}\cos[(\omega - \omega_{0\mathrm{II}})\tau_{\mathrm{II}}]\}d\omega}, \quad (21)$$

by which we can observe changes in the average SIR.

III. NUMERICAL ANALYSIS

An examination of Section II demonstrates that the theoretical analysis of M-QAM signal transmission over dual-polarized facilities in the presence of multipath fading is a computationally exhaustive activity. In this section we provide a brief overview of numerical issues related to our investigation.

3.1 Impulse response description

It is easily seen [see eq. (9) or (10)] that the dispersive nature of the multipath channel is completely described by the superposition of four impulse responses, each independently weighted by an appropriate transmitted symbol state. These impulse responses for the kth transmitted symbol are

$$u_{i,\mathrm{I}}(t) = a_{\mathrm{I}}\{p(t - kT_s)\cos(\phi_{\mathrm{I}}) + \rho_{\mathrm{I}}p(t - kT_s - \tau_{\mathrm{I}})\cos[(\omega_c - \omega_{0\mathrm{I}})\tau_{\mathrm{I}} + \pi + \phi_{\mathrm{I}}]\}, \quad (22a)$$

$$u_{q,\mathrm{I}}(t) = a_{\mathrm{I}}\{p(t - kT_s)\sin(\phi_{\mathrm{I}}) + \rho_{\mathrm{I}}p(t - kT_s - \tau_{\mathrm{I}})\sin[(\omega_c - \omega_{0\mathrm{I}})\tau_{\mathrm{I}} + \pi + \phi_{\mathrm{I}}]\}, \quad (22b)$$

$$u_{i,\mathrm{II}}(t) = a_{\mathrm{II}}\{p(t - kT_s - \tau_m)\cos(\phi_{\mathrm{I}} + \theta_m) + \rho_{\mathrm{II}}p(t - kT_s - \tau_{\mathrm{II}} - \tau_m)\cos[(\omega_c - \omega_{0\mathrm{II}})\tau_{\mathrm{II}} + \pi + \phi_{\mathrm{I}} + \theta_m]\}, \quad (22c)$$

and

$$u_{q,\mathrm{II}}(t) = a_{\mathrm{II}}\{p(t - kT_s - \tau_m)\sin(\phi_{\mathrm{I}} + \theta_m) + \rho_{\mathrm{II}}p(t - kT_s - \tau_{\mathrm{II}} - \tau_m)\sin[(\omega_c - \omega_{0\mathrm{II}})\tau_{\mathrm{II}} + \pi + \phi_{\mathrm{I}} + \theta_m]\}, \quad (22d)$$

where the variables have been previously defined. In the in-phase rail of the main polarization receiver, eqs. (22a) and (22b) describe the distorted in-phase and quadrature-coupled signals from the main polarization transmitter, respectively, and eqs. (22c) and (22d) describe the corresponding signals from the cross-coupled interferer signal. The optimum timing phase, $t_{0,\mathrm{opt}}$, is selected by minimizing the peak distortion embedded in eqs. (22a) and (22b). That peak distortion is

$$D_p(t_0) = \frac{1}{u_{i,\mathrm{I}}(t_0,\ k=0)} \left\{ \sum_{\substack{k=-\infty \\ k\neq 0}}^{\infty} |u_{i,\mathrm{I}}(t_0) + \sum_{k=-\infty}^{\infty} |u_{q,\mathrm{I}}(t_0)| \right\}. \quad (23)$$

Distortion contributed by $u_{i,\mathrm{II}}$ and $u_{q,\mathrm{II}}$ is specifically excluded from eq. (23) since the complex data sources I and II are not coherent; these latter signals appear noiselike to the timing recovery circuit in the main polarization receiver.

From a computational standpoint, the operations described above are carried out by first computing $\phi_{\mathrm{I,opt}}(t_0)$ and $D_p(t_0)$ for all t_0 in the range $[-T_s, T_s]$, and then selecting the single sampling epoch, $t_{0,\mathrm{opt}}$, that minimizes D_p. With $t_{0,\mathrm{opt}}$ identified, the symbol-period spaced samples of $u_{i,\mathrm{I}}$, $u_{q,\mathrm{I}}$, $u_{i,\mathrm{II}}$, and $u_{q,\mathrm{II}}$ are used for the subsequent probability-of-error computations. An a priori selection of $t_{0,\mathrm{opt}}$ obviously expedites the computer time necessary to perform the probability-of-error computations. The use of a timing phase that minimizes peak distortion is one such choice. Another choice could be minimization of mean-square eye closure. This can be shown to be equivalent to an IF timing recovery.

For *illustrative* purposes, we present plots of $u_{i,\mathrm{I}}$, $u_{q,\mathrm{I}}$, $u_{i,\mathrm{II}}$, and $u_{q,\mathrm{II}}$ in Fig. 2 for $\rho_{\mathrm{I}} = 0.80$, $\rho_{\mathrm{II}} = 0.75$, $a_{\mathrm{I}} = 1.0$, $a_{\mathrm{II}} = 0.5$, $f_{0\mathrm{I}} = 1.63$ MHz, $f_{0\mathrm{II}} = 0$ MHz, $\Gamma = 0.45$, $\tau_m = 0$, $\theta_m = 0$, and $T_s = 1/(15$ Mbaud). For this illustration example, $t_{0,\mathrm{opt}} = -0.4T_s$ and $\phi_{\mathrm{I,opt}} = 5.13°$. In particular, note that this carrier phase nulls the $u_{q,\mathrm{I}}$ response at $t = t_{0,\mathrm{opt}}$, as it should. However, even though the cross-coupled interferer corresponds

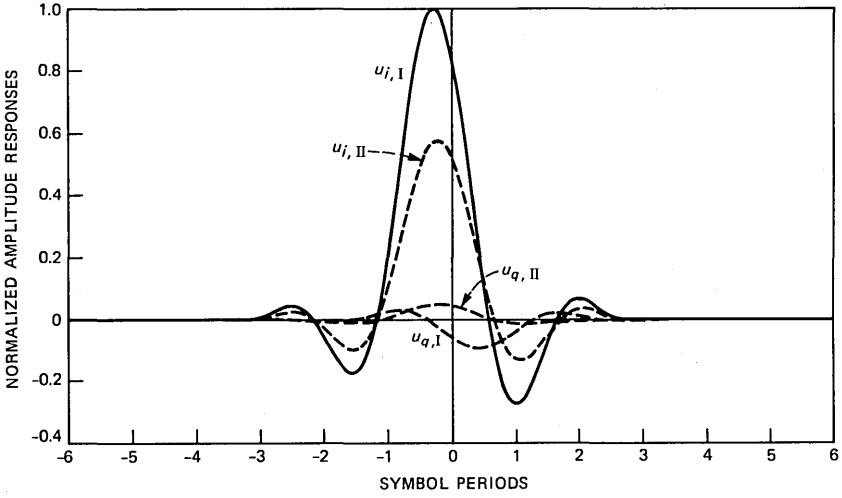


Fig. 2—Dispersive effects on impulse responses of a dual-polarized digital radio system.

to a notch-centered fade and $\theta_m = 0$, $u_{q,\mathrm{II}} \neq 0$ since the carrier phase optimization is independent of channel II.

3.2 Signal-to-interference evaluation

The signal-power-to-interference-power ratio was evaluated from eq. (21). The presence of $|P(\omega)|^2$ in this expression derives from the fact that at the receiver, the main and interference signals have $P(\omega)$ spectral shaping and thus a $|P(\omega)|^2$ spectral power density. The $a_i^2\{1 + \rho_i^2 - 2\rho_i \cos[(\omega - \omega_{0i})\tau_i]\}$, $i = \mathrm{I}, \mathrm{II}$, terms correspond to spectral power density associated with the dispersive fade and the undistorted signal power. The integral is conveniently evaluated using Romberg-quadrature methods.

3.3 Probability-of-error computations

As explained in Section 3.1, the in-phase component of the received signal on the main polarization has basically four parts. The desired symbol to be detected is the $k = 0$ term of $u_{i,\mathrm{I}}(t)$ in eq. (22a). Each of the four aforementioned terms consists of a sum of weighted, independent, identically distributed random variables that together were designated as $\chi(\delta_k^{\mathrm{I}}, \beta_k^{\mathrm{I}}, \delta_k^{\mathrm{II}}, \beta_k^{\mathrm{II}})$ and discussed in Section 3.2. Since the random variable χ is in the form of a sum of sums of independent random variables, it can be considered a long sum of weighted, independent random variables. Thus, to determine a finite number of moments of χ, Prabhu's[15] recursive algorithm can be used: If

$$\Delta_k = \sum_{i=1}^{k} \nu_i, \tag{24}$$

then

$$\begin{aligned} \psi_{N_0}(k) &= E\{[\Delta_{k-1} + \nu_k]^{N_0}\} \\ &= \sum_{t=0}^{N_0} \binom{N_0}{t} \psi_t(k-1) E\{\nu_k^{(N_0 - t)}\}. \end{aligned} \tag{25}$$

Based on this recursive formula, a computer program was written to calculate N_0 moments of the random variable χ.

Since all zero-mean random variables involved in the sum are evenly distributed, the odd moments of the sum become zero. In all our computations, $N_0 = 31$ moments were found to be adequate for the probability-of-error calculations.

Following the computation of conditional moments of χ by averaging over uniformly distributed variables τ_m and θ_m, the absolute moments of χ were found and subsequently used with the GQR method for computing the error probabilities. (For a brief summary of GQRs see Appendix C of Ref. 17.) For additional detail, interested readers are

referred to Ref. 14. Observe that by averaging over $\alpha_k^i (i = \text{I, II})$, τ_m, and θ_m when computing the N_0 moments ψ_{N_0}, a great deal of computation time is saved because these calculations are made *once* for all s/n values.

IV. NUMERICAL AND EXPERIMENTAL RESULTS

In this section we present theoretical and measured performance signature curves for dual-polarized digital radio. It is well known that signature curves provide a locus of fade notch depths (in decibels) and relative fade notch positions (Hz) for a 10^{-3} probability of error. However, unlike their more customary presentation, we must also include parameters that define the character of the interfering cross-coupled multipath channel.

4.1 Theoretical performance signatures

As a reference, we have computed the signature of a singularly polarized 16-QAM system, that is, $a_{\text{II}} = 0$. The data are presented in Fig. 3 and labeled "1." Along this contour we also designate the SIR (in decibels) computed using methods previously described. For the case of no cross-coupled interferer, the SIR is obviously infinite. For all other cases, the SIR value at each point on the signature curve is functionally dependent upon specific fading parameters for the main and cross-coupled signals. Moreover, we associate with each curve a triplet representing the dispersive fading status of the cross-coupled interferer. This triplet is

$$\left[20 \log \frac{a_{\text{II}}}{a_{\text{I}}} \text{ (in dB)}, -20 \log|1 - \rho_{\text{II}}| \text{(in dB)}, \text{ and } \Delta f_{0\text{II}} \text{ (in MHz)}\right],$$

where $\Delta f_{0i}, i = \text{I, II}$, denotes fade notch positions relative to the carrier. For example, the triplet (−30, 5, 11) describes a cross-coupled signal with a flat power level 30 dB below that of the main polarization with a 5-dB inband fade notch located 11 MHz away from the carrier frequency. For the case of curve 1 in Fig. 3, the triplet is simply $(-\infty, \cdot, \cdot)$.

In addition to curve 1 in Fig. 3, we show three other cases corresponding to (−30, 10, 0), (−30, 5, 0), and (−30, 5, 11) and denoted 2, 3, and 4, respectively. A comparison of curves 2 and 3, with the same −30 dB flat power levels and 0-MHz notch offsets, reveals that the fade of curve 2, with a 10-dB inband notch, results in *less* outage time than the fade of curve 3, with 5-dB inband notch. Hence, the greater power loss associated with curve 2 leads to reduced outage, even though the intersymbol interference for curve 2 exceeds that of curve 3. Now

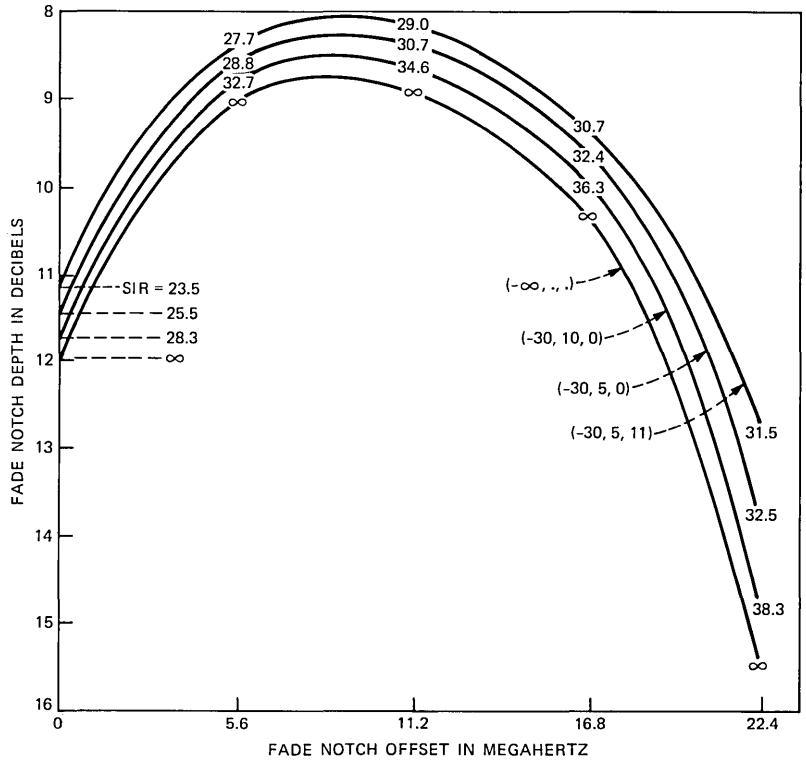


Fig. 3—Performance signature curves for dual-polarized 16-QAM radio; $\Gamma = 0.45$, $T_s = 1/(22.5$ Mbaud), $s/n = 60$ dB. Curves 1 through 4 show relationship of SIR to relative digital radio outage.

consider curves 3 and 4. This data corresponds to identical flat power levels and fade notch depths, with the notch position moving from 0 MHz (notch centered) to 11 MHz (near the band edge). The notch-centered fade apparently causes less outage than the notch offset fade. When we remember that the unfaded signal spectral energy at 0 MHz is much more than that near 11 MHz, it should be clear that the relationship of curves 3 and 4 is again that of diminished net signal power in the interferer that equates to a reduced outage.

SIR values are listed at certain points along curves 2, 3, and 4. It will be noted that moving toward higher notch offset values on each performance signature results in greater SIRs, specifically for the reason cited above. A fade positioned in a region of normally high spectral energy will pull out more power than the same fade in a region of reduced spectral energy. Moreover, at any main polarization notch frequency position, the SIR increases as the performance signature drops, because interference signal power, rather than intersymbol

interference, is the dominant factor for outage due to an independent interferer.

To further verify that interference power plays a major role in outage performance, Fig. 4 presents two relevant cases. In that figure we consider (−30, 0, 0) and (−25, 5, 0) dispersive fading of 16-QAM cross-coupled signals. In both cases the total interference power loss is approximately the same. The former case corresponds to a 30-dB flat fade of the interferer, while in the latter case the flat fade is 25 dB with a 5-dB shaped fade positioned in the center of the passband (a region of high spectral energy). Observe that the outage performance and SIR are virtually identical along the entire signature curves. We therefore conclude that whether fading of the cross-coupled signal is mildly dispersive (i.e., generating intersymbol interference) or flat, outage performance improvement is accompanied by interference power reduction provided the flat level of the cross-coupled signal is considerably below that of the main polarization signal (this is normally the case because of antenna/waveguide polarization isolation) and the dispersive fading of the cross-coupled signal is shallow.

To confirm that the aforementioned property holds at lower isolation levels as well, we have repeated the performance signature cal-

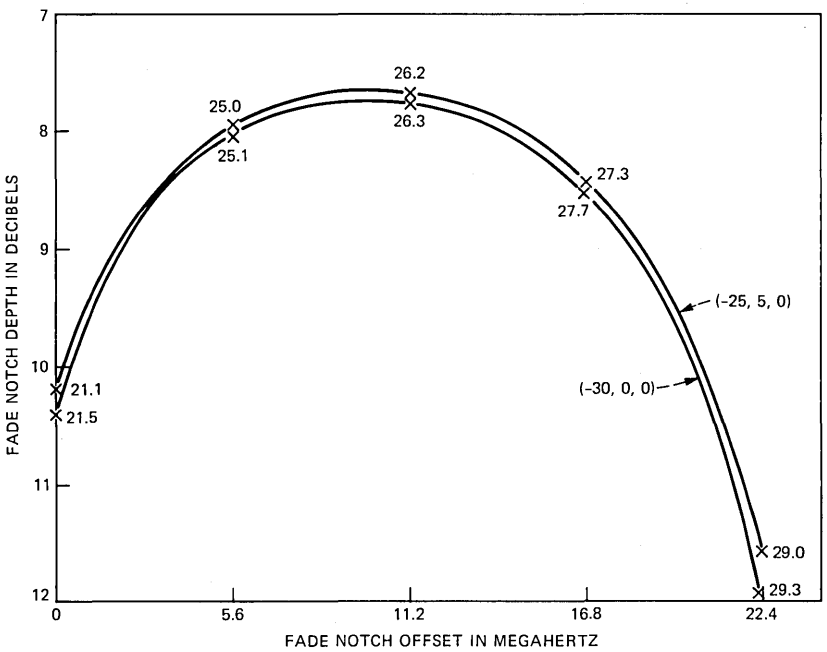


Fig. 4—Performance signature curves for dual-polarized 16-QAM radio; $\Gamma = 0.45$, $T_s = 1/(22.5 \text{ Mbaud})$, $s/n = 60$ dB. The total interference power loss is approximately the same, as is the relative digital radio outage.

culations for an interfering signal flat fade level of −20 dB. The results, presented in Fig. 5, confirm those in Figs. 3 and 4.

Up to now θ_m and τ_m were taken to be random variables. To gain further insight, we now impose $\theta_m = 0$ and $\tau_m = 0$ conditions on the interfering signal, thereby illustrating a case for which timing and carrier phase of the two signals are aligned. We have repeated computations for the case presented in Fig. 5. These computations are presented in Fig. 6. Observe that the same qualitative properties hold. In the case of a (−20, 0, 0) interfering signal, the synchronous system exhibits a lower outage time than the general system for the same interferer (see Fig. 5).

Data in Figs. 3 through 6 were all computed for a 60-dB s/n, 22.5-Mbaud symbol rate, $\Gamma = 0.45$ roll-off, 16-QAM dually polarized radio system. We have repeated these performance signature computations for a 66-dB s/n, 15-Mbaud, $\Gamma = 0.45$, 64-QAM dually polarized radio system. The resulting curves, shown in Fig. 7, are expectedly much wider than the 16-QAM case. However, the general phenomenon exhibited by the data of Figs. 3 through 6 remains appropriate to Fig. 7, as well.

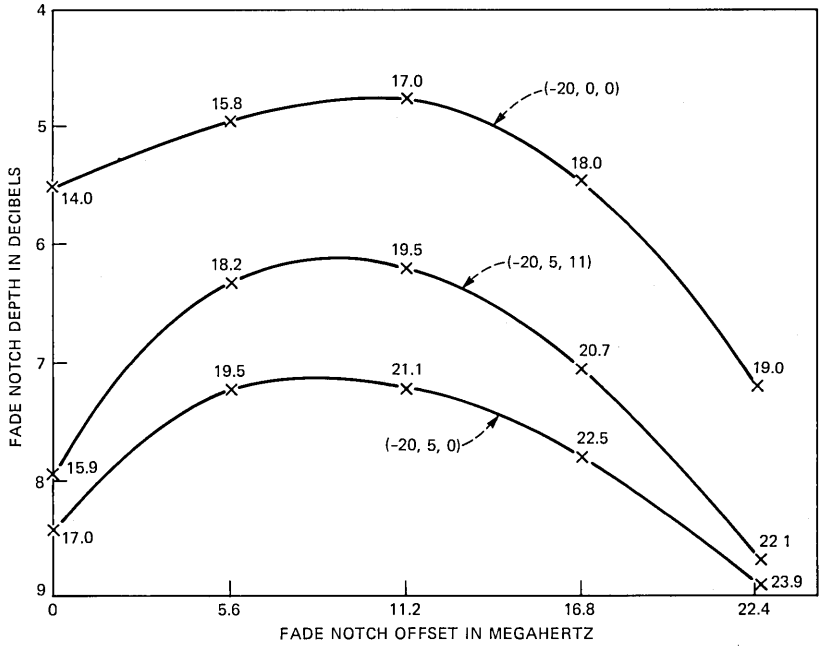


Fig. 5—Performance signature curves for dual-polarized 16-QAM radio; $\Gamma = 0.45$, $T_s = 1/(22.5$ Mbaud), $s/n = 60$ dB. These cases correspond to lower isolation levels than those in Figs. 3 and 4.

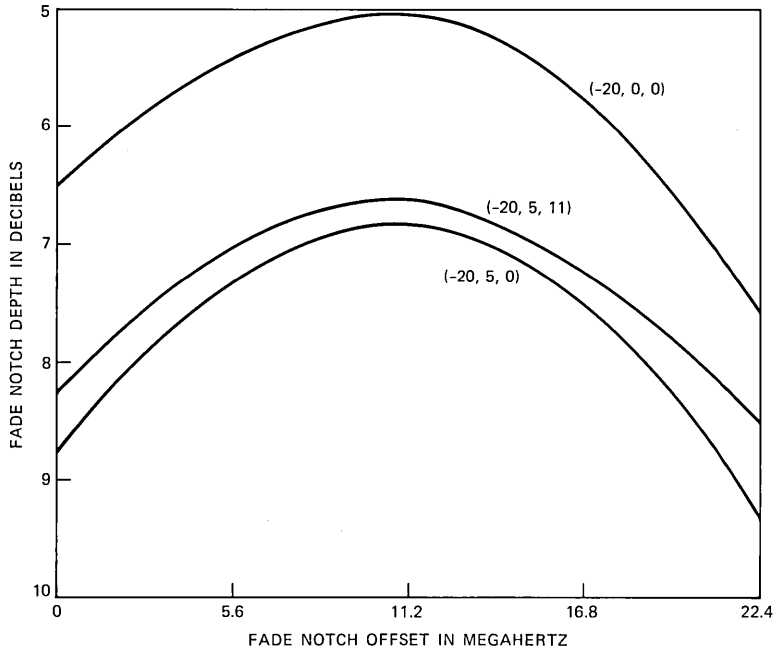


Fig. 6—Performance signature curves for dual-polarized 16-QAM radio; $\Gamma = 0.45$, $T_s = 1/(22.5 \text{ Mbaud})$, $s/n = 60$ dB, $\theta_m = 0$, and $\tau_m = 0$.

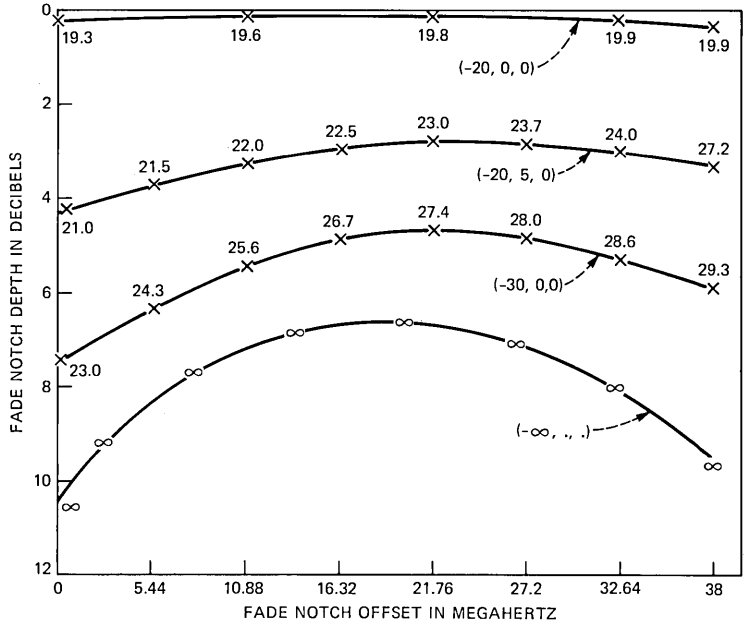


Fig. 7—Performance signature curves for dual-polarized 64-QAM radio; $\Gamma = 0.45$, $T_s = 1/(15 \text{ Mbaud})$, $s/n = 66$ dB.

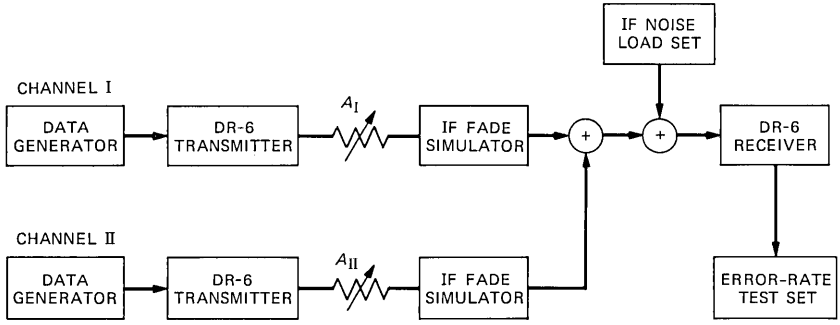


Fig. 8—Experimental facility for the measurement of dual-polarized digital radio performance signatures.

V. EXPERIMENT RESULTS

Using the experimental arrangements functionally illustrated in Fig. 8, we have confirmed the theoretical conclusions discussed in the previous section. As noted in the figure, two independent data sources provide inputs to DR-6* transmitters for channels I and II. The flat power levels of the two outputs are separately adjusted using attenuators A_I and A_{II}, and dispersive fading is provided by IF fade simulators. The channel I and II signals are added together with low-level noise (60-dB s/n) from an IF noise load set. The composite signal, simulating independent fading of a main polarization and a cross-coupled interferer, is then applied to a DR-6 receiver and error-rate test set.

Attenuators A_I and A_{II} were used to adjust the ratio a_{II}/a_I. Power was measured at the output of the IF fade simulators to establish the SIR and again at the input to the radio receiver to assure the demodulator had the appropriate signal level.

Error-rate measurements were made for a number of different fading events. The first set of measurements correspond to the triplets $(-\infty, 0, 0)$ and $(-25, 0, 0)$, the latter triplet representing a *flat*, nondispersive fade of the interferer. The performance signatures are presented in Fig. 9 and show the same trend as the theoretical data of Figs. 3 and 7. The quantitative differences are to be expected owing to equipment imperfections, such as nonideal Nyquist and bandpass filters, and imperfect carrier and timing recovery circuit operations. We next examined the influence of a dispersive fade characterized by $(-25, 5, 0)$. As expected, the performance signature improved, but not to the point of reaching the $(-\infty, 0, 0)$ case. This observation agrees with the data of Figs. 3, 5, and 7.

* DR-6 is a 16-QAM, 90-Mb/s, 22.5-Mbaud digital radio system.[16]

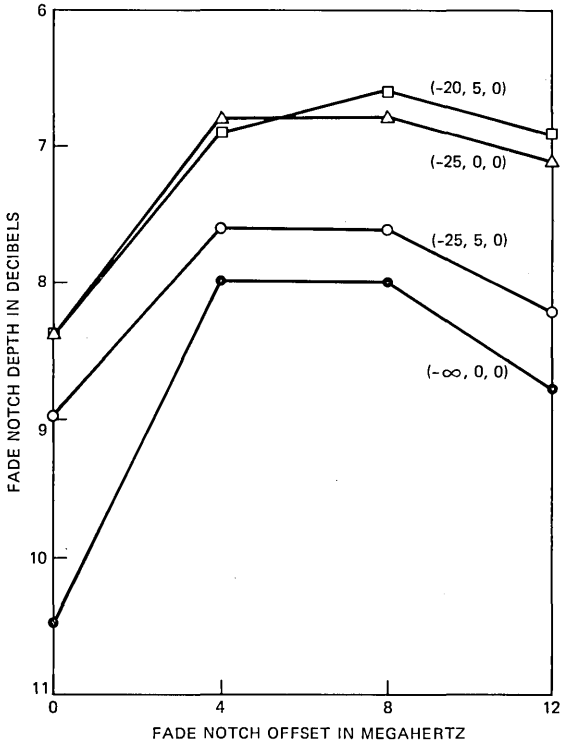


Fig. 9—Measured performance signatures for dual-polarized 16-QAM radio; $\Gamma = 0.45$, $T_s = 1/(22.5$ Mbaud), $s/n = 60$ dB.

The next phase of the experimental study was to increase the flat fade level from −25 to −20 dB and to introduce a notch-centered 5-dB fade, thus the triplet (−20, 5, 0). The resulting performance signature curve is also shown in Fig. 9 and supports the theoretical data of Fig. 4. In general, conclusions drawn from the experimental data support our theoretical findings. Note that in all of the experimental tests, the two transmitters were completely independent of one another; hence, the conditions $0 \leq \tau_m \leq T_s$ and $0 \leq \theta_m \leq 2\pi$ actually held.

VI. CONCLUSIONS

In this paper we present computed performance signatures for dual-polarized transmission of M-QAM signals over dispersive multipath digital radio channels. For the assumed model, we show that a cross-coupled interferer exhibits noiselike behavior, and its power loss, whether flat or midly dispersive, brings about an improvement (reduction) in dual-polarized system outage time. The theoretical findings

are supported by measured performance signatures obtained from a laboratory simulation of the analytic model.

REFERENCES

1. A. J. Giger and W. T. Barnett, "Effects of Multipath Propagation on Digital Radio," IEEE Trans. Commun., *COM-29*, No. 2 (December 1979), pp. 1870–5.
2. G. J. Foschini and J. Salz, "Digital Communications Over Fading Radio Channels," B.S.T.J., *62*, No. 2, Part I (February 1983), pp. 429–59.
3. H. Sari, "A Comparison of Equalization Techniques on 16-QAM Digital Radio Systems During Selective Fading," Globecom '82 Conf. Rec. (November–December 1982), pp. F3.5.1–6.
4. N. Amitay and L. J. Greenstein, "Multipath Outage Performance of Digital Radio Receivers Using Finite-Tap Adaptive Equalizers," IEEE Trans. Commun., *COM-32*, No. 5 (May 1984), pp. 597–608.
5. G. L. Fenderson et al., "Adaptive Equalization of Multipath Propagation for 16-QAM 90-Mb/s Digital Radio," AT&T Bell Lab. Tech. J., *63*, No. 8 (October 1984), pp. 1447–63.
6. C. A. Siller, Jr., "Multipath Propagation: Its Associated Countermeasures in Digital Microwave Radio," IEEE Commun. Mag., *22*, No. 2 (February 1984), pp. 6–15.
7. M. Kavehrad, "Performance of Cross-Polarized M-ary QAM Signals Over Nondispersive Fading Channels," AT&T Bell Lab. Tech. J., *63*, No. 3 (March 1984), pp. 499–521.
8. W. D. Rummler, "A New Selective Fading Model: Application to Propagation Data," B.S.T.J., *58*, No. 5 (May–June 1979), pp. 1037–71.
9. M. Emshwiller, "Characterization of the Performance of PSK Digital Radio Transmission in the Presence of Multipath Fading," ICC '78 (June 1978), pp. 47.3.1–6.
10. C. W. Lundgren and W. D. Rummler, "Digital Radio Outage Due to Selective Fading—Observation Versus Prediction From Laboratory Simulation," B.S.T.J., *5* (May–June 1979), pp. 1073–100.
11. R. W. Lucky, J. Salz, and E. J. Weldon, Jr., *Principles of Data Communications*, New York: McGraw-Hill, 1968.
12. P. E. Butzien, private communication.
13. M. Schwartz et al., *Communication Systems and Techniques*, New York: McGraw-Hill, 1966.
14. G. H. Golub, and J. H. Welsch, "Calculations of Gauss Quadrature Rules," Math. Comp., *26*, No. 106 (April 1969), pp. 221–30.
15. V. K. Prabhu, "Some Considerations of Error Bounds in Digital Systems," B.S.T.J., *50* (December 1971), pp. 3127–52.
16. J. J. Kenny, "Digital Radio for 90-Mb/s, 16-QAM Transmission at 6 and 11 GHz," Microw. J., *25*, No. 8 (August 1982), pp. 71–80.
17. M. Kavehrad, "Performance of Nondiversity Receivers for Spread Spectrum in Indoor Wireless Communications," AT&T Tech. J., *64*, No. 6 (July–August 1985), pp. 1181–210.

AUTHORS

Mohsen Kavehrad, B.S. (Electrical Engineering), 1973, Tehran Polytechnic Institute; M.S. (Electrical Engineering), 1975, Worcester Polytechnic Institute; Ph.D. (Electrical Engineering), 1977, Polytechnic Institute of New York; Fairchild Industries, 1977–1978; GTE, 1978–1981; AT&T Bell Laboratories, 1981—. At AT&T Bell Laboratories Mr. Kavehrad is a member of the Communications Methods Research Department at Crawford Hill Laboratory. His research interests are digital communications and computer networks. He has organized and chaired sessions for IEEE sponsored conferences. Technical Editor, IEEE Communications Magazine; Chairman, IEEE Communications Chapter of New Hampshire, 1984; Member, IEEE, Sigma Xi.

Curtis A. Siller, Jr., B.S.E.E., 1966, M.S. (Plasma Physics), 1967, Ph.D. (Electrical Engineering), 1969, The University of Tennessee at Knoxville;

AT&T Bell Laboratories, 1969–1978, 1979—. At AT&T Bell Laboratories, Mr. Siller has analyzed and designed reflector antennas for terrestrial microwave communications; performed exploratory investigations of digitally implemented adaptive transversal equalizers for high-speed digital radio systems; identified novel approaches to digital FIR filtering and quadrature amplitude modulation; and explored new algorithms for the stable control of fractionally spaced equalizers. Mr. Siller has authored nearly 30 papers in the aforementioned areas and is presently involved in system engineering of future digital transmission systems. He is the recipient of an AT&T Bell Laboratories Distinguished Technical Staff Award. Senior Member, IEEE, where he serves on the Signal Processing and Communication Electronics Technical Committee, and has helped plan the technical program for several international conferences; Member, Phi Eta Sigma, Eta Kappa Nu, Tau Beta Pi, Phi Kappa Phi, Sigma Xi, New York Academy of Sciences.

AT&T Technical Journal
Vol. 64, No. 10, December 1985
Printed in U.S.A.

An Algebraic Approach to a Nonproduct Form Network

By S. W. YOO*

(Manuscript received November 30, 1984)

By applying an algebraic approach to the method of stages, an explicit solution is derived for a closed network consisting of a nonexponential server and a service station with two identical nonexponential servers in parallel. There is a finite number of jobs and the queueing discipline is first-come-first-served in the closed network. The solution is described in a quasi-matrix-geometric form, which is a generalization of the matrix-geometric form.

I. INTRODUCTION

There have been many developments and researches in the queueing networks for modeling of computer systems in the last two decades. Most researchers believe, however, that it is very difficult to conceptualize how to derive steady state solutions for general nonproduct form queueing networks, which do not have a product form solution.[1] For instance, a steady state solution for a general queueing network with first-come-first-served (FCFS) queueing discipline and general service times is unavailable at present. Several recent empirical papers have shown that service time distribution can have a significant effect on performance with particular emphasis on the modeling of computer systems.[2,3] Recently several works describe steady state solutions for the restricted cases of the nonproduct form queueing networks.[4–8] Neuts investigated a single server queue with phase type distribution and also found the solution to the M/G/1 queue.[6] Carroll et al.

* AT&T Information Systems.

approached M/G/1 and GI/M/1 algebraically and found an explicit solution.[5] Van de Liefvoort extended Carroll's work to G/G/1//N type loops.[7] The solutions of these works are described in Neut's book on matrix geometric forms.

This paper attempts to generalize these studies. It focuses on a closed network consisting of a nonexponential server and a service station with two identical nonexponential servers in parallel, which is a typical model of a computer system; a central processing unit and two input/output processors. There are a finite number of jobs, and the queueing discipline is FCFS at each node. An explicit steady state solution for this network is derived in a *quasi matrix geometric* form, where matrices are recursively defined in certain product spaces. Obviously this form is a generalization of the matrix geometric form. This result may provide insight for obtaining exact solutions for general closed networks that do not have a product-form solution.

Section II of this paper introduces notations and definitions for describing the algebraic description of the general distribution server. In Section III, both external states and internal states are introduced to represent the states of the network. In Section IV, the global balance equations of the network is derived in a matrix form. Section V gives the steady state solution of the global balance equation. The conclusions are summarized in Section VI.

II. DESCRIPTION OF THE NETWORK

One of the common approaches for representing general service time distributions is to use the method of stages. That is, each nonexponential service time distribution is replaced by a subnetwork of exponential stages with the constraint that the subnetwork can only be accessed by one job at a time. The principle on which the method of stages is based is the memoryless property of exponential distribution. Thus this method is both general and compatible with the definition of Markov processes. All works to be studied in this paper are based on Cox's statement that any server whose service time distribution function has a rational Laplace transform could be replaced exactly by a subnetwork of exponential distributions.[9]

In this paper, we consider a closed network consisting of two service stations, a nonexponential server (station 0) and a service station (station 1) containing two identical nonexponential servers (see Fig. 1).

Let N denote the total number of jobs (there can be only one job active at any time within each of the servers). When a job completes service at either server of station 1, it leaves station 1 and joins the queue of station 0, while another job (if any) in the queue of station 1 takes its place.

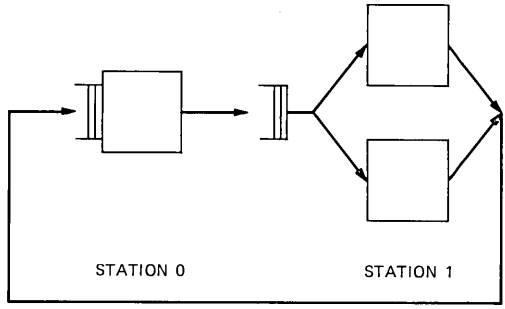


Fig. 1—A closed network with nonexponential servers.

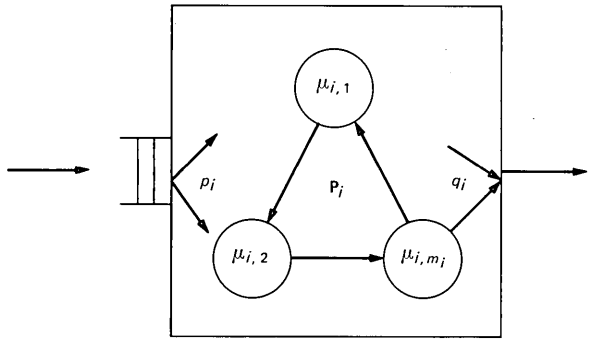


Fig. 2—Subnetwork of exponential stages.

Each of the nonexponential servers in Fig. 1 is represented by a subnetwork of exponential stages, as shown in Fig. 2. There are m_i exponential stages, whose service rates are $\mu_{i_1}, \ldots, \mu_{i,m_i}$, in the server of station i ($i = 0, 1$). The server of station i can be characterized by vectors and matrices. These vectors are denoted by lowercase letters, whereas matrices are denoted by uppercase and bold letters. This convention is followed without exception. The notation that will be used is consistent with that of Refs. 5, 7, and 8.

Definition 1: For each server of station i, define the following:

p_i = the entrance probability vector,
q_i = the departure probability vector,
$\mathbf{P}_i$ = the substochastic transition matrix, and
$\mathbf{M}_i$ = the service rate matrix = diag $(\mu_{i,1}, \ldots, \mu_{i,m_i})$.

For instance, $(p_i)_k$, the kth component of the vector p_i, is the conditional probability that a job, upon entering the server of station i, will first go to stage k. Similarly, $(q_i)_k$ is the conditional probability that a job, upon completing service at the stage k in the server of station i, will leave the server, and $(\mathbf{P}_i)_{kj}$ is the transition probability

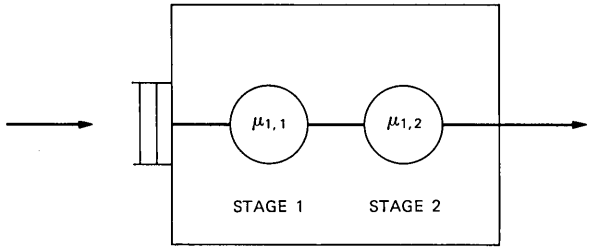


Fig. 3—Erlangian-2.

from stage k to stage j within the server of station i. One simple example is provided to show how to characterize the nonexponential server.

Example 1: If a server of station 1 is Erlangian-2 as shown in Fig. 3, then the server is characterized by the following vectors and matrices:

$$p_1 = (1 \quad 0), \quad q_1 = (0 \quad 1), \quad \mathbf{P}_1 = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad \text{and} \quad \mathbf{M}_1 = \begin{bmatrix} \mu_{1,1} & 0 \\ 0 & \mu_{1,2} \end{bmatrix}.$$

Lemma 1: *For $i = 0, 1$, let $\mathbf{I}_i$ be an identity matrix of order m_i and e_i a vector containing m_i ones. The following relationships hold:*

$$p_i e_i^T = 1 \tag{1}$$

and

$$q_i^T = (\mathbf{I}_i - \mathbf{P}_i) e_i^T, \tag{2}$$

where T denotes the transpose.

Proof: Using the law of total probability, we obtain the following relationships:

$$\sum_{j=0}^{m_i} (p_i)_j = 1, \qquad \text{for} \quad i = 0, 1,$$

$$\sum_{k=1}^{m_i} (\mathbf{P}_i)_{jk} = 1 - (q_i)_j, \qquad \text{for} \quad i = 0, 1 \quad \text{and} \quad 1 \le j \le m_i.$$

Equations (1) and (2) are obtained from the above relationships. This proves the lemma.

Next, three important matrices in the paper are introduced to simplify the balance equations and express the properties of the server.

Definition 2: For $i = 0, 1$, define the following:

$$\mathbf{B}_i = \mathbf{M}_i(\mathbf{I}_i - \mathbf{P}_i), \tag{3}$$

$$\mathbf{V}_i = \mathbf{B}_i^{-1}, \quad \text{and} \tag{4}$$

$$\mathbf{Q}_i = e_i^T p_i, \tag{5}$$

where -1 denotes the matrix inverse.

Note that $\mathbf{Q}_i$ is an idempotent matrix of rank 1. We can assume that $\mathbf{I}_i - \mathbf{P}_i$, i = 0, 1, has an inverse since this expression means that a path exists from each stage out of the server of station i. Carroll et al.[5] and Neuts[4] have shown that traditional probability notations can be expressed by vectors and matrices.

Lemma 2: For $i = 0, 1$, let $b_i(x)$ be the probability density function (pdf) of the server of station i, $E_i(x^n)$ its nth moment, $E_i(x)$ its mean service time, and $B_i^*(s)$ its Laplace transform.
Then

$$B_i^*(s) = p_i(\mathbf{I}_i + s\mathbf{V}_i)^{-1}e_i^T, \tag{6}$$

$$E_i(x^n) = \int_0^\infty b_i(x)x^n dx = n!p_i\mathbf{V}_i^n e_i^T, \tag{7}$$

$$b_i(x) = p_i[\mathbf{B}_i \exp(-\mathbf{B}_i x)]e_i^T, \quad \text{and} \tag{8}$$

$$E_i(x) = p_i\mathbf{V}_i e_i^T. \tag{9}$$

Proof: Omitted.

III. THE STATES OF THE NETWORK

If there is only one job present ($N = 1$) in the system, queueing will never take place. The system is easily solved. If there is more than one job ($N > 1$), the state of the network is characterized by a number of jobs (called *external state*) and positions of active jobs (called *internal state*) at each station. The queueing problem is thus transformed from one involving the remaining service time for a job to one involving the position of the active job in the subnetwork.

In defining the states of the network, it is important to keep the internal state separate from the external state in order to make the balance equation easier to solve. Each external state has a set of internal states which we call the *internal state space*. Thus the state of the network can be represented by a pair, external state and internal state.

The set of possible external states, which we call the *external state space*, is determined as follows:

$$\{(N, 0), (N-1, 1), \cdots, (N-n, n), \cdots, (0, N-1)\},$$

where $1 \le n \le N$.

The cardinality of the external state space is $N + 1$. Next consider the internal state space. Since the number of internal states at station 0

is the same as the number of stages in the server, the internal state space of station 0 is defined as follows:

$$S_0 = \{1, 2, \cdots, m_0\}.$$

If there is only one job at station 1, it must be active at one of the servers. Thus the internal state space for one job at station 1 is defined as follows:

$$S_1 = \{1, 2, \cdots, m_1\}.$$

When there are two or more jobs at station 1, two of them must be active. In this case, a possible internal state would be a pair of integers (j, k), where one job is at stage j and the other is at stage k. Since the two nonexponential servers in station 1 are identical, (k, j) is the same as (j, k). Thus the internal state space for two or more jobs at station 1 is defined as follows:

$$S_2 = \{(i_1, i_2) \mid 1 \leq i_1 \leq i_2 \leq m_1\}.$$

Let $d(i)$ be the dimension (cardinality) of the state space S_i. There are $d(0) = m_0$ states in S_0, $d(1) = m_1$ states in S_1 and $d(2) = m_1(m_1 + 1)/2$ such states in S_2.

Definition 3: For $N > 2$, the *state of the network* is represented by a vector $[x_1, x_2]$, where x_1 is a vector representing an external state and x_2 is a vector representing the internal states of each station. That is

$[(N, 0), (i,)]$ = the state that all N jobs are at station 0, with internal state $i \in S_0$.

$[(N-1, 1), (i, j)]$ = the state that $N-1$ jobs are at station 0, with internal state $i \in S_0, j \in S_1$.

$[(N-n, n), (i, j)]$ = the state that $N-n$ jobs are at station 0, with internal state $i \in S_0$, and n jobs at station 1, with internal state $j \in S_2$, where $2 \leq n \leq N-1$.

$[(0, N), (, j)]$ = the state that all N jobs are at station 1, with internal state $j \in S_2$.

Note that the cardinality of the state spaces is $d(0) + d(0)d(1) + (N-2)d(0)d(2) + d(2)$. Next, the steady state probability vectors are defined similarly.

Definition 4: For $N > 2$, the steady state probability vectors, describing the internal states, are defined as follows:

$[b_0(N, 0)]_i$ = the steady state probability that the network is in the state $[(N, 0), (i,)]$, where $i \in S_0$.

$\pi_1(N, 0)$ = $b_0(N,0) \otimes p_1$, where $\otimes$ denotes Kronecker product[10] (see Appendix).

$[\pi_1(N-1,1)]_{i,j}$ = the steady state probability that the network is in the state $[(N-1,1),(i,j)]$, where $i \in S_0$ and $j \in S_1$.

$[\pi_2(N-n,n)]_{i,j}$ = the steady state probability that the network is in the state $[(N-n,n),(i,j)]$, where $2 \leq n \leq N-1$, $i \in S_0$, and $j \in S_2$.

$[b_2(0,N)]_j$ = the steady state probability that the network is in the state $[(0,N),(,j)]$, where $j \in S_2$.

$\pi_2(0,N) = p_0 \otimes b_2(0,N)$.

For $k = 1, 2$ and $n = 0, 1, \cdots, N$, the steady state probability $[\pi_k(N-n,n)]_{i,j}$ can be ordered lexicographically to create a vector $\pi_k(N-n,n)$. The subscript of these probability vectors denotes the dimension of the objects. For instance, the subscript k is used to denote the steady state probability vector of order $d(0)d(k)$. This vector is essentially decomposed into $d(0)$ subvectors, each of order $d(k)$. The vectors $b_0(N,0)$ and $b_2(0,N)$ are of order $d(0)$ and $d(2)$.

Definition 5: For $N > 2$, the steady state probabilities, describing the external states, are defined as follows:

$\Pr(N,0) = \pi_1(N,0)(e^{(1)})^T$
= the steady state probability that there are all N jobs at station 0.

$\Pr(N-1,1) = \pi_1(N-1,1)(e^{(1)})^T$
= the steady state probability that there is one job at station 1.

$\Pr(N-n,n) = \pi_2(N-n,n)(e^{(2)})^T$
= the steady state probability that there are n $(2 \leq n \leq N-1)$ jobs at station 1.

$\Pr(0,N) = \pi_2(0,N)(e^{(2)})^T$
= the steady state probability that there are all N jobs at station 1,

where $e^{(i)} = e_0 \otimes e_i = e_0\hat{e}_i$, $i = 1, 2$, is a vector of order $d(0)d(i)$ containing all ones (see Appendix).

Next, objects of the state space S_2 are defined similarly. These are also necessary to connect between S_1 and S_2.

Definition 6: For the state space S_2, the following matrices are defined:

$\mathbf{M}_2$ = the diagonal matrix whose (i,i)th element is the probability rate of leaving state $i = (i_1, i_2) \in S_2$. That is, $(\mathbf{M}_2)_{ii} = \mu_{1,i_1} + \mu_{1,i_2}$.

$(\mathbf{P}_2)_{ij}$ = the probability of transition from state (k,i) to state (k,j), where $k \in S_0$, $i = (i_1, i_2) \in S_2$ and $j = (j_1, j_2) \in S_2$.

$$= \begin{cases} 0 & \text{if } \{i_1, i_2\} \cap \{j_1, j_2\} = \phi, \\ \dfrac{\mu_{1,\bar{i}}(\mathbf{P}_1)_{ij}}{(\mathbf{M}_2)_{ii}} & \text{if } \{i_1, i_2\} \cap \{j_1, j_2\} \neq \phi \text{ and } i_1 \neq i_2, \\ (\mathbf{P}_1)_{\bar{i}\bar{j}} & \text{if } \{i_1, i_2\} \cap \{j_1, j_2\} \neq \phi \text{ and } i_1 = i_2, \end{cases}$$

where $\bar{i}$ is the other member of the i-pair, $\bar{j}$ is the other member of the j-pair, and $\cap$ means set intersection.

$(\mathbf{R}_2)_{ij}$ = the probability that upon a job entering station 1, the state is changed from (k, i) to (k, j), where $k \in S_0$, $i \in S_1$, and $j = (j_1, j_2) \in S_2$.

$$= \begin{cases} (p_1)_j & \text{if } j_1 \text{ or } j_2 = 1, \\ 0 & \text{otherwise.} \end{cases}$$

$(\mathbf{Q}_2)_{ij}$ = the probability that upon a job completing at station 1, the state is changed from (k, i) to (k, j), where $k \in S_0$, $j \in S_1$, and $i = (i_1, i_2) \in S_2$.

$$= \begin{cases} \dfrac{\mu_{1,\bar{i}}(q_1)_{\bar{i}}}{(\mathbf{M}_2)_{ii}} & \text{if } i_1 \neq i_2, \\ (q_1)_{\bar{i}} & \text{if } i_1 = i_2. \end{cases}$$

A simple example is provided to show how to construct the objects of the state space S_2.

Example 2: If station 1 has two Erlangian servers ($m_1 = 2$) in parallel as shown in Fig. 4, then the internal state spaces and the dimension of the spaces are

$$S_1 = \{1, 2\},$$

$$S_2 = \{(1, 1), (1, 2), (2, 2)\},$$

$$d(1) = 2, \quad \text{and} \quad d(2) = 3.$$

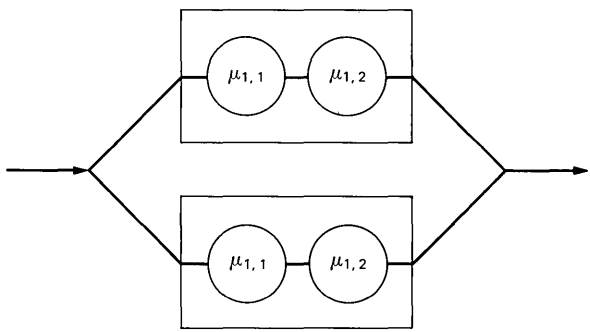


Fig. 4—Station 1 containing two Erlangian servers in parallel.

Using definition 6, the objects of S_2 are determined by

$$\mathbf{M}_2 = \begin{bmatrix} \mu_{1,1} + \mu_{1,1} & 0 & 0 \\ 0 & \mu_{1,1} + \mu_{1,2} & 0 \\ 0 & 0 & \mu_{1,2} + \mu_{1,2} \end{bmatrix}$$

$$\mathbf{P}_2 = \begin{bmatrix} 0 & (\mathbf{P}_1)_{12} & 0 \\ \dfrac{\mu_{1,2}(\mathbf{P}_1)_{21}}{\mu_{1,1} + \mu_{1,2}} & 0 & \dfrac{\mu_{1,1}(\mathbf{P}_1)_{12}}{\mu_{1,1} + \mu_{1,2}} \\ 0 & (\mathbf{P}_1)_{21} & 0 \end{bmatrix}$$

$$\mathbf{R}_2 = \begin{bmatrix} (p_1)_2 & (p_1)_2 & 0 \\ 0 & (p_1)_1 & (p_1)_2 \end{bmatrix}$$

$$\mathbf{Q}_2 = \begin{bmatrix} (q_1)_1 & 0 \\ \dfrac{\mu_{1,2}(q_1)_2}{\mu_{1,1} + \mu_{1,2}} & \dfrac{\mu_{1,1}(q_1)_1}{\mu_{1,1} + \mu_{1,2}} \\ 0 & (q_1)_2 \end{bmatrix}.$$

Note that $\mathbf{R}_2$ is $d(1) \times d(2)$ dimensional rectangular matrix and $\mathbf{Q}_2$ is a $d(2) \times d(1)$ dimensional rectangular matrix. Thus $\mathbf{Q}_2\mathbf{R}_2$ is the probability matrix that upon a completion at station 1, a job leaves and another takes its place. This event can happen only if there are more than two jobs at station 1.

Lemma 3: For the state space S_1 and S_2, let $\mathbf{I}_i$ be an identity matrix of order $d(i)$ and e_i a vector containing $d(i)$ ones. The following relationships hold:

$$\mathbf{R}_2 e_2^T = e_1^T, \tag{10}$$

$$(\mathbf{P}_2 + \mathbf{Q}_2\mathbf{R}_2)e_2^T = e_2^T, \tag{11}$$

$$\mathbf{Q}_2\mathbf{R}_2 e_2^T = \mathbf{Q}_2 e_1^T = (\mathbf{I}_2 - \mathbf{P}_2)e_2^T, \quad \text{and} \tag{12}$$

$$p_1\mathbf{R}_2 e_2^T = 1. \tag{13}$$

Proof: This lemma follows immediately from lemma 1 and the law of total probability.

Next, note that we have encountered objects of different state spaces S_0, S_1, and S_2. They are of different dimensions $d(0)$, $d(1)$, and $d(2)$. Objects of order $d(0)d(1)$ and $d(0)d(2)$ are said to be on the *product spaces*. The product space of order $d(i)d(j)$ is represented by $F_{d_i x d_j}$. Before any operation can be performed, all objects have to be replaced by their images under embedding into product spaces. For instance, objects of order 1 are scalars. Since scalars constitute subspaces of

spaces S_0, S_1, and S_2 and of the product spaces, and there is a 1–1 correspondence between scalars and diagonal matrices all of whose diagonal elements are equal, scalars are isomorphic to scalar multiples of matrix **I**. Similarly, the spaces S_0, S_1, and S_2 are subspaces of the product spaces, so that objects from these spaces have a counterpart in the product spaces while algebraic relationships are preserved. These counterparts will be denoted by adding both superscripts and a hat, these counterparts are the image of an embedding (for details, see Appendix). For instance, $\mathbf{A}_0$ is a matrix in the state space S_0, and $\mathbf{A}_1$ is a matrix in the state space S_1. $\hat{\mathbf{A}}_0^{(1)}$ is a matrix in the product space $F_{d_0 x d_1}$ and has characteristics inherited from matrix $\mathbf{A}_0$. Also $\hat{\mathbf{A}}_1^{(0)}$ is a matrix in the product space $F_{d_0 x d_1}$ and has characteristics from $\mathbf{A}_1$. If there is no possibility of confusion, the superscript (0) is omitted. That is, $\hat{\mathbf{A}}_1^{(0)}$ and $\hat{\mathbf{A}}_2^{(0)}$ are denoted by $\hat{\mathbf{A}}_1$ and $\hat{\mathbf{A}}_2$.

IV. BALANCE EQUATION

In order for the network to be in steady state, the probability of leaving a particular state must be equal to the probability of entering that state. Thus the total flow into a particular internal state must be equal to the total flow out of that state. The steady state balance equations can then be derived for the internal states. All internal states belonging to one external state are combined in one mathematical entity, a vector. The balance equations can then be written in a matrix form, using Kronecker products and generalized embedding of vectorspaces (see Appendix).

Theorem 1: For the closed network with a nonexponential server and a station containing two identical nonexponential servers and $N > 2$, the global balance equations are:

$$\pi_1(N, 0)\hat{\mathbf{B}}_0^{(1)} = \pi_1(N-1, 1)\hat{\mathbf{B}}_1\hat{\mathbf{Q}}_1 \tag{14}$$

$$\pi_1(N-1, 1)[\hat{\mathbf{B}}_0^{(1)} + \hat{\mathbf{B}}_1] = \pi_2(N-2, 2)\hat{\mathbf{M}}_2\hat{\mathbf{Q}}_2 + \pi_1(N, 0)\hat{\mathbf{B}}_0^{(1)}\hat{\mathbf{Q}}_0^{(1)} \tag{15}$$

$$\pi_2(N-2, 2)[\hat{\mathbf{B}}_0^{(2)} + \hat{\mathbf{B}}_2] = \pi_2(N-3, 3)\hat{\mathbf{M}}_2\hat{\mathbf{Q}}_2\hat{\mathbf{R}}_2 + \pi_1(N-1, 1)\hat{\mathbf{B}}_0^{(1)}\hat{\mathbf{Q}}_0^{(1)}\hat{\mathbf{R}}_2 \tag{16}$$

$$\pi_2(N-n, n)[\hat{\mathbf{B}}_0^{(2)} + \hat{\mathbf{B}}_2] = \pi_2(N-n-1, n+1)\hat{\mathbf{M}}_2\hat{\mathbf{Q}}_2\hat{\mathbf{R}}_2 + \pi_2(N-n+1, n-1)\hat{\mathbf{B}}_0^{(2)}\hat{\mathbf{Q}}_0^{(2)}, \quad \text{for} \quad n = 3, \cdots, N-1 \tag{17}$$

$$\pi_2(0, N)\hat{\mathbf{B}}_2 = \pi_2(1, N-1)\hat{\mathbf{B}}_0^{(2)}\hat{\mathbf{Q}}_0^{(2)}, \tag{18}$$

$$\textit{where} \quad \hat{\mathbf{B}}_i = \hat{\mathbf{M}}_i(\hat{\mathbf{I}} - \hat{\mathbf{P}}_i), \qquad i = 1, 2. \tag{19}$$

Proof: The proof will be done in five different parts, one part for each of eqs. (14) through (18).

(1) By equating the flow out of and into a state $[(N, 0), (i, \ )]$ it follows that

$$[b_0(N, 0)]_i(\mathbf{M}_0)_{ii} = \sum_{k=1}^{d(0)} [b_0(N, 0)]_k(\mathbf{M}_0)_{kk}(\mathbf{P}_0)_{ki} + \sum_{k=1}^{d(1)} [\pi_1(N-1, 1)]_{i,k}(\mathbf{M}_1)_{kk}(q_1)_k. \quad (20)$$

π_1 is a vector in the product space $F_{d_0xd_1}$, whereas the other factors are objects from space S_0. In order to write eq. (20) in a matrix form, these objects must be augmented into objects from product space $F_{d_0xd_1}$. By postmultiplying eq. (20) by $\hat{p}_1$ and using generalized embedding, eq. (20) can be written in a matrix form where vectors and matrices are in product space $F_{d_0xd_1}$:

$$\pi_1(N, 0)\hat{\mathbf{M}}_0^{(1)} = \pi_1(N, 0)\hat{\mathbf{M}}_0^{(1)}\hat{\mathbf{P}}_0^{(1)} + \pi_1(N-1, 1)\hat{\mathbf{M}}_1\hat{q}_1^T\hat{p}_1. \quad (21)$$

It can be simplified by substituting $(\hat{\mathbf{I}}_i - \hat{\mathbf{P}}_i)\hat{e}_i^T$ for $\hat{q}_i^T$ and $\hat{\mathbf{B}}_i$ for $\hat{\mathbf{M}}_i(\hat{\mathbf{I}}_i - \hat{\mathbf{P}}_i)$:

$$\pi_1(N, 0)\hat{\mathbf{B}}_0^{(1)} = \pi_1(N-1, 1)\hat{\mathbf{B}}_1\hat{e}_1^T\hat{p}_1. \quad (22)$$

By substituting $\hat{\mathbf{Q}}_1$ for $\hat{e}_1^T\hat{p}_1$, balance equation (14) is obtained.

(2) By equating the flow out of and into a state $[(N-1, 1), (i, j)]$ it follows that

$$[\pi_1(N-1, 1)]_{i,j}\{(\mathbf{M}_0)_{ii} + (\mathbf{M}_1)_{jj}\} = \sum_{k=1}^{d(0)} [\pi_1(N-1, 1)]_{k,j}(\mathbf{M}_0)_{kk}(\mathbf{P}_0)_{ki} + \sum_{k=1}^{d(1)} [\pi_1(N-1, 1)]_{i,k}(\mathbf{M}_1)_{kk}(\mathbf{P}_1)_{kj} + \sum_{k=1}^{d(0)} [b_0(N, 0)]_k(\mathbf{M}_0)_{kk}(q_0)_k(p_0)_i(p_1)_j + \sum_{k=1}^{d(2)} [\pi_2(N-2, 2)]_{i,k}(\mathbf{M}_2)_{kk}(\mathbf{Q}_2)_{kj}. \quad (23)$$

By simplifying eq. (23) and using generalized embedding, balance eq. (15) is obtained.

(3) By equating the flow out of and into a state $[(N-2, 2), (i, j)]$ it follows that

$$[\pi_2(N-2,2)]_{i,j}\{(\mathbf{M}_0)_{ii}+(\mathbf{M}_2)_{jj}\}$$
$$=\sum_{k=1}^{d(0)}[\pi_2(N-2,2)]_{k,j}(\mathbf{M}_0)_{kk}(\mathbf{P}_0)_{ki}$$
$$+\sum_{k=1}^{d(2)}[\pi_2(N-2,2)]_{i,k}(\mathbf{M}_2)_{kk}(\mathbf{P}_2)_{kj}$$
$$+\sum_{k1=1}^{d(1)}\sum_{k0=1}^{d(0)}[\pi_1(N-1,1)]_{k0,k1}(\mathbf{M}_0)_{k0k0}(q_0)_{k0}(p_0)_i(\mathbf{R}_2)_{k1j}$$
$$+\sum_{k=1}^{d(2)}[\pi_2(N-3,3)]_{i,k}(\mathbf{M}_2)_{kk}(\mathbf{Q}_2\mathbf{R}_2)_{kj}. \quad (24)$$

By simplifying eq. (24) and using generalized embedding, balance eq. (16) is obtained.

(4) By equating the flow out of and into a state $[(N-n,n),(i,j)]$ with $n=3,\cdots,N-1$, it follows that

$$[\pi_2(N-n,n)]_{i,j}\{(\mathbf{M}_0)_{ii}+(\mathbf{M}_2)_{jj}\}$$
$$=\sum_{k=1}^{d(0)}[\pi_2(N-n,n)]_{k,j}(\mathbf{M}_0)_{kk}(\mathbf{P}_0)_{ki}$$
$$+\sum_{k=1}^{d(2)}[\pi_2(N-n,n)]_{i,k}(\mathbf{M}_2)_{kk}(\mathbf{P}_2)_{kj}$$
$$+\sum_{k=1}^{d(0)}[\pi_2(N-n+1,n-1)]_{k,j}(\mathbf{M}_0)_{kk}(q_0)_k(p_0)_i$$
$$+\sum_{k=1}^{d(2)}[\pi_2(N-n-1,n+1)]_{i,k}(\mathbf{M}_2)_{kk}(\mathbf{Q}_2\mathbf{R}_2)_{kj}. \quad (25)$$

By simplifying eq. (25) and using generalized embedding, balance eq. (17) is obtained.

(5) By equating the flow out of and into a state $[(0,N),(,j)]$ it follows that

$$[b_2(0,N)]_j(\mathbf{M}_2)_{jj}=\sum_{k=1}^{d(2)}[b_2(0,N)]_k(\mathbf{M}_2)_{kk}(\mathbf{P}_0)_{ki}$$
$$+\sum_{k=1}^{d(0)}[\pi_2(1,N-1)]_{k,j}(\mathbf{M}_0)_{kk}(q_0)_k. \quad (26)$$

By simplifying eq. (26) and using generalized embedding, balance eq. (18) is obtained. That completes the proof.

Next consider the entire Markov chain of this network. The state probability vector of the entire Markov chain can be represented by

$$\pi = (\pi_1(N, 0), \pi_1(N-1, 1), \pi_2(N-2, 2),$$
$$\cdots, \pi_2(N-n, n), \cdots, \pi_2(0, N)), \qquad 2 \le n \le N.$$

This satisfies balance eq. (10) and the law of total probability. The infinitesimal generator $\mathbf{Q}^*$ of the Markov chain can be written as follows:

$$\mathbf{Q}^* = \begin{bmatrix} \hat{\mathbf{B}}_0^{(1)} & -\hat{\mathbf{B}}_0^{(1)}\hat{\mathbf{Q}}_0^{(1)} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ -\hat{\mathbf{B}}_1\hat{\mathbf{Q}}_1 & \hat{\mathbf{B}}_0^{(1)} + \hat{\mathbf{B}}_1 & -\hat{\mathbf{B}}_0^{(1)}\hat{\mathbf{Q}}_0^{(1)}\hat{\mathbf{R}}_2 & \cdots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -\hat{\mathbf{M}}_2\hat{\mathbf{Q}}_2 & \hat{\mathbf{B}}_0^{(2)} + \hat{\mathbf{B}}_2 & \cdots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & -\hat{\mathbf{M}}_2\hat{\mathbf{Q}}_2\hat{\mathbf{R}}_2 & \cdots & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & -\hat{\mathbf{B}}_0^{(2)}\hat{\mathbf{Q}}_0^{(2)} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \hat{\mathbf{B}}_0^{(2)} + \hat{\mathbf{B}}_2 & -\hat{\mathbf{B}}_0^{(2)}\hat{\mathbf{Q}}_0^{(2)} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & -\hat{\mathbf{M}}_2\hat{\mathbf{Q}}_2\hat{\mathbf{R}}_2 & \hat{\mathbf{B}}_2 \end{bmatrix}$$

Wallace also termed such processes *Quasi Birth* and *Death* (QBD) processes.[11] He has shown that if a QBD process is *boundary leading*, the process has a matrix geometric form solution.

The global balance equations were derived by equating the flow in and out of a certain state. Thus the flow into an external state must be equal to the flow out of the external state because the external state is a collection of states.

Lemma 4: (*Flow-balance equations between the external states.*) *For* $N > 2$, *the flow out of a certain external state equals the flow into the external state. That is*

$$\pi_1(N, 0)\hat{\mathbf{B}}_0^{(1)}(e^{(1)})^T = \pi_1(N-1, 1)\hat{\mathbf{B}}_1(e^{(1)})^T \tag{27}$$

$$\pi_1(N-1, 1)\hat{\mathbf{B}}_0^{(1)}(e^{(1)})^T = \pi_2(N-2, 2)\hat{\mathbf{B}}_2(e^{(2)})^T \tag{28}$$

$$\pi_2(N-n, n)\hat{\mathbf{B}}_0^{(2)}(e^{(2)})^T = \pi_2(N-n-1, n+1)\hat{\mathbf{B}}_2(e^{(2)})^T,$$
$$\text{for} \quad n = 2, 3, \ldots, N-1. \tag{29}$$

Proof: The proof will be done in three parts, one part for each of eqs. (27) through (29).

(1) By postmultiplying global balance eq. (14) by $(e^{(1)})^T$, eq. (27) is obtained.

(2) By postmultiplying balance eq. (15) by $(e^{(1)})^T$, we get

$$\pi_1(N, 0)\hat{\mathbf{B}}_0^{(1)}(e^{(1)})^T - \pi_1(N-1, 1)\hat{\mathbf{B}}_1(e^{(1)})^T$$
$$= \pi_1(N-1, 1)\hat{\mathbf{B}}_0^{(1)}(e^{(1)})^T - \pi_2(N-2, 2)\hat{\mathbf{B}}_2(e^{(2)})^T. \tag{30}$$

By applying eq. (27) to eq. (30), we get

$$\pi_1(N-1,1)\hat{\mathbf{B}}_0^{(1)}(e^{(1)})^T = \pi_2(N-2,2)\hat{\mathbf{B}}_2(e^{(2)})^T,$$

which is eq. (28).

(3) By induction on n ($n = 2, 3, \cdots, N-1$),

Basis step: By postmultiplying balance eq. (16) by $(e^{(2)})^T$, it follows that

$$\pi_2(N-2,2)\hat{\mathbf{B}}_0^{(2)}(e^{(2)})^T - \pi_2(N-3,3)\hat{\mathbf{B}}_2(e^{(2)})^T = \pi_1(N-1,1)\hat{\mathbf{B}}_0^{(1)}(e^{(1)})^T - \pi_2(N-2,2)\hat{\mathbf{B}}_2(e^{(2)})^T. \quad (31)$$

By applying eq. (28) to eq. (31), we get

$$\pi_2(N-2,2)\mathbf{B}_0^{(2)}(e^{(2)})^T - \pi_2(N-3,3)\hat{\mathbf{B}}_2(e^{(2)})^T.$$

Inductive step: Suppose that

$$\pi_2(N-k,k)\hat{\mathbf{B}}_0^{(2)}(e^{(2)})^T = \pi_2(N-k-1,k+1)\hat{\mathbf{B}}_2(e^{(2)})^T, \quad 2 \le k \le N-2. \quad (32)$$

By postmultiplying balance eq. (17) by $(e^{(2)})^T$ and rearranging, we get

$$\pi_2(N-k-1,k+1)\hat{\mathbf{B}}_0^{(2)}(e^{(2)})^T = \pi_2(N-k-2,k+2)\hat{\mathbf{B}}_2(e^{(2)})^T. \quad (33)$$

Thus, we have

$$\pi_2(N-n,n)\hat{\mathbf{B}}_0^{(2)}(e^{(2)})^T = \pi_2(N-n-1,n+1)\hat{\mathbf{B}}_2(e^{(2)})^T, \quad \text{for} \quad n = 2, 3, \cdots, N-1,$$

which is eq. (29). This proves the lemma.

V. STEADY STATE SOLUTIONS

The solution of the global balance equations (Theorem 1) can be derived now. A definition is introduced to obtain the steady state solutions.

Definition 7: For $N > 2$, define recursively the following matrices:

$$\mathbf{U}_2(0) = \hat{\mathbf{B}}_0^{(2)}\hat{\mathbf{Q}}_0^{(2)}(\hat{\mathbf{B}}_2)^{-1}, \quad (34)$$

$$\mathbf{U}_2(n) = \hat{\mathbf{B}}_0^{(2)}\hat{\mathbf{Q}}_0^{(2)}[\hat{\mathbf{B}}_0^{(2)} + \hat{\mathbf{B}}_2 - \mathbf{U}_2(n-1)\hat{\mathbf{M}}_2\hat{\mathbf{Q}}_2\hat{\mathbf{R}}_2]^{-1}, \quad \text{for} \quad n = 1, 2, \ldots, N-2, \quad (35)$$

$$\mathbf{U}_1(N-1) = \hat{\mathbf{B}}_0^{(1)}\hat{\mathbf{Q}}_0^{(1)}[\hat{\mathbf{B}}_0^{(1)} + \hat{\mathbf{B}}_1 - \hat{\mathbf{R}}_2\mathbf{U}_2(N-2)\hat{\mathbf{M}}_2\hat{\mathbf{Q}}_2]^{-1}. \quad (36)$$

Since $\pi_1(N, 0)$ depends on one unknown vector $b_0(N, 0)$, we can express all state probability vector π's in terms of it. With notations defined above we are ready to derive the steady state solutions.

Theorem 2: For the closed network with a nonexponential server and a station containing two identical nonexponential servers and $N > 2$, the steady state probability vectors can be described by

$$\pi_1(N, 0) = b_0(N, 0)\hat{p}_1$$

$$\pi_1(N-1, 1) = \pi_1(N, 0)\mathbf{U}_1(N-1) \tag{37}$$

$$\pi_2(N-2, 2) = \pi_1(N, 0)\mathbf{U}_1(N-1)\hat{\mathbf{R}}_2\mathbf{U}_2(N-2) \tag{38}$$

$$\pi_2(N-n, n) = \pi_1(N, 0)\mathbf{U}_1(N-1)\hat{\mathbf{R}}_2\mathbf{U}_2(N-2)\mathbf{U}_2(N-3)$$
$$\cdots\mathbf{U}_2(N-n), \quad \textit{for} \quad n = 3, 4, \cdots, N. \tag{39}$$

Proof: The proof will be done in three parts, one part for each of eqs. (37) through (39).

(1) By induction on $N-n$ ($N-n = 0, 1, \cdots, N-3$),

Basis step: From eq. (18) and definition 7, we get

$$\pi_2(0, N) = \pi_2(1, N-1)\hat{\mathbf{B}}_0^{(2)}\hat{\mathbf{Q}}_0^{(2)}(\hat{\mathbf{B}}_2)^{-1} = \pi_2(1, N-1)\mathbf{U}_2(0). \tag{40}$$

Inductive step: Suppose that

$$\pi_2(k, N-k) = \pi_2(k+1, N-k-1)\mathbf{U}_2(k),$$
$$0 \le k < N-3. \tag{41}$$

From eq. (17) and (41), we get

$$\pi_2(k+1, N-k-1)\{\hat{\mathbf{B}}_0^{(2)} + \hat{\mathbf{B}}_2\}$$
$$= \pi_2(k, N-k)\hat{\mathbf{M}}_2\hat{\mathbf{Q}}_2\hat{\mathbf{R}}_2$$
$$+ \pi_2(k+2, N-k-2)\hat{\mathbf{B}}_0^{(2)}\hat{\mathbf{Q}}_0^{(2)}$$
$$= \pi_2(k+1, N-k-1)\mathbf{U}_2(k)$$
$$\cdot\hat{\mathbf{M}}_2\hat{\mathbf{Q}}_2\hat{\mathbf{R}}_2 + \pi_2(k+2, N-k-2)\hat{\mathbf{B}}_0^{(2)}\hat{\mathbf{Q}}_0^{(2)}.$$

Thus, using definition 7, it follows that

$$\pi_2(k+1, N-k-1) = \pi_2(k+2, N-k-2)\hat{\mathbf{B}}_0^{(2)}\hat{\mathbf{Q}}_0^{(2)}$$
$$\cdot[\hat{\mathbf{B}}_0^{(2)} + \hat{\mathbf{B}}_2 - \mathbf{U}_2(k)\hat{\mathbf{M}}_2\hat{\mathbf{Q}}_2\hat{\mathbf{R}}_2]^{-1}$$
$$= \pi_2(k+2, N-k-2)\mathbf{U}_2(k+1).$$

Therefore, we have

$$\pi_2(N-n, n) = \pi_2(N-n+1, n-1)\mathbf{U}_2(N-n),$$
$$\text{for} \quad N-n = 0, 1, 2, \cdots, N-3. \tag{42}$$

(2) From eqs. (16) and (42), we get

$$\begin{aligned}\pi_2(N-2,2)(\hat{\mathbf{B}}_0^{(2)}+\hat{\mathbf{B}}_2) &= \pi_2(N-3,3)\hat{\mathbf{M}}_2\hat{\mathbf{Q}}_2\hat{\mathbf{R}}_2 \\ &\quad + \pi_1(N-1,1)\hat{\mathbf{B}}_0^{(1)}\hat{\mathbf{Q}}_0^{(1)}\hat{\mathbf{R}}_2 \\ &= \pi_2(N-2,2)\mathbf{U}_2(N-3)\hat{\mathbf{M}}_2\hat{\mathbf{Q}}_2\hat{\mathbf{R}}_2 \\ &\quad + \pi_1(N-1,1)\hat{\mathbf{B}}_0^{(1)}\hat{\mathbf{Q}}_0^{(1)}\hat{\mathbf{R}}_2\end{aligned}$$

Thus, using definition 7, it follows that

$$\begin{aligned}\pi_2(N-2,2) &= \pi_1(N-1,1)\hat{\mathbf{B}}_0^{(1)}\hat{\mathbf{Q}}_0^{(1)}\hat{\mathbf{R}}_2 \\ &\quad \cdot[\hat{\mathbf{B}}_0^{(2)}+\hat{\mathbf{B}}_2-\mathbf{U}_2(N-3)\hat{\mathbf{M}}_2\hat{\mathbf{Q}}_2\hat{\mathbf{R}}_2]^{-1} \\ &= \pi_1(N-1,1)\hat{\mathbf{R}}_2\hat{\mathbf{B}}_0^{(2)}\hat{\mathbf{Q}}_0^{(2)} \\ &\quad \cdot[\hat{\mathbf{B}}_0^{(2)}+\hat{\mathbf{B}}_2-\mathbf{U}_2(N-3)\hat{\mathbf{M}}_2\hat{\mathbf{Q}}_2\hat{\mathbf{R}}_2]^{-1} \\ &= \pi_1(N-1,1)\hat{\mathbf{R}}_2\mathbf{U}_2(N-2). \qquad (43)\end{aligned}$$

(3) From eqs. (15) and (43), we get

$$\begin{aligned}\pi_1(N-1,1)(\hat{\mathbf{B}}_0^{(1)}+\hat{\mathbf{B}}_1) &= \pi_2(N-2,2)\hat{\mathbf{M}}_2\hat{\mathbf{Q}}_2+\pi_1(N,0)\hat{\mathbf{B}}_0^{(1)}\hat{\mathbf{Q}}_0^{(1)} \\ &= \pi_1(N-1,1)\hat{\mathbf{R}}_2\mathbf{U}_2(N-2)\hat{\mathbf{M}}_2\hat{\mathbf{Q}}_2 \\ &\quad + \pi_1(N,0)\hat{\mathbf{B}}_0^{(1)}\hat{\mathbf{Q}}_0^{(1)}.\end{aligned}$$

Thus, using definition 7, it follows that

$$\begin{aligned}&\pi_1(N-1,1) = \pi_1(N,0)\hat{\mathbf{B}}_0^{(1)}\hat{\mathbf{Q}}_0^{(1)} \\ &\cdot[\hat{\mathbf{B}}_0^{(1)}+\hat{\mathbf{B}}_1-\hat{\mathbf{R}}_2\mathbf{U}_2(N-2)\hat{\mathbf{M}}_2\hat{\mathbf{Q}}_2]^{-1} = \pi_1(N,0)\mathbf{U}_1(N-1). \qquad (44)\end{aligned}$$

From eqs. (42), (43), and (44), the steady state vectors can be written as described in eqs. (37), (38), and (39). That completes the proof.

The solution of the network, as described in Theorem 2, is expressed in terms of the vector $b_0(N, 0)$ and the matrix $U_k(N-n)$'s. The vector $b_0(N, 0)$ is thus far unknown, while matrix $U_k(N-n)$'s can be generated. In general, the vector $b_0(N, 0)$ contains $d(0)$ unknowns. Definitions and a lemma are introduced to make the solution explicit.

Definition 8: For $N > 2$, define recursively the following matrix:

$$R_1(N-1) = (\hat{\mathbf{B}}_0^{(1)}+\hat{\mathbf{B}}_1-\hat{\mathbf{R}}_2\mathbf{U}_2(N-2)\hat{\mathbf{M}}_2\hat{\mathbf{Q}}_2)^{-1} \qquad (45)$$

so that $\mathbf{U}_1(N-1) = \hat{\mathbf{B}}_0^{(1)}\hat{\mathbf{Q}}_0^{(1)}\mathbf{R}_1(N-1)$.

Definition 9: For $N > 2$, define the following matrix of order $d(1) \times d(2)$:

$$\mathbf{K}(N) = \mathbf{R}_1(N-1)\{\hat{\mathbf{B}}_1\hat{\mathbf{Q}}_1(\hat{\mathbf{B}}_0^{(1)})^{-1}\hat{\mathbf{R}}_2 + \hat{\mathbf{R}}_2[\hat{\mathbf{I}}_2 + \mathbf{U}_2(N-2) + \cdots$$
$$+ \mathbf{U}_2(N-2)\mathbf{U}_2(N-3)\cdots\mathbf{U}_2(2)$$
$$+ \mathbf{U}_2(N-2)\mathbf{U}_2(N-3)\cdots\mathbf{U}_2(2)\mathbf{U}_2(1)$$
$$+ \mathbf{U}_2(N-2)\mathbf{U}_2(N-3)\cdots\mathbf{U}_2(2)\mathbf{U}_2(1)\mathbf{U}_2(0)]\}. \tag{46}$$

Lemma 5: The following relation holds:

$$b_0(N, 0)\mathbf{B}_0\mathbf{Q}_0\hat{p}_1\mathbf{K}(N)(e^{(2)})^T = 1. \tag{47}$$

Proof: Using the law of total probability and definition 4, we get

$$1 = \sum_{n=0}^{N} r(N-n, n) = \sum_{n=0}^{1} \pi_1(N-n, n)(e^{(1)})^T$$
$$+ \sum_{n=2}^{N} \pi_2(N-n, n)(e^{(2)})^T$$
$$= \{\sum_{n=0}^{1} \pi_1(N-n, n)\hat{\mathbf{R}}_2 + \sum_{n=2}^{N} \pi_2(N-n, n)\}(e^{(2)})^T$$
$$= \pi_1(N, 0)\hat{\mathbf{B}}_0^{(1)}\hat{\mathbf{Q}}_0^{(1)}\mathbf{K}(N)(e^{(2)})^T = b_0(N, 0)\mathbf{B}_0\mathbf{Q}_0\hat{p}_1\mathbf{K}(N)(e^{(2)})^T.$$

This proves the lemma.

To make the solution explicit, a normalization vector must be determined. The normalization vector can be derived now.

Theorem 3: The normalization vector can be determined by

$$b_0(N, 0)\mathbf{B}_0\mathbf{Q}_0 = c(N)p_0, \tag{48}$$

where $c(N) = \dfrac{1}{p_0\hat{p}_1\mathbf{K}(N)(e^{(2)})^T}$ *is a scalar.*

Proof: The normalization vector is proportional to p_0:

$$b_0(N, 0)\mathbf{B}_0\mathbf{Q}_0 = b_0(N, 0)\mathbf{B}_0e_0^Tp_0 = c(N)p_0, \tag{49}$$

where $c(N) = b_0(N, 0)\mathbf{B}_0e_0^T$ is a scalar. Thus, the proportionality is normalization constant $c(N)$. By substituting eq. (49) for eq. (47), it follows that

$$c(N)p_0\hat{p}_1\mathbf{K}(N)(e^{(2)})^T = 1.$$

The proof is now completed.

VI. CONCLUSION

An explicit solution is derived for a closed network consisting of a nonexponential server and a service station with two identical nonexponential servers in parallel (e.g. CPU and I/O devices). There is a

finite number of jobs, and the queueing discipline at each node is FCFS. By applying an algebraic approach to the method of stages, the properties of each nonexponential server are represented by vectors and matrices in a space identified with that server. Product spaces are introduced to combine the properties of these servers and to allow their interactions to be described. Both external states and internal states are introduced to represent the states of the network. That is, the queueing problem is thus transformed from one involving the time of service remaining for a job to one involving the position of an active job in the subnetwork of exponential stages. In the mathematical view, the problem of integral equations (continuous) is transformed to one of algebraic equations (discrete) over a finite dimension.

From Theorems 1, 2, and 3, it turns out that the solution of this network is described in what I call a quasi matrix geometric form, in which the steady state vector is given by

$$\pi_k = \pi_0 \mathbf{U}(N-1)\mathbf{U}(N-2) \cdots \mathbf{U}(N-k), \qquad \text{for} \quad 1 \le k \le N,$$

where π_0 is a normalization vector chosen to make the steady state probabilities sum to 1, and each $\mathbf{U}(n)$ is a matrix which is recursively defined by matrices involved. Obviously this form is a generalization of the matrix geometric form. All of techniques used in this paper are drawn from linear algebra. This algorithm is easy to implement, even for a person not familiar with queueing theory, because it involves only matrix multiplication and inversion (we can use commercial matrix packages).

VII. ACKNOWLEDGMENTS

The author would like to thank Phil J. Fleming and the referees for their valuable comments and suggestions.

REFERENCES

1. R. R. Muntz, "Queueing Networks: A Critique of the State of the Art and Directions for the Future," Computing Survey, *10,* No. 3 (September 1978), pp. 353–9.
2. M. Reiser, and H. Kobayashi, "The Effects of Service Time Distribution on System Performance," Info. Proc. 74, Amsterdam: North-Holland, 1974, pp. 230–4.
3. C. H. Sauer, and K. H. Chandy, "The Impact of Distributions and Disciplines on Multiple Processor Systems," Commun. ACM, *22,* No. 1 (January 1979), pp. 25–33.
4. M. F. Neuts, *Matrix-Geometric Solutions in Stochastic Models—An Algorithmic Approach,* Baltimore: John Hopkins University Press, 1981.
5. J. L. Carroll, A. Van de Liefvoort, and L. Lipsky, "Solution of M/G/1//N-Type Loops With Extensions to M/G/1 and GI/M/1 Queues," Op. Res., *30,* No. 3 (May–June 1982), pp. 490–513.
6. M. F. Neuts, "Explicit Steady-State Solutions to Some Elementary Queueing Models," Op. Res., *30,* No. 3 (May–June 1982), pp. 480–9.
7. A. Van de Liefvoort, "An Algebraic Approach to the Steady State Solution of G/G/1/N Type Loops," Ph.D. Thesis, Univ. of Nebraska, March 1982.
8. S. W. Yoo, "An Explicit Steady State Solution of Closed Networks Consisting of Two General Stations With Parallel Servers," Ph.D. Dissertation, University of Kansas, June 1983.

9. D. R. Cox, "A Use of Complex Probabilities in the Theory of Stochastic Processes," Proc. Cambridge Phil. Soc., *51* (1955), pp. 313–9.
10. A. Graham, *Kronecker Products and Matrix Calculus With Applications*, Ellis Horwood Limited, 1981.
11. V. L. Wallace, "The Solution of Quasi Birth and Death Processes Arising from Multiple Access Computer Systems," Ph.D. Thesis, University of Michigan, Ann Arbor, Michigan, March 1969.

APPENDIX

A.1 Kronecker Products

The Kronecker product is the direct product of two disjoint operator spaces. In particular, if **K** is an m × n and **L** an r × s matrix, then the Kronecker product of **K** and **L**, denoted $\mathbf{K} \otimes \mathbf{L}$, is the matrix of size $mr \times ns$ which is obtained by multiplying each element $(\mathbf{K})_{ij}$ of matrix **K** by the full matrix **L**.

A.2 Embedding of Vectorspaces

In this paper, there are equations involving matrices of different dimensions. Before any operation can be performed, all matrices have to be replaced by their images under embedding into product spaces. According to the definitions in the previous sections, $d(i)$ denotes a dimension of state space $S_i (i = 0, 1)$. Let $F_{d_i x d_j}$ be the additive group of $d(i) \times d(j)$ matrices.

Definition A.1: Define the following mappings between $F_{d_0 x d_0}$, $F_{d_1 x d_1}$, $\cdots$ $F_{d_c x d_c}$ and $F_{d_0 d_1 x d_0 d_1}$, $F_{d_0 d_2 x d_0 d_2}$ as follows:

$$\begin{aligned} \hat{} &: \quad F_{d_i x d_i} \rightarrow F_{d_0 d_i x d_0 d_i} ; \qquad \mathbf{A}_i \rightarrow \hat{\mathbf{A}}_i = \mathbf{I}_0 \otimes \mathbf{A}_i \\ \hat{}(\mathrm{i}) &: F_{d_0 x d_0} \rightarrow F_{d_0 d_i x d_0 d_i} ; \qquad \mathbf{A}_0 \rightarrow \hat{\mathbf{A}}_0^{(i)} = \mathbf{A}_0 \otimes \mathbf{I}_i, \end{aligned}$$

where $\otimes$ stands for the Kronecker product and $i = 1, 2, \ldots, c$.

These mappings are group homomorphisms and preserve all algebraic characteristics of the matrix. The proof is omitted.

AUTHOR

Seung W. Yoo, B.S., 1972, Seoul National University, Korea; M.S., 1980, Ph.D. (Computer Science), 1983, University of Kansas; AT&T Bell Laboratories, 1983–1985, AT&T Information Systems, 1985—. Since joining AT&T Bell Laboratories, Mr. Yoo has been involved in architecture study and performance analysis of computer systems. He was transferred to AT&T Information Systems in 1985. Member, ACM.

AT&T Technical Journal
Vol. 64, No. 10, December 1985
Printed in U.S.A.

Statistical Model for Amplitude and Delay of Selective Fading

By P. BALABAN*

(Manuscript received September 11, 1985)

The transmission performance of digital radio systems is controlled by spectral distortion caused by multipath fading. To evaluate this performance for digital systems with high-order modulation schemes, a statistical model for frequency-selective fading is needed. New propagation data obtained in Gainesville, Florida, were used to generalize Rummler's model to include group delay response. The introduction of the delay response data into the model of the fading channel enabled the classification of the fades as minimum phase and nonminimum phase. We found that 24 percent of all fades have significant delay distortion, and can be characterized as being minimum phase or nonminimum phase. In the range of practical interest, there are as many minimum phase as nonminimum phase fades. The results of this work will facilitate a better understanding of the fading channel, which will be beneficial in the engineering of radio routes and digital radio design. The results also demonstrate the need for a description of the geographical occurrence of dispersion, which will differ from that for multipath fading at a single frequency. This is based on the observation, presented in this paper, that the relative amount of dispersive fading is significantly greater in Gainesville, Florida, than in Palmetto, Georgia. The availability of a dispersive fading map will facilitate the accurate engineering of digital radio routes.

I. INTRODUCTION

Digital radio communication systems are sensitive to selective fading. To evaluate their performance, a statistical model for frequency-

* AT&T Bell Laboratories.

selective fading is needed. Although the amplitude response of multipath fading in a microwave communication channel has been extensively modeled previously,[1–8] statistically based results for the group delay response were not available until recently,[9–11] and work on modeling the delay in a fading channel is just beginning.[12]

A successful model for multipath fading using amplitude data was developed by W. D. Rummler.[3,4] The delay response in this model was inferred from the measured amplitude. This led to an ambiguity, since both minimum phase and nonminimum phase fades are possible. In this paper, we generalize the use of this model by including group delay data. The model parameters are defined to characterize both minimum phase and nonminimum phase behavior, and statistical distributions of these parameters are compiled. Rummler's model, although developed for data obtained in the Palmetto experiment for a 6-GHz carrier with a 25.3-MHz bandwidth, has proven itself robust and applicable for a wide range of frequencies. Our new results are based on amplitude and delay measurements at 6 GHz in a 30-MHz bandwidth over a 23.3-mile path between Gainesville and Sparr, Florida.

Inclusion of the delay response into the statistical model of the dispersive line-of-sight channel will benefit both the engineering of digital radio routes and digital radio equipment design. The description of the occurrence of events with minimum and nonminimum phase response is of immediate practical interest. The need for delay description, such as provided by our model, will increase with the growth of sophistication of digital radio.

The experiment and the modeling procedures are described in Section II. The selection of the scans recorded by the instrumentation is outlined and the fixed delay Rummler model for selective fading[3] is described. It is shown that, by appropriately redefining the parameters of this model, it is suitable for characterization of amplitude and delay.

Section III presents the statistical distributions of the modified model parameters for the data measured in Gainesville. Comparison with Palmetto data have been made that show that the Gainesville fading is more dispersive than that at Palmetto.

II. CHARACTERIZATION AND MODELING

2.1 The experiment

The Gainesville experiment is implemented on a 23.3-mile path between Gainesville and Sparr, Florida, in a 6-GHz channel with a 30-MHz bandwidth. The experimental setup is described elsewhere.[9,13] The amplitude and delay characteristics of the faded link are measured by a Wandel-Goltermann link analyzer and recorded on magnetic

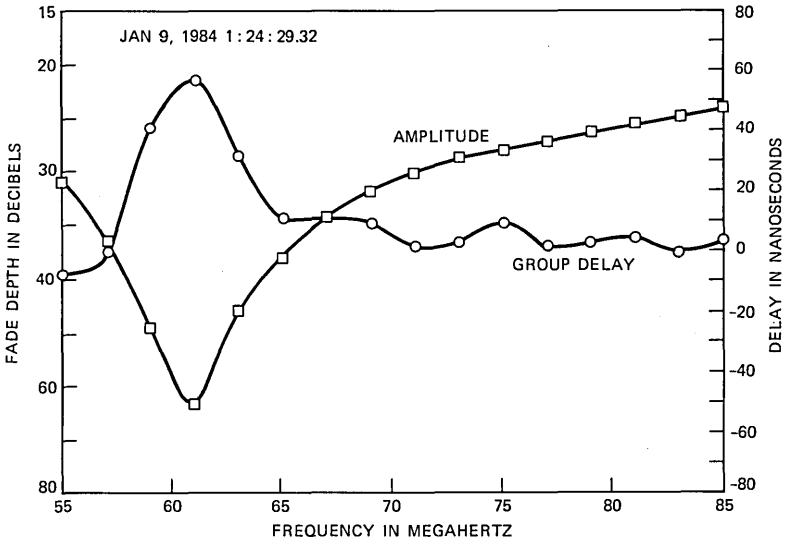


Fig. 1—A typical scan of a fading event.

tape. The amplitude and delay were measured during alternating frequency scans. Each scan duration was 0.1 second followed by a 0.1-second retrace. A measurement period was thus 0.4 second. The amplitude scale was set up for fades between 15 to 65 decibels and the delay between ±50 nanoseconds.* Amplitude fades smaller than 15 decibels were out of scale and recorded at 1 scan/min. Each scan was sampled at intervals of 2 MHz or 16 samples over the 30-MHz bandwidth.

The database used for this study consists of approximately 43,000 scans that represent 17,000 seconds of fading activity. The data were recorded in Gainesville for 11 months during the period from 1982 through 1984.

2.2 Selection of data for modeling

A fading event recorded in Gainesville is shown in Fig. 1. As a rule, fading events were slow enough to be considered static during the frequency scan. Although on some scans a certain amount of distortion that could be attributed to the dynamics of the system was noticeable, it was small enough to be negligible. Many scans were incompletely recorded because of the limited range of both amplitude and delay measurement equipment. Therefore, the recorded fading scans were first screened in order to eliminate fading scans that are not suitable

* Modified to ±100 nanoseconds in February 1985.

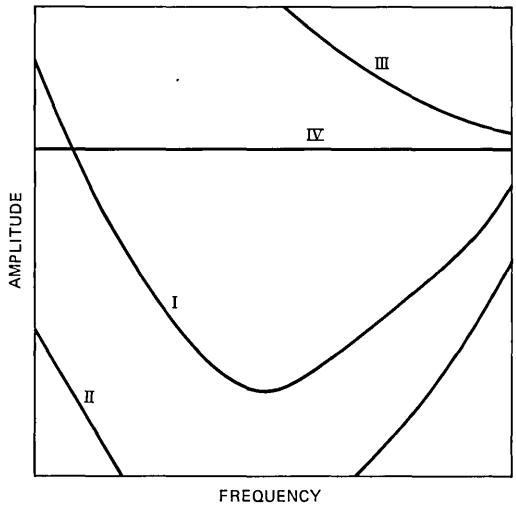


Fig. 2—Complete and incomplete amplitude scans. Scans II and III rejected. Scan IV marked.

for modeling. In addition, scans that do not need dispersive modeling, but have to be included in the total fading statistics, were identified during the screening process. This screening was done as follows.

A. Only those responses recorded at a rate of 2.5 scans/s were selected for processing. Responses recorded at different rates were regarded as unfaded.

B. Data scans were categorized on the basis of amplitude response. Figure 2 shows several amplitude scans as they appear in the recorded data. Only complete scans of types I and IV were modeled. Type II was rejected because part of the scan is out of scale at the high end (>65 dB). This is a rare event; about 30 such scans were recorded. Type III was eliminated because the trace is out of scale at the low end (<15 dB). This is the most common type that was eliminated; about 20 percent of all scans were type III. Flat fades of type IV were marked to indicate that dispersive modeling is not required.

C. Data scans were also categorized on the basis of delay response. Figure 3 shows several delay scans. Scans of types I and II were included as they are. Flat delay scans of type III that have a dispersive amplitude response were marked to indicate that delay modeling is not required. Scans of types IV and V are incomplete and were marked so that they can be modeled using an alternate technique.

Accordingly, 51,953 scans were selected using criterion A. From these, 9103 were eliminated because the amplitude was out of scale; 16,892 were marked because the amplitude was flat; 13,580 were marked because the delay was flat; and 2269 were marked because the

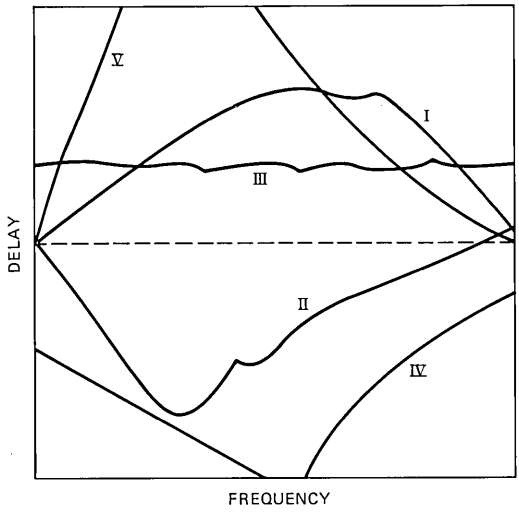


Fig. 3—Complete and incomplete delay scans. Scans III, IV, and V marked.

delay was out of scale. Consequently, 12,538 scans, or 24 percent of the total, had sufficient delay information to be modeled using both the delay and amplitude measurements. For the rest, only the amplitude measurements were used.

2.3 The model

The modeling was done using Rummler's model for selective fading.[3] In addition to wide acceptance and extensive validation this model has the advantage of separating the flat component of the fade from the dispersive component. The dispersive component can thus be statistically characterized by one parameter. The transfer function of the model is

$$H(\omega) = a(1 - be^{-j(\omega-\omega_0)\tau}), \tag{1}$$

where we can regard τ as the relative path delay of the second ray, b and $\omega_0\tau$ the relative amplitude and phase of that ray, and a as the amplitude scale factor.

The squared amplitude (or power) response is

$$Y(\omega) = |H(\omega)|^2 = a^2[1 + b^2 - 2b\cos(\omega - \omega_0)\tau]; \tag{2}$$

and the group delay is

$$D(\omega) = -\frac{d\phi(\omega)}{d\omega} = \tau\frac{b(b - \cos(\omega - \omega_0)\tau)}{1 + b^2 - 2b\cos(\omega - \omega_0)\tau}. \tag{3}$$

The delay parameter τ is constant in this model, at $\tau = 6.25$ ns. Typical

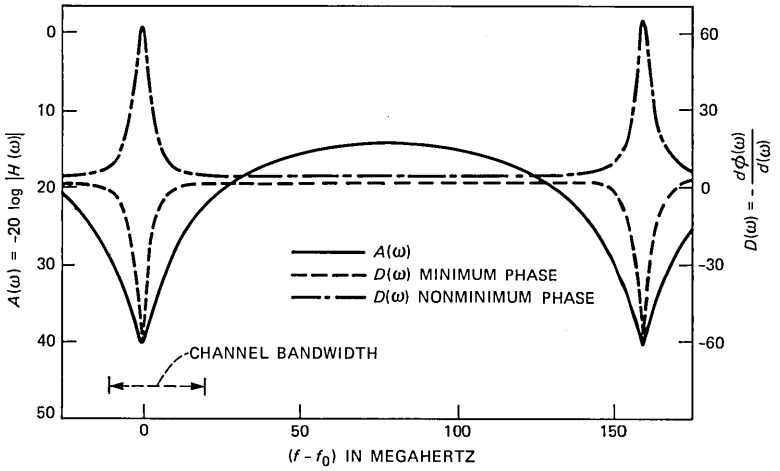


Fig. 4—Amplitude and delay characteristics of the fading model channel response: $a_1 = 0.1$, $b_1 = 0.9$, $a_2 = 0.09$, and $b_2 = 1.111$.

amplitude and delay frequency responses based on eqs. (2) and (3) are shown in Fig. 4.

Although the model was developed using measured amplitude data only, we found it adequate for modeling of selective fading using both measured amplitude and group delay data. The statistical distribution of the parameters naturally differs from those obtained for amplitude only, because we can distinguish between minimum and nonminimum phase scans.

2.4 Fitting the model to the amplitude data

The model parameters were estimated by fitting the squared amplitude in eq. (2) to the measured amplitude responses. The estimation procedure followed was similar to the one described by Rummler in Ref. 3. From eq. (2) it is clear that

$$Y(\omega) = \alpha - \beta \cos(\omega - \omega_0)\tau, \tag{4}$$

where

$$\alpha = a^2(1 + b^2) \tag{5a}$$

$$\beta = 2a^2b. \tag{5b}$$

For convenience, the channel scan is measured in frequency increments of 2 MHz, resulting in 16 samples over the 30-MHz range

$$\omega_n = 2\pi f_n = 2\pi n(2 \cdot 10^6); \qquad n = 1, 16. \tag{6}$$

If we choose

$$\tau = \frac{1}{M(2 \cdot 10^6)}, \tag{7}$$

then the phase increments are

$$\omega_n \tau = 2\pi \frac{n}{M}. \tag{8}$$

Choosing $M = 80$, the delay is $\tau = 6.25$ ns. Thus the in-band frequencies correspond to n values between 1 and 16, and the model transfer function is periodic with $M = 80$, which corresponds to a frequency period $1/\tau = 160$ MHz (see Fig. 4).

The first modeling step is to choose α, β and ω_0 so that the sequence of measured values $\{Y_n\}$, $n = 1, 16$, is closely matched by the sequence $\{Y(\omega_n)\}$ using eqs. (4) and (8). Estimates of α, β and ω_0 were found by minimizing the mean square error, E_{rms}, between these sequences. We define E_{rms} as

$$E_{\text{rms}} = \frac{\sum_{n=1}^{16} C_n (Y_n - Y(\omega_n))^2}{\sum C_n}, \tag{9}$$

where C_n is a weighting coefficient. Since Y_n is derived from data that was uniformly quantized on a logarithmic scale, we use a weighting that is approximately logarithmic. Hence,

$$C_n = \frac{1}{Y_n^2}. \tag{10}$$

A measure of the goodness of this fitting for a given scan is the root-mean-square value of the decibel error over the sampled frequencies, i.e.,

$$E_{\text{dB}} = \left[\frac{1}{16} \sum_{n=1}^{16} (Y_n[\text{dB}] - Y(\omega_n)[\text{dB}])^2\right]^{1/2}. \tag{11}$$

A plot of the distribution of this error for the total population is shown in Fig. 5. As seen from the plot, nearly 99.9 percent of the errors are below 2 dB.

The second step in the modeling is to find a and b given estimates for α and β. From these estimates the parameters a and b are obtained by inverting eq. (5). The set $[a, b]$ has two possible solutions $[a_1, b_1]$ and $[a_2, b_2]$

$$b_1 = \frac{\alpha}{\beta} - \left[\left(\frac{\alpha}{\beta}\right)^2 - 1\right]^{1/2}$$

$$a_1 = \left[\frac{\beta}{2b_1}\right]^{1/2}, \tag{12a}$$

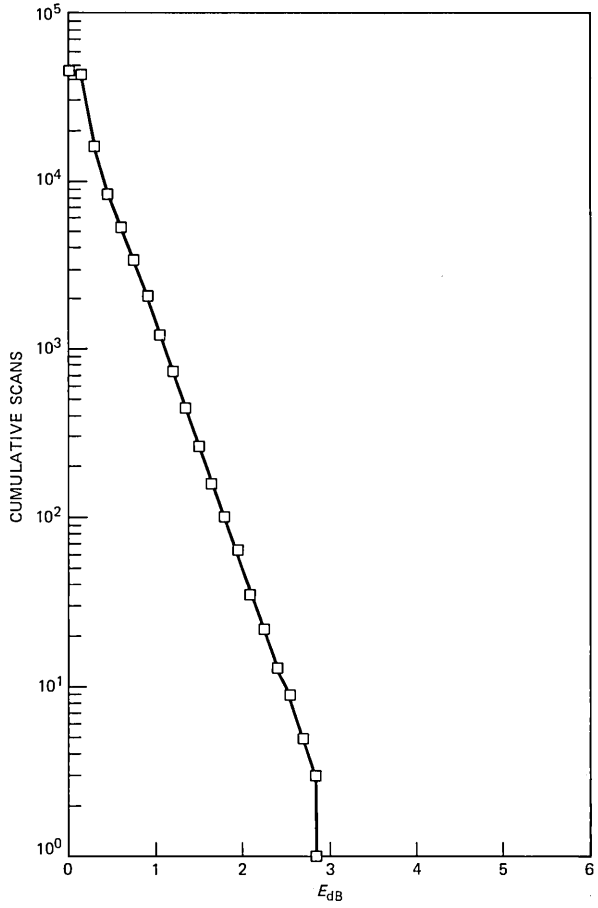


Fig. 5—Distribution of the root-mean-square error of the amplitude fit.

and

$$b_2 = \frac{\alpha}{\beta} + \left[\left(\frac{\alpha}{\beta}\right)^2 - 1\right]^{1/2}$$

$$a_2 = \left[\frac{\beta}{2b_2}\right]^{1/2}. \tag{12b}$$

It is clear from eq. (4) that $\alpha \geq \beta$ in every scan, since $Y(\omega)$ must fit a sequence of nonnegative powers. Thus, both solutions for b are real. It also follows that $b_1 b_2 = 1$.

The two solutions in eqs. (12a) and (12b) satisfy the amplitude response fit. The solution $[a_1, b_1]$ where $b_1 < 1$ corresponds to a minimum phase transfer function, since it produces a zero in the left

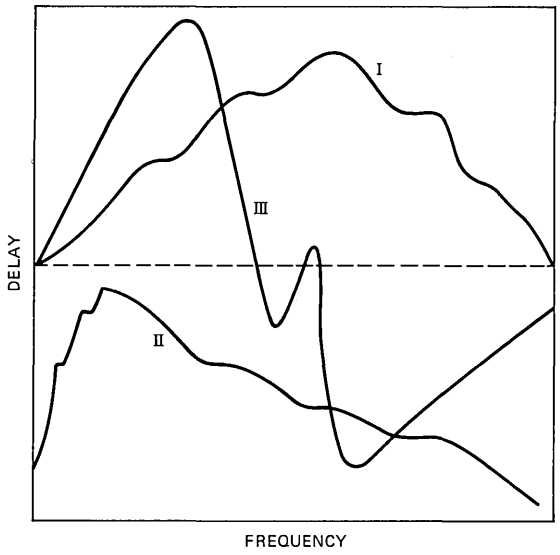


Fig. 6—Fading delay scans.

half of the complex frequency plane, and the solution $[a_2, b_2]$, where $b_2 > 1$ corresponds to a nonminimum solution, since it produces a zero in the right half of that plane.

The decision of whether a given scan corresponds to a minimum phase or nonminimum phase fade is made here by using the delay data. For a substantial fraction of the available data, the delay response was flat. For this population, the estimations were done by using amplitude data only and assuming the minimum phase solution.

2.5 Fitting the model to the delay data

2.5.1 Typical delay data

Figure 6 shows a few typical delay scans. To determine whether a scan is minimum phase or nonminimum phase, we have to examine the shape of the delay response. Scan I shows clearly a nonminimum phase delay shape, but so does scan II, although it is negative. The group delay scans usually exhibit a constant delay offset that has to be subtracted during the fitting process. Scan III is caught in the transition between minimum phase and nonminimum phase; this sometimes happens at very deep fades.

The usual time trajectory of a deep fade is as follows: The fade appears as a minimum phase fade, gradually deepens, crosses to a nonminimum phase fade, then becomes shallower, and disappears. It usually does not cross back to the minimum phase shape.

2.5.2 The fitting procedure

Each delay scan is sandwiched between two amplitude scans that are separated by 0.2 second. The initial parameters ω_{0i}, b_{1i} and b_{2i} used to fit each delay scan [eq. (3)] were interpolated from the two amplitude scans. Two error estimates were then set up, namely,

$$E_1 = \frac{1}{16} \sum_{n=1}^{16} [D_n(b_1, \omega_0) - (\hat{D}_n - D_0)]^2, \tag{13}$$

and

$$E_2 = \frac{1}{16} \sum_{n=1}^{16} [D_n(b_2, \omega_0) - (\hat{D}_n - D_0)]^2,$$

where $D_n(b_1, \omega_0)$ is the minimum phase delay response, and $D_n(b_2, \omega_0)$ the nonminimum phase delay response, as derived by fitting the amplitude data. Further, $\hat{D}_n$ is the measured group delay at frequency ω_n. The group delay scans usually exhibit an unknown constant delay offset D_0 (see Fig. 6, scan II) that has to be estimated during the fitting process.

The above errors were minimized using a quasi-Newton optimization algorithm. First, E_1 and E_2 were minimized with respect to the delay offset D_0. By selecting the response with the better fit, this step was sufficient to decide if the shape is minimum phase or nonminimum phase. The delay responses thus obtained usually fit within a 7-ns root-mean-square error, for scans having swings smaller than 50 ns. The delay response fits were further optimized, by adjusting the parameter b, to yield root-mean-square errors smaller than 4 ns. These further optimization did not change the values of b significantly. The E_{dB} values [eq. (11)], of the resulting amplitude fits changed by less than 1 dB.

Delay scans with swings larger than 50 ns were out of scale and could not be fitted.* For these scans, the choice between minimum phase and nonminimum phase was made by minimizing the error with respect to the offset D_0 only. The initial parameters, interpolated from the adjacent amplitude scans, were used in the statistics.

One type of poor fit occurred when a delay scan was in a transition from minimum phase to nonminimum phase. These scans were eliminated from the modeling.

The delay data could not be fitted more tightly to the model because, as seen from Fig. 7, the residuals have strong harmonic components. The harmonic components indicate long delay echos in the system. We have concluded from our experiments that this distortion is caused

* The instrumentation was changed recently to record delays in the range of ±100 ns.

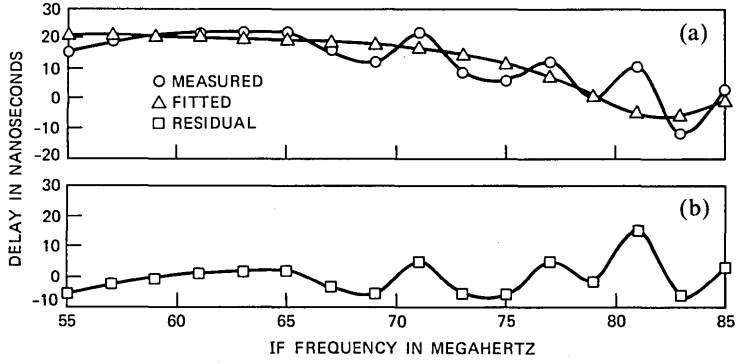


Fig. 7—(a) Measured and fitted delay curves. (b) Delay residual.

by multimoding in the antenna and waveguide system of the transmitter and receiver.

III. STATISTICAL DISTRIBUTIONS OF MODEL PARAMETERS

The statistical distributions for the parameters of the fading model transfer function in eq. (1) were characterized previously using amplitude response data from the Palmetto experiment.[3,4] We used a similar approach to generate parameter distributions for the present case. Essentially, the data scans can be divided into four distinct groups: Minimum-phase fades, nonminimum phase fades, flat amplitude fades and flat delay fades. These four groups are characterized best if the estimated statistical distributions of the model parameters are described separately. The flat amplitude and flat delay groups are degenerate cases, where only limited modeling was required. A scan with a flat amplitude response has also a flat delay response. However, a flat delay scan does not necessarily correspond to flat amplitude response. Since the delay notch is narrower than the amplitude notch (see Fig. 4), many scans with a sloping amplitude response, corresponding to an out-of-band notch, have flat delays (see Section 3.5).

The statistical characterizations of the minimum phase fades and nonminimum phase fades are done first, followed by the characterizations of the flat amplitude and flat delay fades. The statistical distributions of the model parameters for the whole fading population are estimated next, based on amplitude data only. This is done in order to compare the fading characteristics of the Gainesville channel with those of the Palmetto channel.

3.1 Minimum phase fades

For minimum phase fades the distributions are generated for the parameters

ω_0

$$A_1 = -20 \log a_1$$

$$B_1 = -20 \log(1 - b_1), \tag{14}$$

where a_1 and b_1 are the minimum phase solutions of eq. (12a), ω_0 is the notch frequency, B_1 is the relative notch depth in decibels and A_1 is the amplitude scale factor in decibels.

3.2 Nonminimum phase fades

For nonminimum phase fades, the parameter b_2 is unbounded and is not easily characterized. Equation (1) is then rewritten as

$$H(\omega) = a_2 b_2 (e^{-j(\omega-\omega_0)\tau} - 1/b_2), \tag{15}$$

and we define

$$B_2 = -20 \log(1 - 1/b_2)$$

$$A_2 = -20 \log(a_2 b_2). \tag{16}$$

As shown below, the choice of these parameters leads to distributions very similar to those of their minimum phase counterparts.

3.3 Distributions for minimum phase and nonminimum phase fades

3.3.1 The distributions of B_1 and B_2

The cumulative distributions of B_1 and B_2 are shown in Fig. 8. The abscissa is B_1 or B_2 in decibels, and the ordinate shows the time T that B_1 or B_2 is greater than the abscissa. It is clearly seen that, if we ignore the fades with low dispersion ($B \leq 8$ dB) and assume that the data above 30 dB have too few samples to be reliable, the two curves can be fitted to within ±1.75 dB by one exponential distribution. This distribution is

$$T = 1.37 \cdot 10^4 \, e^{-\frac{B}{4.82}} = 1.37 \cdot 10^4 \, L^{1.8}, \qquad \text{for} \quad 8 \leq B \leq 30 \text{ dB}, \tag{17}$$

where $L = 10^{\frac{-B}{20}}$, and represents the minimum fading voltage relative to the amplitude scale factor a_1 (or $a_2 b_2$).

As seen from Fig. 8, there are overall more minimum phase than nonminimum phase fades (58 percent). However, if we restrict ourselves to the practically significant range of $8 \leq B \leq 30$ dB, the proportion of nonminimum phase fades is 50 percent. Both the minimum and nonminimum phase fades distributions can be expressed, within the limits of the approximation, in eq. (17).

3.3.2 The distribution of the parameters A_1 and A_2

The cumulative time distributions for A_1 and A_2 are shown in Fig.

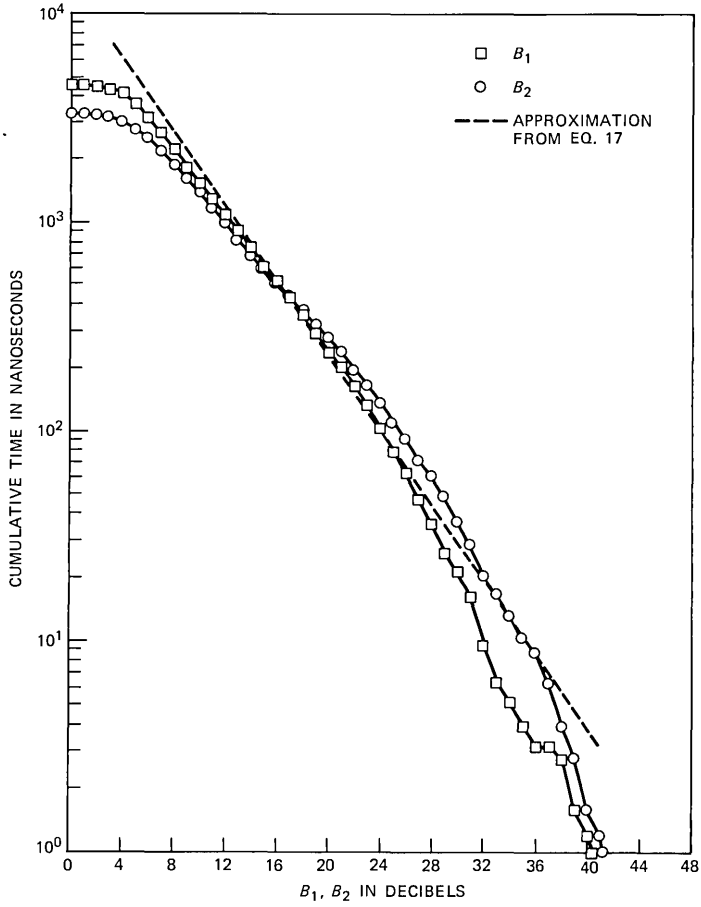


Fig. 8—Distribution of parameters B_1 and B_2.

9. It is clearly seen that the two distributions are almost identical for larger A (above 14 dB). The means and standard deviations are

$$\bar{A}_1 = 15 \text{ dB}, \qquad \sigma_1 = 3.0 \text{ dB}$$
$$\bar{A}_2 = 15.5 \text{ dB}, \qquad \sigma_2 = 3.0 \text{ dB}. \tag{18}$$

The unnormalized probability density functions (pdf's) of A_1 and A_2 are shown in Fig. 10. (These pdf's are scaled in such a way that the ordinate displays on a logarithmic scale the number of seconds that A was within ±0.5 dB of the abscissa. This same format is used in all the pdf's shown later.) The pdf's show a definite asymmetry about the mean. This deviation from a Gaussian distribution* could be

* A was Gaussian for the Palmetto data.

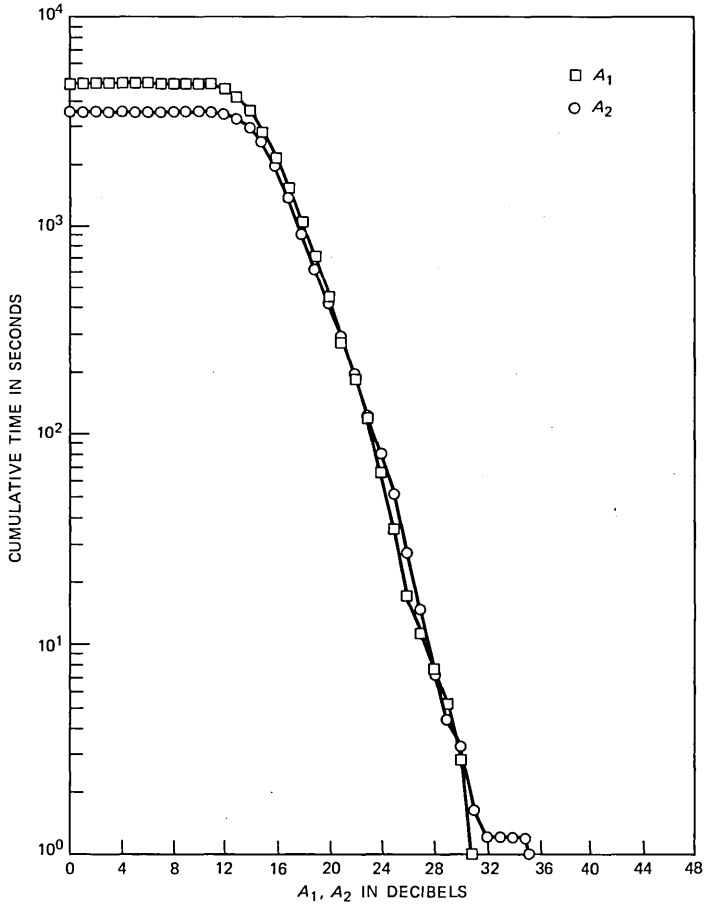


Fig. 9—Distribution of parameters A_1 and A_2.

partially attributed to the data gathering method used in Gainesville (see Section 2.2).

The means and standard deviations of A_1 and A_2 as functions of B_1 and B_2 are shown in Fig. 11. These plots show clearly that the parameters A and B are uncorrelated for the Gainesville data.*

3.3.3 Possible discrepancies in the A and B distributions caused by instrumentation

1. The amplitude and delay characteristics were sampled every 2 MHz by the recording equipment over the 30-MHz frequency scan.

* A and B were correlated for the Palmetto data.

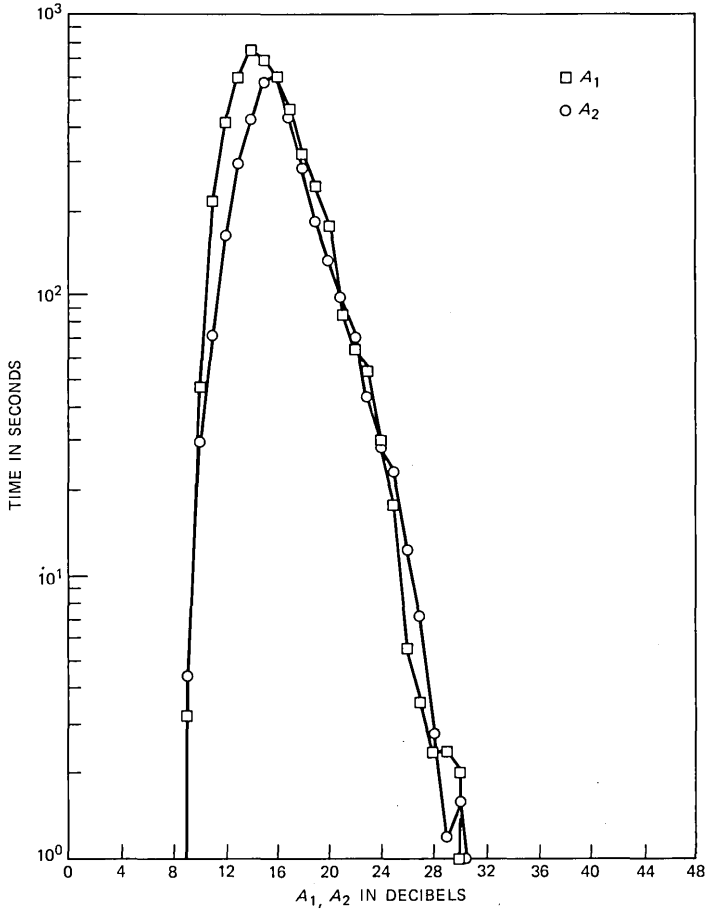


Fig. 10—Unnormalized probability density functions of A_1 and A_2.

The 2-MHz resolution may be too low for very deep dispersive fades, and cause flattening of these curves during the fitting process. This may cause the distributions of B_1 and B_2 to appear steeper for very large B values than they really are.

2. The Microwave Link Analyzer (MLA) was set up to measure amplitudes between 15 to 65 dB only. Therefore, some of the shallow fading events with small A and small B are not recorded. This could cause the distribution of B for small values to appear flatter than they really are, and the distributions of A to appear less symmetric.

3.3.4 The distribution of the notch frequency ω_0

Figure 12 shows the density function of the notch frequency, for both the minimum and nonminimum phase fades, as estimated by the

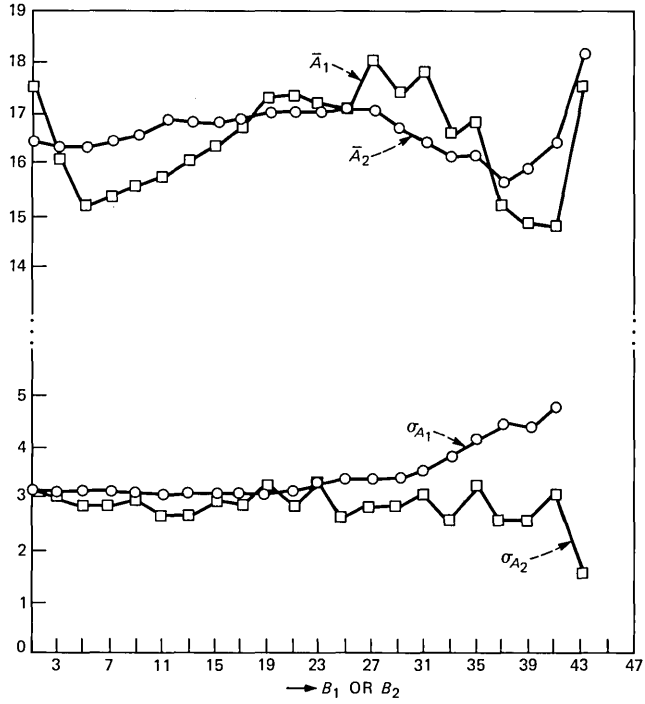


Fig. 11—Mean and standard deviation of $A_1(A_2)$ as functions of $B_1(B_2)$.

modeling procedure. This model estimates that approximately 70 percent of the scans have an in-band notch. This is to be expected because, as seen from Fig. 4, the delay shape is steeper and narrower than the amplitude notch. Therefore, most of the scans that have an in-band slope (out-of-band notch) have a flat delay response, and are therefore not characterized in this group (see Section 2.2).

Fitting the model to amplitude shapes that have in-band slopes is not unique, since small perturbations in data can produce large variations of out-of-band notch depth and frequencies. The modeling algorithm in its present form has a tendency to move the notch for slope shapes close to the edges of the band. We have tried, with variable success, to use also the delay data in placing the out-of-band notches. However, the delay data for shapes with out-of-band notches are limited and heavily corrupted with multimode distortion that made the fitting of the model rather difficult.

3.4 Statistical distribution of flat-amplitude fades

Amplitude scans that were flat within 1 dB were assumed to be nondispersive. For such scans, therefore, A was the only parameter

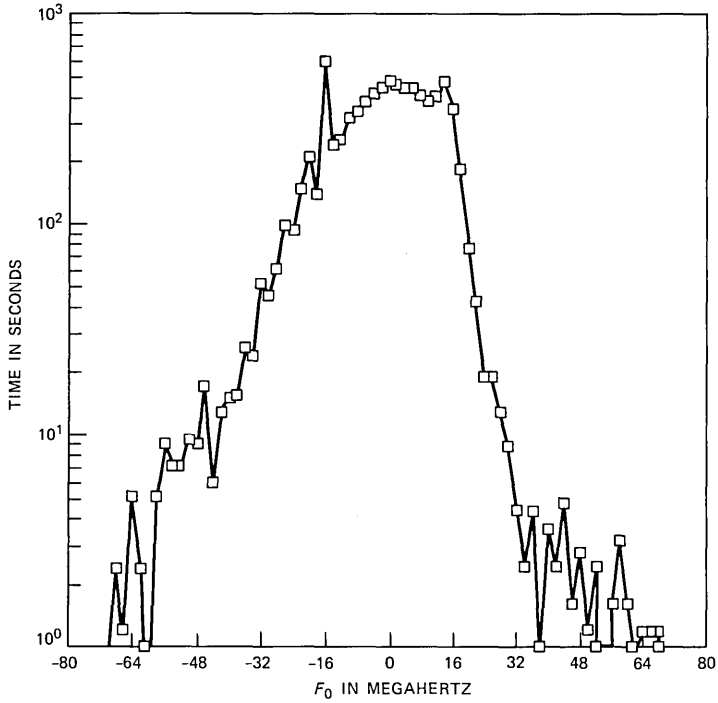


Fig. 12—Unnormalized probability density function of the notch frequency F_0, measured from midband.

modeled. The cumulative distribution of A for this case is shown in Fig. 13. As in previous cases, the distribution exhibits a truncated Gaussian behavior, with

$$\bar{A} = 15.9 \text{ dB}, \qquad \sigma = 2.85 \text{ dB}. \tag{19}$$

3.5 Statistical distribution of flat-delay fades

Scans that did not have a flat amplitude but exhibit a small delay distortion, $D(\omega)_{\text{peak to peak}} \leq 7$ ns, were regarded as having a flat delay. Scans of this type can be put into two categories:

(a) Scans with in-band, or close to in-band, notches and small amplitude deviations. A 7-ns delay notch corresponds to an amplitude notch with $B = 6.5$ dB.

(b) Scans with amplitude slopes that may have a notch so far out of band that the delay distortion is negligible.

In both of the above cases the delay information is too limited to be used to decide whether the scan is minimum phase or nonminimum phase. Since it is relatively unimportant to differentiate, in these

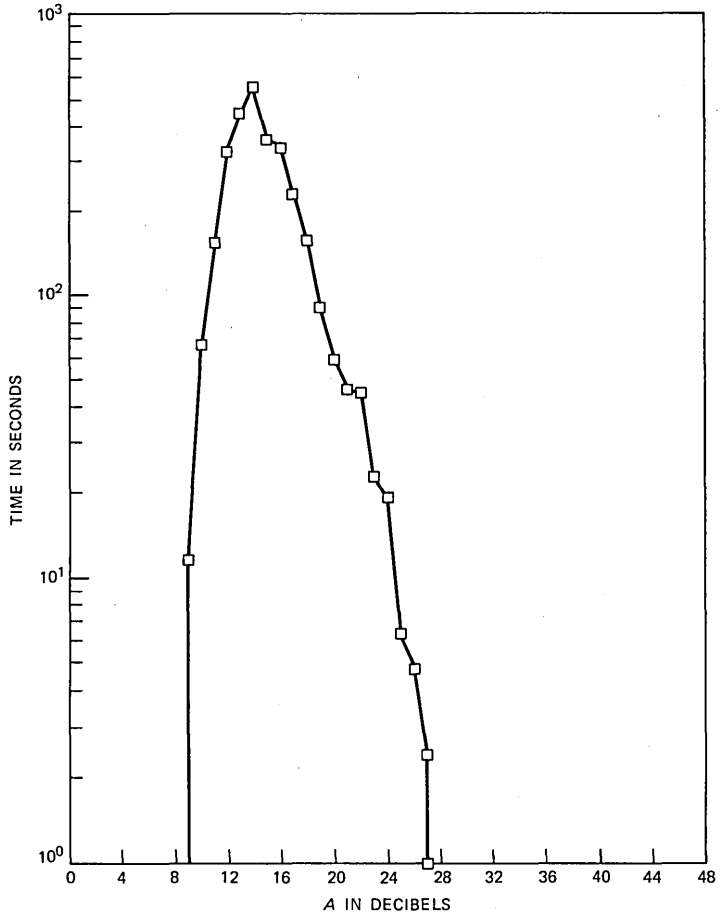


Fig. 13—Unnormalized probability density function of A for flat amplitude fades.

cases, between minimum phase and nonminimum phase fades, the statistics described below were compiled as though all such fades were minimum phase.

Figure 14 shows the distribution of B. As seen from the plot, 96 percent of the scans have $B \leq 6.5$ dB, which puts them in the category (a) above. Only 4 percent belong to category (b).

Figure 15 shows the probability density of A, wherein

$$\bar{A} = 15.8 \text{ dB}, \qquad \sigma = 2.6 \text{ dB}. \tag{20}$$

3.6 Parameter distributions for the total fade population and comparison with Palmetto

The distributions of the model parameters for the whole fade population, derived entirely from amplitude response data and comprising

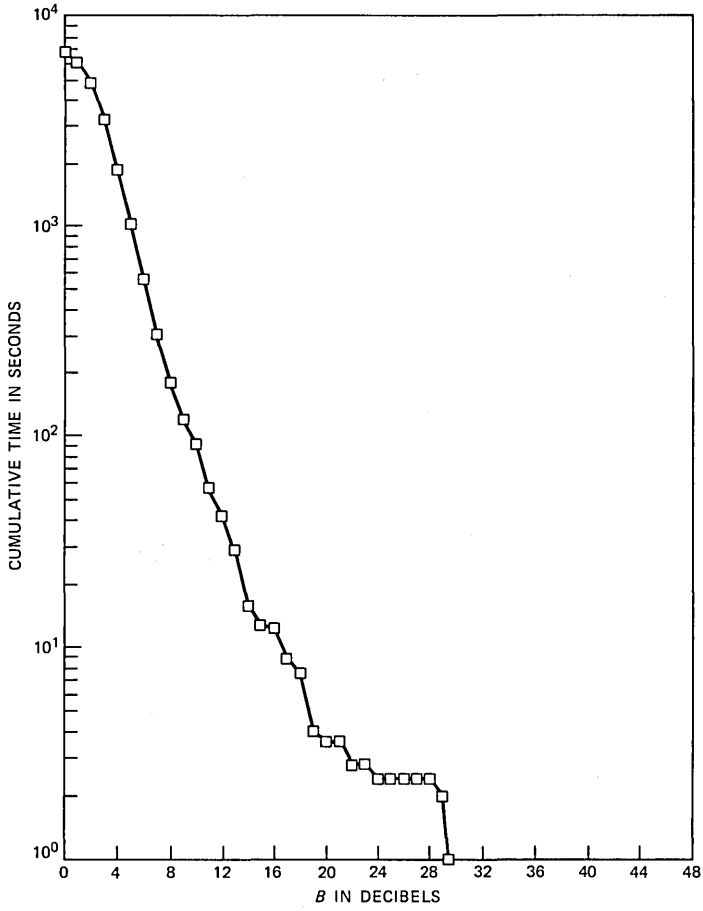


Fig. 14—Distribution of B for flat delay fades.

the four groups characterized in previous sections, are described below. By ignoring the delay characteristics, we computed the model parameters as if all fades were minimum phase. This was done in order to have a basis for comparison with the results obtained in the Palmetto experiment.

3.6.1 B distributions

Figure 16 shows the distribution of the parameter B, as measured in Gainesville. Within reasonable accuracy the distribution can be modeled by the exponential curve

$$T = 2.37 \cdot 10^4 \, e^{-\frac{B}{5}} = 2.37 \cdot 10^4 \, L^{1.72}. \tag{21}$$

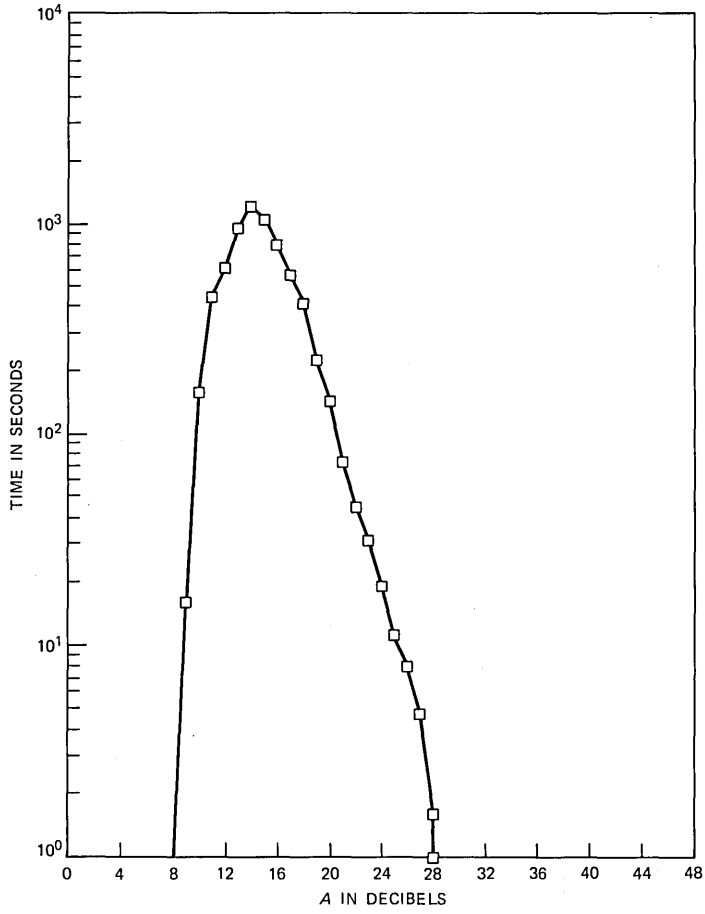


Fig. 15—Unnormalized probability density function of A for flat delay fades.

The fading data were recorded for 11 months, well distributed over all seasons during 1.5 years in 1982–1984. The curve in Fig. 16 can therefore be regarded as an annual average distribution of dispersive fading in Gainesville. The other curve in Fig. 16 shows the annual distribution of B for the Palmetto data. This distribution was extrapolated from data measured in Palmetto in 1977.[4] The total fading for the heaviest month of the year was estimated to be 8000 seconds, where the total fading was defined as the intercept of the B distribution with the $B = 0$ axis. To approximate the average annual fading in Palmetto, we multiplied the seconds for the heaviest fading month by a scale factor of 3.5. The average annual fading in Palmetto was thus estimated to be 28,000 seconds.

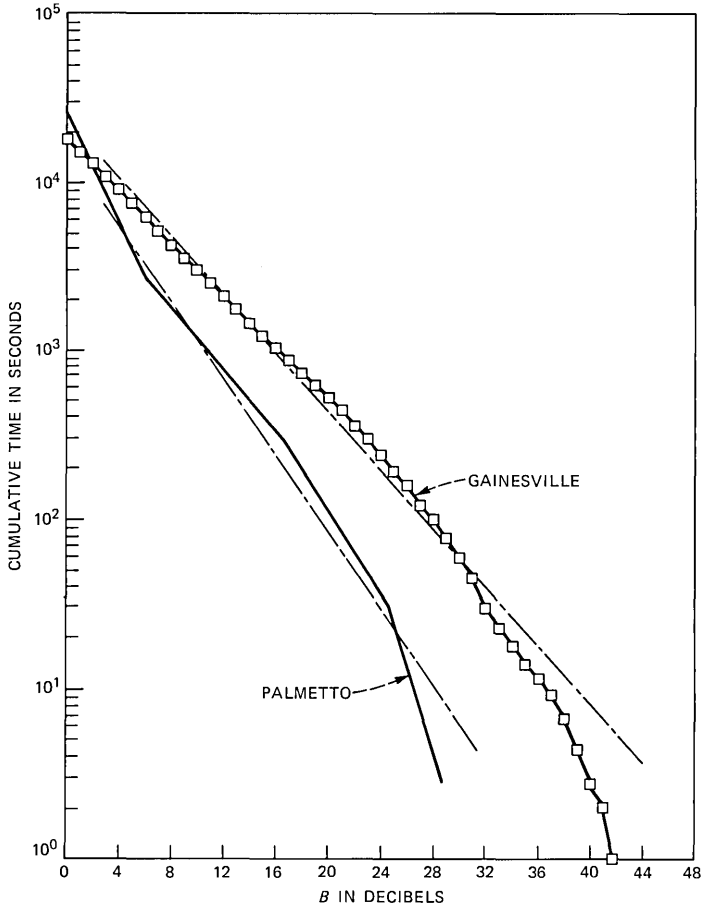


Fig. 16—Average annual probability distributions for B for the total populations of Gainesville and Palmetto.

The Palmetto distribution is steeper than that for Gainesville and can be modeled over its central region by

$$T = 1.6 \cdot 10^4 \, e^{-\frac{B}{3.8}} = 2 \cdot 10^4 \, L^{2.28}, \qquad 4 \text{ dB} \leq B \leq 25 \text{ dB}. \quad (22)$$

It can be surmised from the shape of the Gainesville distribution that fading on that path is more dispersive than the Palmetto fading. By examining, for illustration purposes, the total dispersive fading for $B \geq 20$ dB, we see from Fig. 16 that there is five times more of it in Gainesville than in Palmetto.

3.6.2 A distributions

Figure 17 shows the probability density function of parameter A. It has a truncated Gaussian shape, with mean and standard deviation

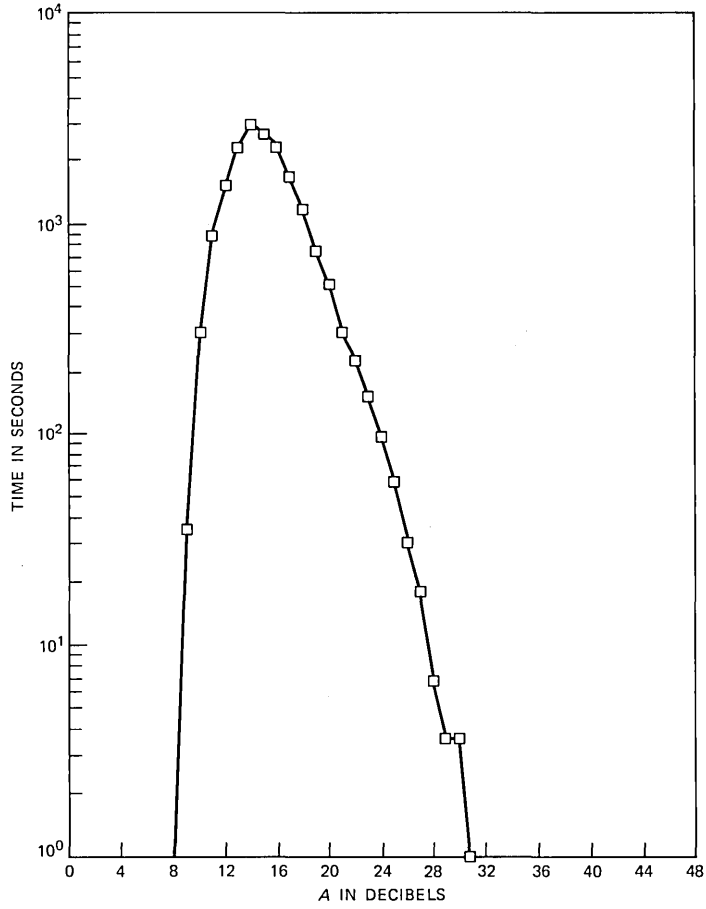


Fig. 17—Unnormalized probability density function of A for the total population.

$$\bar{A}_G = 16.2 \text{ dB}; \qquad \sigma_G = 2.85 \text{ dB}. \tag{23}$$

The corresponding distribution in Palmetto was Gaussian, with mean and standard deviation

$$\bar{A}_P = 25 \text{ dB}; \qquad \sigma_P = 5 \text{ dB}. \tag{24}$$

Moreover, the mean $\bar{A}_P$ was correlated with B_P in the Palmetto experiment. The Gainesville parameters $\bar{A}_G$ and B_G are independent (Fig. 18).

The apparent truncation of the A distribution is probably due to the data gathering method in Gainesville. Specifically, fades below 15 dB were not recorded. This could also be a reason why A and B are uncorrelated in Gainesville. In Palmetto, $\bar{A}$ was correlated with B, for $B < 10$ dB only.

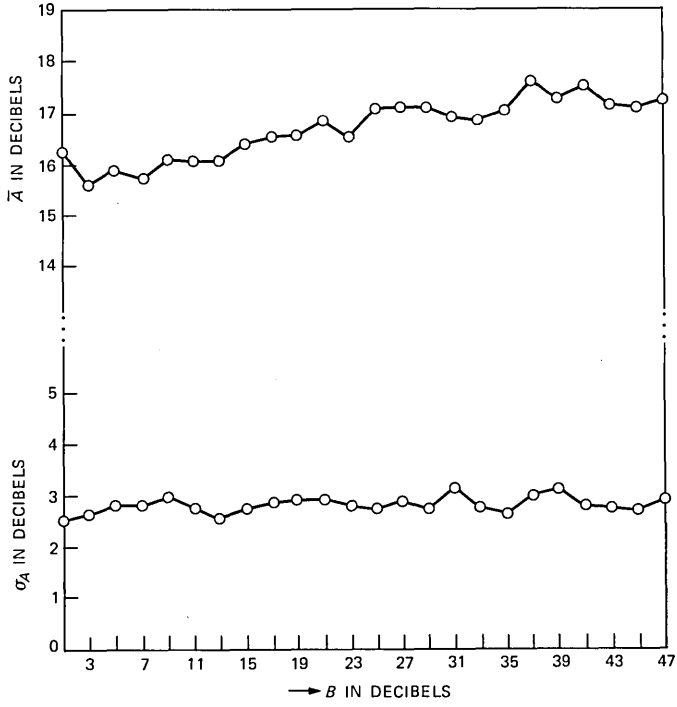


Fig. 18—Mean and standard deviation of A as functions of B.

3.6.3 Distribution of notch frequency ω_0

The probability density function of the notch frequency for the total fade population is shown in Fig. 19. Most of the scans have an in-band notch (55 percent). The notch frequency does not exhibit the bimodal distribution shown in Palmetto.

IV. CONCLUSION

4.1 Summary

The fading data obtained in the Gainesville experiment were for both amplitude and group delay. We generalized the method of Rummler[3,4] to include group delay response data, and thus obtain both minimum phase and nonminimum phase model parameters.

The distributions for minimum phase and nonminimum phase fades can be regarded as identical, and can be given by a single description [see eqs. (17) and (18)], in the significant range of interest, $8 \leq B \leq 30$ dB. However, if the shallower fades are included ($0 \leq B \leq 8$ dB), there are, overall, more minimum phase fades (58 percent).

The Gainesville channel is more dispersive than the Palmetto chan-

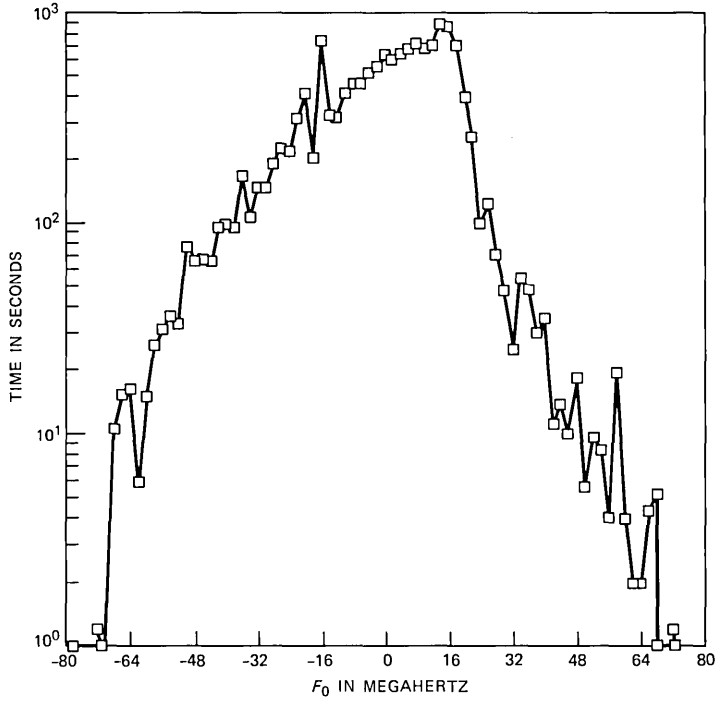


Fig. 19—Unnormalized probability density function of the notch frequency F_0 for the total population, measured from midband.

nel. This is seen from the estimated average annual distributions for the total fade populations. For the dispersive part of the fading, the B parameter, the distribution is steeper in Palmetto. By examining Fig. 16 for illustration purposes, we can deduce that there is five times more fading with $B = 20$ dB in Gainesville. In addition, the mean of A is smaller in Gainesville; $\bar{A}_G \approx 16$ dB, as compared with $\bar{A}_P \approx 26$ dB in Palmetto.

4.2 Implications in terms of performance objectives

AT&T long-haul digital-radio performance objectives limit service failure time to 0.02 percent (two way) annually on a 4000-mile route due to all causes. One-half of this is allocated to causes associated with equipment and maintenance. The allocation to dispersive fading, therefore, is 0.01 percent annually.[14] The one-way annual dispersive fading allocation for the average 25-mile hop is thus 10 seconds.

One measure of the amount of dispersive fading in a radio channel is the distribution of the parameter B. As seen from Fig. 16, a 10-second annual failure limit in Gainesville requires a robust digital radio system designed to operate with notches $B \leq 36$ dB. The same

failure limit in Palmetto requires a radio system designed to operate with notches of $B \leq 26$ dB. These comparisons demonstrate the need for a description of the geographical occurrence of dispersion, which will differ from that for multipath fading for a single frequency. The availability of such a dispersive fading map would facilitate the accurate engineering of digital radio routes.

V. ACKNOWLEDGMENTS

I would like to acknowledge the contributions of G. A. Axeling for aligning and maintaining the measurement equipment in Gainesville, B. J. Engelmann for utilizing her computer expertise to produce the statistical outputs, and G. L. Calhoun for maintaining the database. For many stimulating discussions on this work and comments on this manuscript, I would like to thank W. T. Barnett, L. J. Greenstein, V. K. Prabhu, W. D. Rummler, and A. Vigants.

REFERENCES

1. G. M. Babler, "A Study of Frequency Selective Fading for a Microwave Line-of-Sight Narrowband Radio Channel," B.S.T.J., *51,* No. 3 (March 1972), pp. 731–57.
2. L. J. Greenstein, "A Multipath Fading Channel Model for Terrestrial Digital Radio," IEEE Trans. Commun., *COM-26,* No 8 (August 1978), pp. 1247–50.
3. W. D. Rummler, "A New Selective Fading Model: Application to Propagation Data," B.S.T.J., *58,* No. 5 (May–June 1979), pp. 1037–71.
4. W. D. Rummler, "More on the Multipath Fading Channel Model," IEEE Trans. Commun., *COM-29,* No. 3 (March 1981), pp. 346–52.
5. L. J. Greenstein and B. A. Czekaj, "A Polynomial Model for Multipath Fading Channel Responses," B.S.T.J., *59,* No. 7 (September 1980), pp. 1197–225.
6. M. H. Meyers, "Multipath Fading Characteristics of Broadband Radio Channels," Globecom'84 Conf. Rec., *3* (November 1984), pp. 1460–5.
7. W. C. Jakes, Jr., "An Approximate Method to Estimate an Upper Bound on the Effect of Multipath Delay Distortion on Digital Transmission," IEEE Trans. Commun., *COM-27,* No. 1 (January 1979), pp. 76–81.
8. L. J. Greenstein and V. K. Prabhu, "Analysis of Multipath Outage With Applications to 90 Mbit/s PSK Systems at 6 and 11 GHz," IEEE Trans. Commun., *COM-27,* No. 1 (January 1979), pp. 68–75.
9. M. F. Gardina and A. Vigants, "Measured Multipath Dispersion of Amplitude and Delay at 6 GHz in a 30 MHz Bandwidth," ICC'84 Conf. Rec., *3* (May 1984), pp. 1433–6.
10. M. Liniger, "Sweep Measurements of Multipath Effects on Cross-Polarized RF-Channels Including Space Diversity," Globecom'84 Conf. Rec., *3* (November 1984), pp. 1492–6.
11. L. Martin, "Phase Distortions of Multipath Transfer Functions," ICC'84 Conf. Rec., *3* (May 1984), pp. 1437–41.
12. M. Sylvain and J. Lavergnat, "Modelling the Transfer Function of a 55 MHz Wide Radio Channel Multipath Propagation," ICC'85 Conf. Rec., June 1985, *3,* pp. 1541–6.
13. G. A. Axeling, private communication.
14. A. Vigants, "Space Diversity Engineering," B.S.T.J., *54,* No. 1 (January 1975), pp. 103–42.

AUTHOR

Philip Balaban, Diplom Ingenieur (Electrical Engineering), 1950, Technical University Munich; Ph.D. (Electrical Engineering), 1965, Polytechnic Insti-

tute of Brooklyn; Scientific Department, Israel, 1950–1956; Contraves, Zurich, Switzerland, 1956–1958; Computer Systems, New York, 1958–1962; Polytechnic Institute of Brooklyn, 1962–1967; AT&T Bell Laboratories, 1967—. Prior to joining AT&T Bell Laboratories, Mr. Balaban was engaged in research and development in the areas of circuit design, computers, and communications. At AT&T Bell Laboratories he worked on statistical modeling, simulation and analysis of transmission systems and circuits. Recently he has been involved with problems of channel characterization, and system performance of digital radio communication systems. In 1965–67 he taught at the Polytechnic Institute of Brooklyn, and in 1970–71 he was a visiting professor at the Technion in Israel. Senior Member, IEEE; Member, Sigma Xi.

LETTER TO THE EDITOR

Comments on "Integration With the *5ESS*™ Switching System," by S. A. McRoy, J. H. Miller, J. B. Truesdale, and R. W. Van Slooten*

This letter amplifies the footnote on page 2426 of the article titled, "Integration With the *5ESS*™ Switching System," in the December 1984, issue of the *AT&T Bell Laboratories Technical Journal.*

The information given here has been used in the field for engineering Remote Terminals (RTs) of the *SLC*® 96 digital subscriber loop carrier system on the Digital Carrier Line Unit (DCLU) in a *5ESS* switching office. For engineering RTs, the telephone company engineer can choose one of the three service criteria—1-1/2 percent blocking, 4 percent blocking, and 7 percent blocking. The 1-1/2 percent blocking criteria is recommended for average busy season busy hour engineering, the 4 percent blocking Once-A-Month (OAM) for extreme value engineering, and the 7 percent blocking for high-day engineering. The DCLU traffic capacities for the two types (Mode I and II) of RTs corresponding to the three criteria are given in Table I.

Table I—DCLU CCS capacities per line

No. of Time Slots	Mode I			
	RTs	1-1/2%	4%	7%
64[1]	6	3.19	3.41	3.53
64	5	3.83	4.08	4.24
64	4	4.79	5.10	5.30
128[2]	6	6.93	7.25	7.44
128	5	8.31	8.69	8.92
128	4	10.36	10.85	11.14
128	3	13.78	14.43	14.81
128	2	20.56	21.54	22.12
No. of Time Slots	Mode II			
	RTs	1-1/2%	4%	7%
64	6	3.06	3.27	3.40
64	5	3.65	3.90	4.09
64	4	4.52	4.83	5.04
128	6	6.61	6.93	7.13
128	5	7.81	8.21	8.45
128	4	9.51	10.00	10.32
128	3	10.02	11.07	11.72
128	2	11.85	13.20	14.10

1. A two-peripheral-interface-data-bus DCLU has 64 time slots.
2. A four-peripheral-interface-data-bus DCLU has 128 time slots.

* AT&T Bell Lab. Tech. J., *63*, No. 10, Pt. 2 (December 1984), pp. 2417–37.

The capacities in the table are calculated by the same approach described in the article. However, the 3 percent OAM blocking criterion referenced in the article is not used for engineering *5ESS* switching systems in the field; 4 percent OAM is used. Also, the peak factor formula discussed in Section 5.4 of the article is not used in engineering. The information in the two tables documented in this letter has been used for engineering of *5ESS* switching systems.

K. A. Raschke and P. A. Patankar

PAPERS BY AT&T BELL LABORATORIES AUTHORS

COMPUTING/MATHEMATICS

Bentley J., **Associative Arrays.** Comm ACM 28(6):570–576, Jun 1985.

Bronson G., Lyon K., **Twos Complement Numbers Revisited—A New Tool for Dealing With the Fundamentals of Number Storage.** Byte 10(6):230–231, Jun 1985.

Chung F. R. K., Fishburn P. C., Wei V. K., **Cross-Monotone Subsequences.** Order 1(4):351–369, 1985.

Chung F. R. K., Tarjan R. E., Paul W. J., Reischuk R., **Coding Strings by Pairs of Strings.** SIAM J Alg 6(3):445–461, Jul 1985.

Coffman E. G., Gilbert E. N., **On the Expected Relative Performance of List Scheduling.** Operat Res 33(3):548–561, May–Jun 1985.

Ehrenreich S. L., **Computer Abbreviations—Evidence and Synthesis.** Human Fact 27(2):143–155, Apr 1985.

Fishburn P. C., Trotter W. T., **Angle Orders.** Order 1(4):333–343, 1985.

Johnson C. R., **Convergence of a Nonlinear Sharpening Transformation for Digital Images.** SIAM J Alg 6(3):462–465, Jul 1985.

Johnson D. S., **The NP-Completeness Column—An Ongoing Guide.** J Algorithm 6(2):291–305, Jun 1985.

Kapadia A. S., Chiang Y. K., Kazmi M. F., **Finite-Capacity Priority Queues With Potential Health Applications.** Comput Oper 12(4):411–420, 1985.

Klauder J. R., Petersen W. P., **Spectrum of Certain Non-Self-Adjoint Operators and Solutions of Langevin-Equations With Complex Drift.** J Stat Phys 39(1–2):53–72, Apr 1985.

Kumar A., Fine T. L., **Stationary Lower Probabilities and Unstable Averages.** Z Wahrsch V 69(1):1–17, 1985.

Lagarias J. C., **The Set of Primes Dividing the Lucas Numbers Has Density 2/3.** Pac J Math 118(2):449–461, Jun 1985.

Massey W. A., **Asymptotic Analysis of the Time-Dependent M/M/1 Queue.** Math Oper R 10(2):305–327, May 1985.

Oddson J. K., Aggarwal S., **Discrete-Event Simulation of Agricultural Pest Management Systems.** Simulation 44(6):285–293, Jun 1985.

Odlyzko A. M., Riele H. J. J. T., **Disproof of the Mertens Conjecture.** J Rein Math 357:138–160, 1985.

Padmanabhan K., Lawrie D. H., **Performance Analysis of Redundant-Path Networks for Multiprocessor Systems.** ACM T Comp 3(2):117–144, May 1985.

Sandberg I. W., **Iteration and Functional Expansions.** Circ Syst S 3(4):409–417, 1984.

Yannakakis M., **On a Class of Totally Unimodular Matrices.** Math Oper R 10(2):280–304, May 1985.

ENGINEERING

Abramovici M., Menon P. R., **A Practical Approach to Fault Simulation and Test-Generation for Bridging Faults (Letter).** IEEE Comput 34(7):658–663, Jul 1985.

Alferness R. C., Veselka J. J., **Simultaneous Modulation and Wavelength Multiplexing With a Tunable Ti-$LiNbO_3$ Directional Coupler Filter.** Electr Lett 21(11):466–467, May 23 1985.

Antler M., **Survey of Contact Fretting in Electrical Connectors.** IEEE Compon 8(1):87–104, Mar 1985.

Antreasyan A., Chen C. Y., Logan R. A., **Stop-Cleaved InGaAsP Lasers for Monolithic Optoelectronic Integration.** Appl Phys L 46(10):921–923, May 15 1985.

Banu M., Tsividis Y., **On-Chip Automatic Tuning for a CMOS Continuous-Time Filter.** ISSCC Diges 28:286–287, 1985.

Barber F. E., Eisenberg D. J., Ingram G. A., Strauss M. S., Wik T. R., **A 2K × 9 Dual Port Memory.** ISSCC Diges 28:44–45, 1985.

Bishop D. J., Licini J. C., Dolan G. J., **Lithium Quench-Condensed Microstructures and the Aharonov-Bohm Effect.** Appl Phys L 46(10):1000–1002, May 15 1985.

Bishop D. J., Spencer E. G., Dynes R. C., **The Metal-Insulator Transition in Amorphous Nb-Si.** Sol St Elec 28(1–2):73–79, Jan–Feb 1985.

Chabal Y. J., **Infrared Study of the Chemisorption of Hydrogen and Water on Vicinal Si(100) 2 × 1 Surfaces.** J Vac Sci A 3(3):1448–1451, May–Jun 1985.

Chen C. Y., Chu S. N. G., Cho A. Y., **GaAs/$Ga_{0.47}In_{0.53}As$ Lattice-Mismatched Schottky-Barrier Gates—Influence of Misfit Dislocations on Reverse Leakage Currents.** Appl Phys L 46(12):1145–1147, Jun 15 1985.

Chen C. Y., Dentai A. G., Kasper B. L., Garbinski P. A., **High-Speed Junction-Depleted $Ga_{0.47}In_{0.53}As$ Photoconductive Detectors.** Appl Phys L 46(12):1164–1166, Jun 15 1985.

Cheng J., Stall R., Forrest S. R., Long J., Cheng C. L., Guth G., Wunder R., Riggs V. G., **Self-Aligned $In_{0.53}Ga_{0.47}As$/Semi-Insulating/N+ InP Junction Field-Effect Transistors.** IEEE Elec D 6(7):384–386, Jul 1985.

Chin A. K., Chin B. H., Camlibel I., Zipfel C. L., Minneci, G., **Practical Dual-Wavelength Light-Emitting Double Diode.** J Appl Phys 57(12):5519–5522, Jun 15 1985.

Chraplyvy A. R., Stone J., **Single-Pass Mode-Locked or Q-Switched Pump Operation of D2 Gas-in-Glass Fiber Raman Lasers Operating at 1.56-μm Wavelength.** Optics Lett 10(7):344–346, Jul 1985.

Coldren L. A., Boyd G. D., Burrus C. A., **Dependence of Chirping on Cavity Separation in 2-Section Coupled-Cavity Lasers.** Electr Lett 21(12):527–528, Jun 6 1985.

Dautremont-Smith W. C., Feldman L. C., **Structural Damage Produced in InP(100) Surfaces by Plasma-Employing Deposition Techniques.** J Vac Sci A 3(3):873–878, May–Jun 1985.

Dautremont-Smith W. C., Woelfer S. M., **Stresses in the InP/Ti/Pt and InP/SiO_2/Ti/Pt Multilayer Systems.** Appl Phys L 47(1):31–33, Jul 1 1985.

Dutta N. K., Wessel T., Olsson N. A., Logan R. A., Yen R., Anthony P. J., **Fabrication and Performance Characteristics of InGaAsP Ridge-Guide Distributed-Feedback Multiquantum-Well Lasers.** Electr Lett 21(13):571–573, Jun 20 1985.

Fisanick G. J., Hopkins J. B., Gross M. E., Fennell M. D., Schnoes K. J., **Laser-Initiated Microchemistry—Dynamic Probes of Metallopolymer Thin-Film Decomposition.** Appl Phys L 46(12):1184–1186, Jun 15 1985.

Gossard A. C., **Two-Dimensional Electron-Gas Systems at Semiconductor Interfaces.** Surf Sci 152(Apr):1153–1166, Apr 1985.

Gottscho R. A., Mandich M. L., **Time-Resolved Optical Diagnostics of Radio-Frequency Plasmas.** J Vac Sci A 3(3):617–624, May–Jun 1985.

Gray E. W., **Damage of Flexible Printed Wiring Boards Associated With Lightning Induced Voltage Surges.** IEEE Compon 8(1):214–220, Mar 1985.

Haque C. A., **Temperature Time Effects on Film Growth and Contact Resistance of a Plated Copper Tin Zinc Alloy Used as a Surface Finish on Electronic Components.** IEEE Compon 8(1):153–156, Mar 1985.

Jindal R. P., **Noise Associated With Substrate Current in Fine-Line NMOS Field-Effect Transistors.** IEEE Device 32(6):1047–1052, Jun 1985.

Kastalsky A., Luryi S., Gossard A. C., Chan W. K., **Switching in NERFET Circuits.** IEEE Elec D 6(7):347–349, Jul 1985.

Kershaw R. N., Bays L. E., Freyman R. L., Klinikow J. J., Miller C. R., Mondal K., Moscovitz H. S., Stocker W. A., Tran L. V., Hays W. P., **A Programmable Digital Signal Processor With 32B Floating Point Arithmetic.** ISSCC Diges 28:92, 1985.

King W. C., Chin B. H., Camlibel I., Zipfel C. L., **High-Speed High-Power 1.3-μm InGaAsP/InP Surface-Emitting LEDs for Short-Haul Wide-Bandwidth Optical-Fiber Communications.** IEEE Elec D 6(7):335–337, Jul 1985.

Kirsch H. C. et al., **A 1Mb CMOS DRAM.** ISSCC Diges 28:256–257, 1985.

Koch T. L., Bridges T. J., Burkhardt E. G., Corvini P. J., Coldren L. A., Linke R. A., Tsang W. T., Logan R. A., Johnson L. F., Kazarinov R. F., **1.55-μm InGaAsP**

Distributed Feedback Vapor-Phase Transported Buried Heterostructure Lasers. Appl Phys L 47(1):12–14, Jul 1 1985.

Kollaritsen P. W., Weste N. H. E., **Topologizer—An Expert System Translator of Transistor Connectivity to Symbolic Cell Layout.** IEEE J Soli 20(3):799–804, Jun 1985.

Komal A., Goskel K., Diodato P. W., Fields J. A., Gumaste U. V., Kung C. K., Lin K. Y., Lega M. E., Maurer P. M., Ng T. K., **An IEEE Standard Floating Point Chip.** ISSCC Diges 28:18–19, 1985.

Komlos J., Greenberg A. G., **An Asymptotically Nonadaptive Algorithm for Conflict Resolution in Multiple-Access Channels.** IEEE Info T 31(2):302–306, Mar 1985.

Koren U., Eisenstein G., Bowers J. E., Gnauck A. H., Tien P. K., **Wide-Bandwidth Modulation of Three-Channel Buried-Crescent Laser Diodes.** Electr Lett 21(11):500–501, May 23 1985.

Kull G. M., Nagel L. W., Lee S. W., Lloyd P., Prendergast E. J., Dirks H., **A Unified Circuit Model for Bipolar Transistors Including Quasi-Saturation Effects.** IEEE Device 32(6):1103–1113, Jun 1985.

Levy R. A., Vincent S. M., McGahan T. E., **Evaluation of the Phosphorus Concentration and Its Effect on Viscous Flow and Reflow in Phosphosilicate Glass.** J Elchem So 132(6):1472–1480, Jun 1985.

Lin W., Stavola M., **Oxygen Segregation and Microscopic Inhomogeneity in Czochralski Silicon.** J Elchem So 132(6):1412–1416, Jun 1985.

Locicero J. L., Pazarci M., Rzeszewski T. S., **A Compatible High-Definition Television System (SLSC) With Chrominance and Aspect Ratio Improvements.** SMPTE J 94(5):546–558, May 1985.

Maxemchuk N. F., Netravali A. N., **Voice and Data on a CATV Network.** IEEE J Sel 3(2):300–311, Mar 1985.

Mottine J. J., Reagor B. T., **The Effect of Lubrication on Fretting Corrosion at Dissimilar Metal Interfaces in Socketed IC Device Applications.** IEEE Compon 8(1):173–181, Mar 1985.

Nordland W. A., Kazarinov R. F., Temkin H., Yen R., **Growth of InGaAsP Distributed Feedback Lasers by a Modified Single-Phase LPE Technique.** Sol St Comm 55(6):505–507, Aug 1985.

Odajima A., Wang T. T., Takase Y., **An Explanation of Switching Characteristics in Polymer Ferroelectrics by a Nucleation and Growth Theory.** Ferroelectr 62(1–2):39–46, 1985.

Paalanen M. A., Ruckenstein A. E., Thomas G. A., **Enhanced Spin-Lattice Relaxation Near the Metal-Insulator Transition.** Sol St Elec 28(1–2):121–125, Jan–Feb 1985.

Pearce C. W., Yaney D. S., **Short-Channel Effects in MOSFETS.** IEEE Elec D 6(7):326–328, Jul 1985.

Penzias A., **The Supercollider—Can Science Afford It?** Issues Sci 1(4):66–68, Sum 1985.

People R., Bean J. C., Lang D. V., **Modulation Doping in Ge(X)Si(1 – X)/Si Strained Layer Heterostructures—Effects of Alloy Layer Thickness, Doping Setback, and Cladding Layer Dopant Concentration.** J Vac Sci A 3(3):846–850, May–Jun 1985.

Schluter M., **Theory of Localized Defects in Semiconductors.** Helv Phys A 58(2–3):355–370, 1985.

Sharma R., **Architecture Design of a High-Quality Speech Synthesizer Based on the Multipulse LPC Technique.** IEEE J Sel 3(2):377–383, Mar 1985.

Stillwagon L. E., **Chromatography as a Tool in the Characterization and Quality Control of Resist Materials.** Sol St Tech 28(5):113–118, May 1985.

Stoler D., Saleh B. E. A., Teich M. C., **Binomial States of the Quantized Radiation Field.** Optica ACTA 32(3):345–355, Mar 1985.

Stone J., **Optical-Fibre Fabry-Perot-Interferometer With Finesse of 300.** Electr Lett 21(11):504–505, May 23 1985.

Stone J., Chraplyvy A. R., Kasper B. L., **Long-Range 1.5 μm OTDR in a Single-Mode Fiber Using a D2 Gas-in-Glass Laser (100 Km) or a Semiconductor-Laser (60 Km).** Electr Lett 21(12):541–542, Jun 6 1985.

Swaminathan V. et al., **Characterization of GaAs Films Grown by Metalorganic Chemical Vapor Deposition.** J Appl Phys 57(12):5349–5353, Jun 15 1985.

Szymanski T. G., Vanwyk C. J., **Goalie—A Space Efficient System for VLSI Artwork Analysis.** IEEE Des T 2(3):64–72, Jun 1985.

Tewksbury S. K., **Attojoule MOSFET Logic Devices Using Low-Voltage Swings and Low Temperature.** Sol St Elec 28(3):255–276, Mar 1985.

Tomita A., Cohen L. G., **Leaky-Mode Loss of the Second Propagating Mode in Single-Mode Fibers With Index Well Profiles.** Appl Optics 24(11):1704–1707, Jun 1 1985.

Tung R. T., Gibson J. M., **Single-Crystal Silicide Silicon Interfaces—Structures and Barrier Heights.** J Vac Sci A 3(3):987–991, May–Jun 1985.

Vanderziel J. P., Mikulyak R. M., Logan R. A., **7.5-km Bidirectional Single-Mode Optical-Fibre Link Using Dual-Mode InGaAsP InP 1.3 μm Laser Detectors.** Electr Lett 21(11):511–512, May 23 1985.

Waaben S., Moskowitz I., Federico J., Dyer C. K., **Computer Modeling of Batteries From Nonlinear Circuit Elements.** J Elchem So 132(6):1356–1362, Jun 1985.

Wertheim G. K., Kwo J., Teo B. K., Keating K. A., **XPS Study of Bonding in Ligated Au Clusters.** Sol St Comm 55(4):357–361, Jul 1985.

Woody D. P., Miller R. E., Wengler M. J., **85-115-GHz Receivers for Radio Astronomy.** IEEE Micr T 33(2):90–95, Feb 1985.

PHYSICAL SCIENCES

Becker R. S., Golovchenko J. A., Swartzentruber B. S., **Tunneling Images of Germanium Surface Reconstructions and Phase Boundaries.** Phys Rev L 54(25):2678–2680, Jun 24 1985.

Bevilacqua R. M., Stark A. A., Schwartz P. R., **The Variability of Carbon Monoxide in the Terrestrial Mesosphere as Determined From Ground-Based Observations of the $J = 1^*$ O Emission Line.** J Geo Res-A 90(ND3):5777–5782, Jun 20 1985.

Chemla D. S., **Quantum Wells for Photonics.** Phys Today 38(5):56+, May 1985.

Cohen R. L., West K. W., **Aluminum Spot Weld Strength Determined From Electrical Measurements.** Welding J 64(6):37–41, Jun 1985.

Fisher D. S., **Random Fields, Random Anisotropies, Nonlinear Sigma Models, and Dimensional Reduction.** Phys Rev B 31(11):7233–7251, Jun 1 1985.

Fisher D. S., Friedan D., Qui Z. G., Shenker S. J., Shenker S. H., **Random Walks in Two-Dimensional Random Environments With Constrained Drift Forces.** Phys Rev A 31(6):3841–3845, Jun 1985.

Franey J. P., Graedel T. E., **Corrosive Effects of Mixtures of Pollutants.** J Air Pollu 35(6):644–648, Jun 1985.

Gardner F. F., Hoglund B., Shukre C., Stark A. A., Wilson T. L., **Observations of Ortho-Thioformaldehyde and Parathioformaldehyde.** Astron Astr 146(2):303–306, May 1985.

Henkel C., Matthews H. E., Morris M., Terebey S., Fich M., **Molecular Lines in IRC + 10-Degrees-216 and CIT-6.** Astron Astr 147(1):143–154, Jun 1 1985.

Huse D. A., Henley C. L., **Pinning and Roughening of Domain Walls in Ising Systems Due to Random Impurities.** Phys Rev L 54(25):2708–2711, Jun 24 1985.

Katz H. E., **Chelate and Macrocycle Effects in the 2,2′-Bipyridine N,N′-Dioxide Complexation of Alkyltin Trichlorides.** J Org Chem 50(12):2086–2091, Jun 14 1985.

Levy L. P., **Nonlinear Spin Dynamics of Quantum Paramagnetic Fluids.** Phys Rev B 31(11):7077–7092, Jun 1 1985.

Li T. G., **Lightwave Telecommunication.** Phys Today 38(5):24–31, May 1985.

Lovinger A. J., **Polymorphic Transformations in Ferroelectric Copolymers of Vinylidene Fluoride Induced by Electron Irradiation.** Macromolec 18(5):910–918, May 1985.

Lovinger A. J., Belfiore L. A., Bowmer T. N., **Crystallographic Changes in Cryogenically Pulverized Polymers.** J Pol Sc Pp 23(7):1449–1466, Jul 1985.

Magalhaes F. M., Beccone J. P., Irvin J. C., Perelli S. J., Schlosser W. O., **A Microwave GaAs-FET Power Module With GaAs Matching Circuits—The M-FET (Matched Field-Effect Transistor).** Microwave J 28(5):205+, May 1985.

Panek M. G. et al., **Thermolysis Rates and Products of the Putative Ketochloroallyl Groups in Poly(Vinyl-Chloride), as Inferred From the Behavior of Analogous Model Compounds (Letter).** Macromolec 18(5):1040–1041, May 1985.

Peeters F. M., Jackson S. A., **Frequency-Dependent Response of an Electron on a Liquid-Helium Film.** Phys Rev B 31(11):7098–7108, Jun 1 1985.

Phillips J. C., **Structure and Properties—Mooser-Pearson Plots.** Helv Phys A 58(2–3):209–215, 1985.

Phillips J. C., **Vibrational Thresholds Near Critical Average Coordination in Alloy Network Glasses.** Phys Rev B. 31(12):8157–8163, Jun 15 1985.

Pieranski P. et al., **Steps on Surfaces of Liquid-Crystal Blue Phase-I.** Phys Rev A 31(6):3912–3923, Jun 1985.

Rinnert K., Lanzerotti L. J., Dehmel G., Gliem F. O., Krider E. P., Uman M. A., **Measurements of the RF Characteristics of Earth Lightning With the Galileo Probe Lightning Experiment.** J Geo Res-A 90(ND4):6239–6244, Jun 30 1985.

Roth H. D., Hutton R. S., **Geometrical Isomerism in Divalent-Carbon Intermediates.** Tetrahedron 41(8):1567–1578, 1985.

Sermage B., Heritage J. P., Dutta N. K., **Temperature Dependence of Carrier Lifetime and Auger Recombination in 1.3-μm InGaAsP.** J Appl Phys 57(12):5443–5449, Jun 15 1985.

Starnes W. H., **C-13 NMR-Studies of the Structures and Polymerization Mechanisms of Poly(Vinyl-Chloride) and Related Copolymers.** Pur A Chem 57(7):1001–1008, Jul 1985.

Stoffel N. G., Kevan S. D., Smith N. V., **Experimental Band Structure of 1T-$TiSe_2$ in the Normal and Charge-Density-Wave Phases.** Phys Rev B. 31(12):8049–8055, Jun 15 1985.

Webster I. A., Schwier C. E., Bates F. S., **Using the Rotational Masking Concept to Enhance Substrate Inhibited Reaction Rates—Controlled Pore Supports for Enzyme Immobilization.** Enzyme Micr 7(6):266–274, Jun 1985.

Wescott L. D. et al., **Mechanistic Studies on the Role of Copper-Containing and Molybdenum-Containing Species as Flame and Smoke Suppressants for Poly(Vinylchloride).** J An Ap Pyr 8(1–4):163–172, Apr 1985.

Wilson B. A., Kerwin T. P., Harbison J. P., **Optical Studies of Thermalization Mechanisms in A-Si-H.** Phys Rev B 31(12):7953–7957, Jun 15 1985.

Yurke B., **Wideband Photon Counting and Homodyne Detection.** Phys Rev A 32(1):311–323, Jul 1985.

Yurke B., **Squeezed-Coherent-State Generation Via 4-Wave Mixers and Detection Via Homodyne Detectors.** Phys Rev A 32(1):300–310, Jul 1985.

SOCIAL AND LIFE SCIENCES

Dybwad G. L., **High-School Help (Letter).** Phys Today 38(6):102, Jun 1985.

Sagi D., Julesz B., **Where and What in Vision.** Science 228(4704):1217–1219, Jun 7 1985.

Wachter K. W., Becker R. A., **Are Productive People to Be Found—Robust Analysis of Sparse Two-Way Tables.** J Am Stat A 80(390):266–276, Jun 1985.

SPEECH/ACOUSTICS

Gharavi H., Steele R., **Conditional Entropy Encoding of Log-PCM Speech.** Electr Lett 21(11):475–476, May 23 1985.

STATEMENT OF OWNERSHIP, MANAGEMENT AND CIRCULATION
(Required by 39 U.S.C. 3685)

1. Title of publication: AT&T Technical Journal.
1B. Publication No. 005-8580.
2. Date of filing: October 1985.
3. Frequency of issue: Monthly, except combined May–June and July–August.
4. Publication office: AT&T Bell Laboratories, 101 J. F. Kennedy Pkwy, Short Hills, NJ 07078.
5. Headquarters: AT&T, 550 Madison Ave., New York, NY 10022.
6. Publisher: AT&T, 550 Madison Ave., New York, NY 10022. Managing Editor: Pierce Wheeler, AT&T Bell Laboratories, 101 J. F. Kennedy Pkwy, Short Hills, NJ 07078.
7. Owner: AT&T, 550 Madison Ave., New York, NY 10022.
8. Bondholders, mortgages and other security holders: none.
9. Extent and nature of circulation:

	Average number of copies of each issue during preceding 12 months	Actual number of copies of single issue published nearest to filing date
A. Total no. copies printed (Net press run)	11,822	11,000
B. Paid circulation		
1. Sales through dealers and carriers, street vendors and counter sales	2,273	2,445
2. Mail subscriptions	3,560	3,155
C. Total paid circulation	5,833	5,600
D. Free distribution by mail, carrier or other means, samples, complimentary and other free copies	5,466	4,877
E. Total distribution (sum of C and D)	11,299	10,477
F. Copies not distributed		
1. Office use, left over, unaccounted, spoiled after printing	523	523
2. Returns from news agents	None	None
G. Total (sum of E and F—should equal net press run shown in A)	11,822	11,000

I certify that the statements made by me above are correct and complete.
Pierce Wheeler, Managing Editor

AT&T TECHNICAL JOURNAL is abstracted or indexed by *Abstract Journal in Earthquake Engineering, Applied Mechanics Review, Applied Science & Technology Index, Chemical Abstracts, Computer Abstracts, Current Contents/Engineering, Technology & Applied Sciences, Current Index to Statistics, Current Papers in Electrical & Electronic Engineering, Current Papers on Computers & Control, Electronics & Communications Abstracts Journal, The Engineering Index, International Aerospace Abstracts, Journal of Current Laser Abstracts, Language and Language Behavior Abstracts, Mathematical Reviews, Science Abstracts (Series A, Physics Abstracts; Series B, Electrical and Electronic Abstracts; and Series C, Computer & Control Abstracts), Science Citation Index, Sociological Abstracts, Social Welfare, Social Planning and Social Development,* and *Solid State Abstracts Journal.* Reproductions of the Journal by years are available in microform from University Microfilms, 300 N. Zeeb Road, Ann Arbor, Michigan 48106.


AT&T