THE BELL SYSTEM TECHNICAL JOURNAL

MAY–JUNE 1981
VOL. 60, NO. 5

THE BELL SYSTEM TECHNICAL JOURNAL

THE BELL SYSTEM TECHNICAL JOURNAL is published monthly, except for the May-June and July-August combined issues, by the American Telephone and Telegraph Company, C. L. Brown, Chairman and Chief Executive Officer; W. M. Ellinghaus, President; V. A. Dwyer, Vice President and Treasurer; F. A. Hutson, Jr., Secretary. Editorial inquiries should be addressed to the Editor, The Bell System Technical Journal, Bell Laboratories, Room WB 1L-331, Crawfords Corner Road, Holmdel, N.J. 07733. Checks for subscriptions should be made payable to The Bell System Technical Journal and should be addressed to Bell Laboratories, Circulation Group, Whippany Road, Whippany, N.J. 07981. Subscriptions $20.00 per year; single copies $2.00 each. Foreign postage $1.00 per year; 15 cents per copy. Printed in U.S.A. Second-class postage paid at New Providence, New Jersey 07974 and additional mailing offices.

ISSN0005-8580

Comments on the technical content of any article or brief are welcome. These and other editorial inquiries should be addressed to the Editor, The Bell System Technical Journal, Bell Laboratories, Room WB 1L-331, Crawfords Corner Road, Holmdel, N.J. 07733. Comments and inquiries, whether or not published, shall not be regarded as confidential or otherwise restricted in use and will become the property of the American Telephone and Telegraph Company. Comments selected for publication may be edited for brevity, subject to author approval.

THE BELL SYSTEM TECHNICAL JOURNAL

DEVOTED TO THE SCIENTIFIC AND ENGINEERING
ASPECTS OF ELECTRICAL COMMUNICATION

Volume 60 | May–June 1981 | Number 5

Dispersionless Single-Mode Lightguides With α Index Profiles

By U. C. PAEK, G. E. PETERSON, and A. CARNEVALE

(Manuscript received November 14, 1980)

By means of a numerical solution to Maxwell's equations, we calculate those parameters necessary to design and fabricate single-mode lightguides. These include optimum core radius and profile parameter α. In the design the dispersion is minimized by varying the core radius while the relative index difference Δ, the wavelength λ, and profile parameter are held fixed. A comparison of calculated dispersion with experimental data shows excellent agreement.

I. INTRODUCTION

Recently, single-mode lightguides have been developed that achieve transmission losses as low as 0.5 dB/km and 0.2 dB/km at wavelengths of 1.3 μm and 1.55 μm, respectively.[1] When the total lightguide dispersion is reduced to zero at the operating wavelength, a transmission system can be realized with wide repeater spacings and extremely large bandwidths.[2–4] In fact, bandwidths in excess of 1 GHz/100 km are expected. Clearly, such a lightguide system would be ideal for undersea cable and other long-distance transmission applications.

The design of single-mode lightguides with zero total dispersion requires an accurate description of this parameter in terms of profile shape, materials properties, and core radius. This necessitates a solution of Maxwell's equations for the fundamental HE_{11} lightguide mode as well as the TE_{01} and TM_{01} modes.

For radially inhomogeneous media, it is usually not possible to obtain these solutions as analytical expressions of a closed form. To a very large extent the design of single-mode lightguides has in the past focused on rectangular-shaped index profiles. This is because Maxwell's equations are very much easier to solve in this case. When nonrectangular profiles have been considered, numerous approximations have often been employed.

To avoid these difficulties, we have developed a numerical technique to obtain exact solutions to the vector form of Maxwell's equations.[5] These equations are written as four coupled simultaneous first-order differential equations. The effective index N_e for the single propagating HE_{11} mode is found by solving the characteristic equation derived from the matching conditions of the field components at the core cladding interface. The dispersion is then calculated from N_e. In addition, we find the conditions under which the TE_{01} and TM_{01} modes propagate.

Our computing procedures do not impose any restrictions on the index profile of the fiber.[5,6] However, for the sake of simplicity and familiarity we choose the α-index profile. To confirm the analysis, we compare the computed results with known experiments when possible. The calculated results include the dispersion, the optimum core radius (defined as the value at which the zero total dispersion will occur for a given relative index difference, wavelength, and α value), and the corresponding cutoff frequency.

To keep the accuracy as high as possible, material dispersion is included in the calculation from the onset. The subtle interaction between material properties and waveguide properties determine the propagation characteristics of the lightguide. Details of the numerical procedure can be found in the appendix.

II. DESIGN OF A DISPERSIONLESS SINGLE-MODE FIBER

In the design of a single-mode fiber, the relative index difference Δ, the operating wavelength λ, and the cladding index N_2 are normally specified. The objective of the design is to make the total dispersion for the HE_{11} mode equal to zero, the total dispersion D_t being defined as the time spread of a narrow pulse per fiber length L per source spectral width $d\lambda$. Qualitatively speaking, this can be accomplished by adjusting the radius and profile shape so that the waveguide dispersion exactly counterbalances the material dispersion. In addition, the design must be such that the *TE*, *TM*, and other higher-order *HE* and *EH* modes are not propagating. In the analysis shown in the appendix, the number of modes allowed to propagate in the fibert core are not restricted. However, the single-mode analysis is nothing but a particular case of multimode propagation and it can be easily achieved simply by setting the angular mode number $M = 0$ and 1 in the elements of matrix A [eq. (18) in the appendix] while fixing the radial mode number $Q = 1$. When $M = 0$, the cutoff frequency for the TE_{01} and TM_{01} can be determined. Below this cutoff point, only a fundamental mode (HE_{11}) propagate in the fiber. All the modes other than $M = 0$ are hybrid modes. Further details are explained below.

When eqs. (22) and (23) are substituted into eq. (21), determinant (21) decomposes into two uncoupled equations, giving the *TM* and *TE* modes,

TM mode:

$$\frac{(\beta_1^2\Lambda_{21} - \kappa(\zeta_i)\gamma_0\Lambda_{11})}{(\beta_1^2\Lambda_{22} - \kappa(\zeta_i)\gamma_0\Lambda_{12})} = 1. \tag{1}$$

TE mode:

$$\frac{(\beta_1^2\Lambda_{32} - \gamma_0\Lambda_{42})}{(\beta_1^2\Lambda_{31} - \gamma_0\Lambda_{41})} = 1. \tag{2}$$

When N_1 is the maximum value of the index of refraction in the core and N_2 is the cladding index, the effective index of the bounded modes must satisfy the condition $N_2 < N_e < N_1$. Therefore, from eqs. (1) and (2) the effective index $N_e(0, 1)$ for the TM_{01} and TE_{01} mode can be found.[5] The notation $N_e(M, Q)$ refers to the effective index of a mode having angular mode number M and radial mode number Q.

In practice, the core radius is changed until $N_e(0, 1)$ is equal to N_2. This gives us the cutoff radius $a_c\cdot$ for the TE_{01} and TM_{01} modes. The normalized cutoff frequency is then calculated from

$$V_C = \frac{2\pi a_c}{\lambda}\sqrt{N_1^2 - N_2^2}. \tag{3}$$

Now if $V < V_c$, only a single mode (HE_{11}) propagates. The effective index N_e for this mode is calculated in the single-mode region as a function of the radius and is the principal quantity used to determine the dispersion of the HE_{11} mode.

The relationship between the effective index $N_e(1, 1)$ and the group index N_g is given by

$$N_g = N_e - \lambda\frac{dN_e}{d\lambda}. \tag{4}$$

The propagation time of a pulse through a fiber of length L and group index N_g is

$$t = \frac{L}{C}N_g. \tag{5}$$

If there is a variation of N_g with wavelength, there is a dispersion or spread in pulse width. Thus

$$dt \cong \frac{L}{C}\frac{dN_g}{d\lambda}\,d\lambda, \tag{6}$$

where $d\lambda$ is the spectral width of the source.

We can substitute N_g from eq. (4) into eq. (6) and get

$$dt = -\frac{L}{C}(d\lambda)\lambda\frac{d^2N_e}{d\lambda^2}. \tag{7}$$

The total dispersion D_t is

$$D_t = \frac{dt}{d\lambda}\cdot\frac{1}{L}. \tag{8}$$

Thus,

$$D_t = -\frac{\lambda}{C}\frac{d^2N_e}{d\lambda^2}. \tag{9}$$

Obviously, we need to solve Maxwell's equations for N_e and then calculate $d^2N_e/d\lambda^2$ to obtain D_t.

Following the procedure described in our earlier papers, it is convenient to describe the dispersive properties of the cladding by a modified Sellmeier formula.[5,7] Thus,

$$N_2 = C_0 + C_1\lambda^2 + C_2\lambda^4 + \frac{C_3}{(\lambda^2 - l)} + \frac{C_4}{(\lambda^2 - l)^2} + \frac{C_5}{(\lambda^2 - l)^3}, \tag{10}$$

where $l = 0.035$. The coefficients C_i are given in Table I. Figure 1 shows the variation of the silica cladding's refractive index, for the lightguides considered here.

Since the relative index difference Δ is rather small, typically ranging from 0.2 percent to 0.8 percent, the core center index N_1 can be written as

$$N_1 = \frac{N_2}{[1 - \Delta]}. \tag{11}$$

To obtain a feel for the size of the numbers involved, the following data are useful:

$$\lambda = 1.33\ \mu\text{m},$$

$$N_2 = 1.446925,$$

$$N_1 = 1.449825 \text{ [as calculated by eq. (11)]}.$$

Table I

C_0	=	1.4508554
C_1	=	−0.0031268
C_2	=	−0.0000381
C_3	=	0.0030270
C_4	=	−0.0000779
C_5	=	0.0000018

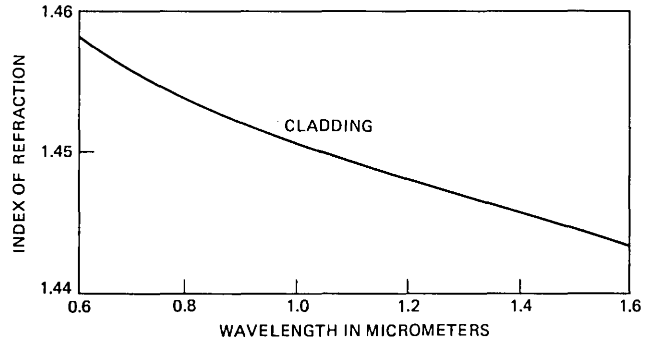


Fig. 1—Refractive index of silica as a function of wavelength.

Since we assume a wavelength dependence for N_2, described by eq. (10), N_1 as calculated by eq. (11) is also wavelength dependent. The following profile formula is particularly useful and will be employed in the fiber design:

$$N(r) = N_1[1 - \Delta r^{\alpha}]. \tag{12}$$

A few comments seem in order: When $\alpha \rightarrow \infty$ the profile is rectangular, when $\alpha = 2$ it is parabolic, when $\alpha = 1$ it becomes linear, and when $\alpha < 1$ the profile develops a cusp. Figure 2 illustrates these shapes.

In practice, one specifies Δ, λ, and the profile parameter α. The computer then searches for a core radius that makes $D_t = 0$ by means of Mueller's iterative method. If it can be assumed that the total dispersion is the sum of waveguide dispersion $D_w(\lambda, a, \alpha, \Delta)$ and material dispersion $D_m(\lambda)$, then the physics of the problem is easy to understand. As will be shown later, this separation is quite accurate. Thus, a dispersionless single-mode fiber is one in which material dispersion is exactly balanced by waveguide dispersion. This is illustrated by Fig. 3.

Curve $D_m(\lambda)$ shows the material dispersion as a function of wavelength for a single-mode lightguide with $\Delta = 0.2$ percent and a rectangular profile. Note that it changes sign and passes through zero at about 1.27 μm. Also plotted are waveguide dispersion curves for three different radius values. Note that all three curves are negative and that as the radius *decreases* the magnitude of the dispersion increases at a given wavelength.

From Fig. 3 we can draw a number of conclusions. First, the lightguide with the materials properties shown can be made totally dispersion free only at wavelengths longer than 1.27 μm. Second, as the wavelength gets longer the guide radius must get smaller. Finally, at longer wavelengths a much larger amount of material dispersion

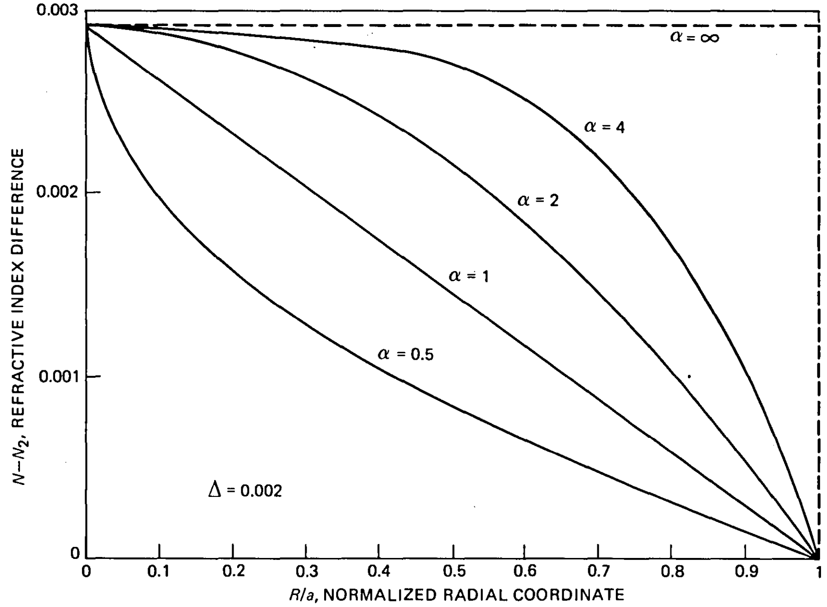


Fig. 2—Index profile shapes for various power laws. When $\alpha \rightarrow \infty$ the profile is rectangular, when $\alpha = 2$ it is parabolic, and when $\alpha = 1$ it is linear. If α is less than 1, the profile develops a cusp.

must be compensated for by waveguide dispersion. This requires greater precision in the waveguide parameters than when the guide is designed to operate at the zero of material dispersion. Hence, it is desirable to have a variety of materials available for lightguide design purposes.

Finally, a typical total dispersion curve D_t, illustrating lightguide operation at 1.3 μm, is plotted in Fig. 3.

III. SINGLE-MODE LIGHTGUIDES WITH α PROFILES FOR 1.33-μm OPERATION

Single-mode lightguides that are designed to operate at 1.33 μm with $\Delta = 0.2$ percent are of much current interest. The cutoff frequencies $V_c(\alpha)$ for the α-index profiles can be found from eqs. (1) and (2). Figure 4 plots the results. There is little change in V_c until the profile parameter is below 10, then it increases rapidly. At $\alpha = 1$, V_c and V_{opt} are 4.383 and 2.724, respectively. The data of Fig. 4 agree well with that in the literature.[8,9]

To determine the radius a_{opt} for zero total dispersion, a computer search is done for values less than a_c, where a_c is the cutoff radius [see eq. (3)]. Figure 5 shows the result for three α values, namely 100, 2, and 1. The radii turn out to be 4.142, 5.725, and 6.294, respectively.

Note that the total dispersion D_t is most sensitive to the radius for $\alpha = 100$, and least sensitive when $\alpha = 1$. For design purposes, we plot a_{opt} as a function of the profile parameter α in Fig. 6. This plot also includes the cutoff wavelength λ_c, where λ_c is defined by the following formula:

$$\lambda_c = (V_{\mathrm{opt}}/V_c)\cdot\lambda. \tag{13}$$

Note that both V_{opt} and V_c are employed in the calculation. Therefore, to allow only one mode to propagate in the core, the operating wavelength must be longer than the cutoff wavelength λ_c. For a step index profile with $\Delta = 0.2$ percent and $\lambda = 1.33$ μm, λ_c is close to 1 μm.

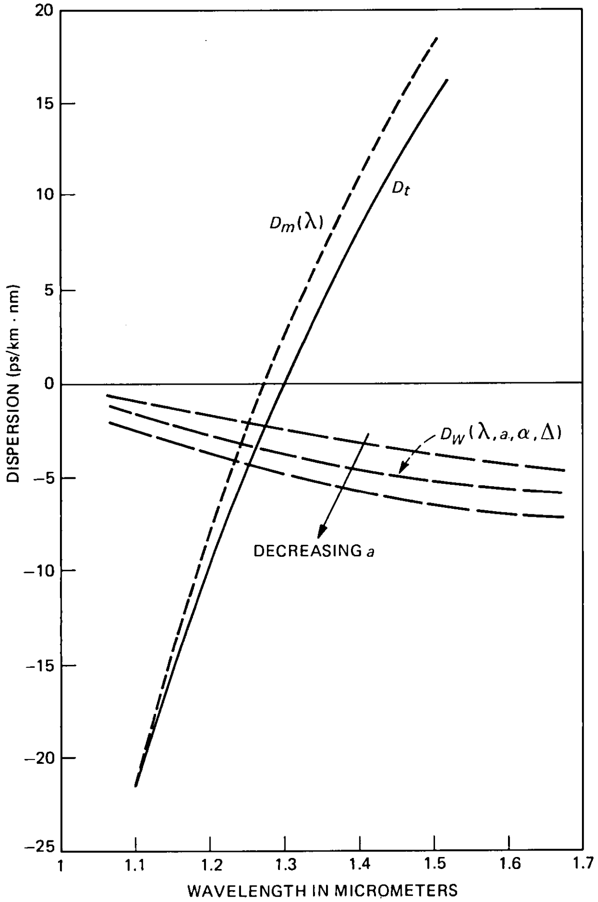


Fig. 3—A plot showing material dispersion D_m, waveguide dispersion D_w, and total dispersion D_t. Note that because of the difference in sign between the material dispersion and the waveguide dispersion it is possible for one to cancel the other at a particular wavelength.

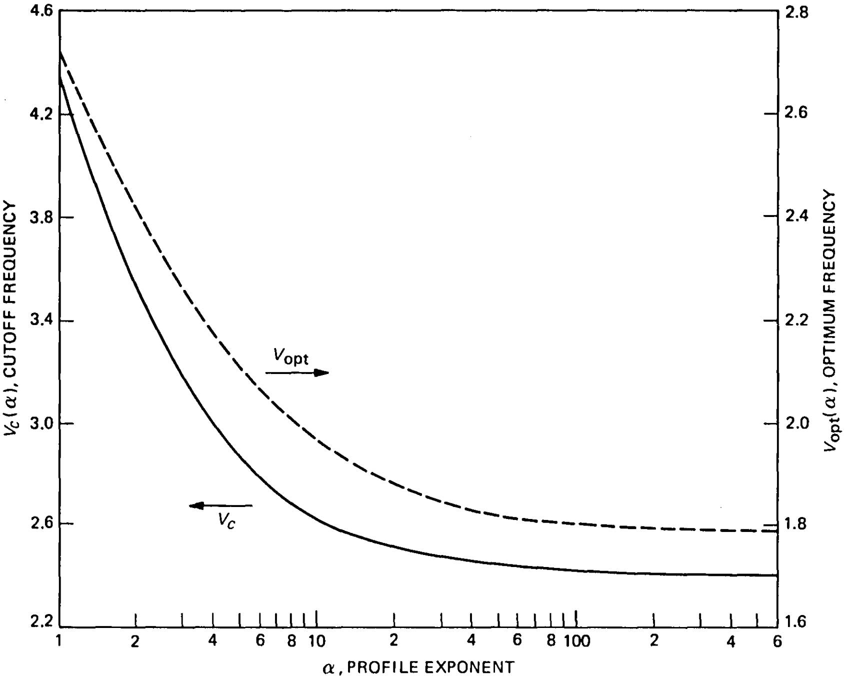


Fig. 4—A plot of V_c and V_{opt} as a function of the profile parameter.

The calculations described thus far in this section have made no assumption concerning the separation of the total dispersion D_t into its component parts D_m and D_w. Our numerical procedure simply searches for a zero in total dispersion and does not assume that $D_t = D_m + D_w$.

However, the material dispersion contribution can be calculated directly by means of eqs. (10) and (9), where N_2 is substituted for N_e. The waveguide dispersion can be found by solving Maxwell's equations numerically via the procedure outlined in the appendix, with N_2 being wavelength independent.

Figures 7a through 7c show the results for $\alpha = 100$, 2, and 1. In all three figures the material dispersion, which is of course independent of core radius, is represented by the horizontal dashed line. The negatives of the waveguide dispersions are dependent on core radius and are the solid lines in the figures. The intersection of the solid and dashed lines yields the optimum radii. As is evident from the figures, the radii are identical with the previous calculations.

IV. A COMPARISON WITH EXPERIMENTAL DATA

Miya, Terunuma, and Hosaka[10] have fabricated GeO_2 doped single-mode lightguides for the 1.3 μm and 1.5 μm region. Their experimental data are very useful to check the theory presented in this paper. They give data on a fiber with $\Delta = 0.2$ percent and a zero total dispersion at $\lambda = 1.33$ μm. A second fiber has $\Delta = 0.74$ percent and a zero total dispersion at $\lambda = 1.54$ μm. Both are step-index fibers. Their experimental data are measured with a fiber Raman laser excited by a Q switched, mode-locked Nd:YAG laser. The spectral range studied is 1.1 to 1.7 μm. The optimum radius is 4.142 μm for $\Delta = 0.2$ percent and

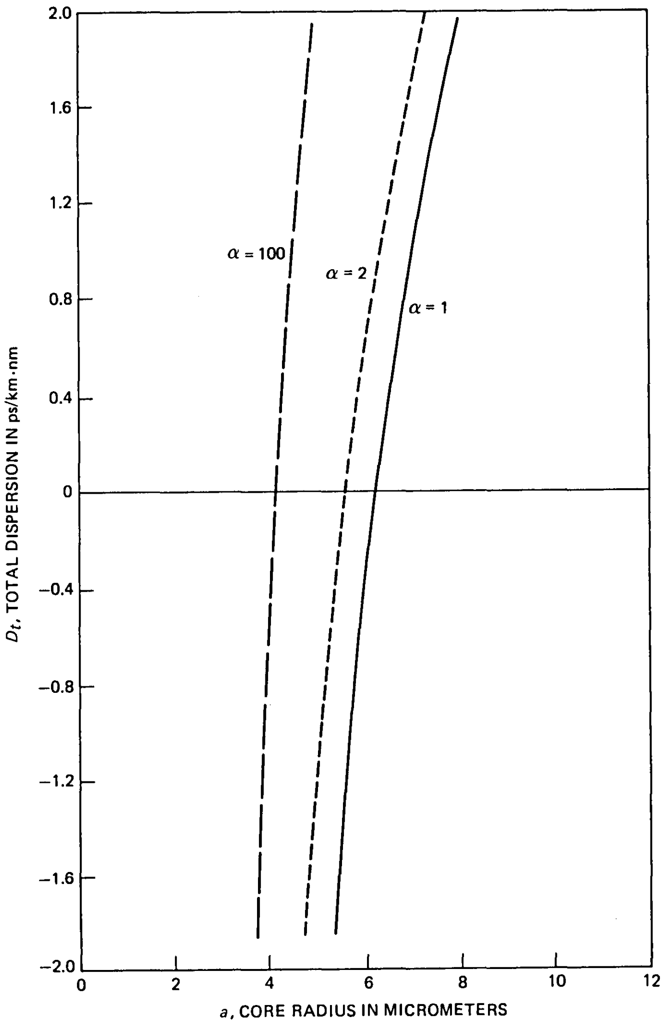


Fig. 5—Total dispersion versus core radius for lightguides with $\alpha = 1$, $\alpha = 2$, and $\alpha = 100$. Note that the total dispersion curve is steepest for $\alpha = 100$ and shallowest for $\alpha = 1$.

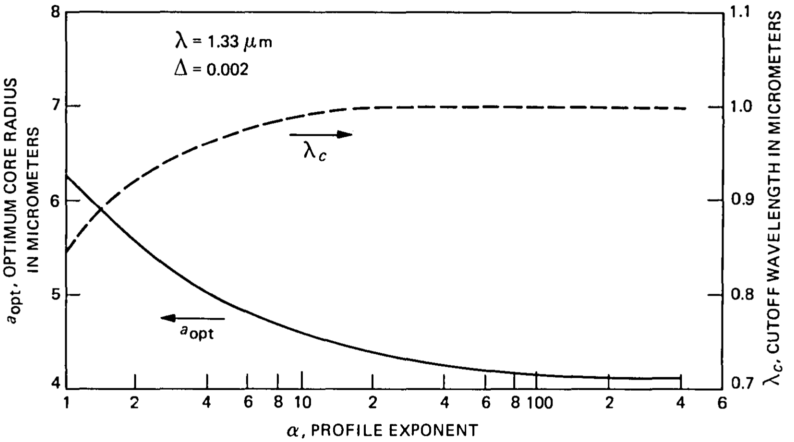


Fig. 6—A plot of cutoff wavelength λ_c and optimum radius a_{opt} versus α. This data is very useful for design purposes.

$\lambda = 1.33$ μm. The optimum radius is 2.30 μm for $\Delta = 0.75$ percent and $\lambda = 1.53$ μm. Figure 8 compares their experimental results (circles and triangles) with our calculations (solid and dashed lines). The agreement is excellent.

V. DISCUSSION

At a wavelength of 1.33 μm, dispersion-free lightguides can be designed with improved properties over the rectangular profile guides. Our calculated results of a_{opt} and cutoff frequency $V_c(\alpha)$ both show a rapid increase as the profile parameter α drops below 10. For a triangular index profile ($\alpha = 1$), the core size is over 50 percent larger than the size of a step-index core. Thus, due to ease of handling, coupling to the source, and splicing, a larger core size is desirable for the same value of Δ and λ.

As the profile parameter α drops below 10, V_{opt} also begins to increase substantially. It is significant that in the case of $\alpha = 1$, $V_{opt} \sim 2.7$, which is a substantial increase over the step-index case, $V_{opt} \sim 1.8$.[11,12]

In addition, Fig. 5 shows that a lightguide with a triangular profile is less susceptible to variations in core radius than a step-index guide. Therefore, because of the larger core size and less sensitivity in its tolerance, a triangular profile may be less difficult to fabricate than a rectangular profile.

VI. ACKNOWLEDGMENTS

We wish to thank M. I. Cohen, R. J. Klaiber, and L. S. Watkins for their helpful discussions and encouragements.

APPENDIX

To study the propagating modes in a fiber, we first assume that the cross section of the fiber is concentric in the core and cladding and that the core diameter $2a$ is much smaller than the outer diameter D (Fig. 9). Further, we assume that the core is a lossless, radially inhomogeneous medium with a scalar permittivity ϵ. A relative permittivity (dielectric constant) is defined by $\kappa = \epsilon/\epsilon_0$, where ϵ_0 is the value in free space. However, as is well known in a nonconducting medium, the permittivity becomes equal to the square of the index of refraction N. For source-free fields in a dielectric waveguide, Maxwell's equations reduce to

$$\begin{aligned} \nabla \times E &= -\mu_0 \frac{\partial H}{\partial t}, \\ \nabla \times H &= \epsilon_0 \kappa \frac{\partial E}{\partial t}, \end{aligned} \tag{14}$$

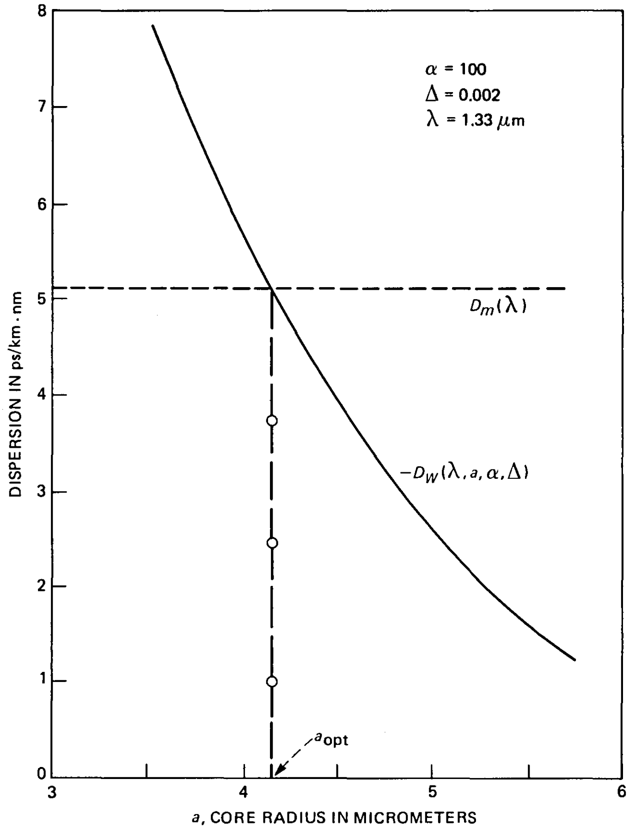


Fig. 7(a)—A plot of waveguide dispersion and material dispersion as a function of core radius, $\alpha = 100$.

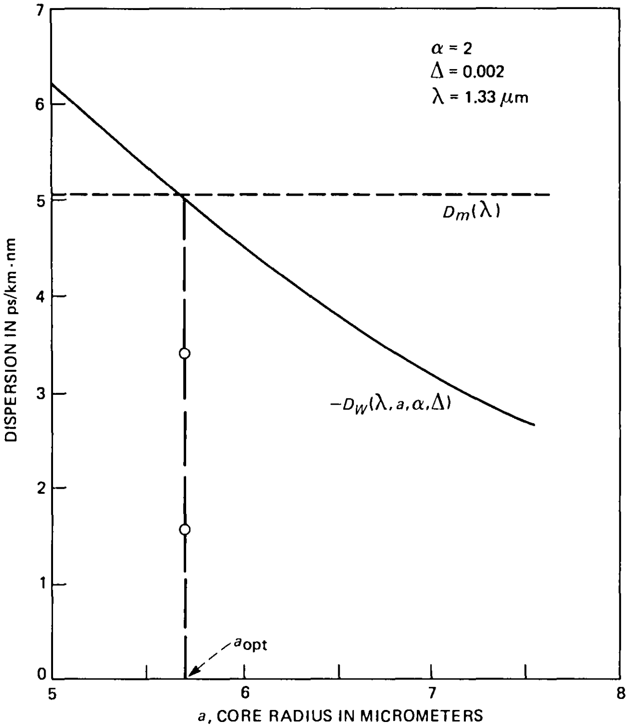


Fig. 7(b)—A plot of waveguide dispersion and material dispersion as a function of core radius, $\alpha = 2$.

where $E(\mathbf{R}, t)$ and $H(\mathbf{R}, t)$ represent the electric and magnetic fields. μ_0 is a scalar permeability in free space.

In a cylindrical coordinate system $\mathbf{R} = \{R, \phi, z\}$, the solution of eq. (14) is described by the vector components of two fields, $\{E_R, E_\phi, E_z\}$ and $\{H_R, H_\phi, H_z\}$. Among these components we are primarily interested in finding the tangential components $\{E_\phi, E_z\}$ and $\{H_\phi, H_z\}$. These will be continuous through the core-cladding interface. Once the tangential components are known, the radial components E_R and H_R can be obtained from these components. To find a complete set of bounded modes we consider the following form of solution:[13]

$$E(\rho, \phi, z, t) = E(\rho) \exp[i(\omega t - M\phi - \beta z)], \tag{15}$$

where ω and M are angular frequency and mode number, and β is propagation constant along the z axis.

Introducing a new variable Λ, the field components are written as

$$\Lambda = \begin{Bmatrix} \Lambda_1 \\ \Lambda_2 \\ \Lambda_3 \\ \Lambda_4 \end{Bmatrix} = \begin{Bmatrix} E_z \\ -iZ_0\rho H_\phi \\ \rho E_\phi \\ -iZ_0 H_z \end{Bmatrix}, \tag{16}$$

where $\rho = KR = 2\pi R/\lambda$, Z_0 is the wave impedance defined by (μ_0/ϵ_0),[1,2] and λ is a wavelength.

Substituting eq. (16) into eq. (15) yields a set of the first-order ordinary differential equations.[13,14] In the following equation,

$$\frac{d\Lambda}{d\rho} = \frac{1}{\rho} A(\rho)\Lambda, \tag{17}$$

the operator $A(\rho)$ is a 4 × 4 matrix that contains important property constants and parameters, for instance, the index of refraction N, angular mode number M, and the effective index N_e defined by β/k.

The 4 × 4 matrix $A(\rho)$ can be written as

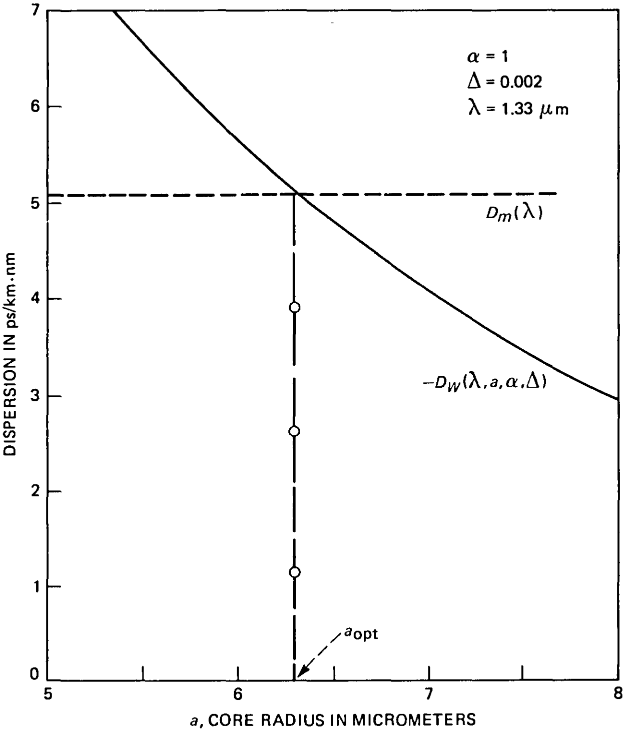


Fig. 7(c)—A plot of waveguide dispersion and material dispersion as a function of core radius, $\alpha = 1$.

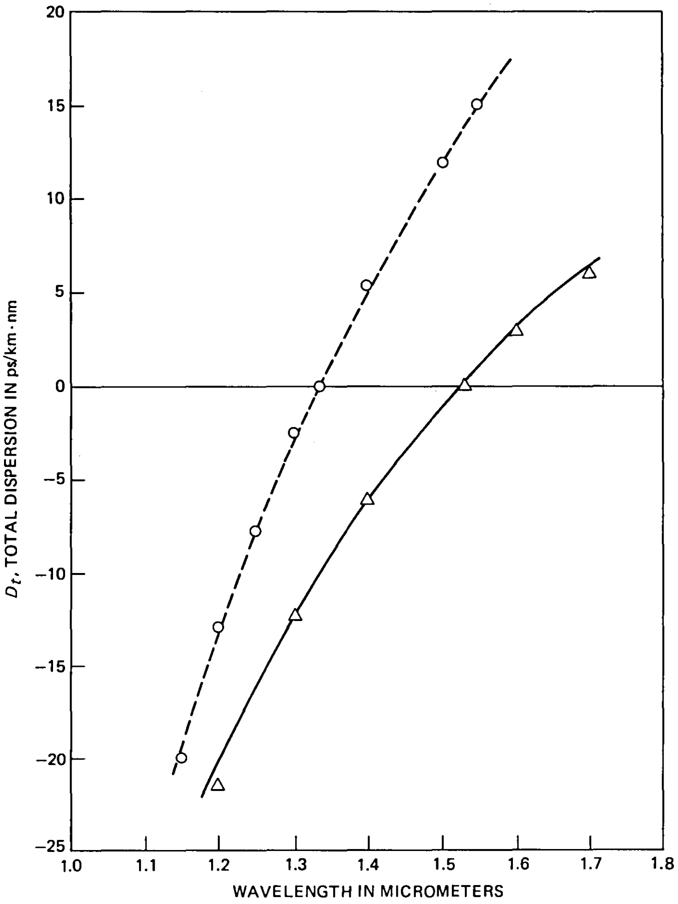


Fig. 8—A comparison of theory (dashed and solid lines) with experiment (see text).

$$\begin{vmatrix} 0 & (N_e^2/\kappa) - 1 & 0 & -MN_e/\kappa \\ \rho^2\kappa - M^2 & 0 & MN_e & 0 \\ 0 & MN_e/\kappa & 0 & \rho^2 - (M^2/\kappa) \\ -MN_e & 0 & N_e^2 - \kappa & 0 \end{vmatrix} . \quad (18)$$

The method of numerical computation has been given in our earlier work,[5] which briefly gives the procedure for finding the solutions and emphasizes what is needed to describe a single-mode fiber design.

Equation (17) is a familiar type of differential equation. A fourth-order Runge-Kutta method is used to solve it numerically. A concise description is as follows. First, we know that there are two possible solutions in both the core and cladding, and they are linearly independent. Therefore, a general solution will be a linear combination of two solutions, denoted by Λ_i in the core and Γ_i in the cladding.

In the core region,

$$\Lambda_i = A_1\Lambda_{i1} + A_2\Lambda_{i2}, \tag{19}$$

and in the cladding region,

$$\Gamma_i = A_3\Lambda_{i3} + A_4\Lambda_{i4}. \tag{20}$$

The solutions Λ_{i1} and Λ_{i2} are transferred to the core-cladding interface through the operation of eq. (17), starting with two initial conditions given at the center of the core.[5,13]

At the interface $\rho = \rho_i$, each has to be matched with the solution in the cladding to satisfy the continuity condition mentioned earlier. Thus, the matching condition yields a set of homogeneous equations for the constants A_i. Hence, the characteristic equation is the vanishing determinant of the system of equations,

$$\det|\Lambda_{ij}| = 0. \tag{21}$$

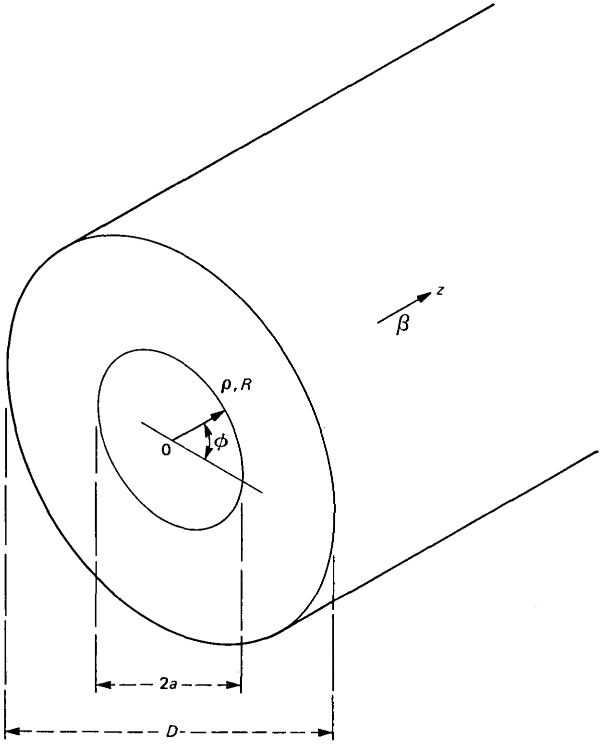


Fig. 9—Cross section of a fiber and cylindrical coordinate system. R or ρ is the radial coordinate, ϕ is the angle, and z is the axial direction ($\rho = KR$).

Particularly, considering $M = 0$, we can write the solutions in the core and cladding as follows. The two core solution vectors at $\rho = \rho_i$ are Λ_{i1} and $\Lambda_{i2}(\rho_i)$. Expressed in terms of their components, they are

$$\Lambda_{i1} = \begin{bmatrix} \Lambda_{11} \\ \Lambda_{21} \\ \Lambda_{31} \\ \Lambda_{41} \end{bmatrix} \quad \text{and} \quad \Lambda_{i2} = \begin{bmatrix} \Lambda_{12} \\ \Lambda_{22} \\ \Lambda_{32} \\ \Lambda_{42} \end{bmatrix}. \tag{22}$$

And the two cladding solutions at $\rho = \rho_i$ are taken as

$$\Lambda_{i3} = \begin{bmatrix} \beta_1^2 \\ \kappa(\zeta_i)\gamma_0(\zeta_i) \\ 0 \\ 0 \end{bmatrix} \cdot W_0(\zeta_i) \quad \text{and} \quad \Lambda_{i4} = \begin{bmatrix} 0 \\ 0 \\ \gamma_0(\zeta_i) \\ \beta_1^2 \end{bmatrix} \cdot W_0(\zeta_i), \tag{23}$$

where

$$\beta_1 = [N_e^2 - \kappa(\zeta_i)]^{1/2}, \qquad \zeta_i = \beta_i \rho_i,$$

$$\gamma_0 = \zeta_i \cdot [K_0'(\zeta_i)/K_0(\zeta_i)], \qquad \text{and} \qquad W_0(\zeta_i) = K_0(\zeta_i)/\beta_1^2.$$

The prime in the above equation denotes differentiation with respect to ζ.

REFERENCES

1. T. Miya, Y. Terunuma, T. Hosaka, and T. Miyashita, "Ultimate Low-Loss Single-Mode Fiber at 1.55 μm," Electron. Lett., *15* (1979), p. 106.
2. L. G. Cohen, C. Lin, and W. G. French, "Tailoring Zero Chromatic Dispersion Into the 1.5–1.6 μm Low-Loss Spectral Region of Single-Mode Fibers," Electron. Lett., *15* (1979), p. 334.
3. W. A. Gambling, H. Matsumura, and C. M. Ragdale, "Zero Total Dispersion in Graded-Index Single-Mode Fibers," Electron. Lett., *15* (1979), p. 474.
4. H. Tsuchiya and N. Imoto, "Dispersion-free Single-Mode Fiber in 1.5 μm Wavelength Region," Electron. Lett., *15* (1979), p. 476.
5. G. E. Peterson, A. Carnevale, U. C. Paek, and D. W. Berreman, "An Exact Numerical Solution to Maxwell's Equations for Lightguides," B.S.T.J., *59,* No. 7 (September 1980), pp. 1175–96.
6. G. E. Peterson, A. Carnevale, and U. C. Paek, "Comparison of Vector and Scalar Modes in a Lightguide with a Hyperbolic Secant Index Distribution," B.S.T.J., *59,* No. 9 (November 1980), pp. 1681–91.
7. J. W. Fleming, "Material Dispersion in Lightguide Glasses," Electron. Lett., *14* (1978), p. 326.
8. W. A. Gambling, D. N. Payne, and H. Matsumura, "Cut-Off Frequency in Radially Inhomogeneous Single-Mode Fiber," Electron. Lett., *13* (1977), p. 139.
9. Y. Kokubun and K. Iga, "Mode Analysis of Graded-Index Optical Fibers Using a Scalar Wave Equation Including Gradient-Index Terms and Direct Numerical Integration," J. Opt. Soc. Am., *70* (1980), p. 388.
10. T. Miya, Y. Terunuma, and T. Hosaka, "Fabrication of Single-Mode Fibers for 1.5 μm Wavelength Region," Dig. Jpn. Top. Meet., *OQE78-76* (1979).
11. D. Marcuse, "Microbending Losses of Single-Mode Step Index and Multimode, Parabolic-Index Fibers," B.S.T.J., *55,* No. 7 (September 1976), pp. 937–55.
12. V. J. Tekippe, "Evanescent Wave Coupling of Optical Fibers," presented at the Conference on the Physics of Fiber Optics at the 82nd Annual Meeting of the American Chemical Society, Chicago, Illinois, April 27–30, 1980.
13. M. O. Vassel, "Calculation of Propagating Modes in a Graded-Index Optical Fiber," Opto-Electron., *5* (1974), p. 271.
14. A. Vigants and S. P. Schlesinger, "Surface Waves On Radially Inhomogeneous Cylinders," IEEE Trans., *MTT-10* (1962), p. 375.

Copyright © 1981 American Telephone and Telegraph Company
THE BELL SYSTEM TECHNICAL JOURNAL
Vol. 60, No. 5, May-June 1981
Printed in U.S.A.

A Class of Closed Markovian Queuing Networks: Integral Representations, Asymptotic Expansions, and Generalizations*

By J. McKENNA, D. MITRA, and K. G. RAMAKRISHNAN

(Manuscript received October 28, 1980)

Closed Markovian networks of queues that have the product form in their stationary probability distributions are useful in the performance evaluation and design of computer and telecommunication systems. Therefore, the efficient computation of the partition function—the key element of the solution in product form—has attracted considerable effort. We present a new and broadly applicable method for calculating the partition function. This method can be applied to very large networks, which were previously computationally intractable. Most of the paper details applications of this approach to a network class which arose in modeling an interactive processor. We show that the partition function and derivatives such as mean values (response times, CPU utilizations, etc.) may be represented by integrals and their ratios. The integrands contain a parameter N which is large for large networks. Next, the classical techniques of asymptotic analysis are applied to derive three main power series expansions in descending powers of N to correspond to normal, high, and very high usage. This work emphasizes multiple terms in the expansions for precision and error analyses.

I. INTRODUCTION

The theoretical results on the product form of the stationary distributions of large classes of Markovian queuing networks continue to have a profound influence on computer communications, computer systems analysis, and traffic theory.[1-4] These results make at least feasible the analysis and synthesis of the large systems of ever increasing complexity being considered in these areas. The subclass of *closed*

* Presented at ORSA-TIMS meeting, Jan. 5–7, 1981, at Boca Raton, Fla.

networks of queues is more difficult to analyze than the open networks because there is no stationary independence of the network nodes. However, the incentive for investigating the closed networks does exist since they have been used to model multiple-resource computer systems,[2,5] multiprogrammed computer systems,[6-8] time-sharing,[2] and window flow control in computer communication networks;[9,10] networks with external inputs subject to blocking require the analysis of a large number of closed networks.[11,12] The closed network model that we shall use for illustrative purposes arose in the modeling of a central processor in a node of a computer network. This network is subject to a variety of processing demands. In recognition of the utility of closed networks, considerable research and commercial interest has been directed towards developing efficient procedures for computing the partition function (the normalizing constant), the only element of the product form solution requiring significant computation.[13-18]

However, as these existing recursive techniques are applied to the problems of particular interest in the Bell System, wherein the constituents of the closed chains are many and the number of chains are many, their shortcomings are observed to be severe in the amount of computing time and memory required and the accuracy attained. A more detailed account appears in Section 2.4 and Section IX. Briefly, the existing recursive techniques are largely ineffectual.

We present a new way to view the problem, which surmounts many of the difficulties associated with large networks. The approach is broadly applicable—as indicated in Section X—even though the paper is a detailed account of applications to a specific class of closed networks. The new approach consists, first, of recognizing that the partition function may be written as an integral with a large parameter N present in the integrand to reflect the large size of the network. Next, the classical techniques of asymptotic analysis are applied to derive an asymptotic power series, typically in descending powers of N. The integrand will have fundamentally different properties in different ranges of the system parameters and this will require correspondingly different expansions. Thus, in this paper we develop three separate series expansions—Proposition 3 (Section IV), Proposition 12 (Section VII), Proposition 17 (Section VIII)—each corresponding to a specific range of values of the usage parameter α. It is worth emphasizing that, commensurate with an objective of providing solutions with any desired accuracy, we give procedures for generating multiple terms in the asymptotic expansions, not just the dominant term. In Section VIII, we unify the preceding results by giving a common expansion that holds uniformly in the system parameters. The uniform expansions introduce in a natural way the parabolic cylinder (or Weber) functions, a classical family of special functions with many antecedents and ties with other special functions.[13] Besides duplicating the specialized expansions derived earlier, the uniform expansion makes available for use the many well-documented and tested expansions that are known in connection with parabolic cylinder functions.

Section IX describes a user-oriented software package that has been written in C-language to implement the approach developed here. We supply results obtained by the package on four test problems that arose in analyzing performance of a Bell System project. Also reported are the results of a comparison with a well known, commercially marketed package that obtains solutions recursively. Our package is able to solve the large problems, which are well beyond the range of the other package, and, surprisingly, solve the small problems as well with errors that have small bounds.

Section X provides the basis for extending the approach developed here to quite general multiprocessor, multidiscipline queuing networks. We show that for most networks that have been shown to have the product form in their stationary probability distribution, the partition function has an integral representation. The expansions appropriate for its computation are not considered here.

Not surprisingly, the new representation of the partition function as an integral—the starting point of our computational procedures—may be exploited anew to derive analytical estimates and bounds of the quantities of interest, such as throughput, mean response time, etc. We demonstrate particularly in Section 5.3 that these formulas explicitly exhibit the system parameters and as such are rather useful as design and synthesis aids. (The bounds are also useful as checks on the computational procedure.) Purely computational procedures by themselves do not yield this particular form of insight into system behavior.

The asymptotic sequences used typically are power series in N^{-1}, where recall N is the generic large parameter.[14] Thus, the number of terms required to achieve the desired accuracy decreases with increasing N. In contrast, with recursive solutions the computational complexity grows with the network size. Also, the asymptotic methods handle increased numbers of *classes* of constituents with little incremental difficulty, while the computational complexity in recursive methods grows geometrically. Thus, the contrasting techniques are not replacements for each other but complementary: loosely speaking, the recursions are most effective for smaller networks, while the asymptotic expansions are most effective for large networks.

The contrasting behavior with respect to a large number of classes is of particular importance in computer communications where, as Reiser,[9] Schwartz,[3] and others have pointed out, traffic corresponding to each source-destination pair is treated as a separate class and the network closure follows from the windows employed in flow control. Reiser has developed heuristics to cope with this situation.

References 11, 15, and 16 also contain results pertaining to large networks.

II. NETWORK MODEL AND KNOWN RESULTS

2.1 Model

In the model (see Fig. 1) each constituent, which may be thought of as a terminal or station, of the closed network spends alternating periods of time in the two nodes that constitute the network—the 'think' node (also node 1) and the 'CPU' node (node 2). The think time in each cycle for each constituent is an independent random variable with an exponential distribution. The time spent in the CPU node depends on many factors since it is here that there occurs interaction between constituents being serviced. We stipulate that the CPU discipline is 'processor sharing'* and that the desired service time (i.e., the time required to service the job if the entire CPU was dedicated to the job) is an independent exponentially distributed random variable.[1,4] The 'think' node is thus an ∞-server center and in the terminology of Ref. 4, nodes 1 and 2 are respectively Type 3 and 2 centers.

We stipulate that there is sufficient statistical inhomogeneity amongst the constituents to justify the existence of several, say p, classes of constituents, with class i having K_i constituents, $1 \leq i \leq p$. Statistical homogeneity applies within a class in that ρ_{i1} and ρ_{i2} will respectively denote the mean think time and the mean desired service time that are common to all in class i. The variations among these mean values may be quite substantial.

Our involvement with this model arose while modeling behavior of traffic through a processing node of a computer network. The number of classes of constituents is at least five, namely, time sharing; inquiry/response and data-base query; batch and remote job entry; messages and broadcast; data entry/collect and screen type jobs. The mean values $\{\rho_{ij}\}$ are obtained from benchmark measurements. In another variation of this problem, a finer classification of constituents was considered. Our interest is in cases where the individual class populations $\{K_i\}$ extend to several hundred, while the number of classes is of the order of ten.

* In the processor-sharing discipline there is no overt queuing because all, say n, jobs present in the node simultaneously receive service at $1/n$ times the rate given to a single job by the processor. Thus, the rate given to a single specific constituent fluctuates with time. This discipline is the limiting case of the round robin discipline as the time quantum given to each job becomes arbitrarily small.

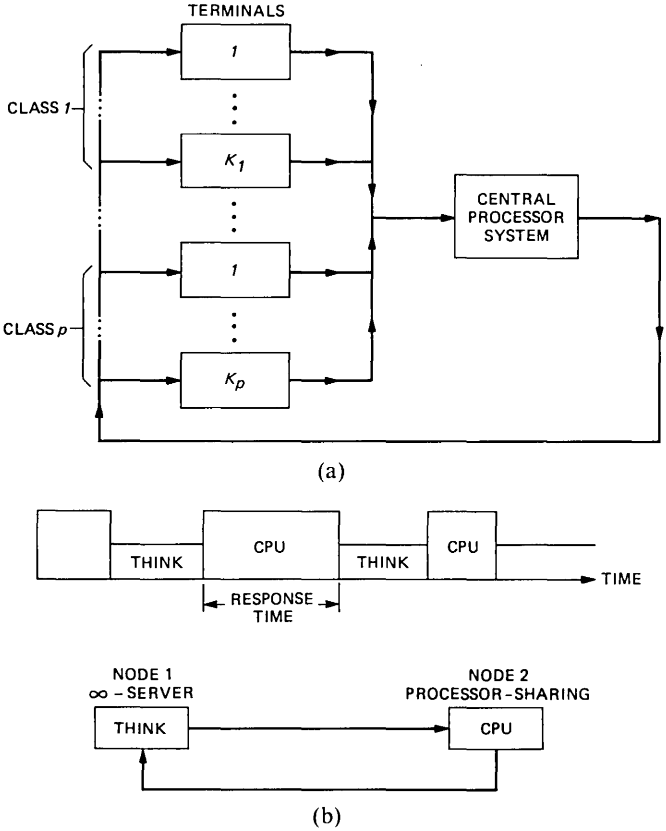


Fig. 1—(a) There are p classes of constituents—shown as terminals—with K_j constituents in class j. (b) Constituents spend alternate periods of time in the think node and processor-sharing CPU node.

2.2 Product form solution

If N_{ij} is the stationary random variable denoting the number of constituents of class i in node j, and if π is the following stationary probability $\pi(n_{11}, \cdots, n_{p1}; n_{12}, \cdots, n_{p2}) = Pr[N_{11} = n_{11}, \cdots, N_{p1} = n_{p1}; N_{12} = n_{12}, \cdots, N_{p2} = n_{p2}]$, then it is known that with the left hand side abbreviated to $\pi(\mathbf{n}_1, \mathbf{n}_2)$,[4]

$$\pi(\mathbf{n}_1, \mathbf{n}_2) = \frac{1}{G(\mathbf{K})} \left(\prod_{i=1}^{p} \frac{\rho_{i1}^{n_{i1}}}{n_{i1}!} \right) \left(\sum_{j=1}^{p} n_{j2} \right)! \left(\prod_{k=1}^{p} \frac{\rho_{k2}^{n_{k2}}}{n_{k2}!} \right), \tag{1}$$

where $G(\mathbf{K}) = G(K_1, K_2, \cdots, K_p)$ is the normalization constant so chosen as to make the sum of all quantities in (1) equal to 1. Explicitly,

$$G(\mathbf{K}) = \sum_{m_p=0}^{K_p} \cdots \sum_{m_1=0}^{K_1} \left(\prod_{i=1}^{p} \frac{\rho_{i1}^{(K_i-m_i)}}{(K_i - m_i)!} \right) \left(\sum_{j=1}^{p} m_j \right)! \left(\prod_{k=1}^{p} \frac{\rho_{k2}^{m_k}}{m_k!} \right). \tag{2}$$

The function $G(\cdot)$ defined on the integer lattice in $\mathbf{R}^P$ is referred to as the partition function.*

2.3 System performance

A number of interrelated system performance measures are obtained from the partition function. We start with $\overline{N_{i1}(\mathbf{K})}$, the mean number of constituents of class i in node 1 ('think'), and obtain directly from (2),

$$\overline{N_{i1}(\mathbf{K})} = \rho_{i1} G(\mathbf{K} - \mathbf{e}_i)/G(\mathbf{K}), \tag{3}$$

where $\mathbf{e}_i$ is our notation for the vector with the i^{th} component unity and all other components zero. Thus, $G(\mathbf{K} - \mathbf{e}_i)$ is the partition function associated with a new population with one less constituent in the i^{th} class. From (3) and Little's theorem applied to class i and node 1, we obtain for the throughput of constituents of class i,

$$\lambda_i(\mathbf{K}) = G(\mathbf{K} - \mathbf{e}_i)/G(\mathbf{K}). \tag{4}$$

The mean response time, i.e., time spent in the CPU in each cycle by class i constituents, is obtained again from an application of Little's theorem:

$$t_i(\mathbf{K}) = K_i G(\mathbf{K})/G(\mathbf{K} - \mathbf{e}_i) - \rho_{i1}. \tag{5}$$

Finally, the utilization of the CPU by constituents of class i,

$$u_i(\mathbf{K}) \triangleq \sum \cdots \sum \frac{n_{i2}}{\sum n_{j2}} \pi(\mathbf{n}_1, \mathbf{n}_2) = \rho_{i2} G(\mathbf{K} - \mathbf{e}_i)/G(\mathbf{K}). \tag{6}$$

The important point to note is that all the mean values given in (3) through (6) are simply obtained from the knowledge of the partition function estimated at two neighboring points of the integer lattice in $\mathbf{R}^P$. As the above quantities are all closely related, we shall henceforth consider only the last, $\{u_i(\mathbf{K})\}$.

Higher moments of N_{i1}, the random number of constituents of class i in node 1, may also be obtained from knowledge of the partition function:

$$\overline{N_{i1}^2(\mathbf{K})} = \{\rho_{i1}^2 G(\mathbf{K} - 2\mathbf{e}_i) + \rho_{i1} G(\mathbf{K} - \mathbf{e}_i)\}/G(\mathbf{K}), \tag{7}$$

and, for $i \neq j$,

$$\overline{N_{i1} N_{j1}(\mathbf{K})} = \rho_{i1} \rho_{j1} G(\mathbf{K} - \mathbf{e}_i - \mathbf{e}_j)/G(\mathbf{K}). \tag{8}$$

Of course, the moments of $\{N_{i2}\}$ are easily derived from moments of $\{N_{i1}\}$.

* The distribution in (1) and (2) is also the stationary distribution of other networks. For example, if node 2 is first-come-first-served with class independent service rate $1/\rho_2$, then (1) and (2) with $\rho_{i2} \equiv \rho_2$ is the solution.

2.4 Recursive solutions

The above results explain why the problem of efficiently computing the partition function has excited so many researchers.[15,17-21] For the problem at hand it is easy to arrive at the following recursion by established techniques:

$$G(\mathbf{K}) = \sum_{j=1}^{p} \rho_{j2} G(\mathbf{K} - \mathbf{e}_j) + \prod_{j=1}^{p} \frac{\rho_{j1}^{K_j}}{K_j!}. \tag{9}$$

The boundary conditions are: $G(\mathbf{K}) = 0$ for $\mathbf{K} \ngeq 0$.

Observe that the partition functions themselves can be scaled on account of the linearity and the fact that only ratios of partition function values have physical content. By the same token, implementations of (9), for large $\mathbf{K}$, typically give rise to values which are either very small or very large leading to rather severe problems of overflow and underflow. Proper scaling is only marginally helpful.

The main problems with implementing (9) are with respect to time, memory required during computation, and accuracy. A straightforward application of (9) would require an estimated K^p iterations, where K is the generic class size. Similarly, the storage required would be approximately K^{p-1}. Now these crude estimates can clearly be improved upon by simply pruning or avoiding the computation of intermediate lattice points, but this would be at the cost of increased algorithmic complexity, and the extent of the accrued benefits are not generally known. The underflow/overflow phenomenon that affects the accuracy of the scheme has already been commented on; a no less severe problem is accumulation of round-off errors in a large number of iterations.

The recursive solution in (9) is one of several that can be generated by recently discovered techniques. However, all solutions that we are aware of are recursive and share to varying degrees the three broad categories of limitations just discussed.

III. INTEGRAL REPRESENTATIONS

3.1 Partition function

We start with Euler's integral

$$n! = \int_0^\infty e^{-t} t^n dt. \tag{10}$$

Substituting for the middle term in braces in (2) we obtain*

* If the range of integer subscripts is not stipulated explicitly, then the range is understood to be $[1, p]$.

$$G(\mathbf{K}) = \int_0^\infty e^{-t} \sum_{m_p=0}^{K_p} \cdots \sum_{m_1=0}^{K_1} \left(\prod \frac{\rho_{j1}^{K_j-m_j}}{(K_j - m_j)!}\right)\left(\prod \frac{(t\rho_{j2})^{m_j}}{m_j!}\right) dt$$

$$= \frac{1}{\{\prod K_j!\}} \int_0^\infty e^{-t} \sum_{m_p=0}^{K_p} \cdots \sum_{m_1=0}^{K_1} \prod \binom{K_j}{m_j} \rho_{j1}^{K_j-m_j} (t\rho_{j2})^{m_j} dt$$

$$= \frac{1}{\{\prod K_j!\}} \int_0^\infty e^{-t} \prod (\rho_{j1} + t\rho_{j2})^{K_j} dt. \tag{11}$$

To obtain our final form for the partition function, let N be the generic large parameter to be associated with a large network. Also, for $j = 1, 2, \cdots p$, let

$$K_j = \beta_j N, \tag{12}$$

$$r_j \triangleq \left.\frac{\text{mean think time}}{\text{mean service time}}\right|_{\text{class } j} = \frac{\rho_{j1}}{\rho_{j2}}$$

$$= \gamma_j N. \tag{13}$$

The suggestion in this notation is that $\{\beta_j\}$ and $\{\gamma_j\}$ are 0(1), which is the situation to which our work is primarily directed.

There is considerable latitude in selecting N. The following choice is certainly not essential but as it does ease some of the manipulations, we use it throughout the paper:

$$N = (\prod r_i^{K_i})^{1/\Sigma K_j}. \tag{14}$$

An implication of this choice of N is that

$$\sum \beta_j \log \gamma_j = 0. \tag{15}$$

We return to (11) to observe

$$G(\mathbf{K}) = \left(\prod \frac{\rho_{j2}^{K_j}}{K_j!}\right) \int_0^\infty e^{-t} \prod (r_j + t)^{K_j} dt$$

$$= \left(\prod \frac{\rho_{j2}^{K_j}}{K_j!}\right) N^{\Sigma K_j + 1} \int_0^\infty e^{-Nz} \prod (\gamma_j + z)^{\beta_j N} dz. \tag{16}$$

Finally, we have after using (14),

Proposition 1:

$$G(\mathbf{K}) = \left(N \prod \frac{\rho_{j1}^{K_j}}{K_j!}\right) \int_0^\infty e^{-Nf(z)} dz, \tag{17}$$

$$\text{where } f(z) \triangleq z - \sum \beta_j \log(\gamma_j + z). \quad \square \tag{18}$$

We shall see that typically there is no need to compute the term in braces in (17).

3.2 Representations of mean values and some higher moments

We have two options concerning representations of the physically interesting quantities in (3) through (7). Concerning ourselves with $u_i(\mathbf{K})$ given in (6), we may simply use the respective integral representations for $G(\mathbf{K} - \mathbf{e}_i)$ and $G(\mathbf{K})$ and obtain for $i = 1, 2, \cdots, p$,

$$u_i(\mathbf{K}) = \frac{\beta_i}{\gamma_i} \frac{\hat{N} \displaystyle\int_0^\infty e^{-\hat{N}\hat{f}(z)} dz}{N \displaystyle\int_0^\infty e^{-Nf(z)} dz}, \tag{19}$$

where $\hat{N}$ and $\hat{f}(\cdot)$ are defined analogously to N and $f(\cdot)$ but for a network with one less constituent in the i^{th} class. Obviously N and $\hat{N}$ as well as $f(\cdot)$ and $\hat{f}(\cdot)$ are going to be close to each other, but the above option takes no further notice of this fact.

In the contrasting option, we proceed as follows. Observe that

$$u_i(\mathbf{K} + \mathbf{e}_i)^{-1} = \frac{1}{\rho_{i2}} \frac{G(\mathbf{K} + \mathbf{e}_i)}{G(\mathbf{K})}. \tag{20}$$

Now, from (16),

$$\begin{aligned} G(\mathbf{K} + \mathbf{e}_i) &= \frac{\rho_{i2}}{K_i + 1} \left(\prod \frac{\rho_{j2}^{K_j}}{K_j!} \right) \int_0^\infty (r_i + t) e^{-t} \prod_j (r_j + t)^{K_j} dt \\ &= \frac{\rho_{i2}}{K_i + 1} \left(N^2 \prod \frac{\rho_{j1}^{K_j}}{K_j!} \right) \int_0^\infty (\gamma_i + z) e^{-Nf(z)} dz, \end{aligned} \tag{21}$$

where to obtain the last equation we have proceeded just as we did from (16) to (17). The point to note is that in the above expression N and $f(\cdot)$ are identical to that used in the expression for $G(\mathbf{K})$ in (17) and (18). Finally, from (20), for $i = 1, 2, \cdots, p$

Proposition 2:

$$u_i(\mathbf{K} + \mathbf{e}_i)^{-1} = \frac{1}{\beta_i + 1/N} \left(\gamma_i + \frac{\displaystyle\int_0^\infty z e^{-Nf(z)} dz}{\displaystyle\int_0^\infty e^{-Nf(z)} dz} \right). \quad \square \tag{22}$$

Notice that the ratio of integrals, the only quantity requiring significant computation, is common to all classes.

The higher moments given in (7) and (8) have similar representations which are easy to derive. We give the form for one term that occurs in (7) which reveals the general pattern:

$$\frac{1}{\rho_{i1}^2}\frac{G(\mathbf{K}+2\mathbf{e}_i)}{G(\mathbf{K})}=\frac{1}{N^2}\frac{1}{\gamma_i^2(\beta_i+1/N)(\beta_i+2/N)}$$

$$\cdot\left(\gamma_i^2+2\gamma_i\frac{\displaystyle\int_0^\infty ze^{-Nf(z)}dz}{\displaystyle\int_0^\infty e^{-Nf(z)}dz}+\frac{\displaystyle\int_0^\infty z^2e^{-Nf(z)}dz}{\displaystyle\int_0^\infty e^{-Nf(z)}dz}\right). \qquad (23)$$

The feature to note is that the i^{th} moment involves integrals of the form $\int_0^\infty z^j e^{-Nf(z)}dz$, $j=0, 1, \cdots, i$.

3.3 Properties of the function f

We shall later need to recall certain properties of the function $f(z)$ in (18), $z \geq 0$. As a consequence of (15),

$$f(0)=0. \qquad (24)$$

Note

$$\begin{aligned} f^{(i)}(z) &= 1-\sum \beta_j/(\gamma_j+z) \qquad \text{for } i=1 \\ &= (-1)^i(i-1)!\sum \beta_j/(\gamma_j+z)^i \qquad \text{for } i>1, \end{aligned} \qquad (25)$$

and the following alternating sign property that holds for $i = 2, 3, \cdots,$

$$(-1)^i f^{(i)}(z)>0, \qquad z \geq 0. \qquad (26)$$

Also, for $i=2, 3, \cdots,$

$$|f^{(i)}(z)| \leq |f^{(i)}(0)|, \qquad z \geq 0. \qquad (27)$$

As the second derivative is positive in $[0, \infty)$, the function is always convex.

Let us view the function $f(z)$ in the interval $\gamma_m < z < \infty$, where $\gamma_m \triangleq -\min_i \gamma_i$ (see Fig. 2). As the figure indicates, the derivative of the function tends to ∞ at both ends of the interval. Coupled with the convexity and the other already established facts, it follows that the function has a unique stationary point, a minimum, in (γ_m, ∞). As in the figure, we let $\hat{z}$ denote this unique stationary point.

This stationary point may be obtained as the unique real solution in (γ_m, ∞) to the equation

$$\sum_{j=1}^{p}\frac{\beta_j}{\gamma_j+z}=1.$$

Thus, the largest real root of a polynomial of order p gives $\hat{z}$.

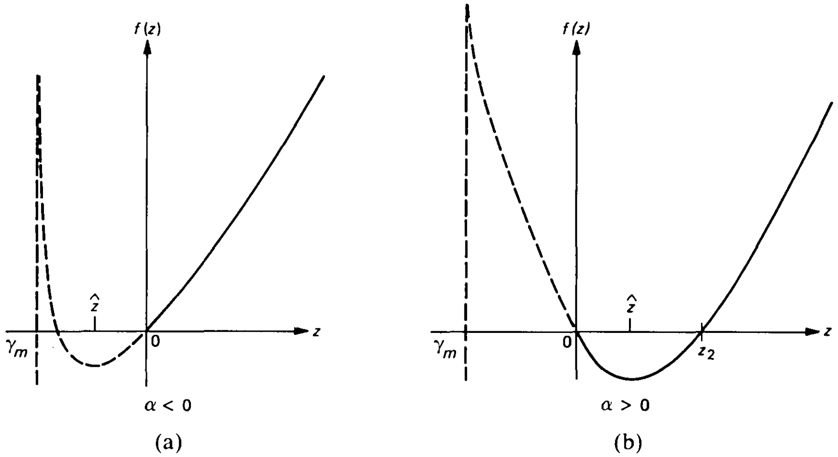


Fig. 2—Sketches of the function $f(z)$.

It will be important for us to distinguish the cases $\hat{z} < 0$ and $\hat{z} \geq 0$. The slope of the function $f(z)$ at the origin effectively indicates which case holds: the first if the slope is positive and the second otherwise. Thus, a key parameter of the system is

$$\begin{aligned} \alpha &\triangleq -f^{(1)}(0) = \sum \beta_i/\gamma_i - 1 \\ &= \sum K_i/r_i - 1. \end{aligned} \tag{28}$$

From the previous discussion we thus have that

$$\begin{aligned} \operatorname*{Min}_{z\geq 0} f(z) &= f(0) \quad \text{if} \quad \alpha \leq 0 \\ &= f(\hat{z}) \quad \text{if} \quad \alpha > 0. \end{aligned} \tag{29}$$

Equation (29) summarizes the background on the stationary point as needed for most of the paper. For example, $\hat{z}$ is needed only if $\alpha > 0$. However, Section VIII is exceptional in that, while considering small possibly negative α, the corresponding stationary point $\hat{z}$ is required to be known. Note that as $\alpha \rightarrow 0$, $\hat{z} \sim \alpha/\sum \beta_i/\gamma_i^2$.

The parameter α, $\alpha > -1$, is an indicator of the traffic intensity, with increasing α corresponding to higher traffic intensities. Our results, theoretical and numerical, show that $\alpha \geq 0$ corresponds to heavy usage corresponding to close to 100 percent utilization of the CPU. 'Normal' usage in large networks will certainly require $\alpha < 0$ and in all likelihood α will not be close to 0. For this reason, the most comprehensive results given here are for the case $\alpha < 0$.

IV. ASYMPTOTIC EXPANSIONS FOR NORMAL TRAFFIC

Throughout Section IV we shall consider $\alpha < 0$.

4.1 Laplace's method

We shall first apply Laplace's method[14,22,23] to obtain asymptotic expansions (see Appendix A for notation) for the integral in the representation of the partition function in (17), and subsequently to the other integrals in (22) and (23). Laplace's method observes that when N is large, the minimum of $Nf(z)$ at $z = 0$ is very sharp and that the dominant contribution to the integral comes from the neighborhood of 0.

In the integral

$$I = \int_0^\infty e^{-Nf(z)}dz, \tag{30}$$

let us change variables,

$$u \triangleq f(z), \tag{31}$$

so that

$$I = \int_0^\infty e^{-Nu}\left(\frac{dz}{du}\right)du. \tag{32}$$

To obtain dz/du, let us begin with the power series convergent in the neighborhood of 0,

$$u = f(z) = \sum_{j\geq 1} f_j z^j, \tag{33}$$

where $f_j = f^{(j)}(0)/j!$. Standard procedures allow the series to be reversed, i.e., give z as a power series in u. Appendix A elaborates on this procedure and explicitly gives the leading terms. Let the series obtained by this procedure be

$$z \triangleq g(u) = \sum_{j\geq 1} g_j u^j, \tag{34}$$

so that

$$\frac{dz}{du} = \sum_{j\geq 0} a_j u^j, \tag{35}$$

where $a_j = (j+1)g_{j+1}$. Substitution of this series in (32) yields

$$I \sim \sum_{j\geq 0} a_j \int_0^\infty e^{-Nu}u^j du$$

and thus

Proposition 3:

$$I \sim \sum_{j\geq 0} \frac{j!a_j}{N^{j+1}} \qquad \text{as} \qquad N \to \infty. \quad \square \tag{36}$$

The leading three terms of the series are obtained from

$$a_0 = 1/f_1; \qquad a_1 = -2f_2/f_1^3; \qquad a_2 = 3(2f_2^2 - f_1 f_3)/f_1^5. \tag{37}$$

The proof that (36) is an asymptotic expansion is available from various sources.[14, 22, 23] Indeed an explicit proof is available from the bounds that we develop in the following section in the course of an error analysis. Nonetheless, we sketch a proof based on Watson's lemma[24] applied to the integral in (32)—the lemma is anyhow used later. Watson's lemma considers the integral $\int_0^\infty e^{-Nu} h(u)du$ in which

(i) $h(u)$ has a convergent power series expansion in the neighborhood of the origin, and

(ii) there exist constants c_1, c_2 such that $|h(t)| < c_1 e^{c_2 t}$ for $t \geq 0$, and asserts that an asymptotic expansion for the integral is obtained by replacing $h(u)$ by its power series and integrating term by term. Thus, series reversion to obtain dz/du and a subsequent application of Watson's lemma gives the asymptotic expansion in (36).

The reader will note for future reference that the asymptotic expansion in (36) may also be written as

$$I \sim \sum_{j=1}^{\infty} g^{(j)}(0)/N^j, \tag{38}$$

where $g(\cdot) = f^{-1}(\cdot)$, as follows from (33) and (34).

Asymptotic expansions of integrals of the form $I^{(k)} = \int_0^\infty z^k e^{-Nf(z)} dz$ follow with only slight modifications. Thus, in lieu of (32) we now have

$$I^{(k)} = \int_0^\infty e^{-Nu} \left(z^k \frac{dz}{du} \right) du. \tag{39}$$

In principle, it is straightforward to obtain a power series for $(z^k dz/du)$, which is convergent in the neighborhood of 0 from the power series in (34). Using the following as the defining relation* for the sequence $\{a_j^{(k)}\}, j = 0, 1, 2, \cdots,$

$$z^k \frac{dz}{du} = \left(\sum_{j \geq 1} g_j u^j \right)^k \left(\sum_{j \geq 0} (j+1) g_{j+1} u^j \right) = \sum_{j \geq 0} a_j^{(k)} u^{j+k}, \tag{40}$$

the asymptotic expansion for the integral is obtained after term-by-term integration, giving

Proposition 4:

$$I^{(k)} = \int_0^\infty z^k e^{-Nf(z)} dz \sim \sum_{j=0}^{\infty} (j+k)! a_j^{(k)} / N^{j+k+1} \tag{41}$$

as $N \to \infty$. □

* In this notation, the sequence $\{a_j\}$ in (35) and (36) is $\{a_j^{(0)}\}$.

Let us consider in greater detail the expansion of the integral $I^{(1)} = \int_0^\infty ze^{-Nf(z)}dz$, which will be needed if (22) is used to compute the mean values. We have derived the following recursive formula which efficiently generates $\{a_j^{(1)}\}$ from the coefficients $\{a_j^{(0)}\}$ that are needed anyhow for the expansion of I: for all $j \geq 0$,

$$a_j^{(1)} = \sum_{k=1}^{j+1} a_{j-k+1}^{(0)} a_{k-1}^{(0)} / k. \tag{42}$$

In particular, the leading three terms of the expansion of $I^{(1)}$ are obtained from,

$$a_0^{(1)} = 1/f_1^2; \quad a_1^{(1)} = -3f_2/f_1^4; \quad a_2^{(1)} = 2(5f_2^2 - 2f_1f_3)/f_1^6.$$

We have derived, but omit to give, a more general recursive formula—of which (42) is a special case—for generating $\{a_j^{(k+1)}\}$ from $\{a_j^{(k)}\}$.

4.2 Asymptotic expansions for the utilizations

As discussed so far, both of the two options stated in Section 3.2 for representing the mean values [see (19) and (22)] require the development of asymptotic expansions of two separate integrals and the subsequent computation of the ratio. Here we observe that a single asymptotic expansion exists for the mean values. The underlying reason is that the asymptotic sequence $\{N^{-j}\}$, $j = 0, 1, \cdots$, form a multiplicative asymptotic sequence.[14,22]

In particular, if as in (36) and (41),

$$\int_0^\infty e^{-Nf(z)}dz \sim \frac{a_0^{(0)}}{N} + \frac{a_1^{(0)}}{N^2} + 2!\,\frac{a_2^{(0)}}{N^3} + \cdots$$

and

$$\int_0^\infty ze^{-Nf(z)}dz \sim \frac{a_0^{(1)}}{N^2} + \frac{2!a_2^{(1)}}{N^3} + \frac{3!a_2^{(1)}}{N^4} \cdots,$$

then

$$\frac{\displaystyle\int_0^\infty ze^{-Nf(z)}dz}{\displaystyle\int_0^\infty e^{-Nf(z)}dz} \sim \sum_{j\geq 1} \frac{b_j}{N^j},$$

where the sequence $\{b_j\}$ is obtained by formal substitution.

The following gives the leading terms for the utilizations derived by the above procedure.

Proposition 5:
For $i = 1, 2, \cdots, p$, as $N \to \infty$,

$$\{u_i(\mathbf{K} + \mathbf{e}_i)\}^{-1}(\beta_i + 1/N) \sim \gamma_i + \frac{A_1}{N} + \frac{A_2}{N^2} + \frac{A_3}{N^3}, \tag{43}$$

where

$$A_1 = -1/\alpha; \quad A_2 = 4f_2/\alpha^3; \quad A_3 = -40f_2^2/\alpha^5 - 18f_3/\alpha^4. \quad \square$$

In accordance with an earlier observation, all terms of the asymptotic series other than the first are independent of i.

Proposition 5 contains a justification for treating the parameter α [see (28)] as an indicator of traffic intensity. Using only the dominant term,

$$\{u_i(\mathbf{K} + \mathbf{e}_i)\}^{-1} \sim \gamma_i/\beta_i \tag{44}$$

and

$$\text{utilization of CPU} = \sum u_i(\mathbf{K}) \sim 1 + \alpha. \tag{45}$$

A necessary caveat is that the above, as indeed all results in this section, has been derived for the assumption $\alpha < 0$.

Since the utilization as given by (45) can come close to unity even with $\alpha < 0$, (45) justifies another earlier statement that for large networks normal usage will not extend beyond the range $\alpha < 0$.

V. ERROR ANALYSIS AND PERFORMANCE BOUNDS FOR NORMAL TRAFFIC

Maintaining the restriction $\alpha < 0$ placed in the preceding section, we supplement in two directions the results obtained so far. First, we obtain essential results on the error incurred in truncating the expansions. These results containing information extending beyond what is required as proof of asymptotic expansions are needed for very practical reasons, such as to know how many terms to use and, more importantly, to help define the regime of applicability. In the second part of the section, certain rather special properties of the functions in the representations are used to derive analytical bounds on the network performance measures.

5.1 Completely monotonic functions

The following result on the function $g(\cdot) = f^{-1}(\cdot)$ is key to much of the error analysis:

Proposition 6:

$$(-1)^j g^{(j)}(u) < 0 \quad \text{for} \quad u \geq 0, \qquad j = 1, 2, 3, \cdots \quad \square \tag{46}$$

An inductive proof is given in Appendix B. By virtue of this result, $g^{(1)}(\cdot)$ is a completely monotonic, or alternating, function (see Ref. 23). The importance of this property stems from the role of $g^{(1)}(u) = dz/du$ in the integral representation (32).

5.2 Error bounds

In connection with the expansion of the integral I in Proposition 3, let R_m denote the error that accrues if only the leading m terms are used, i.e.,

$$R_m = I - \sum_{j=0}^{m-1} j! a_j^{(0)}/N^{j+1} = I - \sum_{j=1}^{m} g^{(j)}(0)/N^j. \tag{47}$$

Now by the mean value theorem,[25] for each u there is a $\xi(u)$ in $[0, u]$ such that

$$g^{(1)}(u) = \sum_{j=1}^{m} g^{(j)}(0)u^{j-1}/(j-1)! + g^{(m+1)}(\xi)u^m/m! \tag{48}$$

On substitution in (32),

$$I = \sum_{j=1}^{m} g^j(0)/N^j + R_m, \tag{49}$$

where

$$R_m = \frac{1}{m!}\int_0^{\infty} e^{-Nu} g^{(m+1)}\{\xi(u)\}u^m du. \tag{50}$$

A simple corollary to (50) and Proposition 6 is

Proposition 7:

$$\begin{aligned} R_m &> 0 \quad \text{if} \quad m \text{ is even,} \\ &< 0 \quad \text{if} \quad m \text{ is odd.} \quad \square \end{aligned}$$

This, of course, means that the terms in the asymptotic expansion alternate in sign and that the partial sums of the asymptotic expansion alternately over- and underestimate the true value of the integral in the following manner.

Proposition 8: For m even,

$$\sum_{i=1}^{m} g^{(i)}(0)/N^i \le \int_0^{\infty} e^{-Nf(z)}dz \le \sum_{i=1}^{m+1} g^{(i)}(0)/N^i. \quad \square$$

The above is quite useful since in most situations the designer would much rather overestimate than underestimate a measure such as CPU utilization. In this context, both the upper and lower bounds are required since ratios of integrals occur in the measures.

An implication of Proposition 6 is that $|g^{(m+1)}(\xi)| < |g^{(m+1)}(0)|$, $\xi > 0$, which together with (50) gives

Proposition 9:

$$\begin{aligned} R_m &< g^{(m+1)}(0)/N^{m+1} \quad \text{if} \quad m \text{ is even} \\ &> g^{(m+1)}(0)/N^{m+1} \quad \text{if} \quad m \text{ is odd.} \quad \square \end{aligned}$$

The above propositions thus state that the error is numerically less than the first neglected term of the series, and has the same sign. In particular, we have an explicit proof that (36) constitutes an asymptotic expansion. More generally, the above results show that, on account of the specialty of the integral, the main results that we require from an error analysis are already present in the expansion.

It is useful to examine in detail $g^{(4)}(0)$ and thence the bound on R_3:

$$|R_3| \le (b_4|\alpha|^2 + 10|\alpha|b_2 b_3 + 15b_2^3)/(|\alpha|^7 N^4), \tag{51}$$

where $b_i = (i - 1)! \sum \beta_j/\gamma_j^i$. (A look at the proof in Appendix B will convince the reader of the presence of $|\alpha|^{2m+1}\mathrm{N}^{m+1}$ in the denominator of the bound for $|R_m|$.) The bound does make the suggestion that in cases where α is so small [i.e., utilization is very large, see (45)] that $\alpha^2 N$ is itself small, then $|R_3|$ is large. More generally, in the case of small $\alpha^2 N$, the number of terms in the series requiring computation to meet specifications on the accuracy may be large. Later we return to consider this case further.

5.3 Bounds on mean values

The following bounds which supplement the computational procedures are presented to serve as aids in design and synthesis. For $i = 1, 2, \cdots, p$,

Proposition 10:

$$\left[\{u_i(\mathbf{K} + \mathbf{e}_i)\}^{-1} - \frac{\gamma_i}{\beta_i + 1/N}\right](\beta_i + 1/N)$$

$$\le \frac{1 + \sqrt{1 + 8f_2/\alpha^2 N}}{2|\alpha|N}, \tag{52}$$

$$\ge \frac{|\alpha|}{2f_2}\left(1 - \frac{2}{1 + \sqrt{1 + 16f_2/(\pi\alpha^2 N)}}\right). \quad \square \tag{53}$$

Recall that $2f_2 = f^{(2)}(0) = \sum \beta_j/\gamma_j^2$ while α (here $\alpha < 0$) and N are as given in (28) and (14).

To prove Proposition 10, we may use the representation in (22) for u_i in which case it remains to bound from above and below the pair of integrals appearing there. This is done in Appendix C by making use of the sign properties of the higher-order derivatives of the function $f(\cdot)$.

Notice that as $N \to \infty$, the upper bound in Proposition 10 on $\{u_i(\mathbf{K} + \mathbf{e}_i)\}^{-1}$ approaches $\{\gamma_i - 1/(\alpha N)\}/(\beta_i + 1/N)$, the sum of the leading *two* terms of the series for u_i in (43). Also, as $N \to \infty$, the expressions in (52) and (53) both approach 0.

VI. EXPANSIONS FOR THE CASE $\alpha \approx 0$

At the end of Section 5.2, we commented that the expansions given earlier may require a large number of terms in regions where $\alpha \approx 0$. For this case, we here generate a somewhat different and more efficient series. We shall, however, be quite brief in our exposition because the uniform asymptotic expansions to be derived in Section VIII also allow appropriate expansions to be obtained. The ad hoc but direct treatment here is supplementary.

We need notation that is specific to this section. Let

$$\left.\begin{aligned} K_i &= b_i N + d_i\sqrt{N} \\ r_i &= a_i(N + c\sqrt{N}) \end{aligned}\right\} \quad i = 1, 2, \cdots, p. \tag{54}$$

Each of the variables $\{d_i\}$ and c may be either positive or negative. However, as our interest in this section is in $\alpha \approx 0$ and as $\alpha \sim \sum b_i/a_i - 1$ as $N \to \infty$, we require that

$$\sum b_i/a_i = 1. \tag{55}$$

In a computational procedure the above restriction poses no particular problem.

Also, we shall mainly consider the integral

$$I \triangleq \left(\prod r_i^{-K_i}\right) \int_0^\infty e^{-t} \prod (r_i + t)^{K_i} dt. \tag{56}$$

Comparison with (16) shows that the integral is related to the partition function thus:

$$G(\mathbf{K}) = \left(\prod \frac{\rho_{i1}^{K_i}}{K_i!}\right) I. \tag{57}$$

As previously, the computation of the quantity in parentheses is not required to obtain the mean values. Our main result is

Proposition 11: As $N \rightarrow \infty$,

$$I \sim \sum_{j=0}^{\infty} c_j / \sqrt{N}^{j-1}, \tag{58}$$

where the sequence $\{c_j\}$ is given below. □

The proof of the proposition is in Appendix D. Here we comment on some features of the sequence:

$$c_j = \int_0^{\infty} e^{-A\nu^2/2 - B\nu} H_j(\nu) d\nu, \tag{59}$$

where $A = \sum b_i/a_i^2 > 0$ and $B = c - \sum d_i/a_i$ and $H_j(\nu)$ is a polynomial of degree $3j$ in ν with coefficients that are fairly straightforward to obtain. The key point is that A, B as well as the coefficients of the polynomials are all 0(1), i.e., N does not enter into their definitions. Results given in Section VIII indicate how the coefficients c_j given in (59) may be effectively computed.

We give below the leading three polynomials in the sequence $\{H_j(\nu)\}$:

$$\begin{aligned} H_0(\nu) &= 1, \\ H_1(\nu) &= c - (\textstyle\sum d_i/a_i^2)\nu^2/2 + (\sum b_i/a_i^3)\nu^3/3, \\ H_2(\nu) &= -c(\textstyle\sum d_i/a_i^2)\nu^2/2 + (\sum d_i/a_i^3 + c \sum b_i/a_i^3)\nu^3/3 \\ &\quad + [(\textstyle\sum d_i/a_i)^2 - 2 \sum b_i/a_i^4]\nu^4/8 \\ &\quad - (\textstyle\sum d_i/a_i^2)(\sum b_i/a_i^3)\nu^5/6 + (\sum b_i/a_i^3)^2 \nu^6/18. \end{aligned} \tag{60}$$

VII. ASYMPTOTIC EXPANSIONS FOR HEAVY TRAFFIC CONDITIONS

Here we obtain asymptotic expansions for the basic integral in (17) and (18) for the case $\alpha > 0$. For reasons similar to those discussed earlier for the case $\alpha < 0$, the expansions to use for $\alpha \approx 0$ are in Sections VI and VIII. Hence, the expansions given below are for exceptionally heavy traffic conditions, where α is not only positive but also not close to 0.

7.1 Laplace's method

The key difference from the treatment in Section 4.1 is the presence of the singularity at $\hat{z}$ (see Fig. 2), which will be assumed to be known. For large N the dominant contribution to the integral

$$I = \int_0^{\infty} e^{-Nf(z)} dz \tag{61}$$

comes from the neighborhood of $\hat{z}$. A Taylor series expansion around $\hat{z}$ gives

$$f(z) - \hat{f}_0 = \sum_{j=2}^{\infty} \hat{f}_j (z - \hat{z})^j, \tag{62}$$

where

$$\hat{f}_j = f^{(j)}(\hat{z})/j!, \qquad j = 0, 1, 2, \cdots. \tag{63}$$

In particular, for $j \geq 2$,

$$\hat{f}_j = (-1)^j \left[\sum_i \beta_i/(\gamma_i + \hat{z})^j \right] \Big/ j. \tag{64}$$

We make the following specific decomposition of I, which is convenient here and even more so in the error analysis to follow with z_2 as in Fig. 2,

$$e^{N\hat{f}_0} I = \int_0^{\hat{z}} e^{-N\{f(z)-\hat{f}_0\}} dz + \int_{\hat{z}}^{z_2} e^{-N\{f(z)-\hat{f}_0\}} dz + \int_{z_2}^{\infty} e^{-N\{f(z)-\hat{f}_0\}} dz. \tag{65}$$

Consider the terms in turn starting with the middle term. If we let

$$u \triangleq f(z) - \hat{f}_0, \qquad z \geq \hat{z}, \tag{66}$$

and use the series in (62) for the right-hand side, then we may reverse the series, as discussed in Appendix A, to obtain

$$z - \hat{z} \triangleq g(u) = \sum_{j=1}^{\infty} g_j u^{j/2} \tag{67}$$

with the coefficient g_j depending only on the coefficients $\hat{f}_2, \hat{f}_3, \cdots, \hat{f}_j$. Now,

$$\int_{\hat{z}}^{z_2} e^{-N\{f(z)-\hat{f}_0\}} dz = \int_0^{-\hat{f}_0} e^{-Nu} g^{(1)}(u) du = \int_0^{-\hat{f}_0} e^{-Nu} \sum_{j=0}^{\infty} a_j u^{(j-1)/2} du, \tag{68}$$

where $a_j = (j+1) g_{j+1}/2$. The individual integrals in the sum will be recognized to be incomplete gamma functions.

On returning to (65) and the first term in the right-hand side, we find by an identical argument that

$$\int_0^{\hat{z}} e^{-N\{f(z)-\hat{f}_0\}} dz = \int_0^{-\hat{f}_0} e^{-Nu} \sum_{j=0}^{\infty} (-1)^j a_j u^{(j-1)/2} du. \tag{69}$$

The two integrals in (68) and (69) may conveniently be combined to give

$$e^{N\hat{f}_0} I = \int_0^{-\hat{f}_0} e^{-Nu} \sum_{j=0}^{\infty} 2a_{2j} u^{j-1/2} du + \int_{z_2}^{\infty} e^{-N\{f(z)-\hat{f}_0\}} dz. \tag{70}$$

At this stage, the following two approximations are made, with their effects bounded in Section 7.2 in the course of the error analysis: the integration interval in the first term is extended to [0, ∞) and the second term is ignored. Nonetheless, the error analysis shows that

$$I \sim e^{-N\hat{f}_0} \int_0^{\infty} e^{-Nu} \sum_{j=0}^{\infty} 2a_{2j} u^{j-1/2} du, \tag{71}$$

giving Proposition 12.

Proposition 12: As $N \rightarrow \infty$,

$$I = \int_0^{\infty} e^{-Nf(z)} dz \sim e^{-N\hat{f}_0} \sum_{j=0}^{\infty} 2\Gamma(j + ½) a_{2j}/N^{j+1/2}. \quad \square \tag{72}$$

Recall that $\Gamma(½) = \sqrt{\pi}$ and for $j = 1, 2, \cdots$,

$$\Gamma(j + ½) = \sqrt{\pi} \textstyle\prod_{i=1}^{j} (i - ½).$$

We give the leading three coefficients:

$$\begin{aligned} a_0 &= (½)/\hat{f}_2^{1/2}, \\ a_2 &= (1/16)(15\hat{f}_3^2 - 12\hat{f}_2\hat{f}_4)/\hat{f}_2^{7/2}, \\ a_4 &= (5/256)(-64\hat{f}_2^3\hat{f}_6 + 224\hat{f}_2^2\hat{f}_3\hat{f}_5 + 112\hat{f}_2^2\hat{f}_4^2 \\ &\qquad - 504\hat{f}_2\hat{f}_3^2\hat{f}_4 + 231\hat{f}_3^4)/\hat{f}_2^{13/2}. \end{aligned} \tag{73}$$

The procedure for obtaining the asymptotic expansion for the integral

$$I^{(1)} = \int_0^{\infty} z e^{-Nf(z)} dz \tag{74}$$

is similar. Notice

$$(I^{(1)} - \hat{z}I) = e^{-N\hat{f}_0} \int_0^{\infty} (z - \hat{z}) e^{-N\{f(z)-\hat{f}_0\}} dz; \tag{75}$$

we find it convenient to expand the integral on the right-hand side. The expression analogous to (71) is

$$(I^{(1)} - \hat{z}I) \sim e^{-N\hat{f}_0} \int_0^\infty e^{-Nu} \sum_{j=1}^{\infty} 2a_{2j-1}^{(1)} u^{j-1/2} du, \tag{76}$$

where, for $j = 0, 1, 2, \cdots$,

$$a_j^{(1)} = \frac{1}{2} \sum_{k=1}^{j+1} k g_{j-k+2} g_k, \tag{77}$$

which is to be compared with the expression for a_j following (68). The following is a useful formula for efficiently generating the sequence $\{a_j^{(1)}\}$ from $\{a_j\}$, which is needed anyhow for computing I:

$$a_j^{(1)} = 2 \sum_{k=0}^{j} a_{j-k} a_k/(k+1), \qquad j = 0, 1, 2, \cdots. \tag{78}$$

Recognizing the gamma functions in (76) gives Proposition 13.

Proposition 13:

$$I^{(1)} - \hat{z}I \sim e^{-N\hat{f}_0} \sum_{j=1}^{\infty} 2\Gamma\left(j + \frac{1}{2}\right) a_{2j-1}^{(1)}/N^{j+1/2} \tag{79}$$

as $N \to \infty$. □

The asymptotic expansions in Propositions 12 and 13 may also be combined, as discussed earlier in Section 4.2 to yield an asymptotic expansion for the mean values. In particular, we obtain Proposition 14.

Proposition 14: As $N \to \infty$,

$$\{u_i(\mathbf{K} + \mathbf{e}_i)\}^{-1}(\beta_i + 1/N) \sim (\gamma_i + \hat{z}) + \frac{A_1}{N} + \frac{A_2}{N^2},$$

where

$$A_1 = -3\hat{f}_3/4\hat{f}_2^2,$$
$$A_2 = (6\hat{f}_2\hat{f}_3\hat{f}_4 - 15\hat{f}_2^2\hat{f}_5/8 - 135\hat{f}_3^2)/\hat{f}_2^5. \; \square \tag{80}$$

7.2 Error analysis

The analysis to be presented supplements the result in Proposition 12 and the error estimates to be given provide guidelines for the use of the expansions. The broad outline of the analysis have been suggested in Ref. 23.

As in Section 5.2, let R_n denote the error incurred when only n leading terms of the series in Proposition 12 is used, i.e.,

$$R_n = I - e^{-N\hat{f}_0} \sum_{j=0}^{n-1} \Gamma\left(j + \frac{1}{2}\right) 2a_{2j}/N^{j+1/2}. \tag{81}$$

For I we will use the expression given in (70) and decompose the error R_n thus

$$R_n = -\epsilon_{n,1}(N) + \epsilon_{n,2}(N) + \epsilon_3(N), \tag{82}$$

where

$$\epsilon_{n,1}(N) = e^{-N\hat{f}_0} \int_{-\hat{f}_0}^{\infty} e^{-Nu} \left(\sum_{j=0}^{n-1} 2a_{2j}u^{j-1/2}\right) du, \tag{83}$$

$$\epsilon_{n,2}(N) = e^{-N\hat{f}_0} \int_{0}^{-\hat{f}_0} e^{-Nu} \left(\sum_{j=n}^{\infty} 2a_{2j}u^{j-1/2}\right) du, \tag{84}$$

$$\epsilon_3(N) = \int_{z_2}^{\infty} e^{-Nf(z)}dz. \tag{85}$$

Thus, the three terms on the right-hand side of (82) respectively denote components arising from the extension of the integration interval from $[0, -\hat{f}_0]$ to $[0, \infty)$ in (70) and (71), the use of only n leading terms from the infinite series in (71), and the neglect of the second term in the right-hand side of (70). Each component is now bounded.

To bound $\epsilon_{n,1}(N)$, we make use of known bounds on the incomplete gamma function:[23]

$$\int_{-\hat{f}_0}^{\infty} e^{-Nu}u^{j-1/2}du = \Gamma\left(j + \frac{1}{2}, -N\hat{f}_0\right)\Big/N^{j+1/2}$$

$$\leq \frac{e^{N\hat{f}_0}(N|\hat{f}_0|)^{j+1/2}}{N^{j+1/2}\left\{N|\hat{f}_0| - \max\left(j - \frac{1}{2}, 0\right)\right\}} \quad \text{for} \quad n|\hat{f}_0| > \max\left(j - \frac{1}{2}, 0\right).$$

Thus,

$$|\epsilon_{n,1}(N)| \leq \frac{2}{N|\hat{f}_0| - \delta_n} \sum_{j=0}^{n-1} |a_{j2}|\,|\hat{f}_0|^{j+1/2}, \tag{86}$$

where $\delta_n = \max(n - 3/2, 0)$.

In bounding $\epsilon_{n,2}(N)$, we will postulate the existence of a finite valued σ_n with the property that

$$\left|\sum_{j=n}^{\infty} 2a_{2j}u^{j-1/2}\right| \leq |2a_{2n}|u^{n-1/2}e^{\sigma_n u}, \; 0 < u < |\hat{f}_0|. \tag{87}$$

This approach fails when σ_n is infinite but the characteristics of the integral and the specific decomposition (65) that has been employed preclude this possibility. Let

$$\sigma_n = \max_{|u|<|\hat{f}_0|} \psi_n(u), \tag{88}$$

where

$$\psi_n(u) = \frac{1}{u} \ln \left| \frac{\sum_{j=n}^{\infty} 2a_{2j}u^{j-1/2}}{2a_{2n}u^{n-1/2}} \right|. \tag{89}$$

Small u is where (see Ref. 23) the danger of unbounded σ_n is usually most manifest. However, as

$$\psi_n(u) \sim \frac{a_{2n+2}}{a_{2n}} + \left(\frac{a_{2n+4}}{a_{2n}} - \frac{a_{2n+2}^2}{a_{2n}^2} \right) u + \cdots \tag{90}$$

when $u \to 0^+$, no problem arises here.

Using (87) in the defining expression for $\epsilon_{n,2}(N)$ in (84)

$$\begin{aligned} |\epsilon_{n,2}(N)| &\le e^{-N\hat{f}_0} |2a_{2n}| \int_0^{|\hat{f}_0|} e^{-(N-\sigma_n)u}\, u^{n-1/2} du \\ &= e^{-N\hat{f}_0} |2a_{2n}| \Gamma(n + ½)/(N - \sigma_n)^{n+1/2}. \end{aligned} \tag{91}$$

The last term to be considered from (82) is $\epsilon_3(N)$. We use the following property.

$$f(z) \ge f'(z_2)(z - z_2), \qquad z \ge z_2, \tag{92}$$

which yields

$$|\epsilon_3(N)| \le \frac{1}{Nf'(z_2)}, \tag{93}$$

a small quantity compared to the right-hand side of (91). This concludes the process of bounding the components of the error term R_n. A corollary is the proof to Proposition 12.

The bound in (91) is the largest component in the error bound. In examining (91), we observe that the condition in which the bound is large is when a_{2n}/N^n is large. Now the expression for a_{2n} contains in the denominator a term $\hat{f}_2^{3n+1/2}$, as (73) attests. Thus, when $\hat{f}_2^3 N$ is small, we expect the asymptotic expansions in Section 7.1 to be inefficient. We return to this case later in Section 8.2, where this as well as the similar difficulty encountered in Section 5.2—where α was negative and small—is treated in a unified manner.

VIII. UNIFORM EXPANSIONS

This section has two objectives. The first is to show that there is a framework that unifies the expansions in Sections IV, VI, and VII.

This consists of showing that the integrals of interest may each be given by a common expansion valid uniformly for the entire range of values of the system parameters. These expansions turn out to be in parabolic cylinder (or Weber) functions.[13,25,26] The advantage derived is that these classical special functions have extensively documented expansions for the entire range of parameter values.[13,26] Indeed, using these expansions we sketch in Sections 8.3 and 8.4, at the cost of some duplication, derivations of the expansions obtained earlier in Sections IV and VII for $\alpha < 0$ and $\alpha > 0$, respectively. The second objective is to derive a computationally efficient expansion for the case $\alpha \approx 0$, i.e., where the stationary point $\hat{z}$ is very close to the boundary of the integration interval. The error analysis in Sections 5.2 and 7.2 has shown the need for a separate treatment. The expansion that is obtained for this case in Section 8.2 is obtained from an appropriate expansion of the parabolic cylinder functions.

8.1 Uniform expansions in parabolic cylinder functions

Consider the integral

$$I = \int_0^\infty e^{-Nf(z)} dz \tag{94}$$

without restrictions on the parameter α. Following Friedman,[24] consider a change of variables from z to v given by

$$v^2 - 2av = f(z), \tag{95}$$

where a is a parameter of the transformation to be fixed later. The objective of the transformation is that the component of the integrand in braces below

$$I = \int_0^\infty e^{-N(v^2-2av)} \left(\frac{dz}{dv}\right) dv \tag{96}$$

satisfy the dual requirements of boundedness and a convergent power series, as required for an application of Watson's lemma (see Section 4.1 following Proposition 3). [The reader may verify that the simpler transformation $v = f(z)$ violates the boundedness requirement whenever $\alpha > 0$ and $z = \hat{z}$ since $f^{(1)}(\hat{z}) = 0$.] For the transformation in (95),

$$\frac{dz}{dv} = \frac{2(v-a)}{f^{(1)}(z)}. \tag{97}$$

This suggests the selection of the parameter a to be such that $v = a$ when $z = \hat{z}$, with the accompanying indeterminacy and the possibility of boundedness of dz/dv. This key clue does indeed give a unique map of the form in (95) with the desired properties, as summarized below.

Proposition 15: For $z \geq 0$, let

$$v(z) = a + \operatorname{sgn}(z - \hat{z})\sqrt{f(z) - f(\hat{z})}, \tag{98}$$

where the constant a depends on all the system parameters:

$$a = (\operatorname{sgn} \alpha)\sqrt{-f(\hat{z})}. \tag{99}$$

The transformation is monotonic, increasing and maps $[0, \infty)$ to $[0, \infty)$. Also dv/dz is continuous and uniformly bounded. □

The transformation is used to derive for dz/dv a convergent power series $\sum_0^\infty C_j v^j$ in a neighborhood of the origin. This is achieved in three steps:

(*i*) Use (98) to obtain

$$v(z) = A_1 z + A_2 z^2 + A_3 z^3 + \cdots . \tag{100}$$

(*ii*) Reverse the series (see Appendix A) to obtain

$$z(v) = B_1 v + B_2 v^2 + B_3 v^3 + \cdots . \tag{101}$$

(*iii*) Differentiate term by term to obtain

$$\frac{dz}{dv} = C_0 + C_1 v + C_2 v^2 + \cdots , \tag{102}$$

where $C_j = (j + 1)B_{j+1}$.

The reader may verify that the leading terms of the sequence $\{C_j\}$ thus obtained are as follows [recall from (33) the definition $f_j = f^{(j)}(0)/j!$]:

$$\begin{aligned} C_0 &= 2a/\alpha, \\ C_1 &= \frac{2}{\alpha}\left[\left(\frac{2a}{\alpha}\right)^2 f_2 - 1\right]. \end{aligned} \tag{103}$$

The above tacitly assumes that $\hat{z}$ and hence a have been evaluated.

The power series expansion for dz/dv may now be substituted in (96) to yield

$$I = \sum_{j=0}^{\infty} C_j \int_0^\infty e^{-N(v^2 - 2av)} v^j dv. \tag{104}$$

The integrals appearing above are simply related to the parabolic cylinder functions $U(\cdot, \cdot)$; thus: for $j = 0, 1, 2, \cdots$,

$$\int_0^\infty e^{-N(v^2-2av)} v^j dv = \frac{e^{Na^2/2} j!}{(2N)^{(j+1)/2}} U\left(j + \frac{1}{2}, -a\sqrt{2N}\right). \tag{105}$$

Expansions for related integrals such as

$$I^{(1)} = \int_0^\infty ze^{-Nf(z)}dz \tag{106}$$

are only slightly different. Here the term dz/dv is replaced by zdz/dv, which has the power series expansion

$$zdz/dv = \sum_{j=1}^{\infty} C_j^{(1)} v^j, \tag{107}$$

where, see (101),

$$C_j^{(1)} = \sum_{k=0}^{j-1} B_{j-k} C_k. \tag{108}$$

Specifically,

$$C_1^{(1)} = \left(\frac{2a}{\alpha}\right)^2,$$

$$C_2^{(1)} = \frac{3}{\alpha}\left(\frac{2a}{\alpha}\right)\left[\left(\frac{2a}{\alpha}\right)^2 f_2 - 1\right]. \tag{109}$$

The following summarizes the expansions in parabolic cylinder functions of the two integrals of main interest, with the expansion valid for all α.

Proposition 16:

$$\begin{aligned} I &= \int_0^\infty e^{-Nf(z)}dz = \sum_{j=0}^{\infty} C_j \int_0^\infty e^{-N(v^2-2av)} v^j dv \\ &= e^{Na^2/2} \sum_{j=0}^{\infty} \frac{j!C_j}{(2N)^{(j+1)/2}} U\left(j+\frac{1}{2}, -a\sqrt{2N}\right), \end{aligned} \tag{110}$$

$$\begin{aligned} I^{(1)} &= \int_0^\infty ze^{-Nf(z)}dz \\ &= e^{Na^2/2} \sum_{j=1}^{\infty} \frac{j!C_j^{(1)}}{(2N)^{(j+1)/2}} U\left(j+\frac{1}{2}, -a\sqrt{2N}\right), \end{aligned} \tag{111}$$

where the sequences $\{C_j\}$ and $\{C_j^{(1)}\}$ are respectively as obtained by the procedures in (100) through (102) and (108). Specifically the leading terms are as given in (103) and (109). □

We should add that the above expansions are not strictly asymptotic expansions since the parabolic cylinder functions do not satisfy the requirements of asymptotic sequences for certain ranges of the parameter a^2N.[14] The interested reader will find in Ref. 27 a description of the process for obtaining uniform asymptotic expansions of the integrals. However, we have not found it necessary to undertake the additional effort required to obtain the coefficients of the uniform asymptotic expansions. This is because for a^2N small, the case treated below in Section 8.2 and of main interest, the functions $U(j + ½, -a\sqrt{2N})$, $j \geq 0$, have all the desirable properties that are required of asymptotic sequences.

A noteworthy property of the functions $U(\cdot, \cdot)$ that can be important in computations is that it satisfy the recursion[13]

$$xU(j + ½, x) = U(j - ½, x) - (j + 1)U(j + 3/2, x). \tag{112}$$

8.2 Expansions for the case of a^2N small

To motivate the results to be given here, observe that the stationary point of the curve of $f(z)$ (see Fig. 2) is close to 0 when α is small (utilization of CPU is high), since

$$\hat{z} \sim \alpha/(2f_2) \qquad \text{as} \qquad \hat{z} \to 0. \tag{113}$$

On enquiring how the parameter a behaves for small $\hat{z}$ and α, we find from (99) that

$$a = \frac{\alpha}{2(f_2)^{1/2}} + O(\alpha^2) \qquad \text{as} \qquad \alpha \to 0. \tag{114}$$

Thus, the case of small α, which we know from Section 5.2 requires special treatment, corresponds to small a.

Small a is also implied by small $\hat{f}_2$ and is therefore also of interest on account of the discussion in Section 7.2. This follows from

$$a = \hat{z}(\hat{f}_2)^{1/2} + O(\hat{z}^{3/2}) \qquad \text{as} \qquad \hat{z} \to 0. \tag{115}$$

For small a^2N, it is known[13] that for $j \geq 0$,

$$\int_0^\infty e^{-N(v^2-2av)}v^j dv = \frac{e^{Na^2/2}j!}{(2N)^{(j+1)/2}}\, U\left(j + \frac{1}{2}, -a\sqrt{2N}\right)$$
$$= \frac{\sqrt{\pi}}{N^{(j+1)/2}}\left[\mu_0^{(j)} + (a\sqrt{N})\mu_1^{(j)} + (a\sqrt{N})^2\mu_2^{(j)} + \cdots\right], \tag{116}$$

where

$$\mu_i^{(j)} = \frac{j!}{i!}\,\frac{2^{i/2-j-1}}{\Gamma(1 + j/2)}\,(j + 1)(j + 3) \cdots (j + i - 1), \qquad i \text{ even}$$
$$= \frac{j!}{i!}\,\frac{2^{(i+1)/2-j-1}}{\Gamma((j + 1)/2)}\,(j + 2)(j + 4) \cdots (j + i - 1), \qquad i \text{ odd}. \tag{117}$$

In these relations, notice first in (116) the desirable presence of the powers of N in the denominator. Secondly, in connection with the sequence $\{\mu_i^{(j)}\}$, observe that $\mu_{i+2}^{(j)}/\mu_i^{(j)} = 2(j+i+1)/\{(i+1)(i+2)\}$. Thus, for fixed j, the sequence converges rapidly to 0 with increasing i. These observations state that when using the expression (116) in the expansions of Proposition 16, first, it is necessary to compute only up to small values of the index j and, secondly, with $a\sqrt{N}$ small, the computation of the bracketed quantity in (116) also needs very few terms.

The following summarizes the important computational procedure described above.

Proposition 17: For small a^2N,

$$I = \sqrt{\pi} \sum_{j\geq 0} \frac{C_j}{N^{(j+1)/2}} \left[\sum_{i\geq 0} (a\sqrt{N})^i \mu_i^{(j)} \right], \tag{118}$$

$$I^{(1)} = \sqrt{\pi} \sum_{j\geq 1} \frac{C_j^{(1)}}{N^{(j+1)/2}} \left[\sum_{i\geq 0} (a\sqrt{N})^i \mu_i^{(j)} \right], \tag{119}$$

where $\mu_i^{(j)}$ is in (117), and a, $\{C_j\}$, $\{C_j^{(1)}\}$ appear in Proposition 16. □

8.3 Expansions for normal traffic

For normal traffic, $\alpha < 0$ and consequently $a < 0$. If in addition, $\alpha \ll 0$ or, specifically, $a^2N \gg j^2$, then[13]

$$\int_0^\infty e^{-N(v^2-2av)} v^j dv$$
$$\sim \frac{1}{(-2aN)^{j+1}} \left(j! - \frac{(j+2)!}{4(a^2N)} + \frac{(j+4)!}{32(a^2N)^2} \mp \cdots \right). \tag{120}$$

It can be shown after some manipulations, which we omit, that this expansion when substituted in Proposition 16 is identical to the main result of Section IV, namely, the expansion in Proposition 3. The bridging relation is

$$(v-a)^2 = u + a^2, \tag{121}$$

where u is the integration variable in Section IV [see (31)] and v is the similar variable in the uniform expansion [see (98)].

8.4 Expansions for heavy traffic

For heavy traffic, $\alpha > 0$, and therefore $a > 0$. When $a^2N \gg 1$ as well, then[13]

$$\int_0^\infty e^{-N(v^2-2av)} v^j dv = e^{Na^2} \frac{\sqrt{\pi}}{\sqrt{N}} a^j \left(1 + \frac{j(j-1)}{4a^2N} \right.$$
$$\left. + \frac{j(j-1)(j-2)(j-3)}{32a^4N^2} + \cdots \right). \tag{122}$$

Notice the departure from the expressions in (116) and (120) in the absence of N^j in the denominator.

It may again be established, although not in a simple manner, that the main result of Section VII, the expansion in Proposition 12, is also obtained by substituting the above expansion in Proposition 16. The bridging relation is

$$(v - a)^2 = u, \tag{123}$$

where the variable u is as in (66).

IX. COMPUTATIONAL NOTES

The asymptotic expansions of integral representations of various quantities in the closed queueing networks discussed above, have been implemented as a user-oriented interactive package on Digital Equipment Corporation's VAX 11/780 operating under programmers work bench *UNIX*[*] system version 3. The package written in C-language has about 400 C-language statements and occupies about 60 Kbytes of storage.

The number of classes and constituents[†] that the package can accommodate is so large that in effect no restriction is placed on these parameters. The other main features of the package are enumerated below.

(*i*) The package is user oriented and easy to use. The user is prompted for relevant problem data. As this is supplied, validation and feasibility checks are made and the user informed of any errors.

(*ii*) The output of the package includes all relevant statistics on each class (response time, utilization, etc.) including the percentage error incurred in the expansion. As an option to the last named, the user can display the various terms used in the expansion.

(*iii*) The package is partly adaptive in that it automatically detects the divergence of the asymptotic expansion and truncates the series at the point of divergence.

(*iv*) Numerical stability is enforced by proper choice of N.

Numerous computational experiments were performed to compare the efficacy of our package, called ASYM, to the current version of a popular commercially available package CADS. CADS is marketed by Information Research Associates and a version of it runs on a VAX 11/780 operating under *UNIX* time sharing system. The test problems run on both these packages are real-world problems encountered in performance analysis of a Bell System project.

[*] Trademark of Bell Laboratories.

[†] In the computer science applications, the class sizes in closed networks $\{K_j\}$ are referred to as degrees of multiprogramming.

The results of the experiments indicate that CADS is unable to solve our moderately sized, closed-queuing-network problems. Two features that accompany the breakdowns are noteworthy. One is numerical instability which manifests itself as overflows or underflows. Recent experience is that rescaling the service rates, an often-mentioned device to combat the problem, does not help in substantially increasing the problem size. This accuracy problem is of course relieved with the use of more powerful floating-point machines like the CDC 6600 (manufactured by Control Data Corporation) or IBM 3033. The other feature is the built-in limitations of the package. For instance, the current version of CADS limits the degree of multiprogramming for any one class to 100. There is also a limit of three classes.

The above discussion is not intended to disparage the usefulness and success of CADS. CADS is extremely powerful in solving *small* queueing-network problems. Packages implementing recursions, as CADS does, and implementing expansions of integrals complement each other and, when integrated, will provide a powerful general-purpose package.

The computational experiments (see Tables I to IV below) yielded the interesting fact that the asymptotic expansions are quite effective (i.e., yield a small percentage of errors) even for *small* problems, provided the right expansion is used. This in conjunction with the linear growth in computation time with increased number of classes

Table I—Results for test problem No. 1 *

Degree of Multiprogramming	Utilization of CPU Given by CADS	Utilization of CPU Given by ASYM
10	0.0417	0.0414
20	0.083	0.0829
30	0.124	0.124
40	0.166	0.165
50	0.207	0.207
60	0.249	0.248
70	0.290	0.289
80	Breakdown	0.331
90	Breakdown	0.372
100	Breakdown	0.413
110	Breakdown	0.618
150	Breakdown	0.618
200	Breakdown	0.819

* Problem specification:
No. of classes = 1
Think time = 240 seconds
Processing time = 1 second

Table II—Results for test problem No. 2*

Degrees of Multiprogramming	Total Utilization of CPU Given by CADS	Total Utilization of CPU Given by ASYM
10/10	0.118	0.119
20/20	0.239	0.23
30/30	0.358	0.35
40/40	0.476	0.48
50/50	0.593	0.60
60/60	Breakdown	0.70
80/60	Breakdown	0.72
90/60	Breakdown	0.97
100/50	Breakdown	0.69
110/50	Breakdown	0.71
140/50	Breakdown	0.79
200/10	Breakdown	0.54
170/40	Breakdown	0.75

* Problem specification:
No. of classes = 2
Think time for class 1 = 450 seconds
Think time for class 2 = 150 seconds
Processing time for class 1 = 1 second
Processing time for class 2 = 1.5 seconds

Table III(a)—Problem specification for test problem 3

Class	Service Rate of Infinite Server for This Class	Service Rate of CPU for This Class	Degree of Multiprogramming for This Class
1	0.0001	6.00	500
2	0.0005	2.00	500
3	0.0006	4.00	500
4	0.0002	0.33	500
5	0.0002	0.60	500
6	0.00005	0.20	500
7	0.00005	1.00	500

Table III(b)—Results of test problem 3 output by ASYM

Class	Response Time (seconds)	Utilization	% Error in Utilization
1	0.95	0.01	0.0025
2	2.85	0.12	0.0025
3	1.42	0.08	0.0025
4	17.21	0.3	0.0025
5	9.48	0.17	0.0025
6	28.46	0.13	0.0025
7	5.70	0.02	0.0025

Table IV(a)—Problem specification for test problem 4

Class	Service Rate of Infinite Server for This Class	Service Rate of CPU for This Class	Degree of Multiprogramming for This Class
1	0.0033	20.0	5
2	0.033	2	5
3	0.0033	4	5
4	0.033	4	5
5	0.033	4	5
6	0.033	6	5
7	0.033	20	5
8	0.00033	0.6	5
9	0.00055	0.6	5
10	0.0033	0.6	5
11	0.00033	0.2	5
12	0.00055	0.2	5
13	0.0003	0.2	5
14	0.033	1	5
15	0.00033	1	5
16	0.00055	1	5
17	0.0003	1	5

Table IV(b)—Results of test problem 4 output by ASYM

Class	Response Time (seconds)	Utilization	% Error in Utilization
1	0.09	0.008	0.05
2	0.834	0.080	0.045
3	0.43	0.004	0.05
4	0.42	0.046	0.048
5	0.42	0.040	0.049
6	0.28	0.027	0.049
7	0.09	0.008	0.05
8	2.85	0.003	0.06
9	2.9	0.005	0.05
10	2.82	0.027	0.05
11	8.530	0.008	0.05
12	8.523	0.013	0.05
13	8.540	0.007	0.05
14	1.63	0.156	0.04
15	1.71	0.002	0.05
16	1.72	0.003	0.05
17	1.71	0.001	0.05

makes the use of the asymptotic expansions attractive even in cases where the recursive implementation does not break down.

Tables I and II display the results of problems run both on CADS and ASYM. Tables III and IV show the results on two large problems that were not admitted by CADS, but were solved with good accuracy by ASYM.

It may be observed from Table I that the results from ASYM and CADS agree rather well, even for small degrees of multiprogramming. In the cases solved by both CADS and ASYM, α was small and N (about 240) large. Observe also that CADS was unable to solve cases with K_1 larger than 70, even though higher values of K_1 correspond to quite low usage of the CPU and are quite interesting.

Table II shows that CADS also broke down on a relatively small problem with 2 classes and 60 customers in each class.

Tables III and IV display the output given by ASYM for the large problems. Table III corresponds to a problem where the total degree of multiprogramming is 3500. For Table IV the total degree of multiprogramming is 85, but there are 17 classes. As shown, the percentage error in both cases is well within an acceptable range.

In conclusion, the computational experiments suggest that the approach based on expansions of integral representations is robust, computationally fast, and to be recommended for a variety of problems, large and small.

X. GENERALIZATIONS: INTEGRALS IN NETWORKS WITH MANY PROCESSORS

This section shows that for a large class of closed Markovian networks, the partition functions possess simple integral representations. This result provides the basis for future work on the computations of the integrals from expansions rather like those given in this paper. The above-mentioned class of networks allows an arbitrary number of service centers with flexibility in operating disciplines. It is in fact the same class of networks shown by Baskett et al., in Ref. 4, that has the product form in the stationary distributions, except that we do not allow the service rate in Type 1 centers to depend on the number in queue. (To some extent this is only for convenience because for some specific and interesting dependencies, we have obtained the integral representations.)

The representation of the partition function that is obtained is as a multiple integral, i.e., as an integral in Euclidean space of dimension q where q is the number of *queuing* centers, which are the centers of Type 1, 2, and 4 in the notation of Ref. 4. However, in spite of the complexity of the partition function, the form of the integrand is remarkable for its simplicity.

10.1 Background on product form solutions

As previously, we let the number of classes of constituents be p. We henceforth consistently index the classes by the symbol j; when the index for summation or multiplication is omitted, it is understood that the missing index is j, where $1 \leq j \leq p$. A total of s service centers are allowed. We will find it natural to distinguish the centers of Types 1, 2, and 4, which have queuing, from the remaining centers of Type 3, which do not. Thus, centers 1 through q will be the queuing centers while $(q + 1)$ through s will be the Type 3 centers, which have also been called think nodes and infinite server nodes.

Let the

$$\text{stationary state probability} = \pi(\mathbf{y}_1, \mathbf{y}_2, \cdots, \mathbf{y}_s), \tag{124}$$

$$\mathbf{y}_i \triangleq (n_{1i}, n_{2i}, \cdots, n_{pi}),\ 1 \leq i \leq s,$$

$$n_{ji} \triangleq \text{number of class—}j \text{ jobs in center } i.$$

The well-known results on Markovian closed queuing networks with product form solutions may be given in the following form:[1,4]

$$\pi(\mathbf{y}_1, \cdots, \mathbf{y}_s) = \frac{1}{G} \prod_{i=1}^{s} \pi_i(\mathbf{y}_i), \tag{125}$$

$$\begin{aligned} \text{where } \pi_i(\mathbf{y}_i) &= (\textstyle\sum n_{ji})! \prod \left(\frac{\rho_{ji}^{n_{ji}}}{n_{ji}!} \right), \qquad 1 \leq i \leq q, \\ &= \prod \left(\frac{\rho_{ji}^{n_{ji}}}{n_{ji}!} \right), \qquad (q+1) \leq i \leq s. \end{aligned} \tag{126}$$

In the above formulas we have taken into account the previously stated assumption, namely, for the first-come-first-served discipline in Type 1 centers the service rate is independent of the number of jobs in queue. Also, in (126),

$$\rho_{ji} = \frac{\text{expected number of visits of class } j \text{ jobs to center } i}{\text{service rate of class } j \text{ jobs in center } i},$$

where the numerator is obtained from the given routing matrix by solving for the eigenvector corresponding to the eigenvalue at 1.

In (125) G is, of course, the partition function and it is explicitly

$$G = \sum_{\mathbf{1}'\mathbf{n}_1 = K_1} \cdots \sum_{\mathbf{1}'\mathbf{n}_p = K_p} \prod_{i=1}^{s} \pi_i(\mathbf{y}_i), \tag{127}$$

where we have written $\mathbf{1}'\mathbf{n}_j$ for $\sum_{i=1}^{s} n_{ji}$ and the condition $\mathbf{1}'\mathbf{n}_j = K_j$ to indicate the conservation of jobs in each class. Thus,

$$G = \sum \cdots \sum \left[\prod_{i=1}^{q} \left((\sum n_{ji})! \prod \frac{\rho_{ji}^{n_{ji}}}{n_{ji}!} \right) \right] \left[\prod_{i=q+1}^{s} \left(\prod \frac{\rho_{ji}^{n_{ji}}}{n_{ji}!} \right) \right]. \qquad (128)$$

10.2 Integral representation

Using Euler's integral, see (10),

$$G = \int_0^\infty \cdots \int_0^\infty \exp\left(- \sum_{i=1}^{q} u_i \right) \sum_{1'\mathbf{n}_1 = K_1} \cdots \sum_{1'\mathbf{n}_p = K_p} \left[\prod_{i=1}^{q} \left(\prod \frac{(\rho_{ji} u_i)^{n_{ji}}}{n_{ji}!} \right) \right]$$

$$\cdot \left[\prod_{i=q+1}^{s} \left(\prod \frac{\rho_{ji}^{n_{ji}}}{n_{ji}!} \right) \right] du_1 \cdots du_q. \qquad (129)$$

Now by the multinomial theorem,

$$G = (\prod K_j!)^{-1} \int_0^\infty \cdots \int_0^\infty \exp\left(- \sum_{i=1}^{q} u_i \right)$$

$$\cdot \prod \left(\sum_{i=1}^{q} \rho_{ji} u_i + \sum_{i=q+1}^{s} \rho_{ji} \right)^{K_j} du_1 \cdots du_q. \qquad (130)$$

It is noteworthy, but not surprising, that the parameters ρ_{ji} for all the Type 3 centers appear lumped together.

We now introduce the large parameter N and define

$$\beta_j = \frac{K_j}{N}, \qquad 1 \le j \le p, \qquad (131)$$

exactly as in (12). However, we define

$$\gamma_{ji} = \left(\frac{\rho_{ji}}{\sum\limits_{m=q+1}^{s} \rho_{jm}} \right) N, \qquad 1 \le j \le p, \quad 1 \le i \le q, \qquad (132)$$

which is reciprocal to the natural extension of the parameters $\{\gamma_j\}$ defined in (13). In common with Section 3.1, the suggestion in the notation is that in the generic large network $\{\gamma_{ji}\}$ and $\{\beta_j\}$ are $O(1)$. A tacit assumption being made is that all job classes are routed through at least one Type 3 (infinite server) center.

On substituting (131) and (132) in (130) and after a change of variables we obtain a form for the partition function, which is summarized below.

Proposition 18:

$$G = \left(N^q \prod_{j=1}^{p} \left[\left(\sum_{i=q+1}^{s} \rho_{ji} \right)^{K_j} / K_j! \right] \right) \int_0^\infty \cdots \int_0^\infty e^{-Nf(\mathbf{z})} d\mathbf{z}, \qquad (133)$$

where

$$\mathbf{z} = [z_1, z_2, \cdots, z_q]',$$

$$f(\mathbf{z}) = \mathbf{1}'\mathbf{z} - \sum_{j=1}^{p} \beta_j \log(1 + \Gamma_j'\mathbf{z}),$$

$$\mathbf{1} = [1, 1, \cdots, 1]',$$

$$\Gamma_j = [\gamma_{j1}, \gamma_{j2}, \cdots, \gamma_{jq}]', \qquad 1 \le j \le p. \qquad \square$$

As before, the term in brackets in (133) will typically not be required to be computed.

Future work will consider the expansions appropriate for the computation of the integral in (133).

APPENDIX A

Notation for Asymptotic Expansion, Series Reversion

Asymptotic expansion: A series $\sum_{j=0}^{\infty} a_j/N^j$ is said to be an asymptotic expansion[14,22,23] of a function $F(N)$ if

$$F(N) - \sum_{j=0}^{n-1} a_j/N^j = 0(N^{-n}) \qquad \text{as} \qquad N \to \infty$$

for every $n = 1, 2, \cdots$. We write

$$F(N) \sim \sum_{j=0}^{\infty} a_j/N^j.$$

The series itself may be either convergent or divergent.

Series reversion: If $u = f(z)$, $u_0 = f(z_0)$, $f'(z_0) \neq 0$, then by Lagrange's expansion[13,25]

$$z = z_0 + \sum_{j=1}^{\infty} \frac{(u - u_0)^j}{j!} \left[\frac{d^{j-1}}{dz^{j-1}} \left(\frac{z - z_0}{f(z) - u_0} \right)^j \right]_{z=z_0} . \tag{134}$$

In particular, if $f(\cdot)$ is specified in a Taylor series, the above expansion identifies the coefficients $\{g_j\}$ in the reversed power series $z - z_0 = \sum_{j=1}^{\infty} g_j(u - u_0)^j$. We specifically identify the leading coefficients of two reversed series that have been used in Section 4.1 and Section 7.1.
If

$$u = f_1 z + f_2 z^2 + \cdots, \tag{135}$$

then

$$z = g_1 u + g_2 u^2 + \cdots, \tag{136}$$

where

$$f_1 g_1 = 1,$$

$$\begin{aligned}
f_1^3 g_2 &= -f_2,\\
f_1^5 g_3 &= 2f_2^2 - f_1 f_3,\\
f_1^7 g_4 &= 5f_1 f_2 f_3 - f_1^2 f_4 - 5f_2^3,\\
f_1^9 g_5 &= 6f_1^2 f_2 f_4 + 3f_1^2 f_3^2 + 14f_2^4 - f_1^3 f_5 - 21 f_1 f_2^2 f_3.
\end{aligned}$$

Similarly, if

$$u = f_2 z^2 + f_3 z^3 + f_4 z^4 + \cdots , \tag{137}$$

then

$$z = g_1 u^{1/2} + g_2 u + g_3 u^{3/2} + \cdots , \tag{138}$$

where

$$\begin{aligned}
f_2^{1/2} g_1 &= 1,\\
f_2^2 g_2 &= -f_3/2,\\
f_2^{7/2} g_3 &= (5f_3^2 - 4f_2 f_4)/8,\\
f_2^5 g_4 &= (3f_2 f_3 f_4 - f_2^2 f_5 - 2f_3^3)/2,\\
f_2^{13/2} g_5 &= (-64 f_2^3 f_6 + 224 f_2^2 f_3 f_5 + 112 f_2^2 f_4^2 - 504 f_2 f_3^2 f_4 + 231 f_3^4)/128.
\end{aligned}$$

APPENDIX B

Proof of Proposition 6

Before giving the proof, it is worthwhile to generate expressions for the leading derivatives of $g(\cdot)$. In the notation of Section 4.1, $u = f(z)$ and $z = g(u)$,

$$\begin{aligned}
g^{(1)}(u) &= \frac{1}{f^{(1)}(z)},\\
g^{(2)}(u) &= -f^{(2)}(z)/(f^{(1)}(z))^3,\\
g^{(3)}(u) &= [-f^{(3)}(z) f^{(1)}(z) + 3(f^{(2)}(z))^2]/(f^{(1)}(z))^5,\\
g^{(4)}(u) &= [-f^{(4)}(z)(f^{(1)}(z))^2 + 10 f^{(1)}(z) f^{(2)}(z) f^{(3)}(z)\\
&\quad - 15(f^{(2)}(z))^3]/(f^{(1)}(z))^7.
\end{aligned} \tag{139}$$

For notational convenience, let γ_i and ϕ_i denote, in this appendix only, $g^{(i)}(u)$ and $f^{(i)}(z)$ respectively. Recall that $\phi_1 > 0$ and that $(-1)^i \phi_i > 0$ for $i = 2, 3, \cdots$. We will show by induction that $(-1)^n \gamma_n < 0$, $n = 1, 2, 3, \cdots$.

Let the induction hypothesis be the following

$$n \text{ even: } \phi_1^{2n-1} \gamma_n = \cdots - (\phi_1)^{2i}(X) + (\phi_1)^{2i+1}(Y) \cdots , \tag{140}$$

where $0 \le i$; $\max\,[2i, 2i+1] < 2n - 1$; X and Y are arbitrary products of $\phi_2, \phi_3, \cdots, \phi_n$ with arbitrary positive numerical weight and the property that $X > 0$, $Y < 0$ for all $z \ge 0$.

$$n \text{ odd: } \phi_1^{2n-1}\gamma_n = \cdots + (\phi_1)^{2i}(U) - (\phi_1)^{2i+1}(V) \cdots , \tag{141}$$

where U and V are like X and Y including that $U > 0$, $V < 0$ for all $z \geq 0$.

For the proof, take the case of n even, first. A key observation is that since ϕ_1 does not occur in either X or Y, $dX/dz < 0$ and $dY/dz > 0$. That is, in common with the functions ϕ_2, ϕ_3, $\cdots$ the derivative has opposite sign from the function. Also, from (140)

$$\gamma_n = \cdots - \frac{X}{\phi_1^{2n-2i-1}} + \frac{Y}{\phi_1^{2n-2i-2}} \cdots . \tag{142}$$

Differentiate with respect to u,

$$\gamma_{n+1} = \cdots - \frac{1}{\phi_1}\left(\frac{-X(2n-2i-1)\phi_2}{\phi_1^{2n-2i}} + \frac{X'}{\phi_1^{2n-2i-1}}\right)$$
$$+ \frac{1}{\phi_1}\left(\frac{-Y(2n-2i-2)\phi_2}{\phi_1^{2n-2i-1}} + \frac{Y'}{\phi_1^{2n-2i-2}}\right) + \cdots .$$

Hence,

$$\phi_1^{2n+1}\gamma_{n+1} = \cdots + (\phi_1)^{2i}(X(2n-2i-1)\phi_2)$$
$$- (\phi_1)^{2i+1}(X' + Y(2n-2i-2)\phi_2) + (\phi_i)^{2i+2}(Y') \cdots \tag{143}$$

which has the form appearing in (141) as part of the hypothesis.

Now take the case of n odd where, from (141),

$$\gamma_n = \cdots + \frac{U}{\phi_1^{2n-2i-1}} - \frac{V}{\phi_1^{2n-2j-2}} \cdots . \tag{144}$$

Differentiate and rearrange to obtain,

$$\phi_1^{2n+1}\gamma_{n+1} = \cdots - (\phi_1)^{2i}(U(2n-2i-1)\phi_2)$$
$$+ (\phi_1)^{2i+1}(U' + V(2n-2j-2)\phi_2) - (\phi_1)^{2i+2}(V') \cdots .$$

As $U' < 0$, $V' > 0$, the form in (140) of the induction hypothesis is satisfied. This concludes the proof.

APPENDIX C

Proof of Proposition 10

In the following we shall need the sign property of the derivatives of $f(\cdot)$ as given in (25) and (26). Also recall $f_1 = f'(0) = -\alpha$

$$\int_0^\infty ze^{-Nf(z)}dz \le \int_0^\infty ze^{-Nf_1z} = 1(N\alpha)^2 \tag{145}$$

$$\begin{aligned} &\ge \int_0^\infty ze^{-N(f_1z+f_2z^2)}dz \\ &= \frac{\alpha}{4f_2}\sqrt{\frac{\pi}{Nf_2}}\,e^{\alpha^2N/4f_2}(1 - erf(-\alpha\sqrt{N/4f_2})) \\ &\quad + \frac{1}{2Nf_2}. \end{aligned} \tag{146}$$

Similarly,

$$\int_0^\infty e^{-Nf(z)}dz \le -1/(\alpha N) \tag{147}$$

$$\ge \sqrt{\frac{\pi}{4Nf_2}}\,e^{\alpha^2N/4f_2}(1 - erf(-\alpha\sqrt{N/4f_2})). \tag{148}$$

Thus,

$$\begin{aligned} \frac{\displaystyle\int_0^\infty ze^{-Nf(z)}dz}{\displaystyle\int_0^\infty e^{-Nf(z)}dz} &\le \frac{\sqrt{4f_2/\pi}}{\alpha^2N^{3/2}}\frac{e^{-\alpha^2N/4f_2}}{[1 - erf(-\alpha\sqrt{N/4f_2})]} \\ &\le \frac{1 + \sqrt{1 + 8f_2/\alpha^2N}}{2|\alpha|N}, \end{aligned} \tag{149}$$

where, to bound the term dependent on the error function, we have used the left inequality in the following[13]

$$(x + \sqrt{x^2 + 2})^{-1} < e^{x^2}\int_x^\infty e^{-y^2}dy < [x(1 + \sqrt{1 + 4/(\pi x^2)})]^{-1}. \tag{150}$$

We obtain from (146), (147) and (150) the results analogous to (149):

$$\frac{\displaystyle\int_0^\infty ze^{-Nf(z)}dz}{\displaystyle\int_0^\infty e^{-Nf(z)}dz} \ge \frac{|\alpha|}{2f_2}\left(1 - \frac{2}{1 + \sqrt{1 + 16f_2/(\pi\alpha^2N)}}\right). \tag{151}$$

Equations (149) and (151) taken together with the representation for the utilization given in (22) is the content of Proposition 10.

APPENDIX D

Proof of Proposition 11

To prove the proposition, we begin by substituting (54) in the expression for I in (56) to obtain after a change of variable

$$I = (\sqrt{N} + c) \int_0^\infty e^{-(\sqrt{N}+c)\nu} \prod (1 + \nu/(a_i \sqrt{N}))^{b_iN+d_i\sqrt{N}}\, d\nu. \tag{152}$$

We may write (152) as

$$I = \int_0^\infty e^{-A\nu^2/2-B\nu} h(\nu, \sqrt{N}) d\nu, \tag{153}$$

where

$$A = \sum b_i/a_i^2,\ B = c - \sum d_i/a_i \tag{154}$$

and

$$h(\nu, \sqrt{N}) = (\sqrt{N} + c) \exp[A\nu^2/2 + B\nu - \nu(\sqrt{N} + c)][\prod(1 + \nu/(a_i \sqrt{N}))^{b_iN+d_i\sqrt{N}}]. \tag{155}$$

The quantity $h(\nu, \sqrt{N})$ has been defined so that when the second bracketed expression $[\cdot]$ is written as $\exp(\log [\cdot])$ and then the log $[\cdot]$ term expanded, a cancellation of the leading terms is effected by multiplication with the exponential term in (155). In this step, notice has to be taken of the constraint in (55) in that $\exp[A\nu^2/2 + B\nu - \nu(\sqrt{N} + c)] = \exp[A\nu^2/2 - \nu(\sqrt{N} \sum b_i/a_i + \sum d_i/a_i)]$. In this manner we obtain,

$$h(\nu, \sqrt{N}) = (\sqrt{N} + c)\exp\left(\sum_{j\geq 1} F_j(\nu)/\sqrt{N}^j\right), \tag{156}$$

where

$$(-1)^j F_j(\nu) = \left(\frac{1}{j+1}\sum \frac{d_i}{a_i^{j+1}}\right)\nu^{j+1} - \left(\frac{1}{j+2}\sum \frac{b_i}{a_i^{j+2}}\right)\nu^{j+2}. \tag{157}$$

In (156) we have an exponential of a power series in $1/\sqrt{N}$ which may equivalently be represented directly as a power series in $1/\sqrt{N}$:

$$h(\nu, \sqrt{N}) = (\sqrt{N} + c) \sum_{j=0}^{\infty} G_j(\nu)/\sqrt{N}^j. \tag{158}$$

For example, $G_0 = 1$, $G_1 = F_1$, $G_2 = F_1^2/2 + F_2$, $G_3 = F_1^3/6 + F_1F_2 + F_3$. A feature to observe is that $G_j(\nu)$ is a polynomial of degree $3j$ in ν.

For the final form,

$$h(\nu, \sqrt{N}) = \sum_{j=0}^{\infty} H_j(\nu)/\sqrt{N}^{j-1}, \tag{159}$$

where $H_j(\nu) = G_j(\nu) + cG_{j-1}(\nu)$, also a polynomial of degree $3j$ in ν. It is understood that $G_{-1}(\nu) = 0$.

Insertion of (159) in (153) yields Proposition 11, namely

$$I = \sum_{j=0}^{\infty} c_j / \sqrt{N}^{j-1}, \tag{160}$$

where

$$c_j = \int_0^{\infty} e^{-A\nu^2/2-B\nu} H_j(\nu) d\nu. \tag{161}$$

REFERENCES

1. F. P. Kelly, *Reversibility and Stochastic Networks*, New York: John Wiley, 1980, Chapter 3.
2. L. Kleinrock, *Queueing Systems*, New York: John Wiley, 1976, Vol. II, *Computer Applications*, Chapter 4.
3. M. Schwartz, *Computer-Communication Network Design and Analysis*, Englewood Cliffs, N.J.: Prentice-Hall, 1977.
4. F. Baskett et al., "Open, Closed and Mixed Networks of Queues with Different Classes of Customers," J. ACM, *22*, No. 2 (April 1975), pp. 248–60.
5. F. R. Moore, "Computational Model of a Closed Queueing Network with Exponential Servers," IBM J. Res. Dev. (1972), pp. 567–572.
6. J. Buzen, "Queueing Network Models of Multiprogramming," Ph.D. Thesis, Harvard University, 1971.
7. R. R. Muntz and J. Wong, "Asymptotic Properties of Closed Queueing Network Models," Proc. 8th Annual Princeton Conf. Info. Sci. and Systems, March 1974, pp. 348–52.
8. D. P. Gaver and G. S. Shedler, "Approximate Models for Processor Utilization in Multiprogrammed Computer Systems," Siam J. Comput. *2* (1973), pp. 183–92.
9. M. Reiser, "A Queueing Network Analysis of Computer Communication Networks with Flow Control," IEEE Trans. Commun., *COM-27*, No. 8 (August 1979), pp. 1199–1209.
10. S. S. Lam and M. Reiser, "Congestion Control of Store-and-Forward Networks by Input Buffer Limits—An Analysis," IEEE Trans. Commun., *COM-27*, No. 1 (January 1979), pp. 127–34.
11. B. Pittel, "Closed Exponential Networks of Queues with Saturation: The Jackson-Type Stationary Distribution and Its Asymptotic Analysis," Math. Oper. Res., *4*, No. 4 (Nov. 1979), pp. 357–78.
12. E. Arthurs and B. W. Stuck, "A Theoretical Performance Analysis of a Markovian Switching Node," IEEE Trans. Commun., *COM-26*, No. 11 (November 1978), pp. 1779–84.
13. *Handbook of Mathematical Functions*, eds. M. Abramowitz and I. A. Stegun, New York: Dover Pub., 1970, Chapter 19.
14. A. Erdelyi, *Asymptotic Expansions*, New York: Dover Publications, 1956.
15. M. Reiser and S. S. Lavenberg, "Mean Value Analysis of Closed Multichain Queueing Networks," J. Assoc. Comput. Mach., *27* (April 1980), pp. 313–22.
16. S. S. Lavenberg, "Closed Multichain Product Form Queueing Networks with Large Population Sizes," IBM Research Report, RC 8496 (#36973), September 30, 1980.
17. J. P. Buzen, "Computational Algorithms for Closed Queueing Networks with Exponential Servers," Commun. ACM, *16* (1973), pp. 527–31.
18. M. Reiser and H. Kobayashi, "Queueing Networks with Multiple Closed Chains: Theory and Computational Algorithms," IBM J. Res. Dev. (May 1975), pp. 285–94.
19. C. H. Sauer and K. M. Chandy, *Computer Systems Performance Modeling, A Primer*, Englewood Cliffs, New Jersey: Prentice Hall, 1981.
20. J. Zahorjan, "An Exact Solution Method for the General Class of Closed Separable Queueing Networks," 1979 Conf. on Simulation, Measurement, and Modeling of Computer Systems, pp. 107–12.
21. G. Balbo, S. C. Bruell, and H. D. Schwetman, "Customer Classes and Closed Network Models—A Solution Technique," *Proc. IFIP Congress 77*, Amsterdam: North Holland, pp. 559–64.

22. E. T. Copson, *Asymptotic Expansions*, Cambridge: Cambridge University Press, 1965, Chapters 5, 6.
23. F. W. J. Olver, *Introduction to Asymptotics and Special Functions*, New York: Academic Press, 1974, Chapter 3.
24. B. Friedman, "Stationary Phase with Neighboring Critical Points," J. Soc. Indust. Appl. Math., *7*, No. 3 (Sept. 1959), pp. 280–289.
25. E. T. Whittaker and G. N. Watson, *A Course of Modern Analysis*, Cambridge: Cambridge University Press, 1958, Chapter 16.
26. W. Magnus, F. Oberhettinger, and R. P. Soni, *Formulas and Theorems for the Special Functions of Mathematical Physics*, New York: Springer-Verlag, 1966, Chapter 8.
27. N. Bleistein, "Uniform Asymptotic Expansions of Integrals with Stationary Point Near Algebraic Singularity," Commun. Pure Appl. Math., *19*, No. 4 (1966), pp. 353–70.

Copyright © 1981 American Telephone and Telegraph Company
THE BELL SYSTEM TECHNICAL JOURNAL
Vol. 60, No. 5, May–June 1981
Printed in U.S.A.

Time-Frequency Multiplexing (TFM) of Two NTSC Color TV Signals—Simulation Results

By B. G. HASKELL

(Manuscript received September 18, 1980)

Simple frequency-division multiplexing (FDM) of two television signals onto a microwave radio channel is frequently unsatisfactory because of nonlinearity-induced crosstalk from one picture into the other. With time-frequency multiplexing (TFM), two successive scan lines (or fields) of one picture are frequency multiplexed so that they can be sent in one line (or field) period. During the next time interval, two successive lines (or fields) from the other picture are transmitted, thus avoiding crosstalk between pictures. To reduce the bandwidth required, one of the two simultaneously transmitted lines (or fields) is sent as an analog differential signal. In this paper, we discuss computer simulations of these techniques with application to a 20-MHz-bandwidth microwave radio channel. Effects of filtering and nonlinearities are included insofar as possible.

I. INTRODUCTION

It has been known for some time that frequency-division multiplexing (FDM) of two 4.2-MHz National Television System Committee (NTSC) color television signals onto one microwave radio channel is often unsatisfactory because nonlinear distortion causes visible crosstalk between the two pictures. Also, we suspected that 2:1 time compression followed by time multiplexing would lead to problems because of the color carrier (initially 3.58 MHz) being transmitted at 7.16 MHz, where it is much more susceptible to degradations such as selective fading. However, there are other techniques which show promise of not having such problems.

Here we propose time-frequency multiplexing (TFM), which involves sending two successive scan lines (or fields) from one picture during one line (or field) interval, followed by two lines (fields) from the other picture in the remaining time interval. The two successive lines (fields) from one picture would be transmitted simultaneously via frequency-division multiplexing. Multiplexing in this manner avoids crosstalk from one picture into the other which would otherwise be caused by nonlinearities in the transmission if simple FDM were used. We hypothesized that nonlinearity induced crosstalk between successive lines (fields) in the same picture will be much less serious because of their high similarity and the fact that the ghost of a picture overlayed upon itself is practically invisible. With these methods the two incoming pictures would have to be in phase synchronism.

While these methods seem feasible on a qualitative basis, quantitatively many questions arise about their behavior. For example, if one of the successive lines (fields) is sent as a differential signal, e.g., line-to-line difference or field-to-field difference, how much can the bandwidth of the differential signal be reduced before detectable picture degradation results? Also, how much intermodulation crosstalk can be tolerated?

To take a first step toward answering these questions and to gain a rudimentary idea as to the limitations of these techniques, we undertook computer simulations. The computer facility display had a resolution of only 256 × 256 pels (picture elements), which is only about one-quarter the resolution of an NTSC picture. Thus, each processed picture can be regarded as one quadrant of a full-resolution NTSC picture. Modulation and filtering inherent in NTSC color multiplexing and demultiplexing were realized by digital filters operating at a simulated sampling rate of four times the color subcarrier frequency. We maintained full NTSC bandwidth for luminance and chrominance, insofar as possible, even though very few home or studio monitors in the U.S. are capable of displaying full resolution (see Appendix A for details). We simulated time and frequency multiplexing inherent in the transmission methods under discussion using baseband digital filters. Effects of nonlinearities were also simulated at baseband, as will be described below.

II. DESCRIPTION OF THE METHOD

Long-haul broadcast color TV transmission is presently by FM, as shown in Fig. 1. One might suggest transmitting a second vestigial-sideband (VSB) signal via FDM as shown in Fig. 2. However, difficulties immediately arise: (*i*) The microwave amplifier nonlinearities can cause intermodulation crosstalk, causing ghosts at an unacceptable level. (*ii*) Modulating the baseband plus VSB signal onto an FM carrier may exceed the bandwidth allowed, since the modulation is not strictly narrowband FM. Carson's rule estimates the required rf bandwidth in Fig. 2 as ≈ 2 (10 MHz + deviation).

Ghosts caused by nonlinearity-induced intermodulation can be made

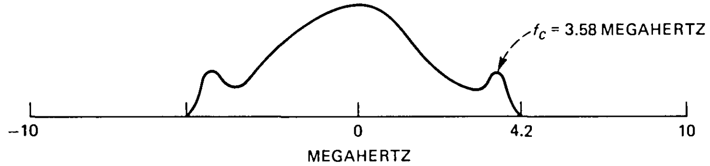


Fig. 1—TV transmission via narrowband FM. The bandwidth actually used is 8.4 MHz plus the bandwidth used by sidelobes (not shown here). If the peak FM deviation is ΔF, then Carson's rule estimates the required bandwidth as 8.4 MHz + $2\Delta F$. The color subcarrier frequency is $f_c \approx 3.58$ MHz.

much less visible if *time*-division multiplexing of the two picture signals is employed instead of frequency-division multiplexing. As explained in the introduction, this entails sending two lines (or fields) from one picture in one line (field) interval followed by the same information from the other picture in the next time interval. This requires that the two incoming pictures be synchronized with each other at all times (if this is not the case, a synchronizing device is needed).* Time multiplexing eliminates crosstalk between the two pictures.

Unfortunately, if FDM is used to transmit simultaneously two successive lines (or fields), nonlinearity-induced crosstalk between these two components will still exist. However, two successive lines (or fields) from the same picture tend to be very much alike. Thus, overlaying the ghost of one onto the other should be practically invisible.

From Fig. 2, we see that there may not be enough bandwidth to send two successive lines (or fields) using simple vestigial-sideband in the upper band. However, if a difference signal, e.g., line-to-line difference or field-to-field difference, is transmitted in the upper band, then the aforementioned difficulties can be reduced. A line (or field) difference signal has much less power than the input video and, judging from picture coding experience, probably requires less bandwidth for transmission as well. It has little or no power near DC and, therefore, lends itself to single-sideband modulation (SSB). The arrangement of Fig. 3 may be suitable. In this case the differential signal in SSB modulated, added to the baseband signal and passed to the radio system for transmission via FM. The single-sideband carrier could be transmitted during horizontal blanking so that it would not use up valuable bandwidth during the visible part of the picture, or it could be generated from the color subcarrier.

In deriving the differential signals, differences should be taken

* Synchronization of the two pictures should be such that horizontal sync pulses align. In this way, switching from one picture to the other can occur, for example, following the color burst, and switching transients will not be visible in either picture. An ordinary frame synchronizer should suffice.

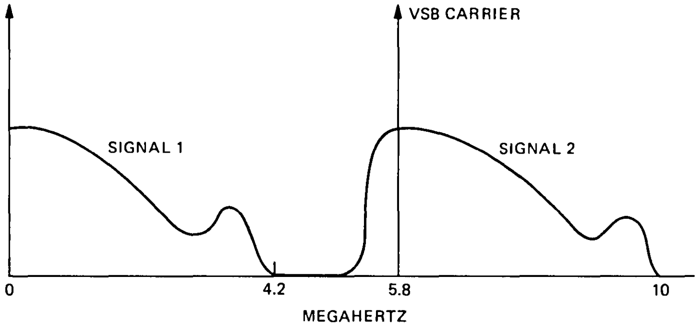


Fig. 2—Transmitting two TV signals via simple frequency-division multiplexing using vestigial-sideband in the upper frequency band. This scheme is unsuitable because of (*i*) nonlinearities and (*ii*) the fact that the modulation is not strictly narrowband FM and may require more than the allowable bandwidth according to Carson's rule, i.e., FM bandwidth ≈ 2 × (message bandwidth + deviation).

between picture elements which on the average have the same color subcarrier phase. This prevents a strong color subcarrier component from appearing in the difference signal. For example, in Fig. 4 three successive lines in one field are shown, and the center one is to be sent via a line difference signal. Pels A, B, C, D, and X all have the same color subcarrier phase (on a flat-colored area). Thus, a suitable line differential signal D_L would be X minus an average of A, B, C, and D. At the receiver, pels A, B, C, and D are averaged in exactly the same way, and the received differential signal is added to recover a replica of the original center line.

In Fig. 5 the center line is from the present field while the other two are from the previous field. Pels A, B, C, and X all have the same color subcarrier phase on a flat-colored area. A suitable differential signal

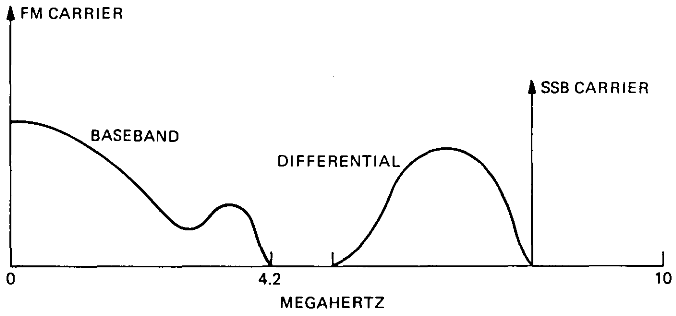


Fig. 3—Time-division multiplexing (baseband *and* differential signals are from the same source). If one of the two lines (or fields) is sent as a differential signal, then it can be placed into the upper band via single-sideband modulation prior to transmission over the radio facility. The single-sideband carrier could be transmitted sometime during horizontal blanking so that it would not use up valuable bandwidth during the visible part of the picture, or it could be generated from the color subcarrier.

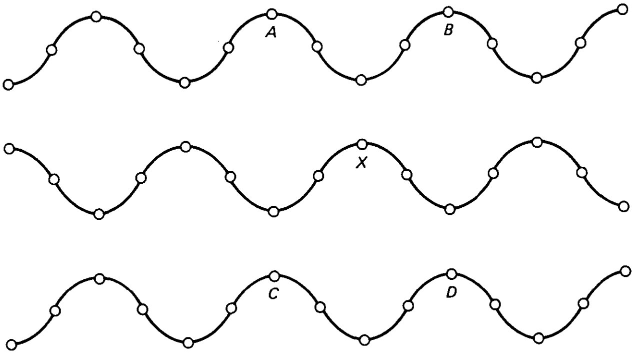


Fig. 4—Three successive lines in one field are shown, and the center one is to be sent via a line-difference signal. Pels A, B, C, D, and X all have the same color subcarrier phase (on a flat colored area). Thus, a suitable line differential signal D_L would be X minus an average of A, B, C, and D, i.e., $D_L = X - \frac{1}{4}(A + B + C + D)$.

D_F is X minus an average of A, B, and C. Since C is closer to X than A or B, it should be weighted relatively more, i.e., $\alpha > \frac{1}{3}$ should be used. At the receiver, we recover the center line by forming the weighted average and adding the received differential signal. In both Figs. 4 and 5 the pel spacing implies a sampling rate of four times the color subcarrier frequency, which is the rate used in the computer simulations.

III. BANDWIDTH REQUIRED FOR DIFFERENTIAL SIGNALS

The first question which we confronted in the computer simulations was "How much can the differential signal be bandlimited before

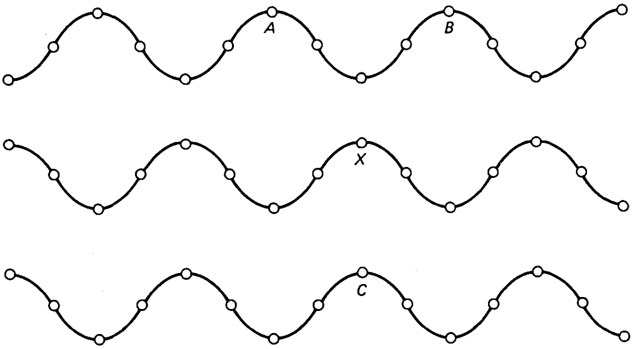


Fig. 5—Three successive lines in one frame are shown, and the center one is to be sent via a field difference signal. The center line is from the present field while the other two are from the previous field. Pels A, B, and C all have the same color subcarrier phase on a flat area. A suitable differential signal D_F is X minus an average of A, B, and C, i.e., $D_F = X - [\alpha C + (A + B)(1 - \alpha)/2]$. Since C is closer to X than A or B, it should be weighted relatively more, i.e., $\alpha > \frac{1}{3}$ should be used. At the receiver the center line is recovered by forming the weighted average and adding the received differential signal.

detectable picture distortion results?" In the simulations, we maintained full NTSC resolution in the color carrier modulation and demodulation (see Appendix A for details), and used a monitor with separate red, blue, green (*RBG*) inputs for display. We used slides from the Society of Motion Picture Television Engineers (SMPTE) with *R*, *B*, and *G* components digitized to eight bits by means of a flying spot scanner for the pictures.

Starting with *RBG* signals for each picture, we first produced NTSC composite signals as described in Appendix A. From these, we obtained reference pictures by NTSC demodulation into the *R*, *B*, *G* components required for display.

To test line-differential transmission, we computed the line difference shown in Fig. 4, from the NTSC composite signals for alternate lines in each field. We then bandlimited the differential signal to various bandwidths, using finite impulse-response digital filters to avoid phase distortions. The bandlimited differential signals were then added to the appropriate averaged signals in alternate lines to recover replicas of the original baseband composite signals. The composite signals were then demodulated to obtain processed pictures.

We compared each processed picture with its corresponding reference picture by switching between the two on the same monitor and closely inspecting small areas for perceivable change. From this we concluded that as long as the line differential signal had a bandwidth of at least 3 MHz, no perceivable difference would result. Below 3 MHz, a slight vertical color shading could be detected in a few areas, such as the boundary beween red lipstick and white teeth.

To test field-differential transmission, we carried out the same procedure using the differential signal shown in Fig. 5. We examined several values of α in the range 0.4 to 0.6, but the results were the same in all cases. We concluded that as long as the field differential signal had a bandwidth of at least 2 MHz, no perceivable difference would result between the reference pictures and the processed pictures. To test the effect on movement rendition, we carried out the processing on sequences of frames moving in pure translation, vertically and horizontally. The results were the same as for still frames and are summarized in Table I.

It is certainly possible to generate test signals which cannot be transmitted without degradation by the methods suggested here. For example, monochrome vertical stripes at the color subcarrier frequency generate a strong line-differential signal at the color subcarrier frequency which would not pass through a 3-MHz low-pass filter. If the vertical interval test signal (VITS) happens to fall on a line that is transmitted via a bandlimited differential signal, then degradation results. The VITS is, after all, designed to test baseband transmission

Table I—Bandwidths required for transmitting differential signals if no perceivable picture degradation is allowed.

Signal	Required Bandwidth
Baseband Video	4 MHz
Line Difference	3 MHz
Field Difference	2 MHz

and horizontal resolution. As such, it should be sent intact without any differential processing. Other test signals should be devised to measure vertical color resolution in systems, such as proposed here, which limit the transmitted vertical color resolution to that actually required by pictures.

IV. APPLICATION TO A 20-MHz BANDWIDTH MICROWAVE RADIO CHANNEL

4.1 Channel characteristics

Figure 6 shows a typical long-haul, microwave, television transmission system. Preemphasis of high frequencies gives some protection against nonlinearities. Figure 7 gives the preemphasis characteristic; de-emphasis is the inverse of preemphasis. The FMT produces 32.0-MHz deviation per volt input. The gain a is chosen so that a nominal deviation of about 2 MHz results from a one-volt peak-to-peak color bar video signal. However, sustained deviations of 3.2 MHz (found by computer search) can occur for certain input signals, and transient deviations as high as 3.7 MHz (preemphasized 2T pulse)[1] are possible. Ac coupling occurs before the FMT. However, this has little effect on possible peak deviations since its passband extends well below most frequencies of interest.

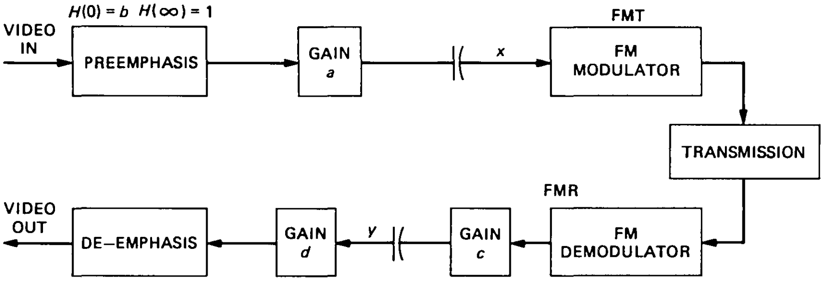


Fig. 6—A typical long-haul television transmission system. $a = -12.6$ dB, $b = -13.3$ dB, $c = 16$ dB, $d = -3.4$ dB, $acd = 1$. De-emphasis is the inverse of preemphasis. The FM modulator produces 32.0-MHz deviation per volt input.

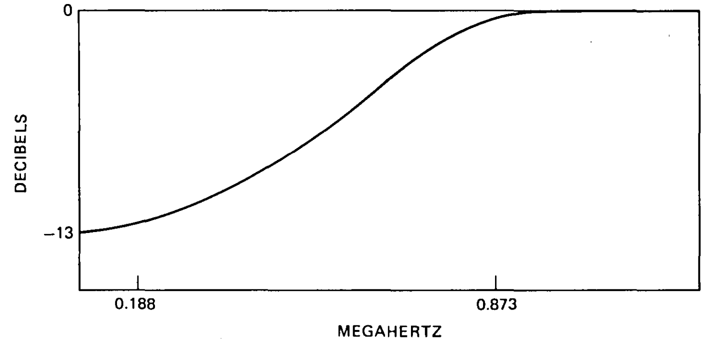


Fig. 7—Preemphasis characteristic. $H(s) = (\omega_1 + s)/(\omega_2 + s)$, where $f_1 = 199$ kHz, $f_2 = 873$ kHz, and $H(0) = b = -13.3$ dB. Preemphasis of high frequencies gives some protection against nonlinearities and improves SNR in colored areas of the picture.

Nonlinearities vary considerably between channels. However, one such microwave radio system (4000-mile terrestrial, 4GHz) was measured, and Appendix B estimates video degradation for a standard VITS. Differential gain is estimated at 8.5 IRE* (13 IRE allowed), and worst case differential phase is estimated at 9.7° (5° allowed). These results indicate that the measured channel was marginal and, therefore, simulation results based on these measurements should be on the conservative side.

4.2 Time-frequency multiplexing

We wish to transmit a baseband signal of 4.2-MHz bandwidth and a single-sideband modulated differential signal in the positive frequency band as shown in Fig. 3. Figure 8 shows a configuration for doing so. It is similar to Fig. 6 except that a 20-MHz bandpass filter following the FM modulator is shown explicitly, and the preemphasis is changed to admit the possibility of attenuating the higher frequencies of the baseband signal somewhat to obtain satisfactory performance.

A line differential signal, which requires 3-MHz bandwidth, is probably not suitable for this application. However, a field differential signal, which requires only 2 MHz of bandwidth, could fit into the 5 to 7-MHz band if sharp cutoff filters were used. Hereafter, results will apply to a system using field differential signals.

The differential signal has no dc component and is small most of the time with ordinary pictures, occasionally reaching values of $\pm 0.36M$.†

* 1 volt = 140 IRE.

† Measured by computer simulation using SMPTE slides. Electronically generated tests signals can be concocted, however, which produce differential signal values as large as $\pm M$. Here, $M = 0.714$ volt (see Fig. 9).

Carson's rule states that the sum of the total message bandwidth and the FM frequency deviation should not exceed one-half the available channel bandwidth. Since the channel bandwidth is 20 MHz, and the proposed message bandwidth is 7 MHz, the FM deviation should not exceed 3 MHz. Unfortunately, as we have seen, the system in its present form already has deviations larger than 3 MHz, even without the addition of a modulated differential signal. However, these deviations are often short lived and occur at sharp edges in the picture where considerable distortion can be withstood. Still, some signal attenuation is obviously necessary to accommodate a 7-MHz message bandwidth. However, it may not be necessary to attenuate the signals so much that the deviation is always below 3 MHz. If the FMT is followed by a reasonably good 20-MHz bandpass filter, then some of the responsibility for maintaining bandwidth might be shifted to it, at the cost of some picture degradation. This degradation due to the bandpass filter will only occur if the preemphasized baseband signal is large, while at the same time the differential signal is large. In this

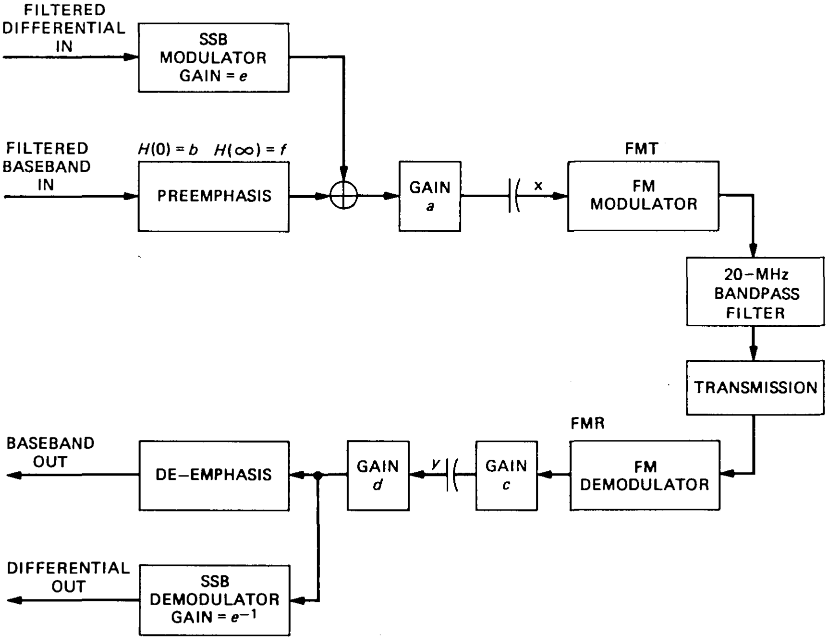


Fig. 8—Configuration for transmitting a single-sideband modulated differential signal. $a = -12.6$ dB, $b = -13.3$ dB, $c = 16$ dB, $d = -3.4$ dB, $acd = 1$. It is similar to Fig. 6 except that a 20-MHz bandpass filter following the FM modulator is shown explicitly, and the preemphasis is changed to admit the possibility of attenuating the higher frequencies of the baseband signal somewhat to obtain satisfactory performance. Gain e causes attenuation of the modulated differential signal (including the effect of single-sideband modulation), and gain f causes attenuation of the high frequencies of the baseband signal. $e = f \approx -3$ dB gives just visible distortion.

case, and in this case only, the bandpass filter will attenuate the modulated differential signal, thus causing distortion. This distortion should be relatively invisible, however, due to its proximity to sharp brightness transitions in the picture. In Fig. 8, gain e causes attenuation of the modulated differential signal (including the effect of single-sideband modulation), gain f causes attenuation of the high frequencies of the baseband signal. Suitable values for e and f must be determined experimentally. However, some guidelines can be obtained intuitively as follows.

If a constant peak-white baseband value of $M = 0.714$ volt (=100 IRE) occurs at the same time as a differential signal peak of $0.36M$, then the input to the FM modulator (assuming worst case DC = 0)* is

$$x = abM + 0.36aM \cos'\omega_D t, \tag{1}$$

where ω_D is the differential signal carrier frequency, and cos′ indicates single-sideband modulation. If the conservative position is taken, of never allowing a deviation above 3 MHz, then from values previously defined, e should be less than −0.4 dB. This is a very modest attenuation.

Let us consider another input signal. A maximum sustained preemphasized baseband signal occurs if the video input is $0.52M + 0.48M \cos\omega_c t$. If this value occurs at the same time as a peak differential signal value of $0.36M$, then again assuming the worst case (DC = 0),

$$x = 0.52abM + 0.48afM \cos\omega_c t + 0.36aeM \cos'\omega_D t. \tag{2}$$

If, in this case, the very conservative approach of 3-MHz maximum deviation is taken, then with an FMT deviation of 32 MHz/volt,

$$0.52abM + 0.48afM + 0.36aeM \leq 0.094 \text{ volt}. \tag{3}$$

If equal values are chosen for e and f,

$$e = f \leq -5.4 \text{ dB}. \tag{4}$$

Although choosing e and f to satisfy Eq. (4) eliminates almost all distortion caused by the 20-MHz bandpass filter, it also leads to a decrease in overall SNR, i.e., fade margin. Since some of the bandpass filter distortion will be invisible, it seems wasteful to use such low values for e and f. Larger values should be feasible with the same subjective picture quality. Thus, another of the reasons for performing the computer simulations was to estimate the range of acceptable values for e and f.

* Average signal level. This is not transmitted in an ac coupled system.

V. COMPUTER SIMULATION OF NONLINEARITIES

Since the transmission channel is not precisely understood and is time varying as well, such simulations can only be expected to give a vague idea of overall performance. Hopefully, however an estimate of worst-case performance can be obtained if a conservative approach is taken. The channel characteristics used in Appendix B and shown in Table II probably represent a marginally acceptable situation insofar as nonlinearities are concerned. Thus, the use of these data in the simulations should give conservative results.

There are several approaches to modeling the nonlinearity, including the rather complicated procedure of numerically carrying out the complete SSB modulation, multiplexing and FM operations, and passing the result through the nonlinearity. However, the simulations to be described below were all carried out at baseband for simplicity and since intermodulation crosstalk was the main item of interest.

The basic approach taken in evaluating the effects of nonlinearities was to introduce degradations into the NTSC composite signal and see if they were visible in the resulting reproduced picture. Two types of degradation were studied: (*i*) intermodulation crosstalk resulting from nonlinearities, and (*ii*) the effect of the 20-MHz bandpass filter following the FM modulator in Fig. 8.

Baseband and differential signals were first computed as in Section III for each of the original SMPTE pictures. Intermodulation crosstalk terms were then estimated as described in Appendix C. Some of the terms degrade the baseband signal, and some degrade the differential signal.

The effect of the 20-MHz bandpass filter was approximated as follows: If at certain times the sum of the baseband and differential signal magnitudes at the input to the FMT was large enough to produce a 3-MHz deviation (again assuming worst case dc = 0), then the magnitude of the differential signal was reduced until the deviation fell below 3 MHz. If the baseband signal by itself was large enough to

Table II—Results of two-tone measurements on a typical long-haul network. $A = B = -26$ dBv, $c = 16$ dB. From the measured harmonic outputs, the coefficients u and v of the third-order model can be obtained.

α (MHz)	β (MHz)	$ucAB$ FMR average dBv at $\beta + \alpha$ and $\beta - \alpha$	$\tfrac{3}{4}vA^2Bc$ FMR average dBv at $\beta + 2\alpha$ and $\beta - 2\alpha$	u	v
0.1	6.55	−41	−49	0.56	5.96
0.1	3.55	−35	−46.5	1.12	7.94
0.1	1.55	−36	−39	1.00	18.84

produce a 3-MHz deviation, then the differential signal was set to zero.*

In the simulations, gain factors *a* through *f* were all incorporated as shown in Fig. 8; factors *a* through *d* were fixed having the values shown, while *e* and *f* were made variable. The nonlinearity coefficients were taken from Table II, depending on the applicable frequency range.

Results were generally encouraging, considering the pessimistic and conservative assumptions built into the simulations. Using attenuation factors $e = f = 0.5$ resulted in no visible degradation, as expected from eq. (4). Larger values in the range 0.7 to 0.8 (−3 to −2 dB) gave rise to barely visible color shading in a few areas such as the boundary between red lipstick and white teeth. Otherwise, the effects were invisible.

VI. CONCLUSION

Color pictures were processed via computer simulation to test the feasiblity of using time-frequency multiplexing to transmit two broadcast-quality color television signals over a 20-MHz bandwidth microwave radio channel. We considered bandwidth allocation and nonlinearities in detail. At every turn in the study, conservative assumptions were made. The displayed pictures produced full-color bandwidth even though very few home or studio monitors in the U.S. are capable of doing so. Nonlinearity parameters which were used in the simulations were obtained from an, at best, marginal channel. Worst-case theoretical nonlinearity impairments were assumed. And, finally, for picture quality assessment an A–B comparison on the same monitor was used, which is one of the most stringent tests known.

Results were rather encouraging. There appeared to be enough bandwidth to accomplish the transmission. With a modest 3-dB reduction of signal level, simulated transmission defects were practically invisible.

Utilization of field differential signals requires field memories at the transmitter and receiver. Although they are expensive at present, their cost is dropping. Moreover, if synchronization of two pictures is required at the transmitter, then the memory of the synchronizer can be used.

Audio transmission has not been considered here. One might be tempted to place audio carriers in the vacant frequency band between the baseband and differential signals. However, this would require a rather linear channel, which is the very problem that the techniques

* An automatic gain control could, in fact, be implemented in this way to guarantee deviation less than 3 MHz.

of this paper are meant to avoid. A better solution would be to transmit the audio digitally,· either in the horizontal sync pulse[2] or during vertical blanking.

APPENDIX A

Computer Simulation of NTSC Color Multiplexing and Demultiplexing

The computer simulation facility has a picture resolution of 256 × 256 pels per frame. At a frame rate of 30 Hz, 15 lines vertical blanking, 17 percent horizontal blanking, this implies a sampling rate of ≈2.51 MHz. Thus, video signal bandwidth of the computer is conservatively estimated to be about 1 MHz. Since U. S. broadcast rate NTSC color television signals have a nominal bandwidth 4.2 MHz, scaling by a factor of approximately four is required to do meaningful simulations.

The NTSC broadcast color subcarrier frequency is approximately 3.58 MHz. Thus, for the simulations, a value around 0.89 MHz should be used. In addition, the simulation color subcarrier frequency *must* be an odd multiple of half the line scanning frequency of the simulation system ($f_H \approx 8.1$ kHz). Thus, a convenient value for color carrier frequency that was chosen for the simulations was $f_c = 108.5 f_H \approx 0.88$ MHz.

The simulated sampling rate of $4 f_c$ equals $434 f_H$ or 434 pels per line. Assuming about 17 percent horizontal blanking, we have 358 pels in the active area of the picture (which corresponds to the 256 pels of the display). To convert the original red, blue, and green (RBG) signals having 256 pels per line to the desired 358 pels per line, linear interpolation was used. This sampling rate conversion is suboptimal in that a frequency rolloff of 4.7 dB at 1 MHz is introduced. The extra effort required to design optimal digital filters for sampling rate conversion did not seem worthwhile at this time. After processing, the conversion back to 256 pels per line was again performed via linear interpolation.

From the RBG signals, the full bandwidth Y, I, Q signals* were obtained by the standard matrix relation[3]

$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = \begin{bmatrix} 0.3 & 0.11 & 0.59 \\ 0.6 & -0.32 & -0.28 \\ 0.21 & 0.31 & -0.52 \end{bmatrix} \begin{bmatrix} R \\ B \\ G \end{bmatrix} \tag{5}$$

The I and Q signals were then bandlimited to ≈0.375 MHz and 0.125 MHz, respectively (0.25 × NTSC values) using symmetrical finite-impulse-response digital filters to avoid phase distortion. Filters were chosen to meet NTSC specifications, yet to be short enough to avoid excessive ringing.

* Y = luminance, I and Q = chrominance signals.

Samples at $4f_c$ were assumed to occur at the peaks and zeros of the I and Q carrier sinusoids. The chrominance (CH) pels could then be formed simply from

$$\pm CH(N) = I(N)\cos(N\pi/2) + Q(N)\sin(N\pi/2), \qquad N = 1, \cdots, 358, \qquad (6)$$

the plus and minus signs being taken on alternate lines of the field. The chrominance and luminance were then added together and filtered to a bandwidth of 1 MHz, again with a FIR digital filter to avoid phase distortion. The result is the composite NTSC color signal.

Recovery of the R, B, G signals from the composite signal is not quite so straightforward if full bandwidth for the luminance and chrominance is to be maintained. Comb filtering of three successive lines ($-0.25L_1 + 0.5L_2 - 0.25L_3$) followed by bandpass filtering (center frequency f_c) was used to recover the chrominance signal for the center line. Simple subtraction of the chrominance from the composite signal yielded the luminance pels.

Recovery of I and Q from the chrominance started with a quadrature AM demodulation,

$$I(N) = \pm CH(N)\cos(N\pi/2),$$
$$Q(N) = \pm CH(N)\sin(N\pi/2), \qquad (7)$$

the plus and minus signs being taken on alternate lines. The required low-pass filtering was implemented simply by replacing zero samples by an average of their neighbors.

At this point, the I signal required equalization since it lost part of its upper sideband during the 1-MHz low-pass filtering of the composite signal. The frequency range from 0.125 MHz to 0.375 MHz must be increased by 6 dB. This was accomplished, again, by a FIR digital filter. Having obtained Y, I, and Q, the R, B, G signals were produced by inverting the matrix equation (5) above.

APPENDIX B

Effects of Nonlinearities on the Vertical Interval Test Signal (VITS)

In Fig. 6, nonlinearities are often modeled as third-order polynomials,[4]

$$y = f(x) = c(x - ux^2 - vx^3). \qquad (8)$$

This model is reasonable for transmission systems with "small" nonlinearities. The nonlinearity coefficients u and v are often estimated by sending the two-tone signal

$$x = A\cos\alpha t + B\cos\beta t, \qquad (9)$$

in which case y/c consists of a number of sum and difference frequencies as follows:[4]

$$\text{DC: } u(A^2 + B^2)/2 \quad \text{(deleted at FMR)} \tag{10}$$

$$\text{First Order: } A \cos\alpha t + B \cos\beta t + \text{very small terms,} \tag{11}$$

$$\begin{aligned} \text{Second Order: } & -\tfrac{1}{2}u(A^2 \cos 2\alpha t + B^2 \cos 2\beta t) \\ & -uAB[\cos(\alpha + \beta)t + \cos(\alpha - \beta)t], \end{aligned} \tag{12}$$

$$\begin{aligned} \text{Third Order: } & -\tfrac{1}{4}v(A^3 \cos 3\alpha t + B^3 \cos 3\beta t) \\ & -\tfrac{3}{4}vA^2B[\cos(2\alpha + \beta)t + \cos(2\alpha - \beta)t] \\ & -\tfrac{3}{4}vB^2A[\cos(2\beta + \alpha)t + \cos(2\beta - \alpha)t]. \end{aligned} \tag{13}$$

In 1974, two-tone measurements were made of a typical long-haul network. From these measurements the coefficients u and v can be estimated. Results are frequency dependent and are shown in Table II for $A = B = -26$ dBv,* and $c = 16$ dB.

These data can be used to estimate degradations which would occur in the transmission of a vertical interval test signal (VITS). Let M = 100 IRE = 0.714 volt be the blanking-to-peak magnitude of the video signal as shown in Fig. 9. Values of gains a, b, and c are given in Fig. 6.

Differential gain (Ref. 1, Section 3.13)

$$\begin{aligned} \text{dc} &= 0, \\ \text{Peak input} &= M(0.9 + 0.2 \cos\omega_c t). \end{aligned} \tag{14}$$

Thus,

$$\begin{aligned} x &= aM(0.9b + 0.2 \cos\omega_c t) \\ &\triangleq A + B \cos\omega_c t, \end{aligned} \tag{15}$$

where A is the low-frequency magnitude, and B is the color-subcarrier magnitude. From eqs. (11), (12), and (13) with $\alpha \approx 0$ and $\beta = \omega_c$, the color-subcarrier terms in y/c (when phases align,[5] which is worst case) are

$$B - 2uAB - \tfrac{9}{2}vA^2B. \tag{16}$$

Thus, at ω_c the signal and distortion components in y are, respectively,

$$\begin{aligned} R &= Bc = 0.2aMc, \\ \Delta R &= -2uABc - \tfrac{9}{2}vA^2Bc. \end{aligned} \tag{17}$$

* Zero to peak.

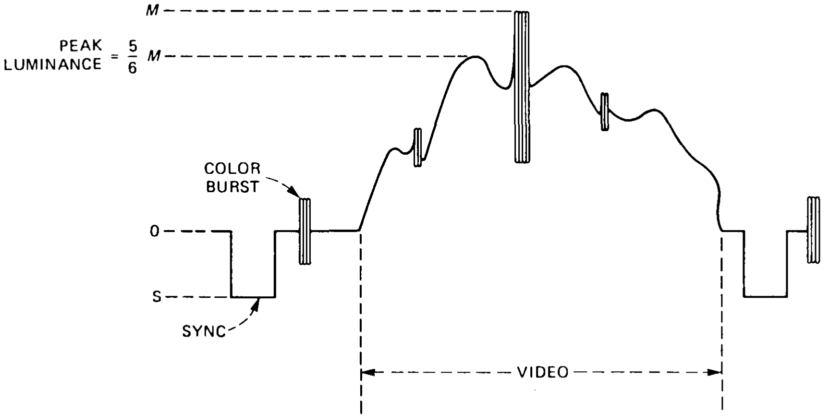


Fig. 9—Composite video signal, including sync, color burst and, video. $S = -0.286$ volt. According to specifications, M can be as large as 0.857 volt. However, to produce, the normally used one-volt peak-to-peak composite signal, $M = 0.714$ volt (=100 IRE).

De-emphasis has no effect at this frequency. Thus, according to the test procedure we should divide the above equations by $0.2ac$ to bring the signal component up to M. Then, peak differential gain is given by

$$\Delta G = \frac{-2uABc - {}^{3}\!/_{2} vA^2Bc}{0.2ac}. \tag{18}$$

Using values defined previously and $u = 1.12$, $v = 7.94$ from Table II,

$$\Delta G = -0.06 \text{ volts} = -8.5 \text{ IRE}, \tag{19}$$

which is well within the 15 IRE allowed.[1]

Differential phase (Ref. 1, Section 3.14)

The same input signal is used as in the differential gain test. Using the same definitions for signal R and distortion ΔR, worst-case differential phase occurs when these two components are exactly 90° out of phase.[5] In this case, peak-to-peak differential phase is

$$\Delta\phi = 2 \tan^{-1} \frac{\Delta R}{R} = 2 \tan^{-1} A\left(.2u + \frac{3}{2} vA\right), \tag{20}$$

which for the values defined previously evaluates to

$$\Delta\phi = 9.7°,$$

which is somewhat outside the 5° allowed. However, this is a worst-case maximum and might not occur in practice.

Conclusion

From these estimates of distortion one concludes that the channel measured was not of extremely high quality and may have been only of marginal quality. Thus, simulation results using those measurements should be on the conservative side.

APPENDIX C

Important Intermodulation Crosstalk Terms

First consider crosstalk between baseband color-subcarrier frequency and the differential signal. Let BB_c be the input baseband chrominance component (frequencies = $\omega_c \approx 3.58$ MHz) and let D be the input differential signal (carrier frequency $\omega_D \approx 7$ MHz). Then, from Fig. 8,

$$\begin{aligned} x &= af\cdot BB_c \cos\omega_c t + aeD \cos\omega_D t \\ &\triangleq A_1 \cos\omega_c t + B \cos\omega_D t, \end{aligned} \tag{21}$$

as in eq. (9). Then, from eqs. (11) to (13) the corresponding in-band distortion components in y are

$$-\tfrac{1}{2}ucA_1^2 \cos 2\omega_c t - uA_1B \cos(\omega_D - \omega_c)t - \tfrac{3}{4}vcA_1^2B \cos(\omega_D - 2\omega_c)t. \tag{22}$$

Combining eqs. (21) and (22) yields estimates of interfrequency crosstalk caused by the above frequency components. In the computer simulations BB_c was obtained by first bandpass filtering the composite signal to retain only components from 2 to 4 MHz. The result was then shifted down in frequency by 2 MHz to obtain the signal BB_c used in eqs. (21) and (22) to compute distortion.

Similar crosstalk occurs between baseband low frequencies and the modulated differential signal. However, because of preemphasis, A in eqs. (11) to (13) is small, making this distortion negligible.

Crosstalk between baseband midfrequencies, e.g., ω_M around 1 MHz, and the modulated differential signal can be estimated similarly. Let BB_M be the input baseband component in the midrange frequencies ($\omega_M \approx 1$ MHz), and define D as above. Then from Fig. 8,

$$\begin{aligned} x &= af\cdot BB_M \cos\omega_M t + aeD \cos\omega_D t \\ &\triangleq A_2 \cos\omega_M t + B \cos\omega_D t. \end{aligned} \tag{23}$$

From eqs. (11) to (13) the in-band distortion components in y are

$$\begin{aligned} &-\tfrac{1}{2}ucA_2^2 \cos 2\omega_M t - ucA_2B \cos(\omega_D - \omega_M)t \\ &-\tfrac{1}{4}cvA_2^3 \cos 3\omega_M t - \tfrac{3}{4}vcA_2^2B \cos(\omega_D - 2\omega_M)t. \end{aligned} \tag{24}$$

In the computer simulations, BB_M was obtained by low-pass filtering (2-MHz cutoff) the composite signal.

REFERENCES

1. *Network Transmission Committee Report No.* 7, June 1975 (revised January 1976), published by the Public Broadcasting Service.
2. H. Dirks et al., "TV-PCM 6 Integrated Sound and Vision Transmission System," Elect. Commun., *52,* No. 1 (1977), pp. 62–7.
3. D. Fink, ed., *Television Engineering Handbook,* New York: McGraw-Hill, 1957, Chapter 9. Now published by University Microfilms, Ann Arbor, Michigan.
4. *Transmission Systems for Communications,* Bell Telephone Laboratories, 1971, 4th. ed. rev., Chapter 10.
5. Ref. 4, pp. 272–7.

Copyright © 1981 American Telephone and Telegraph Company
THE BELL SYSTEM TECHNICAL JOURNAL
Vol. 60, No. 5, May–June 1981
Printed in U.S.A.

Overflow Models for *Dimension*® PBX Feature Packages

By L. KAUFMAN, J. B. SEERY, and J. A. MORRISON

(Manuscript received October 10, 1980)

We present numerical results for some traffic overflow systems with queuing. Traffic is offered by two independent streams to two groups of trunks, with a finite number of waiting spaces for each, and some overflow capability from the primary group to the secondary group. We consider three different overflow systems, two of which model feature packages offered in Dimension® PBX. *For given offered loads and unit mean holding time, we determine the number of trunks and waiting spaces in the two groups so that the blocking probabilities, and the average delays of queued calls do not exceed prescribed values. We also calculate various other quantities, such as the occupancies of the trunk groups and the probability of overflow from the primary group to the secondary group. Finally, we examine the effect on the blocking probabilities and the average delays of varying the loads offered to a given system.*

I. INTRODUCTION

In this paper we present numerical results for some traffic overflow systems with queuing. Traffic is offered by two independent streams to two groups of trunks with a finite number of waiting spaces for each, and some overflow capability from the primary group to the secondary group. The holding times of the calls are independent, and exponentially distributed. In two of the three systems considered, the overflow capability models feature packages (FP) offered in *Dimension*® PBX. The third system, which is considered for comparison, differs from the other two systems in that no overflow is permitted if there is a waiting space available in the primary queue.

Since there are a finite number of trunks and waiting spaces in each group, arriving calls may be blocked and cleared from the system. For given offered loads and unit mean holding time, we determine the number of trunks and waiting spaces in the two groups so that the blocking probabilities and the average delays of queued calls do not exceed prescribed values. We also calculate various other quantities, such as the occupancies of the trunk groups and the probability of overflow from the primary group to the secondary group. In addition, we examine the effect on the blocking probabilities and the average delays of varying the loads offered to a given system.

The numerical results are based on different techniques developed by Kaufman[1] and Morrison.[2,3] The basic problem is to solve a large sparse system of linear equations for the steady-state probabilities of the number of calls in the two groups. Kaufman used a numerical technique involving matrix separability (block diagonalization), and in addition obtained numerical solutions by means of successive over-relaxation techniques. Kaufman has also applied her techniques to overflow systems with more than two groups. Morrison confined his attention to overflow systems with two groups, and he adopted an analytical approach which considerably reduces the dimensions of the problem.

We have been able to obtain very accurate numerical results by the methods presented in this paper. Indeed the various steady-state quantities of interest obtained by the procedures of Kaufman[1] and Morrison[2,3] agree to many significant figures. These results are considerably more accurate, and less expensive to obtain, than simulation results. They may be used as a check on the accuracy of approximate methods which have been developed for dealing with overflow problems, some of which are mentioned later in this section.

We now describe in detail the overflow systems which we consider, as depicted in Fig. 1. There are n_1 trunks and q_1 waiting spaces in the primary group, and n_2 trunks and q_2 waiting spaces in the secondary group. Traffic is offered to the two groups by two independent Poisson streams S_1 and S_2, with rates λ_1 and λ_2, respectively. The holding times of the calls are independent, and exponentially distributed with mean $1/\mu$. If all n_2 trunks in the secondary group are busy when a call from stream S_2 arrives, the call is queued if one of the q_2 waiting spaces is available, otherwise it is blocked and cleared from the system. Calls waiting in the secondary queue are placed in service on a first-in first-out basis as secondary trunks become available.

Three cases are considered for the treatment of calls offered to the primary group. The first two cases model feature packages offered in *Dimension* PBX. For these two cases, if all n_1 trunks in the primary group are busy when a call in S_1 arrives, it is placed in service in the secondary group if there is a trunk available and there are no calls waiting in the secondary queue. If no trunk is available, then the call is queued in the primary group if one of the q_1 waiting spaces is

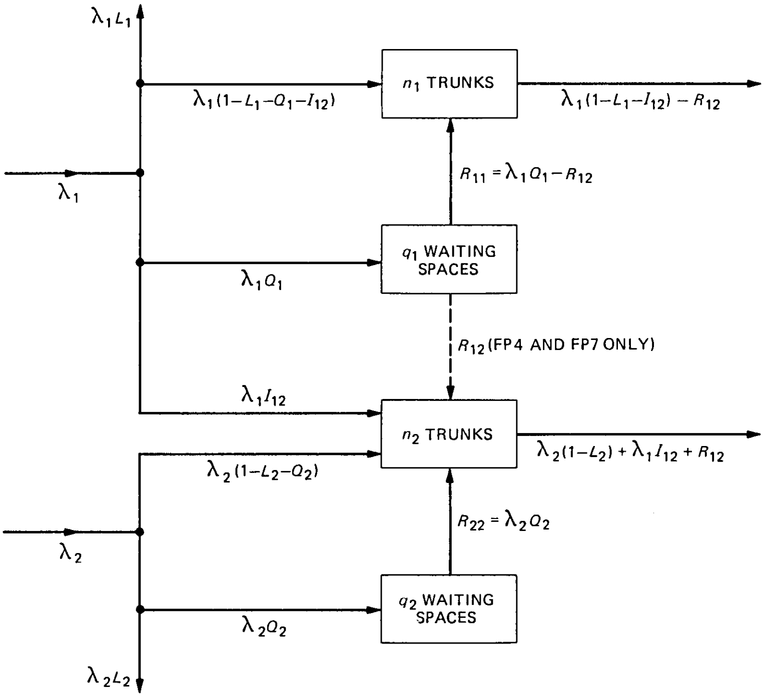


Fig. 1—Mean flow rates for an overflow system with queuing.

available, otherwise it is blocked and cleared from the system. The treatment of calls waiting in the primary queue depends on which feature package is being modeled. Corresponding to FP8, calls that are waiting in the primary queue are not allowed to overflow to the secondary group, but must wait for a trunk in the primary group to become available. Corresponding to FP4 (or FP7), calls waiting in the primary queue may be served by an idle trunk in the primary group, or by an idle trunk in the secondary group, provided that there are no calls waiting in the secondary queue, which have priority. The overflow systems corresponding to FP8 and FP4 are depicted schematically in Figs. 2 and 3, respectively.

The two cases considered above, although an idealization of the actual situation, embody the essential features of the packages. For comparison, we consider a third case, which we denote by FP0 and is depicted schematically in Fig. 4. In this case, if all n_1 trunks in the primary group are busy when a call in S_1 arrives, the call is placed in the primary queue if one of the q_1 waiting spaces is available. As with FP8, a call that is queued in the primary must wait for a trunk in the primary group to become available. If all n_1 trunks in the primary group are busy and all q_1 waiting spaces are occupied when a call in S_1

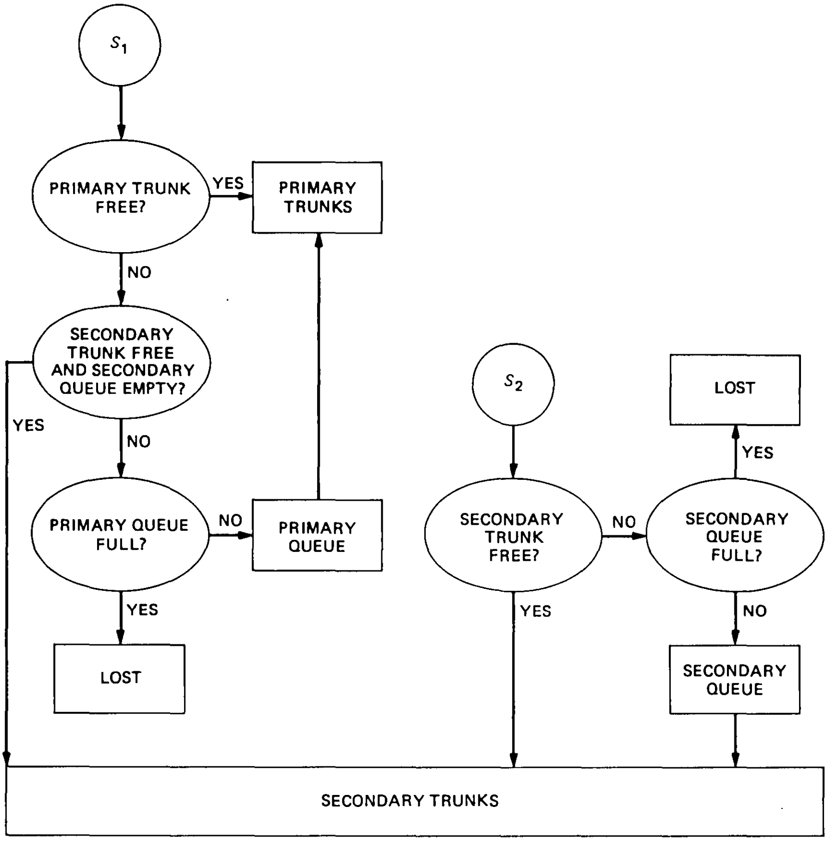


Fig. 2—Schematic of the overflow system corresponding to FP8.

arrives, it is placed in service in the secondary if there is a trunk available and there are no calls waiting in the secondary queue, otherwise it is blocked and cleared from the system. Note that no overflow is permitted if there is a waiting space available in the primary queue. This restriction was invoked by Anderson.[4]

We remark that the systems corresponding to FP4 and FP0 are particular cases of the system considered by Rath in connection with ACD-ESS (automatic call distributor-ESS).[5] He considered a system composed of two queues, in which one of the queues is allowed to overflow to the other, under specified conditions involving the queue lengths. He obtained numerical solutions in the case corresponding to FP4 by using a Gauss-Seidel iteration technique. He also developed an approximate procedure for analyzing his system based on the use of the interrupted Poisson process (IPP).[6] An approximate analysis of the system corresponding to FP4 has been given by Crater under the assumption that the number of waiting spaces in each queue is unlimited.[7] More recently, Shulman described a method of iteration and successive approximation, using the IPP as a traffic model, for analyzing the system corresponding to FP8 with several bands of groups.[8] He has used this method in the optimal design of facilities for *Dimension* PBX with overflow and queuing features.

Section II outlines the numerical procedures for evaluating the quantities of interest, based on the analytical results of Morrison.[2,3] In Section III two iterative techniques used by Kaufman to obtain numerical results are discussed.[1] The numerical results are discussed in Section IV. These results were obtained by Morrison's method and at least one of Kaufman's two methods.

II. AN ANALYTICAL PROCEDURE

We begin by outlining the numerical procedures for evaluating the quantities of interest based on the analytical results of Morrison.[2,3]

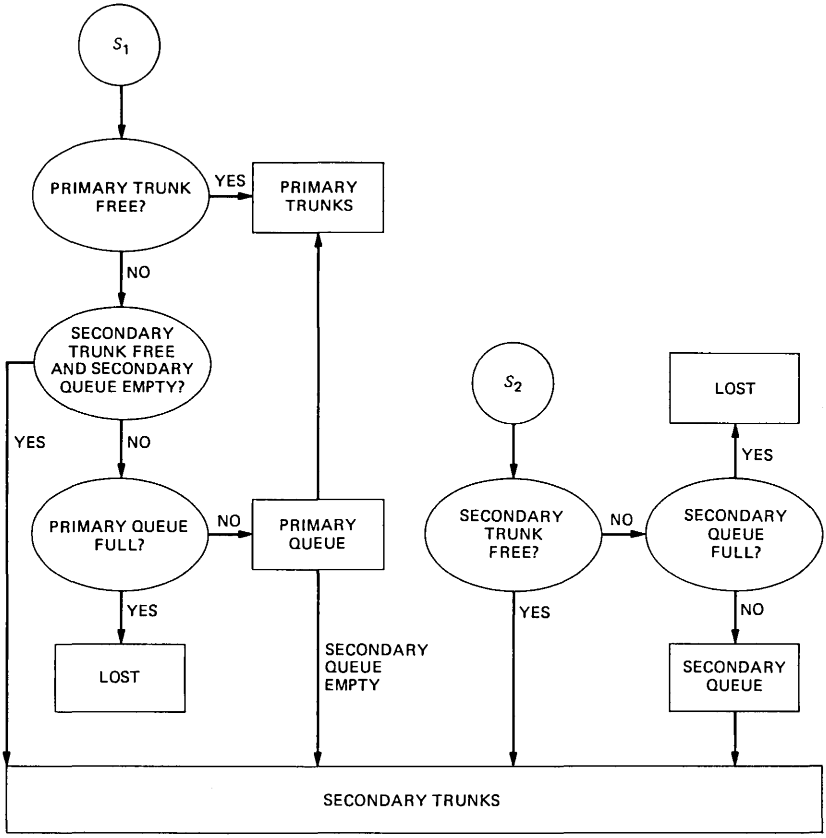


Fig. 3—Schematic of the overflow system corresponding to FP4.

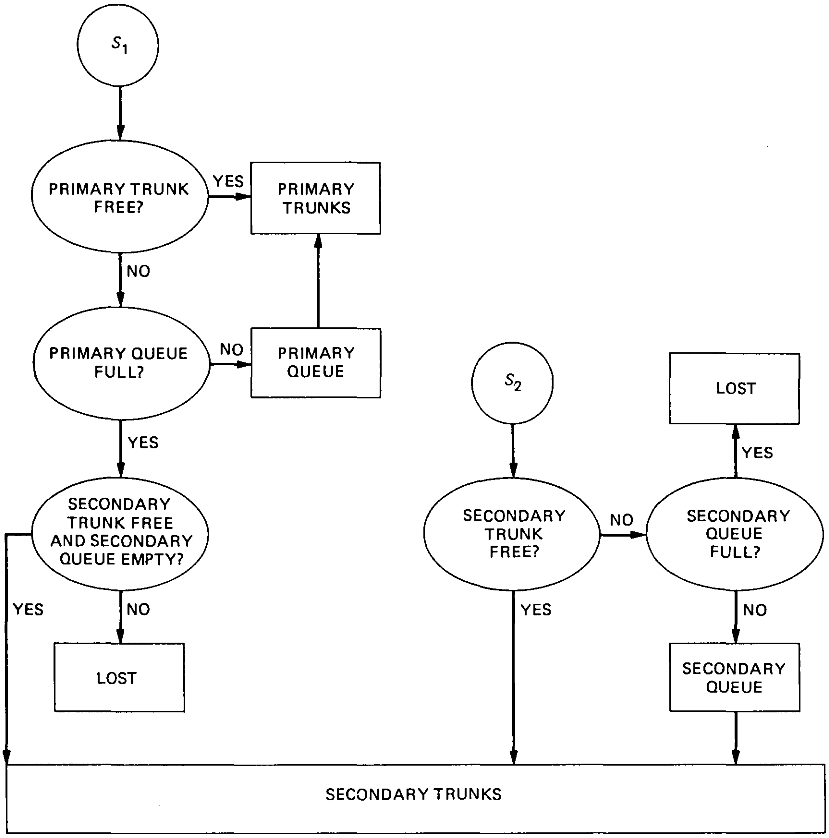


Fig. 4—Schematic of the overflow system designated as FP0.

The interested reader may consult these papers for the details of the analysis. Let p_{ij} denote the steady-state probability that there are i calls in the primary and j calls in the secondary, either in service or waiting. These probabilities satisfy a set of generalized birth-and-death equations, which take the form of partial difference equations connecting nearest-neighboring states. The basic technique is to separate variables in regions away from certain boundaries of the state space. This leads to some eigenvalue problems for the separation constant. The eigenvalues are roots of polynomial equations, and they may be evaluated numerically with the help of some interlacing properties.[2] The probabilities p_{ij} are then represented in terms of the corresponding eigenfunctions, which can be calculated from simple recurrence relations. The constant coefficients in these representations are determined from the boundary conditions (one of which is redundant), and the normalization condition that the sum of the probabilities is unity.

When engineering the system, there are various steady-state quantities of interest that can be expressed in terms of the probabilities p_{ij}. The quantities include the blocking (or loss) probabilities L_1 and L_2, and the average delays (or mean waiting times), W_1 and W_2, of calls that enter the queues. These quantities may be expressed directly in terms of the constant coefficients that occur in the representations for the probabilities p_{ij}. Thus the steady-state quantities of interest can be calculated directly, once the coefficients have been determined from the boundary and normalization conditions, without having to calculate the probabilities p_{ij}.

For FP0 there is just one set of $n_2 + q_2 + 1$ eigenvalues and corresponding eigenfunctions. There are then $n_2 + q_2 + 1$ constants to be determined numerically from the boundary and normalization conditions. We remark that in this case there are $(n_1 + q_1 + 1)(n_2 + q_2 + 1)$ probabilities p_{ij}, so that the analytical approach considerably reduces the dimensions of the problem. For FP4 there is an additional set of q_1 eigenvalues, and corresponding eigenfunctions. In this case there are $q_1 + n_2 + q_2 + 1$ constants to be determined numerically from the boundary and normalization conditions. This compares with the $q_1(q_2 + 1) + (n_1 + 1)(n_2 + q_2 + 1)$ probabilities p_{ij}. We note that there are fewer nonzero probabilities p_{ij} for FP4 since it is impossible for calls to be waiting in the primary queue when there is an idle trunk in the secondary and no calls are waiting in the secondary queue. For FP8 this is not the case, and the probabilities p_{ij} in the corresponding region of state space are expressed in terms of q_1 additional constants and a fundamental solution of a partial difference equation. However, it is possible to solve for these additional constants in terms of the other $q_1 + n_2 + q_2 + 1$ constants, by inversion of a triangular matrix.

Some conservation relations were used as a check on the accuracy of the numerical calculations. These involve the steady-state quantities depicted in Fig. 1. Thus Q_1 and Q_2 are the probabilities that calls from streams S_1 and S_2, respectively, are queued upon arrival. The mean departure rate from the primary queue to the primary trunks is R_{11}, and the mean rate of overflow from the primary queue to the secondary trunks is R_{12}. Note that $R_{12} = 0$ for both FP0 and FP8. The mean departure rate from the secondary queue, to the secondary trunks, is R_{22}. Since the mean rate of arrival of calls at each queue is equal to the mean departure rate, we have

$$\lambda_1 Q_1 = R_{11} + R_{12}, \qquad \lambda_2 Q_2 = R_{22}. \tag{1}$$

Also, I_{12} is the probability that a call from stream S_1 overflows immediately. Moreover, let X_1 and X_2 denote the average number of calls in service in the primary and secondary groups, respectively. According to Little's formula, the average number in a queuing system is equal to the average arrival rate to that system times the average time spent in that system.[9] If we apply this result to the primary and secondary trunk groups, it follows that

$$\mu X_1 = \lambda_1(1 - L_1 - I_{12}) - R_{12}, \qquad \mu X_2 = \lambda_2(1 - L_2) + \lambda_1 I_{12} + R_{12}, \quad (2)$$

since $1/\mu$ is the mean holding time. Since the various quantities in (1) and (2) may be expressed in terms of the constant coefficients that occur in the representations for the probabilities p_{ij},[2,3] these relationships provide a useful numerical check.

A package of computer programs has been developed by Seery to calculate the various steady-state quantities of interest, based on the procedures outlined above. The documentation provides the information necessary to use the programs.[10]

III. SPARSE MATRIX TECHNIQUES

The birth-and-death equations which determine the steady-state probabilities, and which are mentioned in the beginning of Section II, may be written as

$$A\mathbf{p} = \mathbf{0}, \quad (3)$$

where A is a singular, nonsymmetric matrix with negative off-diagonal elements and column sums equal to zero. It has only five nonzero diagonals.

For example, for FP4 when $n_1 \geq 1$, $n_2 \geq 1$, $k_1 = n_1 + q_1$, and $k_2 = n_2 + q_2$, the matrix A is defined by the equations

$$[\lambda_1(1 - \delta_{ik_1}\chi_{j-n_2}) + \lambda_2(1 - \delta_{jk_2}) + \mu \min(i, n_1) + \mu \min(j, n_2)]p_{ij}$$

$$= (1 - \chi_{i-1-n_1}\chi_{n_2-1-j})[\lambda_1(1 - \delta_{i0})p_{i-1,j} + \mu(1 - \delta_{jk_2})\min(j + 1, n_2)p_{i,j+1}]$$

$$+ (1 - \delta_{j0})[\lambda_1\delta_{in_1}\chi_{n_2-j} + \lambda_2(1 - \chi_{i-1-n_1}\chi_{n_2-j})]p_{i,j-1}$$

$$+ \mu(1 - \delta_{ik_1})[(1 - \chi_{i-n_1}\chi_{n_2-1-j}) \min(i + 1, n_1) + n_2\chi_{i-n_1}\delta_{jn_2}]p_{i+1,j},$$

where

$$\delta_{rm} = \begin{cases} 1, & r = m \\ 0, & r \neq m \end{cases} \quad \text{and} \quad \chi_r = \begin{cases} 1, & r \geq 0 \\ 0, & r < 0 \end{cases}$$

and $0 \leq i \leq k_1$ and $0 \leq j \leq k_2$.

We will describe two iterative techniques for solving (3). The first method, based on inverse iteration, requires a few expensive iterations. The second, based on splitting A into the sum of two matrices, requires many inexpensive iterations.

Inverse iteration may be written as follows:

```
Pick $\mathbf{p}^{(0)}$, an approximate null vector of $A$.
Iterate until convergence:
For  $k = 1, 2, \cdots,$
        solve $A\mathbf{v}^{(k)} = \mathbf{p}^{(k-1)}$  for  $\mathbf{v}^{(k)}$,        (4)
                set $\mathbf{p}^{(k)} = \mathbf{v}^{(k)}/\|\mathbf{v}^{(k)}\|_1$.
```

Usually only two iterations are required even if the initial guess is a random vector. Because of the near singularity of the matrix A represented in the computer, the vectors $\mathbf{v}^{(k)}$ will be very large; they will also become richer in the direction of the null space of the A matrix.[11]

Using a linear equation solver designed for band matrices, eq. (4) may be solved in approximately $3k_1k_2\max(k_1, k_2)$ locations. When $k_1k_2 > 500$, using a sparse matrix solver will require fewer operations and less space. However for larger problems (i.e., $k_1k_2 > 1000$) it pays to use some of the algebraic structure of the problem. For example, for FP4 the matrix A and the vectors $\mathbf{p}$ and $\mathbf{v}$ can be permuted and partitioned so that (4) becomes

$$\begin{pmatrix} BOX \\ OCY \\ ZQE \end{pmatrix} \begin{pmatrix} \mathbf{v}_A^{(k)} \\ \mathbf{v}_B^{(k)} \\ \mathbf{v}_C^{(k)} \end{pmatrix} = \begin{pmatrix} \mathbf{p}_A^{(k-1)} \\ \mathbf{p}_B^{(k-1)} \\ \mathbf{p}_C^{(k-1)} \end{pmatrix},$$

where $\mathbf{p}_A$ corresponds to p_{ij} where $0 \le i < n_1$ and $0 \le j \le k_2$, $\mathbf{p}_B$ corresponds to p_{ij} where $n_1 < i \le k_1$ and $n_2 < j \le k_2$, and $\mathbf{p}_C$ corresponds to p_{ij} where $i = n_1$, $0 \le j \le k_2$ and $j = n_2$, $n_1 < i \le k_1$. The matrices B and C have the form

$$B = \begin{pmatrix} S_0 & F_0 & & & \\ H & S_1 & F_1 & & \\ & \cdot & \cdot & \cdot & \\ & & H & S_{n_1-2} & F_{n_1-2} \\ & & & H & S_{n_1-1} \end{pmatrix}, \qquad C = \begin{pmatrix} G_{n_1+1} & J & & & \\ K & G_{n_1+2} & J & & \\ & \cdot & \cdot & \cdot & \\ & & K & G_{k_1-1} & J \\ & & & K & G_{k_1} \end{pmatrix},$$

where

$$F_i = -\mu(i+1)I_{k_2+1\times k_2+1}, \qquad J = -\mu n_1 I_{q_2\times q_2},$$

$$H = -\lambda_1 I_{k_2+1\times k_2+1}, \qquad K = -\lambda_1 I_{q_2\times q_2},$$

$$S_i = S + \gamma_i I_{k_2+1\times k_2+1}, \quad \text{and} \quad G_i = \gamma_i I_{q_2\times q_2} + G,$$

where

$$\gamma_i = \begin{cases} \lambda_1 + \mu\min(i, n_1), & \text{if} \quad i < k_1, \\ \mu n_1, & \text{if} \quad i = k_1, \end{cases}$$

and S and G have the form

$$S = \begin{pmatrix} x_0 & e_0 & & & \\ -\lambda_2 & x_1 & e_1 & & \\ & \cdot & \cdot & \cdot & \\ & & -\lambda_2 & x_{k_2-1} & e_{k_2-1} \\ & & & -\lambda_2 & x_{k_2} \end{pmatrix}, \qquad G = \begin{pmatrix} x_{n_2+1} & e & & & \\ -\lambda_2 & x_{n_2+2} & e & & \\ & \cdot & \cdot & \cdot & \\ & & -\lambda_2 & x_{k_2-1} & e \\ & & & -\lambda_2 & x_{k_2} \end{pmatrix},$$

where

$$x_j = \begin{cases} \lambda_2 + \mu \min(j, n_2), & \text{if } \ j < k_2, \\ \mu n_2, & \text{if } \ j = k_2, \end{cases}$$

$$e_j = -\mu \min(j + 1, n_2),$$

$$e = -\mu n_2.$$

The matrices X, Y, Z, Q, and E are very sparse (there are only $k_2 + 1$ nonzero elements in X) but they do not possess any relevant algebraic structure as B and C do. The eigendecomposition of B can be expressed in terms of the eigendecomposition of an $n_1 \times n_1$ tridiagonal matrix and the eigendecomposition of the $(k_2 + 1) \times (k_2 + 1)$ tridiagonal matrix S. Similarly the eigendecomposition of C can be expressed in terms of the eigendecomposition of a $q_1 \times q_1$ tridiagonal matrix and the $q_2 \times q_2$ tridiagonal matrix G. Using block Gaussian elimination and the eigendecompositions of B and C, solving (4) entails formulating and solving a dense system of $k_2 + 1 + q_1$ equations.

The second method, line successive overrelaxation (SOR), is much easier to implement and can be easily generalized to more queues. it uses the fact that A and $\mathbf{p}$ can be permuted and partitioned so that (3) can be written as

$$\begin{pmatrix} T_0 & E_0 & & & \\ D_1 & T_1 & E_1 & & \\ & \cdot & \cdot & \cdot & \\ & & D_{k_1-1} & T_{k_1-1} & E_{k_1-1} \\ & & & D_{k_1} & T_{k_1} \end{pmatrix} \begin{pmatrix} \mathbf{p}_0 \\ \mathbf{p}_1 \\ \\ \\ \mathbf{p}_{k_1} \end{pmatrix} = 0,$$

where $\mathbf{p}_i$ corresponds to p_{ij}, $0 \le j \le k_2$. The T matrices are tridiagonal and the E and D matrices are diagonal. The exact elements of the matrices depend on the feature package modeled.

In line SOR, initial vectors $\mathbf{p}_i^{(0)}$ and a parameter ω are chosen and the following iteration rule is executed:

```
For $k = 1, 2, \cdots$ until convergence,
    for $i = 0, 1, 2, \cdots, k_1$,
        solve $T_i\mathbf{v} = D_i\mathbf{p}_{i-1}^{(k+1)} + E_i\mathbf{p}_{i+1}^{(k)}$ for $\mathbf{v}$,
        set $\mathbf{p}_i^{(k+1)} = \mathbf{p}_i^{(k)} + \omega(-\mathbf{v} - \mathbf{p}_i^{(k)})$.
```

```
After convergence,
    set $\mathbf{p} \leftarrow \mathbf{p}/|\mathbf{p}|_1$.
```

The off-diagonal elements are never stored, but generated each time. Each iteration requires about $5m$ multiplications, $7m$ additions, and $2m$ divisions, where m is the size of the state space. The number of iterations depends on the ratios of the λ's to μ's, the size of the problem, the choice of ω, the feature package, and the required accuracy. Most of the examples given in Section IV required between 50 and 150 iterations for problems of between 500 and 2500 unknowns.

When $\omega = 1$, line SOR is called block Gauss–Seidel and is known to converge (see Kaufman[1]). For our problems a 10 percent change in ω sometimes doubles the number of iterations, as Table I illustrates. Usually the optimal ω is about 1.7 and choosing ω above 1.9 causes divergence. A heuristic algorithm was developed for adjusting ω as the iterative scheme progresses. The algorithm uses the theory developed in Ref. 12 to obtain an overestimate of ω and takes into consideration the fact that as ω is increased, the relative error tends to initially increase before decreasing if the choice of ω yields a convergent scheme.

Various other splitting schemes have been tried. Chebychev acceleration with Gauss–Seidel preconditioning requires about the same amount of computational effort as line SOR with the optimal ω.[12] Chebychev acceleration also requires the estimation of certain unknown parameters but the course of the algorithm seems to be much less sensitive to their values. Chebychev acceleration with an incomplete LU preconditioning has been successful for problems, such as the ones we have, in which the graph underlying the matrix is two-cyclic. A block SOR method in which each diagonal block is a separable matrix has also been tried. For the problems we have considered, the increase in the amount of work per iteration is not compensated by the decrease in the number of iterations.

Table I—ω vs number of iterations for FP0, $\mu = 1$, $\lambda_1 = 40$, $\lambda_2 = 36$, $n_1 = 40$, $n_2 = 40$, $q_1 = 10$, $q_2 = 10$, stopping criteria $\sim 10^{-9}$

ω	No. of Iterations
1.60	184
1.70	119
1.75	94
1.80	114
1.90	217

IV. NUMERICAL RESULTS

The criteria we use for engineering the system are that the blocking probabilities, L_1 and L_2, and the average delays, W_1 and W_2, of calls which enter the queues, do not exceed specified values. For prescribed loads $a_1 = \lambda_1/\mu$ and $a_2 = \lambda_2/\mu$, the aim is to choose the number of primary and secondary trunks n_1 and n_2, and the number of waiting spaces q_1 and q_2, so that the criteria are met. The procedure used was to select values of n_1 and n_2, and then determine the smallest values of q_1 and q_2 for which the criteria on the blocking probabilities are met. The process was repeated for different sets of values of n_1 and n_2, to find sets for which the criteria on the average delays are also satisfied. A search was made to determine the smallest total number of trunks $n_1 + n_2$ for which the criteria on both the blocking probabilities and the average delays are met, with appropriate choices of q_1 and q_2.

In the numerical results given below, we take $\mu = 1$, so that the average delays are given in units of mean holding time. Two sets of results were obtained. The first set corresponds to primary and secondary loads $a_1 = 3$ and $a_2 = 8$, and the desired criteria were $\max(L_1, L_2) \leq 0.01$ and $\max(W_1, W_2) \leq 1$. The results in Table II, which correspond to four primary and nine secondary trunks, indicate how changes in the number of waiting spaces affect the blocking probabilities and the average delays. The results of the search to minimize the total number of trunks required are depicted in Table III. In none of the cases were we able to satisfy the criteria with a total of only 12 trunks. Moreover, for both FP8 and FP0, for a total of 13 trunks we required 4 in the primary group and 9 in the secondary group. This was not the case for FP4. However, if we take into account the fact that primary trunks are less expensive than secondary ones, then for FP4 we would choose the set with four in the primary group

Table II—Blocking probabilities and average delays for offered loads $a_1 = 3$ and $a_2 = 8$, and unit mean holding time, $n_1 = 4$ and $n_2 = 9$ trunks, and different values of q_1 and q_2

FP	q_1	q_2	10^2L_1	10^2L_2	W_1	W_2
8	6	18	1.063	1.027	0.612	0.727
8	7	18	0.725	1.027	0.650	0.727
8	6	19	1.076	0.904	0.613	0.748
8	7	19	0.735	0.905	0.651	0.748
0	7	17	1.226	1.070	0.730	0.705
0	8	17	0.907	1.069	0.777	0.705
0	7	18	1.232	0.943	0.730	0.727
0	8	18	0.911	0.941	0.777	0.727
4	6	18	0.945	1.059	0.519	0.727
4	7	18	0.637	1.060	0.549	0.727
4	6	19	0.959	0.932	0.521	0.748
4	7	19	0.648	0.934	0.552	0.748

and nine in the secondary group. Also listed in Table III are other steady-state quantities of interest, as defined in Section II. In addition, $O_{12} = \lambda_1 I_{12} + R_{12}$ is the total mean overflow rate from the primary group to the secondary group.

It is of interest to compare the results for the three cases in Table III corresponding to $n_1 = 4$ and $n_2 = 9$, bearing in mind the differences in the number of waiting spaces and the results in Table II. Calls arriving at the primary group are most likely to be queued, and (by far) least likely to overflow (immediately), for FP0, as is to be expected, since overflow is permitted only when the queue is full. Moreover, the average delay in the primary queue is largest for FP0, and larger for FP8 than for FP4, since for FP8 no overflow is permitted from the primary queue. Also, the total mean overflow rate for FP4 exceeds that for FP8, although the immediate overflow probability is smaller for FP4. It is apparently the capacity for overflow from the primary queue which accounts for the other solutions for FP4 with a total of 13 trunks.

The second set of results corresponds to primary and secondary loads $a_1 = 10$ and $a_2 = 5$, and the desired criteria were $\max(L_1, L_2) \leq 0.005$ and, at first, $\max(W_1, W_2) \leq 0.8$. The results of the search with a total of 17 trunks are depicted in Table IV. There are no solutions for FP0 which satisfy the criteria, and just two for FP8. Three solutions are given for FP4. Although it is not possible to satisfy the criteria with a total of 17 trunks and more than 10 in the primary group, it is expected that there are solutions with less than 8 in the primary group, but we have not checked this. There are no solutions for FP4 or FP8 with a total of 16 trunks which satisfy the criteria. If we relax the

Table III—Steady-state quantities for offered loads $a_1 = 3$ and $a_2 = 8$, and unit mean holding time, with $\max(L_1, L_2) \leq 0.01$ and $\max(W_1, W_2) \leq 1$

FP	8	0	4	4	4	4
n_1	4	4	4	3	2	1
n_2	9	9	9	10	11	12
q_1	7	8	6	8	10	11
q_2	19	18	19	12	9	7
10^2L_1	0.735	0.911	0.959	0.955	0.890	0.890
10^2L_2	0.905	0.941	0.932	0.831	0.810	0.934
W_1	0.651	0.777	0.521	0.679	0.805	0.861
W_2	0.748	0.727	0.748	0.411	0.284	0.214
Q_1	0.293	0.477	0.288	0.351	0.391	0.413
Q_2	0.682	0.621	0.703	0.563	0.492	0.451
I_{12}	0.080	0.004	0.056	0.142	0.258	0.400
R_{11}	0.878	1.430	0.743	0.716	0.545	0.291
R_{12}	0	0	0.123	0.336	0.628	0.950
R_{22}	5.455	4.967	5.621	4.506	3.938	3.607
O_{12}	0.241	0.012	0.289	0.762	1.401	2.151
X_1	2.737	2.960	2.682	2.209	1.572	0.823
X_2	8.169	7.937	8.215	8.695	9.336	10.076

Table IV—Steady-state quantities for offered loads $a_1 = 10$ and $a_2 = 5$, and unit mean holding time, with $\max(L_1, L_2) \leq 0.005$ and $\max(W_1, W_2) \leq 0.8$

FP	8	8	4	4	4
n_1	10	9	10	9	8
n_2	7	8	7	8	9
q_1	20	22	19	20	21
q_2	11	8	11	9	7
10^2L_1	0.481	0.473	0.473	0.480	0.445
10^2L_2	0.453	0.481	0.494	0.345	0.430
W_1	0.637	0.773	0.508	0.532	0.547
W_2	0.460	0.309	0.460	0.317	0.237
Q_1	0.433	0.421	0.464	0.474	0.481
Q_2	0.626	0.538	0.683	0.623	0.583
I_{12}	0.117	0.187	0.054	0.087	0.123
R_{11}	4.327	4.208	3.921	3.622	3.268
R_{12}	0	0	0.716	1.120	1.540
R_{22}	3.131	2.692	3.417	3.114	2.915
O_{12}	1.169	1.875	1.257	1.987	2.774
X_1	8.783	8.078	8.696	7.965	7.182
X_2	6.147	6.851	6.232	6.970	7.752

criteria on the average delays to $\max(W_1, W_2) \leq 1$, then additional solutions are obtained, as depicted in Table V. There is now one solution for FP0 with a total of 17 trunks, and two more solutions for FP8. Two additional solutions are given for FP4, one with a total of only 16 trunks (and 34 waiting spaces for the primary group). There are no solutions for FP8 or FP0 with a total of 16 trunks which satisfy the relaxed criteria.

Finally, setting aside the problem of engineering the system, we examined the effect on the blocking probabilities and the average

Table V—Steady-state quantities for offered loads $a_1 = 10$ and $a_2 = 5$, and unit mean holding time, with $\max(L_1, L_2) \leq 0.005$ and $0.8 < \max(W_1, W_2) \leq 1$

FP	8	8	0	4	4
n_1	11	8	11	11	10
n_2	6	9	6	6	6
q_1	18	24	24	18	34
q_2	18	7	17	19	19
10^2L_1	0.461	0.482	0.499	0.437	0.466
10^2L_2	0.495	0.354	0.456	0.428	0.471
W_1	0.523	0.954	0.753	0.478	0.983
W_2	0.883	0.237	0.866	0.898	0.898
Q_1	0.437	0.406	0.654	0.452	0.676
Q_2	0.761	0.479	0.579	0.795	0.874
I_{12}	0.054	0.263	0.002	0.025	0.020
R_{11}	4.373	4.060	6.540	4.187	6.191
R_{12}	0	0	0	0.335	0.568
R_{22}	3.807	2.397	2.895	3.973	4.371
O_{12}	0.540	2.633	0.017	0.587	0.771
X_1	9.414	7.319	9.933	9.370	9.183
X_2	5.515	7.615	4.994	5.565	5.747

Table VI—Blocking probabilities and average delays for offered loads a_1 and $a_2 = 0.9a_1$, unit mean holding time, $n_1 = 40 = n_2$ trunks, $q_1 = 10 = q_2$ waiting spaces, and different values of a_1

FP	a_1	10^2L_1	10^2L_2	W_1	W_2
0	42	5.355	3.654	0.1475	0.1259
0	40	2.906	2.022	0.1375	0.1162
0	38	1.291	0.954	0.1270	0.1065
0	36	0.449	0.378	0.1162	0.0970
8	42	3.796	4.774	0.1313	0.1259
8	40	1.758	2.845	0.1184	0.1162
8	38	0.639	1.436	0.1055	0.1065
8	36	0.177	0.596	0.0930	0.0970
4	42	3.588	5.020	0.1179	0.1259
4	40	1.642	2.995	0.1043	0.1162
4	38	0.590	1.510	0.0909	0.1065
4	36	0.161	0.623	0.0782	0.0970

delays of varying the loads offered to a prescribed system. We did this since in a given situation the configuration of the system is fixed, but one would expect that the loads would vary in the course of the day. We chose a configuration with 40 trunks and 10 waiting spaces in both the primary and the secondary groups, to show that our programs can handle larger problems than those listed so far. The primary and secondary loads were varied simultaneously with $a_2 = 0.9a_1$, and the results are given in Table VI. As expected, the blocking probabilities and the average delays increase with increasing load, the more significant effect being on the blocking probabilities.

For a given load, the blocking probability for calls from stream S_1 is larger for FP0 than for FP8, and slightly larger for FP8 than for FP4. On the contrary, the blocking probability for calls from stream S_2 is smaller for FP0 than for FP8, and slightly smaller for FP8 than for FP4. These orderings are not surprising in view of the fact that no overflow is permitted from the primary queue for either FP0 or FP8, and moreover overflow is permitted for FP0 only when the primary queue is full. As before, the average delay in the primary queue is largest for FP0, and larger for FP8 than for FP4. The average delay in the secondary queue is, of course, the same for FP0, FP4, and FP8.

V. ACKNOWLEDGMENTS

The authors are grateful to G. M. Anderson, J. F. Brown, and T. V. Crater for bringing this problem to their attention, and to J. McKenna for his continued encouragement. They are indebted to J. H. Rath for running his program to check their initial calculations for FP4 and to H. B. Shulman for suggesting the two examples for the engineering problem, and for providing nearly optimal results for FP8.

REFERENCES

1. L. Kaufman, "Solving Large Linear Systems Arising in Queuing Problems," to appear in the proceedings of the Bielefeld conference on large linear systems, Springer Verlag.
2. J. A. Morrison, "Analysis of Some Overflow Problems With Queuing," B.S.T.J., *59,* No. 8 (October 1980), pp. 1427–62.
3. J. A. Morrison, "An Overflow System in Which Queuing Takes Precedence," B.S.T.J., *60,* No. 1 (January 1981), pp. 1–12.
4. G. M. Anderson, "Facilities Design for Automatic Route Selection With Queuing," unpublished work.
5. J. H. Rath, "An Approximation for a Queueing System With Two Queues and Overflows," unpublished work.
6. A. Kuczura, "The Interrupted Poisson Process as an Overflow Process," B.S.T.J., *52,* No. 3 (March 1973), pp. 437–48.
7. T. V. Crater, "Calculation of Trunk Occupancy and Delay for Networks With Automatic Alternate Routing and Queueing," unpublished work.
8. H. B. Shulman, "Engineering Trunk Networks for Queueing and Overflow," unpublished work.
9. L. Kleinrock, *Queueing Systems, Volume I: Theory,* New York: Wiley, 1975.
10. J. B. Seery, "FP048-A Computer Package for Overflow Problems Motivated by *Dimension*® PBX Feature Packages," unpublished work.
11. J. H. Wilkinson, *The Algebraic Eigenvalue Problem,* London: Oxford University Press, 1965.
12. R. S. Varga, *Matrix Iterative Analysis,* Englewood Cliffs, New Jersey: Prentice-Hall, 1962.

THE BELL SYSTEM TECHNICAL JOURNAL
Vol. 60, No. 5, May–June 1981
Printed in U.S.A.

Impact of Forecast Uncertainty on Feeder-Cable Sizing

By N. H. NOE

(Manuscript received September 8, 1980)

This paper estimates the forecast error distribution for outside plant using data from the central office forecast measurement plan. We then determine the impact of the forecast errors on feeder-cable sizing by using this distribution to estimate the conditional distribution of engineered cable size with respect to optimum cable size. The marginal distribution of optimum cable size is estimated from a growth rate distribution which in turn is estimated from cable shipment data. We then use the resulting joint distribution to weight the percentage cost penalty of each possible combination of optimum and forecast size. The impact analysis is done separately for each gauge. By weighting by the million conductor feet of each gauge shipped, we then obtain an estimate of overall sizing-error cost penalty. The resulting penalty estimate is about 0.5 percent of the annual feeder-cable-construction program.

I. INTRODUCTION

A feeder route is a major network of cables extending from the central office to within ½ mile or so of customers.[1] When a feeder route needs relief, a cable size is selected with the goal of minimizing the discounted sum of costs over time. Because of forecast errors, however, sometimes a cable that is larger or smaller than the optimum is placed. One often hears that the feeder-cable sizing curves are so flat that sizing decisions are relatively insensitive to forecast errors. On the other hand, a small percentage of a large construction program still represents a substantial amount of money. In this paper we attempt to quantify the impact of forecast deviations on the feeder network by first estimating the error distribution and then using it to examine the effect of forecast errors on feeder-cable sizing. While it is not presented here, a preliminary study indicates that the impact of forecast deviations on feeder-cable relief timing is at least as great as that on sizing.

II. FORECAST ERROR DISTRIBUTION

In this section, we derive an estimate for the distribution of forecast errors using data from the central office forecast measurement plan. It should be noted that all deviations between forecast and actual are included here under the forecast error category. Thus the forecast errors include some deviations caused by count errors and others caused by boundary changes that have not been reflected accurately in the records.

2.1 Nomenclature

The units of primary concern are available pairs, which include both working and idle pairs. The data available for estimating the forecast error distribution, however, are in terms of main stations (plus equivalent main stations). The distribution will therefore be derived first in terms of main stations plus equivalent main stations and then converted into available pairs.

The basic items of interest are defined here:

b = base in-service or total value (the actual on which the forecast was based),

t = forecast interval, in years,

f = forecast in-service value, and

a = actual in-service value for the date for which the forecast was made.

Several important variables are derived from the above basic ones:

$$\epsilon = f - a = \text{forecast deviation,}$$

$$g_f = \frac{f - b}{t} = \text{forecast average of annual growth rate, and}$$

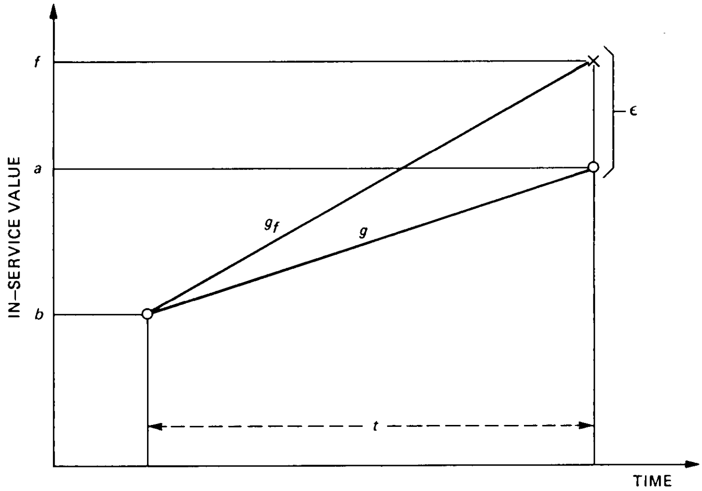
$$g = \frac{a - b}{t} = \text{actual average of annual growth rate.}$$

These items are shown in Fig. 1.

2.2 Data description

Several years ago, the central office forecast measurement plan (COFMP) was established to collect forecast data from the Bell System operating companies. These data were collected for short-term wire center forecasts. The data used in this study were collected in the fourth quarter of 1978. For each of 1266 wire centers, we had the following:

(*i*) identification (company, area, and wire center),



$\epsilon = f - a$ = FORECAST DEVIATION

$g_f = (f - b)/t$ = FORECAST GROWTH RATE

$g = (a - b)/t$ = ACTUAL GROWTH RATE

Fig. 1—Definitions of forecast variables.

(*ii*) b, f, and a, for $t = 1$, and

(*iii*) the number of main stations transferred from a wire center to another one during the forecast interval.

The values b, f, and a are in terms of main stations (plus equivalent main stations).

Table I gives examples of the above data items.

In addition to the above items, we had two items for each wire center that were not used in the study. For each wire center, the month of the end of the forecast period was available. Since in all cases, the month fell within a three-month period, we did not feel that the differences would be significant. A seasonal indicator was also available for each wire center. Several wire centers were flagged as

Table I—Examples of COFMP data

Company	Area	Wire Center	Main stations (plus equivalents)			
			Base	Forecast	Actual	Transfer
4	17	109	17785	18260	18400	+23
7	34	344	871	910	891	0
7	35	368	10277	10964	11021	0
10	40	430	3171	3240	3296	0
14	54	584	5715	5925	6013	0
14	56	596	2187	2365	2286	0
16	67	1141	40317	42205	42508	+374
17	68	1158	21618	24626	24378	0
18	73	1254	2054	2110	2100	0

having the annual maximum occur at some time other than the end of the year. This information was not needed for the study, since all forecast intervals were exactly one year.

Since it was desirable to model the forecast results in terms of growth rates, the forecast and actual values (f and a) were adjusted for any wire center that had a transfer by subtracting the signed value of the transfer from them. This allowed us to compare the forecast and actual growth rates for the original serving areas.

An initial examination of the data showed that one company had a much larger percentage of its wire centers represented than did any other company. Of the 1266 wire centers, 507 were from that company. The COFMP was intended to collect data only for wire centers that have at least 500 main and equivalent-main telephones and that have a traffic order prepared while the forecast is in effect. Eliminating wire centers that are less than 500 in size, and arbitrarily retaining every third wire center of size 500 or greater reduced the representation from that company to the point where it was similar to that of the other companies. At this point, we retained 863 of the 1266 wire centers, and felt that they provided a representative cross section of the Bell System. In Fig. 2, the forecast growth rate is plotted versus the actual growth rate for these 863 wire centers.

2.3 Model development

From Fig. 2, it is obvious that the variance is increasing with g. We tried both square root and log transformations and found that the log transformation, with a shift of 50 for both g_f and g, did an excellent job of stabilizing the variance.

To be able to use a shift of 50, we had to drop 12 data points with g_f and/or g less than or equal to -50. Six of these points were of relatively little interest, for both g_f and g were negative, and we expect the model to be used for cases where cable is placed to accommodate growth ($g_f > 0$) or where it should be placed ($g > 0$). The other six points had either g_f or g less than -60, with the other value positive (all except one were greater than $+120$). These six points would have been outliers and should have received small weights for any reasonable model. Therefore, their loss is not serious.

The remaining 851 data points are shown in Fig. 3, where $\ln(g_f + 50)$ is plotted against $\ln(g + 50)$. At a glance, it appears that the variance is now decreasing with g. That this is not the case will be shown later. Basically, the illusion is due to more points existing at the lower g values.

We used robust regression to estimate the relationship between g_f and g. After an initial ordinary least-squares step, we used a Huber iteration to downweight points with residuals more than 1.5 standard

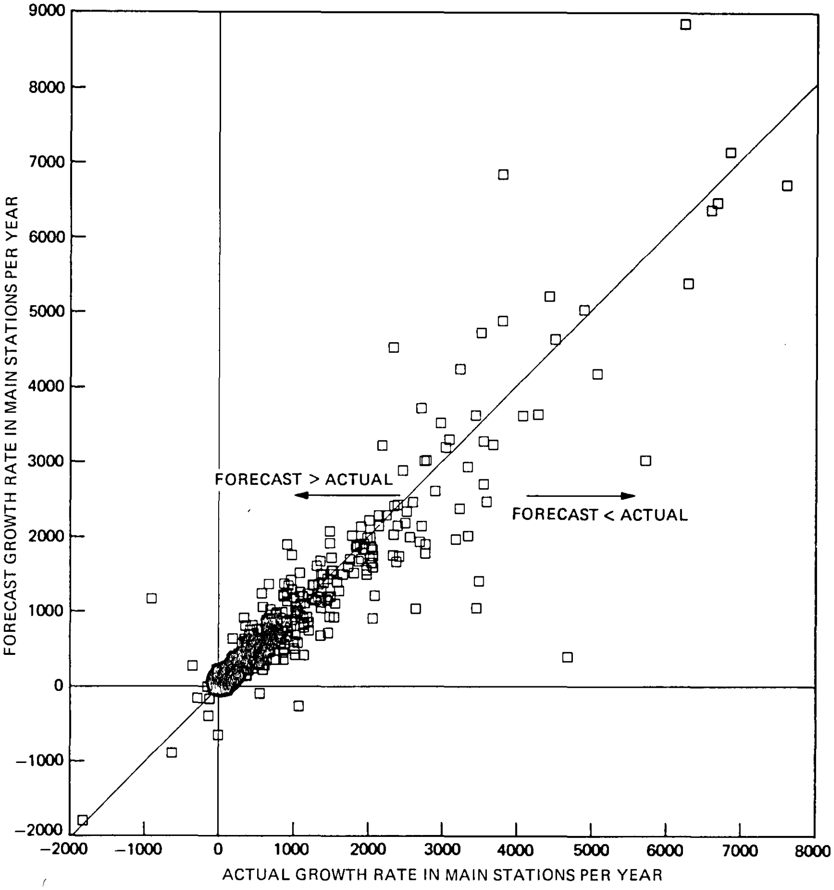


Fig. 2—Forecast vs actual growth rates.

deviations away,[2] using the estimate of the standard deviation obtained from the first step. We followed the Huber step by a biweight iteration,[3] using a dispersion value six times the Huber estimate of the standard deviation.

The growth rate model is

$$\ln(g_f + 50) = \alpha + \beta \ln(g + 50) + \nu, \qquad (1)$$

where α and β are parameters to be estimated and ν is a residual noise term with mean 0 and a variance σ^2 to be estimated. The estimates resulting from each step of the regression are given in Table II.

Figures 4 through 7 show residual plots for the residuals from the final step. The plots against the dependent and independent variables shown in Figs. 4 and 5 indicate reasonably well-behaved residuals. Although it would have been difficult to make use of any relationship

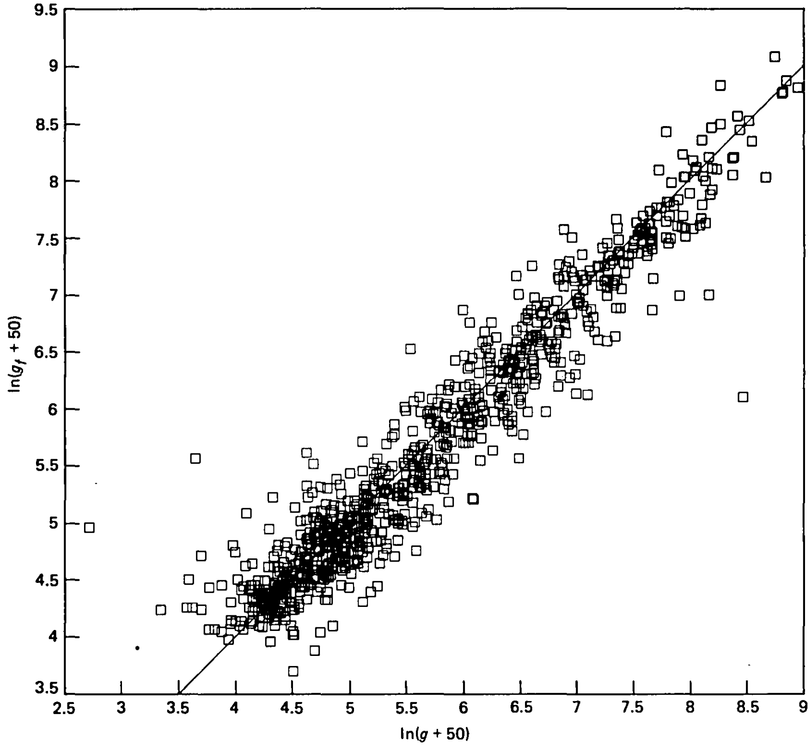


Fig. 3—$\ln(g_f + 50)$ vs $\ln(g + 50)$.

involving the size of an area for which a forecast is produced, the residuals were plotted versus the base size in Fig. 6. Figure 6 indicates that there is no structure involving the base size that needs to be included in the model. Finally, the residuals are shown for each of the 19 Bell System operating companies in Fig. 7. Here, too, there is no obvious need to include a company effect in the model. The larger extreme residuals generally occur for those companies with a larger number of data points.

It should be pointed out that Figs. 4 through 7 show only 847 of the 851 points. The other four points are shown in Table III. Three of these are outliers that received zero weight in the biweight iteration and one lies just beyond the range that was plotted.

Table II—Results of each step of the regression

Step	$\hat{\alpha}$	$\hat{\beta}$	$\hat{\sigma}$	R^2
Ordinary Least Squares	0.363	0.929	0.322	0.920
Huber	0.277	0.943	0.285	0.937
Biweight	0.250	0.948	0.274	0.942

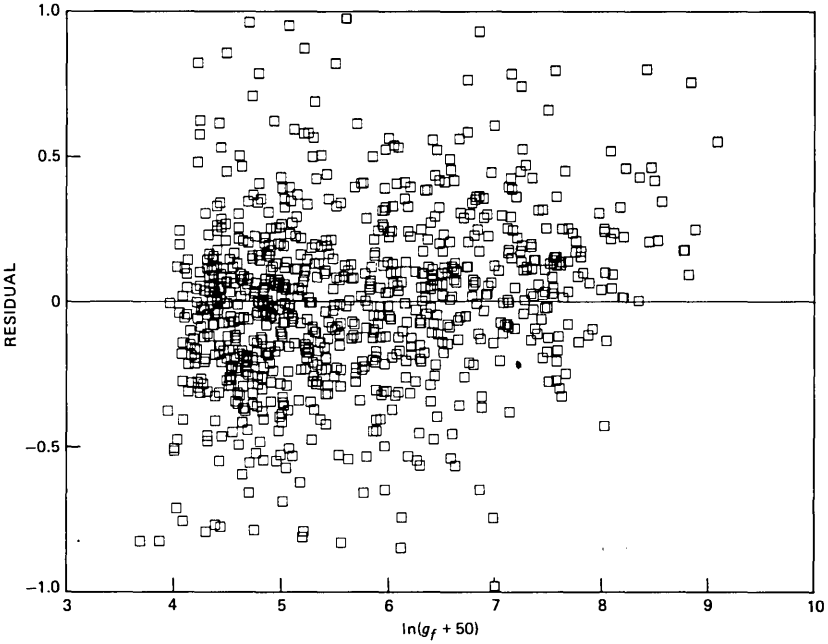


Fig. 4—Residual vs $\ln(g_f + 50)$.

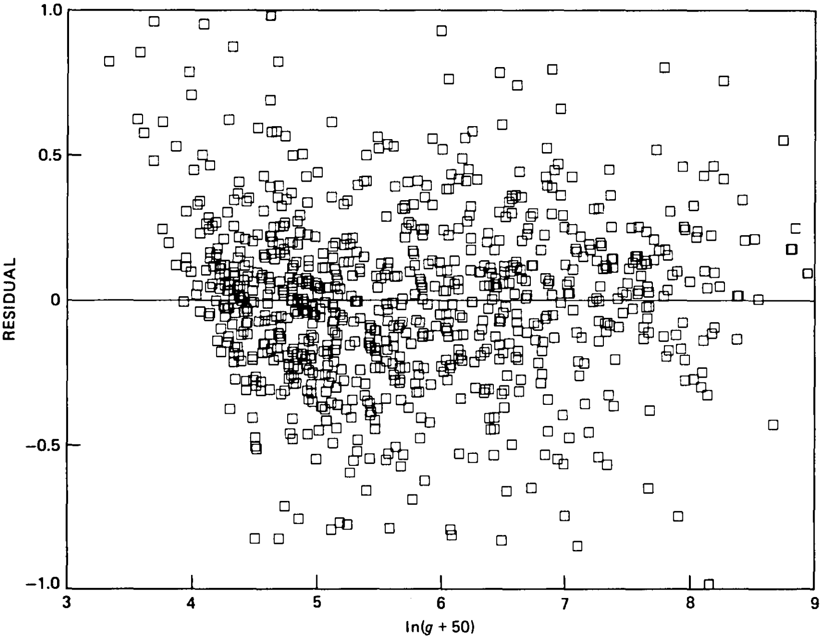


Fig. 5—Residual vs $\ln(g + 50)$.

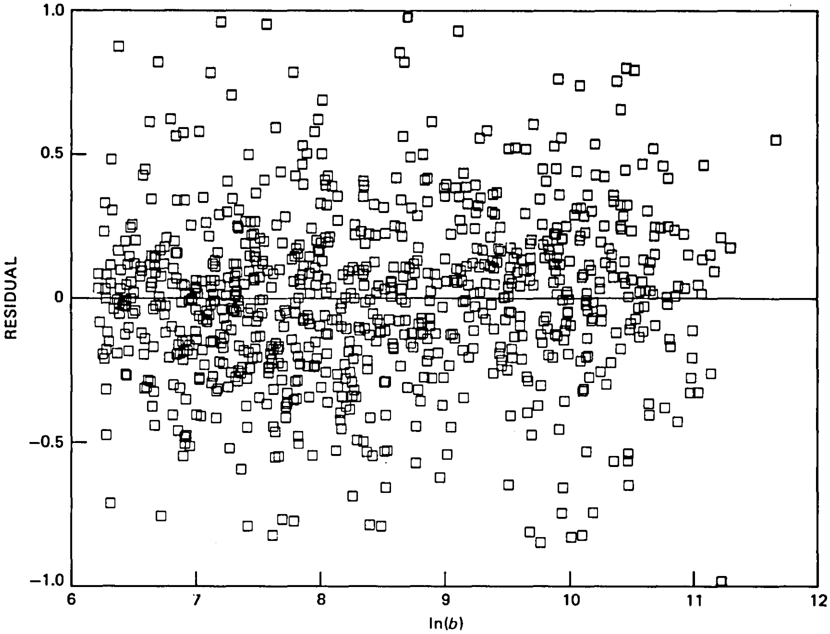


Fig. 6—Residual vs ln(b).

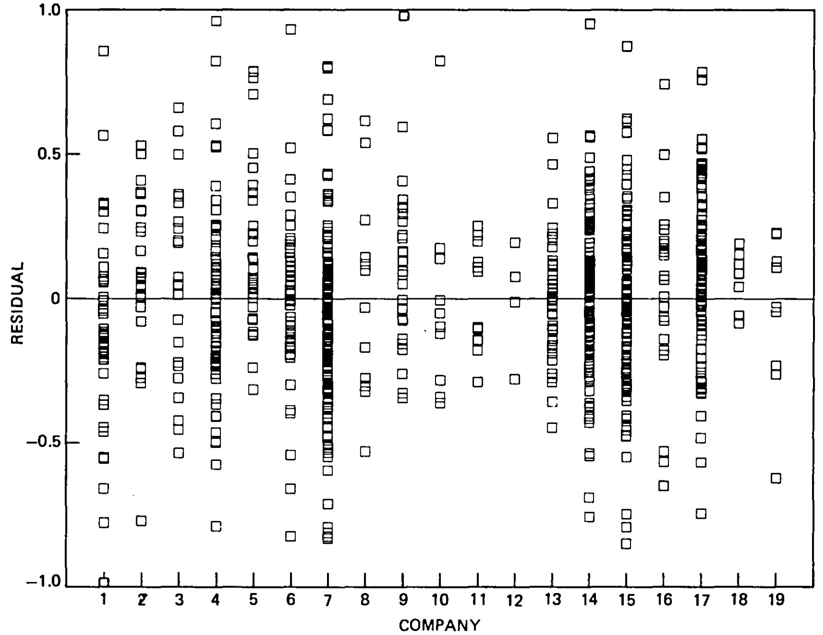


Fig. 7—Residual vs company.

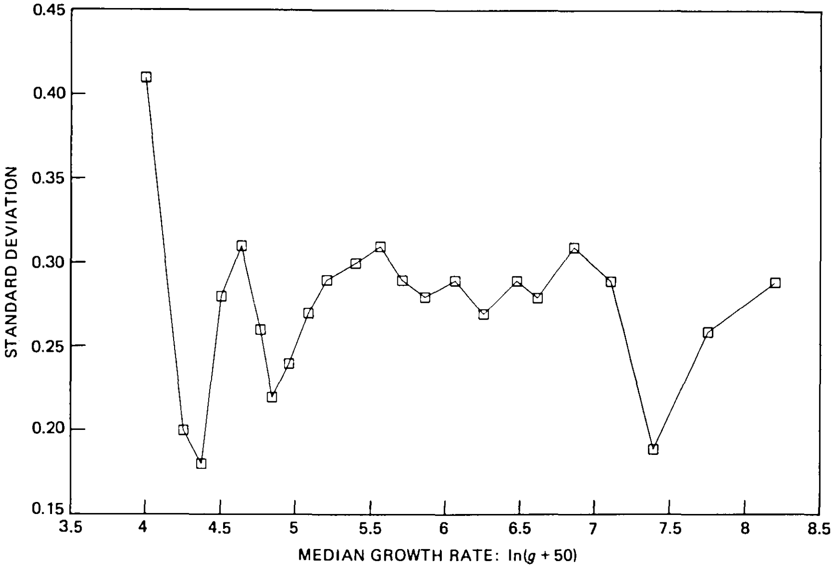


Fig. 8—Standard deviation vs ln(g + 50).

To determine if the variance of the residuals is sufficiently constant with respect to g, the 851 data points were ranked by g and divided into 23 groups of 37 points each. For each group, an unbiased estimate of the standard deviation was calculated, using the weights resulting from the biweight step in the regression. The resulting values are shown in Fig. 8, plotted versus the median values of ln(g + 50). No overall trend is obvious in Fig. 8 and regression confirms that it is reasonable to assume a constant variance.

A standardized deviate was found for each data point by subtracting the value predicted by the regression model (1) and dividing by the regression standard deviation,

$$d \equiv \frac{\ln(g_f + 50) - [0.250 + 0.948 \ln(g + 50)]}{0.274}. \tag{2}$$

A Q-Q plot of these deviates against the standard unit normal showed an excellent fit for the bulk of the data but with tails larger than given

Table III—Residuals for the four points not shown in Figs. 4 to 7

Wire Center	Residual	$\ln(g_f + 50)$	$\ln(g + 50)$	b	Company	Biweight Weight
128	2.14	4.96	2.71	46,328	4	0
138	1.86	5.56	3.64	15,232	4	0
153	−2.17	6.10	8.46	15,587	4	0
467	1.03	6.52	5.53	17,255	13	0.410

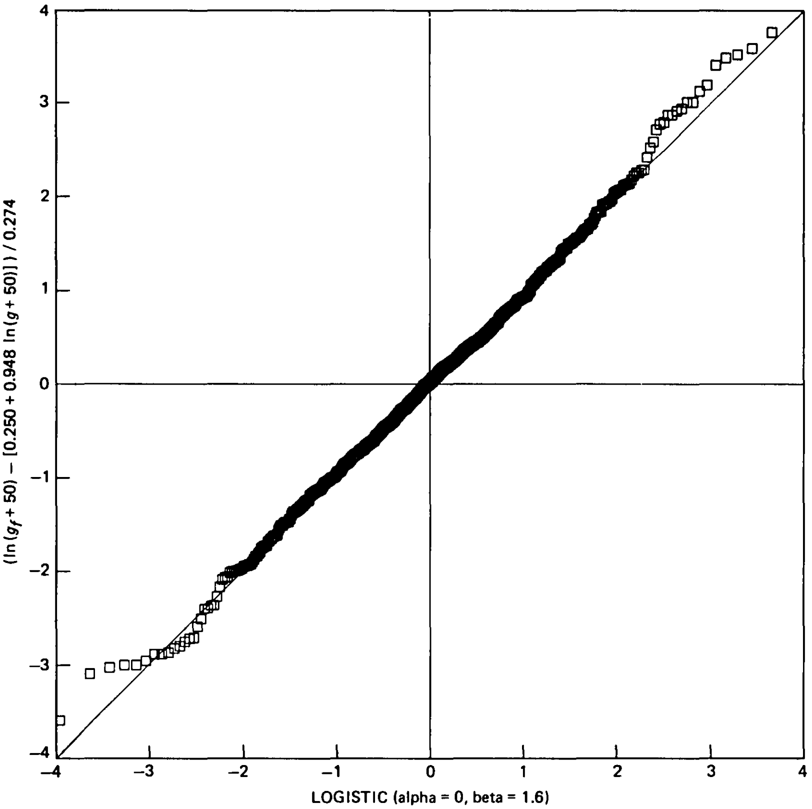


Fig. 9—*Q-Q* plot of empirical distribution vs logistic.

by the normal. Substituting for the normal, a logistic distribution with parameters $\alpha = 0$ and $\beta = 1.6$ (mean 0 and variance 1.29) resulted in the satisfactory *Q-Q* plot shown in Fig. 9. All but three of the 851 points are shown in Fig. 9. The other three are given in Table IV.

We conclude that it is reasonable to assume that the density and distribution functions of d are given by

$$f(d) = \frac{1.6\, e^{-1.6d}}{(1 + e^{-1.6d})^2} \quad \text{and} \quad F(d) = \frac{1}{1 + e^{-1.6d}}. \tag{3}$$

The modeling was done using COFMP data expressed in terms of main stations plus equivalent main stations, so all growth rates used so far have been in terms of main stations plus equivalent main stations per year. To study the impact of forecast errors on the feeder cable network, we need growth rates in terms of available pairs per year. If we now define $g^{(p)}$ to be a growth rate in terms of available pairs per year, and similarly define $g^{(m)}$ to be the corresponding growth rate in terms of main stations plus equivalent main stations per year, the relationship can be estimated as follows:

$$g^{(m)} = 0.65\, g^{(p)}. \tag{4}$$

The 0.65 ratio in the above expression is the 1977 Bell System average main frame fill. Since the actual fill is generally somewhat lower at cross sections out on the route than it is at the main frame, this ratio tends to slightly understate the effect of forecast errors on available pairs. Substituting (4) in (2) for both g_f and g and dropping the superscripts, gives

$$d = \frac{\ln(0.65g_f + 50) - [0.250 + 0.948 \ln(0.65g + 50)]}{0.274}, \tag{5}$$

where the growth rates are now in terms of available pairs per year.

From (3) and (5), we find that the conditional density and distribution functions of g_f with respect to g are

$$f(g_f|g) = \frac{3.8\, e^{-5.84y}}{(0.65g_f + 50)\,(1 + e^{-5.84y})^2},$$

and

$$F(g_f|g) = \frac{1}{1 + e^{-5.84y}}, \tag{6}$$

where

$$y = \ln(0.65g_f + 50) - [0.250 + 0.948 \ln(0.65g + 50)],$$

and where g_f and g are given in terms of available pairs per year. Expression (6) is used in Section III to estimate the impact of forecast errors on feeder-cable sizing.

First, however, three additional aspects of the model derivation need to be discussed. The COFMP data are all for a forecast interval of one year. How well does (6) work for other values of t? One might intuitively expect that forecast errors, even when normalized by divid-

Table IV—Three points omitted in Fig. 9

Wire Center	Empirical Value	Logistic Value
153	−4.65	−7.92
138	3.96	6.79
128	4.65	7.81

ing by the forecast interval as is done when dealing with growth rates, would be larger for longer forecast intervals. Indeed this tended to be the case for 22 wire centers from one company that had data on wire centers forecasts for 1-, 2-, 3-, and 4-year intervals. On the other hand, similar data from another company shows that the relative error tends to decrease for the longer intervals. An experienced forecaster was not surprised at this decrease and explained it as follows. It is often easier to determine the potential development for an area than to determine when an area will achieve that development. In view of the scant and conflicting evidence, we decided that (6) should be used for the forecast intervals encountered in feeder-cable sizing.

The forecast error distribution was derived from wire center data and is to be used in the feeder-cable network. Wire center forecasts are based in part on time series data that often do not exist for portions of a feeder route. Thus one would expect that the forecast deviations for outside plant forecasts may be somewhat larger than indicated by expression (6). In the absence of specific data, however, (6) is used as the estimate for outside plant forecast errors.

Also, if the main frame fill becomes lower than 0.65, the estimated forecast deviations in terms of available pairs will be somewhat greater than given by (6).

III. FORECAST ERROR IMPACT ON FEEDER-CABLE SIZING

We use the forecast error distribution derived in Section II to estimate for each gauge the conditional distribution of discrete-engineered cable sizes (with size based on forecast) with respect to each possible discrete optimum cable size based on actual growth. Important assumptions are that growth is linear and that there are no structure congestion problems or opportunities to use pair gain systems. If inflation is not considered, it would be appropriate to use a 12 percent discounting rate. A 6 percent inflation rate for underground cable leads to a 6 percent discounting rate when inflation is considered.

The marginal distribution of optimum cable sizes for each gauge is determined from the distribution of growth rate in that gauge, and that is, in turn, estimated from cable-shipment data. The conditional and marginal cable-size distributions give the joint distribution for each gauge of optimum and forecast (i.e., engineered) cable sizes.

The percentage cost penalty of each possible combination of optimum and forecast size is determined and multiplied by the probability of that combination occurring. These values are summed to give an overall sizing-error cost penalty for each gauge. When weighted by MCF (million conductor feet) of each gauge shipped, they provide an estimate of the cost impact of forecast errors on feeder-cable sizing.

The following parts of this section describe the steps in detail.

3.1 Growth rate distribution

An estimate of the marginal probability distribution of the optimum cable size is needed for each gauge. Were it not for the discounting rates used in the past by some companies to size cables, it would be possible to estimate these distributions directly from cable shipment data. This section estimates the growth rate distribution for each gauge, using cable shipment data and an economic sizing relationship that relates growth rate to cable size and gauge under the discounting rates used. In Section 3.2.1, we use the growth rate distribution derived here to estimate, for each gauge, the marginal probability distribution of the optimum cable size under the discounting rate that considers inflation.

1977 shipments of pulp-insulated exchange cable provide the base for estimating the growth-rate distributions of feeder cable. The cable shipment data give the MCF of each size and gauge shipped. Let $F(x_i)$ be the probability that an MCF of pulp cable of the gauge being considered is of size less than or equal to x_i.

The economic sizing relationship is used to relate points on these cable-size cumulative distribution functions to points on the growth-rate cumulative distribution functions. Assuming linear growth, the present worth cost of using a cable size, x, to meet a growth rate, g, is[4]

$$PW(x, g) = \frac{(a + bx)/r}{1 - e^{-rx/g}}, \tag{7}$$

where

r = the discounting rate,

a = the cable intercept cost (\$/year/sheath foot), and

b = the cable incremental cost (\$/year/pair foot).

The values of a and b used are the annual charge values for underground cables, since most feeder cables are placed in ducts:

gauge	a	b
26	0.38	0.0011
24	0.36	0.0014
22	0.33	0.0020
19	0.30	0.0036

(8)

It has been estimated that about one third of the companies considered inflation for sizing cables that were shipped in 1977. Thus we used a weighted average of the two discounting rates to estimate the growth rate distribution,

$$r = 0.10. \tag{9}$$

Let $g(x_i)$ be the growth rate such that the optimum discrete cable size is x_i for g just less than $g(x_i)$ and x_{i+1} for g just greater than $g(x_i)$. It can be found by iteratively solving $PW(x_i, g) = PW(x_{i+1}, g)$ for g, where x_{i+1} is the next-larger discrete cable size. Substituting from (7), $g(x_i)$ is the value that yields the following equality:

$$\frac{(a + bx_i)/r}{1 - \exp[-rx_i/g(x_i)]} = \frac{(a + bx_{i+1})/r}{1 - \exp[-rx_{i+1}/g(x_i)]}. \tag{10}$$

Because some cable shipments were affected by structure congestion and other factors, the raw growth rate distribution found by substituting (8) and (9) in (10) is not as smooth as one would expect the actual distribution must be. Therefore, a specific form was assumed for the distribution, and parameters were determined by fitting the data. We found that it is reasonable to assume that $[g(x)]^{1/2}$ is normally distributed. Figure 10 shows the data for all four gauges plotted on normal probability paper. Using ordinary least-square fits of $[g(x)]^{1/2}$ versus unit normal standard deviations corresponding to $F(x)$, gives the following growth rates:

$$\sqrt{g} \sim N(\mu_g, \sigma_g^2), \tag{11}$$

gauge	μ_g	σ_g
26	27.23	8.38
24	24.08	6.49
22	19.42	4.77
19	12.12	3.18

The above distributions are shown as the straight lines on Fig. 10; Fig. 11 shows the density functions for 26, 24, and 22 gauge. As one would expect, the growth rates for the finer gauge demand are generally larger than for coarser gauge demand.

One could argue that the cable shipment data more properly lead to an estimate of the forecast growth rate distribution, instead of the actual growth rate distribution as assumed here. It is not expected that this would result in a significant change in the sizing penalty estimate, but it could be studied by iteratively assuming distributions for the actual growth rate and solving through the conditional cable-size distributions until the resulting marginal distribution for the forecast cables agreed sufficiently closely with cable-shipment data.

3.2 Cable-size errors

For each gauge, the joint distribution of optimum and forecast cable sizes is found. Let

x^* = an optimum cable size,

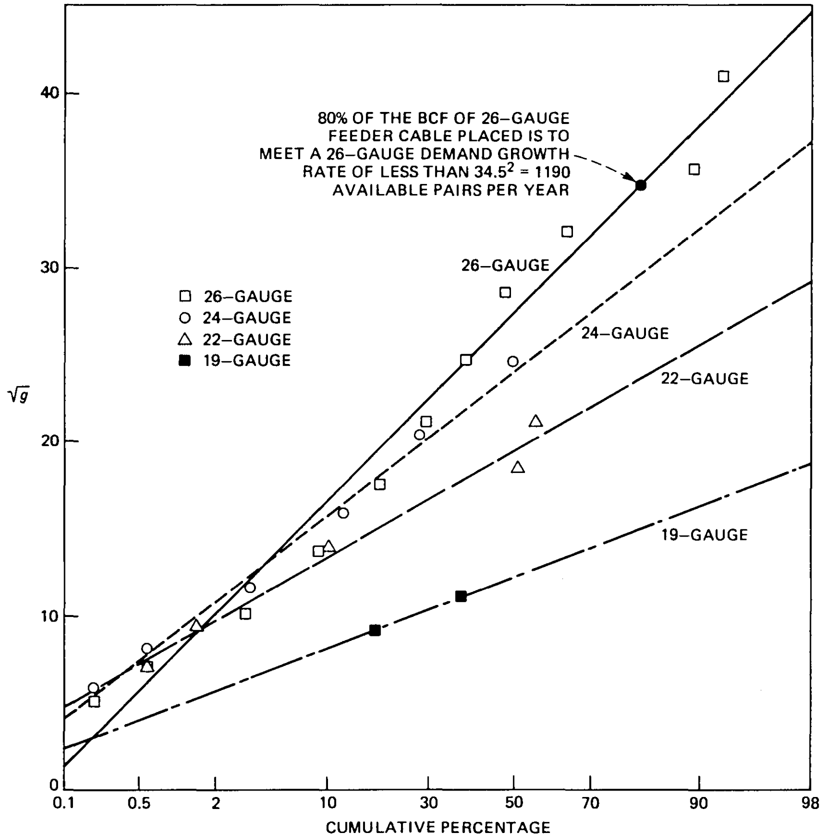


Fig. 10—Growth rate distribution. Normal probability plot of the square root of the growth rate is based on 1977 Western Electric shipments and 10 percent discounting rate.

x_f = a forecast cable size (i.e., an engineered size based on a forecast),

$p(x_f|x^*)$ = conditional probability of x_f given x^*,

$p_{x^*}(x^*)$ = marginal probability of x^*, and

$p(x^*, x_f)$ = joint probability of x^* and x_f.

*3.2.1 Marginal probability distribution of x**

The distribution of x^* is found from the growth rate distribution of Section 3.1. The discounting rate that considers inflation, 6 percent, was used to determine the growth rate intervals for which each discrete cable size is optimum. That is,

$$r = 0.06. \tag{12}$$

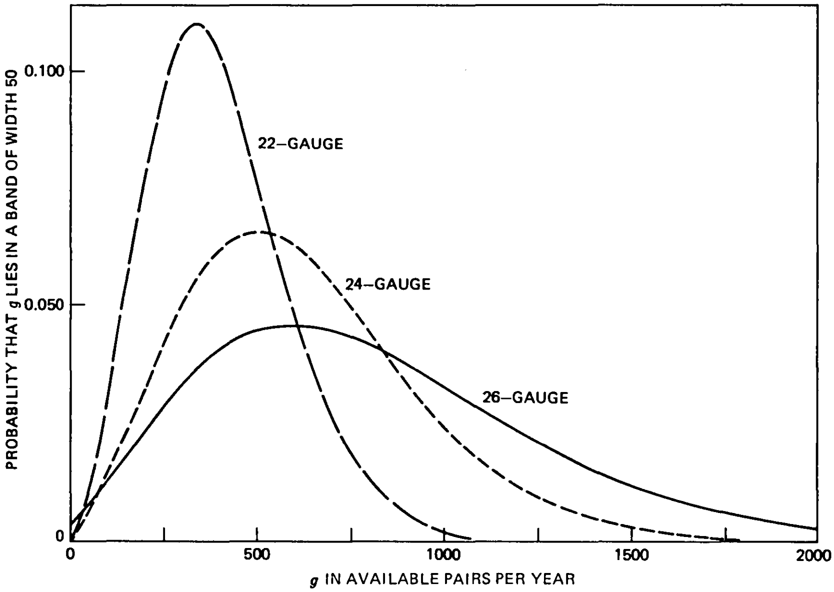


Fig. 11—Growth rate density function. Distribution of actual growth rates is based on 1977 Western Electric shipments and 10 percent discounting rate.

The growth rate intervals were determined by substituting (8) and (12) in (10). The probability of the growth rate being in each of the intervals is then found from (11). Table V gives 26-gauge values. Actually the interval for the smallest cable size should start at a growth rate of 0, but since the probability that $g < 0$ is so small, it was included with that for the smallest cable.

3.2.2 Conditional probability distribution of x_f given x^*

The distribution of x_f given x^* is found for each gauge from the forecast error distribution of Section II. Again, using a 6 percent

Table V—Twenty-six gauge values

x^*	g	$P_{x^*}(x^*)$
300	$-\infty$ – 16.65	0.003
400	16.65 – 29.97	0.002
600	29.97 – 61.05	0.005
900	61.05 – 114.16	0.014
1200	114.16 – 182.96	0.027
1500	182.96 – 267.47	0.046
1800	267.47 – 367.67	0.071
2100	367.67 – 483.58	0.098
2400	483.58 – 615.19	0.120
2700	615.19 – 762.51	0.132
3000	762.51 – 1007.01	0.186
3600	1007.01 – ∞	0.296
		1.000

discounting rate to consider inflation, the forecast growth rate intervals that correspond to each discrete value of x_f being called for are those found in Section 3.2.1. This assumes that each cable is sized to minimize the present worth cost based on the forecast growth rate.

The probability that x_f is selected when x^* is the optimum size is then found from the forecast error distribution (6). In that expression, g is the value for which x^* minimizes (7) when x^* is considered as a continuous variable. Taking the derivative of (7) with respect to x, setting it to zero, and replacing x with x^* gives

$$e^{rx^*/g} - \frac{rx^*}{g} - 1 = \frac{ar}{bg}.$$

A value for g is found for each gauge and x^* such that the above equality holds when (8) and (12) are substituted for a, b, and r. For a 26-gauge example, let $x^* = 1200$. The corresponding growth rate, g, is 148.6 available pairs per year. Figure 12 shows the distribution of g_f, given this value of g, and the intervals corresponding to each cable size being selected. Figure 13 then shows the distribution of x_f, given $x^* = 1200$.

Since one would expect the cost penalty for not placing a cable when one should have been placed to be at least as large as the penalty due to placing the smallest cable available, the probability that g_f is negative is added to that for the smallest x_f.

3.2.3 Joint probability distribution of x^* and x_f

The joint probability is the product of the marginal probability of Section 3.2.1 and the conditional probability of Section 3.2.2. That is,

$$p(x^*, x_f) = p_{x^*}(x^*)p(x_f|x^*). \tag{13}$$

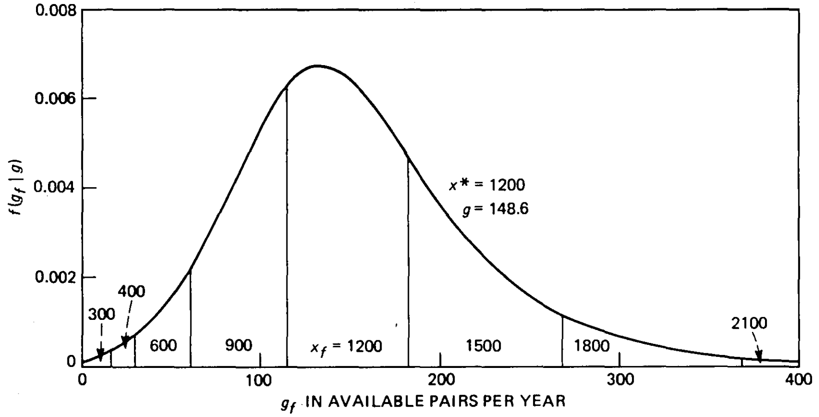


Fig. 12—Conditional density of g_f given g.

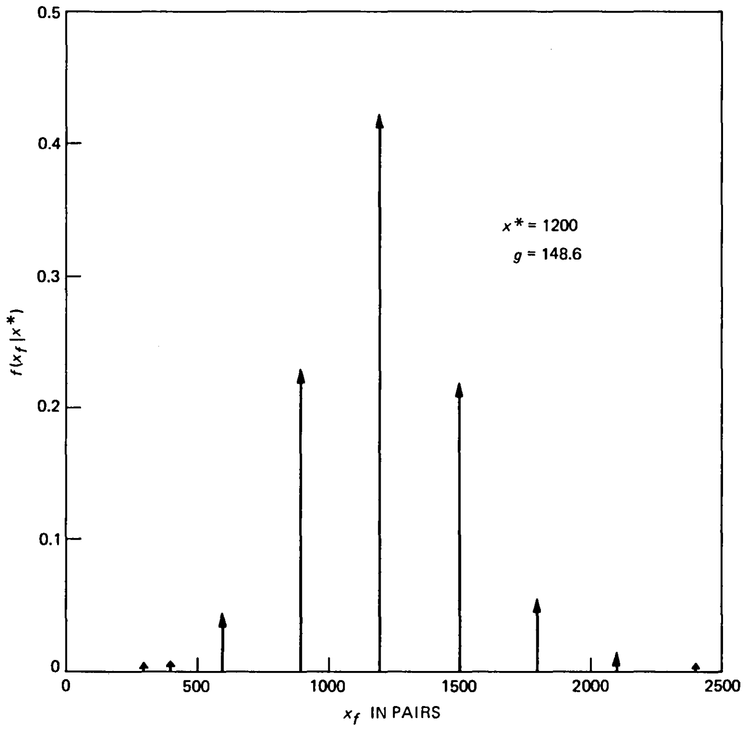


Fig. 13—Conditional density of x_f given x^*.

3.3 *Penalties for size errors*

Let $C(x^*, x_f)$ denote the percentage cost penalty when x_f is selected and x^* is optimum. Using the present worth notation of (7),

$$C(x^*, x_f) = 100\frac{PW(x_f, g) - PW(x^*, g)}{PW(x^*, g)}. \tag{14}$$

As in Section 3.2.2, $g = g(x^*)$ is the value for which x^* minimizes $PW(x^*, g)$, for each discrete value of x^*. Expression (14) is evaluated by substituting (7) for $PW(x, g)$, (8), and (12) for a, b, and r.

3.4 *Overall cost of size errors*

For each gauge, the overall penalty is the sum of the cost penalties (14), weighted by the probabilities (13). That is, the expected penalty for one gauge is

$$\sum_{x^*}\sum_{x_f} p(x^*, x_f)C(x^*, x_f).$$

The overall penalty is the sum of the above penalties weighted by the 1977 Western Electric shipments of pulp-insulated exchange cable of

Table VI—Penalty for each gauge and overall weighted penalty

Gauge	Percent Sizing Error Cost Penalty	Cable Shipments (%)	Contribution to Overall Sizing Error Cost Penalty	
26	0.534	44.6	0.238	
24	0.462	37.0	0.171	
22	0.502	18.2	0.091	
19	0.542	0.2	0.001	
		100.0	0.501%	Expected overall cost penalty

each gauge. Table VI gives the penalty for each gauge and the overall weighted penalty.

Figure 14 shows graphically the main factors that contribute to the sizing-error cost penalty of 26-gauge feeder cable. The solid curves give contours of equal cost penalty, $C(x^*, x_f)$. The dashed curves give the 1, 10, 90, and 99 percent points on the cumulative conditional distri-

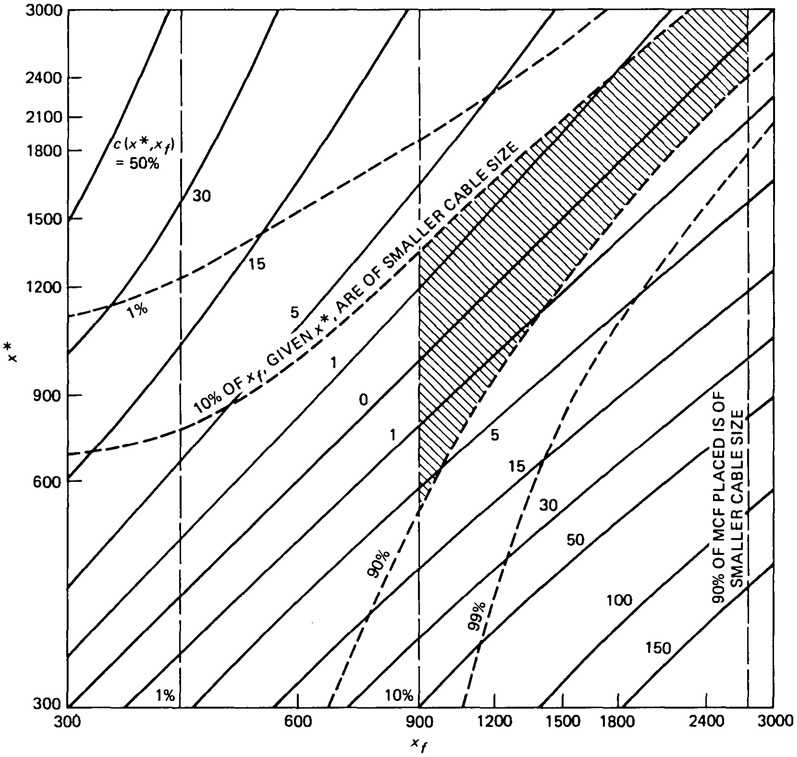


Fig. 14—Factors contributing to sizing-error cost penalty of 26-gauge feeder cable (6-percent discounting rate).

bution of $p(x_f|x^*)$. Vertical lines show the 1, 10, and 90 percent points on the cumulative MCF distribution of cable shipments of 26-gauge pulp cable. The cross hatches emphasize the area of greatest interest, where 80 percent of the shipments occur and there is an 80 percent chance of finding x_f, given x^*. The cost penalty is less than 1 percent in most of this region.

IV. SUMMARY

We have derived an estimate for the outside plant forecast error distribution. We give this distribution, expression (6), as the conditional distribution of the forecast growth rate with respect to the actual growth rate. We then used it with cable-shipment data to estimate that the feeder-cable sizing penalty due to forecast errors is about 0.5 percent of the annual feeder-cable-construction program. This represents a substantial amount of money, even though it is a small percentage, because of the large construction program.

V. ACKNOWLEDGMENTS

Thanks are due to N. B. Robbins, F. M. Stumpf, J. H. Irven, and R. Sherman for their helpful comments and suggestions. Thanks are also due to C. L. Mallows, B. Kleiner, and A. E. Freeny for their assistance with the forecast error distribution modeling.

REFERENCES

1. N. G. Long, "Loop Plant Modeling: Overview," B.S.T.J., *57,* No. 4 (April 1978), pp. 797–806.
2. D. F. Andrews, "A Robust Method for Multiple Linear Regression," Technometrics, *16,* No. 4 (November 1974), pp. 523–31.
3. F. Mosteller and J. W. Tukey, *Data Analysis and Regression,* Reading, Mass.: Addison-Wesley, 1977, pp. 351–8.
4. J. Freidenfelds, "A Simple Model for Studying Feeder Capacity Expansion," B.S.T.J., *57,* No. 4 (April 1978), pp. 807–23.

Copyright © 1981 American Telephone and Telegraph Company
THE BELL SYSTEM TECHNICAL JOURNAL
Vol. 60, No. 5, May-June 1981
Printed in U.S.A.

The Effects of Misclassification Error on the Estimation of Several Population Proportions

By J. D. HEALY

(Manuscript received October 21, 1980)

Assume that each item from a set of items is classified into one of several categories. We then use the proportion of items classified into a category to estimate the true proportion of items from the category in the population. This article models the effect of misclassification error on the estimate of the true proportion. We discuss two conditions which can be used to determine the adequacy of a classifier. We present an optimal classification algorithm which can be used when the joint distribution of the variables on which classifications are based is known separately for items from each category.

I. INTRODUCTION

Let us suppose that we observe a set of items which can be split into several distinct categories. Each item is measured and classified by some device into one of these various categories. However, the classified category for an item and the true category may not be the same, i.e., the device may make a misclassification error. The observed proportion of items in a category is then used to estimate the true proportion.

The preceding scenario often occurs in quality control[1] and medical research.[2,3] In quality control, individual manufactured items from a sample or lot are often classified by a mechanical device as defective or not and the proportion of defectives in the sample is then used to estimate the proportion of defectives from the entire process. In medical research, the items are people and the idea is to estimate the proportion of people with various diseases. In a Bell system example, the items would be phone calls and the categories would be busies, completed calls, reorders, etc. An automated device would attempt to determine the true category for each call. The output of the device would then be the estimated proportion of calls in each category. This last example motivated the analysis contained in this article.[4] Note that the problems discussed here are quite different from the traditional classification problem,[5] where the goal is to maximize the probability of correctly classifying each item. In all the above examples, errors from misclassification can have a serious effect.

In this article, we attack two separate problems. In the first problem, we assume we have little or no control over the internal design of the classifier; all that we have is an estimate of the probability of classifying items from each category into each of the other categories. The object is to develop a simple way to specify how good the classifier must be. Also, we should indicate to the designer the direction in which improvements are necessary. The second problem handles the case when we do have control over the design of the classifier. In this case, we assume that object is to design the classifier so that the effects of misclassification error are minimized. In this article, we are concerned only with a classifier's ability to estimate proportions, i.e., our loss function is entirely different than the usual loss function.

This article is organized in the following way. Section II introduces notation and explains the effects of misclassification error on the estimated proportion of items in a category. Section III discusses the case when we have little or no control over the design of the classifier. In Section IV, we discuss the case when we have control over the classifier. The resulting minimization problem involves a function that is quadratic in the probabilities of classifying items.

II. EFFECTS OF MISCLASSIFICATION

Let m be the number of categories. All vectors in this paper are m-dimensional column vectors, and matrices are m by m matrices. Also let $\mathbf{p}$, $\mathbf{p}^*$, and $\hat{\mathbf{p}}$ be the m-dimensional vectors of the true probabilities of items from the various categories, the probabilities of classifying items into various categories, and the observed proportion of items actually classified into different categories, respectively. Let $A = (a_{ij})$ be the m by m misclassification matrix which contains the conditional probabilities of classifying items into different categories. For example, a_{12} would be the probability of classifying an item from category 2 as an item from category 1. Each column of A sums to one. The diagonal elements of A are the probabilities of correct classification, and the off-diagonal elements give the probabilities of misclassification. A perfect classifier would have $A = I$, the identity matrix. By the law of total probability, $\mathbf{p}^* = A\mathbf{p}$.

In measuring the effectiveness of $\hat{\mathbf{p}}$ as an estimator of $\mathbf{p}$, we use the matrix of mean-squared errors. The matrix of mean-squared errors contains the mean-squared error of the individual terms and also the cross product terms which indicate how the different errors are related. The m by m matrix of mean-squared errors (MMSE) is

$$E[(\hat{\mathbf{p}} - \mathbf{p})(\hat{\mathbf{p}} - \mathbf{p})' \mid \mathbf{p}] = \operatorname{cov}(\hat{\mathbf{p}} \mid \mathbf{p}) + [E(\hat{\mathbf{p}} \mid \mathbf{p}) - \mathbf{p}][E(\hat{\mathbf{p}} \mid \mathbf{p}) - \mathbf{p}]', \quad (1)$$

where "cov" means the covariance matrix. The diagonal of the covariance matrix measures precision of an estimator while the diagonal of the second term in (1) is the bias squared.

We assume that n items are to be classified. The expected number of items from each category is $n\mathbf{p}$. If items are classified independently, the distribution of $n\hat{\mathbf{p}}$ will be a multinomial distribution with parameters $A\mathbf{p}$ and sample size n. From Ref. 6, we obtain the $\operatorname{cov}(\hat{\mathbf{p}} \mid \mathbf{p})$,

$$\operatorname{cov}(\hat{\mathbf{p}} \mid \mathbf{p}) = [D^* - A\mathbf{p}\mathbf{p}'A']/n, \quad (2)$$

where D^* is a diagonal matrix with diagonal equal to $\mathbf{p}^*$. The $\operatorname{cov}(\hat{\mathbf{p}} \mid \mathbf{p})$ can be separated into two parts, one part which is the covariance matrix if the classifier were perfect, and the second part which is an adjustment in the covariance matrix because the classifier is not perfect:

$$\operatorname{cov}(\hat{\mathbf{p}} \mid \mathbf{p}) = [D - \mathbf{p}\mathbf{p}']/n + [D^* - D - A\mathbf{p}\mathbf{p}'A' + \mathbf{p}\mathbf{p}']/n,$$

where D is a diagonal matrix with diagonal equal to $\mathbf{p}$.

Since $E(\hat{\mathbf{p}} \mid \mathbf{p}) = A\mathbf{p}$, the bias term in (1) becomes

$$[E(\hat{\mathbf{p}} \mid \mathbf{p}) - \mathbf{p}][E(\hat{\mathbf{p}} \mid \mathbf{p}) - \mathbf{p}]' = (A - I)\mathbf{p}\mathbf{p}'(A - I)'. \quad (3)$$

Note that this term is not divided by n. Putting the above statements together, (1) becomes

$$E[(\hat{\mathbf{p}} - \mathbf{p})(\hat{\mathbf{p}} - \mathbf{p})' \mid \mathbf{p}] = (D - \mathbf{p}\mathbf{p}')/n + (D^* - D - A\mathbf{p}\mathbf{p}'A' + \mathbf{p}\mathbf{p}')/n + [(A - I)\mathbf{p}\mathbf{p}'(A - I)']. \quad (4)$$

This equation includes the sampling-error effect (first term on right) and the effect of a misclassification error (other terms on right).

III. SPECIFYING ACCURACY WITH LITTLE CONTROL OVER THE CLASSIFIER

Assume that our information on a classifier is confined to the matrix A, i.e., someone else is responsible for the design of the classifier. We may influence the form of A but we have no control over the actual functioning of the classifier. Suppose we have a good idea of how large the mean-squared errors for each category estimate [the diagonal elements of (4)] can be for the application of the classifier. We need several guidelines on the form of A which will insure that the classifier and its resulting mean-squared errors are adequate. Clearly, we do not want to put constraints on every element of A. Also, we cannot tell the designer of the classifier that the mean-squared errors of his classifier are inadequate without providing some guidance on how to improve the classifier.

Intuitively, we would like to pick A so that the bias term in (4) disappears. We cannot pick A so that the bias term in (4) disappears for every $\mathbf{p}$. For each A, however, there is some value of $\mathbf{p}$ for which the bias term disappears; in fact, A tends to produce $\hat{\mathbf{p}}$, which are collapsed toward the value of $\mathbf{p}$ for which the bias term disappears. A reasonable strategy is to pick A so that the bias term disappears for our "best guess" for $\mathbf{p}$ which we denote by $\mathbf{p}_0$. This means A should be picked so that $(A - I)\mathbf{p}_0 \approx 0$, i.e., $\mathbf{p}_0$ is approximately an eigenvector of A with eigenvalue 1. There are many A which satisfy $(A - I)\mathbf{p}_0 \approx 0$, but which have large mean-squared errors for values of $\mathbf{p}$ near $\mathbf{p}_0$. To ensure that mean-squared errors are small for values of $\mathbf{p}$ near $\mathbf{p}_0$, we must additionally require that the a_{ii} (diagonal elements of A) be reasonably large; note that if the eigenvector condition is nearly satisfied, the requirement on the a_{ii} may, in many cases, be quite loose.

These two conditions: (1) $(A - I)\mathbf{p}_0 \approx 0$, and (2) a_{ii} large, are generally easy to check. Assume we have a set of s items which we know contain $s\mathbf{p}_0$ items from the respective categories. These items are classified and are the results used to estimate A. The first condition states that the number of items classified into a category is roughly equal to $s\mathbf{p}_0$. If this condition is not originally satisfied, the designer can usually satisfy it by adjusting several thresholds which determine where.the classifier places items. The second condition says that the classifier cannot misclassify a high proportion of items from any one category. The designer is then told which categories do not satisfy this condition.

We now show that these two conditions can be justified analytically when we assume that $\mathbf{p}$ has an underlying Dirichlet distribution. That is, we now allow $\mathbf{p}$ to vary, for example, with environment. The Dirichlet distribution is the natural multivariate generalization of the beta distribution and it is the conjugate prior for the multinomial distribution. We pick the parameters of the Dirichlet distribution so that

$$E(\mathbf{p}) = \mathbf{p}_0, \tag{5}$$

$$\operatorname{cov}(\mathbf{p}) = -\mathbf{p}_0\mathbf{p}_0'/(v + 1) + D_0/(v + 1), \tag{6}$$

where D_0 is a diagonal matrix whose ith diagonal element is the ith element of $\mathbf{p}_0$, and v is a parameter that indicates how spread out the Dirichlet distribution is. These parameters, $\mathbf{p}_0$ and v, can be chosen so that the resulting Dirichlet distribution models the expected environmental variability of $\mathbf{p}$. If we take expected values of the terms in (4) using (5) and (6), we obtain

$$\begin{aligned} E(\hat{\mathbf{p}} - \mathbf{p})(\hat{\mathbf{p}} - \mathbf{p})' &= D_0^*/n - AD_0A'/n(v+1) \\ &\quad - A\mathbf{p}_0\mathbf{p}_0'A'v/n(v+1) + (A-I)\mathbf{p}_0\mathbf{p}_0'(A-I)'v/(v+1) \\ &\quad + (A-I)D_0(A-I)'/(v+1), \qquad (7) \end{aligned}$$

where D_0^* is a diagonal matrix whose diagonal equals $A\mathbf{p}_0$. This equation incorporates the effects of varying $\mathbf{p}$. If n and v are at all large, the fourth term in (7) will be important. If $(A - I)\mathbf{p}_0 \approx 0$ holds, this term will drop out. The fifth term is minimized if the a_{ii} are large. Since the second and third terms are generally unimportant [they are divided by $n(v + 1)$ which should be large] and since the first term is present even with a perfect classifier, satisfying the two conditions will minimize the effects of misclassification. In short, we have presented a way to require the A matrix to be "near" the identity matrix without putting constraints on each and every element of A.

IV. DESIGNING A CLASSIFIER THAT MINIMIZES MEAN-SQUARED ERROR

Assume that our job is to develop a classifier that minimizes the effects of mean-squared error. More specifically, assume we measure a vector of variables ($\mathbf{x}$) on each item. Regions R_i are defined such that if $\mathbf{x} \in R_i$, we classify the item into category i. We want to define the R_i that minimizes a weighted sum of the mean-squared errors for the various categories, i.e., that minimizes

$$\operatorname{tr}[Q\,E(\hat{\mathbf{p}} - \mathbf{p})(\hat{\mathbf{p}} - \mathbf{p})'], \qquad (8)$$

where Q is a known positive-diagonal matrix, tr is the trace operator, and $E(\hat{\mathbf{p}} - \mathbf{p})(\hat{\mathbf{p}} - \mathbf{p})'$ is defined by (7). The matrix Q is just a weighting factor that can be used to emphasize the important categories. Minimizing (8) is equivalent to minimizing

$$\operatorname{tr}(QABA' - AC), \qquad (9)$$

where

$$B = \frac{n-1}{n(v+1)}(D_0 + \mathbf{p}_0\mathbf{p}_0'v), \qquad (10)$$

$$C = \left(\frac{2}{(v+1)}(D_0 + \mathbf{p}_0\mathbf{p}_0'v) - \frac{1}{n}\mathbf{p}_0(1, 1, \cdots, 1)\right)Q. \qquad (11)$$

We handle a more general case by allowing B to be any known symmetric, nonnegative definite matrix, and C any known matrix.

Equation (9) is interesting because it is quadratic in A. The usual Bayes multiple decision rule[7] is a special case of (9), since it is the rule that minimizes (9) when $B = \mathbf{0}$, and C is a diagonal matrix with diagonal equal to the vector of prior probabilities. Classical discriminant analysis[5] is a special case of the Bayes multiple decision rule for which the distribution of $\mathbf{x}$ when the item comes from category i is multivariate normal with mean μ_i and covariance matrix $\mathbf{\Sigma}$.

Let $f_i(\mathbf{x})$ be the density of $\mathbf{x}$ if $\mathbf{x}$ is measured on an item from category i. The optimal classification algorithm is given in the following theorem:

Theorem 1: Equation (9) will be minimized if x is classified into the ith category when the ith element of

$$(f_1(\mathbf{x}), f_2(\mathbf{x}), \cdots, f_k(\mathbf{x}))(2BAQ' - C) \tag{12}$$

is the smallest.

(*Proof*: See appendix.)

In general, applying Theorem 1 should be quite difficult since A has to be solved for in (12). We now discuss the two-category case and then specialize to the case when $\mathbf{x}$ has a multivariate normal distribution. We give a simple iterative procedure to calculate the required quantities for this last case.

For the two-category case, Theorem 1 reduces to the following corollary.

Corollary 1: If there are only two categories of measurements, then eq. (9) will be minimized if x is placed into category 1 if

$$f_1(\mathbf{x})/f_2(\mathbf{x}) > K, \tag{13}$$

where

$$\begin{aligned} K = [&2b_{22}(q_{11} + q_{22})a_{21} - 2b_{21}(q_{11} + q_{22})a_{12} \\ &\qquad + 2(b_{21}q_{11} - b_{22}q_{22}) + (c_{22} - c_{21})] \\ /[&2b_{11}(q_{11} + q_{22})a_{12} - 2b_{12}(q_{11} + q_{22})a_{21} \\ &\qquad + 2(b_{12}q_{22} - b_{11}q_{11}) + (c_{11} - c_{12})], \end{aligned} \tag{14}$$

and b_{ij}, q_{ii}, and c_{ij} are elements of B, Q, and C, respectively.

Let us now assume that, if $\mathbf{x}$ is an observation from category i ($i = 1$ or 2), it has a multivariate normal distribution with known mean vector μ_i and known covariance matrix $\mathbf{\Sigma}$. Then (13) becomes

$$\begin{aligned} \log(f_1(\mathbf{x})/f_2(\mathbf{x})) = \mathbf{x}'\mathbf{\Sigma}^{-1}(\mu_1 - \mu_2) - 1/2(\mu_1 \\ + \mu_2)'\mathbf{\Sigma}^{-1}(\mu_1 - \mu_2) > \log K. \end{aligned}$$

As in classical discriminant analysis,[5] the distribution of $\log(f_1(\mathbf{x})/f_2(\mathbf{x}))$ is normal with mean $\alpha/2$ or $-\alpha/2$ when $\mathbf{x}$ comes from categories 1 or 2, respectively, where

$$\alpha = (\mu_1 - \mu_2)'\Sigma^{-1}(\mu_1 - \mu_2). \tag{15}$$

In either case, the variance of $\log(f_1(\mathbf{x})/f_2(\mathbf{x}))$ is α. Therefore, when $\mathbf{x}$ comes from category 1,

$$[\log(f_1(\mathbf{x})/f_2(\mathbf{x})) - \alpha/2]/\sqrt{\alpha}$$

has a standard normal distribution. Similarly,

$$[\log(f_1(\mathbf{x})/f_2(\mathbf{x})) + \alpha/2]/\sqrt{\alpha}$$

has a standard normal distribution when $\mathbf{x}$ comes from category 2. The misclassification probabilities can be defined in terms of a standard normal random variable,

$$a_{21} = P\left(Z < \frac{\log K - \alpha/2}{\sqrt{\alpha}}\right), \tag{16}$$

$$a_{12} = P\left(Z > \frac{\log K + \alpha/2}{\sqrt{\alpha}}\right), \tag{17}$$

where Z has a standard normal distribution. Using (14), (16), and (17) the values of a_{12}, a_{21}, and K may be calculated iteratively:

(1) Let $K = 1$ and calculate a_{12} and a_{21} using (16) and (17);
(2) Obtain a new value of K by substituting a_{12} and a_{21} into K; and
(3) Repeat the entire process.

In summary, assume we are trying to minimize $\operatorname{tr}[QE(\hat{p} - p)(\hat{p} - p)']$, where Q is a known weighting matrix. Also assume we have only two categories and that if $\mathbf{x}$ comes from category i, it has a multivariate normal distribution with mean vector μ_i and covariance matrix Σ. The parameter α may be calculated using (15). The B and C matrices should be calculated using (10) and (11). Equation (13) gives the decision rule for classification into one of the two categories, where the values of K, a_{12}, and a_{21} are obtained in an iterative manner using (14), (16), and (17).

To apply any of the preceding theory, some prior knowledge of the distribution of $\mathbf{p}$ is required to estimate $\mathbf{p}_0$ and v. If the parameters μ_1, μ_2, and Σ are unknown, they may be estimated from a sample of data with the usual sample means and pooled covariance matrix. An estimator of the parameter α could then be calculated using (15) with the estimates of μ_1, μ_2, and Σ substituted into (15). The algorithm discussed above could then be used to generate A and K where the estimator of α is used in (16) and (17) instead of α. The properties of the procedure when estimators of the parameters are used require further study.

V. SUMMARY

This article presents a model that incorporates the effects of misclassification error. Several guidelines are presented which can be used to determine if a given classifier is adequate. For the case when the classifier is yet to be designed, we have given an optimal classification algorithm. We discuss the two-category case in detail.

APPENDIX

Proof of Theorem 1

Let R_i^0 be the regions that result if the theorem is applied. Let A_0 be the resulting misclassification matrix. Let R_i^1 and A_1 be the corresponding elements for some other decision rule. Now consider (9) evaluated at A_1 minus (9) evaluated at A_0:

$$\begin{aligned}\operatorname{tr}(QA_1BA_1' - QA_0BA_0' - A_1C + A_0C) \\ = \operatorname{tr}(A_1 - A_0)B(A_1 - A_0)'Q + 2\operatorname{tr}(A_1 - A_0)BA_0'Q \\ - \operatorname{tr}(A_1 - A_0)C. \qquad (18)\end{aligned}$$

The first term on the right of (18) is nonnegative since B and Q are nonnegative definite. We still have to show that the rest of (18) is nonnegative. Consider

$$2\operatorname{tr}(A_1 - A_0)BA_0'Q - \operatorname{tr}(A_1 - A_0)C = \operatorname{tr}A_1E - \operatorname{tr}A_0E, \qquad (19)$$

where $E = 2BA_0'Q - C$. Let e_{ij} be the i, jth element of E. Equation (19) now becomes

$$\begin{aligned}\operatorname{tr}(A_1E) - \operatorname{tr}(A_0E) &= \sum_{i,j} \int \phi_1(i|\mathbf{x}) f_j(\mathbf{x})\, d\mathbf{x} \cdot e_{ji} \\ &\quad - \sum_{k,m} \int \phi_0(k|\mathbf{x}) f_m(\mathbf{x})\, d\mathbf{x} \cdot e_{mk} \\ &= \sum_{i,k} \int \phi_1(i|\mathbf{x}) \phi_0(k|\mathbf{x}) \\ &\qquad \cdot \left[\sum_j f_j(\mathbf{x}) e_{ji} - \sum_m f_m(\mathbf{x}) e_{mk} \right] d\mathbf{x}, \qquad (20)\end{aligned}$$

where

$$\phi_0(i|\mathbf{x}) = \begin{cases} 1, & \mathbf{x} \in \mathbf{R}_i^0, \\ 0, & \mathbf{x} \notin R_i^0, \end{cases}$$

$$\phi_1(i|\mathbf{x}) = \begin{cases} 1, & \mathbf{x} \in R_i^1, \\ 0, & \mathbf{x} \notin R_i^1. \end{cases}$$

Since $\phi_0(k|\mathbf{x})$ will be zero whenever $[\sum_j f_j(\mathbf{x}e_{ji} - \sum_m f_m e_{mk}]$ is negative, (20) is always nonnegative. Q.E.D.

REFERENCES

1. W. M. Wooding, "A Source of Bias in Attributes Testing, and a Remedy," J. Quality Technol., *11,* No. 4 (October 1979), pp. 169-76.
2. P. Armitage, *Statistical Methods in Medical Research,* New York: Wiley, 1971.
3. T. Colton, *Statistics in Medicine,* Boston: Little, Brown, and Co., 1974, pp. 87–92.
4. J. D. Healy, unpublished work.
5. T. W. Anderson, *An Introduction to Multivariate Statistical Analysis.* New York: Wiley, 1958, Chap. 6.
6. N. L. Johnson and S. Kotz, *Discrete Distributions,* New York: Wiley, 1969, pp. 281–91.
7. T. S. Ferguson, *Mathematical Statistics A Decision Theoretic Approach,* New York: Academic, 1973, Chap. 6.

THE BELL SYSTEM TECHNICAL JOURNAL
Vol. 60, No. 5, May–June 1981
Printed in U.S.A.

Adaptive Post-Filtering of ADPCM Speech

By N. S. JAYANT

(Manuscript received December 3, 1980)

We show that the quality of adaptive differential PCM *(*ADPCM*) speech can be significantly improved by passing it through a reconstruction low-pass filter that is matched to an appropriately defined short-time speech cutoff frequency. Practically, the adaptive procedure involves switching the decoder output into one of a bank of N low-pass filters whose cutoff frequencies span the expected range of input speech bandwidth. For the case of equally spaced filter cutoffs, and with uniform probability density function models for the quantization noise spectrum and the cutoff frequency, more than one-half of the maximum adaptive filtering gain is realizable by a bank of four filters. Computer simulations of 16- and 24-kilobit/s* ADPCM *coders indicate that perceived quality gains are in fact greater than what is indicated by an analytically predicted objective gain of 2.6 dB.*

I. SHORT-TIME CUTOFF FREQUENCY

ADPCM (adaptive-quantization/differential PCM) speech coding usually assumes a time-invariant model for speech bandwidth (such as 3200 Hz for telephone quality applications), and a corresponding time-invariant low-pass filter with cutoff $f_0 = 3200$ Hz for the decoded speech. However, short-time speech spectra of 3200-Hz-filtered speech exhibit cutoff frequencies f_c, that are often significantly smaller than the long-time nominal cutoff frequency f_0.[1] Figure 1 sketches a short-time spectrum at time t, and defines $f_c(t) = f_L^T(t)$ as the low-pass cutoff frequency that includes all but T percent of short-time spectral energy. Figure 2 shows long-time-averaged spectra for four sentence-length 3200 Hz band-limited utterances [L, B, C, and D], denoting ["A *l*athe is a big tool" (female utterance), "An icy wind raked the *b*each" (female utterance), "The *c*hairman cast three votes" (female utterance), and "This is a computer test of a *d*igital speech coder" (male utterance)], respectively. It is clear that all the four spectra are low

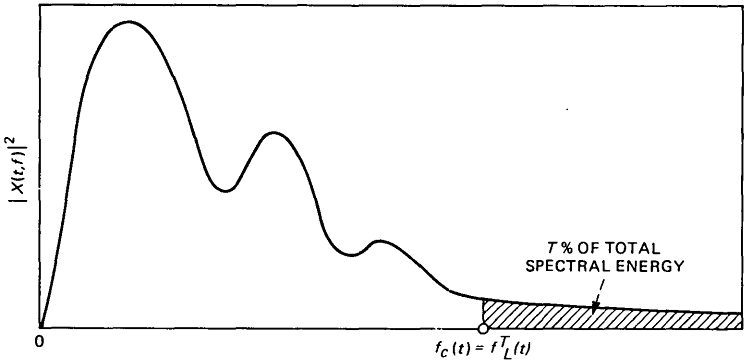


Fig. 1—Definition of short-time cutoff frequency $f^T_L(t)$. The shaded area includes T percent of short-time speech power.

pass in a long-time-average sense, a fact that is well exploited in fixed-prediction differential coding.[2] Figure 3 shows corresponding histograms of short-time cutoff frequency $f^T_L(t)$ for a threshold $T = 1$ percent. It is seen that on a short-time basis, speech segments can be either low pass (say, $f_L(t) < f_0/2$) or all pass (say, $f_L(t) > f_0/2$), although both of these segment types come from inputs that are low pass from a long-time-averaged energy viewpoint. It is also clear that the four histograms of Figure 3 are very different; however, as a single descriptor of these histograms, we propose a uniform probability density model for $f_L(t)$,

$$p(f^T_L(t)) = \frac{1}{f_0}, \qquad 0 < f_L(t) < f_0. \tag{1}$$

The adequacy of the above model is clearly a function of the threshold T. Clearly for the extreme cases of $T = 100$ percent and $T = 0$ percent, the pdf of $f^T_L(t)$ would degenerate into delta functions at $f = 0$ and f_0, respectively, with corresponding low-pass counts of 100 and 0 percent. The uniform density model (1), on the other hand, implies equal, 50 percent occurrences of low-pass and all-pass segments, and Fig. 4 shows that this is reasonable as a sentence-ensemble average for thresholds $T = 1$ and 2 percent. A threshold of $T = 1$ percent has been used in all of our ADPCM simulations.

The value of $T = 1$ percent produces a filtering distortion of $10 \log(1/100) = -20$ dB. More relevantly, it constitutes a "perceptually acceptable" low-pass threshold, with a filtering distortion that is obvious only in critical listening. On the other hand, at $T = 2$ percent, the filtering distortion begins to get obvious, and undesirable even for low-quality application such as 16 kilobit/s. The role of threshold T is discussed at greater length in a recent work which relates to the refinement of ADM (adaptive delta modulation) with adaptive post-filtering.[1] This work also shows that consideration of the low-pass threshold $f_L^T(t)$ is more useful, for purposes of reducing coder noise, than the consideration of a high-pass threshold $f_H^T(t)$; in other words, providing a band-pass reconstruction filter matched to short-time high-pass and low-pass cutoffs $[f_H(t),\ f_L(t)]$ was not significantly better than providing a low-pass reconstruction filter matched to the range $[0,\ f_L(t)]$.

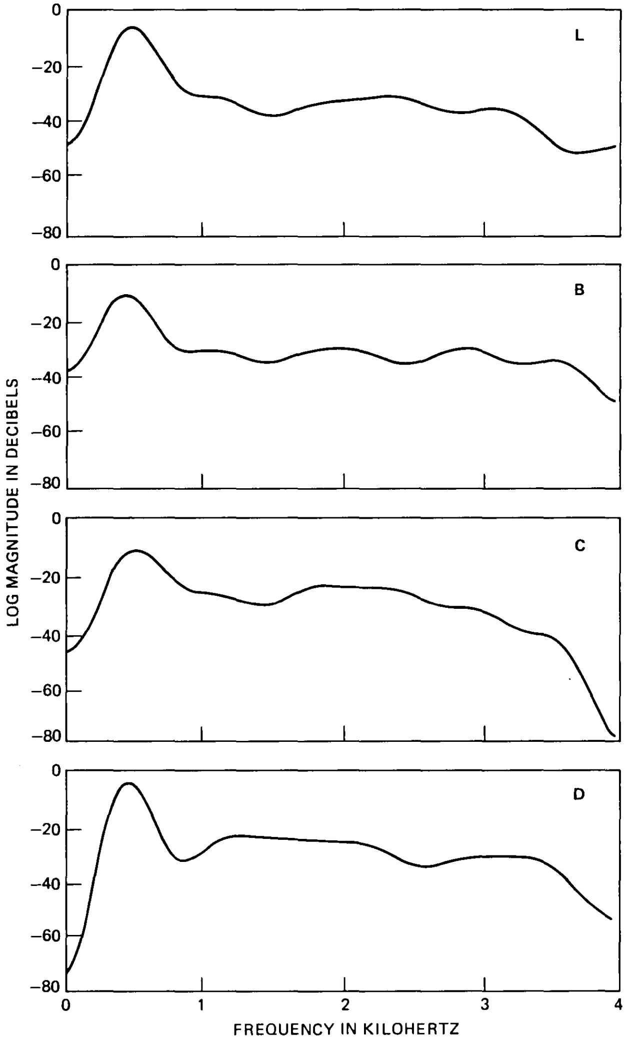


Fig. 2—Long-time averaged spectra for four sentence-length inputs. A 3.2-kHz band limitation is obvious in all four examples.

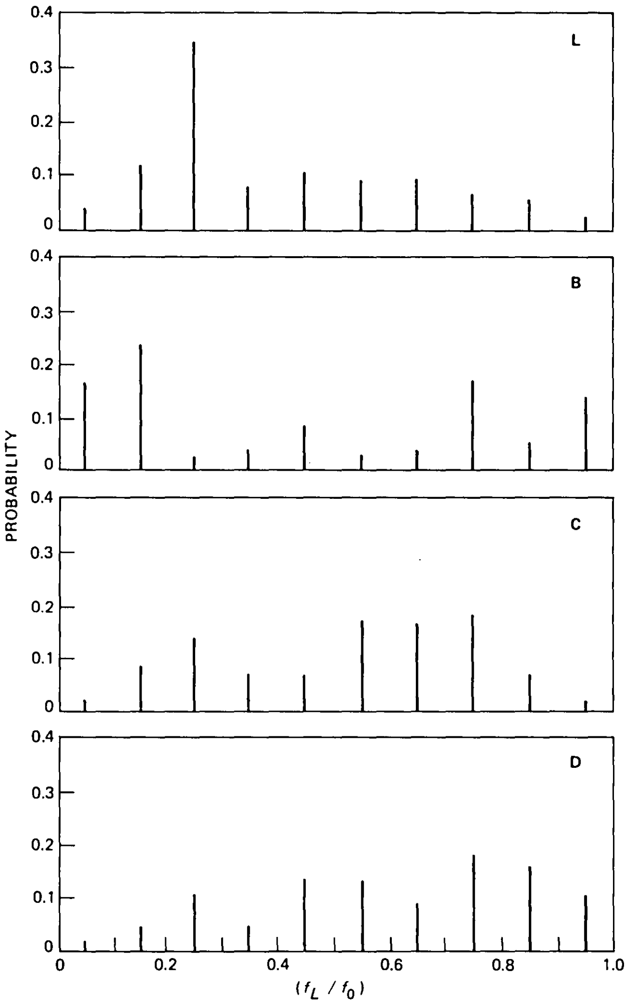


Fig. 3—Histograms of short-time cut-off frequency $f_L^T(t)$ ($T = 1$ percent) for the four inputs of Fig. 2. The short-time cutoff is in general significantly less than 3.2 kHz, the cutoff in the long-time-averaged spectra of Fig. 2.

II. ADAPTIVE LOW-PASS FILTERS

Figure 5a shows the effect of a low-pass reconstruction filter in ideal adaptive post-filtering; out-of-band ADPCM quantization noise components in the cross-hatched range $[f_L(t), f_0]$ in the noise spectrum are rejected by an ideal low-pass filter matched to $f_L(t)$.

Figures 5b and 5c show suboptimal but practical versions of 5a, where the value of $f_L(t)$ causes switching the decoder output into one of a bank of N low-pass filters, leading to a degree of noise-rejection

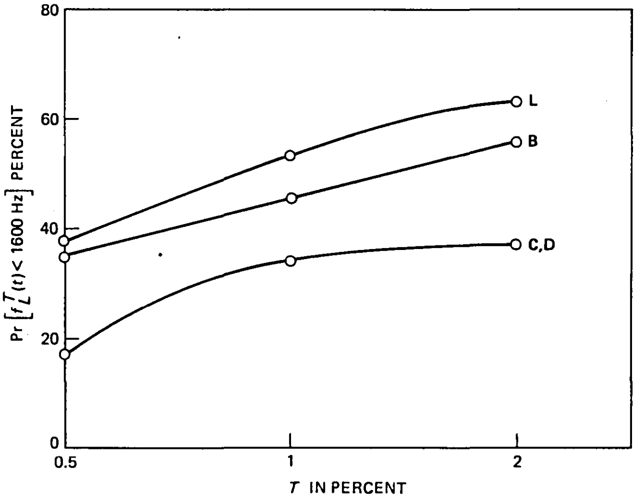


Fig. 4—Percentages of segments with $f_L^T(t) < f_0/2$ in the four inputs of Fig. 2. For $T = 1$ and 2 percent, the mean percentage of such segments is in the order of 50 percent.

that is always less than that in Fig. 5a. For example, in the $N = 2$ example of Fig. 5b, the filter bank consists of two filters with fixed cutoffs $f_{c2}(= f_0)$ and f_{c1}; in the upper illustration in Fig. 5b, the value of $f_L(t)$ is not small enough to switch in the lower filter f_{c1}; consequently there is no out-of-band noise rejection similar to that in the upper example of Fig. 5a. With the uniform pdf model (1), the two-filter bank system realizes out-of-band noise rejection only 50 percent of the time (when $f_L(t) < f_{c1} = f_0/2$). The four-filter system of Fig. 5c is clearly more effective; it realizes nonzero noise rejections in three out of the four cases shown, and indeed for 75 percent of all speech segments, if the uniform pdf (1) is valid.

A block diagram of an N-filter-bank adaptive system appears in Fig. 6. Note that for simplicity, the cutoff frequencies are equally spaced,

$$f_{cn} = f_0 \frac{n}{N}, \tag{2}$$

and that filter n is switched on when the input frequency cutoff is in the appropriate (f_0/N)-wide range,

$$\text{Switch to filter } n \quad \text{if} \quad \frac{n-1}{N} < \frac{f_L(t)}{f_0} \leq \frac{n}{N}. \tag{3}$$

III. SEGMENTAL s/n GAINS $G(N)$

The spectrum of quantization noise depends on many factors, including the nature of adaptive quantization, the input spectrum, and

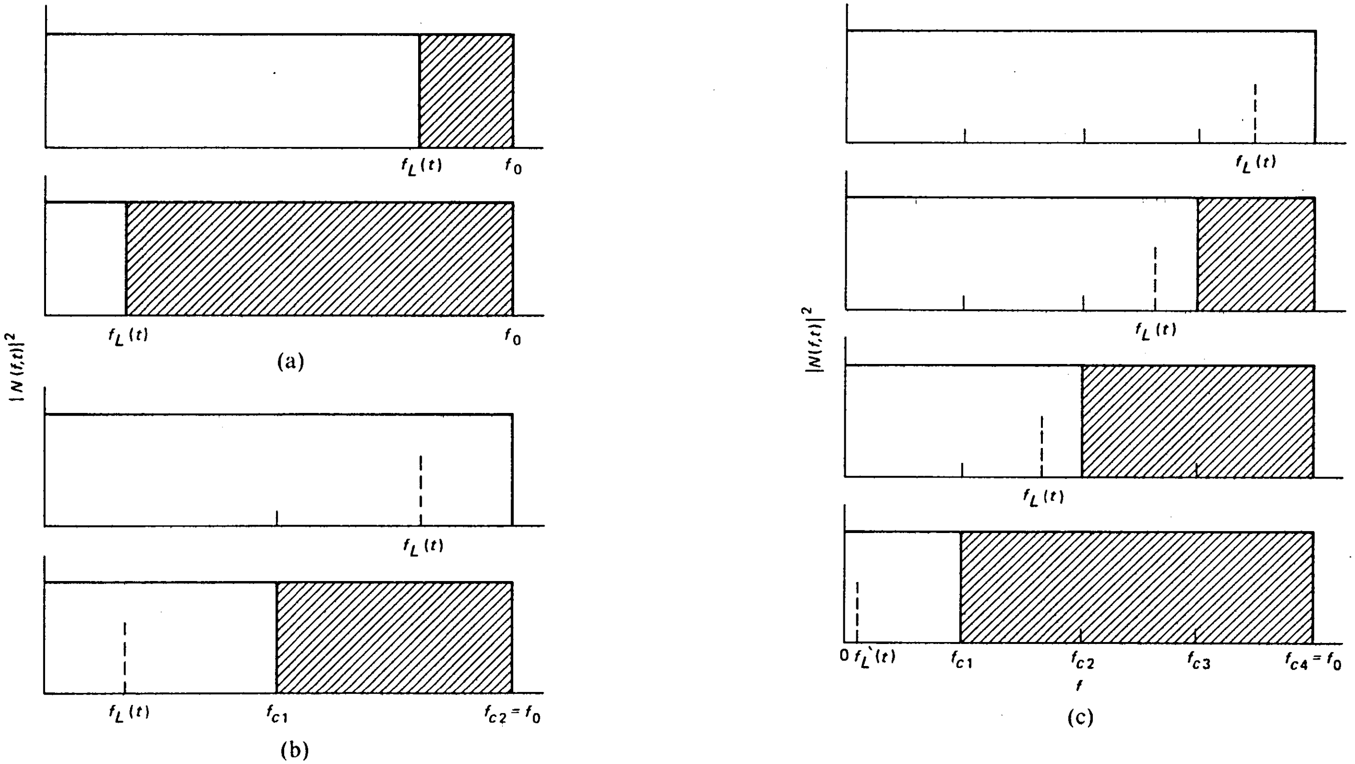


Fig. 5—Out-of-band noise rejection in (a) ideal adaptive filtering and in practical filter-bank schemes with (b) $N = 2$, and (c) $N = 4$ low-pass filters. Cross-hatched regions refer to rejected portions of noise spectrum. In the practical schemes (b) and (c), if f_{CL} refers to the allowed filter cutoff f_{cr} ($r = 1, 2, 3, 4$) that is nearest to and higher than $f_L(t)$, the out-of-band noise in the frequency range ($f_L(t)$, f_{CL}) is left unrejected.

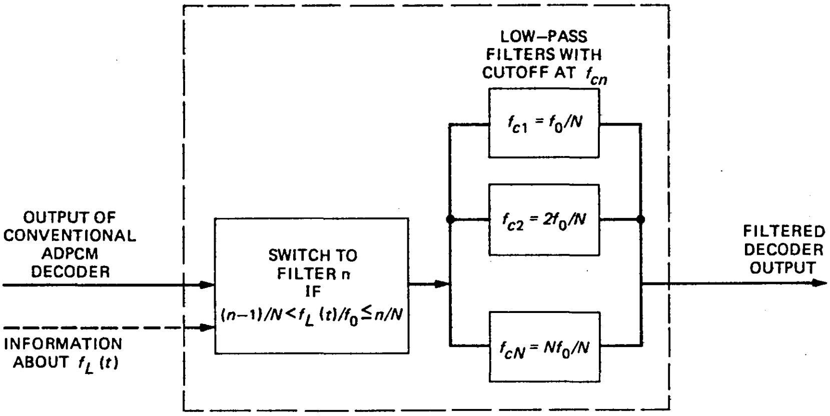

LOW–PASS FILTERS WITH CUTOFF AT f_{cn}
$f_{c1} = f_0/N$
$f_{c2} = 2f_0/N$
$f_{cN} = Nf_0/N$
OUTPUT OF CONVENTIONAL ADPCM DECODER
INFORMATION ABOUT $f_L(t)$
SWITCH TO FILTER n IF $(n-1)/N < f_L(t)/f_0 \leq n/N$
FILTERED DECODER OUTPUT

Fig. 6—Block diagram of adaptive low-pass filtering system with an N-filter bank. Allowed cutoff frequencies are equally spaced, being integral multiples of f_0/N. The entire dashed box constitutes an optional refinement of a conventional ADPCM decoder.

the effect of predictor; but experience indicates that the white-noise spectrum of Fig. 5 is indeed the single most reasonable model.[1,3] Combining this assumption with the uniform pdf for the cutoff frequency $f_L(t)$, we shall now develop expressions for expected gains in segmental[2] s/n due to adaptive post-filtering.

When the input cutoff $f_L(t)$ is such as to switch filter n ($n = 1, 2, \cdots, N$), the noise rejection factor is N/n (see Fig. 5), with a maximum value of N for the extremely low values of $f_L(t)$, and a minimum value of 1 for extremely high values of $f_L(t)$. The expected gain (in dB) is therefore given by

$$G(N) = \sum_{n=1}^{N} \left(10 \log \frac{N}{n}\right) \cdot \left(\Pr[n] = \frac{1}{N}\right) \text{dB}, \tag{4}$$

where the probability of switching in filter n is $\Pr[n] = 1/N$, a result of the uniform pdf (1). The filtering gain $G(N)$ can be simply rewritten in the form

$$G(N) = 10 \log N - \frac{10}{N} \sum_{1}^{N} \log n. \tag{5}$$

Figure 7 plots $G(N)$ as a function of N.

The asymptotic value

$$G(\infty) = \int_0^{f_0} \left(10 \log \frac{f_0}{f_L(t)}\right) \cdot p(f_L(t)) df_L(t) \tag{6}$$

can be evaluated simply by using the identity

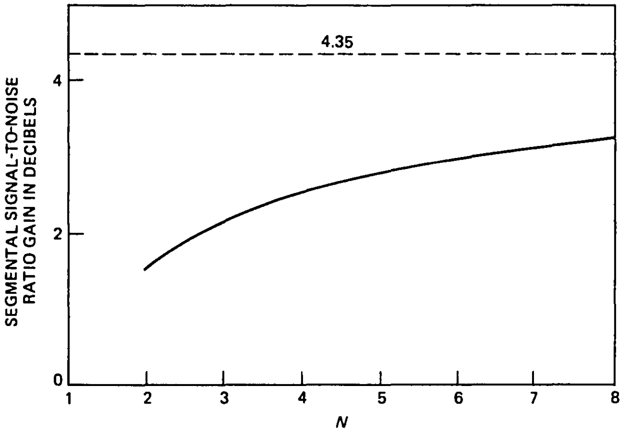


Fig. 7—Segmental s/n gain $G(N)$ versus N. This characteristic assumes uniform pdf models for coding noise and $f_L(t)$. The asymptotic value is $G(\infty) = 4.35$ dB. More than half this gain is realized with $N = 4$.

$$\int \ln z = z \ln z - z,$$

and this results in

$$G(\infty) = 10/e = 4.35 \text{ dB}. \tag{7}$$

This asymptotic value is indeed close to the ideal adaptive filtering gains reported in earlier-cited ADM experiments at several bit rates.[1] It is also seen from Fig. 6 that the four-filter bank method of Fig. 5c theoretically realizes more than one-half of the maximum possible dB gain $G(\infty)$ in segmental s/n: $G(4) \simeq 2.6$ dB.

The above analytical formulation can also be used to assess the efficiency of the equispaced filter design in (2) and Fig. 6. For illustration, $G(2) = 1.5$ dB with $N = 2$. It can be shown analytically that an optimal design, one that maximizes adaptive filtering gain for $N = 2$, is one for which $f_{c1} = (f_0/e)$, rather than $f_0/2$. Figure 8 plots the theoretical expected gain in segmental s/n as a function of f_{c1}. The maximum gain is 1.6 dB, only 0.1 dB better than that in the simple design of (2), which suggests $f_{c1} = f_0/2$ for $N = 2$.

IV. RESULTS OF COMPUTER SIMULATIONS

ADPCM coders with two- and four-bank filtering systems were simulated with the speech inputs mentioned in Section I. The coder bit rates were 24 and 16 kilobit/s, corresponding to 3- and 2-bit/sample quantizers, and an 8-kHz sampling of the inputs. The quantizers were adaptive. Both backward adaptive (AQB) and forward-adaptive (AQF) quantizers were simulated;[4] filtering gains were evident in both cases,

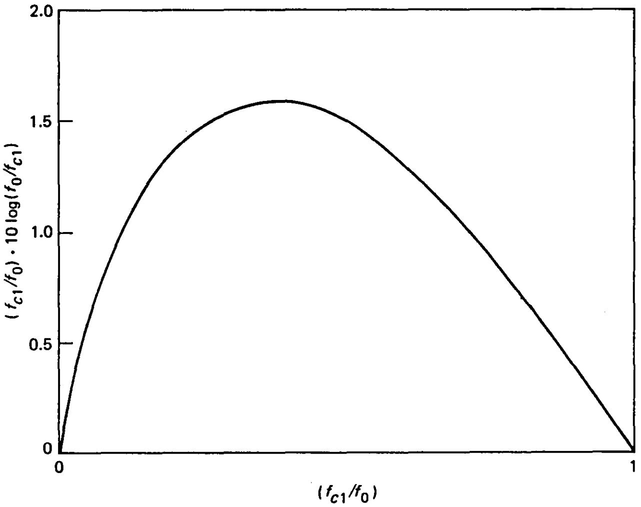


Fig. 8—Segmental s/n gain with $N = 2$ as a function of f_{CL}/f_0. The maximum value of 1.6 dB occurs for $f_{c1}/f_0 = e^{-1}$. The scheme of Fig. 6 suggests $f_{c1}/f_0 = 0.5$ for $N = 2$. This produces an s/n gain only 0.1 dB less than the maximum value of 1.6 dB.

but in terms of absolute quality, the AQF coders were clearly better, especially at 16 kilobit/s. AQF coders, however, require the explicit transmission of step-size information; using for example, four bits once for every segment of 256 samples (16 ms).

The cutoff frequency $f_L(t)$ was computed once for every segment of 256 samples. These segments were Hamming-windowed, zero-padded for better frequency resolution, and analyzed by means of a 512-sample FFT.

The filter banks consisted of 33-point FIR filters whose frequency responses are shown in Figure 9. Although gains due to a two-filter system were almost always noticeable, they were not always significant; specifically they were not significant for inputs with predominant all-pass ($> f_0/2$) segments; see Figs. 3 and 4. A four-filter bank, on the other hand, provided significant quality improvement with all the four test inputs. The perceived improvement in quality was much greater in all cases than what the theoretically predicted objective gain of $G(4) = 2.6$ dB indicates. The measured gains were very input-dependent with an average value slightly less than the theoretical value of $G(4)$. This result reinforces earlier results for time-varying filtering of ADM speech,[1] where again perceptual gains were in excess of objectively measured segment s/n improvements; this is probably because the residual in-band noise after adaptive filtering is much less annoying because of masking by the input signal. Adaptive bandwidth ADPCM, of course, has the additional possibility of variable bit allocation. For

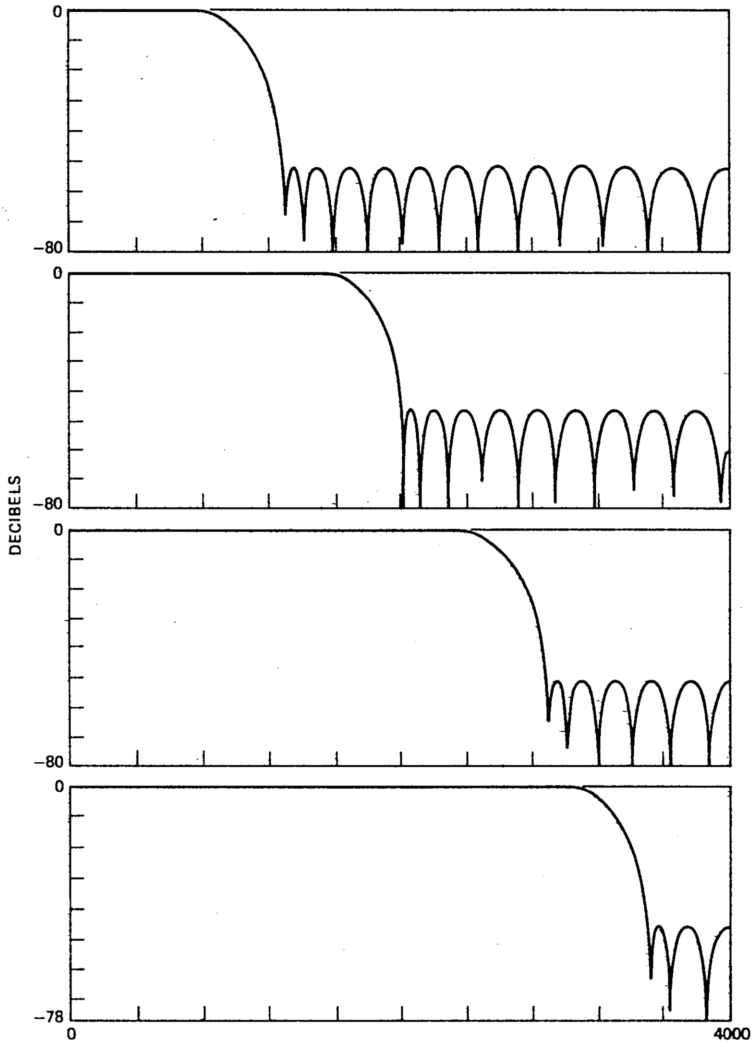


Fig. 9—Frequency responses of low-pass filters in an $N = 4$ filter bank. Each filter is a 33-point FIR design, and the cutoffs correspond to those in Fig. 5c and Fig. 6, with $f_0 = 3.2$ kHz.

example, significant quality improvements have been noticed in a system where the input was subsampled at 5.33 kHz whenever $f_L(t) < 2.4$ kHz, and more quantization bits were allocated for subsampled segments. For example, in 16 kilobit/s operation these segments were coded with three-bits/sample instead of two. This type of variable bit allocation is ruled out by definition in ADM which is a one-bit/

sample system. Notice finally that ADPCM with out-of-band noise rejection and variable bit allocation is very similar to frequency-domain subband coding.[2]

V. CONCLUSIONS

We have indicated a conceptually simple but very effective procedure for improving the quality of ADPCM speech coding. The extra information for effecting this improvement is very little; with a four-filter design this extra information would be $\log_2 4$ = two bits, corresponding to four possible ranges of cutoff frequency $f_L(t)$. The complexity involved in terms of computing $f_L(t)$ is quite significant in relation to the simplicity of the conventional, basic ADPCM coder. The increased complexity, however, will be relatively less objectionable in voice storage applications than in transmission systems; and in either case, an attractive feature is that the post-filtering procedure (the boxed portion of Fig. 6) can be used as an entirely optional refinement. Also, as noted earlier, the adaptive post-filtering technique discussed in this paper can be used with significant gains in the context of coders other than ADPCM.[1]

VI. ACKNOWLEDGMENT

This work was prompted by a summer project on post-filtered ADM by J. B. Allen and J. O. Smith.[1]

REFERENCES

1. J. O. Smith and J. B. Allen, "Variable Bandwidth Adaptive Delta Modulation," B.S.T.J., this issue.
2. J. L. Flanagan et al., "Speech Coding," IEEE Trans. Commun., *COM-27*, No. 4 (April 1979), pp. 710–37.
3. N. S. Jayant, "A First-Order Markov Model for Delta-Modulation Noise Spectra," IEEE Trans. Commun. (August 1978), pp. 1316–18.
4. N. S. Jayant, "Step-Size Transmitting Differential Coders for Mobile Telephony," B.S.T.J., *54*, No. 9 (November 1975), pp. 1557–81.

Copyright © 1981 American Telephone and Telegraph Company
THE BELL SYSTEM TECHNICAL JOURNAL
Vol. 60, No. 5, May–June 1981
Printed in U.S.A.

Variable Bandwidth Adaptive Delta Modulation

By J. O. SMITH* and J. B. ALLEN

(Manuscript received December 3, 1980)

The ADM(adaptive delta modulation) speech coder is generally used with a time-invariant low-pass filter at the decoder output. The purpose of this low-pass filter is to reject coder noise at frequencies above the fixed speech passband. The speech spectrum, however, tends to occupy only the lower frequencies within the passband during voiced speech, and is somewhat "high pass" during unvoiced speech. In this paper, we show how the quality of ADM may be significantly improved by adaptively filtering the coder output such as to follow the natural bandwidth of the speech. This was found to reduce drastically the perceived noise in the output of the ADM coding system at low bit rates. The use of an adaptive low-pass filter realizes almost all of this quality gain. (An adaptive high-pass filter seems to reject less audible noise components and seems more prone to introducing objectionable artifacts.) We also discuss a method for reducing the bit rate with little or no sacrifice in quality (relative to normal ADM) by adapting the sampling rate along with the time-varying low-pass filter.

I. INTRODUCTION

In this paper, we explore two methods for better utilizing the time-varying bandwidth of speech in ADM (adaptive delta modulation) coders. In the first method, the ADM speech quality is shown to be improved by filtering the reconstructed (decoded) speech with a time-varying filter tailored to the natural speech bandwidth. In this case, adaptive bandpass filtering of the ADM output signal reduces coder noise by rejecting noise components at frequencies outside of the principal speech spectrum. Experimentally, we found that eliminating the upper 2 percent of the spectrum energy gave a reduction in average

* Presently a graduate student at the Information Systems Laboratory, Department of Electrical Engineering, Stanford University.

bandwidth on the order of a factor of two relative to an initial 3-kHz bandwidth for typical speech samples. If the coder noise is white, there is an average noise power reduction by a factor proportional to the bandwidth reduction. Furthermore, the remaining portion of the noise power lies entirely within the band of the speech so that for reasonably good signal-to-noise ratios, some masking of the noise by the speech can be expected.

The second case we explore is one of an adaptive sampling rate. In this case, the noise is again eliminated outside the principal speech bandwidth with a time-dependent low-pass filter. Then the average bit rate is reduced by a time-dependent decimation.

1.1 Block diagram

Figure 1 shows a block diagram of the system implemented in software for the tests to be presented. The system includes estimation of the short-time bandwidth (discussed in Section II), time-dependent filtering to this bandwidth, sampling rate conversion via decimation/interpolation,[1] and a 1-bit memory ADM coder with exponential step-size adaption.[2] For discursive purposes, we regard each of the two bandpass filters as a cascade of independently controlled low-pass and high-pass filters. The details on the implementation of this system are given in Appendix A.

1.2 Test cases studied

For clarity we define names for the following four cases studied:

Normal ADM (ADM)—Both bandpass filters in Fig. 1 are fixed at the full voice-channel bandwidth, and the sampling rate is fixed. For example, 24 kbps ADM is implemented with a constant sampling rate and both filters are set to pass frequencies from 200 to 3200 Hz at all times (see Ref. 2).

Post-filtered (ADM-PF)—Only the receiver reconstruction filter (at the far right in Fig. 1) varies to match the speech spectrum. The transmitter input filter is fixed at the channel bandwidth, and the sampling rate is fixed.

Pre- and Post-filtered (ADM-PPF)—Both the input and output filters are made to track the speech bandwidth, but the sampling rate remains fixed as in ADM-PF. The addition of adaptive prefiltering allows the ADM coder to track the speech waveform with less error.[3]

Adaptive Rate (ADM-AR)—The sampling rate varies at twice the upper cutoff frequency, and both the input and output filters track the speech bandwidth.

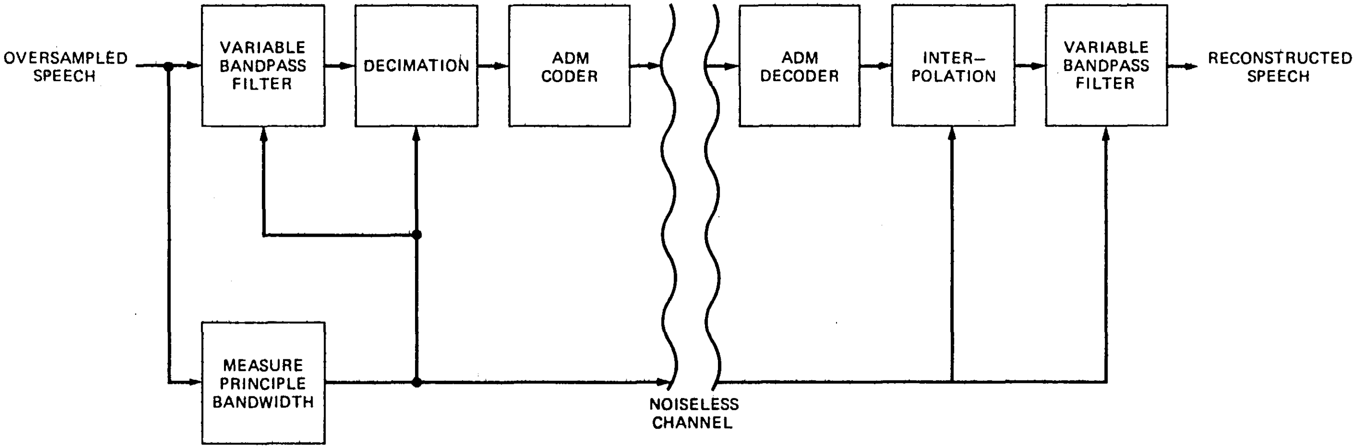


Fig. 1—Block diagram of the variable-rate ADM coding system with adaptive bandpass filtering. The speech bandwidth is measured in real time to control the variable bandpass filters and ampling-rate conversion (decimation/interpolation). In the simplest case, only the right-most filter is varied along with the speech spectrum to provide coder noise suppression. In the case of adaptive sampling rate, both filters and the sampling rate are tailored to the speech.

1.3 Results

The main conclusions are:

(*i*) Post-filtering gives a significant increase in quality. For example, 16 kbps ADM-PF gives a quality commensurate with 24 kbps ADM without post-filtering. The signal to noise ratio (s/n) is increased primarily in the low bandwidth segments such as back vowels, nasals, and voiced stops.[4] Almost all of the improvement arises from the adaptive low-pass component of the filtering. The adaptive high-pass filter contributes only slight noise suppression, and can introduce undesired side effects; for example, when there is a rapid transition from voiced to unvoiced, in which the speech band goes from low pass to high pass, an audible and objectionable change in the coder noise can occur even though there is an improved s/n due to the rejection of out-of-band low-frequency coder noise. Thus, adaptive low-pass filtering improves quality without serious side effects, while adaptive high-pass filtering only slightly improves s/n and causes significantly more audible noise modulation. For ADM, these effects are most pronounced at 24 kbps and below.

(*ii*) For the case of ADM-AR (prefiltering, adaptive sampling rate, and post-filtering) we found that adapting the sampling rate causes low-pass time frames (such as voiced segments) to have a degraded s/n compared to the ADM s/n; however, for these frames, the bit rate is substantially reduced. Furthermore, while the in-band coder noise is increased, the out-of-band coder noise is eliminated. Consequently, the quality of ADM-AR is different from ADM but not easily judged to be worse. Informal listening tests indicated no reliable preference for one over the other for the few samples of speech tested (base bit rates of 16, 24, and 32 kbps).

Summarizing positive practical results, our simulations indicate that time-dependent (adaptive) low-pass post-filtering yields a significant quality increase (for bit rates of 24 kbps and below) and adaptive low-pass pre- and post-filtering plus adaptive sampling rate yields reduced average bit rate with little change in quality.

In Section II, we discuss how the time-dependent filter cutoff frequencies are measured from the short-time speech spectrum. Section III presents simulation results for the four cases defined above.

II. MEASUREMENT OF THE TIME-VARYING SPEECH BANDWIDTH

Given the short-time spectrum of the speech at a given time, we wish to define the upper and lower cutoff frequencies of the spectrum in a way that minimizes bandwidth without introducing significant quality loss in the bandlimited speech. For this purpose, we define two constants T_L and T_U which may be thought of as the fractional energy of the speech bandwidth to be removed.[5] T_L and T_U are taken to lie between 0 and 1. We call T_L the lower cutoff threshold and T_U the upper cutoff threshold. If $X(t, f)$ denotes the short-time spectrum of the speech at time t, then the time-varying high-pass and low-pass cutoff frequencies are found by solving

$$T_U = \frac{1}{E(t)} \int_{f_U(t)}^{\infty} |X(t, f)|^2 \, df,$$

$$T_L = \frac{1}{E(t)} \int_{0}^{f_L(t)} |X(t, f)|^2 \, df, \tag{1}$$

for $f_U(t)$ and $f_L(t)$, where $E(t)$ is the total spectrum energy at time t given by

$$E(t) = \int_{0}^{\infty} |X(t, f)|^2 \, df. \tag{2}$$

Note that $f_U(t)$, the high-frequency (or low-pass) cutoff, and $f_L(t)$, the low-frequency cutoff, vary to maintain constant T_U, T_L.

The discrete-time, discrete-frequency definitions that result from using the discrete Fourier transform (DFT) to generate short-time spectra are exactly analogous. When discussing sampled data, we write n in place of t, and the sampling rate will be denoted by $f_s = 1/T$.

The upper cutoff threshold T_U is the fixed fraction of the total energy that is rejected by the time-varying low-pass filter, and similarly, T_L controls the time-varying high-pass filter. These constants are chosen in accord with desired coder quality. Ideally, the values of T_L, T_U might be optimized to trade off bandwidth for suppressed coder noise. In the spirit of Wiener filter theory, we might define the optimum thresholds as the values for which a decrease of either results in more added coder noise than added signal in the reconstructed speech, and where an increase of either value causes more distortion loss due to bandlimiting than quality gain from noise excision. However, we do not know how to define objective measures of subjective degradations due to changes in bandwidth and coder noise. In our tests, the thresholds T_L and T_U were set such that they did not appreciably degrade the speech quality in the absence of coder noise. That is, rather than attempt to define optimal thresholds for each bit rate, we wish merely to estimate the benefits of variable bandwidth when no perceptually significant distortion results from the bandlimiting alone. Accordingly, in all ADM coder simulations, where the initial speech bandwidth is 0.2 to 3.2 kHz, the values $T_L = 2$ percent and $T_U = 1$ percent were used to specify the time-varying filters (and sampling rate when applicable).

An example of the passband behavior for these threshold values is given in Fig. 2. The phrase analyzed was from an adult male speaker. Note that within the telephone passband, the speech is basically either low pass or high pass at any given time. It is these temporal speech bandwidth variations that we exploit for noise and sampling rate reduction in ADM.

Figure 3a gives a spectrogram of the same speech sample, and Fig. 3b shows the spectrogram after filtering the speech to the bandlimits shown in Fig. 2. We see that the 1 percent energy upper cutoff limit tends to follow the third formant during voiced regions, and the 2 percent lower cutoff limit has an almost unobservable effect. (The 2 percent high pass has an audible effect on unvoiced phonemes, however.) Figure 4a shows the output of a 16-kbps normal ADM coding system (time-invariant filters), and Fig. 4b shows the effect of post-filtering. The audible improvement due to post-filtering is much like Fig. 4b suggests, namely the out-of-band noise has been removed in Fig. 4b. This particular sample of post-filtered 16-kbps coded speech sounds about as good as when coded with normal 24-kbps ADM.

Figure 5 gives a plot of the *average* bandlimits (averaged over the entire utterance) as a function of thresholds. When $T_U = T_L = 0$, the passband is identical to the original speech passband; as the thresholds approach one half, the passband converges to zero. Comparing the two traces, we see that the speech is primarily low pass, which correlates with the fact that the utterance is predominantly voiced. Note that Fig. 5 implies an average bandwidth of only one half the maximum bandwidth using the values $T_U = T_L = 1$ percent. In other words, it is possible to reduce the average sampling rate by a factor of two while sacrificing only 2 percent of the spectral energy.

A few remarks are in order concerning practical issues associated with the measurement of the time-varying speech bandwidth. When tracking any spectrum over time, it is necessary to employ the proper balance of frequency resolution versus time resolution in the spectral analysis.[4] For speech, we wish to track bandwidth changes correspond-

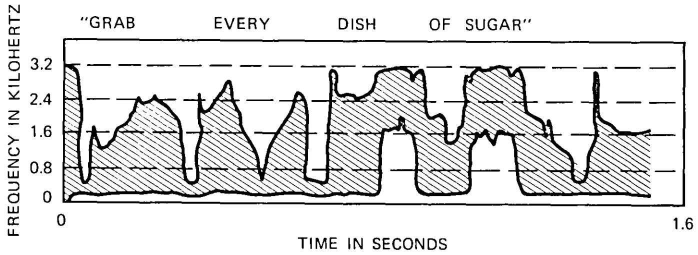


Fig. 2—Spectral band edges vs time for a male speaker, obtained by rejecting the upper 1 percent and the lower 2 percent of the 200–3200 Hz spectrum energy. Band-edge values are computed every 12.5 ms.

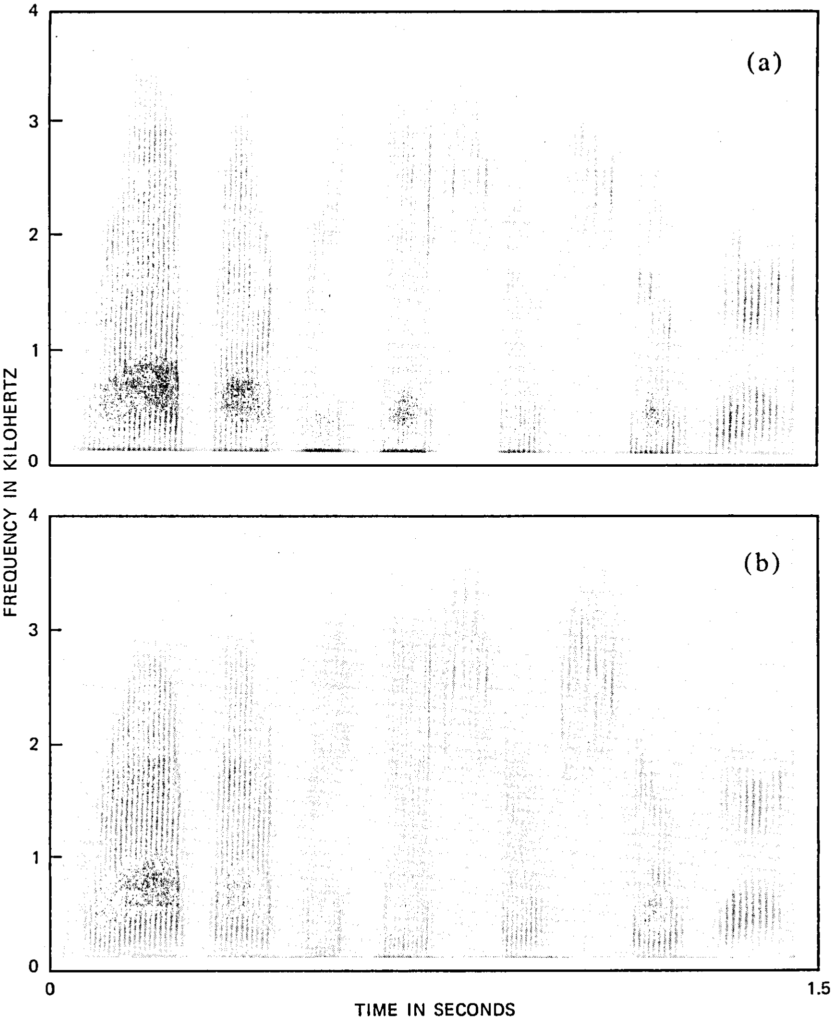


Fig. 3—Spectrograms of 8 kHz sampled speech, before and after filtering to the measured speech bandwidth. (a) Original speech spectrum within the fixed channel bandwidth of 200–3200 Hz. (b) Same speech after filtering with a time-varying bandpass, which rejects the upper 1 percent and lower 2 percent of the spectrum energy in the band. This filtering is approximately transparent.

ing to the articulation of phonemes. Tracking should be rapid so that there is little or no "smearing" of the estimated band-edges at the juncture of two dissimilar phonemes, and this implies using a small integration frame (window) in computing the short-time spectrum. However, when the spectrum is based on a frame length that is less than a pitch period, we obtain spectra that fluctuate excessively (at the pitch frequency) due to differing decay times of the vocal tract

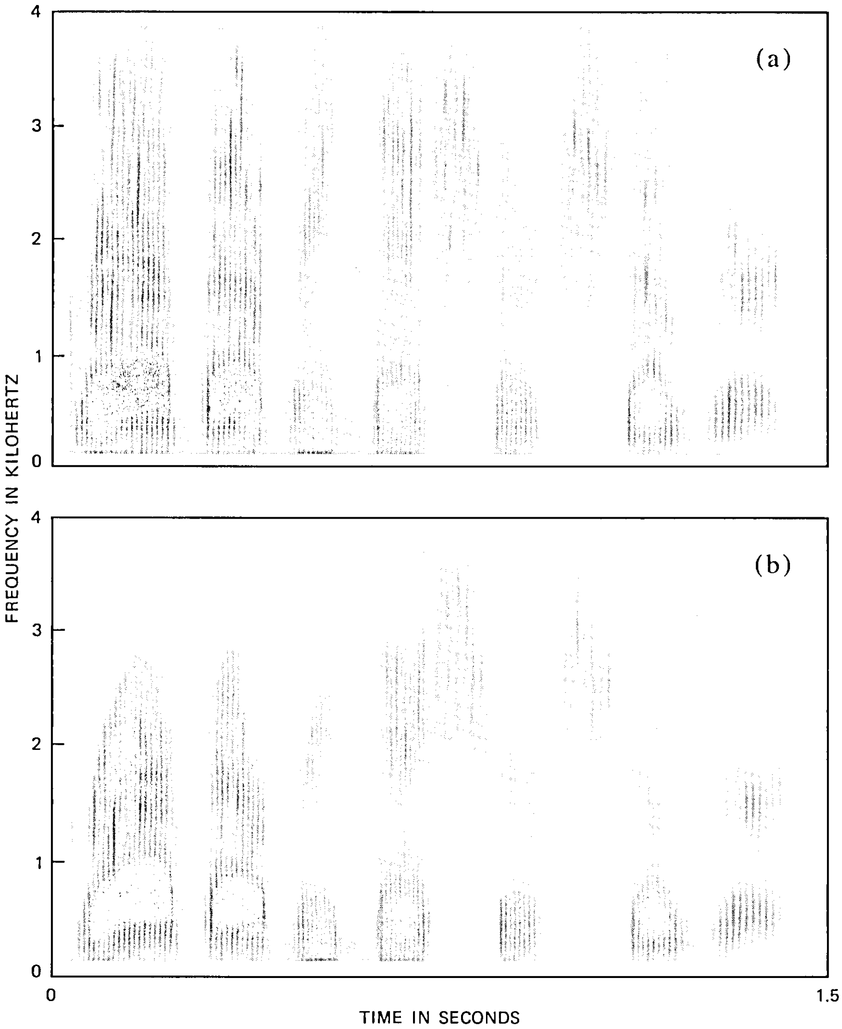


Fig. 4—Spectrograms of the same speech as in Fig. 3 at the output of a 16 kbps ADM system before and after time-varying filtering. (a) Fixed-bandwidth coder output. (b) Variable-bandwidth coder output. The time-varying filter is controlled by the same band edges as in Fig. 3b, i.e., the band edges are measured from the speech at the coder input. The effect of this filtering is to reduce significantly the ADM coder noise.

impulse response components. For this reason, it is desirable to include at least 20 ms of speech in each spectrum computation, corresponding to the observation that the pitch of voiced speech rarely, if ever, falls below 50 Hz. As Fig. 6a shows, when the spectrum analysis integration time is less than a pitch period, the time-varying low pass cutoff $f_U(n)$ can oscillate quite significantly (e.g., ±20 percent) at the pitch frequency. In Figs. 6b,c we show the effects of a seven-point median smoother and a seven-point moving average smoother on the data of Fig. 6a.

Another implementation issue is that of using the time-varying cutoff frequencies to control the output filters. Care must be taken to match the spectral analysis integration time, sampling interval for the cutoff frequencies, and filter impulse-response duration. These three times should be comparable in magnitude. In Ref. 6, it is shown that for the case of FFT-based (fast Fourier transform) analysis and filtering, the minimum sampling rate for the filter cutoff frequencies is determined by the window used on the input to the FFT. For a length N FFT with a Hamming window, the band-edges may be sampled every $N/4$ samples (i.e., successive FFTS used for calculating f_U, f_L may be offset in time by $N/4$ samples). It is shown in Ref. 7 that the resulting time-varying filter will have properly bandlimited coefficients regardless of the spectral modifications made on each FFT.

Our formulation may be altered slightly to provide excellent performance during regions of silence. This is called the "idle channel" condition in the ADM literature. Inspection of (1) and (2) reveals that

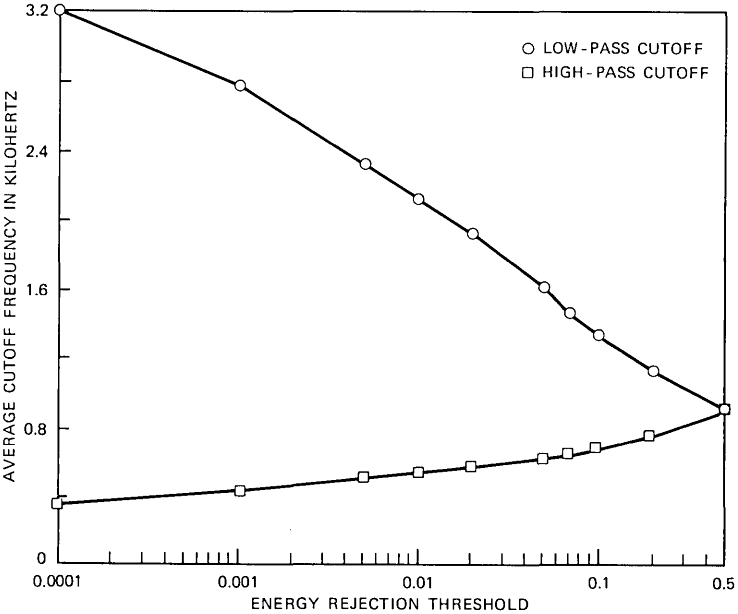


Fig. 5—Time-averages of the upper and lower band edges vs spectrum energy-rejection threshold. The average is taken over the entire utterance of Fig. 2 for each threshold. At the far left of the figure, the energy-rejection thresholds are near zero so that the band edges lie at the outer extremes of the true speech band. At the far right, the two thresholds are equal to 0.5 corresponding to rejection of the upper and lower 50 percent of the speech spectrum; consequently, the band edges meet at the median frequency.

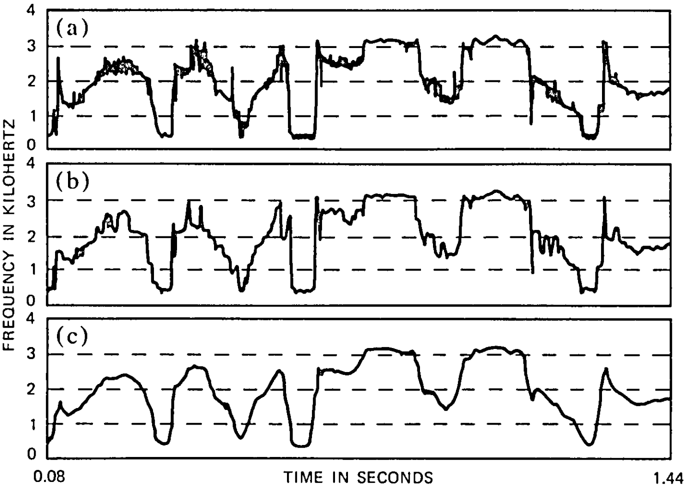


Fig. 6—Illustration of the effects of using an under-sized Fourier transform, and the possibility of compensation via filtering of the band-edge waveforms. (a) Upper 1 percent band edge for the same speech sample as in Fig. 2 using only a 10-ms time frame for spectrum computation. This causes oscillation of the measured band edge at the pitch frequency. (b) The same band-edge function of (a) after filtering with an order 7 median smoother. (c) The same curve of (a) after filtering with an order 7 moving average. Linear smoothing results in band-edge time behavior similar to that obtained when using a larger Fourier transform; however, the frequency resolution of the band-edge values is still sacrificed.

the cutoff frequencies are indeterminate in this situation $[|X(\omega, t)| = 0]$. If there is any amount of white noise present in the input signal, then the time-varying bandwidth, $f_U - f_L$, will open to full bandwidth as if the speech itself were spectrally flat. This undesirable behavior may be suppressed by various ad hoc schemes. In the simulations, we added a small positive value to $E(t)$. That is, (2) is replaced by

$$\tilde{E}(t) = \int_0^\infty |X(t, f)|^2 \, df + \sigma_{\min}^2, \tag{3}$$

where $\sigma_{\min}^2$ may be thought of as noise energy, or as a lower bound on the acceptable speech level. As the speech energy falls to zero, the band edges cross, corresponding to disjoint low-pass and high-pass filters, and we must therefore define all negative values of $f_U - f_L$ to be zero bandwidth. Also, there exists the possibility that no solution to (1) exists, for $\sigma_{\min}^2 > 0$, in which case the bandwidth is again set to zero. Thus, when the channel is idle, there will be zero bandwidth and subsequently no output signal. This fact can be used to advantage when optimizing the step-size adaption algorithm in the ADM coder.[3]

Our simulations assume that the band edges f_U and f_L are transmitted as side information in the variable bandwidth coding system. However, it is worth considering filter adaption based only on the received speech data. Increasing the receiver energy rejection thresholds T_U and T_L relative to the transmitter thresholds will contract the (estimated) band edges so as to compensate for the artificial band expansion that occurs because of coder noise in the received spectrum.

If the bandlimits are transmitted as side information, then the increase in data rate is relatively small. As a practical example, if the FFT length is 512, a Hamming window is used, and the speech sampling rate is 8 kHz, then we have a pair of band edge values every 16 ms. Furthermore, the band edge values are quite smoothly behaved, and can be coded more efficiently. It appears that the band-edge waveform signals $f_L(t)$ and $f_U(t)$ have a bandwidth on the order of 30 Hz for speech.[5]

III. RESULTS OF ADM SIMULATIONS

In this section, we present s/n evaluations of the four ADM coder configurations (described in Section I) ADM, ADM-PF, ADM-PPF, and ADM-AR. The comparisons are made using the s/n and segmental s/n[8] measures defined in Appendix B. Detailed parameter information may be found in Appendix C.

The degree to which quality is enhanced by adaptive post-filtering depends on the character of the coder noise. If the coder noise is known to be stationary additive white noise, uncorrelated with the speech, then the gain in s/n may be predicted in advance from f_U and f_L. Given that bandlimiting the speech causes no distortion, the s/n of each segment will increase by

$$\begin{aligned}\text{s/n increase (dB)} &= 10 \log \frac{\text{(maximum bandwidth)}}{\text{(short-time bandwidth)}} \\ &= 10 \log \left(\frac{f_{\max}}{f_U - f_L} \right), \qquad (4)\end{aligned}$$

where $f_{\max}$ is the full channel bandwidth. For a "typical" frame ($f_U =$ 2 kHz, $f_L = 400$ Hz, $C = 3$ kHz), this is about 2 dB. The gain in quality at lower bit rates is dramatically greater than indicated by the s/n. This is perhaps due to the high perceptual significance of out-of-band coder noise and/or auditory masking of in-band noise by the speech.

To anticipate the improvement of ADM due to post-filtering, we need to know the spectral distribution of ADM coder noise. While some theoretical work along these lines has been done,[9] it is difficult analytically to derive general estimates of the short-time noise power spectral density. Some intuition may be obtained, however, from simulations on isolated, quasi-stationary speech segments.

Figure 7a shows a tenth-order LPC spectral envelope[10] for the front

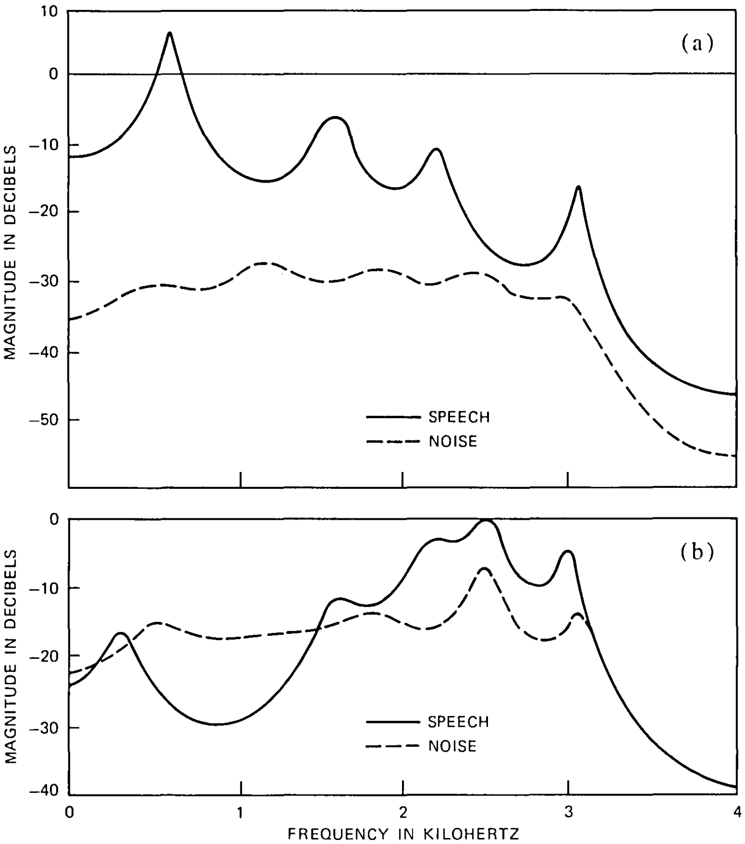


Fig. 7—Spectral envelopes of signal and noise for two representative sounds. All spectral envelopes were calculated using a tenth-order linear prediction on 1024 samples of data. The noise was isolated from the speech by subtracting the noiseless (precoder) speech from the ADM coder output (which used time-invariant filtering). (a) Spectral envelope for the vowel "a" from "grab" superimposed with the spectral envelope of its associated ADM coder noise. (b) Spectral envelopes for the "s" sibilant in "sugar" and its corresponding coder noise.

vowel "a" (as in "grab") superimposed with the tenth-order spectral envelope of the normal 24-kbps ADM coder noise generated by this vowel. (Appendix C gives detailed analysis parameters.) The measured s/n is 15 dB, and the noise spectrum within the passband is relatively flat. It should be noted that the slight ripple in the spectral envelope of the error signal depends on the order of the linear predictor.

Figure 7b shows the same comparison of signal and noise spectral envelopes for the "sh" sound in "sugar." Note that in this case, the noise is fairly flat out to 2.4 kHz after which it begins to follow the speech spectrum. The noise has a significant peak near 2.6 kHz indicating that these spectral components could not be properly tracked. In this example, it was evident from the time-domain waveform that the coder was tracking dominant high-frequency components with a large positive error in the amplitude difference estimate (adaptive step-size inside the ADM coder).[2] The measured s/n for this sibilant is only 1.6 dB, and the primary character of the noise is that of rough loud "static."

Generalizing from Fig. 7, we might expect low-pass signals to generate coder noise that may be approximately modeled as white, and high-pass speech segments to correspond to relatively strong correlated noise. Such heuristics, while over-simplified, serve to point out the more generally observed differences in ADM noise characteristics for voiced vs unvoiced speech. Awareness of these two contrasting cases aids in the interpretation of the segmental s/n in which the s/n for individual phonemes is evident.

Figure 8b gives a plot of the segmental s/n (defined in Appendix B) versus time for the three cases ADM, ADM-PF, and ADM-AR. The bit rates of ADM and ADM-PF are 32 kbps. ADM-AR has 32 kbps as its maximum instantaneous bit-rate while the average rate for this particular phrase is 23 kbps. Figure 8a shows the segmental input rms level from which the various phonemes may be located. The segment size is

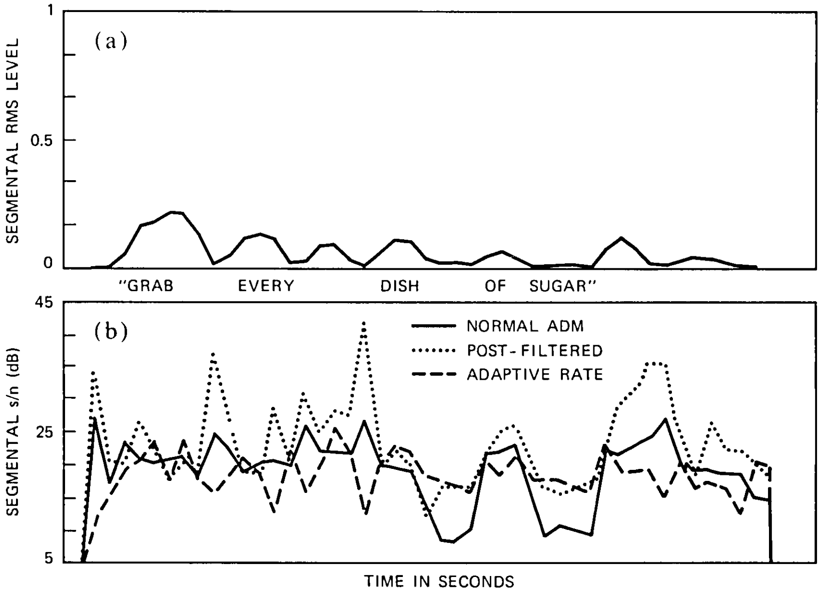


Fig. 8—Time behavior of ADM noise for three cases as defined by the s/n of each 32-ms time frame (segmental s/n). (a) Segmental rms amplitude of speech utterance vs time indicating phoneme locations. (b) Segmental s/n vs time for normal ADM and post-filtered ADM (ADM-PF) at a bit rate of 32 kbps, and adaptive-rate ADM (ADM-AR) having a peak bit rate of 32 kbps.

32 ms in both Figs. 8a and 8b. We may observe several features in the behavior of the segmental s/n due to post-filtering and adaptive rate:

(*i*) There is little difference among the three cases for front vowels such as "a" in "grab" and "e" at the beginning of "every." From Fig. 2, we see that during these segments, the speech bandwidth is wide and almost fully occupies the channel bandwidth. Consequently, the adaptive low pass is almost the same as the fixed low pass, and ADM-AR is running at maximum sampling rate during the greater portion of these vowels.

(*ii*) When the low-pass cutoff $f_U(n)$ is small, ADM-PF realizes large quality gains due to rejection of much out-of-band coder noise. In contrast, ADM-AR exchanges these gains in return for reduced sampling rate. Examples of this may be seen at the phonemes corresponding to "b," "v," "d," and "u."

(*iii*) When the high-pass cutoff $f_L(n)$ is large [at which time $f_U(n)$ is maximum], ADM-AR reduces to the case ADM-PPF, and its performance is close to that of ADM-PF. Both exhibit higher segmental s/n than normal ADM due to elimination of low-frequency noise. This condition may be observed at the two unvoiced regions "sh" and "s."

We now turn to plots of segmental s/n averaged over the entire utterance, and we denote the average segmental s/n by s/n$_{\text{seg}}$. Figure 9 shows s/n$_{\text{seg}}$ vs bit rate for all four test cases. The post-filtered case, ADM-PF, is 2.8 dB better than normal ADM on the average. Note that the prefiltering in ADM-PPF, which reduces ADM tracking error, adds

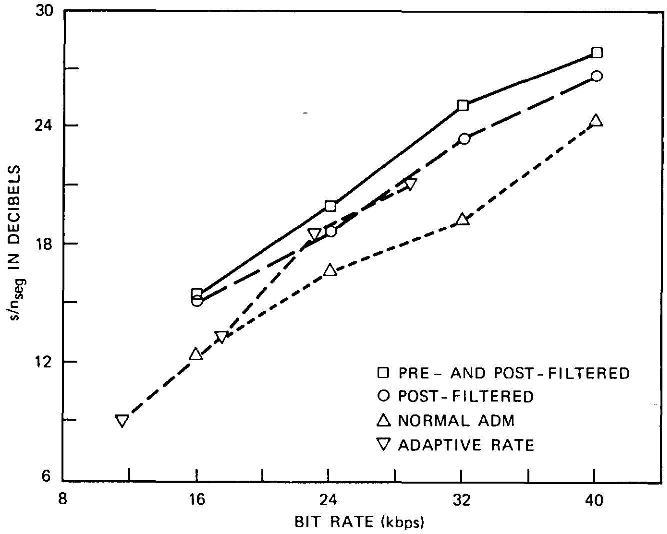


Fig. 9—Segmental s/n averaged over the entire utterance for four cases, plotted as a function of bit rate. For the case of adaptive rate, the average bit rate is used as abscissa.

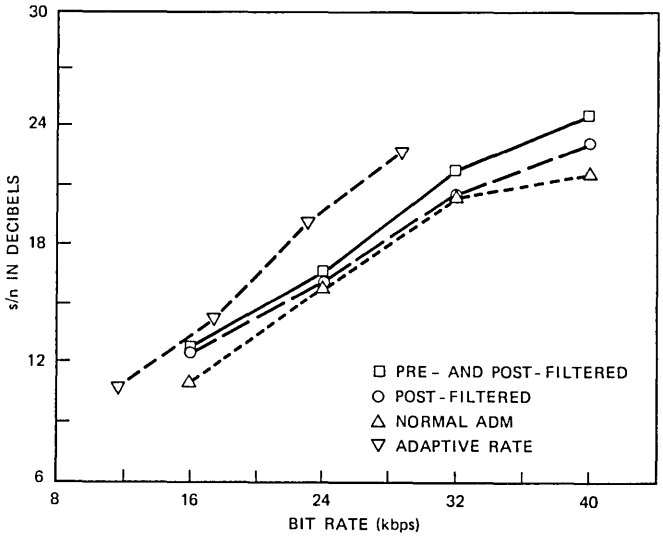


Fig. 10—Normal s/n computed on entire utterance for four cases, plotted as a function of bit rate. For the case of adaptive rate, the average bit rate is used as abscissa.

still another dB or so to the s/n_{seg} for ADM-PF, and the improvement is always within the short-time speech band. The adaptive rate coder, ADM-AR, exhibits s/n_{seg} between normal and post-filtered ADM. Overall, the s/n_{seg} measure corresponds well with subjective quality ratings. From informal listening to the speech samples represented in Fig. 9, we feel that ADM-PPF is not noticeably better than ADM-PF; ADM-PF is somewhat "cleaner" than ADM-AR (comparing where the maximum ADM-AR rate equals the ADM-PF rate), and normal ADM is definitely inferior due to the audible high-frequency noise which is allowed to pass.

Figure 10 gives s/n (as opposed to s/n_{seg}) for the same four cases (cf. Appendix B). All deviations from Fig. 9 are due to the fact that the s/n_{seg} measure is an average of the s/n's (in dB) obtained from disjoint 256-point frames while the s/n measure treats the entire speech sample as one frame. Since the regions of quality gain in ADM-PF and ADM-PPF are of low relative energy, they contribute little to the s/n. ADM-AR appears in this figure to be significantly superior, but this is misleading; ADM-AR has high distortion in the relatively low-energy low-bandwidth regions due to the large reduction in sampling rate, and the s/n measure does not adequately penalize it. For example, the consonants "b" and "d" might be least distinguishable in the ADM-AR case, relative to the other three, even though it scores the highest s/n. Thus, the s/n measure is overly insensitive to low-amplitude intelligibility loss, especially in the case of ADM-AR.

IV. CONCLUSIONS

It has been shown that the quality of ADM coded speech can be significantly improved by employing a time-dependent low-pass filter matched to the short-time speech bandwidth. A time-varying high-pass cutoff may be added with little additional computational cost, but its contribution to quality is small and sometimes perceptually distracting due to audible noise modulation at bit rates below 24 kbps. Transmission of the slowly varying cutoff frequencies adds only slightly to the transmission bit rate. Two uses of the adaptive low-pass cutoff were discussed. First, time-varying low-pass filtering of the ADM decoded signal was found to add quality commensurate with a large increase in ADM bit rate (e.g., 24 kbps quality at 16 kbps). Secondly, time-varying low-pass filtering before and after the ADM coder, coupled with a time-varying sampling rate, gave nearly the same quality as normal ADM but with a large reduction in the average bit rate (e.g., 24 kbps from 32 kbps). The gains cited are for continuous speech, and better relative performance is to be expected for speech containing regions of silence. The final conclusions concerning quality are based on casual listening tests and are only indirectly supported by the s/n measures employed.

V. ACKNOWLEDGMENTS

The authors wish to thank N. S. Jayant for sharing his expertise on ADM coding, and for recurrent help throughout the project. We also thank J. L. Flanagan for numerous ideas and suggestions which were incorporated into this investigation.

APPENDIX A

Implementation of Variable Bandwidth ADM Simulation

Referring again to Fig. 1, the software implementation is as follows. The input speech is sampled at 8 kHz, bandlimited to the typical channel bandwidth for telephone communication (200–3200 Hz), and is then resampled at 16, 24, 32, or 40 kHz. The data is partitioned into overlapping frames of 512 samples, a Hamming window is applied,[11] and the FFT of each frame is taken. The speech cutoff frequencies f_U and f_L are computed for each frame, as discussed in Section II.

If prefiltering is included or if the sampling rate is to be lowered, the spectrum values outside the cutoff frequencies are tapered to zero using a precomputed filter band edge. The filter band edge is computed using a window design method based on a Kaiser window.[11] Next, an inverse FFT is taken on each frame, and the time-domain waveform is reconstructed by adding the frames back together, partially overlapped in time (overlap-add synthesis[6]).

The decimation stage is only active during sampling rate reduction, and it operates by selecting every mth sample, where $m = \lceil 0.5\, f_s/f_U \rceil$ is the sampling rate reduction factor. $\lceil x \rceil$ denotes the smallest integer $\geq x$, namely, m is the greatest integer such that m times the low-pass cutoff frequency for the current frame does not exceed the upper channel band edge. Note that the integer decimation method of varying the sampling rate does not take full advantage of the unused bandwidth; however, it has the advantage that it is quite simple to implement.

The coder is a one-bit ADM coder with exponential step-size adaptation as described in Ref. 2. The coder output and the time-varying bandwidth information are assumed to be transmitted through a noiseless channel.

The ADM decoder is followed by a sample interpolator to restore the original sampling rate (when applicable), and the interpolator is followed by a time-varying filter. This filter is also implemented via short-time spectrum analysis, modification, and synthesis; it restricts the decoded speech spectrum to its original natural bandwidth, when post-filtering is employed, thus removing out-of-band coder noise. This filter is also part of the interpolation process as the interpolator merely inserts $m = \lceil 0.5\, f_s/f_U \rceil$ zeros between each sample.

APPENDIX B

Signal-to-Noise Ratio Calculation

Two types of signal-to-noise ratio are defined. The most common form is

$$\text{s/n} \triangleq 10 \log \left| \frac{\sum_{m=0}^{N-1} (x(m) - \mu_x)^2}{\sum_{m=0}^{N=1} (e(m) - \mu_e)^2} \right|,$$

where $x(m)$ is the signal with sample mean

$$\mu_x \triangleq \frac{1}{N} \sum_{m=0}^{N-1} x(m),$$

$e(m)$ is the noise with sample mean μ_e, and N is the total number of samples available for the s/n measurement.

The definition of s/n diverges from subjective quality ratings for large N due to the fact that high-amplitude signal regions dominate the influence of low-amplitude signal regions during the s/n calculation. This insensitivity may be partially circumvented by computing s/n values over segments of some reasonably small size M (e.g., spanning 20 ms), and averaging the s/n (dB) values of the segments. Accordingly we define

$$\text{segmental s/n}(k) \triangleq 10 \log \left| \frac{\sum_{m=0}^{M-1} [x_k(m) - \mu_x(k)]^2}{\sum_{m=0}^{M-1} [e_k(m) - \mu_e(k)]^2} \right|,$$

$$\text{s/n}_{\text{seg}} \triangleq \frac{1}{N} \sum_{k=0}^{N-1} \text{segmental s/n}(k)$$

$$= \frac{1}{N} \sum_{k=0}^{N-1} 10 \log \left| \frac{\sum_{m=0}^{M-1} [x_k(m) - \mu_x(k)]^2}{\sum_{m=0}^{M-1} [e_k(m) - \mu_e(k)]^2} \right|,$$

where N is the number of segments of length M, $x_k(\cdot)$ is the kth segment of the signal, and $\mu_x(k)$ is the sample mean of the kth segment.[8] In the case of s/n$_{\text{seg}}$, the measure is vulnerable to domination by segments having insignificant signal energy (i.e., the s/n can approach $-\infty$ in a time frame where the signal is silent and where there is any amount of noise). Consequently, if the total energy (sum of samples squared) in a given segment is below a prescribed energy threshold, the segment is eliminated from the computation of s/n$_{\text{seg}}$. (This feature was not needed for the continuous speech samples used in the ADM simulations.)

In all ADM tests, the noise $e(m)$ is calculated as the point-wise difference between the noisy coded signal and a signal which was generated in precisely the same way but bypassing the ADM coder. In this way, all side effects of bandlimiting, processing delay, etc., are eliminated from the calculated error. Measurement of s/n in an ADM coding system is facilitated by the fact that it is a waveform coder (as opposed to source coder), and thus does not have the inherent delay, phase-dispersion, or level-offset characteristics that commonly impede the objective measurement of subjective signal quality.

APPENDIX C

System Parameters Used in Generating s/n Curves

Coder input: Phrase = "*Grab every dish of sugar*" from an adult male speaker, sampled at 8 kHz, and bandlimited to 200–3200 Hz with a 256-point FIR bandpass.

Time-varying filters: In all runs, the filters were implemented via modified FFTs of length $N = 512$. To prevent time-aliasing, the number of data points N_x brought into the FFT input buffer plus the length N_h

of the Kaiser window (used as the basis of the time-varying filter) cannot exceed N. Furthermore, short-time spectral modification theory requires that the step-size through the data (time offset between successive FFTS) not exceed $N_x/4$ for the case of a Hamming window on the FFT input.[6] The table below gives the employed data frame size N_x and time-varying filter length N_h as a function of sampling rate f_s for all ADM simulations.

f_s(kHz)	N_x	N_h
16	304	208
24	456	56
32	456	56
40	400	112

The filter controls $f_U(n)$ and $f_L(n)$ are each eight-bit values at a sampling rate of $4f_s/N_x$.

ADM coder: The step-size multipliers were experimentally found to give good results with $P = 1.2$, $Q = 0.9$.[2] These values did better than $P = 1/Q = 1.5$, $P = 1/Q = 1.2$, and a few other trial settings, in terms of the s/n and s/n_{seg} measures.

LPC spectral envelopes: The short data segments "a" and "s" were each processed with $N = 512$, $f_s = 24$ kHz, $N_x = 456$, $N_h = 56$, fixed filters, and nonadaptive sampling rate. The tenth-order LPC spectral envelopes were calculated using 1024 data samples.

REFERENCES

1. R. W. Schafer and L. R. Rabiner, "A Digital Signal Processing Approach to Interpolation," *Digital Signal Processing II*, New York: IEEE Press, 1976. (Also Proc. IEEE, June 1973.)
2. N. S. Jayant, "Adaptive Delta Modulation with a One-Bit Memory," B.S.T.J., *49*, No. 3 (March 1970), pp. 321–42.
3. N. S. Jayant, "Digital Coding of Speech Waveforms: PCM, DPCM, and DM Quantizers," Proc. IEEE, *62*, No. 5 (May 1974), pp. 611–32.
4. L. R. Rabiner and R. W. Schafer, *Digital Processing of Speech Signals*, Englewood Cliffs, N.J.: Prentice-Hall, 1978.
5. J. B. Allen, "A Method for Simultaneous Transmission of Data Over Voice," unpublished work.
6. J. B. Allen, "Short Term Spectral Analysis, Synthesis, and Modification by Discrete Fourier Transform," IEEE Trans. on Acoustics, Speech, and Signal Proc., *ASSP-25*, No. 3 (June 1977), pp. 235–8.
7. J. B. Allen and L. R. Rabiner, "A Unified Theory of Short-Time Spectrum Analysis and Synthesis," Proc. IEEE, *65*, No. 11 (November 1977), pp. 1558–64.
8. P. Noll, "Nonadaptive and Adaptive DPCM of Speech Signals," Polytech. Tiijdschr. Ed. Elektrotech/Elektron (The Netherlands), No. 19 (1972).
9. N. S. Jayant, "A First-Order Markov Model for Understanding Delta Modulation Noise Spectra," IEEE Trans. Comm., *COM-26*, No. 8 (August 1978), pp. 1316–8.
10. J. D. Markel and A. H. Gray, *Linear Prediction of Speech*, Berlin: Springer-Verlag, 1976.
11. L. R. Rabiner and B. Gold, *Theory and Application of Digital Signal Processing*, Englewood Cliffs, N.J.: Prentice-Hall, 1975.
12. J. L. Flanagan et al., "Speech Coding," IEEE Trans. Comm., *COM-27*, No. 4 (April 1979), pp. 710–37.

THE BELL SYSTEM TECHNICAL JOURNAL
Vol. 60, No. 5, May–June 1981
Printed in U.S.A.

A Two-Pass Pattern-Recognition Approach to Isolated Word Recognition

By L. R. RABINER and J. G. WILPON

(Manuscript received December 15, 1980)

One of the major drawbacks of the standard pattern-recognition approach to isolated word recognition is that poor performance is generally achieved for word vocabularies with acoustically similar words. This poor performance is related to the pattern similarity (distance) algorithms that are generally used in which a global distance between the test pattern and each reference pattern is computed. Since acoustically similar words are, by definition, globally similar, it is difficult to reliably discriminate such words, and a high error rate is obtained. By modifying the pattern-similarity algorithm so that the recognition decision is made in two passes, we can achieve improvements in discriminability among similar words. In particular, on the first pass the recognizer provides a set of global distance scores which are used to decide a class (or a set of possible classes) in which the spoken word is estimated to belong. On the second pass we use a locally weighted distance to provide optimal separation among words in the chosen class (or classes), and make the recognition decision on the basis of these local distance scores. For a highly complex vocabulary (letters of the alphabet, digits, and three command words), we obtain recognition improvements of from 3 to 7 percent using the two-pass recognition strategy.

I. INTRODUCTION

As illustrated in Fig. 1, the "standard" pattern recognition approach to isolated word recognition is a three-step method consisting of feature measurement, pattern similarity determination, and a decision rule for choosing recognition candidates. This pattern recognition model has been applied to a wide variety of word recognition systems with great success.[1-8] However, the simple, straightforward approach to word recognition, shown in Fig. 1, runs into difficulties for complex vocabularies, i.e., vocabularies with phonetically similar words. For

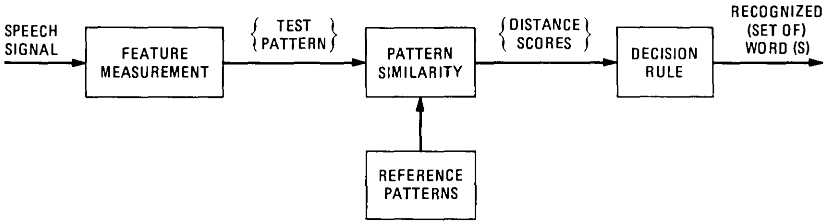


Fig. 1—Block diagram of standard approach to isolated word recognition.

example, recognition of the vocabulary consisting of the letters of the alphabet would have problems with letters in the sets

$$\begin{aligned}\phi_1 &= \{A, J, K\},\\ \phi_2 &= \{B, C, D, E, G, P, V, T, Z\},\\ \phi_3 &= \{Q, U\},\\ \phi_4 &= \{I, Y\},\\ \phi_5 &= \{L, M, N\},\\ \phi_6 &= \{F, S, X\}.\end{aligned}$$

Similarly, recognition of the computer terms of Gold[9] might lead to confusions among the set containing four, store, and core. In the above cases the problems are due to the inherent acoustic similarity (overlap) between sets of words in the vocabulary. It should be clear that this type of problem is essentially unrelated to vocabulary size (except when we approach very large vocabularies), since a large vocabulary may contain no similar words (e.g., the Japanese cities list of Itakura[4]), and a small vocabulary may contain many similar words (e.g., the letters of the alphabet).

The purpose of this paper is to propose, discuss, and evaluate a modified approach to isolated word recognition in which a two-pass method is used. The output of the first recognition pass is an ordered set of word classes in which the unknown spoken word is estimated to have occurred, and the output of the second pass is an ordered list of word candidates within each class obtained from the first pass. The computation for the first pass is similar in nature but often reduced in magnitude from that required for the standard one-pass word recognizer. The computation of the second pass consists of using an "optimally" determined word discriminator to separate words within the equivalence class. In Section II, we present the two-pass recognizer, and discuss its philosophy and method of implementation. In Section III, we give an evaluation of the effectiveness of the two-pass approach for a vocabulary consisting of the 26 letters of the alphabet, the 10 digits, and the command words STOP, ERROR, and REPEAT. Finally, in Section IV, we summarize the results and show how they are applicable to practical speech recognition systems.

II. THE TWO-PASS RECOGNIZER

Assume the word vocabulary consists of V words. The ith word, v_i, is represented by the word template $\mathbf{R}_i$, $i = 1, 2, \cdots, V$, where each $\mathbf{R}_i$ is a multidimensional feature vector. Similarly, we denote the test pattern as $\mathbf{T}$ (corresponding to the spoken word q in the vocabulary), where $\mathbf{T}$ is again a multidimensional feature vector. For simplicity we assume that the pattern similarity and distance computation is carried out using the "normalize and warp" procedure described by Myers et al.,[10] and illustrated in Fig. 2. A "standard" word duration of N frames is adopted, and each reference pattern is linearly warped to this duration. We call the warped reference patterns $\tilde{\mathbf{R}}_i$. Similarly, the test pattern is linearly warped to a duration of N frames, yielding the new pattern $\tilde{\mathbf{T}}$. A dynamic time-warping alignment algorithm then computes the "standard" distance

$$D(\tilde{\mathbf{T}}, \tilde{\mathbf{R}}_i) = \frac{1}{N} \sum_{k=1}^{N} d(\tilde{\mathbf{T}}(k), \tilde{\mathbf{R}}_i(w(k))), \tag{1}$$

where $d(\tilde{\mathbf{T}}(k), \tilde{\mathbf{R}}_i(l))$ is the local distance between frame k of the test pattern, and frame l of the ith reference pattern, and $w(k)$ is the time-alignment mapping between frame k of the test pattern, and frame $w(k)$ of the ith reference pattern. The total distance D of eq. (1) is only a function of i.

We define the local distance of the kth frame of the test pattern to the $w(k)$th frame of the ith reference pattern as $d_i(k)$, where

$$d_i(k) = d(\tilde{\mathbf{T}}(k), \tilde{\mathbf{R}}_i(w(k))), \tag{2}$$

so $D(\tilde{\mathbf{T}}, \tilde{\mathbf{R}}_i)$ of eq. (1) can be written as

$$D(\tilde{\mathbf{T}}, \tilde{\mathbf{R}}_i) = \frac{1}{N} \sum_{k=1}^{N} d_i(k). \tag{3}$$

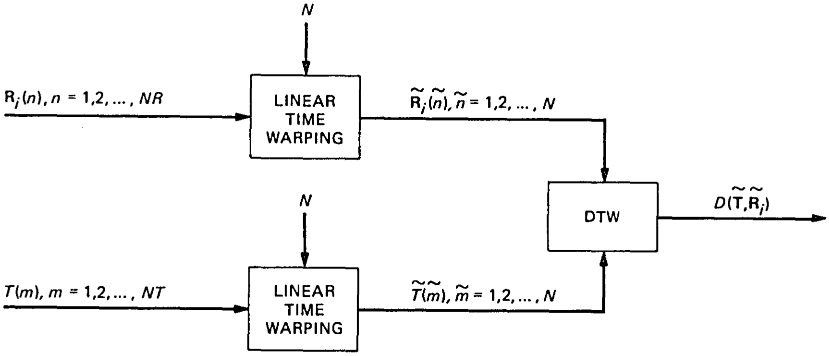

N
$R_i(n), n = 1,2,\ldots, NR$
LINEAR TIME WARPING
$\tilde{R}_i(\tilde{n}), \tilde{n} = 1,2,\ldots, N$
N
DTW
$D(\tilde{T},\tilde{R}_i)$
$T(m), m = 1,2,\ldots, NT$
LINEAR TIME WARPING
$\tilde{T}(\tilde{m}), \tilde{m} = 1,2,\ldots, N$

Fig. 2—Block diagram of the normalize-and-warp procedure for equalizing the lengths of words.

If $\tilde{\mathbf{R}}_i$ corresponds to the correct reference for the spoken word $\tilde{\mathbf{T}}$ (i.e., $i = q$), then we would theoretically expect the local distance $d_q(k)$ to be *independent* of k, with d assuming values from a χ^2 distribution with p (eight for the system we are using) degrees of freedom for the case where the speech features are those of an LPC model and the log likelihood distance measure is used for the local distance.[11,12] Thus, if we plotted $d_q(k)$ versus k, we would expect it to vary around some expected value $\hat{d}$ where

$$\hat{d} = E[d_q(k)] = E[\chi_p^2]. \tag{4}$$

An example of a typical curve of $d_q(k)$ versus k is given in Fig. 3a.

If we now examine the typical behavior of the curve of $d_i(k)$ versus k when $i \neq q$, we see that one of two types of behavior generally occurs. When word q is acoustically very different from word i, then $d_i(k)$ is generally large [compared to $\hat{d}$ of eq. (4)] for *all* values of k, and the overall distance score D of eq. (3) is large. This case is illustrated in Fig. 3b. However, when we have acoustically similar

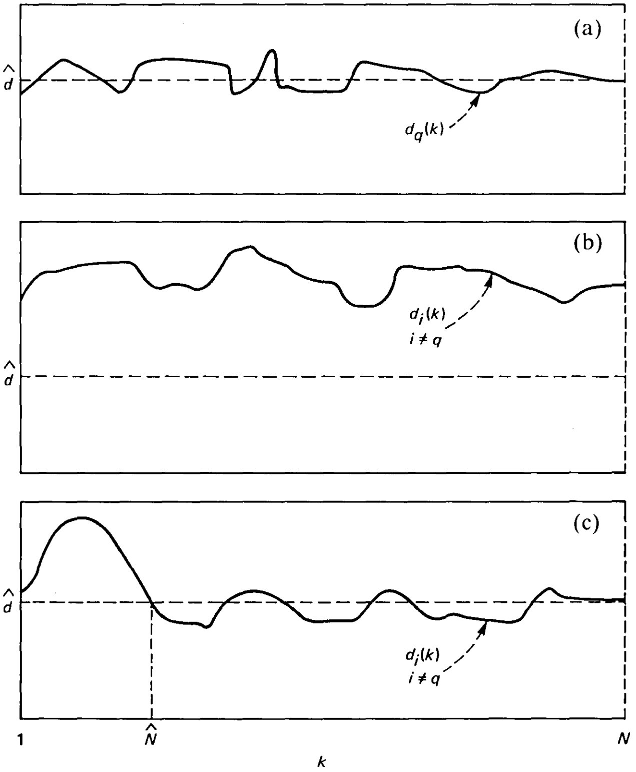


Fig. 3—Curves of $d_i(k)$ versus k for three cases.

words, then generally $d_i(k)$ will be approximately equal to $d_q(k)$ for all values of k in acoustically *identical* regions, and will be larger than $d_q(k)$ only in acoustically dissimilar regions. An example in which the dissimilar region occurs at the beginning of the word (the first $\hat{N}$ frames) is shown in Fig. 3c.

The key point to be noted from the above discussion is that when the vocabulary contains words that are acoustically similar, and one of these similar words is spoken (i.e., it is the test utterance), then the total distance scores for these similar words consists of a random component [because of the variations of $d(k)$ in the similar regions] and a deterministic difference (because of the differences in the dissimilar regions). In cases when the size of the dissimilar region is small (i.e., $\hat{N} \ll N$ in Fig. 3c), then the random component of the distance score can (and often does) outweigh the true difference component, causing a potential recognition error. For highly complex vocabularies (e.g., the letters of the alphabet), this situation occurs frequently.

One possible solution to the above problem would be to modify the overall distance computation so that more weight is given to some regions of the pattern than others. For example, we could consider a weighted overall distance of the form

$$D(\tilde{\mathbf{T}}, \tilde{\mathbf{R}}_i) = \frac{\sum_{k=1}^{N} W(k) d(\tilde{\mathbf{T}}(k), \tilde{\mathbf{R}}_i(w(k)))}{\sum_{k=1}^{N} W(k)}, \tag{5}$$

where $W(k)$ is an arbitrary frame weighting function, and the denominator of eq. (5) is used for distance normalization. The problem with eq. (5) is that a "good" weighting function is difficult to define since the "optimal" set of weights is clearly a function of the "actually" spoken word (q) and the reference pattern being used (i). Furthermore, any weighting that would help discriminate between acoustically similar words, would tend to hurt the discrimination between acoustically different words.

The above discussion suggests that a reasonable approach would be a two-pass recognition strategy in which the first pass would decide on an ordering of word "equivalence" classes (in which sets of acoustically similar words occurred), and the second pass would order the individual words within each equivalence class. For the first-pass recognition an unweighted (normal) distance would be used, and for the second pass a weighted distance would be used. In order to implement such a two-pass recognizer, a number of important questions must be answered, including:

(*i*) How do we "automatically" choose the word equivalence classes for each new vocabulary?

(*ii*) How do we determine class distance scores for the first recognition pass?

(*iii*) How do we determine weighting functions for the second recognition pass?

(*iv*) How do we generate weighted distance scores for the second recognition pass?

(*v*) How do we combine results from both recognition passes to give a final, overall set of distance scores and word orderings?

Some possible answers to each of these questions are given in the following sections.

2.1 Generation of word equivalence classes

Given the V vocabulary words $v_1, v_2, \cdots, v_V$, we would like to find a procedure for mapping words into acoustic equivalence classes ϕ_j, $j = 1, 2, \cdots, J$, where $J \leq V$. There are at least two reasonable approaches for solving this problem; one is a theoretical approach, the other an experimental one.

For the theoretical approach we can generate a "word-by-word" distance matrix D_w, on the basis of the phonetic transcriptions of the vocabulary entries. In order to do this we need to define a "phoneme" distance matrix, d_p, a distance cost for inserting a phoneme, d_I, and a distance cost for deleting a phoneme, d_D. The phoneme distance matrix could be a count of the number of distinctive features that have to be changed to convert from one phoneme to another.[13] A total word-by-word distance is then defined by a dynamic time-warp match between the words, with a vertical step representing an insertion, and a horizontal step representing deletion. Figure 4a illustrates this procedure for the words eight and J, and Figure 4b for the words one and nine. For the words eight and J, the optimum path is an insertion (of J), match between e^I and e^I, and a deletion of t, giving a distance

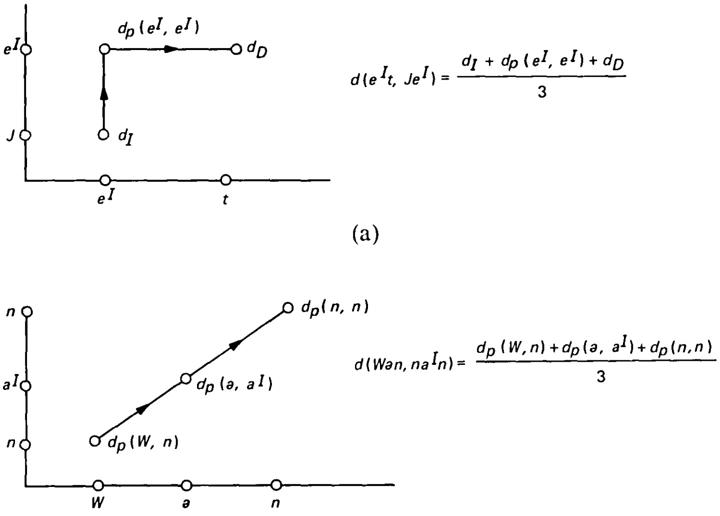
$$d(e^It, Je^I) = \frac{d_I + d_p(e^I, e^I) + d_D}{3}, \tag{6a}$$

whereas for one and nine, the optimum path is a straight line giving

$$d(w\,ə\,n, na^In) = \frac{d_p(w, n) + d_p(ə, a^I) + d_p(n, n)}{3}. \tag{6b}$$

It should be clear that once $d_p(p_1, p_2)$, d_I, and d_D are defined, the word-by-word distance scores can be generated.

A second approach to obtaining word-by-word distance scores is to use real tokens of the vocabulary words and do the actual dynamic time warping of the feature sets and obtain actual word distances. If several tokens have been recorded, averaging of distances increases the reliability of the final results.


$d_p(e^I, e^I)$
e^I
d_D
J
d_I
e^I
t
$d(e^I t, Je^I) = \frac{d_I + d_p(e^I, e^I) + d_D}{3}$
n
$d_p(n, n)$
a^I
$d_p(a, a^I)$
n
$d_p(W, n)$
W
a
n
$d(Wan, na^I n) = \frac{d_p(W,n) + d_p(a, a^I) + d_p(n,n)}{3}$

(a)

(b)

Fig. 4—Examples illustrating "word" alignment based on dynamic time warping.

From the word-by-word distance matrices, word equivalence classes may be obtained using the clustering procedures of Levinson et al.,[14] in which the vocabulary words are grouped into clusters (equivalence sets) based entirely on pairwise distance scores.

As an example of the use of the above techniques, consider the 39-word vocabulary consisting of the 26 letters of the alphabet, the 10 digits, and the 3 command words STOP, ERROR, and REPEAT. These 39 words become clustered into the sets

			Tokens
ϕ_1	=	{B, C, D, E, G, P, T, V, Z, 3, REPEAT},	11
ϕ_2	=	{A, J, K, 8, H},	5
ϕ_3	=	{F, S, X, 6},	4
ϕ_4	=	{I, Y, 5, 4},	4
ϕ_5	=	{Q, U, 2},	3
ϕ_6	=	{L, M, N},	3
ϕ_7	=	{O},	1
ϕ_8	=	{R},	1
ϕ_9	=	{W},	1
ϕ_{10}	=	{STOP},	1
ϕ_{11}	=	{ERROR},	1
ϕ_{12}	=	{0},	1
ϕ_{13}	=	{1},	1
ϕ_{14}	=	{7},	1
ϕ_{15}	=	{9}.	1

We discuss this vocabulary and the resulting equivalence sets a great deal more in Section III.

2.2 Determination of class-distance scores

Once all the vocabulary words have been assigned to one of the J classes, the first recognition pass estimates an ordering of the word classes in terms of class-distance scores. The class-distance scores can be determined in one of two ways. First they can be computed as the minimum of the word-distance scores, for all words in the class, i.e.,

$$\tilde{d}(\phi_j) = \min_{v_i \in \phi_j} D(\tilde{\mathbf{T}}, \tilde{\mathbf{R}}_i), \qquad j = 1, 2, \cdots, J. \tag{7}$$

This computation is similar to the one used by Aldefeld et al.[15] for directory listing retrieval.

An alternative method of obtaining class-distance scores would be to obtain "class-reference" templates (as well as word-reference templates) and to measure distance directly from the class-reference templates. Clearly with multiple templates per class, the K-nearest neighbor (KNN) rule can be used as effectively for class templates as for word templates.

The reason for considering class-reference templates for obtaining the class-distance scores is that the number of word classes is clearly smaller than the number of words. Hence, the number of distance calculations required to establish class-distance scores is generally much lower for class templates than for word templates. For example, for the 39-word vocabulary discussed previously, there are 15 word classes. Hence there is almost a 3 to 1 reduction from words to word classes. However, it should be clear that the danger in using class templates is that errors in determining class distances can be made from the reduced number of templates. This point will be discussed later in this paper.

2.3 Choice of weighting functions for the second pass of recognition

The output of the first recognition pass is an ordered set of word class-distance scores. For the second recognition pass, all words *within* the top class (or classes) are compared to the unknown test-word pattern ($\tilde{\mathbf{T}}$) using a weighted distance of the type discussed in eq. (5), and an ordering of words *within* the class is made. If several classes have similar class distance scores, the words within each of these classes are ordered in the same manner.

The key question that remains is how do we choose the weighting function, $W(k)$, of eq. (5) in an optimal or reasonable manner. The reader should recall, at this point, that the optimal weighting function, $W(k)$, is assumed to be a function of the pair of indices i (the reference

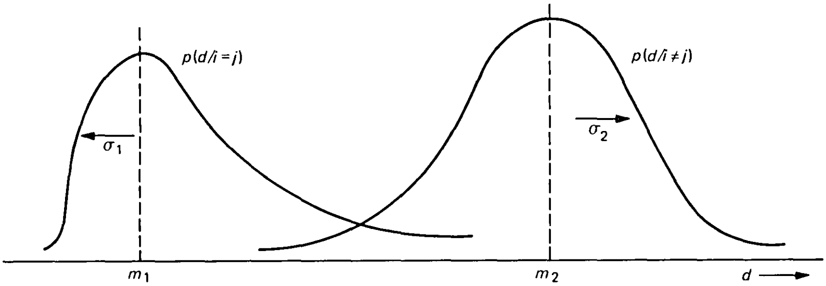


Fig. 5—Simple Gaussian model for frame distance distributions.

word) and j (the proposed test word). Hence if there are L words in an equivalence class, then there are $L(L-1)$ sets of weighting functions [the cases $i=j$ have $W(k) = 1$].

We have investigated two ways of determining $W(k)$ for the second recognition pass. Optimality theory says that to maximize the weighted distance of eq. (5),[16] the value of $W(k)$ should be

$$W(k) = 1 \qquad k = k_0, \tag{8a}$$

$$= 0 \qquad \text{all other } k, \tag{8b}$$

where k_0 is the index where the distance between $\tilde{\mathbf{R}}_i$ and $\tilde{\mathbf{T}}$ is, on average, the maximum. In this manner, the algorithm places all its reliance on the single frame where one would expect the maximum difference between reference and test patterns to occur. In practice, this weighting does not work since the variability in location of the frame $k = k_0$ of eq. (7) is large. Hence, on several trials the distances, using the weighting of eq. (7), can vary considerably.

A more effective manner of determining a good (but not optimal) set of weights is as follows. Consider the model for the distribution of distances for a single frame as shown in Fig. 5. The curve on the left in Fig. 5 is the assumed distribution of distances in the case when $i=j$ (i.e., the reference and test patterns are from the same word). In this case, we expect a χ^2 distribution with p (order of the LPC model) degrees of freedom for the frame distance. For convenience, we model this distribution as a Gaussian distribution with mean m_1 and standard deviation σ_1.*

For the case when $i \neq j$ (i.e., the reference and test patterns are from different words), we assume the frame distance has a Gaussian distri-

* This assumption is reasonable since the word distance, which is a sum of frame distances, has a Gaussian distribution (by the central limit theorem), and the actual probability of word error is directly related to the word distance.

bution (as shown to the right in Fig. 5) with mean m_2 and standard deviation σ_2.

We now make a simple recognition model that says the probability of recognition error for the word is proportional to the probability of error for single frames (since the word distance is the sum of frame distances). Then, based on the model of Fig. 5 with assumed Gaussian statistics, the probability of correct classification (i.e., finding a smaller frame distance for the spoken word, than for any other word) for a single frame is

$$P(C) = \int_{-\infty}^{\infty} P[p(d/_{i=j}) = \lambda] \cdot P[p(d/_{i\neq j}) > \lambda] \, d\lambda, \tag{9}$$

where $P[x]$ is the probability of the event x occurring. Equation (9) says that the probability of correct frame classification is the integral of the probability that for the correct word ($i = j$) we get a frame distance λ, and for the *closest* incorrect word ($i \neq j$) we get a frame distance greater than λ. Thus the probability of a frame error is

$$P(E) = 1 - P(C), \tag{10}$$

which becomes

$$P(E) = 1 - \int_{-\infty}^{\infty} N[\lambda - m_1, \sigma_1] \int_{\lambda}^{\infty} N[\eta - m_2, \sigma_2] \, d\eta \, d\lambda, \tag{11}$$

which can be put into the form

$$P(E) = \int_{-\infty}^{(m_2-m_1)/(\sigma_1^2+\sigma_2^2)^{1/2}} \frac{\exp(-x^2/2)}{\sqrt{2\pi}} dx = Erf\left(\frac{m_2 - m_1}{\sqrt{\sigma_1^2 + \sigma_2^2}}\right). \tag{12}$$

The form of eq. (12) can be verified for the simple cases $m_2 = m_1$, where $P(E) = 0.5$, and $m_2 \gg m_1$, where $P(E) \rightarrow 0$.

The above discussion suggests that a reasonable choice for frame weighting would be

$$W^{j,i}(k) = \frac{|\langle \hat{d}_{ii}(k) \rangle - \langle \hat{d}_{ji}(k) \rangle|}{(\sigma^2_{d_{ii}(k)} + \sigma^2_{d_{ji}(k)})^{1/2}}, \tag{13}$$

where $\hat{d}_{ii}(k)$ is the local distance between repetitions of word i for frame k, and $\hat{d}_{ji}(k)$ is the local distance between spoken words j and i for frame k, and where the expectations are performed statistically over a large number of occurrences of the words v_i and v_j.

By way of example, Fig. 6 shows examples of plots of $\langle d_{ji}(k) \rangle$ versus

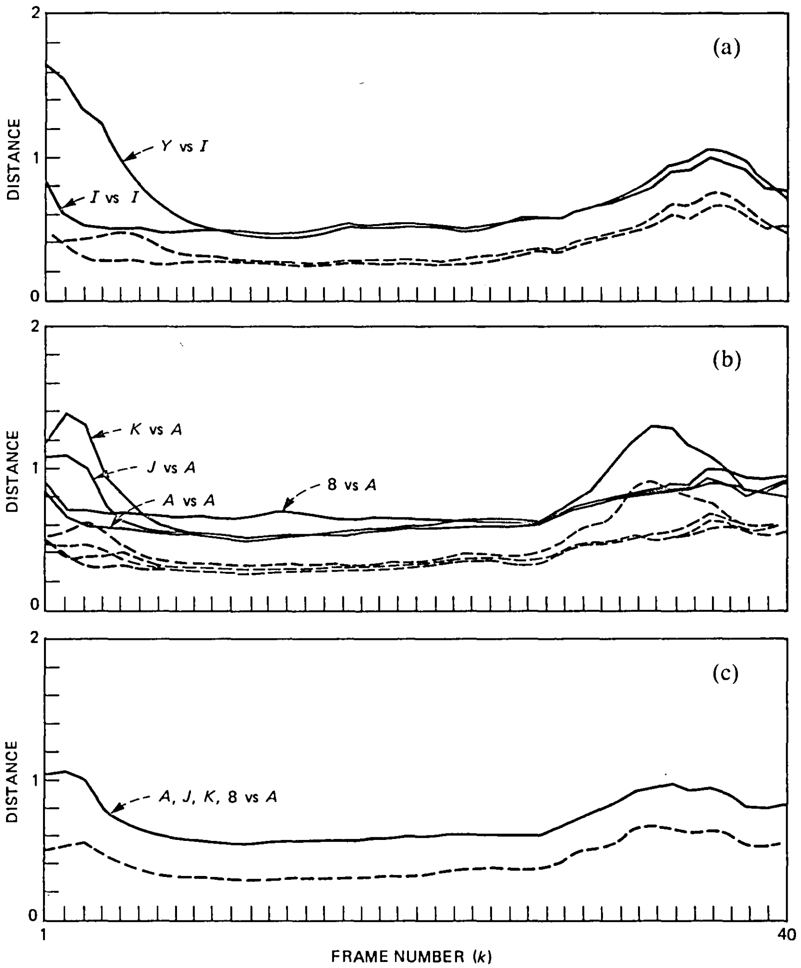


Fig. 6—Examples of frame-by-frame distances for words within word equivalence classes.

k and $W(k)$ versus k for some typical cases.* Figure 6 shows a series of plots for the following cases:

(i) (Fig. 6a) Curves of $\langle \hat{d}_{ji}(k) \rangle$ and $\sigma_{\hat{d}_{ji}(k)}$ for the case where word i was the letter I, and word j was the letter Y. We can see that $\langle \hat{d}_{ii}(k) \rangle$ (the solid curves) is approximately constant whereas $\langle \hat{d}_{ji}(k) \rangle$ differs from $\langle \hat{d}_{ii}(k) \rangle$ only at the beginning of the word (i.e., the first eight

* The data of Fig. 6 were obtained from about 10,000 comparisons for each word, i.e., a large data base was used.

frames). We also see that the curves of $\sigma_{\hat{d}_{ji}(k)}$ (the dashed curves) are comparable for the cases $j = i$ and for $j \neq i$, with only small differences occurring in the first eight frames.

(*ii*) (Fig. 6b) Curves of $\langle \hat{d}_{ji}(k) \rangle$ and $\sigma^2_{\hat{d}_{ji}(k)}$ for the case where word i was the letter A, and where j corresponded to the letters J and K for word 8. Similar behavior to that of Fig. 6a is seen, in that $\langle \hat{d}_{ii}(k) \rangle$ is approximately constant, and $\langle \hat{d}_{ji}(k) \rangle$ is larger than $\langle \hat{d}_{ii}(k) \rangle$ at the beginning of the word, for words J and K, and at the end of the word, for word 8. For the word 8, the curve of $\sigma_{\hat{d}_{ji}(k)}$ is also fairly large at the end of the word, indicating the high degree of variability in the plosive release of the word 8.

(*iii*) (Fig. 6c) The part shows the results of averaging the data of Fig. 6b over all $j \neq i$ with j in the class of word i, i.e., class-weighting templates. In this case the curve of $\langle \hat{d}_{ji}(k) \rangle$ shows flat behavior except at the beginning (due to J, K) and end (due to 8). If storage of word-weighting curves is burdensome, the use of class-weighting curves could be considered as a viable alternative.

Figure 7 shows a set of two weighting curves $W^{j,i}(k)$ for the words I and Y. Figure 7a shows the weighting curve for reference word I and test word Y, and Fig. 7b shows the weighting curve for reference word Y and test word I. Several interesting properties of the curves should be noted. First we see that $W^{j,i}(k)$ generally consists of a large pulse (for these examples this occurs near $k = 1$) and a residual tail. The tail is a measure of the statistical noise level, i.e., the statistical difference between $\langle \hat{d}_{ji}(k) \rangle$ and $\langle \hat{d}_{ii}(k) \rangle$ in the region of acoustical similarity. Typically the peak amplitude in the tails is less than 10 percent of the peak amplitude in the main pulse.

Another interesting property of the weighting curves is that there is no symmetry, in that

$$W^{i,j}(k) \neq W^{j,i}(k). \tag{14}$$

An explanation of this behavior is given in Fig. 8, which shows two plots of dynamic time-warping paths for the words I and Y, where it is assumed that the word Y is simply the word I with a prefix phoneme $/w/$. Figure 8a shows that when I is warped to Y, there is a discrepancy region in which the $/w/$ is being warped to the initial region of the $/a^I/$ and large distances result. The $/a^I/$ is warped to itself (the "ideal" path) and no further distance is accumulated. Figure 8b shows that the discrepancy region is considerably smaller when Y is mapped to I. The resulting weighting curves agree in form with the results given in Fig. 7.

2.4 Generation of distance scores for the second recognition pass

We have now shown how to assign words to classes, how to get class

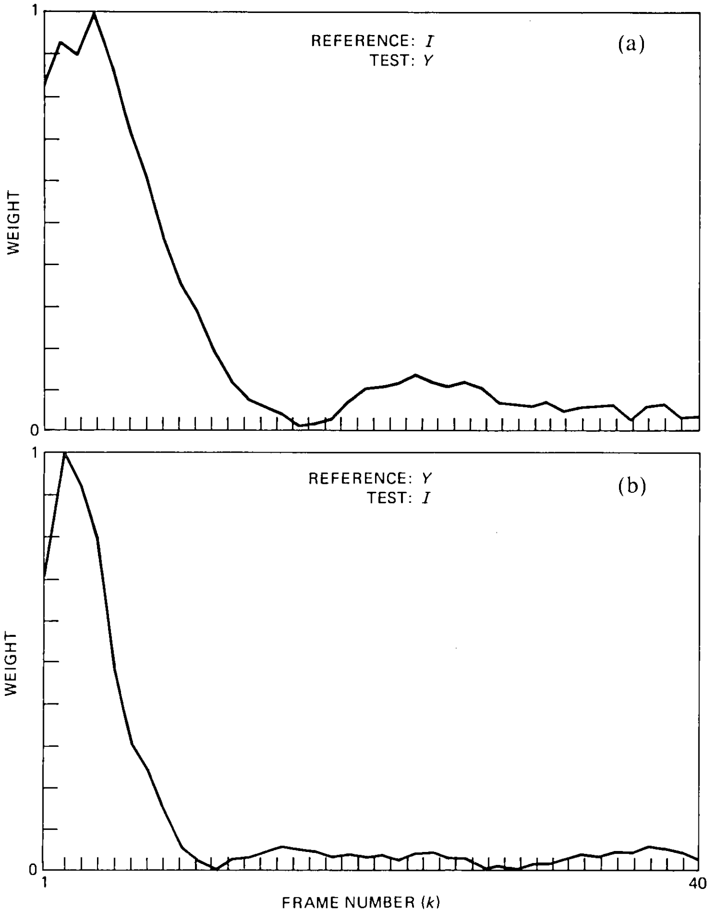


Fig. 7—Weighting curves for comparing the words /*I*/ and /*Y*/.

distance scores for the first recognition pass, and how to assign weights for pairs of words within a word class. The next step in the procedure is the determination of the distance for the second recognition pass based on the pairwise weighted distance scores.

To see how this is accomplished, we define a pairwise weighted distance $D_{j,i}$ as

$$D_{j,i} = \frac{\sum_{k=1}^{N} W^{j,i}(k) d_i(k)}{\sum_{k=1}^{N} W^{j,i}(k)}, \tag{15}$$

where i is the index of the reference pattern (i.e., one of the words in

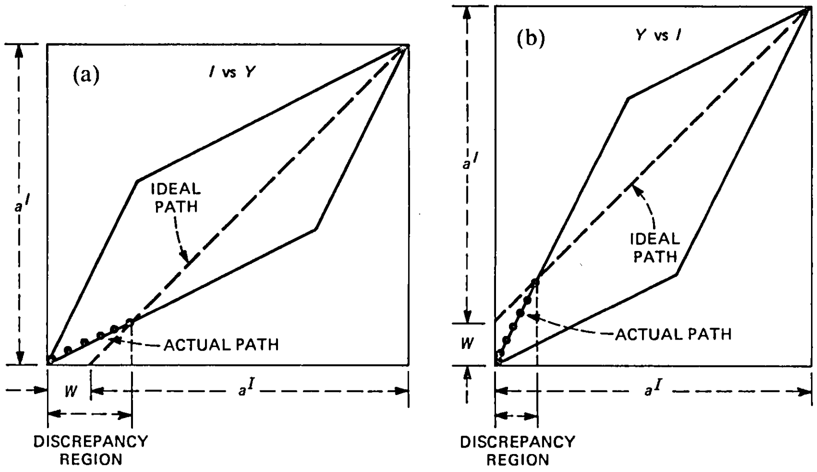


Fig. 8—An example showing why the word weighting curves are not symmetrical.

the equivalence class) and j is the (assumed) index of the test pattern (again one of the words in the equivalence class).

The quantity $D_{j,i}$ of eq. (15) is computed for all i, j pairs (with $i \neq j$) in the word class with minimum class distance, and a matrix of pairwise distances, **D** is obtained. The word distance, D_i, can be obtained in one of two ways, namely:

(i) Averaging over the j index, giving

$$D_i = \sum_{\substack{j \\ j \neq i}} D_{j,i}. \qquad (16a)$$

(ii) Finding the minimum over the j index, i.e.,

$$D_i = \min_{\substack{j \\ j \neq i}} \{\mathrm{D}_{j,i}\}. \qquad (16b)$$

The advantage of averaging is that D_i tends to be more reliable, since averaging is equivalent to adding weighted distances over a larger number of frames than would be used for a single comparison. The minimum computation is useful, especially when several of the $D_{j,i}$ are about the same. We examine both these scoring methods in Section III.

For the case of averaging pairwise distance scores [eq. (16a)], the computation can be carried out more efficiently as follows. By combining eqs. (15) and (16a) we get

$$D_i = \sum_j D_{j,i} = \sum_j \left(\frac{\sum\limits_{k=1}^{N} W^{j,i}(k)\, d_i(k)}{\sum\limits_{k=1}^{N} W^{j,i}(k)} \right) \tag{17a}$$

$$= \sum_j \sum_{k=1}^{N} \left(\frac{W^{j,i}(k)\, d_i(k)}{\sum\limits_{k=1}^{N} W^{j,i}(k)} \right) \tag{17b}$$

$$= \sum_{k=1}^{N} \sum_j \left(\frac{W^{j,i}(k)}{\sum\limits_{k=1}^{N} W^{j,i}(k)} \right) d_i(k) \tag{17c}$$

$$= \sum_{k=1}^{N} \hat{W}^i(k)\, d_i(k), \tag{17d}$$

where

$$\hat{W}^i(k) = \sum_j \frac{W^{j,i}(k)}{\sum\limits_{k=1}^{N} W^{j,i}(k)}. \tag{18}$$

Thus, for L words in the equivalence class, we can compute D_i with N multiplications and additions [rather than the $N(L-1)$ computations of eq. (16a)], and only L vectors of N averaged weights $[\hat{W}^i(k)]$ need be stored, rather than $L(L-1)$ vectors as implied by eq. (15).

Another variation on the distance weighting that was studied here was the effect of applying a nonlinearity to the weighting function, $W^{j,i}$, before computing $D_{j,i}$. The nonlinearity was to replace $W^{j,i}(k)$ by $\tilde{W}^{j,i}(k)$, defined as

$$\tilde{W}^{j,i}(k) = \begin{cases} W^{j,i}(k) & \text{if } W^{j,i}(k)/W_{\mathrm{MAX}} > T, \\ 0 & \text{otherwise,} \end{cases} \tag{19}$$

where

$$W_{\mathrm{MAX}} = \max_k \left[W^{j,i}(k) \right], \tag{20}$$

and T is a threshold which is specified in the algorithm. The nonlinearity of eq. (19) truncates (to 0) the weighting curve whenever its relative amplitude falls below the threshold. Figure 9 illustrates a typical curve $W^{j,i}(k)$ and its truncated version $\tilde{W}^{j,i}(k)$. The new weighting function was then applied directly in eq. (15) in place of $W^{j,i}(k)$. Clearly, when $T = 0$, $W^{j,i}(k)$ and $\tilde{W}^{j,i}(k)$ are identical. Again, when averaging is used, the computation of eq. (17) gives a reduced set of weights.

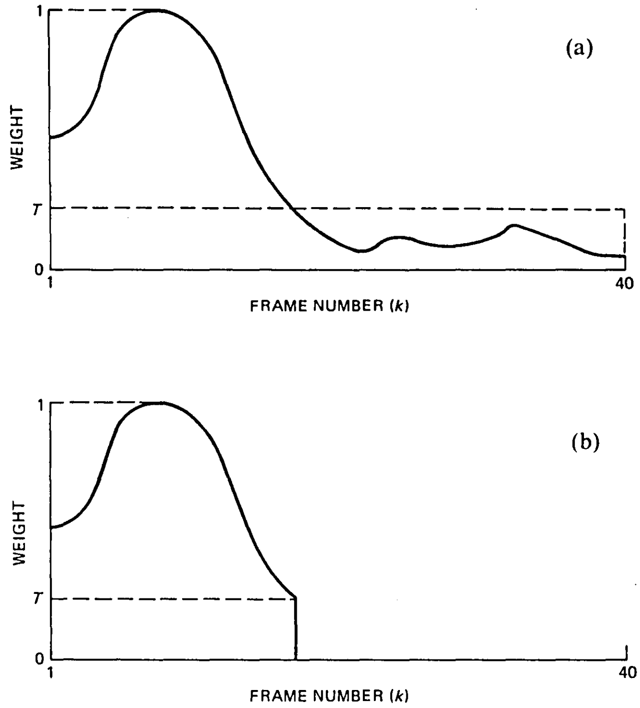


Fig. 9—An example of a weighting curve and its truncated version.

2.5 Overall distance computation

If we can make the assumption that the probability of a class error on the first recognition pass is significantly smaller than the probability of a word error on the first pass, then the final distance for each word of the minimum class is the distance obtained on the second recognition pass. However there are applications in which it is desirable to have a distance score for *every* word in the vocabulary. Hence, in these cases, it is necessary to combine the ordering from the second pass, with the distances from the first pass. The basis for such a strategy is that distances on the first pass are statistically more reliable than distances on the second pass, whereas order statistics (within the class) are more reliable on the second pass than on the first pass. One very simple way of combining distances and word orders is to obtain second-pass ordering for every word in the vocabulary (i.e., apply the method of Section 2.4 to all word classes), and then reorder the word list using distances from the first pass, and ordering within the class from the second pass.

2.6 An example of the use of the two-pass system

To illustrate this entire procedure, Tables I to III show an example

Table I—Recognition results for a simple example (first pass)

Word Index	Word Class	Word Distance First Pass	Word Position First Pass	Class Number	Class Distance First Pass
1	1	0.47	4	1	0.47
2	3	0.39	2	2	0.66
3	3	0.51	5	3	0.37
4	2	0.72	10		
5	3	0.42	3		
6	1	0.60	6		
7	1	0.67	9		
8	2	0.83	12		
9	3	0.37	1		
10	2	0.78	11		
11	2	0.66	8		
12	1	0.62	7		

of the recognition steps for a 12-word vocabulary with three word equivalence classes. Table I shows the results of the first recognition pass. The class distance scores are assigned as the minimum word distance for words within the class. The "best" class in the first pass is class 3 with a distance score of 0.37, with class 2 having a somewhat higher distance of 0.47. In the second recognition pass the words within the best class (or classes) are compared using the optimally determined

Table II—Second recognition pass results for the example in Table I

			j					
		1	6	7	12			
	1	X	0.43	0.52	0.47	0.47	1	
i	6	0.57	X	0.62	0.62	0.60	3	Class 1
	7	0.72	0.75	X	0.60	0.69	4	
	12	0.60	0.57	0.63	X	0.60	2	
			$D_{j,i}$			D_i(avg)	Order	

			j					
		4	8	10	11			
	4	X	0.87	0.82	0.85	0.85	3	
i	8	0.80	X	0.84	0.86	0.83	2	Class 2
	10	0.92	0.77	X	0.91	0.87	4	
	11	0.78	0.80	0.80	X	0.79	1	
			$D_{j,i}$			D_i(avg)	Order	

			j					
		2	3	5	9			
	2	X	0.33	0.25	0.28	0.29	1	
i	3	0.47	X	0.67	0.50	0.55	4	Class 3
	5	0.45	0.56	X	0.57	0.53	3	
	9	0.27	0.37	0.30	X	0.31	2	
			$D_{j,i}$			D_i(avg)	Order	

weighting functions. The results for each of the three classes are shown in Table II. In practice, one would usually need to compute the $D_{j,i}$ scores only for the best one or two classes. However, for explanatory purposes, results are shown for all three classes. Also, as discussed above, in the case of distance averaging, the $D_{j,i}$ scores need not be computed since the D_i scores can be obtained directly via eqs. (17) and (18). Using the technique of averaging leads to the within-class distances and orderings as shown in the table. Finally, Table III shows the results of reordering the words using the distances obtained from pass 1, and the within-class orderings obtained from pass 2. Thus word 2 is the best recognition candidate (with a distance of 0.37), whereas word 9 was the best recognition candidate at the end of the first pass. Other, within-class reshufflings of word position occur as a result of the two recognition passes as shown in Table I.

2.7 Summary of the two-pass recognizer

Figure 10 shows a block diagram of the full two-pass isolated word recognition system. In the first pass a DTW distance is computed between the unknown test word and the reference templates for each word class. The outputs of the first pass are ordered sets of word distance scores and class distance scores.

For the second pass a set of pairwise weighted distances is determined for all words within each word class with suitably low scores on the first recognition pass. The final recognition output is a combination of distance scores from the first pass and word orderings from the second pass. In the next section we demonstrate how this procedure works in some practical recognition examples.

Table III—Overall word positions and distances for the example given in Tables I and II

Word Index	Word Position	Word Distance
1	4	0.47
2	1	0.37
3	5	0.51
4	11	0.78
5	3	0.42
6	7	0.62
7	9	0.67
8	10	0.72
9	2	0.39
10	12	0.83
11	8	0.66
12	6	0.60

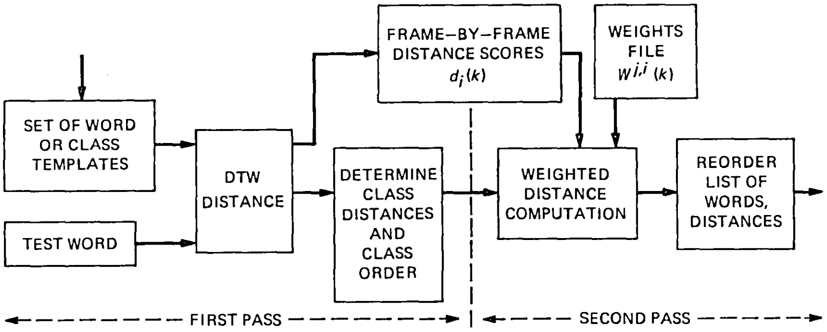

FRAME–BY–FRAME DISTANCE SCORES $d_j(k)$
WEIGHTS FILE $w^{i,j}(k)$
SET OF WORD OR CLASS TEMPLATES
DTW DISTANCE
DETERMINE CLASS DISTANCES AND CLASS ORDER
WEIGHTED DISTANCE COMPUTATION
REORDER LIST OF WORDS, DISTANCES
TEST WORD
FIRST PASS
SECOND PASS

Fig. 10—Block diagram of the overall two-pass recognizer.

III. EVALUATION OF THE TWO-PASS RECOGNIZER

To test the ideas behind the two-pass recognizer, we used a data base of existing recordings. The word vocabulary consisted of the $V = 39$ word vocabulary of the letters of the alphabet, the digits (0 to 9), and the three command words STOP, ERROR, and REPEAT. The training data for obtaining word and class reference templates, and pairwise word weighting curves, consisted of one replication of each word by each of 100 talkers (50 men, 50 women).* The word reference templates (12 per word) were obtained from a clustering analysis of the training data.[14,6] A set of "class" reference templates (12 per class) was obtained from a second clustering analysis in which the words within a class were combined prior to the clustering. The pairwise word weighting curves were obtained by cross-comparing all word tokens within a word class, averaging the time-aligned distance curves, and computing both the averages and standard deviations for each frame.

To test the performance of the overall system, two test sets of data were used. These included:

1. TS1—10 talkers (not used in the training) spoke the vocabulary one time over a dialed-up telephone line.
2. TS2—10 talkers (included in the training) spoke the vocabulary one time over a dialed-up telephone line.

Two sets of performance statistics were measured. For the first recognition pass the ability of the recognizer to determine the correct word class was measured. For the second recognition pass the improvement in word recognition accuracy (over the standard one-pass approach) was measured. The results obtained are presented in the next two sections.

* All results presented here are for speaker independent systems.

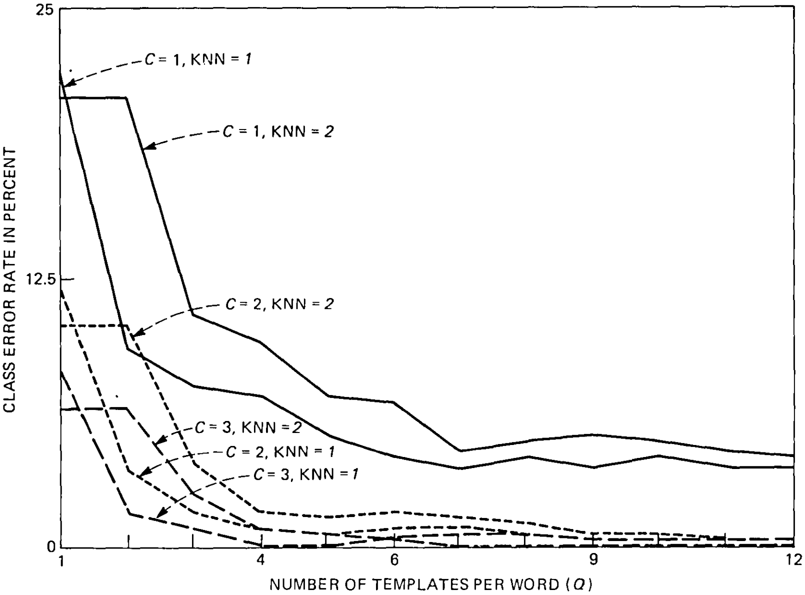


Fig. 11—Plots of class accuracy as a function of the number of templates per word (Q), class position (C), and KNN rule (KNN) for a 15-class vocabulary.

3.1 Class recognition accuracy for the first pass

The ability of the recognizer to determine the "correct" word class of the spoken word was measured using both word templates (and obtaining class-distance scores from the word-distance scores as discussed previously), and class templates (obtaining class-distance scores directly). The number of templates per word (or per class) varied from 1 to 12 in the tests to see the effects of the number of reference templates on the class accuracy. The K-nearest neighbor (KNN) rule was used to measure class scores with values of KNN = 1 (minimum distance), KNN = 2 (average of two best scores), and KNN = Q (average of Q best scores), where Q was the total number of templates used per word (or per class).

The results of the class recognition accuracy tests are given in Figs. 11 and 12.* Figures 11 and 12 show plots of class error rate (based on the top C classes) as a function of the number of templates per word (Fig. 11) or templates per class (Fig. 12), for values of KNN = 1 and 2, and for C = 1 (top candidate), C = 2 (two best classes), and C = 3 (three best classes). Figure 11 shows results when each class is represented by word templates, and Fig. 12 shows results when each class is represented by class templates.

* The reader should note the difference in vertical scales between Figs. 11 and 12.

Several interesting observations can be made from Figs. 11 and 12. These include:

(*i*) The KNN = 1 rule performs consistently better than the KNN = 2 rule for class discrimination, for *all values* of C and Q. This result is in contradiction with the results of Rabiner et al.[6] who found significantly better performance for KNN = 2 than for KNN = 1. The explanation of this behavior is that the KNN = 2 rule provides significantly improved, within-class discrimination (at the expense of slightly worse between class discrimination), and that when the only function is to determine the class, the KNN = 1 rule is superior. In fact when the KNN rule was used with a value of KNN = Q (i.e., averaging over all Q reference templates), the class accuracy on the first candidate decreased by about 20 percent—a highly significant loss of accuracy. This result again demonstrates that the minimum distance rule (KNN = 1) is best for *class* discrimination.

(*ii*) The use of word-reference templates provides significantly better performance than obtained from class-reference templates. For example, the class error rate for the top three classes (C = 3) with Q = 4 templates per word is essentially 0; whereas the class error rate for the top three classes with four templates per class is about 4 percent. This result shows clearly the importance of representing each word in

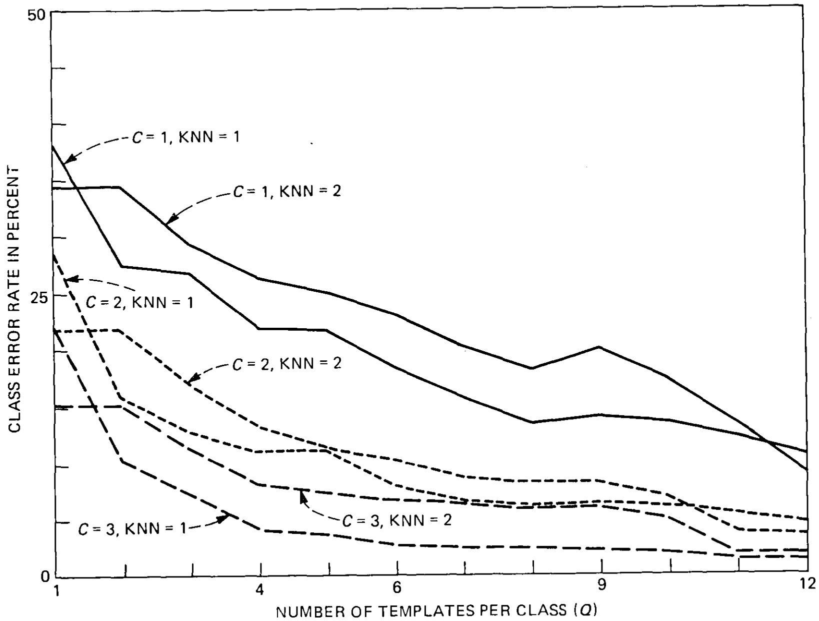


Fig. 12—Plots of class accuracy as a function of the number of templates per class (Q), class position (C), and KNN rule (KNN) for a 15-class vocabulary.

the equivalence class by an adequate number of word-reference templates.

(*iii*) With six templates per word, error rates of about 4 percent ($C = 1$), 1 percent ($C = 2$), and 0 percent ($C = 3$) are obtainable, indicating that the full contingent of 12 templates per word is unnecessary for proper class determination. Using 6, rather than 12 templates per word reduces the computation in the first recognition pass by 50 percent. If we *always* use two or more word classes, the required number of templates per word for the first pass can be reduced to four, with no serious loss in class accuracy.

The results shown in Fig. 11 indicate that high accuracy can readily be achieved in determining the correct equivalence class for each word in a very complex vocabulary. Hence there would appear to be no problems in implementing the first pass of the recognition system.

3.2 Within-class word discrimination for the second pass and overall performance scores

The two-pass word recognizer was tested on the words of TS1 and TS2. For each test set a total of 390 words were used (39 words × 10 talkers). For TS1, the word recognition accuracy (for the best candidate) on the first pass was 78 percent, and for TS2 (with talkers from the training set) the word recognition accuracy on the first pass was 85 percent. At the output of the second pass, the word recognition accuracy for the best candidate [using the averaging technique of eq. (16a) and assuming the correct word equivalence class was found] was 84.6 percent for TS1 and 88.5 percent for TS2, representing potential improvements of 6.6 percent and 3.5 percent, respectively. The reason that a larger improvement in accuracy was obtained for TS1 data than for TS2 data was that the accuracy on the first pass was lower for TS1 than for TS2 (where the talkers were in the training set) and hence there was more room for improvement within the word classes.

Figures 13 and 14 show plots of the changes in accuracy that are obtained for TS1 (Fig. 13) and TS2 (Fig. 14) data when a threshold is imposed on the distance scores at the output of the first recognition pass. The threshold specifies that the second recognition pass is skipped if the distance of the second word candidate is more than the threshold greater than the distance of the first word candidate. Clearly this procedure is a strictly computational one, since low-distance scores for a single word on the first pass are highly reliable indicators that no second pass is necessary. The data plotted in Figs. 13 and 14 show the percentage of cases where the actual spoken word comes in a lower position on the second pass than in the first pass within the word class; it also shows the percentage of cases when the spoken word comes in a higher position on the second pass than the first pass, and the

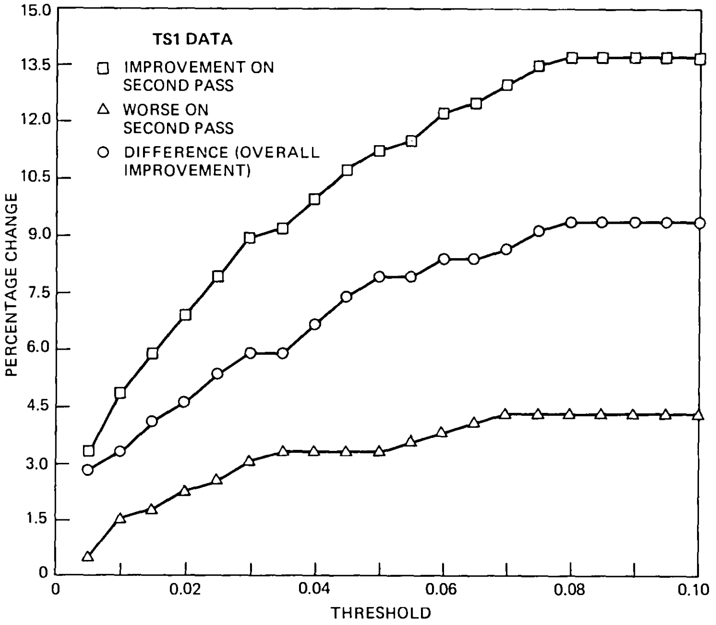


Fig. 13—Percentage improvement, decrease, and the resulting difference in word position at the output of the second recognition pass for TS1 data as a function of the distance threshold using the averaging method.

difference (the improvement) between the two curves. All the results are plotted as a function of the distance threshold for performing the second-pass computation. It can be seen from these figures that the two-pass recognizer is not ideal, i.e., there is a significant fraction of words for which a worse position results at the output of the second pass. However, on balance, it is seen that a real improvement in recognition accuracy results, and it is this improvement that makes the procedure a viable one.

A similar set of results obtained using the minimum computation of eq. (16b) on the second pass rather than the average computation of eq. (16a) are shown in Figs. 15 and 16 for TS1 and TS2, respectively. These plots show the same information as those of Figs. 13 and 14 for the averaging procedure. A comparison of these results shows that the averaging computation performs as well as, or better than, the minimum computation for the whole range of distance thresholds, and for both data sets. These results indicate that the averaging method provides a small but important statistical stability to the computation.

3.3 The effect of thresholding on the weighting curves

We ran a series of tests with the data from TS1 and TS2 to investigate the effects of applying thresholds to the weighting curves as illustrated

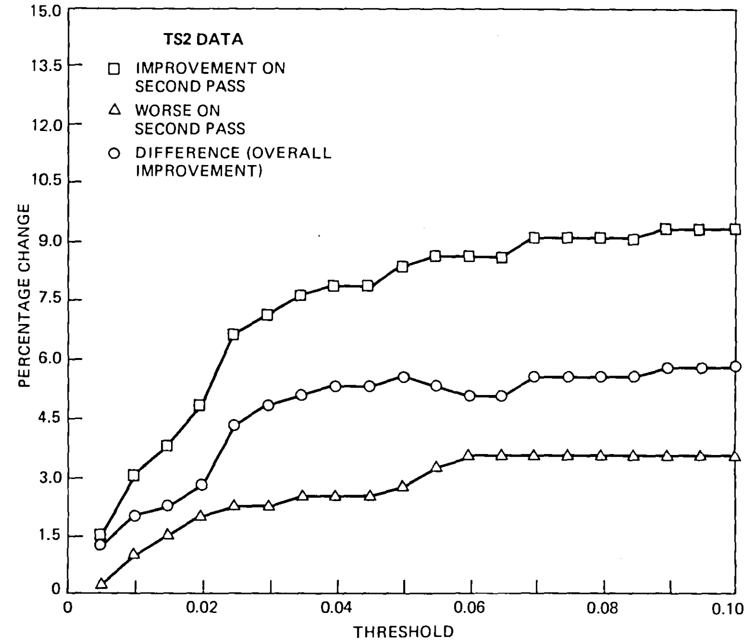


Fig. 14—Percentage improvement, decrease, and the resulting difference in word position at the output of the second recognition pass for TS2 data as a function of the distance threshold using the averaging method.

in Fig. 9. The results indicated that poorer performance *always* resulted when any significant part of the weighting curve was zeroed out. Thus the gain achieved by removing the "statistical" low-level parts of the weighting curve was canceled by the "deterministic" loss from the rest of the weighting curve. Hence the conclusion was to use the entire weighting curve as derived from the statistical model.

3.4 Computation for the two-pass recognizer

We have seen in Section 3.3 that word recognition accuracy improvements of from 3.5 to 6.6 percent result for the 39-word vocabulary using the two-pass recognizer. A key question that must be answered is what is the cost of the computation for the two-pass system.

To answer this question we must examine the computation in each pass of the recognizer. In the first recognition pass, for a V-word vocabulary with Q templates per word, a total of QV DTW comparisons are made. For a value of $N = 40$, each DTW comparison requires about 500 nine-point dot-product computations, so a total rate, R_1, of

$$R_1 = Q \cdot V \cdot 500 \cdot 9 \tag{21}$$

multiplications and additions are required.

If we assume that the local distances $d_{j,i}(k)$ associated with the optimum warping paths are saved for *each* reference template, then for *each* pairwise comparison of the second pass a total of N (typically 40) multiplications and additions are required. For L words in the equivalence class, a total of

$$R_2 = L \cdot (L - 1) \cdot N \tag{22}$$

multiplications and additions are required for the second-pass computation for a single equivalence class. For the averaging procedure of eq. (17), R_2 is reduced to LN multiplications and additions.

If we assume typical values of $V = 39$, $Q = 12$, $L = 7$, $N = 40$, we get $R_1 = 2{,}106{,}000$ and $R_2 = 1680$, i.e., the computation of the second pass is insignificant compared to the first pass computation. Furthermore since we can use reduced values of Q for the first pass (i.e., $Q = 6$ or $Q = 4$) the overall computation can be significantly reduced from the standard isolated word recognizer, with the same improvement in accuracy!

IV. DISCUSSION

The results presented in the preceding section show that improved recognition accuracy can be obtained via a two-pass recognition algorithm. It was shown that the improvements were both global, i.e., in

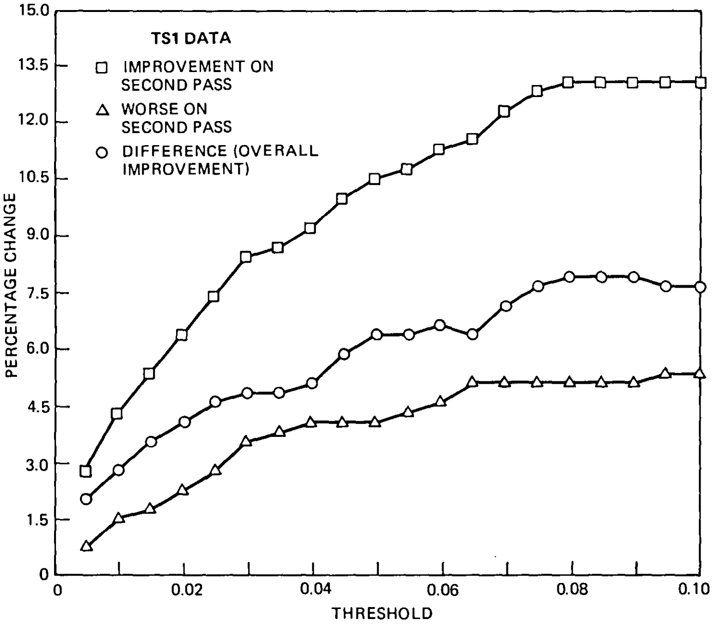


Fig. 15—The same results as in Fig. 13 obtained using the minimum method.

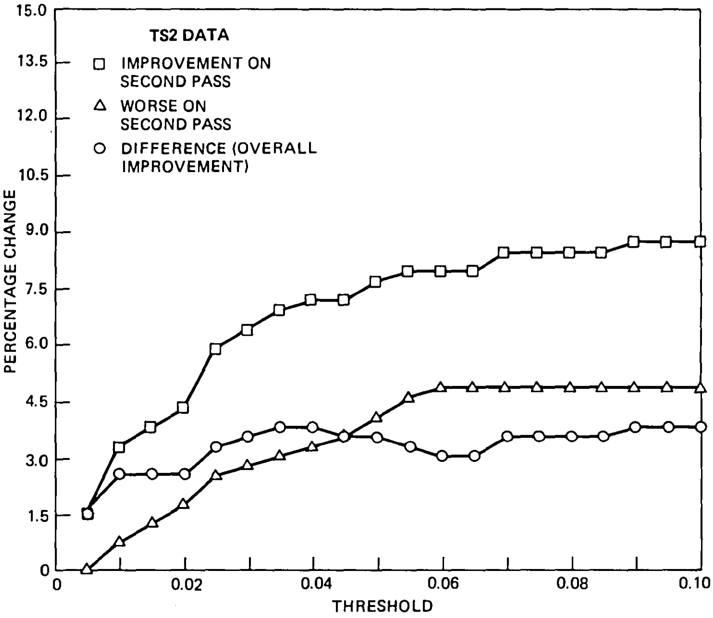


Fig. 16—The same results as in Fig. 14 obtained using the minimum method.

an absolute recognition sense, and local, i.e., within the classes of equivalent words. Although the proposed two-pass recognizer has a number of possible implementations, it was shown that the best choices were to use a reduced set of word templates on the first pass, and to use all word classes that had reasonably small distance scores on the second pass.

One of the major issues that remains unresolved in the two-pass recognizer is the choice of weighting curve used in the second-pass distance computation. The assumed Gaussian model which led to the variance-weighted difference of means for the weights is, at best, an approximation to the actual situation. Experimentation with modified forms of the weighting curve of eq. (13) led to poorer recognition performance. Thus, because we lacked a viable alternative, the weighting curve of eq. (13) is the only one we investigated for use in the two-pass recognizer.

An interesting question that arises as a result of this study is how could this two-pass recognizer aid in practical recognition tasks. As one would anticipate, the answer to this question is that it depends on the specific recognition task. For example, for the backtracking directory listing retrieval system of Rosenberg and Schmidt,[17] the improvement in recognition accuracy could provide significant reductions in search time. However, for the search procedure of Aldefeld et al.,[15] the increased word accuracy would have no effect on the search time, but could increase the name accuracy, especially when similar names exist in the directory (e.g., T. Smith and P. Smith). For applications like the airlines reservation system of Levinson and Rosenberg,[18] the increased word accuracy would reduce the load on the syntax analyzer; however, it needn't necessarily increase the overall accuracy of the system.

The above examples show that the two-pass recognition strategy can be useful for some applications, but one must examine carefully the specific task before claiming how useful it will potentially be.

V. SUMMARY

We have shown that a two-pass approach to isolated word recognition is viable when the word vocabulary consists of sets of acoustically similar words. The first recognition pass attempts to determine accurately the class within which the spoken word occurs, and the second recognition pass attempts to order the words within the class, based on weighted distances of pairwise comparisons of all words within the class. We discussed several alternatives for implementing this two-pass recognizer, and we made a performance evaluation which showed that a reliable class decision could be made based on a reduced set of template scores, and an improved word decision could be made from weighted pairwise distance scores.

REFERENCES

1. T. B. Martin, "Practical Applications of Voice Input to Machine," Proc. IEEE, *64* (April 1976), pp. 487–501.
2. S. Moshier, "Talker Independent Speech Recognition in Commercial Environments," Speech Communication Papers at the 97th ASA Meeting, June 1979, pp. 551–3.
3. H. Sakoe, "Two-Level DP Matching—A Dynamic Programming Based Pattern Matching Algorithm for Connected Word Recognition," IEEE Trans. Acoustics, Speech, and Signal Proc., *ASSP-27,* No. 6 (December 1979), pp. 588–95.
4. F. Itakura, "Minimum Prediction Residual Principle Applied to Speech Recognition," IEEE Trans. Acoustics, Speech, and Signal Proc., *ASSP-23,* No. 1 (February 1975), pp. 67–72.
5. M. R. Sambur and L. R. Rabiner, "A Speaker-Independent Digit-Recognition System," B.S.T.J., *54,* No. 1 (January 1975), pp. 81–102.
6. L. R. Rabiner, S. E. Levinson, A. E. Rosenberg, and J. G. Wilpon, "Speaker-Independent Recognition of Isolated Words Using Clustering Techniques," IEEE Trans. Acoustics, Speech, and Signal Proc., *ASSP-27,* No. 4 (August 1979), pp. 336–49.
7. J. S. Bridle and M. D. Brown, "Connected Word Recognition Using Whole Word Templates," Proc. Inst. Acoustics, Autumn Conf., 1979.
8. C. S. Myers and L. R. Rabiner, "Connected Digit Recognition Using a Level Building DTW Algorithm," IEEE Trans. Acoustics, Speech, and Signal Proc., *ASSP-29,* No. 3 (June 1981).
9. B. Gold, "Word Recognition Computer Program," MIT, RLE Tech. Report 452, June 1966.
10. C. S. Myers, L. R. Rabiner, and A. E. Rosenberg, "Performance Tradeoffs in Dynamic Time Warping Algorithms for Isolated Word Recognition," IEEE Trans.

Acoustics, Speech, and Signal Proc., *ASSP-28,* No. 6 (December 1980), pp. 622–35.

11. P. V. de Souza, "Statistical Tests and Distance Measures for LPC Coefficients," IEEE Trans. Acoustics, Speech, and Signal Proc., *ASSP-25,* No. 6 (December 1977), pp. 554-9.
12. J. M. Tribolet, L. R. Rabiner, and M. M. Sondhi, "Statistical Properties of an LPC Distance Measure," IEEE Trans. Acoustics, Speech, and Signal Proc., *ASSP-27,* No. 5 (October 1979), pp. 550–8.
13. N. Chomsky and M. Halle, *The Sound Pattern of English,* New York: Harper and Row, 1968.
14. S. E. Levinson, L. R. Rabiner, A. E. Rosenberg, and J. G. Wilpon, "Interactive Clustering Techniques for Selecting Speaker-Independent Reference Templates for Isolated Word Recognition," IEEE Trans. Acoustics, Speech, and Signal Proc., *ASSP-27,* No. 2 (April 1979), pp. 134–41.
15. B. Aldefeld, L. R. Rabiner, A. E. Rosenberg, and J. G. Wilpon, "Automated Directory Listing Retrieval System Based on Isolated Word Recognition," Proc. IEEE, *68,* No. 11 (November 1980), pp. 1364–79.
16. G. S. Sebestyen, *Decision Making Processes in Pattern Recognition,* New York: MacMillan, 1962.
17. A. E. Rosenberg and C. E. Schmidt, "Automatic Recognition of Spoken Spelled Names for Obtaining Directory Listings," B.S.T.J., *58,* No. 8 (October 1979), pp. 1797–823.
18. S. E. Levinson and A. E. Rosenberg, "Some Experiments With a Syntax-Directed Speech Recognition System," Proc. Int. Conf. Acoust., Speech, and Signal Proc. (April 1978), pp. 700–3.

THE BELL SYSTEM TECHNICAL JOURNAL
Vol. 60, No. 5, May–June 1981
Printed in U.S.A.

B.S.T.J. BRIEF

Fast Decryption Algorithm for the Knapsack Cryptographic System

By P. S. HENRY

(Manuscript received December 5, 1980)

I. INTRODUCTION

Public-key cryptosystems offer a degree of flexibility not available with conventional (private-key) systems.[1,2] In particular, the key required for decryption in a public-key system can be changed at will, even in the middle of a message. This makes the task of the eavesdropper very difficult indeed. A frequently cited disadvantage of public-key systems is their relative slowness (typically a few kilobit/s) caused by the large amount of number-crunching they require.[3,4] This has led to the development of hybrid cryptosystems in which a key, exchanged via a slow public-key system, is subsequently used in a fast conventional system, such as the Data Encryption Standard (DES).[5] In this paper we present a fast algorithm for executing the knapsack cipher (a public-key cryptosystem).[6] When implemented with TTL integrated circuitry, this algorithm should permit data rates in the neighborhood of 10 Mbit/s. This speed is sufficient to provide security for a wide range of voice, data, and narrowband video traffic without the need for a hybrid cryptosystem.

Section II presents an elementary example of the knapsack cipher to show how it operates. In Section III we describe the fast algorithm, and in Section IV we discuss a more sophisticated knapsack cipher.

II. AN EXAMPLE OF THE KNAPSACK CIPHER

A very simple (and insecure) knapsack cipher begins with an "easy" knapsack vector generated by a party who wishes to receive encrypted data [eq. (29) of Ref. 6],

$$\mathbf{E} = (1, 2, 4, 8, 17, 35, 68, 142). \tag{1}$$

The eight components of **E** form a super-increasing sequence: Each term is larger than the sum of all those preceding it. Let the data to be encrypted be represented as a vector with eight binary components,

$$\mathbf{D} = (1, 0, 0, 1, 0, 1, 1, 1). \tag{2}$$

To encrypt **D** using **E**, form the dot product,

$$S_E = \mathbf{E} \cdot \mathbf{D} = 254. \tag{3}$$

The number S_E is an encrypted form of **D**.

The super-increasing property of **E** guarantees that **D** can be recovered from S_E by subtracting successive components of E (beginning with the largest) from S_E and keeping the residue. If a component of **E** is less than or equal to the residue at any stage in the subtraction, the corresponding component of **D** is 1. If a component of **E** is larger than the residue, the corresponding **D** component is 0 and we try the next (smaller) component of **E**. This process is illustrated below for $S_E = 254$ [eq. (3)],

$$\begin{array}{cccccccc}
 &&&&&&& \text{Begin}\\
 &&&&&&& \downarrow\\
1 \leftarrow & 1 \leftarrow & 1 \leftarrow & 9 \leftarrow & 9 \leftarrow & 44 \leftarrow & 112 \leftarrow & 254\\
\underline{-1} & \underline{-2} & \underline{-4} & \underline{-8} & \underline{-17} & \underline{-35} & \underline{-68} & \underline{-142}\\
0 & X & X & 1 & X & 9 & 44 & 112
\end{array}$$

$$\mathbf{D} = (1 \quad 0 \quad 0 \quad 1 \quad 0 \quad 1 \quad 1 \quad 1).$$

Of course, **E** cannot be used for secure encryption, because if **E** were obtained by an eavesdropper he could use it to decrypt any transmitted message. The knapsack cipher provides security by transforming **E** into a "hard" knapsack vector **H** (the public key), which can be used for encryption, but which is useless for decryption. To generate this transformation, the receiver chooses two secret integers M and W such that: (*i*) M is larger than the sum of all the components in E, and (*ii*) W and M are relatively prime. (This condition means that W is invertible modulo M: $W^{-1} \cdot W \equiv 1 \bmod M$.) Following Ref. 6, we choose $M = 291$ and $W = 176$ (which implies $W^{-1} = 167$). **H** is generated from **E** by

$$H_j \equiv W \cdot E_j \bmod M, \tag{4}$$

yielding

$$\mathbf{H} = (176, 61, 122, 244, 82, 49, 37, 257).$$

In the ideal case **H** looks like a random sequence; the super-increasing structure of the original **E** is completely obliterated. **H**, the public key, is sent to the transmitter and need not be kept secret.

To encrypt **D** using **H**, form the dot product as before,

$$S_H = \mathbf{H}\cdot\mathbf{D} = 763. \tag{5}$$

S_H is the encrypted data. If the number of components in **H** is large, say 100 or more, then an eavesdropper, even though he has **H** and S_H, cannot recover **D** in a reasonable time. The legitimate receiver, however, can recover **D** easily by using the inverse transformation,

$$S_E \equiv S_H \cdot W^{-1} \bmod M. \tag{6}$$

That is, by using his secret M and W^{-1}, the receiver can convert S_H into the number S_E [eq. (3)], the same number that would have been obtained if **D** had been encrypted with **E** instead of **H**. Once he has S_E, the receiver simply subtracts off successive components of **E** to recover **D**.

III. A FAST DECRYPTION ALGORITHM

The most time-consuming step of the knapsack cipher is the modular multiplication of eq. (6). In practice, the quantities S_H, W^{-1}, and M might be 100 to 200 bits long, making computation of S_E very slow. The calculation can be expedited by considering the n-bit binary expansion of S_H,

$$S_H = b_{n-1}\cdot 2^{n-1} + \cdots + b_0 \cdot 2^0. \tag{7}$$

Substituting eq. (7) into eq. (6), we have

$$S_E \equiv [b_{n-1}(2^{n-1}W^{-1} \bmod M) + \cdots + b_0(2^0 W^{-1} \bmod M)] \bmod M. \tag{8}$$

Each term in the square brackets is the product of a binary digit (0 or 1) and a fixed quantity (in parentheses), which can be computed ahead of time and stored in a memory. Evaluation of S_E thus reduces to a sequence of table lookups and accumulations, one lookup for each bit in S_H. After all the bits in S_H have been processed, the final reduction mod M is accomplished by an easy long division. [The division is "easy" because each term in eq. (8) can be no bigger than M, so the final sum can be no bigger than nM; division by M can therefore be accomplished with only approximately $\log_2 n$ substract-and-shift operations in binary arithmetic.]

Table I shows the contents of the lookup table required for decryption of the example in Section II, along with the binary representation of $S_H = 763$. The value of the sum within the square brackets of eq. (8) is seen to be 1127, which is equivalent to 254 in mod 291 arithmetic, as required.

Table I—Lookup table for decryption of the example in Section II

k	$2^k \cdot 167$ mod 291	b_k
9	241	1
8	266	0
7	133	1
6	212	1
5	106	1
4	53	1
3	172	1
2	86	0
1	43	1
0	167	1

Figure 1 shows a block diagram of the decryption process. The basic steps of lookup, accumulation, reduction mod M and successive subtraction are pipelined, and within each step most of the processing can be performed on all bits in parallel. This architecture results in very fast operation, the speed limitation being either the memory access time or the accumulator add time, whichever is greater. With Schottky TTL and carry-lookahead addition, these times are both in the neighborhood of 50 ns, so a throughput rate of 10 Mbit/s is reasonable.

Implementation of the decryption algorithm using very large scale integration appears attractive. Most of the circuitry is simply a large lookup table, as shown at the top of Fig. 1. Its capacity is determined primarily by the number of components in **E** and the allowed range (number of possible values) for each component of **E**. We can achieve reasonable security by using 100 and 2^{100}, respectively, for these two parameters; this leads to a value for the modulus M in the neighborhood of 2^{200}. Since each component of **H** is less than M, S_H [eq. (5)] will be less than 2^{207}. The lookup table must therefore contain 207 words, each 200 bits long, implying a memory size of approximately 41 kilobits. Additional memory (~15 kilobits) is required to store the components of **E**. Thus, approximately 56 kilobits of memory and some simple arithmetic logic to perform the steps of accumulation, long division, and successive subtraction are adequate to implement the decryption process. This level of complexity is within the range of current VLSI technology.[7,8]

Finally, we remark that a straightforward implementation of Fig. 1 may not be the best approach; several modifications of the basic decryption algorithm must be investigated. For example, the lookup table can be eliminated by calculating the numbers $2^k \cdot W^{-1}$ mod M one-by-one as they are needed for each incoming bit of S_H. Starting with W, successive numbers can be generated by a simple left shift (and subtraction of M if necessary).

IV. ITERATED KNAPSACK TRANSFORMATIONS

The security of the knapsack cipher is enhanced if iterated (multiple) knapsack transformations are employed.[9] For example, the "hard" vector **H** [eq. (4)] can be kept secret and used to generate a "harder" public vector **H**′,

$$H_j' \equiv W' \cdot H_j \bmod M'. \tag{9}$$

Data can be encrypted with **H**′ in the usual fashion,

$$S_{H'} = \mathbf{H}' \cdot \mathbf{D}. \tag{10}$$

If M' is chosen to be greater than the sum of all the components of **H**, then data encrypted using **H**′ may be decrypted using two successive inverse transformations having the form of eq. (6). The cost of this double-iteration technique in terms of the bandwidth efficiency of the cipher is modest. For a 100-component knapsack, the modulus M' will

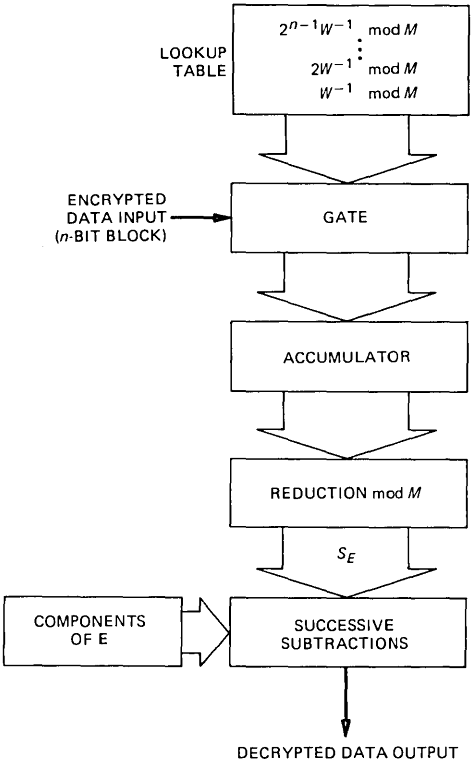


Fig. 1—The fast knapsack decryption algorithm. (Wide arrows signify parallel data transfer.) Pipeline architecture and parallel processing contribute to a high throughput rate. Hardware implementation would require approximately 56 kilobits of memory and a small amount of arithmetic logic.

be roughly 100 times bigger than M; thus $S_{H'}$ will require only about seven more bits than S_H would have required.

We illustrate the double-iteration technique by continuing with the example of Section II. Let $M' = 2001$ and $W' = 1984$, giving $(W')^{-1} = 1177$. From eq. (9) we have

$$\mathbf{H}' = (1010, 964, 1928, 1855, 607, 1168, 1372, 1634). \qquad (11)$$

Encrypting **D** [eq. (2)] with **H'** yields $S_{H'} = 7039$.

Decryption requires two inverse transformations,

$$\begin{aligned} S_E &\equiv S_H \cdot W^{-1} \bmod M \\ &\equiv [S_{H'} \cdot (W')^{-1} \bmod M'] \cdot W^{-1} \bmod M \\ &\equiv [7039 \cdot 1177 \bmod 2001] \cdot 167 \bmod 291 \\ &\equiv 763 \cdot 167 \bmod 291 \\ &= 254. \end{aligned} \qquad (12)$$

The cascaded inverse transformations in eq. (12) can be executed in tandem using the algorithm of Section III. Thus, the decryption process will entail a longer total delay (compared with the single-iteration case), but the net throughput rate will be essentially unchanged.

We mentioned earlier that straightforward use of the multiple iteration technique reduces the bandwidth efficiency of the cipher. It is possible, however, that for a given level of security, multiple interations may actually be more efficient than a single knapsack transformation. This is because the enhanced security associated with repeated transformations might permit a smaller range for the components of **E**, and hence smaller values for the moduli M, M', etc. The consequent reduction in the encrypted block length could offset the seven-bit increase normally associated with each iteration.

V. CONCLUSIONS

The existence of a fast algorithm for decryption of the knapsack cipher means that the advantages of public-key cryptosystems can be realized even in high-speed applications. Full integration of the decryption process onto a single chip appears feasible with current VLSI technology. The relationships among cipher security, bandwidth efficiency, and number of iterations need further investigation.

VI. ACKNOWLEDGMENT

F. H. Myers has been generous with his time and insights; I am grateful for his help.

REFERENCES

1. M. E. Hellman, "An Overview of Public Key Cryptography," IEEE Commun. Soc. Mag., *16* (November 1978), pp. 24-32.
2. W. Diffie and M. E. Hellman, "Privacy and Authentication: An Introduction to Cryptography," Proc. IEEE, *67* (March 1979), pp. 397-427.
3. B. P. Schanning, "Data Encryption with Public Key Distribution," EASCON Conf. Rec., Washington, D.C., October 9-11, 1979, pp. 653-60.
4. G. J. Simmons, "Symmetric and Asymmetric Encryption," Comput. Surv., *11* (December 1979), pp. 306-30.
5. F. H. Myers, "A Data Link Encryption System," Conf. Rec. Nat. Telecommun. Conf., Washington, D.C., November 27-29, 1979, Paper 43.5.
6. R. C. Merkle and M. E. Hellman, "Hiding Information and Signatures in Trapdoor Knapsacks," IEEE Trans. Inf. Theor., *IT-24* (September 1978), pp. 525-30.
7. S. Matsue et al., "A 256K Dynamic RAM," 1980 IEEE Int. Solid State Circuits Conf. Dig. Tech. Papers, February 13-15, 1980, pp. 232-3.
8. P. M. Russo, "VLSI Impact on Microprocessor Evolution, Usage, and System Design," IEEE J. Solid State Circuits, *SC-15* (August 1980), pp. 397-405.
9. A. Shamir and R. E. Zippel, "On the Security of the Merkle-Hellman Cryptographic Scheme," IEEE Trans. Inf. Theor., *IT-26* (May 1980), pp. 339-40.

CONTRIBUTORS TO THIS ISSUE

Jont B. Allen, B.S. (E.E.), 1966, University of Illinois; M.S., 1968, Ph.D., 1970, University of Pennsylvania; Bell Laboratories, 1970—. Mr. Allen is presently working in the areas of cochlear modeling, small room acoustics, dereverberation of speech signals, and digital signal processing. His main efforts have been directed toward cochlear mathematical modeling, the problem of removing room reverberation from recorded speech signals by digital signal processing, and modeling the psychophysical effects of room reverberation.

A. Carnevale, B.S. (Physics), 1960, Fairleigh Dickinson University; Bell Laboratories, 1969—. At Bell Laboratories, Mr. Carnevale has been engaged in nuclear magnetic resonance, electron paramagnetic resonance, and computer software. For the last two years, his work has been devoted to fiber optics.

Barry G. Haskell, B.S., 1964, M.S., 1965, and Ph.D., 1968, all in electrical engineering, University of California, Berkeley; University of California, 1965–1968; Bell Laboratories, 1968—; Rutgers University, 1977, 1979. Mr. Haskell was a Research Assistant in the University of California Electronics Research Laboratory and a part-time faculty member of the Department of Electrical Engineering at Rutgers University. At Bell Laboratories, he is presently head of the Radio Communications Research Department, where his research interests include television picture coding and transmission of digital and analog information via microwave radio. Member, IEEE, Phi Beta Kappa, Sigma Xi.

John D. Healy, B.S. (Mathematics), 1971, St. Bonaventure University; Ph.D. (statistics), 1975, Purdue University; Bell Laboratories, 1976—. At Bell Laboratories, Mr. Healy has been involved in planning and analyzing service measurements, measuring and reporting the reliability of Western Electric products, and designing and analyzing complex surveys. He is currently the supervisor of the network performance characterization analysis group.

Nuggehally S. Jayant, B.Sc. (Physics and Mathematics), 1962, Mysore University; B.E., 1965, and Ph.D. (Electrical Communication Engineering), 1970, Indian Institute of Science, Bangalore; Research Associate at Stanford University, 1967–1968; Bell Laboratories, 1968—. Mr. Jayant was a Visiting Scientist at the Indian Institute of Science from January-March 1972 and August-October 1975. He has worked in the field of digital coding and transmission of waveforms, with special reference to speech communications. He is editor of an IEEE Reprint Book on *Waveform Quantization and Coding.*

Linda Kaufman, Sc.B. (Applied Mathematics), 1969, Brown University; M.S., 1971, Ph.D., 1973 (Computer Science), Stanford University; Bell Laboratories, 1976—. Before joining Bell Laboratories, she taught at the University of Aarhus, Denmark, and the University of Colorado, Boulder, Colorado. Her primary interests include numerical linear algebra and numerical techniques for function minimization. Member, Association for Computing Machinery, SIAM.

James McKenna, B.Sc., 1951 (Mathematics), Massachusetts Institute of Technology; Ph.D., 1960 (Mathematics), Princeton University; Bell Laboratories, 1960—. Mr. McKenna has done research in quantum mechanics, classical mathematical physics, stochastic differential equations, and numerical analysis. More recently he has been interested in stochastic problems arising from communication and computer networks, and computer performance evaluation. He is head of the Mathematics of Physics and Networks Department.

Debasis Mitra, B.Sc., 1965, and Ph.D., 1967 (Electrical Engineering), London University; United Kingdom Atomic Energy Authority Research Fellow, 1966-1967; Bell Laboratories, 1968—. Mr. Mitra has worked on the stability analysis of nonlinear systems, semiconductor networks, growth models for new communication systems, speech waveform coding, nonlinear phenomenon in digital signal processing, adaptive filtering, and network synchronization. Most recently he has been involved in the analytic and computational aspects of stochastic networks and computer communications. Member, IEEE, SIAM.

John A. Morrison, B.Sc. (Mathematics), 1952, King's College, University of London; Sc.M. (Applied Mathematics), 1954, Ph.D. (Applied Mathematics), 1956, Brown University; Bell Laboratories, 1956—. Mr. Morrison has done research in various areas of applied mathematics and mathematical physics. He has recently been interested in queuing problems associated with data communications networks. He was a Visiting Professor of mechanics at Lehigh University during the fall semester 1968. Member, American Mathematical Society, SIAM, IEEE, Sigma Xi.

Norman H. Noe, B.E.E., 1962, Rensselaer Polytechnic Institute; S.M.E.E., 1964, Massachusetts Institute of Technology; Ph.D. (Operations Research), 1971, New York University; Bell Laboratories, 1962—. Mr. Noe has been involved in designing software modules for studying metropolitan and toll alternate routing networks, forecasting rural telephone demand for small geographic areas, developing algorithms and software for a loop inventory system, and analyzing forecast errors and their impact on the feeder-cable network. Currently, he is developing economic models for mechanized assignment systems. Member, Tau Beta Pi, Eta Kappa Nu, Sigma Xi (Associate), IEEE, ORSA.

George E. Peterson, B.S. (Physics), 1956, Ph.D. (Solid State Physics), 1961, University of Pittsburgh; Bell Laboratories, 1961—. At Bell Laboratories, Mr. Peterson did studies on low-noise amplifiers for a short period of time. He then studied laser crystals, nonlinear optic materials, glass structure, and recently propagation of light in optical fibers. Member, American Physical Society, American Ceramic Society, Society for Glass Technology, American Crystallographic Association.

Un-Chul Paek, B.S. (Engineering), 1957, Korea Merchant Marine Academy, Korea; M.S., 1965, Ph.D., 1969, University of California, Berkeley; Western Electric, 1969—. At the Western Electric Engineering Research Center, Princeton, N. J., Mr. Paek has been engaged primarily in research on laser material interaction phenomena and fiber optics. Member, Optical Society of America, American Ceramic Society, Sigma Xi.

Lawrence R. Rabiner, S.B. and S.M., 1964, Ph.D. (electrical engineering), Massachusetts Institute of Technology; Bell Laboratories, 1962—. From 1962 through 1964, Mr. Rabiner participated in the cooperative plan in electrical engineering at Bell Laboratories. He worked on digital circuitry, military communications problems, and problems in binaural hearing. Presently, he is engaged in research on speech communications and digital signal processing techniques. He is coauthor of *Theory and Application of Digital Signal Processing* (Prentice-Hall, 1975) and *Digital Processing of Speech Signals* (Prentice-Hall, 1978). Former President, IEEE, G-ASSP Ad com; former Associate Editor, G-ASSP Transactions; former member, Technical Committee on Speech Communication of the Acoustical Society. Member, G-ASSP Technical Committee of the Acoustical Society, G-ASSP Technical Committee on Speech Communication, IEEE Proceedings Editorial Board, Eta Kappa Nu, Sigma Xi, Tau Beta Pi. Fellow, Acoustical Society of America, IEEE.

K. G. Ramakrishnan, B.S., 1970, M.S., 1972 (Electrical Engineering), Indian Institute of Technology, Kanpur, India; M.S., 1976, Ph.D., 1978 (Computer Science), Washington State University; Bell Laboratories, 1978—. Mr. Ramakrishnan has worked on developing analytical and simulation models for the DIRECT II project. He is currently working on the performance analysis and architecture for a new communication system. Member, ORSA, SIAM, ACM.

Judith B. Seery, B.A. (Mathematics), 1968, College of St. Elizabeth; M.S. (Mathematics), 1972, New York University; Bell Laboratories, 1968—. Ms. Seery does analysis and computation in the Mathematics and Statistics Research Center. She has recently participated in problems in signal detection, fiber optics, and multidimensional scaling. Member, SIAM, Mathematical Association of America, Association for Women in Mathematics.

Julius O. Smith, III, B.S. (E.E.), 1975, Rice University; M.S. (E.E.), 1978, Stanford University; Bell Laboratories, summer of 1979. At Bell Laboratories, Mr. Smith worked on speech coding and spectrum analysis in the Acoustics Research Department. He is presently a graduate student at the Information Systems Laboratory, department of electrical engineering, Stanford University. Member, IEEE, Tau Beta Pi, Audio Engineering Society, Center for Computer Research in Music and Acoustics.

Jay G. Wilpon, B.S. (mathematics), A.B. (economics), cum laude, 1977, Lafayette College; Bell Laboratories, 1977—. At Bell Laboratories, Mr. Wilpon has been engaged in speech communications research and is presently concentrating on problems of speech recognition.

PAPERS BY BELL LABORATORIES AUTHORS

PHYSICAL SCIENCES

AES Study of Tin-Lead Alloys: Effects of Ion Sputtering and Oxidation on Surface Composition and Structure. R. P. Frankenthal and D. J. Siconolfi, J Vac Sci Technol, *17,* No. 6 (1980), pp. 1315–19.

Brillouin Scattering from Polymers. G. D. Patterson and J. P. Latham, J Polymer Sci Macromol Rev, *15* (1980), pp 1–27.

Comments on "On the Mechanism of Transduction in Optical Fiber Hydrophones" [J Acoust Soc Am, *66* (1979), pp 976–9]. R. N. Thurston, J Acoust Soc Am, *67* (1980), pp 1072–3.

Conduction-Electron Screening in Metallic Oxides. IrO_2. G. K. Wertheim and H. J. Guggenheim, Phys Rev B, *22* (November 15, 1980), pp 4680–3.

Cosputtered Molybdenum Silicides on Thermal SiO_2. S. P. Murarka, D. B. Fraser, T. F. Retacjzyk, Jr., and T. T. Sheng, J Appl Phys, *51,* No. 10 (1980), pp 5380–5.

Detailed Comparison of the Williams-Watts and Cole-Davidson Functions. C. P. Lindsey and G. D. Patterson, J Chem Phys, *73,* No. 7 (October 1980), pp 3348–57.

The Effect of Added Acetylene on the rf Discharge Chemistry of C_2F_6. A Mechanistic Model for Flurocarbon Plasmas. E. A. Truesdale, G. Smolinsky, and T. M. Mayer, J Appl Phys, *51* (May 1980), pp 2909–13.

The Effect of Gold Film Grain Size on Copper Diffusion and Surface Accumulation by Oxidation. M. R. Pinnel, H. G. Tompkins, and J. A. Augis, Microstructural Sci, *8* (1980), pp 13–27.

Electronic Properties of CsSn. C. W. Bates, Jr., P. M. Th. M. Van Attekum, G. K. Wertheim, K. W. Witt, and D. N. E. Buchanan, Phys Rev B, *22* (1980), pp 3968–72.

Electronic Properties of Some C_aF_2—Structure Intermetallic Compounds. P. M. Th. M. Van Attekum, G. K. Wertheim, G. Crecelius, and J. H. Wernick, Phys Rev B, *22* (1980), pp 3998–4004.

GaAs Oxidation and the Ga-As-O Equilibrium Phase Diagram. C. D. Thurmond, G. P. Schwartz, G. W. Kammlott, and B. Schwartz, J Electrochem Soc, *127* (June 1980), pp 1366–71.

Hypersonic Relaxation in Amorphous Polymers. G. D. Patterson, Crit Rev Solid State Mater Sci, *9,* No. 4 (November 1980), pp 373–97.

Increasing the Quantum Efficiency of Nickel Phthalocyanine Films by Oxygen Adsorption. M. E. Musser and S. C. Dahlberg, Surf Sci, *100* (1980), pp 605–12.

Limits on the Transition State Geometry for Metal Insertion into a Carbon-Hydrogen Bond. J. W. Suggs and G. D. N. Pearson, Tetrahedron Lett, *21* (1980), pp 3853–6.

Mechanically Bistable Liquid Crystal Display Structures. R. N. Thurston, J. Cheng, and G. D. Boyd, IEEE Trans Electron Devices, *ED-27,* pp 2069–80.

Mixed Valency of TmSe. G. K. Wertheim, W. Eib, E. Kaldis, and M. Campagna, Phys Rev B, *22* (December 15, 1980), pp 6240–6.

New Rectifying Semiconductor Structure by Molecular Beam Epitaxy. C. L. Allyn, A. C. Gossard, and W. Wiegmann, Appl Phys Lett, *36,* No. 5 (March 1980), pp 373–6.

On the Possible Increase of the Atmospheric Methane. T. E. Graedel and J. E. McRae, Geophys Res Lett, *7,* No. 11 (November 1980), pp 977–9.

Phase Separation in a Spin Glass. T. M. Hayes, J. W. Allen, J. B. Boyce, and J. J. Hauser, Phys Rev, *22* (November 1, 1980), pp 4503–9.

Photoluminescence Dynamics in Chalcogenide Glasses and Crystals. M. A. Bosch, R. W. Epworth, and D. Emin, J Non-Crystal Solids, *40* (1980), pp 587–94.

Photon Correlation Spectroscopy Near the Glass Transition in Polymers. G. D. Patterson and J. R. Stevens, ACS Polymer Preprints, *21,* No. 2 (1980), pp 16–7.

Photon Correlation Spectroscopy of Polystyrene Solutions. G. D. Patterson, J. P. Jarry, and C. P. Lindsey, Macromolecules, *13* (May 1980), pp 668–70.

Piezoelectric $KCo[Au(N)_2]_3$: Room Temperature Crystal Structure of a Cobalt-Hardened Gold Electrodeposition Process Component. S. C. Abrahams, J. L.

Bernstein, R. Liminga, and E. T. Eisenmann, J Chem Phys, *73,* No. 9 (November 1, 1980), pp 4585-90.

Polarized Light. P. F. Liao, in *Encyclopedia of Physics*, edited by R. G. Lerner and G. L. Trigg, Reading, Mass: Addison-Wesley, 1980, pp 771-2.

Pyroelectricity. S. C. Abrahams, in *Encyclopedia of Physics*, edited by R. G. Lerner and G. L. Trigg, Reading, Mass: Addison-Wesley, 1980, pp 798-9.

Rayleigh-Brillouin Scattering in Polymers. G. D. Patterson, in *Methods of Experimental Physics*, edited by R. A. Faver, New York: Academic, Vol 16A, pp 170-204.

Refractory Silicides of Titanium and Tantalum for Low-Resistivity Gates and Interconnects. S. P. Murarka, D. B. Fraser, A. K. Sinha, and H. J. Levinstein, IEEE J Solid State Circuits, *SC15,* No. 4 (August 1980), pp 474-81.

Scaling the Micron Barrier with X-Rays. M. P. Lepselter, IEDM Tech Dig, December 8-10, 1980, pp 42-4.

Secondary Ion Mass Spectrometric Analysis of Cobalt-Hardened Gold Electroplate Surfaces. R. Schubert, J Electrochem Soc, *128,* No. 1 (January 1981), pp 126-31.

Stability of LPCVD Polysilicon Gates on Thin Oxides. S. P. Murarka, A. K. Sinha, and H. J. Levinstein, J Electrochem Soc, *127* (1980), pp 2447.

The Temperature Dependence of the Surface Photovoltage of Nickel Phthalocyanine Films. S. C. Dahlberg and M. E. Musser, J Chem Phys, *73,* No. 8 (1980), pp 4135-6.

Thermal Oxidation of Hafnium Silicide Films on Silicon. S. P. Murarka and C. C. Chang, Appl Phys Lett, *37,* No. 7 (July 1980), pp 639-41.

Thin Film Interaction Between Titanium and Polycrystalline Silicon. S. P. Murarka and D. D. Fraser, J Appl Phys, *51,* No. 1 (January 1980), pp 342-9.

Transmission Electron Microscopy of Au-Based Ohmic Contacts to n-AlGaAs. C. C. Chang, T. T. Sheng, R. J. McCoy, S. Nakahara, V. G. Keramidas, and F. Ermanis, J Appl Phys, *50,* No. 11 (November 1977), pp 7030-3.

Use of a Concentric-Arc Grating as a Thin-Film Spectrograph for Guided Waves. P. K. Tien and R. J. Capik, Appl Phys Lett, *37,* No. 6 (September 1980), pp 524-6.

COMPUTING

Computer-Aided Design—An Overview. H. Y. Chang, Proc Infotech State Art Rev 1980, *2* (November 1980), pp 27-42.

A Distributed *UNIX* System—The Tandem Experiment. A. M. Usas, Proc Natl Electron Conf, *34* (October 1980), pp 16-8.

Generating Exploratory Telecommunication Software Using Functional Simulation. W. Montgomery, Proc Natl Telecommun Conf (December 1, 1980).

ENGINEERING

Administration of Pair Gain Systems. G. R. Boyer and M. A. Schwartz, Proc Natl Telecommun Conf, *1* (December 1980), pp 12.1.1-12.1.6.

Cable Sheath Buckling Studies and the Development of a Bonded Stalpeth Sheath. G. M. Yanizeski, E. L. Johnson, and R. G. Schneider, Proc 29th Int Wire Cable Symp, Cherry Hill, N. J., November 18, 1980.

CCIS Network Performance. A. J. Rupert and J. M. Sebeson, Proc Natl Electron Conf 1980, October 27-29, 1980, *34,* pp 167-71.

Comments on "Characterization of Random Array Peak Sidelobe." V. D. Agrawal and Y. T. Lo, IEEE Trans Antennas Prop, *AP-28* (November 1980), pp. 946-8.

Customer Perceived Delays in a Packet Switched Network. P. K. Verma, Proc Natl Telecommun Conf, November 30, 1980, pp 25.1.1-25.1.6.

Design and Evaluation of a Contactless Piezoelectric Keyboard Using Polyvinylidene Fluoride as the Active Element. G. T. Pearman, J. L. Hokanson, and T. R. Meeker, Ferroelectrics, 1979 IEEE Int Appl Ferroelectrics, June 1979, *27, 28* (Special Issue 1980), pp 311-4.

Designing Network Diagrams. J. B. Kruskal and J. B. Seery, in US Bureau of the Census, Proc First General Conf Social Graphics, Washington, DC (July 1980), pp 22-50.

Effect of Routing Control on an Economic-Service Protection Tradeoff in Network Design. J. N. Anderson, Proc Natl Telecommun Conf, *2* (November 1980), pp 35.4.1–35.4.7.

Effects of Heat Treatment on Surface Conditions of Iron-Nickel Alloys and the Influence on Mercury Switch Behavior. J. E. Bennett, IEEE Trans, *CHMT-3,* No. 1 (March 1980), pp 172–80.

Ferrites for Non-Microwave Applications. P. I. Slick, in *Ferromagnetic Materials,* New York: Elsevier, 1980, Vol 2, pp 189–241.

FT3—A Metropolitan Trunk Lightwave System. I. Jacobs and J. R. Stauffer, Proc IEEE, *68* (October 1980), pp 1286–90.

Functional Level, Concurrent Fault Simulation. L. P. Henckles, K. M. Brown, and C. Y. Lo, Dig 1980 Test Conf, Philadelphia, PA, November 11–13, 1980, pp 479–85.

Impact of Interference on 16 QAM System Requirements. J. J. Kenny, Proc Natl Telecommun Conf, *2* (November 1980), pp 43.4.1–43.4.6.

Lightguide Cable Installation in Underground Plant. A. L. Hale, D. L. Pope, and D. R. Rutledge, Proc Third Int Fiber Optics and Commun Exposition, San Francisco, CA, September 16–18, 1980, pp 227–31.

Long-Wavelength Propagation in Composite Elastic Media. I. Spherical Inclusions. J. G. Berryman, J Acoust Soc Am, *68,* No. 6 (1980), pp 1809–19.

Long-Wavelength Propagation in Composite Elastic Media. II. Ellipsoidal Inclusions. J. G. Berryman, J Acoust Soc Am, *68,* No. 6 (1980), pp 1820–31.

A Microcomputer-Controlled Test Set for Measuring Tensile Strength of Connections with Real-Time Failure Mode Identification. R. G. Fekula, Proc NEPCON West 1980, pp 180–8.

Miniature Packaged Crystal Oscillators. R. E. Paradysz, D. M. Embree, V. R. Saari, and R. J. McClure, Proc 34th Annu Symp Frequency Control 1980, pp 475–87.

New Methods of Fine Feature Fabrication Using e-Beam Lithography. E. L. Hu, L. D. Jackel, R. E. Howard, L. A. Fetter, P. Grabbe, and D. M. Tennant, Proc Symp Electron Ion Beam Sci Technol (Ninth Intl Conf), *80-6* (1980), pp 200–5.

Observations of the Polarization Dependent Properties of Rain and Ice Depolarization. H. W. Arnold, D. C. Cox, H. H. Hoffman, and R. P. Leck, Proc Natl Telecommun Conf, *2* (November 1980), pp 43.3.1–43.3.6.

Optical Sources for Fiber Transmission Systems. A. A. Bergh, J. A. Copeland, and R. W. Dixon, Proc IEEE, *68,* No. 10 (October 1980), pp 1240–7.

Planning for the Application of CCITT No. 7 in the CCIS Network. G. G. Schlanger, Proc Natl Telecommun Conf, *1* (December 1980), pp 2.5.1–2.5.4.

A Practical Ground Potential Rise Prediction Technique for Power Stations. F. P. Zupa and J. F. Laidig, IEEE Trans Power Apparatus Systems, *PAS-99* (January/February 1980), pp 207–16.

Precise Characterization of Long Nonidentical-Fiber Splice Loss Effects. R. B. Kummer, Proc Sixth European Conf Optical Commun, September 16–19, 1980, pp 302–5.

A Procedure for Estimating Percentile Response Times in a Packet Switched Network. P. K. Verma, Proc Intl Conf Commun, *ICC-80* (June 1980), pp 61.5.1–61.5.4.

Rain Attenuation at 10–30 GHz along Earth-Space Paths: Elevation Angle, Frequency, Seasonal, and Diurnal Effects. H. W. Arnold, D. C. Cox, and A. J. Rustako, Jr., Proc Intl Conf Commun, *ICC-80,* No. 3 (June 1980), pp 40.3.1–40.3.7.

Semiconductor/Device Development in the 1970s and 1980s—A Perspective. S. M. Sze, IEDM Tech Dig, December 8–10, 1980, pp 7–12.

A Signaling Protocol for Digital Subscriber Lines. R. G. Cornell and D. J. Stelte, Proc Natl Telecommun Conf, December 3, 1980, pp 45.5.1–45.5.5.

Single-Mode Optical Fiber Switch. C. M. Miller, R. B. Kummer, S. C. Mettler, and D. N. Ridway, Electron Lett, *16,* No. 20 (September 1980), pp 783–4.

Solution to Core Growth in Connectorized Cable. M. R. Reynolds, G. M. Yanizeski, and R. N. McIntyre, Proc 29th Int Wire Cable Symp, November 18–20, 1980, pp 38–47.

T1D Line Repeater Design. A. Anuff and J. J. Ludwick, Proc Natl Telecommun Conf, *2* (December 1980), pp 39.4.1–39.4.5.

Traffic Network Design of New Private Switched Services. A. L. Kalro, Proc Networks '80 Conf (Telecommun Networks Plan), September 29, 1980, pp 109–15.

VLSI in Telecommunications. G. A. Bulger, Proc Natl Electron Conf, *34* (October 1980), pp 249–53.

SOCIAL AND LIFE SCIENCES

Analysis of Time-Varying Imagery Through the Representation of Position and Shape Changes. M. O. Ward and Y. T. Chien, Proc 5th Int Conf Pattern Recognition, *2* (December 1980), pp 1236–9.

Educational Opportunities and the AES. R. G. Baker, Plating Surf Finish, *67,* No. 11 (November 1980), p 8.

Perception of Size of One Object Among Many. J. Z. Leumson and F. S. Frome, Science, *206* (December 1979), pp 1425–6.

Shifts in Perception of Size After Adaptation to Grating. F. S. Frome, J. Z. Leumson, J. T. Danielson, and J. E. Clavadetscher, Science, *206* (December 1979), pp 1327–9.

Telematics and Social Services. D. Gillette, in *Telecommunications and Productivity*, edited by M. L. Moss, Reading, MA: Addison-Wesley, 1981, pp 334–7. (Based on the Int. Conf. Telecommun. Productivity at the NYU Center for Science and Technology Policy held January 29-30, 1980.)

MANAGEMENT AND ECONOMICS

Comments on a Queueing Inequality. D. P. Heyman, Management Sci, *26* (September 1980), pp 954–9.

What is Technical Innovation? R. Mueser, 1980 IEEE Management Conf Rec (November 1980), pp 12–4.

CONTENTS, JULY–AUGUST 1981

Part 1

T CARRIER CHARACTERIZATION

Part 2

No. 4 ELECTRONIC SWITCHING SYSTEM

THE BELL SYSTEM 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.

Bell System