

THE MAY/JUNE 1982  
VOL. 61, NO. 5



BELL SYSTEM  
TECHNICAL JOURNAL

---

<b>Integral Representations and Asymptotic Expansions for Closed Markovian Queueing Networks: Normal Usage</b> J. McKenna and D. Mitra	661
<b>Parasitic Insensitive, Biphase Switched Capacitor Filters Realized With One Operational Amplifier Per Pole Pair</b> K. R. Laker, P. E. Fleischer, and A. Ganesan	685
<b>Moment Formulae for a Class of Mixed Multi-Job-Type Queueing Networks</b> H. Heffes	709
<b>Adaptive Intra-Interframe DPCM Coder</b> P. Pirsch	747
<b>Human Performance Engineering Considerations for Very Large Computer-Based Systems: The End User</b> I. S. Yavelberg	765
<b>Graphic Displays of Combined Presentations of Acoustic and Articulatory Information</b> J. E. Miller and O. Fujimura	799
<b>1A VOICE STORAGE SYSTEM</b>	
<b>Voice Storage in the Network—Perspective and History</b> E. Nussbaum	811
<b>Prologue</b> G. D. Bergland, W. T. Hartwell, and G. W. Smith, Jr.	815
<b>New Custom Calling Services</b> D. P. Worrall	821
<b>Architecture and Physical Design</b> R. G. Cornell and J. V. Smith	841

*(Contents continued on back cover)*

# THE BELL SYSTEM TECHNICAL JOURNAL

## ADVISORY BOARD

D. E. PROCKNOW, *President*, *Western Electric Company*  
I. M. ROSS, *President*, *Bell Telephone Laboratories, Incorporated*  
W. M. ELLINGHAUS, *President*, *American Telephone and Telegraph Company*

## EDITORIAL COMMITTEE

A. A. PENZIAS, *Chairman*

M. M. BUCHNER, JR.	R. A. KELLEY
A. G. CHYNOWETH	R. W. LUCKY
R. P. CLAGETT	L. SCHENKER
T. H. CROWLEY	W. B. SMITH
B. P. DONOHUE, III	G. SPIRO
I. DORROS	J. W. TIMKO

## EDITORIAL STAFF

B. G. KING, *Editor*  
PIERCE WHEELER, *Managing Editor*  
HEDWIG A. DEUSCHLE, *Assistant Editor*  
H. M. PURVIANCE, *Art Editor*  
B. G. GRUBER, *Circulation*

**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 1J-319, 101 J. F. Kennedy Parkway, Short Hills, N. J. 07078. Checks for subscriptions should be made payable to The Bell System Technical Journal and should be addressed to Bell Laboratories, Circulation Group, 101 J. F. Kennedy Parkway, Short Hills, N.J. 07078. 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.

© 1982 American Telephone and Telegraph Company. ISSN0005-8580

Single copies of material from this issue of The Bell System Technical Journal may be reproduced for personal, noncommercial use. Permission to make multiple copies must be obtained from the editor.

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 1J-319, 101 J. F. Kennedy Parkway, Short Hills, N. J. 07078. 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 61

May-June 1982

Number 5

---

Copyright © 1982 American Telephone and Telegraph Company. Printed in U.S.A.

## **Integral Representations and Asymptotic Expansions for Closed Markovian Queueing Networks: Normal Usage**

By J. McKENNA and D. MITRA

(Manuscript received October 28, 1982)

*In designing a computer system, it is vitally important to be able to predict the performance of the system. Often, quantities such as throughput, processor utilization, and response time can be predicted from a closed queueing network model. However, until now the computations involved were not feasible for large systems which call for models with many processing centers and many jobs distributed over many classes. We give a radically new approach for handling such large networks—an approach that begins with a representation of the quantities of interest as ratios of integrals. These integrals contain a large parameter reflecting the size of the network. Next, expansions of the integrals in inverse powers of this large parameter are derived. For cases in which the number of processing centers is greater than one, this is the only technique we know of that yields the complete asymptotic expansion. Our method for computing the terms of the expansion can be interpreted as decomposing the original network into a large number of small “pseudonetworks.” Our technique also yields easily computed error bounds when only the first few terms of the expansion are used. This method has been implemented in a software package with which we can analyze systems larger by several orders of magnitude than was previously possible.*

### **I. INTRODUCTION**

Closed Markovian queueing networks, which are tractable in having

the product form (or separability) in their stationary distribution, continue to have a profound influence on computer communication, computer systems analysis, and traffic theory.<sup>1-4</sup> The closed networks have been used to model multiple-resource computer systems,<sup>2,5</sup> multiprogrammed computer systems,<sup>2,6-8</sup> time-sharing,<sup>2</sup> and window flow control in computer communication networks;<sup>9,10</sup> networks with blocking<sup>11,12</sup> require the analysis of a large number of closed networks. Not surprisingly, considerable effort has gone into devising efficient procedures for computing the partition function,<sup>13-17</sup> an element of the product form solution requiring significant computation. More recently, mainly spurred by parallel technological development in computer communication, there has been a focusing of effort on *large* closed networks<sup>11,15,18-21</sup> with many classes of jobs and transactions and large populations in each class. The point of departure of this effort is the realization that the earlier recursive techniques for computing the partition function are severely limited in terms of computing time, memory storage, and attainable accuracy when it comes to the large networks presently demanding analysis.

In an earlier paper,<sup>2</sup> we introduced a new approach to calculating the partition function. We showed there that the partition function could be represented as an integral containing a large parameter which, in some sense, reflected the large size of the network. In general, the partition function is represented by a multiple integral. However, in the special case where there is only one node at which queueing can occur (a node of type 1, 2, or 4 in the terminology of Ref. 4), the partition function is represented by a single integral. In Ref. 21 we applied standard theory to obtain asymptotic expansions of the integral representation. The standard techniques, however, cannot be extended directly to multiple integrals. In this paper, we make use of the special properties of our integral representation and obtain a method for generating asymptotic expansions of our integrals which works equally well for single or multiple integrals. For our single integrals, the techniques developed in this paper are easier to apply than the standard techniques we used in Ref. 21. For our multiple integrals, our technique is the only one we know of which can be used to obtain higher order terms in the asymptotic expansion.

The computational effort of solving a large network with  $p$  classes and with populations in each class of the order of magnitude of 100 is here reduced to be roughly as complex as solving by older techniques for the partition functions of ( $p$  choose 4) networks where each of these networks has a total population of, at most, seven allocated over, at most, four classes. Thus, we have reduced the problem to the solution of a large number of small problems. One consequence is that for  $p$  large enough, even our technique will be computationally intrac-

table. Our results nevertheless allow large, previously intractable networks to be solved with vastly reduced computational effort and with great accuracy. For example, a network with 20 classes, each class having a population of 100, can be handled easily by our technique. Two noteworthy aspects of this work are (i) the built-in notion of depth of accuracy: by computing more terms of the asymptotic expansion, it is possible to match computational effort to desired degree of accuracy, and (ii) a comprehensive error analysis that allows estimates to be accompanied by sharp error bounds with little incremental effort. Let us elaborate.

Section III recapitulates and extends the results in Ref. 21 to obtain representations as integrals of the partition function of most, but not all, closed product-form networks.<sup>4</sup> The class-by-class breakdown of the utilization of each processor, itself simply related to mean response time and throughput, is given in terms of a ratio of two integrals. These are multiple integrals with multiplicity equal to the number of centers in the network where queues may form.

The asymptotic expansions are in powers of  $(1/N)$  where  $N$  is a parameter designed to reflect network size. Our computational experience has been that five terms in the mean value expansions are generally adequate both for large networks, for which our asymptotic technique is particularly well suited, and, to a surprising extent, for small networks as well.<sup>21</sup> Section IV gives the procedure for generating the general coefficient of the expansion, while the leading coefficients are explicitly derived.

A remarkable feature of the composition of the coefficients make their computation amenable to various techniques. As shown in Section 4.4, the coefficients turn out to be very simply related to the partition function of a certain hypothetical network which we call the pseudonetwork. The topology is related but not identical to that of the given network and the processing rates are quite different. Most importantly, however, to compute the leading expansion coefficients, it is necessary to compute the partition function for only small populations in the pseudonetwork. Thus, to compute five terms of the utilization expansion, we need only consider the total population over all classes to be at most seven in the pseudonetwork. Small population in the pseudonetwork has the consequence of its partition function being solvable by existing recursive techniques of proven efficacy.

Section V proves that the series in  $(1/N)$  given in Section IV is endowed with properties substantially more desirable than those possessed by asymptotic expansions in general. It is shown in Section 5.2 that the truncation error is numerically less than the first neglected term of the expansion, and has the same sign. Thus, except for the effort in computing an additional term, all that is generally needed for

an error analysis is already available in the basic series that is computed.

We note here that while the mathematical literature on single integrals is extensive,<sup>22-24</sup> there is little on asymptotic expansions of multiple integrals. Two notable investigations along lines different from here are Bleistein's<sup>25</sup> and Skinner's.<sup>26</sup> Both are marked by extreme complexity. Bleistein gives the leading coefficient, while Skinner obtains the second term. Both terms are quite complicated.

Let us now elaborate on some limitations of the paper. An important conclusion of Ref. 21 is that qualitatively different expansions of the integrals exist depending on whether usage is "normal," "high," or "very high." This is even more true in the present context of multiple processing centers. Therefore, this paper is devoted exclusively to the case of normal usage. We propose to consider the remaining usage conditions in the future. Exactly what is meant by normal usage is explained in Section 4.1.

The paper also assumes that, for each class of jobs, the routing through the network contains at least one infinite server (IS) center. It turns out that for networks in which this is not true, the asymptotic expansions are more appropriately derived in the context of either high or very high usage conditions. However, certain basic results on the integral representations of partition functions and mean values are derived in Section III regardless of whether IS centers exist for all classes. This paper does *not* allow load dependent service rates in the first-come-first-serve centers.

The results in this paper have been incorporated in a large software package and this will be reported elsewhere. No results on moments of individual queue lengths are given here. However, their integral representations and asymptotic expansions are very similar,<sup>21</sup> and the details will be published elsewhere.

## II. PRODUCT FORM IN STATIONARY DISTRIBUTIONS: PRELIMINARIES

### 2.1 Product form

We recapitulate some of the well-known results<sup>4</sup> concerning product form in stochastic networks and present them in the form that will be used later.

Let  $p$  be the number of classes of jobs and reserve the symbol  $j$  for indexing class. Hence, 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 queueing from the remaining centers of type 3 which do not. (The definition of type 1 to 4 centers is given in Ref. 4.) Thus, centers 1 through  $q$  will be the

queueing centers, while  $(q + 1)$  through  $s$  will be the type 3 centers, which have also been called think nodes and IS nodes. We reserve the symbol  $i$  for indexing centers. Also, whenever class and center indices appear together, the first always refers to class.

Let the equilibrium probability of finding  $n_{ji}$  jobs of class  $j$  at center  $i$ ,  $1 \leq j \leq p$ ,  $1 \leq i \leq s$ , be  $\pi(\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_s)$ , where

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

Closed networks are characterized by conservation of jobs in each class. That is, the population of jobs of the  $j$ th class is constant at  $K_j$ , say. The well-known results on closed networks with the product form in its stationary distribution may be given in the following form:

$$\pi(\mathbf{y}_1, \dots, \mathbf{y}_s) = \frac{1}{G} \prod_{i=1}^s \pi_i(\mathbf{y}_i), \quad (2)$$

where

$$\begin{aligned} \pi_i(\mathbf{y}_i) &= (\sum n_{ji})! \prod \left( \frac{\rho_{ji}^{n_{ji}}}{n_{ji}!} \right), \quad 1 \leq i \leq q, \\ &= \prod \left( \frac{\rho_{ji}^{n_{ji}}}{n_{ji}!} \right), \quad (q + 1) \leq i \leq s. \end{aligned} \quad (3)$$

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 (3),

$$\rho_{ji} = \frac{\text{expected number of visits of class } j \text{ jobs to center } i}{\text{service rate of class } j \text{ jobs in center } i}, \quad (4)$$

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

In (2),  $G$  is the partition function, and it is explicitly

$$G(\mathbf{K}) = \sum_{\mathbf{1}'\mathbf{n}_1=K_1} \dots \sum_{\mathbf{1}'\mathbf{n}_p=K_p} \prod_{i=1}^s \pi_i(\mathbf{y}_i), \quad (5)$$

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,

$$G(\mathbf{K}) = \sum \dots \sum \left[ \prod_{i=1}^q \left\{ (\sum n_{ji})! \prod \left( \frac{\rho_{ji}^{n_{ji}}}{n_{ji}!} \right) \right\} \right] \left[ \prod_{i=q+1}^s \left\{ \prod \left( \frac{\rho_{ji}^{n_{ji}}}{n_{ji}!} \right) \right\} \right]. \quad (6)$$

In Section III, we will refer to this expression for the partition function.

## 2.2 Asymptotic expansions

A series

$$\sum_{k=0}^{\infty} A_k/N^k$$

is said to be an asymptotic expansion<sup>22-24</sup> of a function  $I(N)$  if

$$I(N) - \sum_{k=0}^{m-1} A_k/N^k = o(N^{-m}) \text{ as } N \rightarrow \infty \quad (7)$$

for every  $m = 1, 2, \dots$ . We write

$$I(N) \sim \sum_{k=0}^{\infty} A_k/N^k.$$

The series may be either convergent or divergent.

## III. INTEGRAL REPRESENTATIONS

As the representations presented here are basic to the subsequent development, we have allowed some duplication in Section 3.1 with Section 10 of Reference 21.

### 3.1 Partition function

We start with Euler's integral representation for the factorial,

$$n! = \int_0^{\infty} e^{-u} u^n du. \quad (8)$$

Returning to (6), we use this representation to write

$$(\sum n_{ji})! = \int_0^{\infty} e^{-u_i} \prod u_i^{n_{ji}} du_i, \quad i = 1, 2, \dots, q. \quad (9)$$

Substitution in (6) gives

$$G = \int_0^{\infty} \dots \int_0^{\infty} \exp\left(-\sum_{i=1}^q u_i\right) \sum_{1'n_1=K_1} \dots \sum_{1'n_p=K_p} \left[ \prod_{i=1}^q \left\{ \prod \frac{(\rho_{ji} u_i)^{n_{ji}}}{n_{ji}!} \right\} \right] \cdot \left[ \prod_{i=q+1}^s \left\{ \prod \frac{\rho_{ji}^{n_{ji}}}{n_{ji}!} \right\} \right] du_1 \dots du_q. \quad (10)$$

Now by the multinomial theorem,

$$G = (\prod K_j!)^{-1} \int_0^{\infty} \dots \int_0^{\infty} \exp\left(-\sum_{i=1}^q u_i\right) \cdot \prod \left\{ \sum_{i=1}^q \rho_{ji} u_i + \sum_{i=q+1}^s \rho_{ji} \right\}^{K_j} du_1 \dots du_q. \quad (11)$$

It is noteworthy but not surprising that the parameters  $\rho_{ji}$  for all the type 3 centers appear lumped together. Hence, we may simplify the notation by introducing  $\rho_{j0}$ , where

$$\rho_{j0} = \sum_{i=q+1}^s \rho_{ji}, \quad j = 1, 2, \dots, p. \quad (12)$$

Another consequence of the notation is that the center index  $i$  may henceforth be understood to range over the processing centers only, i.e.,  $1 \leq i \leq q$ .

The new quantity  $\rho_{j0}$  has the physical significance of being the weighted combination of all the mean think times of the IS centers in the routing of the  $j$ th class. In particular, if the routing of the  $j$ th class contains at least one IS center, then  $\rho_{j0} > 0$  and otherwise  $\rho_{j0} = 0$ . Let  $I$  be the collection of indices of classes of the former type and let  $I^*$  be the complementary collection, i.e.,

$$j \in I \Leftrightarrow \rho_{j0} > 0 \quad \text{and} \quad j \in I^* \Leftrightarrow \rho_{j0} = 0. \quad (13)$$

With this notation,

$$G = \left[ \prod_{j \in I} \rho_{j0}^{K_j} / \prod_j K_j! \right] \int_0^\infty \cdots \int_0^\infty \exp - (\sum u_i) \\ \times \prod_{j \in I} \left\{ 1 + \sum_i \frac{\rho_{ji}}{\rho_{j0}} u_i \right\}^{K_j} \prod_{j \in I^*} \left\{ \sum_i \rho_{ji} u_i \right\}^{K_j} du_1 \cdots du_q. \quad (14)$$

In vector notation, which we shall use widely, this reduces to

$$G = \left[ \prod_{j \in I} \rho_{j0}^{K_j} / \prod_j K_j! \right] \int_{Q^+} e^{-\mathbf{1}'\mathbf{u}} \prod_j (\delta_{jI} + \mathbf{r}'_j \mathbf{u})^{K_j} d\mathbf{u}, \quad (15)$$

where\*

$$\begin{aligned} \mathbf{u} &= (u_1, u_2, \dots, u_q)' \\ \mathbf{1} &= (1, 1, \dots, 1)' \\ \mathbf{r}_j &= (r_{j1}, r_{j2}, \dots, r_{jq})', \quad 1 \leq j \leq p \\ r_{ji} &= \rho_{ji}/\rho_{j0} \quad \text{if } j \in I \\ &= \rho_{ji} \quad \text{if } j \in I^* \\ \delta_{jI} &= 1 \quad \text{if } j \in I \\ &= 0 \quad \text{if } j \in I^* \\ Q^+ &= \{\mathbf{u} | \mathbf{u} \geq \mathbf{o}\}. \end{aligned} \quad (16)$$

\* Unfortunately,  $r_{ji}$  here is defined to be the reciprocal of the natural extension of  $r_j$  in Ref. 21.

We now introduce the large parameter  $N$  and define

$$\beta_j \triangleq K_j/N, \quad 1 \leq j \leq p, \quad (17)$$

$$\Gamma_j \triangleq Nr_j, \quad 1 \leq j \leq p. \quad (18)$$

The suggestion in the notation is that in the generic large network  $\{\beta_j\}$  and  $\{\Gamma_{ji}\}$  are  $O(1)$ . That is, the ratio of processing time to think time is, in order of magnitude estimation, proportionately less for increased populations. There is great latitude in the choice of the large parameter. The guiding principle in choosing it should be that the resulting values of  $\{\beta_j\}$  and  $\{\Gamma_{ji}\}$  are as uniformly close to 1 as possible. In practice, we have used

$$N = \max_{ij} \left\{ \frac{1}{r_{ji}} \right\}. \quad (19)$$

On substituting (17) and (18) into (15) and after the change of variables  $\mathbf{z} = \mathbf{u}/N$ , we obtain from (15) another useful integral representation of the partition function which is distinguished by its dependence on  $N$ . Summarizing for future reference, we have

*Proposition 1:*

$$G(\mathbf{K}) = \left[ \prod_{j \in I} \rho_{j0}^{K_j} / \prod_j K_j! \right] \int_{Q^+} e^{-\mathbf{1}'\mathbf{u}} \prod_j (\delta_{jI} + \mathbf{r}'_j \mathbf{u})^{K_j} d\mathbf{u} \quad (20)$$

$$= \left[ N^q \prod_{j \in I} \rho_{j0}^{K_j} / \prod_j K_j! \right] \int_{Q^+} e^{-Nf(\mathbf{z})} d\mathbf{z}, \quad (21)$$

where

$$f(\mathbf{z}) \triangleq \mathbf{1}'\mathbf{z} - \sum_{k=1}^p \beta_k \log(\delta_{kI} + \Gamma'_k \mathbf{z}). \quad (22)$$

□

### 3.2 Mean values

We restrict our attention to the mean value  $u_{oi}(\mathbf{K})$  which gives the utilization of the  $i$ th processor by jobs of the  $\sigma$ th class for a population distribution by class in the network denoted by  $\mathbf{K} = (K_1, K_2, \dots, K_p)$ . Other interesting mean performance indices, such as throughput and mean response time are known to be simply related to  $\{u_{oi}(\mathbf{K})\}$  and the interested reader may consult Ref. 17.

In Ref. 17,

$$u_{oi}(\mathbf{K}) = \rho_{oi} \frac{G(\mathbf{K} - \mathbf{e}_o)}{G(\mathbf{K})}, \quad 1 \leq \sigma \leq p, 1 \leq i \leq q, \quad (23)$$

where  $\mathbf{e}_o$  is our notation for the vector with the  $\sigma$ th component unity and all other components zero. Thus, the value for the partition

function is needed for the given population distribution and also for the population in the  $\sigma$ th class reduced by 1.

Now from (20)

$$G(\mathbf{K} + \mathbf{e}_\sigma) = \frac{\rho_{\sigma 0} \prod_{j \in I} \rho_{j 0}^{K_j}}{(K_\sigma + 1) \prod K_j!} \int_{Q^+} e^{-1\mathbf{u}} (1 + \mathbf{r}'_\sigma \mathbf{u}) \times \prod_j (\delta_{jI} + \mathbf{r}'_j \mathbf{u})^{K_j} d\mathbf{u} \quad \text{if } \sigma \in I \quad (24)$$

$$= \frac{\prod_{j \in I} \rho_{j 0}^{K_j}}{(K_\sigma + 1) \prod K_j!} \int_{Q^+} e^{-1\mathbf{u}} (\mathbf{r}'_\sigma \mathbf{u}) \times \prod_j (\delta_{jI} + \mathbf{r}'_j \mathbf{u})^{K_j} d\mathbf{u} \quad \text{if } \sigma \in I^*. \quad (25)$$

From (23) to (25) and the same change of variables, namely  $\mathbf{z} = \mathbf{u}/N$ , employed in transforming (20) to (21) we obtain the following representation of the utilization in terms of integrals,

*Proposition 2:* For class index  $\sigma$ ,  $1 \leq \sigma \leq p$ , and center index  $i$ ,  $1 \leq i \leq q$ ,

$$u_{\sigma i}(\mathbf{K} + \mathbf{e}_\sigma)^{-1} = \left\{ \frac{1}{r_{\sigma i}(K_\sigma + 1)} \right\} \left[ \delta_{\sigma I} + \frac{\int_{Q^+} (\mathbf{\Gamma}'_\sigma \mathbf{z}) e^{-Nf(\mathbf{z})} d\mathbf{z}}{\int_{Q^+} e^{-Nf(\mathbf{z})} d\mathbf{z}} \right]. \quad (26)$$

□

Some digressory comments are as follows. Note that the above conceals that  $r_{\sigma i}$  is normalized differently, but not unexpectedly, depending on whether  $\sigma \in I$  or otherwise [see (16)]. Also note that in the typical large network for normal operating conditions we always expect  $r_{\sigma i}$  to be  $0(1/N)$ , precisely because of the normalization used, and  $K_\sigma$  to be  $0(N)$  so that the term in braces in (26) is then  $0(1)$ .

#### IV. ASYMPTOTIC EXPANSIONS

We henceforth consider only networks in which the route for each class always contains an infinite server center. Specifically,

$$\rho_{j0} > 0, \quad j = 1, 2, \dots, p, \quad (27)$$

and the set  $I^*$  is empty.

##### 4.1 The assumption of "normal usage"

Define

$$\alpha \triangleq \mathbf{1} - \sum \beta_j \mathbf{\Gamma}_j \quad (28)$$

so that in terms of the original network parameters

$$\alpha_i = 1 - \sum K_j \frac{\rho_{ji}}{\rho_{j0}}, \quad i = 1, 2, \dots, q. \quad (29)$$

It is important to note that  $\alpha$  is independent of the choice of  $N$ . The parameter  $\alpha_i$  ( $-\infty < \alpha_i < 1$ ) is an indicator of the unutilized processing capability of the  $i$ th center. Positive values of  $\alpha_i$  correspond to less than 100 percent utilization of the processor and negative values which, of course, can occur to very high utilizations.\* Normal usage in large networks will almost certainly require  $\alpha_i > 0$ , and in all likelihood  $\alpha_i$  will not be close to 0 for all  $i$ .

Hereafter, we assume

$$\alpha_i > 0, \quad i = 1, 2, \dots, q, \quad (30)$$

which condition we refer to as normal usage. Moreover, as  $\alpha_i$  for some  $i$  comes close to 0, the expansions given here are not as efficient as those derived specifically for such conditions and which we propose to give in the future.

A justification of the usage interpretation that we have given to  $\alpha$  is provided by a result obtained later (see below Proposition 6) which states that, asymptotic with network size,

$$u_i = \text{utilization of } i\text{th processor} \sim 1 - \alpha_i. \quad (31)$$

An obvious caveat is that this result is derived for the assumption in (30). However, as the utilization in (31) can come close to unity even while (30) is satisfied, (31) suggests that for large networks normal usage will not extend beyond the range  $\alpha > 0$ .

Observe that

$$f(\mathbf{0}) = 0 \quad (32)$$

$$\text{and } \nabla f(\mathbf{z}) = \mathbf{1} - \sum_j \frac{\beta_j}{1 + \Gamma'_j \mathbf{z}} \Gamma_j \quad (33)$$

so that

$$\alpha = \nabla f(\mathbf{0}). \quad (34)$$

The assumption of  $\alpha > 0$  and the form in (33) ensures that the function  $f$  has no stationary points in  $Q^+$  since

$$\nabla f(\mathbf{z}) \geq \nabla f(\mathbf{0}) > \mathbf{0}, \quad \mathbf{z} \in Q^+. \quad (35)$$

Also observe that

---

\* Unfortunately,  $\alpha$  here has an opposite sign from the natural extension of  $\alpha$  as defined in Ref. 21.

$$\left\{ \frac{\partial^2 f}{\partial z_{i_1} \partial z_{i_2}} \right\} = \sum \frac{\beta_j}{(1 + \Gamma_j' \mathbf{z})^2} \Gamma_j \Gamma_j' \quad (36)$$

from which we note that the Hessian is positive semidefinite.

To conclude, with normal usage,  $f$  is a convex function, with its minimum in  $Q^+$  attained at 0 and with no point in  $Q^+$  where its gradient vanishes.

#### 4.2 Transformations on integrals exploiting normal usage

Consider the following transformations on the basic integral:

$$\begin{aligned} \int_{Q^+} e^{-Nf(\mathbf{z})} d\mathbf{z} &= \int_{Q^+} e^{-Nf(\mathbf{z}) + N \sum \beta_j (\Gamma_j' \mathbf{z}) - N \sum \beta_j (\Gamma_j' \mathbf{z})} d\mathbf{z} \\ &= \int_{Q^+} e^{-N\alpha' \mathbf{z}} \exp \\ &\quad - \left[ N \sum_j \beta_j \{ \Gamma_j' \mathbf{z} - \log(1 + \Gamma_j' \mathbf{z}) \} \right] d\mathbf{z} \\ &= N^{-q} \int_{Q^+} e^{-\alpha' \mathbf{u}} \exp \\ &\quad - \left[ \sum_j \beta_j \left\{ \Gamma_j' \mathbf{u} - N \log \left( 1 + \frac{1}{N} \Gamma_j' \mathbf{u} \right) \right\} \right] d\mathbf{u}, \quad (37) \end{aligned}$$

where  $\mathbf{u} = N\mathbf{z}$ . Now make the following change of variables,

$$v_i \triangleq \alpha_i u_i, \quad 1 \leq i \leq q \quad (38)$$

and normalize the system parameters with respect to  $\alpha$ , thus,

$$\tilde{\Gamma}_{ji} \triangleq \Gamma_{ji} / \alpha_i, \quad 1 \leq j \leq p, 1 \leq i \leq q. \quad (39)$$

Observe that in particular

$$\Gamma_j' \mathbf{u} = \tilde{\Gamma}_j' \mathbf{v}. \quad (40)$$

From (37),

$$\int_{Q^+} e^{-Nf(\mathbf{z})} d\mathbf{z} = \frac{N^{-q}}{(\prod \alpha_i)} \int_{Q^+} e^{-\mathbf{1}' \mathbf{v}} H(N^{-1}, \mathbf{v}) d\mathbf{v}, \quad (41)$$

where

$$H(N^{-1}, \mathbf{v}) \triangleq e^{s(N^{-1}, \mathbf{v})}, \quad (42)$$

$$s(N^{-1}, \mathbf{v}) \triangleq - \sum_{j=1}^p \beta_j \left\{ \tilde{\Gamma}_j' \mathbf{v} - N \log \left( 1 + \frac{1}{N} \tilde{\Gamma}_j' \mathbf{v} \right) \right\}. \quad (43)$$

Our notation here suggesting  $N^{-1}$  as the independent variable may be

perplexing at this stage, but it provides a clue to the direction of the analysis.

Here are two further digressionary comments: the above transformations are meaningful only in the context of normal usage, i.e.,  $\alpha > \mathbf{0}$ . In a similar vein, an interpretation of  $\{\tilde{\Gamma}_{ji}\}$  as renormalized  $\{\Gamma_{ji}\}$  is only meaningful for  $\alpha > \mathbf{0}$ . It is noteworthy that hereafter we shall be dealing exclusively with  $\{\tilde{\Gamma}_{ji}\}$  and not at all with  $\{\Gamma_{ji}\}$ .

We need to repeat the transformations given above for the integral  $\int (\Gamma'_\sigma \mathbf{z})^m e^{-Nf(\mathbf{z})} d\mathbf{z}$ . The result may be combined with (42) and (43) to give the following compact representation:

$$\int_{Q^+} (\Gamma'_\sigma \mathbf{z})^m e^{-Nf(\mathbf{z})} d\mathbf{z} = \frac{N^{-(q+m)}}{(\prod \alpha_i)} \int_{Q^+} e^{-1'v} (\tilde{\Gamma}'_\sigma v)^m H(N^{-1}, v) dv, \quad (44)$$

$$m = 0, 1, 2, \dots \quad \square$$

We may now use these expressions in Proposition 2 to obtain for  $1 \leq \sigma \leq p$  and  $1 \leq i \leq q$ ,

$$u_{oi}(\mathbf{K} + \mathbf{e}_\sigma)^{-1} = \left\{ \frac{\rho_{\sigma 0}}{\rho_{\sigma i}(K_\sigma + 1)} \right\} \cdot \left[ 1 + \frac{1}{N} \frac{\int_{Q^+} e^{-1'v} (\tilde{\Gamma}'_\sigma v) H(N^{-1}, v) dv}{\int_{Q^+} e^{-1'v} H(N^{-1}, v) dv} \right]. \quad (45)$$

Note that the bracketed term is independent of the processing center index  $i$ .

Let us agree to call

$$I(N) \triangleq \int_{Q^+} e^{-1'v} H(N^{-1}, v) dv \quad (46)$$

and

$$I_\sigma^{(1)}(N) \triangleq \int_{Q^+} e^{-1'v} (\tilde{\Gamma}'_\sigma v) H(N^{-1}, v) dv, \quad (47)$$

where the superscript (1) is a mnemonic for first moment. In this notation, we have for future reference

*Proposition 3:*

$$u_{oi}(\mathbf{K} + \mathbf{e}_\sigma)^{-1} = \left\{ \frac{\rho_{\sigma 0}}{\rho_{\sigma i}(K_\sigma + 1)} \right\} \left[ 1 + \frac{1}{N} \frac{I_\sigma^{(1)}(N)}{I(N)} \right]. \quad (48)$$

□

The asymptotic expansions of  $I(N)$  and  $I_\sigma^{(1)}(N)$  are considered next.

### 4.3 Asymptotic expansions

This section will outline the procedure for obtaining the asymptotic expansions together with plausibility arguments. The proofs of the assertions concerning asymptoticity will follow in Section V. Moreover, we defer till Section 4.4 certain observations which make feasible the efficient computation of the coefficients of the asymptotic expansions.

Our procedure for  $I(N)$  is to first obtain a power series in  $N^{-1}$  of  $H(N^{-1}, \mathbf{v})$  and then to integrate term by term. Thus, we let

$$H(N^{-1}, \mathbf{v}) = \sum_{k=0}^{\infty} \frac{h_k(\mathbf{v})}{N^k} \quad (49)$$

and

$$A_k \triangleq \int_{Q^+} e^{-1^{\mathbf{v}}} h_k(\mathbf{v}) d\mathbf{v} \quad (50)$$

and claim that

$$I(N) \sim \sum_{k=0}^{\infty} \frac{A_k}{N^k}. \quad (51)$$

Let us elaborate on the coefficients  $\{h_k(\mathbf{v})\}$  in (49).

$$h_k(\mathbf{v}) = \frac{1}{k!} \frac{\partial^k}{\partial(1/N)^k} H(0, \mathbf{v}), \quad k = 0, 1, 2, \dots \quad (52)$$

To make these explicit, we need to first introduce

$$f_k(\mathbf{v}) \triangleq \frac{(-1)^k}{k} \sum_j \beta_j (\tilde{\Gamma}_j \mathbf{v})^k, \quad k = 1, 2, \dots \quad (53)$$

Their role becomes clear if we recall (42):

$$H(N^{-1}, \mathbf{v}) = e^{s(N^{-1}, \mathbf{v})}, \quad (54)$$

and note from (43) that for fixed  $\mathbf{v} \in Q^+$ ,  $s(N^{-1}, \mathbf{v})$  and, hence,  $H(N^{-1}, \mathbf{v})$  are functions of  $N^{-1}$ , analytic in  $\text{Re}(N^{-1}) > \epsilon(\mathbf{v})$ , where  $\epsilon(\mathbf{v}) < 0$ . Then,

$$s^{(k)}(0, \mathbf{v}) = -k! f_{k+1}(\mathbf{v}), \quad k = 1, 2, \dots \quad (55)$$

To proceed now to the derivatives of  $H(\cdot, \mathbf{v})$  itself, we will find useful the following expression in which it is understood that all derivatives are with respect to  $N^{-1}$ :

$$H^{(k+1)}(N^{-1}, \mathbf{v}) = \sum_{m=0}^k \binom{k}{m} s^{(k+1-m)}(N^{-1}, \mathbf{v}) H^{(m)}(N^{-1}, \mathbf{v}),$$

$$k = 0, 1, \dots \quad (56)$$

From (52), (55) and the above, it is easy to see that there exists the following recursive scheme for generating  $\{h_k(\mathbf{v})\}$ :

$$\left. \begin{aligned} h_0(\mathbf{v}) &\equiv 1 \\ h_{k+1}(\mathbf{v}) &= -\frac{1}{k+1} \sum_{m=0}^k (k+1-m) f_{k+2-m}(\mathbf{v}) h_m(\mathbf{v}), \\ &\cdot k = 0, 1, 2, \dots \end{aligned} \right\} \quad (57)$$

In particular, the leading elements are

$$\begin{aligned} h_k(\mathbf{v}) &\equiv 1, & k = 0 \\ &= -f_2(\mathbf{v}), & k = 1 \\ &= -f_3(\mathbf{v}) + \frac{1}{2} f_2^2(\mathbf{v}), & k = 2 \\ &= -f_4(\mathbf{v}) + f_2(\mathbf{v}) f_3(\mathbf{v}) - \frac{1}{6} f_2^3(\mathbf{v}), & k = 3. \end{aligned} \quad (58)$$

To summarize the steps discussed so far in the generation of the asymptotic expansion, we have

*Proposition 4:*

$$I(N) \sim \sum_{k=0}^{\infty} \frac{A_k}{N^k}, \quad (59)$$

where  $A_k = \int e^{-1\mathbf{v}} h_k(\mathbf{v}) d\mathbf{v}$ ,  $\{h_k(\mathbf{v})\}$  is obtained recursively from (57) with leading elements exhibited in (58), and  $\{f_k(\mathbf{v})\}$  is as in (53).  $\square$

The aspect of the above asymptotic expansion of the integral  $I(N)$ , which consists of decomposing the integrand into the product of an exponential and a function, the expansion of the latter function in a power series, and the final term-by-term integration is like the procedure which, in the context of single integrals, is justified by Watson's Lemma<sup>23</sup> under certain conditions. Our contribution has been to show that a generalization of this fundamental result exists for the multiple integrals of interest in stochastic networks.

Our procedure for obtaining the asymptotic expansion for  $I_{\sigma}^{(1)}(N)$  is very similar and consists of obtaining a power series in  $N^{-1}$  of  $(\tilde{\Gamma}_{\sigma}^{\mathbf{v}} \mathbf{v}) H(N^{-1}, \mathbf{v})$  and integrating term by term. However, we notice the simplifying fact that

$$(\tilde{\Gamma}_{\sigma}^{\mathbf{v}} \mathbf{v}) H(N^{-1}, \mathbf{v}) = \sum_{k=0}^{\infty} \frac{(\tilde{\Gamma}_{\sigma}^{\mathbf{v}} \mathbf{v}) h_k(\mathbf{v})}{N^k}. \quad (60)$$

Thus, the procedure for this integral is as follows:

*Proposition 5:*

$$I_{\sigma}^{(1)}(N) \sim \sum_{k=0}^{\infty} \frac{A_{\sigma,k}^{(1)}}{N^k}, \quad (61)$$

where

$$A_{\sigma,k}^{(1)} \triangleq \int_{Q^+} e^{-1^v} (\tilde{\Gamma}_{\sigma}^v \mathbf{v}) h_k(\mathbf{v}) d\mathbf{v}, \quad k = 0, 1, 2, \dots \quad (62)$$

□

As observed in Ref. 21, the asymptotic expansion for the integrals may be used to generate asymptotic expansions for their ratios on account of powers of  $N^{-1}$  forming a multiplicative sequence.<sup>22</sup> Thus, the coefficients of an asymptotic expansion for  $u_{oi}(\mathbf{K} + \mathbf{e}_{\sigma})^{-1}$  may be obtained from formal substitution in Proposition 2 of the expansions in Propositions 4 and 5. This gives

*Proposition 6:*

$$\left\{ \frac{\rho_{oi}(K_{\sigma} + 1)}{\rho_{\sigma 0}} \right\} u_{oi}(\mathbf{K} + \mathbf{e}_{\sigma})^{-1} \sim 1 + \frac{1}{N} \sum_{k=0}^{\infty} \frac{B_{\sigma,k}}{N^k}, \quad (63)$$

$$B_{\sigma,k} = \frac{A_{\sigma,k}^{(1)}}{A_0}, \quad k = 0 \quad (64)$$

$$= \frac{1}{A_0} \left[ A_{\sigma,k}^{(1)} - \sum_{m=1}^k A_m B_{\sigma,k-m} \right], \quad k = 1, 2, \dots \quad \square$$

With the above proposition we may generate  $(k + 1)$  terms of the expansion for  $u_{oi}$  from  $k$  terms of  $I(N)$  and  $I_{\sigma}^{(1)}(N)$ .

An immediate corollary to the above proposition is

$$u_{oi}(\mathbf{K}) \sim \frac{\rho_{oi} K_{\sigma}}{\rho_{\sigma 0}}$$

and

$$u_i(\mathbf{K}) = \text{utilization of } i\text{th processor} = \sum_j u_{ji}(\mathbf{K}) \\ \sim 1 - \alpha_i, \quad i = 1, 2, \dots, q, \quad (65)$$

which was claimed earlier in Section 4.1 in the course of giving physical meaning to the parameters  $\{\alpha_i\}$ .

This corollary illustrates the important point that the terms in the asymptotic expansions  $A_k/N^k$ ,  $A_{\sigma,k}^{(1)}/N^k$ , and  $B_{\sigma,k}/N^{k+1}$  are all independent of  $N$  and depend only on the network parameters. The dummy parameter  $N$  serves to show how to group the terms of the same magnitude. This independence from  $N$  follows from the fact that  $\alpha$  is independent of  $N$ , as noted earlier, and from the fact  $f_k(v)/N^{k-1}$  is easily seen to be independent of  $N$ . Once this is noted, the result

follows easily from the definitions of the terms. The choice of  $N$  is important numerically, which will be discussed in a subsequent paper.

#### 4.4 Pseudonetworks and the computation of expansion coefficients

Here, we consider the compositions of coefficients,  $\{A_k\}$  and  $\{A_{\sigma,k}^{(1)}\}$ , and find that quite remarkably they are related intimately to the partition function of a certain hypothetical network which we call the pseudonetwork. It turns out that to compute the leading elements of  $\{A_k\}$  and  $\{A_{\sigma,k}^{(1)}\}$ , we need to consider the pseudonetwork with only small populations. Thus, existing techniques known to be effective for computing partition functions for small populations may be used to compute the leading coefficients of our asymptotic expansions.

An example will prove useful. From Proposition 4 and eq. (58) we see that

$$A_3 = \int e^{-1\mathbf{v}} \left\{ -f_4(\mathbf{v}) + f_2(\mathbf{v})f_3(\mathbf{v}) - \frac{1}{6}f_2^3(\mathbf{v}) \right\} d\mathbf{v}. \quad (66)$$

Now consider only the third term after denoting it by  $A_{33}$ . Using the expression for  $f_2(\mathbf{v})$  as given by (53), we obtain

$$\begin{aligned} A_{33} = & -\frac{1}{48} \sum_j \beta_j^3 \int e^{-1\mathbf{v}} (\tilde{\Gamma}_j^{\mathbf{v}})^6 d\mathbf{v} \\ & -\frac{1}{16} \sum_{j \neq k} \beta_j^2 \beta_k \int e^{-1\mathbf{v}} (\tilde{\Gamma}_j^{\mathbf{v}})^4 (\tilde{\Gamma}_k^{\mathbf{v}})^2 d\mathbf{v} \\ & -\frac{1}{48} \sum_{j \neq k \neq l \neq j} \beta_j \beta_k \beta_l \int e^{-1\mathbf{v}} (\tilde{\Gamma}_j^{\mathbf{v}})^2 (\tilde{\Gamma}_k^{\mathbf{v}})^2 (\tilde{\Gamma}_l^{\mathbf{v}})^2 d\mathbf{v}, \quad (67) \end{aligned}$$

where the subscripts  $j$ ,  $k$ , and  $l$  are class indices ranging over  $[1, p]$ . We now make the observation that the generic integral in the composition of the asymptotic expansion coefficients is within a multiplicative constant of

$$g(\mathbf{m}) = g(m_1, m_2, \dots, m_p) \triangleq \frac{1}{\left(\prod_j m_j!\right)} \int_{Q^+} e^{-1\mathbf{u}} \prod_j (\tilde{\Gamma}_j^{\mathbf{u}})^{m_j} d\mathbf{u}. \quad (68)$$

The above is an important form for we may now identify it with quantities previously encountered.

We first give the following equivalent expression for  $g(\mathbf{m})$ .

$$g(\mathbf{m}) = \sum_{1^n \mathbf{1} = m_1} \dots \sum_{1^n \mathbf{1} = m_p} \prod_{i=1}^q \left\{ \left( \sum_j n_{ji} \right)! \prod_j \frac{\tilde{\Gamma}_j^{n_{ji}}}{n_{ji}!} \right\}. \quad (68')$$

Recall the expression in Proposition 1, eq. (20), for the integral repre-

sentation of the partition function. Specialize the expression there to a network with no infinite server centers, i.e., set  $I$  is empty, and find that it reduces to

$$G(\mathbf{K}) = \frac{1}{\left(\prod_j K_j!\right)} \int_{Q^+} e^{-\mathbf{1}\cdot\mathbf{u}} \prod_j (\mathbf{r}'\mathbf{u})^{K_j} d\mathbf{u}. \quad (69)$$

On comparing (68) and (69), or (68') and (6), we may conclude that  $g(\mathbf{m})$  is the partition function of a certain network.

Call this hypothetical network the pseudonetwork. What characterizes the pseudonetwork? To begin with, it is closed and lacks its centers. There are, as in the original network, exactly  $q$  processing centers and  $p$  classes of jobs. The processing rate of jobs from the  $j$ th class in the  $i$ th center of the pseudonetwork is  $\tilde{\Gamma}_{ji}$ , where, you will recall,  $\tilde{\Gamma}_{ji} = \Gamma_{ji}/\alpha_i$ . In agreement with past convention,  $(m_1, m_2, \dots, m_p)'$  denotes in vector form the population distribution by class in the network.

We may follow the procedure outlined in the example concerning  $A_{33}$  above to express all the leading coefficients  $A_k$ ,  $k = 0, 1, 2, 3$  in terms of the partition function of the pseudonetwork. This gives

$$\begin{aligned} A_0 &= 1 \\ A_1 &= -\sum_j \beta_j g(2\mathbf{e}_j) \\ A_2 &= 2 \sum_j \beta_j g(3\mathbf{e}_j) + 3 \sum_j \beta_j^2 g(4\mathbf{e}_j) + \frac{1}{2} \sum_{j \neq k} \beta_j \beta_k g(2\mathbf{e}_j + 2\mathbf{e}_k) \\ A_3 &= -6 \sum_j \beta_j g(4\mathbf{e}_j) - 20 \sum_j \beta_j^2 g(5\mathbf{e}_j) - 15 \sum_j \beta_j^3 g(6\mathbf{e}_j) \\ &\quad - 2 \sum_{j \neq k} \beta_j \beta_k g(2\mathbf{e}_j + 3\mathbf{e}_k) \\ &\quad - 3 \sum_{j \neq k} \beta_j^2 \beta_k g(4\mathbf{e}_j + 2\mathbf{e}_k) \\ &\quad - \frac{1}{6} \sum_{j \neq k \neq l \neq j} \beta_j \beta_k \beta_l g(2\mathbf{e}_j + 2\mathbf{e}_k + 2\mathbf{e}_l). \end{aligned} \quad (70)$$

In these expressions,  $j$ ,  $k$ , and  $l$  are class indices each with range  $[1, p]$ .

Notice that in the computation of  $A_k$ ,  $k = 0, 1, 2, 3$ , the population distribution  $\mathbf{m}$  that appears in  $g(\mathbf{m})$  may be characterized as being quite small. The total population in the pseudonetwork over all classes,  $\sum m_j$ , is at most 6. A further simplifying condition is that we need consider only distributions with three classes at most with nonzero populations—the extreme distribution arises in  $g(2\mathbf{e}_j + 2\mathbf{e}_k + 2\mathbf{e}_l)$ .

Since we are interested in population distributions where all but a

small number of classes have no members at all, we may equivalently choose to view the pseudonetwork as a collection of smaller networks each with a small (less than  $p$ ) number of classes of jobs. For example,  $g(2e_j + 3e_k)$  may be viewed either as the partition function of the pseudonetwork in which all but classes  $j$  and  $k$  have zero population, or as originating from a network with only two classes with population distribution (2, 3) but one which is otherwise unchanged. Such distinctions, while not material to the procedures given here, may be consequential in the efficiency of the computations.

The coefficients  $\{A_{\sigma,k}^{(1)}\}$  of the expansion for  $I_{\sigma}^{(1)}(N)$  may also be expressed in terms of the partition function of the pseudonetwork. As the derivation is similar, it will suffice to give the results, which we do in the Appendix.

It will be observed that to compute  $\{A_{\sigma,k}^{(1)}\}$ ,  $k = 0, 1, 2, 3$  we need to consider various allocations to classes of a total population in the pseudonetwork of at most 7. Thus, to compute the leading five terms of the utilization  $u_{\sigma i}$ , we need  $A_k$  and  $\{A_{\sigma,k}^{(1)}\}$  for  $k = 0, 1, 2, 3$ , and these are computed from the values of the partition function of the pseudonetwork for various allocations to classes of a total network population of at most 7. This is an elaboration of a claim made in the Introduction.

## V. ERROR ANALYSIS AND PROOF OF ASYMPTOTICITY

Equations (59) and (61) contain claims requiring proof on the asymptoticity of the expansions of  $I(N)$  and  $I_{\sigma}^{(1)}(N)$ . This is provided here as a corollary to a complete error analysis. In fact we show that the expansions given earlier have properties more attractive than that required of asymptotic expansions. For instance, asymptoticity requires that errors incurred in the estimation of the integrals from the use of, say,  $m$  leading terms is of the same *order* as the  $(m + 1)$ th term as  $N \rightarrow \infty$ . We prove that the expansions derived have the stronger property that the truncation error is bounded by the  $(m + 1)$ th term. The practical benefit of this analysis is that with very little incremental effort we can accompany our estimates of the mean values with sharp estimates of the estimation error.

### 5.1 Completely monotonic functions <sup>24</sup>

We need the following definition: for any nonnegative  $R$ , let  $Q^+(R)$  be the set of nonnegative vectors with norm bounded by  $R$ , i.e.,  $Q^+(R) = \{\mathbf{v} | \mathbf{v} \geq 0 \text{ and } \|\mathbf{v}\| \leq R\}$ . Note that  $Q^+(R) \rightarrow Q^+$  as  $R \rightarrow \infty$ .

The proposition below (cf. Ref. 21) states a remarkable property of the function  $H(N^{-1}, \mathbf{v})$  which is a key to much of the error analysis.

*Proposition 7:* For all  $\mathbf{v} \in Q^+(R)$ ,  $R < \infty$ ,  $H(N^{-1}, \mathbf{v})$  is a completely monotonic (or alternating) function of  $N^{-1}$ . That is,

$$(-1)^k \frac{\partial^k}{\partial (1/N)^k} H(N^{-1}, \mathbf{v}) \geq 0 \quad \text{for } k = 0, 1, 2, \dots \quad \text{and } 0 \leq N^{-1} < \infty. \quad (71)$$

*Proof:* Consider the form for  $H(N^{-1}, \mathbf{v})$  from (42 to 43), Section 4.2:

$$H(N^{-1}, \mathbf{v}) = \prod_j \left\{ e^{-\beta_j (\tilde{\Gamma}'_j \mathbf{v})} \left( 1 + \frac{1}{N} \tilde{\Gamma}'_j \mathbf{v} \right)^{\beta_j N} \right\}. \quad (72)$$

Because products of completely monotonic functions are also completely monotonic, it suffices to show that

$$\left( 1 + \frac{1}{N} \tilde{\Gamma}'_j \mathbf{v} \right)^{\beta_j N}$$

is a completely monotonic function of  $N^{-1}$ . Let us write

$$\left( 1 + \frac{1}{N} \tilde{\Gamma}'_j \mathbf{v} \right)^{\beta_j N} = e^{t(w)}, \quad (73)$$

where

$$t(w) = \frac{\beta_j}{w} \log(1 + aw), \quad (74)$$

by identifying  $w = 1/N$  and  $a = \tilde{\Gamma}'_j \mathbf{v}$ . We note that  $0 \leq w < \infty$  and that  $a$  is nonnegative and bounded, with the latter property being ensured by the restriction of  $\mathbf{v}$  to  $Q^+(R)$ ,  $R < \infty$ . Thus, all derivatives of  $\log(1 + aw)$ , and consequently of  $t(w)$ , exist and are continuous for  $0 \leq w < \infty$ . Since

$$\frac{d^{k+1} \{e^{t(w)}\}}{dw^{k+1}} = \sum_{m=0}^k \binom{k}{m} \left\{ \frac{d^{k+1-m}}{dw^{k+1-m}} t(w) \right\} \left[ \frac{d^m}{dw^m} \{e^{t(w)}\} \right], \quad (75)$$

we may conclude from a simple inductive argument that

$$\begin{aligned} \text{if } t(w) \text{ is a completely monotonic function of } w, \\ \text{then } e^{t(w)} \text{ is a completely monotonic function of } w. \end{aligned} \quad (76)$$

Finally, to show that  $t(w)$  is a completely monotonic function of  $w$  is to show that  $\{\log(1 + x)\}/x$  is completely monotonic. This is true, but we omit the proof.  $\square$

We need the following analogous property in connection with the integrand of  $I_\sigma^{(1)}(N)$ :

$$\left. \begin{aligned} \text{For all } \mathbf{v} \in Q^+(R), R < \infty, (\tilde{\Gamma}'_\sigma \mathbf{v}) H(N^{-1}, \mathbf{v}) \text{ is completely} \\ \text{monotonic in } N^{-1}. \end{aligned} \right\} \quad (77)$$

The proof is immediate from the preceding proposition since the additional factor  $(\tilde{\Gamma}'_\sigma \mathbf{v})$  does not depend on  $N^{-1}$ .

## 5.2 Error bounds

*Proposition 8:* For all  $N$ ,  $0 < N \leq \infty$

$$\begin{aligned} \frac{A_m}{N^m} < I(N) - \sum_{k=0}^{m-1} \frac{A_k}{N^k} < 0, \quad m = 1, 3, 5, \dots \\ 0 < I(N) - \sum_{k=0}^{m-1} \frac{A_k}{N^k} < \frac{A_m}{N^m}, \quad m = 2, 4, 6, \dots \end{aligned} \quad (78)$$

*Proof:* We initially require  $\mathbf{v} \in Q^+(R)$ ,  $R < \infty$  so that the preceding proposition is applicable. Viewing  $H(N^{-1}, \mathbf{v})$  as a function of  $N^{-1}$ , we may use a version of Taylor's theorem<sup>24</sup> that is accompanied by an estimate of the truncation error for the series to obtain

$$H(N^{-1}, \mathbf{v}) = \sum_{k=0}^{m-1} \frac{h_k(\mathbf{v})}{N^k} + \frac{1}{N^m} \frac{1}{m!} \frac{\partial^m}{\partial (1/N)^m} H(\xi, \mathbf{v}), \quad (79)$$

where

$$\xi \in [0, N^{-1}].$$

Consider first the case of  $m$  odd. From the preceding proposition, the  $m$ th derivative of  $H(N^{-1}, \mathbf{v})$  is a nonpositive and monotonically nondecreasing function of  $N^{-1}$ . Hence,

$$\frac{\partial^m}{\partial (1/N)^m} H(0, \mathbf{v}) \leq \frac{\partial^m}{\partial (1/N)^m} H(\xi, \mathbf{v}) \leq 0, \quad \xi \geq 0. \quad (80)$$

Substituting in (79),

$$\frac{h_m(\mathbf{v})}{N^m} \leq H(N^{-1}, \mathbf{v}) - \sum_{k=0}^{m-1} \frac{h_k(\mathbf{v})}{N^k} \leq 0. \quad (81)$$

Hence,

$$\begin{aligned} \int_{Q^+(R)} e^{-1^v} H(N^{-1}, \mathbf{v}) d\mathbf{v} - \sum_{k=0}^{m-1} \frac{1}{N^k} \int_{Q^+(R)} e^{-1^v} h_k(\mathbf{v}) d\mathbf{v} \\ \leq 0 \\ \geq \frac{1}{N^m} \int_{Q^+(R)} e^{-1^v} h_m(\mathbf{v}) d\mathbf{v}. \end{aligned} \quad (82)$$

The pair of bounds holds uniformly in  $R$ . Consequently, we may let  $R \rightarrow \infty$  and drop the restriction on  $R$  to obtain (78).

The proof for  $m$  even is very similar with the starting point being the following replacement for (80),

$$0 \leq \frac{\partial^m}{\partial (1/N)^m} H(\xi, \mathbf{v}) \leq \frac{\partial^m}{\partial (1/N)^m} H(0, \mathbf{v}). \quad (83)$$

□

The rest of the proof is omitted.

The above proposition states that the error incurred from using only a certain number of leading terms of the expansion for  $I(N)$  is numerically less than the first neglected term of the series and has the same sign.

Another implication that can be quite useful in practice is that the estimate with an odd number of terms is an upper bound on the integral and an even number of terms gives a lower bound. Thus, the error sequence alternates in sign. (It is also true but less consequential that the terms of the expansion also alternate in sign.)

In particular the above proposition proves the asymptoticity of the expansion in Section 4.3.

By a matching argument and with recourse to the complete monotonicity of  $(\mathbf{\Gamma}'_\sigma \mathbf{v})H(N^{-1}, \mathbf{v})$ , see (77), we also have

*Proposition 9:* For any class index  $\sigma$  and all  $N$ ,  $0 \leq N \leq \infty$ ,

$$\begin{aligned} \frac{A_{\sigma,m}^{(1)}}{N^m} &\leq I_\sigma^{(1)}(N) - \sum_{k=0}^{m-1} \frac{A_{\sigma,k}^{(1)}}{N^k} \leq 0, \quad m = 1, 3, 5, \dots \\ 0 &\leq I_\sigma^{(1)}(N) - \sum_{k=0}^{m-1} \frac{A_{\sigma,k}^{(1)}}{N^k} \leq \frac{A_{\sigma,m}^{(1)}}{N^m}, \quad m = 2, 4, 6, \dots \end{aligned} \quad (84)$$

With error estimates available for both  $I(N)$  and  $I_\sigma^{(1)}(N)$ , it is straightforward to use these to obtain an error estimate for the mean value given in Proposition 3, eq. (48).

## APPENDIX A

### *The Coefficients $\{A_{\alpha k}^{(l)}\}$ in Terms of the Partition Function of the Pseudo-Network*

Here  $\sigma$  is a given fixed class index, while  $j$ ,  $k$ , and  $l$  are also class indices each ranging over  $[1, p]$ . It is also understood that  $j$ ,  $k$ ,  $l$ , and  $\sigma$  are all distinct.

$$A_{\sigma,0}^{(1)} = g(\mathbf{e}_\sigma)$$

$$A_{\sigma,1}^{(1)} = -3\beta_\sigma g(3\mathbf{e}_\sigma) - \sum_j \beta_j g(\mathbf{e}_\sigma + 2\mathbf{e}_j)$$

$$A_{\sigma,2}^{(1)} = 8\beta_\sigma g(4\mathbf{e}_\sigma) + 15\beta_\sigma^2 g(5\mathbf{e}_\sigma)$$

$$+ \sum_j [2\beta_j g(3\mathbf{e}_j + \mathbf{e}_\sigma) + 3\beta_j^2 g(4\mathbf{e}_j + \mathbf{e}_\sigma) + 3\beta_\sigma \beta_j g(3\mathbf{e}_\sigma + 2\mathbf{e}_j)]$$

$$+ \frac{1}{2} \sum_{j,k} \beta_j \beta_k g(2\mathbf{e}_j + 2\mathbf{e}_k + \mathbf{e}_\sigma).$$

$$A_{\sigma,3}^{(1)} = -30\beta_\sigma g(5\mathbf{e}_\sigma) - 120\beta_\sigma^2 g(6\mathbf{e}_\sigma) - 105\beta_\sigma^3 g(7\mathbf{e}_\sigma)$$

$$- \sum_j [6\beta_j g(4\mathbf{e}_j + \mathbf{e}_\sigma) + 20\beta_j^2 g(\mathbf{e}_\sigma + 5\mathbf{e}_j) + 6\beta_\sigma \beta_j g(3\mathbf{e}_\sigma + 3\mathbf{e}_j)]$$

$$+ 2\beta_\sigma \beta_j g(4\mathbf{e}_\sigma + 2\mathbf{e}_j)$$

$$\begin{aligned}
& + 15\beta_j^3 g(\mathbf{e}_\sigma + 6\mathbf{e}_j) + 15\beta_\sigma^2 \beta_j g(5\mathbf{e}_\sigma + 2\mathbf{e}_j) + 9\beta_\sigma \beta_j^2 g(3\mathbf{e}_\sigma + 4\mathbf{e}_j) \\
& - \sum_{j,k} \left[ 2\beta_j \beta_k g(\mathbf{e}_\sigma + 2\mathbf{e}_j + 3\mathbf{e}_k) + 3\beta_j^2 \beta_k g(\mathbf{e}_\sigma + 4\mathbf{e}_j + 2\mathbf{e}_k) \right. \\
& \quad \left. + \frac{3}{2} \beta_\sigma \beta_j \beta_k g(3\mathbf{e}_\sigma + 2\mathbf{e}_j + 2\mathbf{e}_k) \right] \\
& - \frac{1}{6} \sum_{j,k,l} \beta_j \beta_k \beta_l g(2\mathbf{e}_j + 2\mathbf{e}_k + 2\mathbf{e}_l + \mathbf{e}_\sigma).
\end{aligned}$$

## REFERENCES

1. F. P. Kelly, "Reversibility and Stochastic Networks," New York: John Wiley, 1980, Chapter 3.
2. L. Kleinrock, "Queueing Systems. Vol II: Computer Applications," New York: John Wiley, 1976, Chapter 4.
3. M. Schwartz, "Computer-Communication Network Design and Analysis," Englewood Cliffs, N.J.: Prentice-Hall, 1977, Chapter 11.
4. F. Baskett et al., "Open, Closed and Mixed Networks of Queues with Different Classes of Customers," *J. of the 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. of Res. and Dev.* (1972), pp. 567-72.
6. J. Buzen, "Queueing Network Models of Multiprogramming," Ph.D. Thesis, Div. of Eng. and App. Sci., Harvard University, Cambridge Mass., 1971.
7. R. R. Muntz and J. Wong, "Asymptotic Properties of Closed Queueing Network Models," *Proc. 8th Annual Princeton Conf. on 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. on Computing*, 2 (1973), pp. 183-92.
9. M. Reiser, "A Queueing Network Analysis of Computer Communication Networks with Flow Control," *IEEE Trans. Comm.*, COM-27 No. 8 (August 1979), pp. 1199-209.
10. S. S. Lam and M. Reiser, "Congestion Control of Store-and-Forward Networks by Input Buffer Limits-An Analysis," *IEEE Trans. Comm.*, 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. of Oper. Res.*, 4, No. 4 (November 1979) pp. 357-78.
12. E. Arthurs and B. W. Stuck, "A Theoretical Performance Analysis of a Markovian Switching Node," *IEEE Trans. Comm.*, COM-26, No. 11 (Nov. 1978), pp. 1779-84.
13. J. P. Buzen, "Computational Algorithms for Closed Queueing Networks with Exponential Servers," *Commun. of the ACM*, 16 (1973), pp. 527-31.
14. M. Reiser and H. Kobayashi, "Queueing Networks with Multiple Closed Chains: Theory and Computational Algorithms," *IBM J. of Res. and Dev.*, 19, No. 3 (May 1975), pp. 283-94.
15. M. Reiser and S. S. Lavenberg, "Mean Value Analysis of Closed Multichain Queueing Networks," *J. Assoc. Comput. Mach.*, 27 (April 1980).
16. C. H. Sauer and K. M. Chandy, unpublished work.
17. S. C. Bruell and G. Balbo, "Computational Algorithms for Closed Queueing Networks," New York: North-Holland, 1980.
18. S. S. Lavenberg, "Closed Multichain Product Form Queueing Networks with Large Population Sizes," *IBM Research Report*, RC 8496 (No. 36973), September 1980.
19. J. Zahorjan, "The Approximate Solution of Large Queueing Network Models," *Tech. Report CSRG-122*, University of Toronto, August 1980.
20. Y. Bard, "Some Extensions to Multiclass Queueing Network Analysis," 4th Int. Symp. Model. and Perf. Eval. of Comp. Systems, Vienna (February 1979).
21. J. McKenna, D. Mitra, and K. G. Ramakrishnan, "A Class of Closed Markovian Queueing Networks: Integral Representations, Asymptotic Expansions, Generalizations," *B.S.T.J.*, 60, No. 5 (May-June 1981), pp. 599-641.

22. A. Erdelyi, "Asymptotic Expansions," New York: Dover Publications, 1956.
23. E. T. Copson, "Asymptotic Expansions," Cambridge, U.K.: Cambridge Univ. Press, 1965, Chapters 5 and 6.
24. F. W. J. Olver, "Introduction to Asymptotics and Special Functions," New York: Academic Press, 1974, Chapter 3.
25. N. Bleistein and R. A. Handelsman, "*Asymptotic Expansions of Integrals*," N.Y.: Holt, Rinehart, and Winston, 1975, Chapter 8.
26. L. A. Skinner, "Note on the Asymptotic Behavior of Multidimensional Laplace Integrals," *SIAM J. Math. Anal.*, 11, No. 5 (September 1980), pp. 911-7.



## Parasitic Insensitive, Biphase Switched Capacitor Filters Realized With One Operational Amplifier Per Pole Pair

By K. R. LAKER, P. E. FLEISCHER, and A. GANESAN

(Manuscript received October 9, 1981)

*Practical techniques are given for reducing the number of operational amplifiers (op-amps) in switched capacitor filters. Op-amp count is typically reduced to one op-amp per pole pair, while maintaining the insensitivity to top and bottom plate parasitics heretofore associated with one-op-amp-per-pole structures. These techniques are used to develop a parasitic insensitive single amplifier resonator and a general single amplifier biquad. Next, complete design procedures are given for these circuits. Finally, the same methods are applied to leapfrog structures, where similar op-amp savings are demonstrated.*

### I. INTRODUCTION

The use of active switched capacitor (sc) filters<sup>1-3</sup> as constituents in large-scale integrated (LSI) subsystems<sup>3-10</sup> has been rapidly expanding. Crucial to the realization of manufacturable metal-oxide-semiconductor (MOS) sc filters has been the development of parasitic insensitive sc networks.<sup>6,11,12</sup> As a consequence, it has become a commonly accepted notion that insensitivity to both top and bottom plate parasitic capacitances requires realization with one op-amp per pole (i.e., two op-amps per pole pair). However, since op-amps consume power, represent about 20 percent of a filter's die area, and are sources of noise and power supply feed, it is useful to consider techniques that reduce the number of op-amps required to implement a given transfer function. The purpose of this paper is to introduce practical techniques for achieving this, while retaining the crucial parasitic insensitivities mentioned previously. A straightforward technique for reducing op-amp count is to time share or multiplex<sup>13,14</sup> each op-amp among two or more storage capacitors. A problem with this approach is that each op-amp operates without feedback during the dead zones of the non-

overlapping clocks and can drift to one of the supply rails. To remedy this problem, a simple sc feedback circuit is introduced which ensures that closed loop stability is maintained at all times. In addition to the stability problem, multiplexing within a single filter tends to result in a parasitic-sensitive implementation.<sup>14</sup> This is because many operations within such a filter involve sensing voltages during a clock phase and transferring a proportional charge during a latter phase. In the past, when inverting transfers of this nature were needed, the only recourse has been to use the parasitic-sensitive "toggle" switched capacitor. However, using the parasitic-compensated integrators<sup>15,16</sup> introduced by the authors, sc filters involving multiplexed op-amps can be realized which are insensitive to parasitics. In this paper, we show how additional features provided by these new integrator realizations enhance the design flexibility available with multiplexed op-amp sc structures. In addition, we develop practical, parasitic-insensitive realizations for single-amplifier biquads and  $N/2$  op-amp  $N$ th order\* leapfrog structures. The op-amp-reduced circuits are derived step-by-step from conventional one-op-amp-per-pole prototypes.

## II. NEW PARASITIC-COMPENSATED SC INTEGRATOR REALIZATIONS

Prior to developing the single-amplifier biquad, let us review the parasitic-compensated integrators<sup>15,16</sup> mentioned in the previous section. Consider first the integrator in Fig. 1a with the relevant parasitic capacitances at nodes 1 and 2 denoted as  $C_{p1}$  and  $C_{p2}$ , respectively. To establish the conditions for parasitic insensitivity, let us first determine the voltage across the holding capacitor at node 2 during the even ( $e$ ) time slot. From Fig. 1b it follows that

$$V_2^e = \frac{2C}{4C + C_{p1} + C_{p2}} V_{in}^e. \quad (1)$$

Hence, the net charge transferred to the integrating capacitor  $D$ , during the odd( $o$ ) time slot (obtained from the circuit in Fig. 1c) is

$$Q^o = (2C + C_{p2})z^{-1/2}V_2^e \quad (2)$$

$$= \frac{2C(2C + C_{p2})z^{-1/2}}{4C + C_{p1} + C_{p2}} V_{in}^e. \quad (3)$$

Note that if we set  $C_{p1} = C_{p2} = C_p$ , terms involving the parasitics cancel, i.e.,

$$Q^o = Cz^{-1/2}V_{in}^e$$

---

\* For  $N$  odd,  $(N + 1)/2$  op-amps are required.

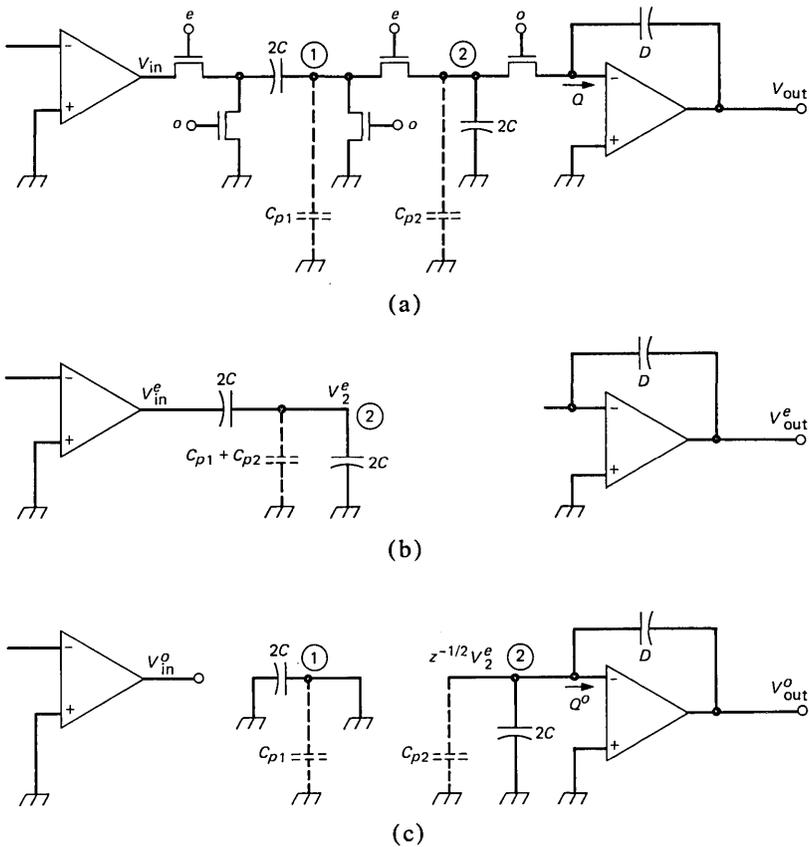


Fig. 1—Parasitic-compensated inverting integrator. (a) Complete circuit with parasitic capacitances. (b) *e*-phase connectivity. (c) *o*-phase connectivity.

when

$$C_{p1} = C_{p2} = C_p. \quad (4)$$

In practice, equality of  $C_{p1} = C_{p2}$  is obtained by matching the geometries of the switches connected to nodes 1 and 2, and by matching the routing associated with nodes 1 and 2 using careful layout. In contrast, predistorting<sup>14</sup> the capacitors to compensate for parasitics is an imprecise operation because of the lack of tracking between unlike capacitors. It should be noted that in the proof given by eqs. (1) to (4), the parasitic capacitors were assumed to be linear. A more general proof establishing the validity of the matching conditions when the parasitics are nonlinear is given in Ref. 16. Throughout the remainder of this text, we shall assume parasitic matching when such an integrator is used. Under these circumstances, the input/output relations become

$$V_{\text{out}}^o = \frac{-[C/D]z^{-1/2}}{1 - z^{-1}} V_{\text{in}}^e \quad (5a)$$

and

$$V_{\text{out}}^e = \frac{-[C/D]z^{-1}}{1 - z^{-1}} V_{\text{in}}^e. \quad (5b)$$

Note that the classical inverting toggle-switched integrator shown in Fig. 2 has the same transfer functions. However, by virtue of the matched parasitics, the implementation in Fig. 1 is parasitic insensitive. We, therefore, refer to the input circuit of this integrator as a parasitic-compensated toggle (PCT).

The noninverting integrator realization in Fig. 3 obtains its parasitic insensitivity via the same matching condition derived for the inverting integrator in Fig. 1. Derivation of the conditions for parasitic insensitivity follows identically that given in eqs. (1) through (4); hence, it will not be shown. Perhaps more interesting are the input/output relations, which are given below:

$$V_{\text{out}}^e = \frac{[C/D]z^{-1}}{1 - z^{-1}} V_{\text{in}}^e \quad (6a)$$

and

$$V_{\text{out}}^o = \frac{[C/D]z^{-3/2}}{1 - z^{-1}} V_{\text{in}}^e. \quad (6b)$$

Particular attention is directed to the extra one-half-clock-period delay produced by this circuit when compared to the standard noninverting structure.<sup>1-3</sup> We shall see in later sections that this extra delay, heretofore available only by using additional clock phases, provides considerable flexibility for multiplexing op-amps in switched capacitor filters. In further references to the structure in Fig. 3, proper parasitic matching will be assumed. Since the input circuit to the integrator in Fig. 3 provides inversion at the summing junction, it will be referred to as an inverting parasitic-compensated toggle (IPCT).

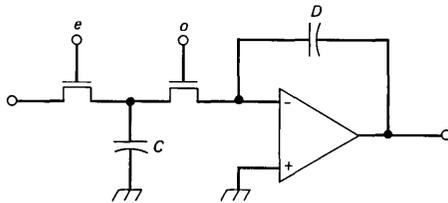


Fig. 2—Classical inverting toggle-switched integrator.

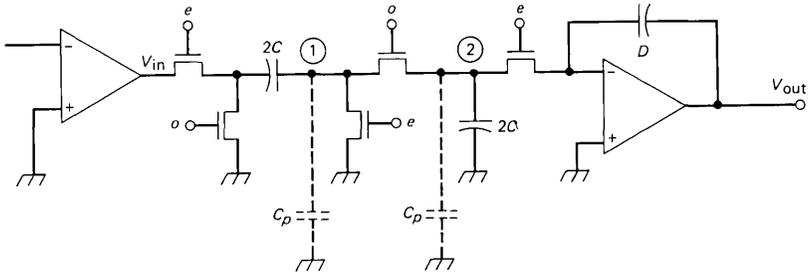


Fig. 3—Parasitic-compensated noninverting integrator.

### III. ALL POLE, PARASITIC-INSENSITIVE, SINGLE-AMPLIFIER SC BIQUAD

To begin the derivation of the single-amplifier biquad, consider the conventional, parasitic-insensitive two-op-amp biquad<sup>11,12</sup> in Fig. 4a. The transfer functions<sup>12</sup> to each of the op-amp outputs ( $V_{o1}$ ,  $V_{o2}$ ) are readily obtained from the  $z$ -domain equivalent circuit<sup>3,12,17</sup> in Fig. 5a. That is,

$$T = \frac{V_{o2}^e}{V_{in}^e} = \frac{-AGz^{-1}}{D(B + F) - (2DB + DF - AC)z^{-1} + DBz^{-2}} \quad (7a)$$

and

$$T' = \frac{V_{o1}^e}{V_{in}^e} = \frac{-G(B + F - Bz^{-1})}{D(B + F) - (2DB + DF - AC)z^{-1} + DBz^{-2}} \quad (7b)$$

Let us now replace the sc elements  $G$  and  $C$  with corresponding PCTs according to the schematic in Fig. 1a. The resulting circuit is shown in

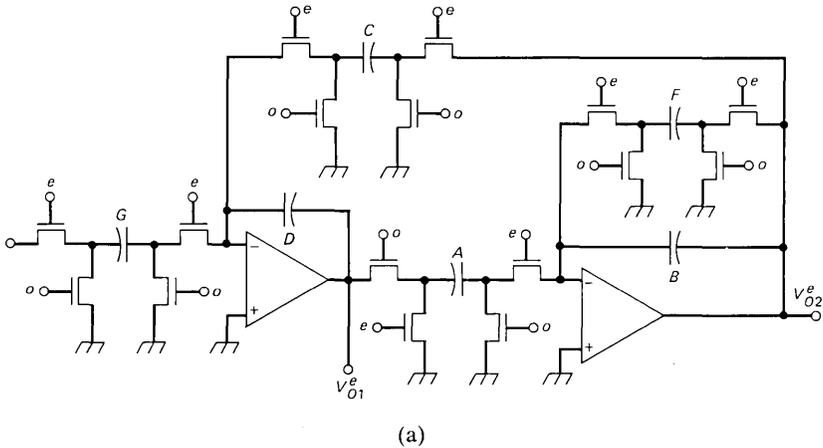
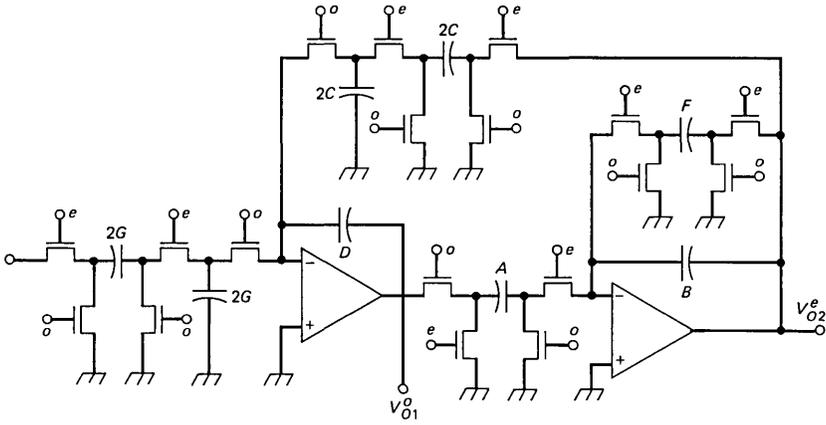
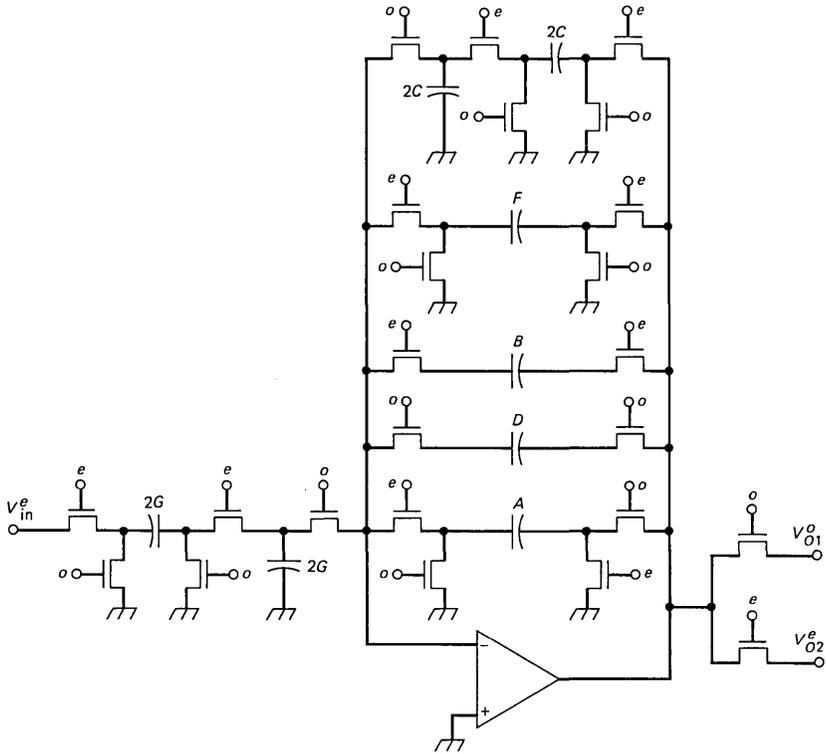


Fig. 4a—Switched-capacitor all-pole biquad—parasitic-insensitive realization.



(b)

Fig. 4b—Switched-capacitor all-pole biquad—parasitic-compensated equivalent to Fig. 4a.



(c)

Fig. 4c—Switched-capacitor all-pole biquad—single-amplifier equivalent to Fig. 4b.

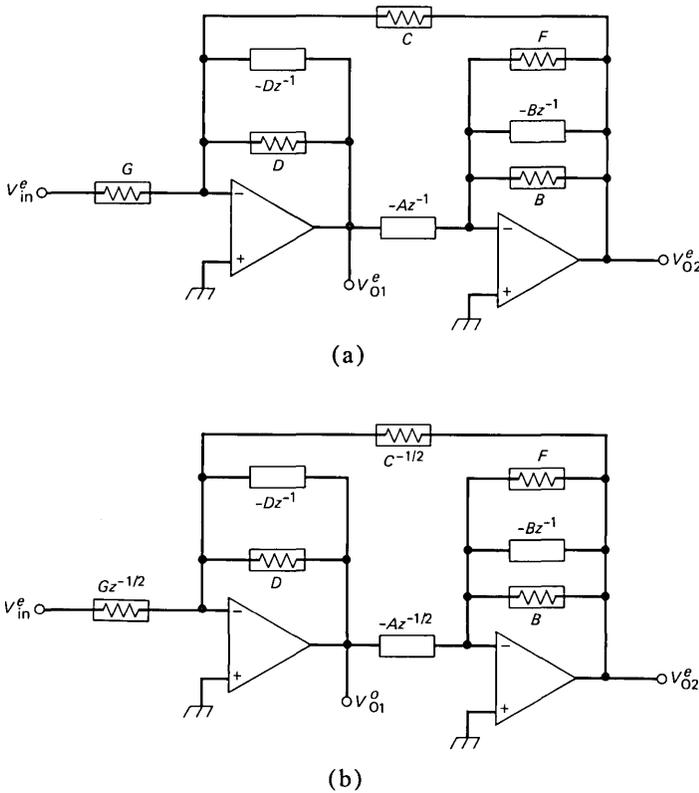


Fig. 5—Switched-capacitor all-pole biquad equivalent circuits. (a) Equivalent circuit for the sc biquad in Fig. 4a. (b) Equivalent circuit for the sc biquads in Figs. 4b and 4c.

Fig. 4b and its  $z$ -domain equivalent in Fig. 5b. The following observations regarding the biquads in Figs. 4a and 4b can be made:

(i) In both circuits, the integrating (storage) capacitors are updated during only one clock phase. During their respective hold phases, the op-amps are idle.

(ii) In the circuit of Fig. 4a, storage capacitors  $D$  and  $B$  are updated during the same phase (in this case the  $e$ -phase). In contrast, for the circuit of Fig. 4b, storage capacitors  $D$  and  $B$  are updated on alternate phases.

(iii) Referring to the equivalent circuits in Fig. 5a and 5b, we see that the loop gains for both circuits are identical. The only difference is the additional half-clock-period delay in the transfer function introduced for the first op-amp output. The delay between the filter input and the output of the second op-amp is the same in both cases. The transfer functions for the biquad in Fig. 4b are

$$\frac{V_{o2}^e}{V_{in}^e} = \frac{-AGz^{-1}}{D(B+F) - (2DB + DF - AC)z^{-1} + DBz^{-2}} \quad (8a)$$

and

$$\frac{V_{o1}^e}{V_{in}^e} = \frac{-z^{-1/2}G(B+F - Bz^{-1})}{D(B+F) - (2DB + DF - AC)z^{-1} + DBz^{-2}} \quad (8b)$$

Reflecting on observation (1), it is seen that since each op-amp is idle during half the time, all the processing might be performed by one op-amp. Since, in addition, capacitors  $B$  and  $D$  are updated during alternate clock phases, according to observation (2), the two op-amps in Fig. 4b can be replaced with a single op-amp, multiplexed between the two storage capacitors  $D$  and  $B$ . The resulting single amplifier circuit is shown in Fig. 4c.

Let us now make some observations regarding the single-amplifier biquad in Fig. 4c.

(i) Biquads 4b and 4c share the same equivalent circuit (Fig. 5b). It will often be found that the equivalent circuit is a convenient mechanism for "untangling" single-amplifier biquads so that they may be analyzed in a straightforward fashion. Thus, the equivalent circuit provides a useful link between the single-amplifier biquad and its two-op-amp counterpart.

(ii) It has been shown that to multiplex the op-amps within a two-amplifier biquad, the storage capacitors must be updated during different clock phases. Hence, the use of either PCTs or parasitic-sensitive toggles<sup>14</sup> seems unavoidable.

(iii) The two transfer functions in eqs. (8a) and (8b) are available at the op-amp output on alternate phases. Sampling the output on the  $e$ -phase yields eq. (8a), while sampling on the  $o$ -phase yields eq. (8b).

(iv) Damping is provided by a switched capacitor which has been called " $F$ -type" damping.<sup>12</sup> Referring to the two op-amp biquad in Fig. 4a, an alternative form of damping is obtained by placing an unswitched capacitor across switched capacitor  $C$ . This has been called<sup>12</sup> " $E$ -type" damping. Unfortunately,  $E$ -type damping constrains the  $D$  and  $B$  charge updates to occur on the same phase. Consequently,  $E$ -type damping is not available in single-amplifier biquads.

Now that the basic implementation has been derived, let us examine the integrity of the feedback loop as the switches open and close. Since the clocks are nonoverlapping, it is readily seen that there are periods of time during which the op-amp in Fig. 4c operates without feedback. Clearly, this is a potentially dangerous circuit condition.

An interesting but impractical way to maintain loop closure is to place an unswitched capacitor of value  $M$  across the op-amp as shown in Fig. 6a. Although the  $M$  capacitor solves the stability problem, an

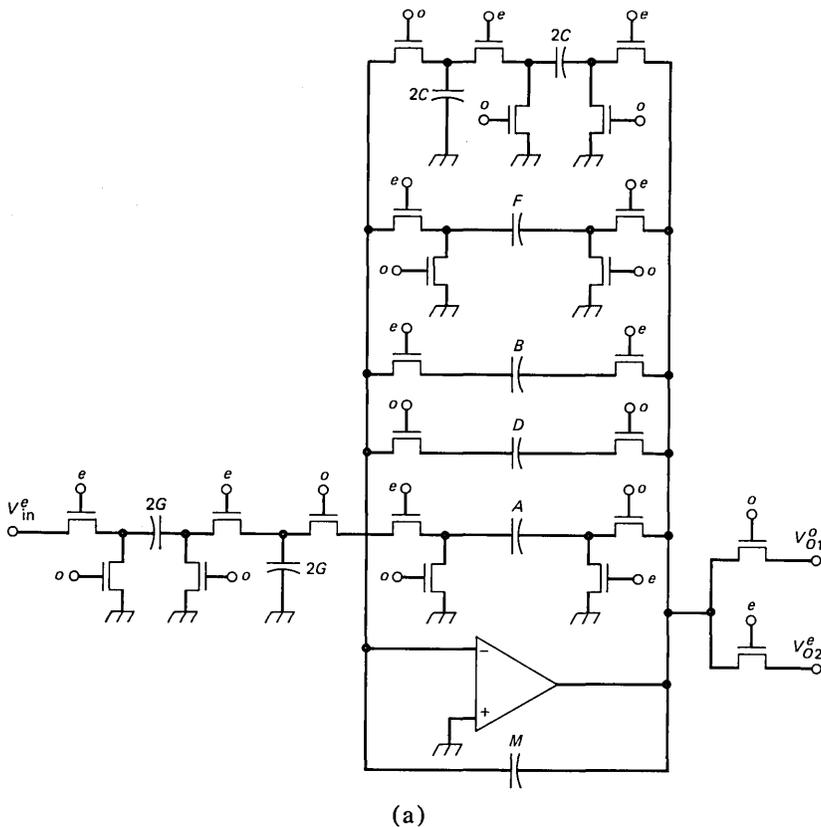


Fig. 6a—Single-amplifier biquad stabilization via continuous feedback—circuit schematic.

analysis of this circuit reveals the impracticality of the solution. Since the results of the analysis are rather interesting we present them, in spite of our reluctance to recommend the  $M$  capacitive feedback. First, let us examine the equivalent circuit in Fig. 6b. Note that while in the actual SC circuit the  $M$  capacitor appears once, in the equivalent circuit it appears virtually everywhere. This is a consequence of the fact that  $M$  introduces coupling between  $B$  and  $D$ . Using the equivalent circuit in Fig. 6b, the transfer functions for the two-op-amp outputs can be easily derived. For this presentation we need only concern ourselves with output  $V_{o2}^e$ , i.e.,

$$\frac{V_{o2}^e}{V_{in}^e} = \frac{-(A + M)Gz^{-1}}{\{D(B + F) + M(B + D + F) + M^2\} - \{2DB + DF - AC + M(B + D + A - C)\}z^{-1} + DBz^{-2}} \quad (9)$$

Because of the coupling provided by  $M$ , the coefficient for the constant

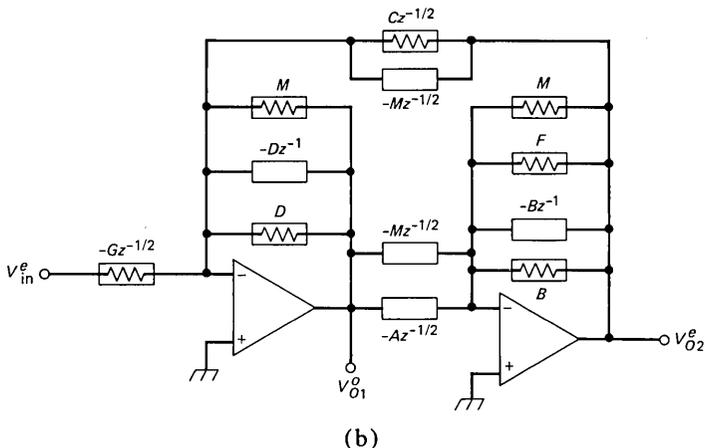


Fig. 6b—Single-amplifier biquad stabilization via continuous feedback—equivalent circuit.

term of the denominator depends quadratically on  $M$ . In addition,  $M$  provides a leakage path for both  $B$  and  $D$  which can severely enhance the damping heretofore provided by  $F$  alone. In order for  $M$  to have an insignificant effect on the circuit performance,  $DF \gg M(B + D)$ , where  $B$  and  $D$  are typically large, while  $F$  is near minimum value. Hence, if  $B$  and  $D$  are roughly equal, then  $M \ll F/2$ . Capacitor  $M$  is then too small to be practical.

A practical solution which does not affect the biquad's transfer characteristic is illustrated in Fig. 7. The sc network involving  $X$  and  $Y$  is seen to provide feedback around the op-amp when both clocks  $e$  and  $o$  are low, i.e., the dead zones of the nonoverlapping clocks. When either  $e$  or  $o$  is high,  $X$ ,  $Y$  simply introduce capacitance to ground from the op-amp summing junction and output, respectively. The values of  $X$  and  $Y$  are relatively unimportant and can be set conveniently to one unit each. The crosstalk introduced by this arrangement is insignificant. If the clock rise and fall times are reasonably fast, the opportunities for crosstalk, i.e., interaction between  $X$ ,  $Y$  and  $B$ ,  $D$ , are minimal. Even if this is not the case, the interaction is self-compensating. For example, if  $X$ ,  $Y$  happen to pull out of the loop more slowly than  $D$  enters the loop, some fraction of the charge stored on  $D$  will leak onto  $X$ ,  $Y$ . However, when the  $X$ ,  $Y$  common junction is grounded, the charge which previously leaked off  $D$  is returned to it. This self-compensating property will be imperfect as a consequence of finite op-amp gain. However, for op-amp dc gains of 1000 or more, the interaction between  $X$ ,  $Y$  and  $B$ ,  $D$  will be negligible. Thus, the  $X$ ,  $Y$  sc network provides an elegant and practical solution to the op-amp stability problem.

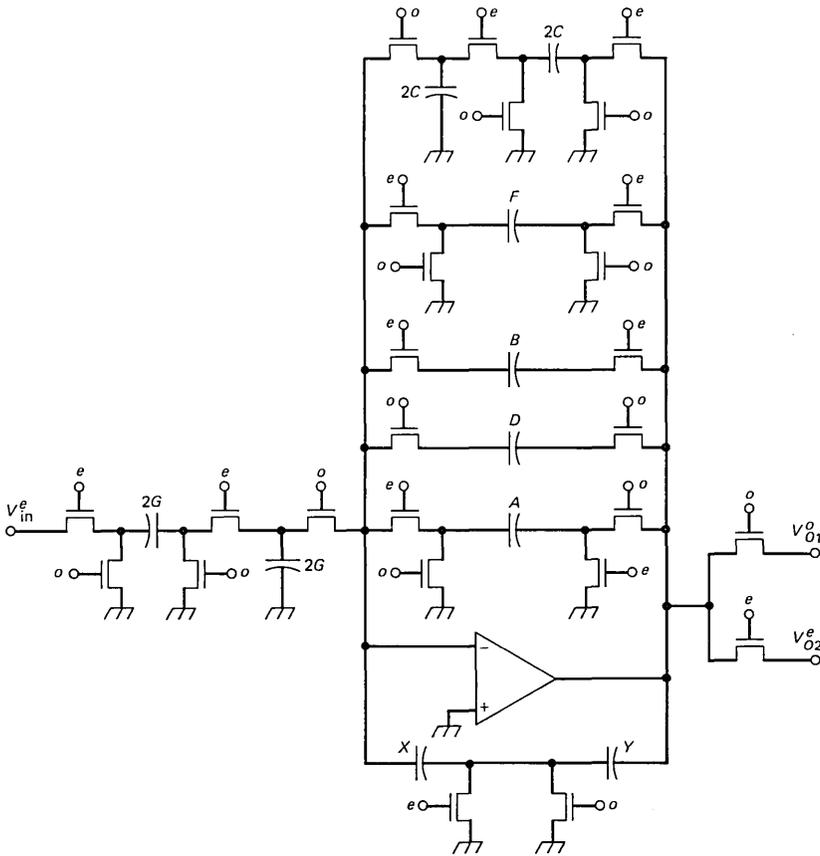


Fig. 7—Single-amplifier biquad stabilization via  $XY$  sc feedback.

#### IV. REALIZING SINGLE-AMPLIFIER BIQUADS WITH TRANSMISSION ZEROS

The all-pole resonator in Fig. 7 can be easily generalized to realize transfer functions with transmission zeros by adding the feed forward circuitry shown in Fig. 8a. The  $z$ -domain equivalent circuit is given in Fig. 8b. The transfer function for each time slot can be readily derived from the equivalent circuit:

$$T = \frac{V_{o2}^e}{V_{in}^e} = \frac{-\{D(K + I) - (2DK + D(I + J) - AG)z^{-1} + D(K + J)z^{-2}\}}{D(B + F) - (2DB + DF - AC)z^{-1} + DBz^{-2}} \quad (10a)$$

and

$$T^v = \frac{V_{o1}^e}{V_{in}^e} = \frac{z^{-1/2}\{C(K + I) - G(B + F) - (C(K + J) - GB)z^{-1}\}}{D(B + F) - (2DB + DF - AC)z^{-1} + DBz^{-2}} \quad (10b)$$

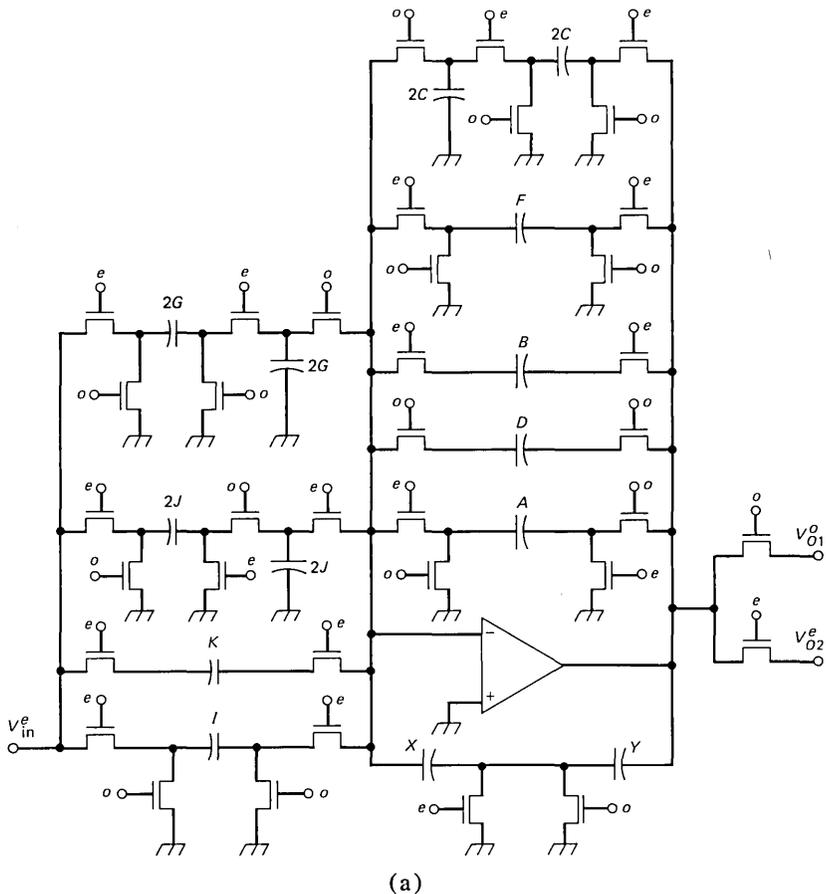


Fig. 8a—Single-amplifier biquad which realizes the transfer functions in eq. (10)—circuit schematic.

The design of this single-amplifier biquad follows along previously published design procedures<sup>12</sup> for two-op-amp biquads. Note the manner in which the delay property of the IPTCT has been exploited. Comparing the transfer functions  $T$  and  $T'$ , we observe that only  $T$  has a quadratic numerator. Hence, by focusing our attention on  $T$ , little generality and flexibility is lost. As in Ref. 12, we can simplify  $T$  by arbitrarily setting  $A = B = D = 1$ . At the end of the design process, one can adjust the values of  $A$ ,  $B$ , and  $D$  to set the gain level of  $T'$  and to independently scale the capacitances, which share common time slots. Setting  $A = B = D = 1$  in eq. (10a) yields

$$T = \frac{V_{o2}^e}{V_{in}^e} = \frac{-\{(K + I) - (2K + I + J - G)z^{-1} + (K + J)z^{-2}\}}{(1 + F) - (2 + F - C)z^{-1} + z^{-2}} \quad (11)$$

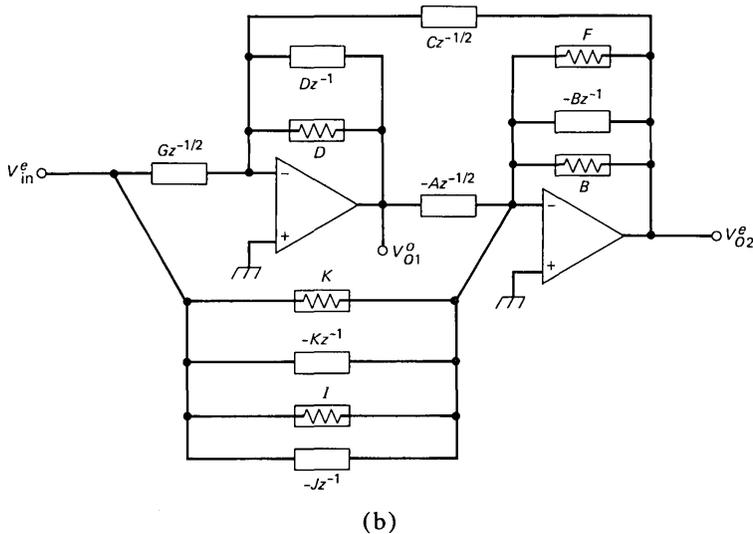


Fig. 8b—Single-amplifier biquad which realizes the transfer functions in eq. (10)—equivalent circuit.

If the desired transfer function is given by

$$T = \frac{m\{\gamma - \epsilon z^{-1} + \delta z^{-2}\}}{1 - \alpha z^{-1} + \beta z^{-2}}, \quad (12)$$

the unscaled values for capacitors  $C$ ,  $F$ ,  $G$ ,  $I$ ,  $J$ , and  $K$  are obtained by matching the like coefficients of eqs. (11) and (12). For the poles, the design equations are

$$F = \frac{1 - \beta}{\beta} \quad (13a)$$

and

$$C = \frac{1 + \beta - \alpha}{\beta}. \quad (13b)$$

To place the transmission zeros, the general design equations and a simple solution, are given in Table I for each generic filter type.<sup>12</sup> The generic filter types are listed in Table II for convenient reference. Note that the actual capacitor values  $G$ ,  $I$ ,  $J$ , and  $K$  are related to the scaled values  $\hat{G}$ ,  $\hat{I}$ ,  $\hat{J}$ , and  $\hat{K}$  by the relation

$$\hat{x} = (1 + F)x, \quad (14)$$

where

$$x = G, I, J, K.$$

Examining the numerator of  $T$  in eq. (11), one observes that real zeros

Table I—Zero placement formulas for  $T$

Filter Type	Design Equations	Simple Solution
LP20	$\hat{K} + \hat{I} =  m $ $2\hat{K} + \hat{I} + \hat{J} - \hat{G} = -2 m $ $\hat{K} + \hat{J} =  m $	$\hat{I} = \hat{J} = 0, \hat{K} =  m , \hat{G} = 4 m $
LP11	$\hat{K} + \hat{I} = 0$ $2\hat{K} + \hat{I} + \hat{J} - \hat{G} = - m $ $\hat{K} + \hat{J} =  m $	$\hat{I} = \hat{K} = 0, \hat{J} =  m , \hat{G} = 2 m $
LP10	$\hat{K} + \hat{I} =  m $ $2\hat{K} + \hat{I} + \hat{J} - \hat{G} = - m $ $\hat{K} + \hat{J} = 0$	$\hat{J} = \hat{K} = 0, \hat{I} =  m , \hat{G} = 2 m $
LP02	$\hat{K} + \hat{I} = 0$ $2\hat{K} + \hat{I} + \hat{J} - \hat{G} = 0$ $\hat{K} + \hat{J} =  m $	$\hat{I} = \hat{K} = 0, \hat{J} =  m , \hat{G} =  m $
LP01	$\hat{K} + \hat{I} = 0$ $2\hat{K} + \hat{I} + \hat{J} - \hat{G} = - m $ $\hat{K} + \hat{J} = 0$	$\hat{I} = \hat{J} = \hat{K} = 0, \hat{G} =  m $
LP00	$\hat{K} + \hat{I} =  m $ $2\hat{K} + \hat{I} + \hat{J} - \hat{G} = 0$ $\hat{K} + \hat{J} = 0$	$\hat{J} = \hat{K} = 0, \hat{I} =  m , \hat{G} =  m $
BP10	$\hat{K} + \hat{I} =  m $ $2\hat{K} + \hat{I} + \hat{J} - \hat{G} = 0$ $\hat{K} + \hat{J} = - m $	Not realizable
BP01	$\hat{K} + \hat{I} = 0$ $2\hat{K} + \hat{I} + \hat{J} - \hat{G} =  m $ $\hat{K} + \hat{J} =  m $	$\hat{G} = \hat{I} = \hat{K} = 0, \hat{J} =  m $
BP00	$\hat{K} + \hat{I} =  m $ $2\hat{K} + \hat{I} + \hat{J} - \hat{G} =  m $ $\hat{K} + \hat{J} = 0$	$\hat{G} = \hat{J} = \hat{K} = 0, \hat{I} =  m $
HP	$\hat{K} + \hat{I} =  m $ $2\hat{K} + \hat{I} + \hat{J} - \hat{G} = 2 m $ $\hat{K} + \hat{J} =  m $	$\hat{G} = \hat{I} = \hat{J} = 0, \hat{K} =  m $
HPN and LPN	$\hat{K} + \hat{I} =  m $ $2\hat{K} + \hat{I} + \hat{J} - \hat{G} =  m \epsilon$ $\hat{K} + \hat{J} =  m $	$\hat{I} = \hat{J} = 0, \hat{K} =  m , \hat{G} =  m (2 - \epsilon)$
AP	$\hat{K} + \hat{I} =  m \beta$ $2\hat{K} + \hat{I} + \hat{J} - \hat{G} =  m \alpha$ $\hat{K} + \hat{J} =  m $	$\hat{I} = 0, \hat{K} =  m \beta, \hat{J} =  m (1 - \beta), \hat{G} =  m (1 + \beta - \alpha)$

on alternate sides of  $z = 0$  are nonrealizable. As a result, the bilinear bandpass BP10 function is not realizable with this circuit. This is not a severe restriction since the other bandpass types are realizable.

As noted previously, the synthesis equations result in unscaled capacitor values. To obtain properly scaled capacitors, one must first scale the level of  $V_{o1}^{\circ}$  to a suitable level. By adjusting  $A$  and  $D$ , the level of  $V_{o1}^{\circ}$  (i.e.,  $T'$ ) can be scaled without affecting  $V_{o2}^{\circ}$  (i.e.,  $T$ ). More precisely, if the gain constant of  $T'$  is to be scaled to

Table II—Generic biquad transfer functions

Generic Form	Numerator $N(z)$
LP20 (bilinear transform)	$m(1+z^{-1})^2$
LP11	$mz^{-1}(1+z^{-1})$
LP10	$m(1+z^{-1})$
LP02	$mz^{-2}$
LP01	$mz^{-1}$
LP00	$m$
BP10 (bilinear transform)	$m(1-z^{-1})(1+z^{-1})$
BP01	$mz^{-1}(1-z^{-1})$
BP00	$m(1-z^{-1})$
HP	$m(1-z^{-1})^2$
LPN	$m(1-\epsilon z^{-1}+z^{-2}), \epsilon > \alpha/\sqrt{\beta}, \beta > 0$
HPN	$m(1-\epsilon z^{-1}+z^{-2}), \epsilon < \alpha/\sqrt{\beta}, \beta > 0$
AP	$m(\beta-\alpha z^{-1}+z^{-2})$
General	$m(\gamma-\epsilon z^{-1}+\sigma z^{-2})$

$$T' \rightarrow \mu T', \tag{15}$$

then it is only necessary to scale

$$(A, D) \rightarrow \left[ \frac{1}{\mu} A, \frac{1}{\mu} D \right]. \tag{16}$$

The gain constant associated with  $T$  will remain constant under this scaling.

Once satisfactory gain levels have been achieved, one can scale all capacitors associated with each time slot so that the minimum capacitance corresponds to a single unit. The two groups of capacitors which are to be scaled together are

- Group 1:  $(C, D, G)$
- Group 2:  $(A, B, F, I, J, K)$ .

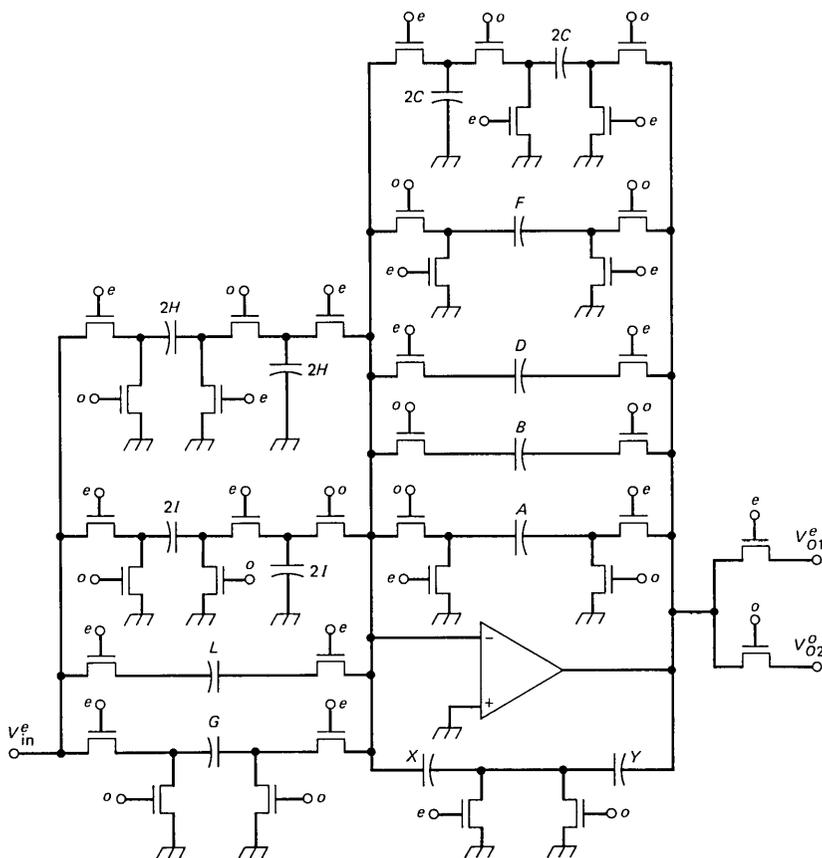
Note that the capacitors in each group are distinguished by the fact that they are connected to the input node of the op-amp during the same clock phase. The groupings are easily identified in the equivalent circuit where the groupings are determined by the amplifier input they are incident on.

To cascade single-amplifier biquads, it is important that the input be accepted by all feed-ins on the same clock phase. The biquad in Fig. 8 possesses this property.

Note that the inputs to the single-amplifier resonator can be chosen differently from those selected in Fig. 8. An alternate realization is shown in Fig. 9. The transfer functions for this circuit are

$$T = \frac{V_{o1}^e}{V_{in}^e} = \frac{-\{(B+F)(G+L) - [2BL + B(G+H) + F(H+L) + CI]z^{-1} + B(H+L)z^{-2}\}}{D(B+F) - (2DB + DF - AC)z^{-1} + DBz^{-2}} \tag{17a}$$

and



(a)

Fig. 9a—Single-amplifier biquad which realizes the transfer functions in eq. (17)—circuit schematic.

$$T' = \frac{V_{O2}^o}{V_{in}^e} = \frac{-z^{-1/2}\{A(G + L) + DI - (A(H + L) + DI)z^{-1}\}}{D(B + F) - (2DB + DF - AC)z^{-1} + DBz^{-2}}. \quad (17b)$$

This circuit is not as general as the circuit in Fig. 8, hence, it will not be pursued further.

It is well known<sup>12</sup> that, by sharing switches operating in synchronism between common nodes, the switch count for a given sc filter can be significantly reduced. Circuits involving PCTs can also share switches. Switch sharing for PCT branches is demonstrated in Fig. 10. Note that sharing switches and combining the toggling capacitors 2G and 2C, as shown in Fig. 10b, retains the parasitic cancellation property. In this operation, three switches are saved. Sharing of switches within the

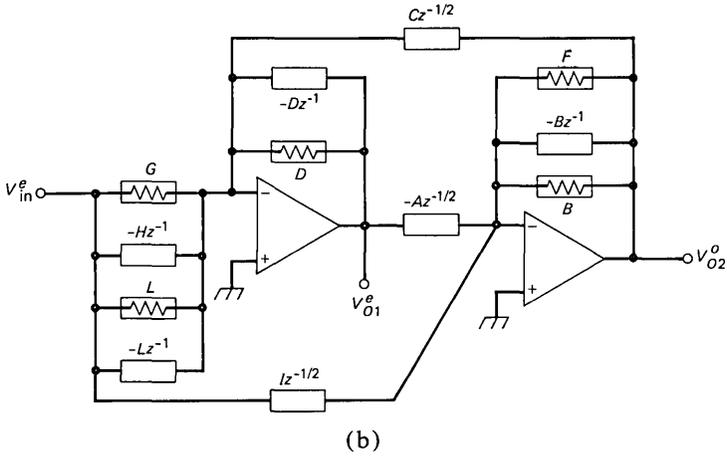


Fig. 9b—Single-amplifier biquad which realizes the transfer functions in eq. (17)—equivalent circuit.

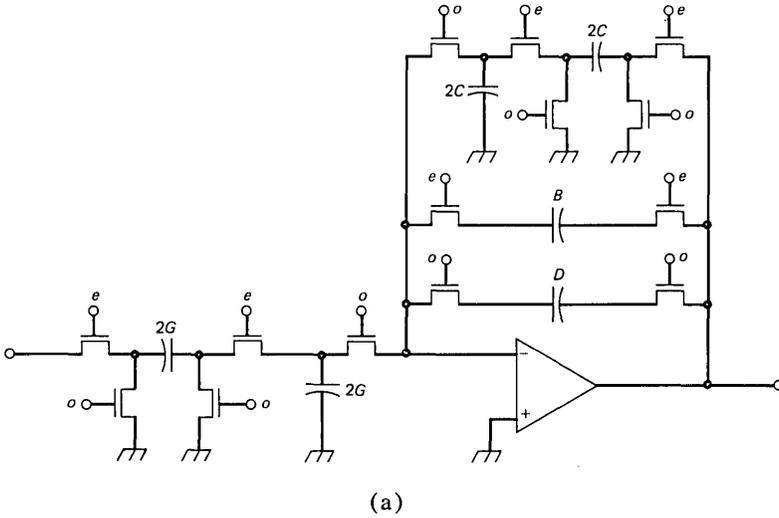
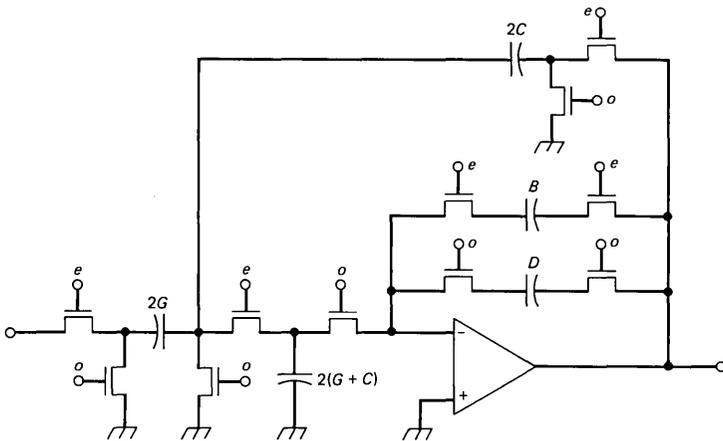


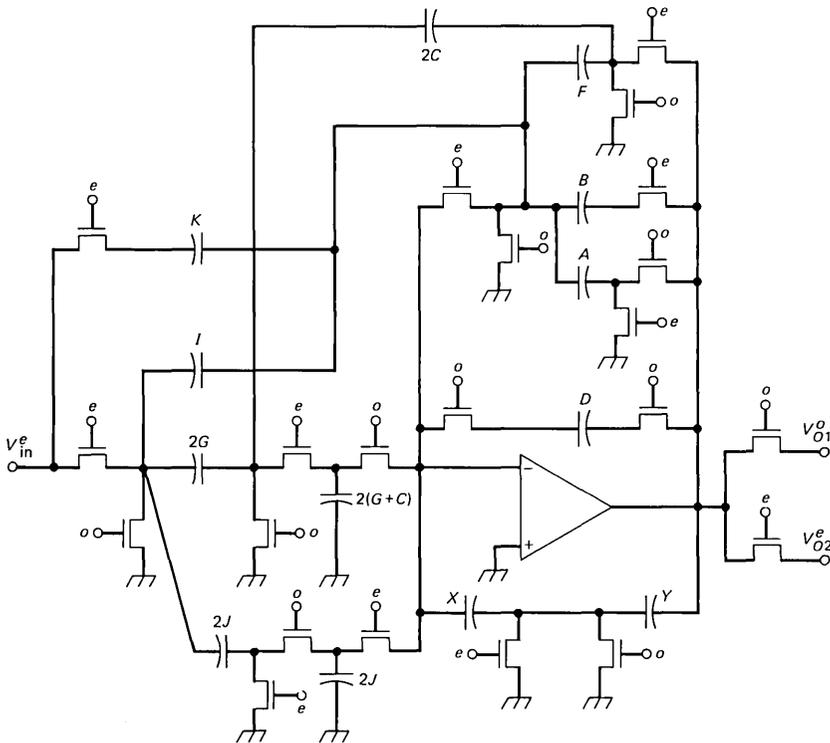
Fig. 10a—Switch sharing among parasitic-compensated toggles—original circuit.

biquads in Figs. 8 and 9 yield the switch-reduced circuits in Figs. 11a and 11b, respectively. Also note that one of the switches associated with a storage capacitor can often be eliminated by judicious sharing with one of the single-pole-double-throw (SPDT) switches associated with a switched capacitor. For example in Fig. 11a, the  $e$  switch on the top plate of  $B$  is shared with the  $e$  half of the SPDT switch on the top plates of  $F$  and  $A$ . The fact that the top plate of  $B$  is switched to ground during  $o$  does not adversely affect circuit operation. However,



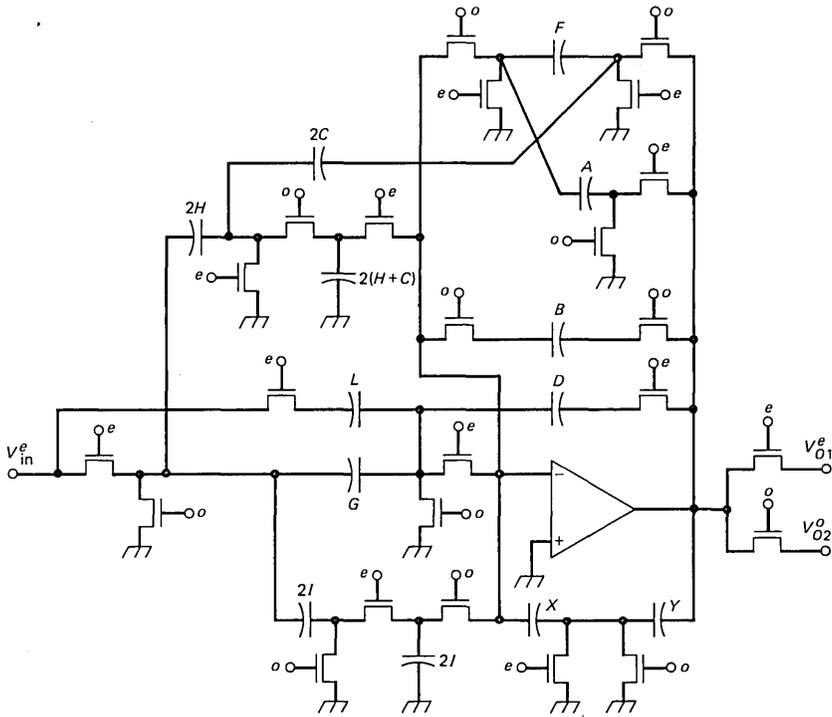
(b)

Fig. 10b—Switch sharing among parasitic-compensated toggles—switch-reduced equivalent of Fig. 10a.



(a)

Fig. 11a—Switch sharing in single-amplifier biquads—switched shared equivalent to the circuit in Fig. 8a.



(b)

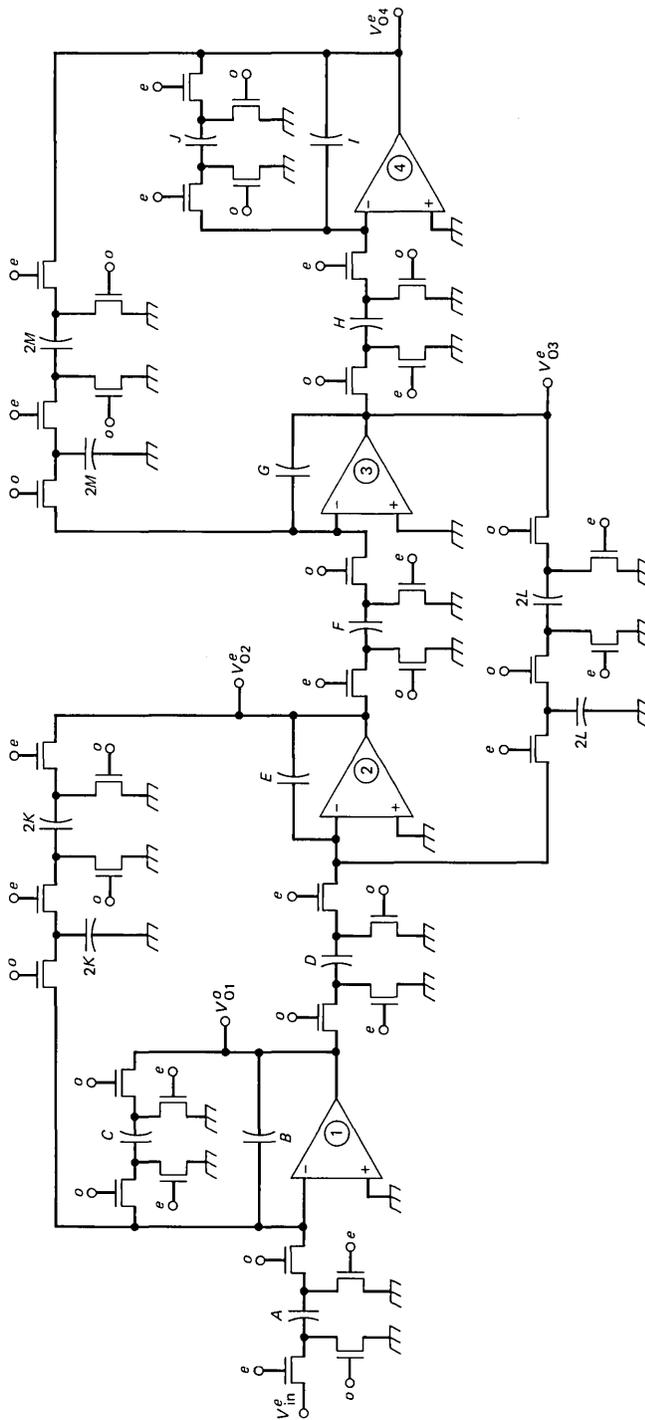
Fig. 11b—Switch sharing in single-amplifier biquads—switched shared equivalent to the circuit in Fig. 9a.

when sharing storage-capacitor switches, one must be careful to avoid inadvertently including storage capacitors in a closed loop of capacitors during their holding phases.

## V. LEAPFROG STRUCTURES

The circuit techniques illustrated in the previous sections are readily applied to reduce op-amp count in leapfrog structures. The resulting structures are parasitic insensitive. As before, closed loop stability during clock dead zones can be ensured by using the feedback network of Fig. 7 for every op-amp. It is readily shown that  $N$ th order leapfrog structures can be implemented with  $N/2$  op-amps for  $N$  even and  $(N + 1)/2$  op-amps for  $N$  odd.

It is interesting to note that lossless discrete integrator (LDI) derived<sup>1-3</sup> leapfrog structures are inherently multiplexable. An implementation of a four op-amp, fourth-order low-pass leapfrog structure is shown in Fig. 12a. Note that all parasitic-sensitive toggle-switched



(a)

Fig. 12a—Fourth-order low-pass SC leapfrog filter—four-amplifier realization.

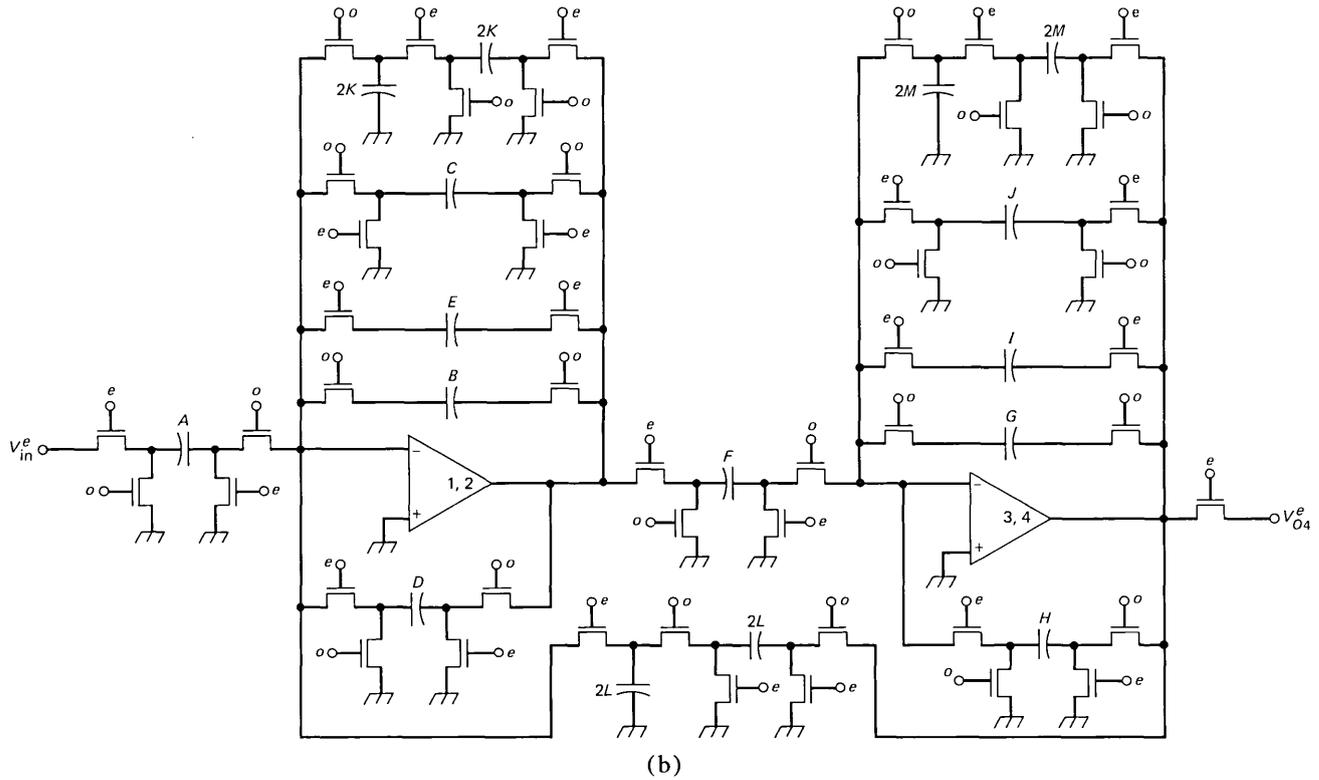


Fig. 12b—Fourth-order low-pass SC leapfrog filter—two-amplifier realization.

capacitors have been replaced by PCTs. Hence, the implementation in Fig. 12a is parasitic insensitive. In the implementation shown, op-amps 1 and 3 update charge during the odd phase, while op-amps 2 and 4 update charge on the even phase. To derive the two-op-amp implementation, one simply multiplexes alternate time-slotted op-amps in a pairwise fashion. For the implementation in Fig. 12b, op-amps 1, 2 and op-amps 3, 4 are pairwise multiplexed. One could alternatively pairwise multiplex op-amps 1, 4 and 2, 3. However, it appears that the interconnecting network is simpler if adjacent op-amps are multiplexed as is the case in Fig. 12b. To complete the structure,  $X$ ,  $Y$  sc feedback circuits should be added to each op-amp. Switch sharing can also be used to reduce the switch count, but this will not be shown here.

## VI. CONCLUSION

Circuit techniques which reduce by a factor of two the number of op-amps required to implement a given sc filter, while maintaining all parasitic insensitivity, have been given. Both biquad and leapfrog structures have been considered. Although only all-pole leapfrog structures were discussed in this paper, the techniques presented are readily applicable to elliptic-type leapfrog structures as well.

Breadboards have been constructed to establish feasibility and to examine the severity of the stability problem. Our experience has been that some sort of feedback is required around the op-amps at all times and that the  $X$ ,  $Y$  sc feedback circuit works as expected. However, to determine whether such op-amp multiplexed circuits are really practical, sample designs will have to be realized in integrated form and characterized. In particular, the effects of op-amp slew rate and settling characteristics, as well as the noise and power supply rejection properties of these circuits, need to be investigated further.

## REFERENCES

1. P. R. Gray, D. A. Hodges, and R. W. Brodersen, *Analog MOS Integrated Circuits*, New York: IEEE Press, 1980.
2. R. Shaumann, M. A. Soderstrand, and K. R. Laker, *Modern Active Filter Design*, New York: IEEE Press, 1981.
3. M. S. Ghausi and K. R. Laker, *Modern Filter Design; Active-RC and Switched Capacitor*, Englewood Cliffs, New Jersey: Prentice Hall, Inc., 1981.
4. P. R. Gray et al., "A Single-Chip NMOS Dual Channel Filter for PCM Telephony Applications," *IEEE J. Solid State Circuits*, SC-14 (December, 1979), pp. 981-91.
5. D. J. Allstot, R. W. Brodersen, and P. R. Gray, "An Electrically-Programmable Switched Capacitor Filter," *IEEE J. Solid State Circuits*, SC-14 (December, 1979), pp. 1034-41.
6. B. J. White, G. M. Jacobs, and G. F. Landsburg, "A Monolithic Dual Tone Multifrequency Receiver," *IEEE J. Solid State Circuits*, SC-14 (December, 1979) pp. 991-7.
7. R. Gregorian, G. Wagner, and W. E. Nicholson, Jr., "An Integrated Single Chip PCM Voice Codec with Filters," *IEEE J. Solid State Circuits*, SC-16 (August, 1981), pp. 322-32.

8. D. G. Marsh et al., "A Single Chip CMOS PCM Codec with Filters," *IEEE J. Solid State Circuits*, *SC-16* (August, 1981), pp. 308-14.
9. P. E. Fleischer et al., "An NMOS Switched Capacitor Building Block for Telecommunications Applications," *IEEE Trans. Circuits and Systems*, *CAS-27* (June, 1980), pp. 552-9.
10. A. Iwata et al., "A Single Chip Codec with Switched Capacitor Filters," *IEEE J. Solid State Circuits*, *SC-16* (August, 1981), pp. 315-21.
11. K. Martin and A. S. Sedra, "Strays Insensitive Switched Capacitor Filters Based on the Bilinear  $z$ -Transform," *Electronics Letters*, *15* (July, 1979), pp. 365-6.
12. P. E. Fleischer and K. R. Laker, "A Family of Active Switched Capacitor Biquad Building Blocks," *B.S.T.J.*, *58* (December, 1979), pp. 2235-69.
13. P. W. Bosshart, "A Multiplexed Switched Capacitor Filter Bank," *IEEE J. Solid State Circuits*, *SC-15* (December, 1980), pp. 939-45.
14. T. Inoue and F. Ueno, "New Switched Capacitor Ladder Filter with Reduced Number of Op Amps," *Electronics Letters*, *17* (March, 1981), pp. 209-10.
15. K. R. Laker, P. E. Fleischer, and A. Ganesan, "Parasitic Insensitive Switched Capacitor Biquads: A Review," presented at the 1981 ISCAS, April, 1981.
16. P. E. Fleischer, A. Ganesan, and K. R. Laker, "Parasitic Compensated Switched Capacitor Circuits," *Electronics Letters*, *17*, No. 24 (November 26, 1981), pp. 929-31.
17. K. R. Laker, "Equivalent Circuits for the Analysis and Synthesis of Switched Capacitor Networks," *B.S.T.J.*, *58* (March, 1979), pp. 727-67.



## Moment Formulae for a Class of Mixed Multi-Job-Type Queueing Networks

By H. HEFFES

(Manuscript received August 26, 1981)

*Queueing network models play an important role during each stage of a computer system's life cycle (from initial conception to system maturity), where in each stage broadly applicable performance analysis tools are needed. This paper presents new results which contribute to the foundations of a tool to support performance analysis and modeling activities. In dealing with some performance issues, it is important to be able to quantify distribution or moment information, because these quantities can influence system capacity and service and performance measures. It is also important that models include the effect of congestion adaptive I/O devices, in a stable and efficient manner, for this inclusion can significantly affect the outcome of studying certain performance issues. We address the problem of direct, recursive computation of moments of the queue size distributions at a class of service centers embedded in a mixed network of queues. The parameterized class includes state-dependent processing rates useful in modeling congestion adaptive I/O devices. We also present results for calculating moments of both the waiting time and virtual delay (work backlog) distributions at a class of service centers. In addition, we obtain a Little's Law type of relation between delay moments and queue size factorial moments. For a class of networks, an algorithm is given for the direct, recursive computation of the tail of the node delay distribution.*

### I. INTRODUCTION

Performance analysis and modeling activities are essential for answering key questions at various stages of a computer system's life-cycle, ranging from initial conception to maintaining and growing a mature system. Although both the questions asked and the analysis approaches may differ from stage to stage, in each of these stages broadly applicable performance analysis tools are needed to support

such activities. This paper presents new results which contribute to the foundation of one such tool: algorithmic techniques for efficiently solving a class of queueing networks.

In dealing with some performance issues, it is important to be able to quantify distribution or moment information (e.g., delay variability as opposed to only the mean delay)<sup>1</sup> because these quantities can influence system capacity and service and performance measures. It is also important that models include the effect of congestion adaptive I/O devices, in a stable and efficient manner, for this inclusion can significantly affect the outcome of studying certain performance issues (e.g., the impact of multiprogramming).<sup>2,3</sup>

In this paper, we present results for the direct recursive computation of moments of the queue size distributions at a class of service centers embedded in a mixed network of queues. The class of service centers allows us to efficiently treat, in a stable manner, a parameterized class of state-dependent processing rates useful in modeling congestion adaptive I/O devices.<sup>3,4,5</sup> By dealing with mixed systems, we allow consideration of systems with workloads from a finite population (e.g., a collection of terminals), multiprogrammed systems, together with workloads from basically infinite customer populations. We present results for calculating moments of the waiting time and virtual delay distributions at a class of service centers that enable us to quantify the variability of node delays, as well as work backlogs.

The well-known class of multiple resource models, usually referred to as product form queueing networks,<sup>6</sup> have been used to address a wide range of performance issues, such as capacity estimation and planning, bottleneck identification, performance prediction,<sup>7</sup> memory interference, and software lockout in multiprocessor systems.<sup>8</sup> Usually the models used to address these issues have approximately included the factors of interest (e.g., priority processor scheduling disciplines).<sup>9</sup> A considerable amount of effort has been devoted to the study of this class of queueing networks, and efficient computational algorithms exist<sup>1,8,10</sup> that allow one to obtain, for example, mean values of the desired quantities. While, in principle, the entire network queue size state description can be obtained from the above, one may be interested in obtaining results, directly, for a more moderate level of detail, e.g., moments of queue sizes, as well as in quantifying variability of delays at a network node. Existing algorithms for calculating even only mean values can become much more complex and sometimes exhibit chaotic behavior.<sup>11</sup> This situation arises, for example, when a state-dependent service rate is used to model a class of devices, such as an efficiently scheduled disk or drum,<sup>3,4,5\*</sup> whose efficiency is a function

---

\* See Ref. 5 for examples of secondary storage units that employ a scheduling algorithm which attempts to minimize rotational latency and/or seek times.

of the number of queued requests. The complexity results from a requirement that the entire\* marginal queue size distribution is needed at each step of the computation.<sup>1</sup>

The results in this paper overcome some deficiencies in present methods for analyzing networks of queues and enable us to quantify important performance measures for which, previously, no efficient computational methods existed. In Section 1.1, we discuss how a tool for evaluating networks of queues can fit into a performance analysis and modeling methodology, and in Section 1.2, we give a more specific definition of the problems treated and outline the remaining sections of this paper.

### **1.1 Use of network of queues models**

To illustrate the aforementioned need for broadly applicable tools and how a queueing network analysis tool can fit into a performance analysis and modeling methodology, we consider the various stages of a system's life cycle. During the system's conception stage, where one is concerned with the services to be offered, broad objectives—performance, reliability, etc.—basic architecture, proposed components and initial sizing, questions involving initial feasibility arise. Initial feasibility studies generally require several tools to address the following type of question: Given a proposed architecture and assumptions concerning the way the system is to be used (e.g., obtained from a gross workload characterization tool which may indicate various usage scenarios), how well can the system be expected to perform? The answer to this question often leads to a modification of the proposed architecture and/or what the system is planned to do. During this stage, where system specification is often at a macroscopic level of detail, a tool based on a network-of-queues methodology can be useful for the performance prediction. Here, for example, use of a tool incorporating a central-server<sup>12</sup> queueing network model can be helpful in answering questions such as the effects of CPU size, number of disks, and the number of terminals that can be supported, while meeting broad performance objectives, as well as estimates of cost. The particular form of input that may be available from a workload characterization is particularly suitable for use by a network-of-queues methodology at this stage.

During a system design phase (when one is attempting design optimization), one may tend to focus more on subsystems, and models would tend to include more microscopic details, such as a disk schedule model or a CPU process schedule model. Although this stage generally requires more detail than is attainable with a network-of-queues model,

---

\* Some simplifications result when the state dependence disappears above a given loading,<sup>1,10</sup> the so-called limited queue-dependent server.

issues such as the effect of the degree of multiprogramming\*<sup>2</sup> and the effect of a disk schedule which uses a SCAN<sup>†</sup> algorithm versus a FIFO discipline, can be treated. We note that at this stage it is important to have models which include, at least parametrically, the effect of a device being congestion adaptive since this can significantly influence the performance that can be expected.<sup>‡</sup> In this stage, distributional information can also be important, as opposed to just dealing with mean values, since performance criteria and service objectives may be in terms of the tail of a distribution.

During the stage when one is dealing with and maintaining a mature system—adding features, growing, doing capacity estimation and planning—a tool or methodology for viewing the overall system is desirable. Here, measurement tools, providing resource utilizations, workload characterizations, and response times, together with proposed changes in system use and growth, can be used with an overall system performance model to answer such questions as Is the system adequate? How much more load can it handle? or, What system changes are required to handle a further load increase? A capacity planning methodology which accepts, say, maximum allowable utilizations on various system resources and measurements of current utilizations and workloads, and predicts allowable increases in load could use the output of a network-of-queues tool to determine what the maximum resource utilizations should be. We note that the determination of these levels, and thus the system capacity, could very well be based on distribution information.

## **1.2 Outline of paper and summary**

In this paper we address the problem of recursively computing moments of the queue size distribution at a service center embedded in a class of product form networks. Parameterizing the service center with respect to its state dependence allows us to treat, in an efficient manner, congestion adaptive models with either improved or degraded<sup>§</sup> efficiency. We treat mixed systems where the population corresponding to a given job type may be fixed (i.e., a closed chain) and where system requests corresponding to other job types may arrive exogenously from basically an infinite population (i.e., an open chain). Closed chains

---

\* Reference 2 also shows that the multiprogramming effect, representing an increase in potential throughput of up to 75 percent, can depend strongly on the disk scheduling algorithm.

<sup>†</sup> Actually Ref. 2 considers the LOOK algorithm which is similar to SCAN.

<sup>‡</sup> Reference 3, which presents a performance comparison of two I/O access disciplines, notes that the relative comparisons, representing capacity improvements in excess of 100 percent, can depend strongly on whether the I/O device is congestion adaptive (e.g., a fixed head drum employing the shortest access time first schedule) or congestion independent (e.g., FIFO).

<sup>§</sup> Resulting, for example, from increased overhead with increased processor queuing.

arise when dealing with finite population models where, for example, a finite number of terminals places requests into a computer system. They also arise in modeling multiprogrammed systems where the size of the finite population corresponds to the degree of multiprogramming.

Figure 1 shows an example of a mixed system where the closed chains correspond to each of the terminal groups, each group with possibly different think-time distributions and routing and service requirements (workloads). Figure 2 shows a closed network model representing requests arriving to a system,  $S$ , over a finite collection of access trunk groups. A request is blocked if all trunks in the group are occupied. Note that a trunk is held during the entire time the request is in  $S$ . When a trunk is available in group  $i$ , requests enter  $S$  from group  $i$  at rate  $\lambda_i$  (the Poisson arrival rate to group  $i$ ). The population of chain  $i$  is  $K_i$ , the size of trunk group  $i$ , and the blocking probability at group  $i$  corresponds to  $1$ -utilization of the node with service rate  $\lambda_i$ . In addition to throughputs and resource utilizations as the usual quantities of interest, other quantities of interest may be queue size moments, and the distribution, or moments, of the time from trunk activation to its first entry into the CPU (reaction time).

We treat the problem of recursively calculating moments of the waiting time and virtual delay distributions at first-come-first-served (FCFS) state-independent service centers, which enables us to get a

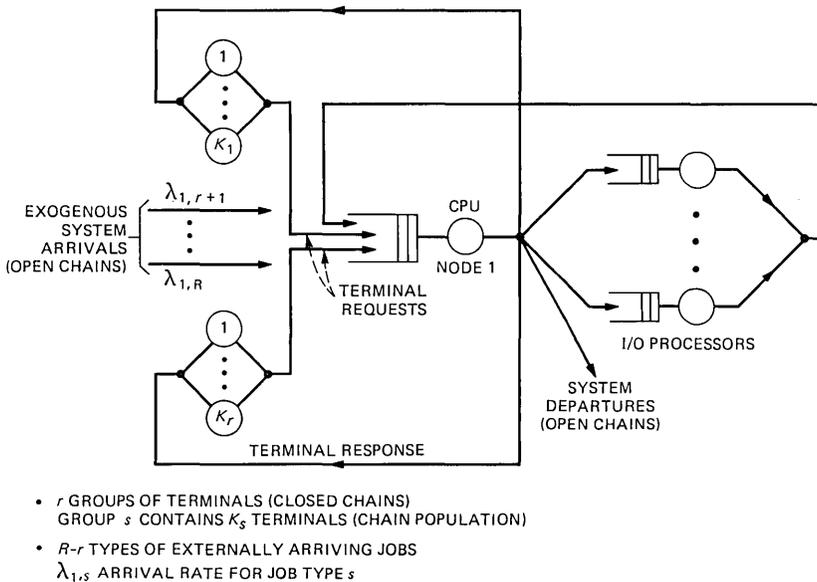


Fig. 1—Mixed system illustration.

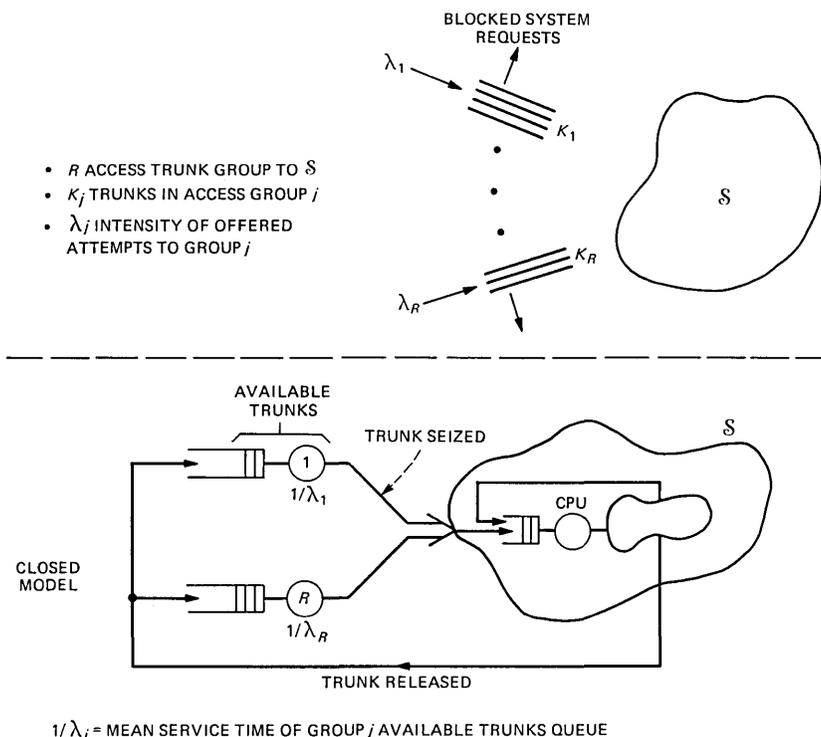


Fig. 2—Closed system illustration.

handle on the variability of nodal delays and to estimate the distribution of such delays. With knowledge of the virtual delays, we can compare performance, for example, as viewed by arriving customers to that viewed by an outside observer. We can thus estimate performance measures such as the system reaction time<sup>13</sup> distribution, which is a measure of the time between a terminal request and this request first getting the attention of the CPU. In another application, it could correspond to the time between a request coming into a system over an access trunk group and the start of processing.

In addition to obtaining algorithmic results, we obtain a relation between moments of node delays and queue sizes at a class of FCFS service centers. We finish with an algorithm for the recursive computation of the tail of the nodal delay distribution at a FCFS service center embedded in a network consisting of either single-server, state-independent nodes or infinite server nodes, e.g., the central server model of multiprogramming.<sup>12</sup>

The organization of this paper is as follows: In Section II, we present the class of networks under consideration, including specification of

the job-type characteristics and specification of the class of service centers. In Section III, we consider closed systems and present results for the factorial moments of queue size distributions, for different levels of customer aggregations. The aggregations considered are the total number of jobs of a given type at a network node and total number of jobs of all types at a network node, the former requiring consideration of joint factorial moments and correlations. Results for mixed systems are given in Section IV by considering a mixed system as the limiting case of a multi-job-type closed system, formed by augmentation, as the population increases and the augmented node becomes the bottleneck.

The recursions for nodal delay and flow time moments are presented in Section V for closed systems and in Section VI for mixed systems. Delay results are obtained as experienced by arriving customers or as experienced by an outside observer i.e., the virtual delay. Delay distribution results appear in Section VII, and the appendices contain details of the investigations.

## II. CLASS OF NETWORKS: SERVICE CENTER AND JOB CHARACTERISTICS

We define the structure of the networks in terms of the types of service centers or nodes and the job-type characteristics in terms of their routing through the network, their service requirements per service center visit,\* their populations for closed job types, and their exogenous arrival rates for open job types.

We consider a network with  $R$  types of customers (jobs, chains) where  $s = 1, 2, \dots, r$  correspond to closed chains and  $s = r + 1, \dots, R$  correspond to open chains. The nodes in the network are either single-server nodes, using a FCFS,<sup>†</sup> processor sharing or last-come-first-served preemptive resume (LCFS-PR) queueing discipline or infinite server nodes. At the single-server nodes, the processing rate can depend on the total number of customers present.

Customers of a given job type (chain) may change class membership as they traverse the network but always remain in the same chain. Allowing customer class change allows one to model a broad class of routing scenarios including a deterministic, fixed sequence of node visits and also allows for different visits to, for example, a processor

---

\* Job characteristics can also be defined in terms of the workload requirements, the average resource usage/job lifetime for each resource where job lifetime is suitably defined for closed customer types.

<sup>†</sup> Other types of customer selection rules, not depending on actual customer service requirements, (e.g., random selection, LCFS nonpreemptive) can also result in product form.<sup>14</sup>

sharing node by a given customer to have different service requirement distributions.

For the open chains,  $s = r + 1, \dots, R$ , we let  $\lambda_{i,0,c,s}$  = rate of exogenous arrivals of chain  $s$  customers, with class membership  $c$ , to node  $i$ . Exogenous arrivals of customers, of a given chain-class pair, to a particular node are assumed to be Poisson.\*

If we let  $p_{kl,ic}^s$  = the probability that a chain  $s$  job completing service at node  $k$  as a class  $l$  customer will next enter node  $i$  with class membership  $c$ , then the actual node-class-chain flow rates  $\lambda_{i,c,s}$  satisfy the so-called traffic equation,

$$\lambda_{i,c,s} = \lambda_{i,0,c,s} + \sum_{(k,l) \in I_s} \lambda_{k,l,s} p_{kl,ic}^s; \\ s = r + 1, \dots, R; \quad (i, c) \in I_s, \quad (1)$$

where  $I_s$  is the set of feasible node-class pairs for chains  $s$  jobs.

We note that for each closed chain, the node-class-chain rates satisfy

$$\lambda_{i,c,s} = \sum_{(k,l) \in I_s} \lambda_{k,l,s} p_{kl,ic}^s; \\ s = 1, \dots, r; \quad (i, c) \in I_s. \quad (2)$$

The closed chains are further specified by their population,

$$\mathbf{K} = (K_1, K_2, \dots, K_r), \quad (3)$$

where  $K_s$  denotes the system population corresponding to the closed chain  $s$ . We denote the rate of customers of a given chain,  $s$ , flowing into a node as

$$\lambda_{i,s} = \sum_{c \in C_i(s)} \lambda_{i,c,s}, \quad (4)$$

where

$$C_i(s) = [c; (i, c) \in I_s].$$

We assume that for each open chain the traffic equation (1) has a unique solution<sup>†</sup> for the traffic rates

$$(\lambda_{i,c,s}; (i, c) \in I_s); \quad s = r + 1, \dots, R,$$

and that every closed chain has an irreducible routing matrix so that

\* This does not preclude having a large class of state-dependent arrival processes, including examples where customer sources are turned off or blocked when their system population reaches a threshold. These can easily be transformed into a mixed network of the type being considered.

<sup>†</sup> This precludes certain pathological cases where customers enter the system and never exit.

the solution to (2) is unique up to a scalar.\* Thus, the actual flow rates, for a given job type, can be specified by an arbitrary solution to (2), yielding relative arrival rates for classes in a closed chain, and a proportionality constant.

We note that the actual proportionality constant, for a given closed chain, say  $\bar{\lambda}_s(\mathbf{K})$ , depends on the population vector  $\mathbf{K}$  and that the actual flow rates,  $\bar{\lambda}_{ics}(\mathbf{K})$ , can be written as

$$\bar{\lambda}_{ics}(\mathbf{K}) = \bar{\lambda}_s(\mathbf{K})\lambda_{ics}, \quad (i, c) \in I_s. \quad (5)$$

Thus, the actual node-chain flow rates  $\bar{\lambda}_{is}(\mathbf{K})$  can be written as

$$\bar{\lambda}_{is}(\mathbf{K}) = \bar{\lambda}_s(\mathbf{K})\lambda_{is}, \quad (6)$$

where  $\lambda_{is}$  is given by (4).

The assumptions on the service time distributions are those for which the product-form solution holds (see Ref 6). We denote

$\mu_{ics}^{-1}$  = mean amount of service required by a chain  $s$  class  $c$  customer at node  $i$  when only one customer is present at node  $i$  (note that at a FCFS node it is required that  $\mu_{ics} = \mu_{ief}$  for all  $c, s, e$  and  $f$ ),

and

$\mu_i(k)$  = the processing or service rate of node  $i$  when a total of  $k$  customers (regardless of type) are present.†

Clearly, the average service requirement of an arbitrary, with respect to class, chain  $s$  customer, denoted  $\mu_{is}^{-1}$ , is given by the weighted average of the individual class average service requirements.

$$\mu_{is}^{-1} = \sum_{c \in C_i(s)} \frac{\lambda_{ics}}{\lambda_{is}} \mu_{ics}^{-1}. \quad (7)^\ddagger$$

While we allow the network to contain nodes with general state-dependent processing rates, we focus on getting queue size moments at nodes with a parameterized class of processing rates defined by

---

\* This follows from the assumption of irreducibility of the routing matrix

$$P^s = [p_{kl,ic}^s]; \quad kl, ic \in I_s$$

and precludes the pathological case where the set of communicating node-class pairs can be decomposed into disjoint subsets. We note that the matrix can be made irreducible by using enough chains.

† For example, the number of seconds of processing per second of elapsed time. Without loss of generality we have let  $\mu_i(1) = 1$ .

‡ Recall that these are not adjusted for state-dependent processing rates. Another interpretation of (7) is that  $\mu_{is}^{-1}$  represents the average service time of an arrival to node  $i$  in a system whose only customer is a single chain  $s$  customer.

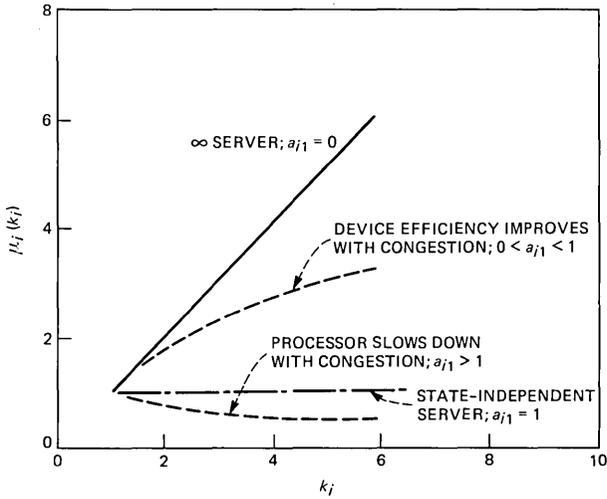


Fig. 3—A class of state-dependent service rates.

$$\mu_i(k) = \frac{k}{a_{i0} + a_{i1}k}, \quad (8)^*$$

where  $a_{i1} \geq 0$ ;  $a_{i0} + a_{i1} = 1$ . We note that  $a_{i1} = 1$  corresponds to a single server with state-independent processing rate, while  $a_{i1} = 0$  corresponds to an infinite-server node, often useful in modeling a finite population source—a collection of terminals—or a random delay. These are shown in Fig. 3, along with cases  $0 < a_{i1} < 1$  and  $a_{i1} > 1$  that can be used to represent congestion adaptive devices. We also note that the family of service rates may also be useful in approximating subsystems with restricted entry<sup>15,16</sup> by a single state-dependent node as is done in multiprogramming models.<sup>17</sup> What is needed here is the subsystem throughput as a function of the sizes of its populations, and recursive methods are naturally suitable for obtaining this.

### III. QUEUE SIZE FACTORIAL MOMENTS—CLOSED SYSTEMS ( $R = r$ )

We present recursions for two different levels of customer aggregation: (i) for the total number of jobs of all types at a network node, and (ii) for the total number of jobs of a given type, the latter requiring consideration of joint factorial moments and correlations.

We denote

\* The methods of this paper can be extended to include a generalization of this family of processing rates. The above family provides a good fit to the empirically obtained (by R. J. T. Morris) state dependence used in Ref. 2 for an efficiently scheduled (similar to SCAN) moving head disk over the multiprogramming range considered.

$$\beta_{ij}(\mathbf{K}) = E[k_i(k_i - 1) \cdots (k_i - j + 1); \mathbf{K}] \quad (9)$$

as the  $j$ th factorial moment of the total number of customers,  $k_i$ , at node  $i$  at an arbitrary time in equilibrium, for a system with population vector  $\mathbf{K}$ . Appendix A shows that  $\beta_{ij}(\mathbf{K})$  satisfies the following recursions

$$\beta_{ij}(\mathbf{K}) = \sum_{s=1}^R \frac{\bar{\lambda}_{is}(\mathbf{K})}{\mu_{is}} \{a_{i1}\beta_{ij}(\mathbf{K} - \mathbf{e}_s) + [1 + (j - 1)a_{i1}]\beta_{i,j-1}(\mathbf{K} - \mathbf{e}_s)\}, \quad (10)^*$$

with initialization  $\beta_{ij}(\mathbf{0}) = 0; j > 0$  and  $\beta_{i0}(\mathbf{K}) = 1$ . The quantity  $\mathbf{e}_s$  is a unit vector in direction  $s$ . We thus obtain the  $j$ th moment at population  $\mathbf{K}$  by updating the  $j$ th moments corresponding to systems with one less customer for each chain in addition to similarly including the effect of the  $(j - 1)$ st moments. The node-chain throughputs,  $\bar{\lambda}_{is}(\mathbf{K})$ , are available via standard mean-value analysis;<sup>1</sup> however, the standard analysis requires computations<sup>†</sup> of marginal probabilities at nodes with state-dependent processing rates. For networks, with nodes of the type under consideration, we can generalize mean-value analysis, in a stable manner, without ever computing marginal probabilities and, furthermore, obtain the higher order moments. When  $j = 1$ , (10) yields the generalization

$$\beta_{i1}(\mathbf{K}) = \sum_{s=1}^R \frac{\bar{\lambda}_{is}(\mathbf{K})}{\mu_{is}} [a_{i1}\beta_{i1}(\mathbf{K} - \mathbf{e}_s) + 1]. \quad (11)$$

Defining the mean node flow time for a chain  $s$  customer as the mean time a chain  $s$  customer spends at node  $i$  per visit<sup>‡</sup> to the node  $i$  queue,  $T_{is}(\mathbf{K})$ , Appendix A shows that

$$T_{is}(\mathbf{K}) = \frac{1}{\mu_{is}} [1 + a_{i1}\beta_{i1}(\mathbf{K} - \mathbf{e}_s)] \quad (12)$$

The mean number of chain  $s$  customers at node  $i$

$$\beta_{i,1,s}(\mathbf{K}) = E(k_{is}; \mathbf{K})$$

satisfies

$$\beta_{i1s}(\mathbf{K}) = \bar{\lambda}_{is}(\mathbf{K})T_{is}(\mathbf{K}), \quad (13)^{\S}$$

where the  $(i, s)$  throughput satisfies [see (6)]

\* In all recursions, terms with a negative population component are zero, e.g.,  $\mathbf{K} - \mathbf{e}_s$  with  $\mathbf{K} = \mathbf{0}$ .

† Those computations can become unstable.<sup>11,18</sup>

‡ If a customer is fed back to the same queue after a service completion, as in the central server model of multiprogramming, this corresponds to starting a new flow time.

§ The relation with (12) is via Little's Law for chain  $s$  customers at node  $i$ .

1. Initialize  $\beta_{ij}(\mathbf{0}) = 0 \ j > 0, \beta_{i0}(\mathbf{0}) = 1$ .
2. Loop on  $\mathbf{K}$  until desired population  $\mathbf{K}^*$ .
3. Compute node flow time means,  $T_{is}(\mathbf{K})$ , from (12),  $i \in N(s), s = 1, 2, \dots, R$ .
4. Compute throughput proportionality constants,  $\bar{\lambda}_s(\mathbf{K})$  from (15),  $s = 1, 2, \dots, R$ .
5. Compute node-chain throughputs,  $\bar{\lambda}_{is}(\mathbf{K}), i \in N(s), s = 1, 2, \dots, R$ , from (14).
6. Compute  $\beta_{i1s}(\mathbf{K})$  from (13),  $i \in N(s), s = 1, 2, \dots, R$  and  $\beta_{j1}(\mathbf{K})$  from (16).
7. If do not desire  $\beta_{ij}(\mathbf{K}) \ j > 1$  or  $\beta_{ijs}(\mathbf{K}) \ j > 1$ , go to 2.
8. If desire  $\beta_{ij}(\mathbf{K}) \ j > 1$ , compute  $\beta_{ij}(\mathbf{K})$  from (10) at desired nodes for  $1 < j < J$ .  
If do not want  $\beta_{ijs}(\mathbf{K}) \ j > 1$ , go to 2.
9. If desire  $\beta_{ijs}(\mathbf{K}^*)$ : If  $\mathbf{K} = \mathbf{K}^* - j\mathbf{e}_s$ , initialize  $\gamma_{i,0,\ell,s}(\mathbf{K}) = \beta_{j\ell}(\mathbf{K}), \ell = 0, 1, \dots, J$ .  
If  $\mathbf{K} = \mathbf{K}^* - n\mathbf{e}_s, n < J$ , compute  $\gamma_{i,j-n,\ell,s}(\mathbf{K})$  from (21) as shown in Fig. 5. Go to 2.

Fig. 4—Algorithm for queue size moments—closed systems.

$$\bar{\lambda}_{is}(\mathbf{K}) = \lambda_{is} \bar{\lambda}_s(\mathbf{K}). \quad (14)$$

In terms of the flow times, the chain  $s$  proportionality constant is

$$\bar{\lambda}_s(\mathbf{K}) = \frac{K_s}{\sum_{i \in N(s)} \lambda_{is} T_{is}(\mathbf{K})}; \quad s = 1, 2, \dots, R, \quad (15)$$

where  $N(s)$  is the set of nodes visited by chain  $s$  customers. To close the algorithmic loop, we use

$$\beta_{i1}(\mathbf{K}) = \sum_{s=1}^R \beta_{i1s}(\mathbf{K}). \quad (16)$$

The algorithm (12)  $\rightarrow$  (15)  $\rightarrow$  (14)  $\rightarrow$  (13)  $\rightarrow$  (16) (see Fig. 4), with initial condition  $\beta_{i1}(\mathbf{0}) = 0$  and with the relative traffic rates,  $\lambda_{is}$ , as in the previous section,\* represents a simple, stable,<sup>†</sup> modification of mean value analysis for the desired class of state dependencies. With the node-chain throughputs from (14), the aggregate higher order factorial moments are obtained, in a numerically stable manner, by (10). We note that, unlike the mean values, it is only necessary to compute higher order moments at those nodes of interest. Thus, at the desired node we have the algorithm (see Fig. 4) (12)  $\rightarrow$  (15)  $\rightarrow$  (14)  $\rightarrow$  (13)  $\rightarrow$  (16)  $\rightarrow$  (10). We now turn our attention to obtaining the higher order moments for the lower level of aggregation.

To obtain the higher order ( $>1$ ) factorial moments of the number of each type of customer at a node

\* We recall the relative traffic levels need not be computed for each population vector since they do not depend on  $\mathbf{K}$ .

<sup>†</sup> Note that no subtractions appear in the computations and recall  $a_{i1} \geq 0$ .

$$\beta_{i,j,s}(\mathbf{K}) = E[k_{is}(k_{is} - 1) \cdots (k_{is} - j + 1); \mathbf{K}], \quad (17)$$

we consider the joint factorial moments

$$\gamma_{i,j,l,s}(\mathbf{K}) = E[k_{is}(k_{is} - 1) \cdots (k_{is} - j + 1) \cdot k_i(k_i - 1) \cdots (k_i - l + 1); \mathbf{K}]. \quad (18)$$

Note that

$$\gamma_{i,0,l,s}(\mathbf{K}) = \beta_{i,l}(\mathbf{K}), \quad (19)$$

the high level  $l$ th factorial moment defined by (9) and computable from (10), and

$$\gamma_{i,j,0,s}(\mathbf{K}) = \beta_{i,j,s}(\mathbf{K}) \quad (20)$$

the desired low level  $j$ th factorial moment.

From Appendix A, we have the recursion

$$\begin{aligned} \gamma_{i,j,l,s}(\mathbf{K}) = & \frac{\bar{\lambda}_{is}(\mathbf{K})}{\mu_{is}} \{a_{i1}\gamma_{i,j-1,l+1,s}(\mathbf{K} - \mathbf{e}_s) \\ & + (1 + 2la_{i1})\gamma_{i,j-1,l,s}(\mathbf{K} - \mathbf{e}_s) + [1 + (1 + a_{i1})(l - 1) \\ & + a_{i1}(l - 1)^2]\gamma_{i,j-1,l-1,s}(\mathbf{K} - \mathbf{e}_s)\}; \quad j > 0. \end{aligned} \quad (21)$$

For  $j = 0$  the computation is made via (10) and the identification (19). We note that for  $l = 0$ ,  $j = 1$  (21)<sup>†</sup> reduces to (13). Making the identification (20), gives

$$\beta_{ijs}(\mathbf{K}) = \frac{\bar{\lambda}_{is}(\mathbf{K})}{\mu_{is}} [\beta_{i,j-1,s}(\mathbf{K} - \mathbf{e}_s) + a_{i1}\gamma_{i,j-1,1,s}(\mathbf{K} - \mathbf{e}_s)]. \quad (22)$$

We note that if we are interested in the  $J$ th factorial moment of the number of chain  $s$  customers at node  $i$  for a target population  $K^*$ , i.e.,  $\beta_{iJ,s}(K^*)$ , then (21) and (22) are initialized at population  $\mathbf{K}^* - J\mathbf{e}_s$  with

$$\begin{aligned} \gamma_{i,0,l,s}(\mathbf{K}^* - J\mathbf{e}_s) &= \beta_{i,l}(\mathbf{K}^* - J\mathbf{e}_s), \\ l &= 0, 1, \dots, J, \end{aligned} \quad (23)$$

the simple high-level aggregation result. We further note that (21) and (22) only need be updated along parameter direction  $\mathbf{e}_s$ . This is shown schematically in Fig. 5 and the steps in the algorithm in Fig. 4. If, for example, one is interested in a mean and variance analysis, this specializes to

$$\beta_{i2s}(K^*) = \frac{\bar{\lambda}_{is}(K^*)}{\mu_{is}} [\beta_{i1s}(K^* - \mathbf{e}_s) + a_{i1}\gamma_{i,1,1,s}(K^* - \mathbf{e}_s)], \quad (24)$$

<sup>†</sup> Terms with a negative subscript are zero.

- TO GET  $\beta_{iJ_s}(\mathbf{K}^*) = \gamma_{iJ_s}(\mathbf{K}^*)$
- DENOTE  $\gamma_{ij\ell_s}$  BY  $(i, \ell)$

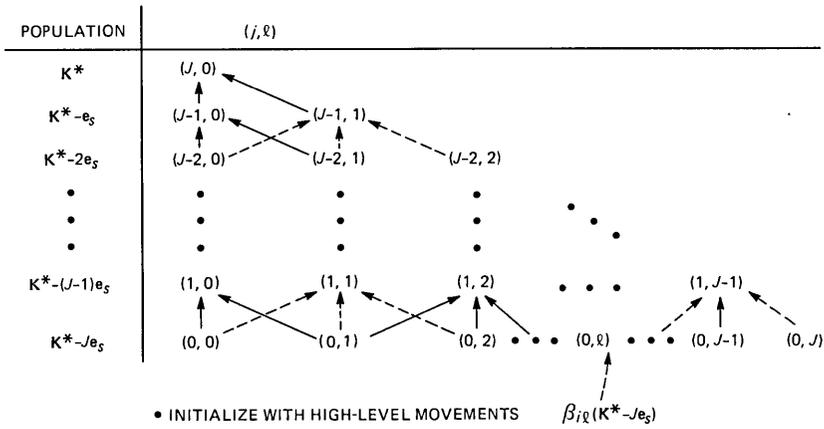


Fig. 5—Schematic for moments of low-level aggregation—closed systems.

where the correlation is given by

$$\gamma_{i,1,1,s}(\mathbf{K}^* - \mathbf{e}_s) = \frac{\bar{\lambda}_{is}(\mathbf{K}^* - \mathbf{e}_s)}{\mu_{is}} [a_{i1}\beta_{i2}(\mathbf{K}^* - 2\mathbf{e}_s) + (1 + 2a_{i1})\beta_{i1}(\mathbf{K}^* - 2\mathbf{e}_s) + 1]. \quad (25)$$

#### IV. QUEUE SIZE FACTORIAL MOMENTS—MIXED SYSTEMS

To obtain the mixed network results, we consider an augmented closed system which contains an external node for each open chain. The service rate of an external node corresponds to the system arrival rate for the corresponding open chain, i.e.,  $\lambda_{0s} = \sum_{i \in N_s} \lambda_{i,0,s}$ ;  $s = r + 1, \dots, R$ . When customers depart in the original system, they are routed to the appropriate external node in the augmented system. A departure from a given external node, say, corresponding to the open chain  $s$ , is routed to node  $i$  as a class  $c$  customer with probability  $\lambda_{i0cs} / \lambda_{0s}$ . Taking the limit of the closed network results as the populations of chains  $r + 1, \dots, R \rightarrow \infty$ , and assuming the external nodes become the bottlenecks,<sup>†</sup> we obtain the desired recursion for the high-level aggregate factorial moments

<sup>†</sup> Note that if an internal node becomes a bottleneck, the original mixed system is unstable.

$$\beta_{ij}(\mathbf{K}) = [1+(j-1)a_{i1}] \frac{\rho_i^0}{1 - a_{i1}\rho_i^0} \beta_{i,j-1}(\mathbf{K}) + \sum_{s=1}^r \frac{\bar{\lambda}_{is}(\mathbf{K})}{\mu_{is}(1 - a_{i1}\rho_i^0)} \{a_{i1}\beta_{ij}(\mathbf{K} - \mathbf{e}_s) + [1 + (j - 1)a_{i1}]\beta_{i,j-1}(\mathbf{K} - \mathbf{e}_s)\}, \quad (26)$$

where

$$\rho_i^0 = \sum_{s=r+1}^R \frac{\lambda_{is}}{\mu_{is}} \quad (27)$$

is the unadjusted utilization [unadjusted for state-dependent service rates  $\mu_i(k) \neq 1$ ] at node  $i$ , corresponding to all open chains. We note that the stability condition is given by

$$\rho_i^0 a_{i1} < 1, \quad (28)$$

where we note that  $\rho_i^0 a_{i1}$  is the limiting (as  $k_i \rightarrow \infty$ ) utilization due to customers belonging to open chains. Equation (26) is initialized by

$$\beta_{i,0}(\mathbf{K}) = 1$$

and

$$\beta_{ij}(\mathbf{0}) = \frac{[1 + (j - 1)a_{i1}]\rho_i^0}{1 - a_{i1}\rho_i^0} \beta_{i,j-1}(\mathbf{0}), \quad (29)$$

the open network factorial moments. For  $j = 1$ , (26) yields

$$\beta_{i1}(\mathbf{K}) = \frac{\rho_i^0}{(1 - a_{i1}\rho_i^0)} + \sum_{s=1}^r \frac{\bar{\lambda}_{is}(\mathbf{K})}{\mu_{is}(1 - a_{i1}\rho_i^0)} [1 + a_{i1}\beta_{i1}(\mathbf{K} - \mathbf{e}_s)], \quad (30)$$

with initial condition

$$\beta_{i1}(\mathbf{0}) = \frac{\rho_i^0}{1 - a_{i1}\rho_i^0}. \quad (31)$$

The required node-chain throughputs,  $\bar{\lambda}_{is}(\mathbf{K})$ , in (26) can be obtained via a standard type of mean analysis. However, as before, we are interested in a generalized analysis that does not involve marginal probabilities. These throughputs are obtained via the limiting argument which yields the recursions

$$\beta_{i1s}(\mathbf{K}) = \frac{\bar{\lambda}_{is}(\mathbf{K})}{\mu_{is}} [1 + a_{i1}\beta_{i1}(\mathbf{K} - \mathbf{e}_s)]; \quad s \leq r \quad (32)$$

for the closed chains and

$$\beta_{i1s}(\mathbf{K}) = \frac{\lambda_{is}}{\mu_{is}} [1 + a_{i1}\beta_{i1}(\mathbf{K})]; \quad s > r, \quad (33)$$

for the open chains. We note that in (32),  $\bar{\lambda}_{is}(\mathbf{K})$ , the closed chain  $s$ -

node  $i$  throughput, is to be solved for, whereas in (33),  $\lambda_{is}$  is simply the solution to the traffic equation for open chain  $s$ . Summing (33) over all chains, we obtain

$$\beta_{i1}^o(\mathbf{K}) = \frac{\rho_i^o}{1 - \alpha_{i1}\rho_i^o} [1 + \alpha_{i1}\beta_{i1}^c(\mathbf{K})], \quad (34)$$

where

$$\beta_{i1}^o(\mathbf{K}) = \sum_{s=r+1}^R \beta_{i1s}(\mathbf{K}), \quad (35)$$

$$\beta_{i1}^c(\mathbf{K}) = \sum_{s=1}^r \beta_{i1s}(\mathbf{K}), \quad (36)$$

which relates the mean value of the total number of open customers at node  $i$  to the mean value of the total number of closed customers at node  $i$ . To obtain the desired recursion for  $\beta_{i1s}$ ,  $s \leq r$  we use (34) in (32), which results in

$$\beta_{i,1,s}(\mathbf{K}) = \frac{\bar{\lambda}_{is}(\mathbf{K})}{\mu_{is}(1 - \alpha_{i1}\rho_i^o)} [1 + \alpha_{i1}\beta_{i1}^c(\mathbf{K} - \mathbf{e}_s)], \quad s \leq r; \quad (37)$$

and from Little's Law, the mean chain  $s$  flow times

$$T_{is}(\mathbf{K}) = \frac{1}{\mu_{is}(1 - \alpha_{i1}\rho_i^o)} [1 + \alpha_{i1}\beta_{i1}^c(\mathbf{K} - \mathbf{e}_s)], \quad s \leq r. \quad (38)$$

The node chain throughputs are calculated from

$$\bar{\lambda}_{is}(\mathbf{K}) = \lambda_{is}\bar{\lambda}_s(\mathbf{K}), \quad s \leq r, \quad (39)$$

where,  $\bar{\lambda}_s(\mathbf{K})$ , the chain  $s$  proportionality constant (independent of  $i$ ) is given by

$$\bar{\lambda}_s(\mathbf{K}) = \frac{K_s}{\sum_i \lambda_{is} T_{is}(\mathbf{K})}; \quad s \leq r. \quad (40)$$

The algorithm,

$$(38) \rightarrow (40) \rightarrow (39) \rightarrow (37) \rightarrow (36),$$

(see Fig. 6) with initial condition

$$\beta_{i1}^c(\mathbf{0}) = 0,$$

is identical to the algorithm (see Fig. 4) for closed systems,

$$(12) \rightarrow (15) \rightarrow (14) \rightarrow (13) \rightarrow (16).$$

However, the closed chain service rates have been adjusted to account for the presence of the open chain customers, i.e.,

$$\mu'_{is} = \mu_{is}(1 - \alpha_{i1}\rho_i^o). \quad (41)$$

1. Initialize  $\beta_{j1}(\mathbf{O})$  from (31) and  $\beta_{ij}(\mathbf{O})$  from (29) for  $1 < j \leq$  desired order moment.  $\beta_{j1}^c(\mathbf{O}) = 0$ .
2. Loop on  $\mathbf{K}$  until desired population.
3. Compute node flow time means,  $T_{is}(\mathbf{K}), s \leq r$  (i.e., for the closed chains) from (38) for  $i \in N(s)$ .
4. Compute throughput proportionality constants  $\bar{\lambda}_s(\mathbf{K})$  from (40) for  $s \leq r$ .
5. Compute node-chain throughputs,  $\bar{\lambda}_{is}(\mathbf{K}), s \leq r$ , from (39),  $i \in N(s)$ .
6. Compute  $\beta_{j1s}(\mathbf{K})$  from (37) for  $s \leq r, i \in N(s)$  and  $\beta_{j1}^c(\mathbf{K})$  from (36).
7. If desire  $\beta_{j1}^o(\mathbf{K})$  or  $\beta_{j1s}(\mathbf{K})$ , for some  $s > r$ , use (34) for  $\beta_{j1}^o(\mathbf{K})$  and (33) with  $\beta_{j1}$  given by the sum of  $\beta_{j1}^c(\mathbf{K})$  and  $\beta_{j1}^o(\mathbf{K})$ .
8. If do not desire  $\beta_{ij}(\mathbf{K}) j > 1$  or  $\beta_{ijs}(\mathbf{K}) j > 1$ , go to 2.
9. If desire  $\beta_{ij}(\mathbf{K}) j > 1$ , compute  $\beta_{ij}(\mathbf{K})$  from (26) for  $1 < j \leq J$ .  
If do not want  $\beta_{ijs}(\mathbf{K}) j > 1$ , go to 2.
10. If desire  $\beta_{ijs}(\mathbf{K}^*) s \leq r$ : if  $\mathbf{K} = \mathbf{K}^* - j e_s$ , initialize  $\gamma_{io\ell s}(\mathbf{K}) = \beta_{j\ell}(\mathbf{K}), \ell = 0, 1, \dots, J$ .  
If  $\mathbf{K} = \mathbf{K}^* - n e_s, n < J$  Compute  $\gamma_{i,j-n,\ell,s}(\mathbf{K})$  from (42) as shown in Fig. 5. Go to step 2.
11. At the desired population  $\mathbf{K}^*$ , if desire  $\beta_{ijs}(\mathbf{K}^*)$  for  $s > r$ , initialize  $\gamma_{io\ell s}(\mathbf{K}^*) = \beta_{j\ell}(\mathbf{K}^*) \ell = 0, 1, \dots, J$ . Compute  $\gamma_{i,j-n,\ell,s}(\mathbf{K}^*)$  from (45) as shown in Fig. 7.

Fig. 6—Algorithm for queue size moments—mixed systems.

We note, however, that the adjustment is not directly related to the actual open-chain utilization (as in the state-independent case<sup>11</sup>), but rather to the limiting open-chain utilization.

The above algorithm, together with (33) and (34) describing the open chains, represents a simple, stable modification to a standard mean value analysis for mixed systems with the desired family of state dependencies.

With the node-chain throughputs from (39), the aggregate higher order factorial moments can be recursively computed in a numerically stable manner from (26). We note that, unlike the mean values, it is only necessary to compute higher order moments at those nodes of interest. Thus, at the desired node we have the algorithm (see Fig. 6)

$$(38) \rightarrow (40) \rightarrow (39) \rightarrow (37) \rightarrow (36) \rightarrow (26).$$

We now turn our attention to obtaining the higher order moments for the lower aggregation level.

To obtain the higher order factorial moments of the number of each type of customer at a node,  $\beta_{ijs}(\mathbf{K})$  defined by (17), we require the joint

factorial moments  $\gamma_{ijls}(\mathbf{K})$  defined in (18), noting the relations (19) and (20). Treating the mixed system as a limiting closed system, we obtain

$$\begin{aligned} \gamma_{i,j,l,s}(\mathbf{K}) = & \frac{\bar{\lambda}_{is}(\mathbf{K})}{\mu_{is}} \{a_{i1}\gamma_{i,j-1,l+1,s}(\mathbf{K} - \mathbf{e}_s) \\ & + (1 + 2la_{i1})\gamma_{i,j-1,l,s}(\mathbf{K} - \mathbf{e}_s) + [1 + (1 + a_{i1})(l - 1) \\ & + a_{i1}(l - 1)^2]\gamma_{i,j-1,l-1,s}(\mathbf{K} - \mathbf{e}_s)\}; \quad j > 0; s \leq r, \quad (42) \end{aligned}$$

which is the same form as (21). For  $l = 0, j = 1$  (42) reduces to (32). For  $j = 0$ , use (19) and (26). Making the identification (20), we get

$$\begin{aligned} \beta_{i,j,s}(\mathbf{K}) = & \frac{\bar{\lambda}_{is}(\mathbf{K})}{\mu_{is}} [\beta_{i,j-1,s}(\mathbf{K} - \mathbf{e}_s) \\ & + a_{i1}\gamma_{i,j-1,1,s}(\mathbf{K} - \mathbf{e}_s)], \quad s \leq r. \quad (43) \end{aligned}$$

If we are interested in  $\beta_{i,J,s}(\mathbf{K}^*); s \leq r$ , then (42) and (43) are initialized at population  $\mathbf{K}^* - J\mathbf{e}_s$  with

$$\gamma_{i,0,l,s}(\mathbf{K}^* - J\mathbf{e}_s) = \beta_{il}(\mathbf{K}^* - J\mathbf{e}_s); \quad l = 0, 1, \dots, J, s \leq r \quad (44)$$

the high-level aggregation result. Thus the computation proceeds as in the closed system case shown in Fig. 5.

For the open chains, the limiting system argument results in the relation

$$\begin{aligned} \gamma_{i,j,l,s}(\mathbf{K}) = & \frac{\lambda_{is}}{\mu_{is}} \{a_{i1}\gamma_{i,j-1,l+1,s}(\mathbf{K}) \\ & + (1 + 2la_{i1})\gamma_{i,j-1,l,s}(\mathbf{K}) + [1 + (1 + a_{i1})(l - 1) \\ & + a_{i1}(l - 1)^2]\gamma_{i,j-1,l-1,s}(\mathbf{K})\} \quad j > 0, s > r. \quad (45) \end{aligned}$$

At  $l = 0, j = 1$  (45) reduces to (33) and for  $j = 0$ , use (19) and (26). We note that (45) is not a recursion in  $\mathbf{K}$  and can be evaluated as an a posteriori computation. Thus, if it is desired to compute  $\beta_{i,J,s}(\mathbf{K}^*); s > r$  at a fixed desired population vector  $\mathbf{K}^*$ , all that is needed is the high-level aggregate  $\beta_{il}(\mathbf{K}^*); l = 0, 1, \dots, J$ , which starts off the (45) computation. This is shown in Fig. 7 and the steps in the algorithm in Fig. 6. As in the previous section, it is a straightforward matter to write down the expressions needed to do a mean and variance analysis.

## V. DELAY AND FLOW TIME MOMENTS—CLOSED SYSTEMS

We consider the problem of obtaining recursions for the moments of the time a customer spends at a FCFS, single-server, state-independent node embedded in a closed network. As earlier, we define the node flow time as the length of time an arriving customer, either arriving from another node or being fed back from the node output, spends

- TO GET  $\beta_{iJs}(\mathbf{K}^*) = \gamma_{iJos}(\mathbf{K}^*) ; s > r$
- DENOTE  $\gamma_{ij\ell s}$  BY  $(i, \ell)$

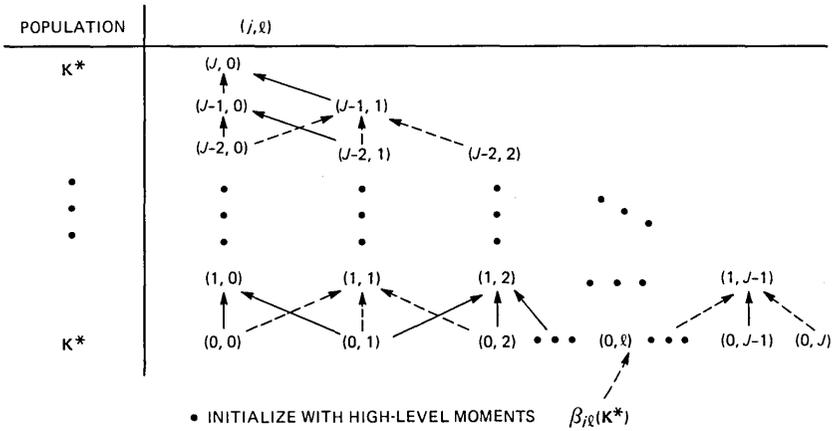


Fig. 7—Schematic for moments of low-level aggregation mixed systems—open chains.

until he next exits the server. The node delay is correspondingly that portion of the flow time spent in the queue. We also present a Little's Law type of relation between moments of the node flow times and queue size moments.

Denoting  $T_{ijs}(\mathbf{K})$  as the  $j$ th moment of the flow time for an arbitrary chain  $s$  customer at node  $i$ , Appendix B shows that

$$T_{ijs}(\mathbf{K}) = \sum_{l=1}^R \frac{\bar{\lambda}_{il}(\mathbf{K} - \mathbf{e}_s)}{\bar{\mu}_i} T_{ijl}(\mathbf{K} - \mathbf{e}_s) + \frac{j}{\bar{\mu}_i} T_{i,j-1,s}(\mathbf{K}); \quad j \geq 1, \quad (46)$$

where the quantity  $\bar{\mu}_i^{-1}$  is the common mean service time at node  $i$  (i.e.,  $\bar{\mu}_i = \mu_{is} = \mu_{il}$ ).<sup>†</sup> From the initial conditions with respect to  $j$

$$T_{i,0,s}(\mathbf{K}) = 1, \quad (47)$$

and (46), we have

$$T_{ijs}(\mathbf{e}_s) = \frac{j!}{\bar{\mu}_i^j}. \quad (48)$$

We note that an alternate form of (46), which only involves the flow time moments corresponding to the job type of interest, is obtained by recognizing that

$$T_{ijl}(\mathbf{K} - \mathbf{e}_s) = T_{ijs}(\mathbf{K} - \mathbf{e}_l).$$

<sup>†</sup> Required at a FCFS node to have a product form solution. Since node  $i$  has a state-independent service rate,  $\mu_i(k) = 1$ .

It can similarly be shown (using the approach discussed in Appendix B) that the  $j$ th moment of the delay distribution corresponding to a chain  $s$  customer satisfies

$$W_{ijs}(\mathbf{K}) = \sum_{l=1}^R \frac{\bar{\lambda}_{il}(\mathbf{K} - \mathbf{e}_s)}{\bar{\mu}_i} \cdot W_{ijl}(\mathbf{K} - \mathbf{e}_s) + \frac{j}{\bar{\mu}_i} W_{i,j-1,s}(\mathbf{K}); \quad j > 1, \quad (49)^*$$

and

$$W_{i1s}(\mathbf{K}) = \sum_{l=1}^R \frac{\bar{\lambda}_{il}(\mathbf{K} - \mathbf{e}_s)}{\bar{\mu}_i} W_{i1l}(\mathbf{K} - \mathbf{e}_s) + \frac{1}{\bar{\mu}_i} \sum_{l=1}^R \frac{\bar{\lambda}_{il}(\mathbf{K} - \mathbf{e}_s)}{\bar{\mu}_i} \quad (50)$$

for  $j = 1$ .

As with the flow time moments, the delay moments at a given population are formed by updating the same order moment for reduced populations, together with including the effect of the lower order moment at the same population. We, thus, note that  $j$ th moment of the delay and node flow time distributions satisfy the same recursions, (46) and (49), for  $j > 1$ , the difference lying in the boundary equation (50), which results in

$$W_{i,j,s}(\mathbf{e}_i) = 0; \quad j > 0. \quad (51)$$

It is *not* necessary, however, to have a separate computation for  $W_{ijs}(\mathbf{K})$  since  $W_{ijs}$  is obtainable from  $T_{ijs}$  by the relation

$$W_{ijs}(\mathbf{K}) = \sum_{l=1}^R \frac{\bar{\lambda}_{il}(\mathbf{K} - \mathbf{e}_s)}{\bar{\mu}_i} T_{ijl}(\mathbf{K} - \mathbf{e}_s); \quad j > 0. \quad (52)^\dagger$$

We note that the distributions of delays, as seen by arriving customers, will usually differ from the work backlog at a node at an arbitrary point in time (i.e., the virtual delay) as might be measured by an outside observer. If we define

$$V_{ij}(\mathbf{K}) = E(W_i^j(t); \mathbf{K}), \quad (53)$$

where  $W_i(t)$  is the work present at node  $i$  at an arbitrary time in equilibrium, as the  $j$ th moment of the virtual delay, then it is shown in the next section that  $V_{ij}(\mathbf{K})$  can be obtained from the moments of customer delay using the relations

\* The computation of these delay moments or the preceding flow time moments are easily included in the algorithm shown in Fig. 4.

† Note that for  $j = 1$ , this simply states the known result  $W_{i1s}(\mathbf{K}) = \beta_{i1}(\mathbf{K} - \mathbf{e}_s)/\bar{\mu}_i$ . Relations (49), (50), and (52) can be written in a form containing only moments corresponding to the job type of interest by using  $W_{ijl}(\mathbf{K} - \mathbf{e}_s) = W_{ijs}(\mathbf{K} - \mathbf{e}_l)$ .

$$V_{ij}(\mathbf{K}) = \sum_{l=1}^R \frac{\bar{\lambda}_{il}(\mathbf{K})}{\bar{\mu}_i} W_{ijl}(\mathbf{K}) + \frac{j}{\bar{\mu}_i} V_{i,j-1}(\mathbf{K}); \quad j > 1 \quad (54)^*$$

and

$$V_{i1}(\mathbf{K}) = \sum_{l=1}^R \frac{\bar{\lambda}_{il}(\mathbf{K})}{\bar{\mu}_i} \left( \frac{1}{\bar{\mu}_i} + W_{i1l}(\mathbf{K}) \right). \quad (55)$$

Having obtained results both for the queue size factorial moments,  $\beta_{ij}(\mathbf{K})$ , in Section III and the ordinary moments of the flow time distribution  $T_{ijs}(\mathbf{K})$  (see above), we now present a relationship between them analogous to a well-known Little's Law type of relationship for the  $M/G/1$  queue in isolation. Appendix B shows that these quantities are related via the  $j$ -fold summation

$$\beta_{ij}(\mathbf{K}) = \sum_{s_1=1}^R \cdots \sum_{s_j=1}^R \bar{\lambda}_{is_1}(\mathbf{K}) \bar{\lambda}_{is_2}(\mathbf{K} - \mathbf{e}_{s_1}) \cdots \bar{\lambda}_{is_j} \left( \mathbf{K} - \sum_{q=1}^{j-1} \mathbf{e}_{s_q} \right) T_{ijs_j} \left( \mathbf{K} - \sum_{q=1}^{j-1} \mathbf{e}_{s_q} \right).$$

The special, single chain, case  $R = 1$  gives

$$\beta_{ij}(K) = \left( \prod_{l=K-j+1}^K \bar{\lambda}_i(l) \right) T_{ij}(K - j + 1), \quad (56)$$

where  $\bar{\lambda}_i(l)$  is the node  $i$  arrival rate when our closed system contains  $l$  customers and  $T_{ij}(K - j + 1)$  is the  $j$ th moment of the node  $i$  flow time when our closed system contains  $K - j + 1$  customers. Relation (56) is analogous to a known result<sup>19</sup> for an  $M/G/1$  queue in isolation which states

$$\beta_j = \lambda^j T_j. \quad (57)$$

Noting that  $\lambda$  is replaced by the product of flow rates, each with a different population, we see that (56) generalizes (57) in the sense of holding for a queue embedded in a closed network with the corresponding, complicated arrival process.

## VI. DELAY AND FLOW TIME MOMENTS—MIXED SYSTEMS

To obtain the mixed network results, we consider the augmented system as in Section IV. We obtain recursions for the delay and flow time moments at a FCFS node with a single state-independent server

---

\* As pointed out in Section VI, the virtual work results also apply to a limited class of processor sharing or LCFS-PR nodes; the limitations being that the service times are exponential with  $\mu_{is} = \mu_{il}$  (i.e., the same assumptions being made for the FCFS nodes). Inclusion of this computation in the closed network algorithm shown in Fig. 4 is straightforward.

embedded in a mixed network. Results are obtained for delays experienced by customers in each of the closed chains and for customers in any\* of the open chains, thus, yielding moments for the virtual delay or unfinished work at a node.

Denoting  $\bar{\mu}_i^{-1}$  as the common mean service time at node  $i$  (i.e.,  $\bar{\mu}_i = \mu_{is} = \mu_{il}$ )<sup>†</sup> and  $\rho_i^o$  as the actual utilization at node  $i$  due to customers belonging to open chains,

$$\rho_i^o = \sum_{l=r+1}^R \frac{\lambda_{il}}{\bar{\mu}_i}, \quad (58)^\ddagger$$

we obtain

$$T_{ijs}(\mathbf{K}) = \sum_{l=1}^r \frac{\bar{\lambda}_{il}(\mathbf{K} - \mathbf{e}_s)}{\bar{\mu}_i(1 - \rho_i^o)} T_{ijl}(\mathbf{K} - \mathbf{e}_s) + \frac{j}{\bar{\mu}_i} T_{i,j-1,s}(\mathbf{K}) + \frac{j\rho_i^o}{\bar{\mu}_i(1 - \rho_i^o)} T_{i,j-1}^o(\mathbf{K} - \mathbf{e}_s); \quad s \leq r, \quad (59)$$

where  $T_{i,j}(\mathbf{K})$  is the  $j$ th moment of the flow time experienced by customers belonging to any open chain at node  $i$  and satisfies

$$T_{ij}^o(\mathbf{K}) = \sum_{l=1}^r \frac{\bar{\lambda}_{il}(\mathbf{K})}{\bar{\mu}_i(1 - \rho_i^o)} T_{ijl}(\mathbf{K}) + \frac{j}{\bar{\mu}_i(1 - \rho_i^o)} T_{i,j-1}^o(\mathbf{K}). \quad (60)$$

The recursion (60) is initialized with

$$T_{ij}^o(\mathbf{0}) = \frac{j}{\bar{\mu}_i(1 - \rho_i^o)} T_{i,j-1}^o(\mathbf{0}) = \frac{j!}{[\bar{\mu}_i(1 - \rho_i^o)]^j}, \quad (61)$$

the open network flow time moments, and (59) is initialized with

$$T_{ijs}(\mathbf{e}_s) = T_{ij}^o(\mathbf{0}). \quad (62)$$

The initialization (62) can be obtained by recognizing that the distribution of the number of customers found at node  $i$  by the single closed chain customer is identical to the distribution of the number at node  $i$  at an arbitrary time point, in equilibrium, in a system with no customers belonging to closed chains, i.e.,  $\mathbf{K} = \mathbf{0}$ . Since the distribution at an arbitrary point in time is identical with that seen by an arbitrary open customer, we have (62).<sup>§</sup>

The results for the moments of the delay distributions can be found using a similar argument. For the closed chains, this results in

\* We note that from Ref. 20 it is easy to show that delays experienced by customers belonging to different open chains have the same moments.

<sup>†</sup> Recall that this condition is required at a FCFS node in order to have a product form solution.

<sup>‡</sup> Since node  $i$  has a state-independent processing rate,  $\mu_i(k) = 1$  and  $\rho_i^o$  is the actual utilization as indicated.

<sup>§</sup> An alternate way of obtaining (62) is by use of (59) at  $\mathbf{K} = \mathbf{e}_s$ , use of (61) and induction.

$$W_{ijs}(\mathbf{K}) = \sum_{l=1}^r \frac{\bar{\lambda}_{il}(\mathbf{K} - \mathbf{e}_s)}{\bar{\mu}_i(1 - \rho_i^o)} W_{ijl}(\mathbf{K} - \mathbf{e}_s) + \frac{j}{\bar{\mu}_i} W_{i,j-1,s}(\mathbf{K}) \\ + \frac{j\rho_i^o}{\bar{\mu}_i(1 - \rho_i^o)} W_{i,j-1}^o(\mathbf{K} - \mathbf{e}_s); \quad s \leq r; \quad j > 1, \quad (63)$$

where  $W_{i,j}^o(\mathbf{K})$ , the  $j$ th moment of the delay experienced by customers belonging to the open chains, is also the  $j$ th moment of the virtual delay or unfinished work at node  $i$  at an arbitrary point in time. The corresponding result for  $j = 1$  is

$$W_{i1s}(\mathbf{K}) = \sum_{l=1}^r \frac{\bar{\lambda}_{il}(\mathbf{K} - \mathbf{e}_s)}{\bar{\mu}_i(1 - \rho_i^o)} \left( \frac{1}{\bar{\mu}_i} + W_{i1l}(\mathbf{K} - \mathbf{e}_s) \right) \\ + \frac{\rho_i^o}{\bar{\mu}_i(1 - \rho_i^o)}; \quad s \leq r; \quad j = 1. \quad (64)$$

The moments of the virtual delay, or delays experienced by open customers, are given by

$$W_{ij}^o(\mathbf{K}) = \sum_{l=1}^r \frac{\bar{\lambda}_{il}(\mathbf{K})}{\bar{\mu}_i(1 - \rho_i^o)} W_{ijl}(\mathbf{K}) \\ + \frac{j}{\bar{\mu}_i(1 - \rho_i^o)} W_{i,j-1}^o(\mathbf{K}); \quad j > 1, \quad (65)$$

and for  $j = 1$

$$W_{i1}^o(\mathbf{K}) = \sum_{l=1}^r \frac{\bar{\lambda}_{il}(\mathbf{K})}{\bar{\mu}_i(1 - \rho_i^o)} \left( \frac{1}{\bar{\mu}_i} + W_{i1l}(\mathbf{K}) \right) + \frac{\rho_i^o}{1 - \rho_i^o} \frac{1}{\bar{\mu}_i}; \quad j = 1. \quad (66)$$

We note that for both the open and closed chains, the  $j$ th moment of the delay and node flow time distributions satisfy the same recursions, (59) and (63), for the closed chains and (60) and (65) for the open chains, the difference lying in the boundary equations (64) and (66). The initial conditions

$$W_{i1l}(\mathbf{0}) = 0; \quad l \leq r,$$

together with (66) and (65), give

$$W_{ij}^o(\mathbf{0}) = \frac{j}{\bar{\mu}_i} \frac{\rho_i^o}{1 - \rho_i^o} W_{i,j-1}^o(\mathbf{0}) = j! \left[ \frac{\rho_i^o}{\bar{\mu}_i(1 - \rho_i^o)} \right]^j, \quad (67)$$

the open network result. Using the same argument as was used for the closed flow time moments, we have

$$W_{ijs}(\mathbf{e}_s) = W_{ij}^o(\mathbf{0}) \quad (68)$$

which can be used to initialize (63) and (64). It is not necessary to have a separate computation for  $W_{ijs}(\mathbf{K})$  if the flow time moments have been computed since they are related by

$$W_{ijs}(\mathbf{K}) = \sum_{l=1}^r \frac{\bar{\lambda}_{il}(\mathbf{K} - \mathbf{e}_s)}{\bar{\mu}_i} T_{ijl}(\mathbf{K} - \mathbf{e}_s) + \rho_i^o T_{ij}^o(\mathbf{K} - \mathbf{e}_s); \quad s \leq r \quad (69)$$

and

$$W_{ij}^o(\mathbf{K}) = \sum_{l=1}^r \frac{\bar{\lambda}_{il}(\mathbf{K})}{\bar{\mu}_i} T_{ijl}(\mathbf{K}) + \rho_i^o T_{ij}^o(\mathbf{K}); \quad s > r \quad (70)^*$$

for the open chains or virtual work.

To obtain the moments of virtual delay for closed systems, denoted by  $V_{ij}(\mathbf{K})$ , we observe that

$$V_{ij}(\mathbf{K}) = \lim_{\rho_i^o \rightarrow 0} W_{ij}^o(\mathbf{K}),$$

which results in the relations given in (66) and (67).

As a final note here we observe that in addition to applying to the FCFS nodes as stated, the virtual work results,  $W_{ij}^o$ ,  $V_{ij}$ , apply to a limited class of processor sharing or LCFS-PR nodes<sup>†</sup> the limitation being that service times are exponential with rates  $\mu_{is} = \mu_{il} = \bar{\mu}_i$ . This follows from the insensitivity of the stationary queue size distribution to the queueing discipline for this case and the memoryless property of the exponential distribution.

## VII. DELAY DISTRIBUTIONS

We consider a multi-job-type closed<sup>‡</sup> network containing either state-independent service centers or infinite server nodes, e.g., central server model of multiprogramming, and present results for the tail of the delay distribution experienced by a type  $s$  job arrival to a FCFS service center embedded in a product-form network. Appendix C, which uses multidimensional generating functions and the known relation between the stationary distributions and those seen by customer arrivals to a node,<sup>21</sup> contains the details of the investigation.

We denote

$$\bar{W}_{is}(t; \mathbf{K}) = P(d_{is} > t; \mathbf{K}) \quad (71)$$

as the probability that a chain  $s$  arrival to node  $i$  is delayed in excess of  $t$  in a system with population  $\mathbf{K}$ . Figure 8 shows a single-chain example where the delay shown corresponds to the time between a

\* The inclusion of the results of this section into the mixed network algorithm shown in Fig. 6 is straightforward.

<sup>†</sup> Or other types not affecting the stationary queue-size distributions, e.g., random selection or LCFS nonpreemptive. The identification as actual delay moments is, however, lost.

<sup>‡</sup> It is a straightforward matter to treat the limiting mixed network, as well as a class of state-dependent nodes (e.g., multiprocessor nodes) by the techniques presented.

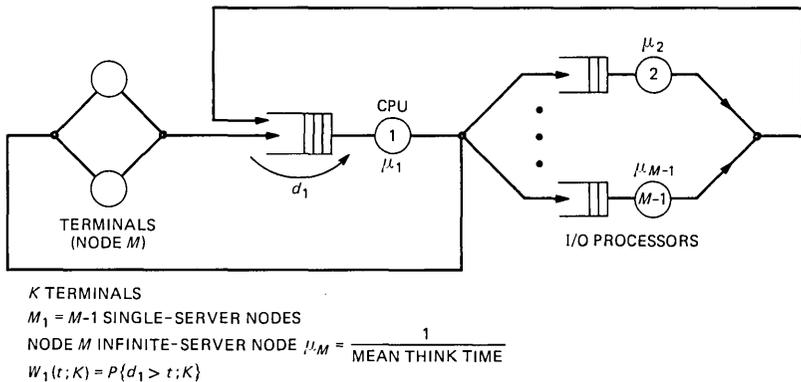


Fig. 8—Closed system illustration—delay distribution.

terminal request and the time the request first gets the attention of the CPU in a system with  $K$  terminals. References 22 and 23 contain a study of the response time distribution (queuing and service) for a single node being fed traffic from a collection of terminals, the classical machine repair problem with multiple repairmen. This could correspond to a multiprocessor version of Fig. 8, but without the I/O processors. In Ref. 24 the asymptotic behavior, as the number of terminals increases, is studied. We note that the methods described in this paper can easily be used to obtain results for the response time distribution (delay plus service time) for the above example.\* In general, the state of the art for obtaining response time distributions for multiple resource systems is quite limited.†

Figure 9 shows a representation of two loosely coupled systems with shared mass storage devices modeled as two central server models with some shared I/O queues.<sup>29</sup> For I/O requests served on a FCFS basis in each I/O queue, the delay distribution of interest shown corresponds to the delay in accessing a disk, either dedicated or shared, for each of the component systems and to the delay in getting each of the CPU's for FCFS scheduling algorithms. We note that the population vector here,  $\mathbf{K} = (K_1, K_2)$ , could correspond to the degree of multiprogramming for each component system.

We start our presentation of results with the single-chain case and then generalize to the multichain case. For the single-chain case, letting

$$\bar{W}_i(t; K) = P(d_i > t; K), \quad (72)$$

\* The single node being a limited queue-dependent server.<sup>10</sup>

† We note that these response times could involve several visits to a given resource, as well as visits to other resources. This is studied in Ref. 25 for the single resource queue with feedback. Some other recent work is discussed in Section VIII.<sup>26,27,28</sup>

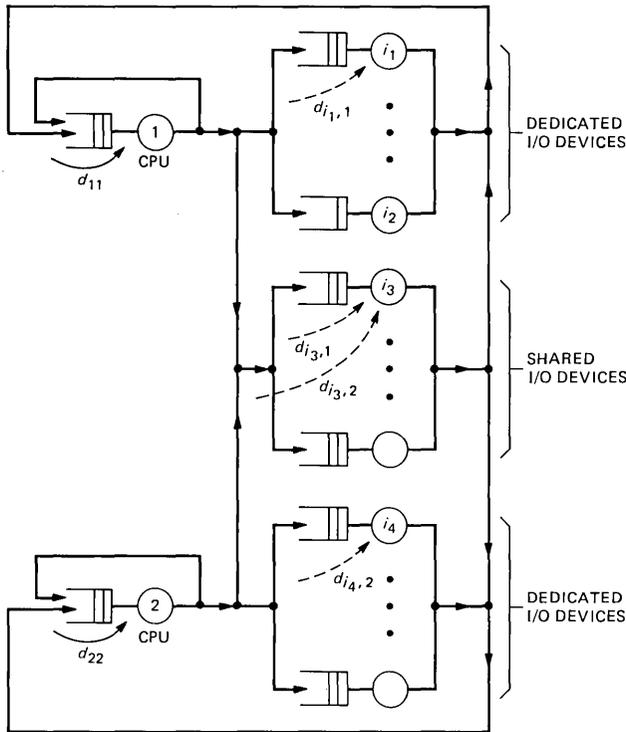


Fig. 9—Two systems sharing mass storage devices.

it is shown in Appendix C that  $\bar{W}_i(t; K)$  satisfies the  $M_1^{\text{th}}$  order recursion

$$\bar{W}_i(t; K) = y_i(t; K - 1) - \sum_{j=1}^{M_1} \bar{\alpha}_j(K - 1) \bar{W}_i(t; K - j), \quad (73)$$

where  $M_1 \leq M$  is the number of single-server nodes in the network. The coefficients  $\bar{\alpha}_j(K - 1)$  are related to the coefficients of  $Z^j$ ,  $\alpha_j$ , in

$$P(Z) = \prod_{i=1}^{M_1} \left( 1 - \frac{\lambda_i}{\mu_i} Z \right) = \sum_{j=0}^{M_1} \alpha_j Z^j, \quad (74)^*$$

where  $\lambda_i$  is the relative arrival rate to node  $i$ .

Clearly,

$$\alpha_j = (-1)^j \sum_{1 \leq i_1 < i_2 < \dots < i_j \leq M_1} \rho_{i_1} \rho_{i_2} \dots \rho_{i_j}, \quad (75)$$

where  $\rho_{i_j}$  is the relative utilization  $\lambda_{i_j}/\mu_{i_j}$ . The coefficients of (73)  $\bar{\alpha}_j(K - 1)$  are obtained from

\* We have arbitrarily labeled nodes 1 through  $M_1$  to correspond to the single-server nodes. Since these nodes are state-independent service centers, in this section we denote the common mean service time at node  $i$  as  $\mu_i^{-1}$ ,  $i = 1, 2, \dots, M_1$ .

$$\bar{\alpha}_j(K-1) = (-1)^j \sum_{1 \leq i_1 < \dots < i_j \leq M_1} \bar{\rho}_{i_1}(K-1) \cdot \bar{\rho}_{i_2}(K-2) \dots \bar{\rho}_{i_j}(K-j), \quad (76)$$

which corresponds to (75) with the actual utilizations

$$\bar{\rho}_{i_j}(K-j) = \frac{\bar{\lambda}_{i_j}(K-j)}{\mu_{i_j}} \quad (77)^*$$

evaluated at the appropriate population replacing the relative utilizations  $\rho_{i_j}$ . Recall that the actual utilizations are available via standard analysis. The forcing function on difference equation (73),  $y_i(t; K-1)$ , can be obtained recursively (see Appendix C) from

$$y_i(t; K) = \frac{\mu_i t}{K-1} \bar{\rho}_i(K) \left( 1 + \frac{\alpha_\infty}{\lambda_i t} \right) y_i(t; K-1) \quad (78)$$

and the initial condition

$$y_i(t; 1) = e^{-\mu_i t} \bar{\rho}_i(1). \quad (79)$$

The quantity  $\alpha_\infty$  represents the relative loading on all the infinite-server nodes

$$\alpha_\infty = \sum_{j=M_1+1}^M \frac{\lambda_j}{\mu_j}. \quad (80)$$

We note that for the network example of Fig. 8,

$$y_1(t; K) = \frac{1}{K-1} \bar{\rho}_1(K) (pr + \mu_1 t) y_1(t; K-1),$$

where  $p$  is the fraction of CPU requests feedback to the terminals,

$$r = \frac{\text{mean think time}}{\text{mean CPU service time}},$$

and  $\mu_i t$  represents the point at which we are evaluating the tail of the delay distribution in units of CPU service times.

For the multi-job-type networks, the tail of the node  $i$  delay distribution as seen by a chain  $s$  arrival is shown in Appendix C to satisfy the multidimensional recursion

$$\bar{W}_{is}(t; \mathbf{K}) = y_i(t; \mathbf{K} - \mathbf{e}_s) - \sum_{(0 < j_1 + j_2 + \dots + j_R \leq M_1)} \bar{\alpha}_{j_1, j_2, \dots, j_R}(\mathbf{K} - \mathbf{e}_s) \cdot \bar{W}_{is}(t; \mathbf{K} - j_1 \mathbf{e}_1 - j_2 \mathbf{e}_2 - \dots - j_R \mathbf{e}_R). \quad (81)$$

The quantities  $\bar{\alpha}_{j_1, \dots, j_R}(\mathbf{K} - \mathbf{e}_s)$  are the multidimensional analogues of  $\bar{\alpha}_j(K-1)$  in (76), i.e., they are related to the coefficients of  $Z_1^i$ .

\* Here  $\bar{\lambda}_i(K)$  denotes the actual arrival rate to node  $i$  for a system with population  $K$ .

$Z_2^{j_2} \dots Z_R^{j_R}, \alpha_{j_1, \dots, j_R}$  in

$$P(\mathbf{Z}) = \prod_{i=1}^{M_1} \left( 1 - \sum_{s=1}^R \frac{\lambda_{is}}{\mu_{is}} Z_s \right). \quad (82)$$

Analogous to (76), we have

$$\begin{aligned} \bar{\alpha}_{j_1 \dots j_R}(\mathbf{K} - \mathbf{e}_s) &= (-1)^{j_1 + \dots + j_R} \sum_{\{i_{pq}\} \in L} \{ [\bar{\rho}_{i_1, 1:1}(\mathbf{K} - \mathbf{e}_s) \dots \bar{\rho}_{i_j, 1:1}(\mathbf{K} - \mathbf{e}_s \\ &\quad - (j_1 - 1)\mathbf{e}_1)] \dots [\bar{\rho}_{i_R, R}(\mathbf{K} - \mathbf{e}_s - j_1\mathbf{e}_1 \\ &\quad - \dots - j_{R-1}\mathbf{e}_{R-1}) \dots \bar{\rho}_{i_R, R}(\mathbf{K} - \mathbf{e}_s - j_1\mathbf{e}_1 - \dots - (j_R - 1)\mathbf{e}_R)] \}, \end{aligned}$$

where

$$0 < \sum_{r=1}^R j_r \leq M_1 \quad (83)$$

and  $\{i_{pq}\} \in L$  corresponds to

$$\left( \begin{array}{c} i_{11} < i_{21} < \dots < i_{j_1 1} \\ \vdots \\ i_{1R} < i_{2R} < \dots < i_{j_R R} \end{array} \right); \quad i_{mn} \neq i_{jk}; \quad (84)$$

i.e.,  $\bar{\alpha}_{j_1, j_2, \dots, j_R}(\mathbf{K} - \mathbf{e}_s)$  is a sum of products of utilizations with  $\bar{\rho}_{i_{p,q}; r}$  evaluated at the population  $(\mathbf{K} - \mathbf{e}_s - j_1\mathbf{e}_1 - j_2\mathbf{e}_2 - \dots - j_{r-1}\mathbf{e}_{r-1} - (p-1)\mathbf{e}_r)$ . The forcing function is obtained recursively from

$$y_i(t; \mathbf{K}) = \begin{cases} \frac{(\bar{a}_{s\infty}(\mathbf{K}) + \bar{\lambda}_{is}(\mathbf{K})t)}{K_s} y_i(t; \mathbf{K} - \mathbf{e}_s); & s \notin \mathcal{R}(i), s \in \mathcal{R}^*(i), K_r = 0 \forall r \notin \mathcal{R}^*(i), \\ \frac{(\bar{a}_{s\infty}(\mathbf{K}) + \bar{\lambda}_{is}(\mathbf{K})t)\delta(\mathbf{K})}{K_s\delta(\mathbf{K} - \mathbf{e}_s)} y_i(t; \mathbf{K} - \mathbf{e}_s); & s \in \mathcal{R}(i), K_r = 0 \forall r \notin \mathcal{R}^*(i), \delta(\mathbf{K} - \mathbf{e}_s) > 0^\dagger \\ 0; & K_r > 0 \text{ for some } r \notin \mathcal{R}^*(i), \end{cases} \quad (85)$$

with initial condition

$$y_i(t; \mathbf{e}_r) = \begin{cases} \bar{\rho}_{ir}(\mathbf{e}_r)e^{-\mu_r t}; & r \in \mathcal{R}(i) \\ 0 & r \notin \mathcal{R}(i). \end{cases} \quad (86)$$

$\mathcal{R}(i)$  denotes the set of chains passing through node  $i$ , and  $\mathcal{R}^*(i)$  denotes the set of chains which either pass through node  $i$  or through an infinite-server node. The quantity

$$\delta(\mathbf{K}) = \sum_{r \in \mathcal{R}^*(i)} \frac{K_r}{\left( 1 + \frac{a_{r\infty}}{\lambda_{ir}t} \right)}, \quad (87)$$

<sup>†</sup> If  $\delta(\mathbf{K} - \mathbf{e}_s) = 0$  and  $\mathbf{K} - \mathbf{e}_s \neq \mathbf{0}$  [if  $\mathbf{K} - \mathbf{e}_s = \mathbf{0}$ , use initial condition (86)]  $y_i(t; \mathbf{K})$  can be computed from the first relation for some  $s \notin \mathcal{R}(i)$ ,  $s \in \mathcal{R}^*(i)$  corresponding to  $K_s > 0$ .

the relative chain  $r$  loading on the infinite-server nodes

$$a_{r\infty} = \sum_{j=M_1+1}^M \frac{\lambda_{jr}}{\mu_{jr}}, \quad (88)$$

and the actual chain  $r$  infinite-server loading

$$\bar{a}_{r\infty}(\mathbf{K}) = \sum_{j=M_1+1}^M \frac{\bar{\lambda}_{jr}(\mathbf{K})}{\mu_{jr}} = \bar{\lambda}_r(K) a_{r\infty}. \quad (89)$$

We note that it is only necessary to compute  $y_i(t, \mathbf{K})$  in the subspace spanned by chains passing through node  $i$  or any infinite-server node. For the special case where  $M_1 = M$  (e.g., the central server model)  $\mathcal{R}(i) = \mathcal{R}^*(i)$  and (85) becomes

$$y_i(t, \mathbf{K}) = \begin{cases} \frac{\mu_i t}{K_s |\mathbf{K}| - 1} \bar{\rho}_{is}(\mathbf{K}) y_i(t, \mathbf{K} - \mathbf{e}_s); \\ s \in \mathcal{R}(i), K_r = 0 \forall r \notin \mathcal{R}(i), |\mathbf{K}| > 1, \\ 0; \quad K_r > 0 \text{ for some } r \notin \mathcal{R}(i), \end{cases} \quad (90)$$

where

$$|\mathbf{K}| = \sum_{r=1}^R K_r. \quad (91)$$

### VIII. SUMMARY

This paper has presented contributions to the foundations of a tool to support performance analysis and modeling activities aimed at answering some key questions at various stages of a computer system's life cycle. The emphasis here has been on presenting easily and efficiently computable results for calculating distributional information and a stable, efficient method for dealing with congestion adaptive devices.<sup>†</sup> Mixed systems have been considered to allow us the generality of dealing with traffic sources which are fundamentally different in their behavior. By obtaining results for different levels of customer aggregation, we allow one to consider a macroscopic or more microscopic level of detail. The virtual delay results allow us to quantify differences between service as perceived by an arriving customer and that perceived by a measuring device.

We note that many open questions exist in the areas of obtaining results related to the distribution of total time a customer spends in a subnetwork consisting of several nodes with feedback (e.g., Fig. 1,

---

<sup>†</sup> In a recent paper,<sup>30</sup> a modification of mean value analysis is introduced to eliminate numerical instabilities when dealing with general state-dependent service rates. The method involves analysis of complementary systems and evaluation of marginal queue size distributions.

where the time of interest is the response time).<sup>\*</sup> This is an area of active research in the literature (e.g., see Refs. 26 and 28); however, the results available are fairly restrictive with respect to the network topology<sup>†</sup> or customer paths, and do not apply to, for example, the network of Fig. 1. Approximating the moments of the distribution of the overall time to transit a network from the individual node flow time moments is one possible approach which would have to be studied and evaluated. The problem arises because of statistical dependence of a given customer's flow times as he sojourns the network. For open Jackson networks, Reiman<sup>33</sup> uses a heavy traffic limit theorem to obtain a diffusion approximation for the network sojourn times. Another area of importance relates to the inclusion of priorities in, for example, the CPU schedule. We note that an approximation technique (based on utilization adjustments) does exist<sup>9</sup> for handling a class of priority disciplines and can perform quite satisfactorily in many cases.<sup>‡</sup> The approximation is such that it enables us to compute performance measures using results in this paper.

## APPENDIX A

### *Recursions for Queue Size Factorial Moments*

For closed systems, it is known<sup>1</sup> that the marginal, stationary probability distributions satisfy

$$p(k_i = k; \mathbf{K}) = \frac{1}{\mu_i(k)} \sum_{s=1}^R \frac{\bar{\lambda}_{is}(\mathbf{K})}{\mu_{is}} \cdot p(k_i = k - 1; \mathbf{K} - \mathbf{e}_s); \quad k > 0. \quad (92)$$

While conceptually, the desired moments could be computed from recursively computed marginal probability distributions, it is not recommended. Other, more computable, approaches could involve the use of generating functions.<sup>35</sup> Our approach is to directly obtain a recursive relation for the moments. We use (92), (8), and (9) to obtain (10), with the indicated initial conditions in a straightforward manner. We obtain the required node-chain throughputs,  $\bar{\lambda}_{is}(\mathbf{K})$ , by considering the lower level aggregation

<sup>\*</sup> Or, for that matter, at a processor sharing node embedded in a general closed network. The waiting time distribution for a specific closed network consisting of a single processor sharing node fed by a single finite population class is treated in Ref. 31.

<sup>†</sup> In Ref. 27 a computational methodology is given for obtaining upper and lower bounds where an arbitrary network topology is allowed. When applied to two  $M/M/1$  queues in tandem (for which the exact solution is known) the upper and lower bounds are close; however, over 30,000 states were used in the computation at 80 percent occupancy. Computational aspects are presented in Ref. 32.

<sup>‡</sup> Reference 34, which obtains a closed form solution for a two-node closed network with priorities, proposes a criterion under which the approximation technique would be expected to perform well.

$$\mathbf{k}_i = (k_{i1}, k_{i2}, \dots, k_{iR}),$$

where  $k_{ir}$  represents the number of chain  $r$  customers at node  $i$ . Denoting  $k_i$  as the total number of customers at node  $i$ , we can write the recursive relation

$$p_i(\mathbf{k}_i; \mathbf{K}) = \frac{k_i}{k_{is}} \frac{\bar{\lambda}_{is}(\mathbf{K})}{\mu_i(k_i)\mu_{is}} p_i(\mathbf{k}_i - \mathbf{e}_s; \mathbf{K} - \mathbf{e}_s); \quad k_{is} > 0, \quad (93)$$

which is obtainable from the product form solution. From (93), (8), the standard Little's law argument at each node and about the entire system, and the irreducibility of the routing chains we get (12) through (15). To obtain the desired recursion for the joint factorial moments, we use (8) and (93) in (18), make the appropriate identification corresponding to a system with reduced population and obtain (21).

The mixed system result (26) is obtained directly from (10) by considering the augmented system with population

$$\mathbf{K}' = (\mathbf{K} | K_{r+1}, \dots, K_R),$$

denoting

$$\beta_{ij}(\mathbf{K}) = \lim_{K_{r+1}, \dots, K_R \rightarrow \infty} \beta_{ij}(\mathbf{K}')$$

and decomposing the sum over the open and closed chains. The lower level aggregation results are obtained by use of

$$\lim_{K_{r+1}, \dots, K_R \rightarrow \infty} p_i(\mathbf{k}_i; \mathbf{K}') = \begin{cases} \frac{k_i}{k_{is}} \frac{\lambda_{is}}{\mu_i(k_i)\mu_{is}} p_i(\mathbf{k}_i - \mathbf{e}'_s; \mathbf{K}); & s > r \\ \frac{k_i}{k_{is}} \frac{\bar{\lambda}_{is}(\mathbf{K})}{\mu_i(k_i)\mu_{is}} p_i(\mathbf{k}_i - \mathbf{e}_s; \mathbf{K} - \mathbf{e}_s); & s \leq r, \end{cases} \quad (94)$$

where  $\mathbf{e}'_s$  is an  $R$ -dimensional unit vector in direction  $s$  as opposed to  $\mathbf{e}_s$ , the corresponding  $r$  dimension vector for  $s \leq r$ .

## APPENDIX B

### Recursions for Node Delay and Flow Time Moments

Denoting  $\bar{W}_{is}(t, \mathbf{K})$  as the complementary delay distribution experienced by a chain  $s$  customer at node  $i$  when the system population is  $\mathbf{K}$ , we can write

$$\bar{W}_{is}(t, \mathbf{K}) = \sum_{k=1}^{|\mathbf{K}|-1} \frac{(\mu_{is}t)^{k-1}}{(k-1)!} e^{-\mu_{is}t} p(k_i(t_{is}^-) \geq k; \mathbf{K}), \quad (95)$$

where  $\mu_{is} = \mu_i$  at the FCFS state-independent node under consideration,  $k_i(t_{is}^-)$  represents the number of customers at node  $i$  seen by a chain  $s$

arrival to node  $i$ , and  $|\mathbf{K}| = K_1 + K_2 + \dots + K_R$ . For the class of closed\* systems, we are initially considering

$$p(k_i(t_{is}^-) = k; \mathbf{K}) = p(k_i = k; \mathbf{K} - \mathbf{e}_s), \quad (96)$$

i.e., the distribution as seen by an arriving chain  $s$  customer is equal to the distribution at an arbitrary point in time in equilibrium (i.e., the stationary distribution) for a system with one less chain  $s$  customer.<sup>20</sup> Using (96) in (95), we can obtain the Laplace-Stieltjes transform of the flow time distribution for node-chain pair  $(i, s)$

$$T_{is}(\eta, \mathbf{K}) = \frac{\mu_i}{\eta + \mu_i} - \frac{\eta}{\mu_i} \sum_{k=1}^{|\mathbf{K}|-1} p(k_i \geq k; \mathbf{K} - \mathbf{e}_s) \left( \frac{\mu_i}{\eta + \mu_i} \right)^{k+1} \quad (97)$$

The  $j$ th moment, obtained by differentiation, satisfies

$$T_{ijs}(\mathbf{K}) = \frac{j}{\mu_i} T_{i,j-1,s}(\mathbf{K}) + \frac{j}{\mu_i^2} \sum_{k=1}^{|\mathbf{K}|-1} k(k+1) \dots (k+j-2) \cdot p(k_i \geq k; \mathbf{K} - \mathbf{e}_s), \quad (98)$$

and the summation in (98) can be written as

$$\sum_{q=0}^{|\mathbf{K}|-2} (q+1) \dots (q+j-1) \sum_{l=1}^R \frac{\bar{\lambda}_{il}(\mathbf{K} - \mathbf{e}_s)}{\mu_i} \cdot p(k_i \geq q; \mathbf{K} - \mathbf{e}_l - \mathbf{e}_s), \quad (99)$$

where we have used (92) to get the one-step recursion on the tail of the marginal queue size distribution. Inserting (99) in (98) and making the appropriate identification, we get (46), with initial and boundary conditions as indicated.

To obtain the Little's Law type moment relation, we use the definition of  $\beta_{ij}(\mathbf{K})$  and summation by parts to get

$$\beta_{ij}(\mathbf{K}) = j \sum_{l=0}^{|\mathbf{K}|-j} (l+1)(l+2) \dots (l+j-1) p(k_i \geq l+j; \mathbf{K}), \quad (100)$$

which upon repeated application of (92) results in

$$\beta_{ij}(\mathbf{K}) = \sum_{s_1=1}^R \dots \sum_{s_j=1}^R \bar{\lambda}_{is_1}(\mathbf{K}) \bar{\lambda}_{is_2}(\mathbf{K} - \mathbf{e}_{s_1}) \dots \bar{\lambda}_{is_j} \left( \mathbf{K} - \sum_{q=1}^{j-1} \mathbf{e}_{s_q} \right) \cdot \frac{j}{\mu_i} \sum_{l=0}^{|\mathbf{K}|-j} (l+1) \dots (l+j-1) p \left( k_i \geq l; \mathbf{K} - \sum_{q=1}^j \mathbf{e}_{s_q} \right). \quad (101)$$

Identifying the last term, we obtain

---

\* The mixed system results obtained by the usual limiting system argument are reported in Section VI.

$$\beta_{ij}(\mathbf{K}) = \sum_{s_1=1}^R \cdots \sum_{s_j=1}^R \lambda_{is_1}(\mathbf{K}) \lambda_{is_2}(\mathbf{K} - \mathbf{e}_{s_1}) \cdots \lambda_{is_j} \left( \mathbf{K} - \sum_{q=1}^{j-1} \mathbf{e}_{s_q} \right) \times T_{ijs_j} \left( \mathbf{K} - \sum_{q=1}^{j-1} \mathbf{e}_{s_q} \right), \quad (102)$$

which for the special, single chain, case  $R = 1$  gives the moment relation

$$\beta_{ij}(K) = \left( \prod_{l=K-j+1}^K \lambda_i(l) \right) T_{ij}(K - j + 1). \quad (103)$$

## APPENDIX C

### Recursions for the Tail of the Node Delay Distributions

We consider a closed system of  $M_1$  state-independent, single-server nodes and  $M - M_1$  infinite-server nodes and obtain an  $M_1^{\text{th}}$  order difference equation for the tail of the customer delay distribution at a FCFS node. Denoting

$$\bar{W}_{is}(t; \mathbf{K}) = P(d_{is} > t; \mathbf{K})$$

as the probability that a chain  $s$  arrival to node  $i$  is delayed in excess of  $t$  in a system with population  $\mathbf{K}$ , we can write

$$\bar{W}_{is}(t; \mathbf{K}) = \sum_{k=1}^{|\mathbf{K}|-1} \frac{(\mu_i t)^{k-1}}{(k-1)!} e^{-\mu_i t} p(k_i \geq k; \mathbf{K} - \mathbf{e}_s). \quad (104)$$

We denote the product-form normalization constant\* as

$$G(\mathbf{K}) = \sum_{\substack{\mathbf{k} \\ \sum \mathbf{k}_j = \mathbf{K}}} g_1(\mathbf{k}_1) g_2(\mathbf{k}_2) \cdots g_M(\mathbf{k}_M),$$

where

$$g_j(\mathbf{k}_j) = \frac{|\mathbf{k}_j|!}{k_{j1}! \cdots k_{jR}!} \left( \frac{\lambda_{j1}}{\mu_{j1}} \right)^{k_{j1}} \cdots \left( \frac{\lambda_{jR}}{\mu_{jR}} \right)^{k_{jR}}; \\ j = 1, 2, \dots, M_1, \\ \left( \frac{\lambda_{j1}}{\mu_{j1}} \right)^{k_{j1}} \cdots \left( \frac{\lambda_{jR}}{\mu_{jR}} \right)^{k_{jR}} \\ g_j(\mathbf{k}_j) = \frac{\quad}{k_{j1}! \cdots k_{jR}!}; \quad j = M_1 + 1, \dots, M, \quad (105)$$

and we have labeled nodes  $j = 1, \dots, M_1$  as the single-server nodes. Multiplying (104) by  $G(\mathbf{K} - \mathbf{e}_s)$  and obtaining the generating function, we have

---

\* Our final result will not involve computation of the normalization constant, the calculation of which can result in numerical problems.

$$L_{is}(\mathbf{Z}) = \frac{S_i(\mathbf{Z}) \left( \sum_{r=1}^R \frac{\lambda_{ir}}{\mu_i} Z_r \right)}{\left( 1 - \sum_{r=1}^R \frac{\lambda_{ir}}{\mu_i} Z_r \right)} \exp \left[ -\mu_i t \left( 1 - \sum_{r=1}^R \frac{\lambda_{ir}}{\mu_i} Z_r \right) \right], \quad (106)$$

where

$$L_{is}(\mathbf{Z}) = \sum_{K_1=0}^{\infty} \cdots \sum_{K_R=0}^{\infty} \bar{W}_{is}(t; \mathbf{K} + \mathbf{e}_s) G(\mathbf{K}) Z_1^{K_1} \cdots Z_R^{K_R}, \quad (107)$$

and

$$S_i(\mathbf{Z}) = \sum_{K_1=0}^{\infty} \cdots \sum_{K_R=0}^{\infty} \mathcal{G}_i(\mathbf{K}) Z_1^{K_1} \cdots Z_R^{K_R}, \quad (108)$$

where  $\mathcal{G}_i(\mathbf{K})$  is the normalization constant for a reduced network (node  $i$  absent)

$$\mathcal{G}_i(\mathbf{K}) = \sum_{\substack{\mathbf{k}_j = \mathbf{K} \\ j \neq i}} g_1(\mathbf{k}_1) \cdots g_{i-1}(\mathbf{k}_{i-1}) g_{i+1}(\mathbf{k}_{i+1}) \cdots g_R(\mathbf{k}_R). \quad (109)$$

Inserting

$$S_i(\mathbf{Z}) = \prod_{j \neq i} \bar{S}_j(\mathbf{Z}), \quad (110)$$

where

$$\bar{S}_j(\mathbf{Z}) = \begin{cases} \frac{1}{\left( 1 - \sum_{r=1}^R \frac{\lambda_{jr}}{\mu_j} Z_r \right)}; & j \leq M_1, \\ e^{\sum_{r=1}^R \frac{\lambda_{jr}}{\mu_j} Z_r}; & j > M_1. \end{cases} \quad (111)$$

into (106), we get

$$P(\mathbf{Z}) L_{is}(\mathbf{Z}) = e^{-\mu_i t} \left( \sum_{r=1}^R \frac{\lambda_{ir}}{\mu_i} Z_r \right) e^{\sum_{r=1}^R (a_{r\infty} + \lambda_{ir} t) Z_r} \triangleq H_i(t, \mathbf{Z}), \quad (112)$$

where  $P(\mathbf{Z})$  is the polynomial given by (82) and  $a_{r\infty}$  given by (88). Inversion of (112) and division by the appropriate normalization constant results in

$$\sum_{\substack{j_1=0 \\ \dots \\ j_1 + \dots + j_R \leq M_1}}^{M_1} \cdots \sum_{j_R=0}^{M_1} \alpha_{j_1, \dots, j_R} \frac{l_{is}(\mathbf{K} - \mathbf{e}_s - j_1 \mathbf{e}_1 - \dots - j_R \mathbf{e}_R)}{G(\mathbf{K} - \mathbf{e}_s)} \\ = \frac{h_i(t, \mathbf{K} - \mathbf{e}_s)}{G(\mathbf{K} - \mathbf{e}_s)} \triangleq y_i(t, \mathbf{K} - \mathbf{e}_s), \quad (113)$$

where  $\alpha_{j_1, \dots, j_R}$  is the coefficient of  $Z_1^{j_1} Z_2^{j_2} \dots Z_R^{j_R}$  in the polynomial (82),  $h_i(t; \mathbf{K})$  is the inverse generating function of  $H_i(t; \mathbf{Z})$  given by

$$h_i(t, \mathbf{K}) = e^{-\mu_i t} \left[ \prod_{r \in \mathcal{R}^*(i)} \frac{(a_{r\infty} + \lambda_{ir} t)^{K_r}}{K_r!} \right] \sum_{r \in \mathcal{R}(i)} \frac{\lambda_{ir} K_r}{\mu_i (a_{r\infty} + \lambda_{ir} t)}; \quad (114)$$

$\mathcal{R}(i)$  denotes the set of chains passing through node  $i$  and  $\mathcal{R}^*(i)$  denotes the set of chains which either pass through node  $i$  or an infinite server node. The quantity  $l_{is}(\mathbf{K})$  in (113) satisfies

$$l_{is}(\mathbf{K}) = \bar{W}_{is}(t, \mathbf{K} + \mathbf{e}_s) G(\mathbf{K}). \quad (115)$$

As it stands (113) represents an  $M_1^{\text{th}}$  order recursion for the distribution tail; however, the coefficient and forcing function involve the normalization constant. Upon use of (115) in (113), and recognizing that

$$\frac{G(\mathbf{K} - \mathbf{e}_s - j_1 \mathbf{e}_1 - \dots - j_R \mathbf{e}_R)}{G(\mathbf{K} - \mathbf{e}_s)}$$

can be written as a product of node throughput proportionality constants

$$\left[ \prod_{l_1=0}^{j_1-1} \bar{\lambda}_1(\mathbf{K} - \mathbf{e}_s - l_1 \mathbf{e}_1) \right] \cdot \left[ \prod_{l_2=0}^{j_2-1} \bar{\lambda}_2(\mathbf{K} - \mathbf{e}_s - j_1 \mathbf{e}_1 - l_2 \mathbf{e}_2) \right] \dots \left[ \prod_{l_R=0}^{j_R-1} \bar{\lambda}_R(\mathbf{K} - \mathbf{e}_s - j_1 \mathbf{e}_1 - \dots - j_{R-1} \mathbf{e}_{R-1} - l_R \mathbf{e}_R) \right],$$

we obtain (81) and (83).

The recursions (85) for the forcing function

$$y_i(t; \mathbf{K}) = \frac{h_i(t, \mathbf{K})}{G(\mathbf{K})} \quad (116)$$

follow from

$$h_i(t, \mathbf{K}) = \frac{(a_{s\infty} + \lambda_{is} t)}{K_s} \frac{\delta(\mathbf{K})}{\delta(\mathbf{K} - \mathbf{e}_s)} h_i(t, \mathbf{K} - \mathbf{e}_s); \quad s \in \mathcal{R}(i), \quad (117)$$

where  $\delta(\mathbf{K})$  is given by (87), and from

$$h_i(t, \mathbf{K}) = \frac{(a_{s\infty} + \lambda_{is} t)}{K_s} h_i(t, \mathbf{K} - \mathbf{e}_s); \quad s \notin \mathcal{R}(i), s \in \mathcal{R}^*(i). \quad (118)$$

## REFERENCES

1. M. Reiser and S. S. Lavenberg, "Mean Value Analysis of Closed Multichain Queueing Networks," *J. ACM*, 27, No. 2 (April 1980), pp. 313-22.
2. A. A. Fredericks, "Analysis of the Impact of Multiprogramming on a Transaction Oriented System," *IFIP Proc. Performance 1981*, Amsterdam: North Holland, November 1981, pp. 489-502.
3. R. J. T. Morris and J. S. Kaufman, unpublished work.
4. P. J. Denning, "A Note on Paging Drum Efficiency," *Computing Surveys*, 4, No. 1 (March 1972), pp. 1-3.

5. E. G. Coffman, Jr. and P. J. Denning, *Operating Systems Theory*, Englewood Cliffs, N.J.: Prentice Hall, 1973.
6. 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.
7. J. P. Buzen, "A Queueing Network Model of MVS," *ACM Computing Surveys*, 10, No. 3 (September 1978), pp. 319-31.
8. H. Kobayashi, "System Design and Performance Analysis Using Analytic Models," *Current Trends in Programming Methodology, Volume III, Software Modeling*, eds. K. M. Chandy and R. T. Yeh, Englewood Cliffs, N.J.: Prentice Hall, 1978, pp. 72-114.
9. K. C. Sevcik, "Priority Scheduling Disciplines in Queueing Network Models of Computer Systems," *Proceedings of IFIP Congress 77*, Amsterdam, The Netherlands: North Holland, 1977, pp. 565-70.
10. M. Reiser and C. H. Sauer, "Queueing Network Models: Methods of Solution and Their Program Implementation," *Current Trends in Programming Methodology, Volume III, Software Modeling*, eds. K. M. Chandy, R. T. Yeh, Englewood Cliffs, N.J.: Prentice-Hall, 1978, pp. 115-67.
11. K. M. Chandy and C. H. Sauer, "Computational Algorithms for Product Form Queueing Networks," *Proc. Performance 80—The 7th IFIP W. G. 7.3 International Symposium on Computer Performance Modeling, Measurement and Evaluation*, Toronto, May 1980, Extra Papers, pp. 1-10.
12. J. P. Buzen, "Analysis of System Bottlenecks Using a Queueing Network Model," *Proc. ACM-SIGOPS Workshop on System Performance Evaluation*, New York: ACM April 1971, pp. 82-103.
13. D. Ferrari, *Computer System Performance Evaluation*, Englewood Cliffs, N.J.: Prentice-Hall, 1978.
14. F. P. Kelly, "Reversibility and Stochastic Networks," Chichester: John Wiley, 1979.
15. M. Reiser and A. G. Konheim, "Finite Capacity Queueing Systems with Applications in Computer Modeling," *Siam J. Computing*, 7, No. 2 (May 1978), pp. 210-29.
16. G. Latouche, "Algorithmic Analysis of a Multiprogramming Multiprocessor Computer System," *J. ACM* 28, No. 4 (October 1981), pp. 662-79.
17. B. Avi-Itzhak and D. P. Heyman, "Approximate Queueing Models for Multiprogramming Computer Systems," *Operations Research*, 21, No. 6 (1973), pp. 1212-30.
18. C. H. Sauer and K. M. Chandy, *Computer Systems Performance Modeling*, Englewood Cliffs, N.J.: Prentice Hall, 1981.
19. D. Gross and C. M. Harris, *Fundamentals of Queueing Theory*, New York: John Wiley, 1974.
20. K. C. Sevcik and I. Mitrani, "The Distribution of Queueing Network States at Input and Output States," 4th Int. Symp. on Modeling and Performance Analysis of Computer Systems, Vienna (February 1979), pp. 319-35.
21. S. S. Lavenberg and M. Reiser, "Stationary State Probabilities at Arrival Instants for Closed Queueing Networks with Multiple Types of Customers," *J. Appl. Prob.*, 17, No. 4 (December 1980), pp. 1048-61.
22. A. Sekino, "Performance Evaluation of Multiprogrammed Time-Shared Computer System," Project MAC, M.I.T., MAC-TR-103, September 1972.
23. A. Sekino, "Response Time Distribution of Multiprogrammed Time-Shared Computer Systems," *Proc. Sixth Annual Princeton Conf. on Information Sciences and Systems*, pp. 613-9, 1972.
24. J. W. Wong, "Response Time Distribution of the  $M/M/m/N$  Queueing Model," *Operations Research*, 27, No. 6 (November-December 1979), pp. 1196-202.
25. S. S. Lam and A. U. Shankar, "Response Time Distributions for a Multi-class Queue with Feedback," *Proc. of Performance 80—The 7th IFIP W. G. 7.3 Int. Symp. Computer Performance Modeling Measurement and Evaluation*, Toronto (May 1980), pp. 225-34.
26. B. Melamed, unpublished work.
27. B. Melamed and M. Yadin, "Randomization Procedures in the Computation of Cumulative Time Distribution Over Discrete State Markov Processes," Dept. of Industrial Engineering, Northwestern University, Department of Industrial Engineering, Technion, Israel Institute of Technology, June, 1980.
28. J. Walrand and P. Varaiya, "Sojourn Times and the Overtaking Condition in Jacksonian Networks," *Adv. Appl. Prob.*, 12 (December 1980), pp. 1000-18.
29. J. P. Buzen, "Modeling Computer System Performance," *Proc. 7th Int. Conf. of the Computer Measurement Group, Atlanta* (November 16-19), 1976, pp. 230-8.
30. M. Reiser, "Queue-Dependent Servers in Closed Queueing Networks," *Performance Evaluation*, 1 (January 1981) pp. 7-18.

31. D. Mitra and R. Pinsky, "Waiting Time Distributions from Closed Queueing Network Models of Interactive, Shared-Processor Systems," IFIP Proc. Performance 81, Amsterdam: North Holland, November 1981, pp. 113-31.
32. B. Melamed and M. Yadin, "Numerical Computation of Sojourn Time Distributions in Queueing Networks," Dept. of Ind. Eng. and Math. Sci., Northwestern Univ., Dept. of Ind. Eng., Technion, Israel Institute of Technology, 1981.
33. M. I. Reiman, "The Diffusion Approximation for Sojourn Times in Jackson Networks," Proc. Appl. Prob.—Computer Sci.—The Interface ORSA/TIMS Conf., Boca Raton, January 1981, ed. R. L. Disney and T. J. Ott, Boston: Birkhauser, 1982.
34. R. J. T. Morris, "Priority Queueing Networks," B.S.T.J., 60, No. 8 (October 1981), pp. 1745-69.
35. M. Reiser and H. Kobayashi, "Queueing Networks with Multiple Closed Chains: Theory and Computational Algorithms," I.B.M. J. Res. Dev. 19, No. 3 (May 1975), pp. 283-94.



## Adaptive Intra-Interframe DPCM Coder

By P. PIRSCH

(Manuscript received August 4, 1981)

*Adaptive prediction schemes provide lower transmission rates than those obtained by simple previous frame prediction. In this paper, we measure the entropy of prediction errors for two types of adaptive intra-interframe prediction algorithms. In the first case, that predictor which results in the least prediction error for previously transmitted neighboring pels is selected from a set of predictor functions. In the second case, prediction is a weighted sum of previous frame and intraframe predictions, where the weights are changed from pel to pel by gradient techniques. We also investigate various modifications of the basic methods. Further, a new type of variable length encoding in which the locations of the nonzero prediction errors are coded by horizontal run lengths is discussed. Compared with the pel entropy of previous frame prediction, the run length coding gives a gain of 2 to 16 percent, depending on the scene. Compared to simple previous frame prediction the first type of adaptive scheme in combination with horizontal run length coding provides a gain in entropy of 18 to 29 percent, whereas the second type of adaptive scheme provides a gain of 20 to 32 percent.*

### I. INTRODUCTION

The bit rate required for digital transmission of television pictures can be significantly reduced by interframe DPCM encoding. The coding method which has been widely proposed for video-telephone and video-conference application is conditional replenishment.<sup>1,2</sup> In conditional replenishment, each frame of a television sequence is segmented into changed and unchanged areas. Various methods can be used for encoding the changed parts of a frame. Intraframe predictive coding is very efficient for these parts.<sup>3,4</sup> In conditional replenishment, no information about the unchanged areas is transmitted. At the receiver, the unchanged areas are reconstructed by repeating from the previous frame. However, it is necessary to transmit address information that

indicates the location of the changed areas. Several modifications and improvements of the basic method of conditional replenishment have been made. Most by them are described in a survey by Haskell.<sup>5</sup>

This paper describes adaptive intra-interframe prediction. It is obvious that stationary background of a frame is best predicted from a pel in the previous frame which has the same position as the pel to be predicted, whereas parts of a frame with moving objects are better predicted by an intraframe predictor. Therefore, a prediction scheme which provides automatic switching between the two types of predictors, depending upon the part of the picture, will result in lower bit rates. To avoid the transmission of additional predictor control information, the adaptive prediction schemes described here are based on previously transmitted reconstructed pels. Further, no forward segmenter like that of conditional replenishment is used. Therefore, only the quantized prediction error has to be coded and transmitted.

A block diagram of such a DPCM encoder with adaptive prediction is shown in Fig. 1. The investigations in this paper concern a comparison of the performance of two types of adaptive predictors. The first one, denoted by predictor selection, is a scheme where one predictor is selected from a set of predictors. In the second scheme, the predictor is a weighted sum of predictors and the prediction coefficients are changed continuously by a gradient algorithm. As a measure of predictor performance, the entropy of the quantized prediction error is used. For three different television scenes an estimate of the entropy is obtained from DPCM simulations. The necessary measures against buffer overflow and underflow, in case of variable length encoding, have not been considered here.

This paper is organized as follows. Section II gives a detailed description of the two basic algorithms and their modifications. Section III describes a variable length encoding scheme which is especially suited for DPCM coders that have improved prediction. Results of simulations on real scenes are given in Section IV.

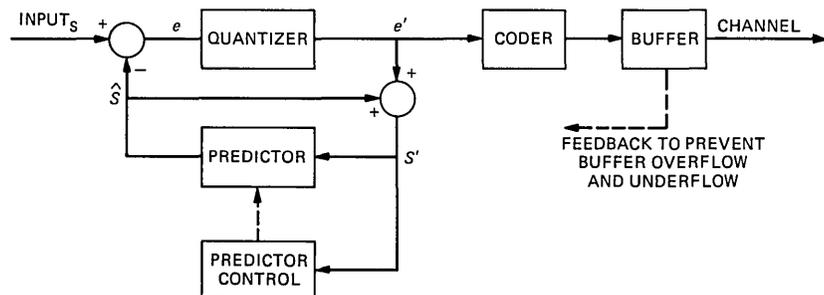


Fig. 1—Block diagram of a DPCM coder with adaptive predictor.

## II. DESCRIPTION OF THE PREDICTION ALGORITHMS

Let  $f_i$  be one of  $M$  predictor functions. If each  $f_i$  is a linear predictor function, then

$$f_i = \sum_{j=1}^N a_{ij}s'_j, \quad (1)$$

where  $a_{ij}$  are the weighting coefficients and  $s'_j$  are previously transmitted pels. The prime in  $s'_j$  indicates that these are reconstructed pels which are known at the receiver. The subscripting for pels neighboring the present pel  $s_0$  is shown in Fig. 2. The predictor functions  $f_i$ ,  $i = 1, 2, \dots, M$ , are linear combinations of  $N$  pels,  $s'_j$ ,  $j = 1, 2, \dots, N$ , which form a vector

$$\mathbf{s}' = \begin{pmatrix} s'_1 \\ s'_2 \\ \vdots \\ s'_N \end{pmatrix}. \quad (2)$$

In vector notation, equation (1) can be written as

$$f_i = \mathbf{a}_i^T \mathbf{s}'. \quad (3)$$

Here the superscript  $T$  denotes the transpose of a vector or matrix, and  $\mathbf{a}_i$  is the vector formed by the coefficients  $a_{ij}$ ,  $j = 1, 2, \dots, N$ . The prediction value  $\hat{s}_0$  is a weighted sum of all predictor functions,

$$\hat{s}_0 = \sum_{i=1}^M b_i f_i. \quad (4)$$

If  $\mathbf{f}$  denotes the vector of elements  $f_i$ ,  $i = 1, 2, \dots, M$ , and  $\mathbf{b}$  denotes the vector of elements  $b_i$ ,  $i = 1, 2, \dots, M$ , then

$$\hat{s}_0 = \mathbf{b}^T \mathbf{f}. \quad (5)$$

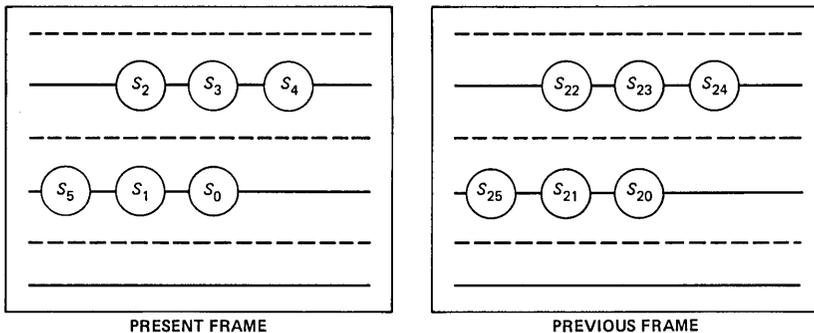


Fig. 2—Configuration and subscripting of picture elements. Pel  $s_0$  is the present pel to be predicted. Dotted lines denote scan lines from previous fields.

This description is general and includes switched prediction by allowing special values of  $\mathbf{b}$ , such as  $b_k = 1$  and  $b_i = 0$  for all  $i \neq k$ . Combining (3) and (5) it follows that

$$\mathbf{f} = \mathbf{A}\mathbf{s}' \quad (6)$$

$$\hat{s}_0 = \mathbf{b}^T \mathbf{A}\mathbf{s}', \quad (7)$$

where

$$\mathbf{A} = \begin{bmatrix} \mathbf{a}_1^T \\ \mathbf{a}_2^T \\ \vdots \\ \mathbf{a}_M^T \end{bmatrix} \quad (8)$$

is an  $M \times N$  matrix. The set of predictor functions is described by the matrix  $\mathbf{A}$ , with the coefficient vectors  $\mathbf{a}_i$  chosen such that a particular predictor function provides a good prediction for a specific area of a television scene, like stationary background, moving objects, etc. The algorithm then seeks to automatically adapt the vector  $\mathbf{b}$  to various areas of a scene so as to minimize the prediction error.

In this investigation, the set of predictor functions is restricted to a previous frame predictor

$$f_1 = s'_{20} \quad (9)$$

and an intraframe predictor

$$f_2 = a_{21}s'_1 + a_{22}s'_2 + a_{23}s'_3. \quad (10)$$

The following prediction algorithms are described for two predictor functions, but they can easily extend to more than two.

### 2.1 Predictor selection schemes

From a set of predictor functions, the predictor which results in the least prediction error for previously transmitted neighboring pels is selected as the predictor for the present pel. For each predictor function, a decision function  $u_i$  is defined, which is the sum of the amount of the prediction errors for each pel in a small window of neighboring pels. The predictor which has the smallest value for the decision function is chosen as predictor. This criterion was also used by Stuller et al.<sup>6</sup> for gain and displacement compensation. The basic selection rule for two predictor functions is as follows:

$$\hat{s} = \begin{cases} f_1 & \text{if } u_1 \leq u_2 \\ f_2 & \text{if } u_1 > u_2, \end{cases} \quad (11)$$

where

$$u_i = \sum_{k \in W} |s'_k - f_i(s'_k)|. \quad (12)$$

The subscript  $k$  denotes a pel in a neighborhood  $W$ . The window  $W$  is chosen such that  $s'_k$  is known at the receiver. The decision function  $u_i$  can be evaluated at the receiver without transmission of additional information about predictor selection. For previously transmitted nearest neighbors, the index set  $W$  is

$$W_\alpha = \{1, 2, 3, 4\}. \quad (13)$$

For real-time implementation the choice of  $W_\alpha$  creates problems, because of the use of the pel  $s'_1$ . The time constraint for calculation of  $u_i$  can be reduced by using the index set

$$W_\beta = \{2, 3, 4, 5\} \quad (14)$$

or

$$W_\gamma = \{2, 3, 4\} \quad (15)$$

instead of  $W_\alpha$ . The window  $W_\gamma$  is also used by Stuller et al.<sup>6</sup>

A further simplification for hardware implementation can be obtained by introducing a quantizer function  $Q_s[\cdot]$  in (12). Then the decision functions  $u_i$  are given by

$$u_i = \sum_{k \in W} Q_s [|s'_k - f_i(s'_k)|]. \quad (16)$$

A modification which leads to a simpler implementation than the basic selection rule (11) can be described as follows. Choose the predictor function  $f_i$  which has within a window  $W$  most frequent minimum magnitude of the difference

$$d_{ik} = s'_k - f_i(s'_k). \quad (17)$$

In the case of two predictor functions at each position  $k$ , a binary variable  $v_k$  which describes which predictor function is better, is defined as follows,

$$v_k = \begin{cases} 1 & |d_{1k}| \leq |d_{2k}| \\ 0 & |d_{1k}| > |d_{2k}|. \end{cases} \quad (18)$$

The decision functions  $u_i$  are now given by

$$\begin{aligned} u_1 &= \sum_{k \in W} v_k \\ u_2 &= \sum_{k \in W} \bar{v}_k, \end{aligned} \quad (19)$$

where  $\bar{v}_k$  is the complement of  $v_k$ . The predictor with smallest value  $u_i$  is chosen. The selection rules as discussed above require that one predictor function be chosen even if both decision functions  $u_i$  are identical. An improvement can be obtained by using a "soft-predictor

switch," i.e., the prediction value is a weighted sum of predictor functions, as given by (4), with the weights  $b_i$  being proportional to the frequency of preference of the predictor function  $f_i$ . Hence, for two predictor functions,

$$\begin{aligned} b_1 &= \frac{1}{n} \sum_{k \in W} v_k \\ b_2 &= \frac{1}{n} \sum_{k \in W} \bar{v}_k, \end{aligned} \quad (20)$$

where  $n$  is the number of pels in the window  $W$ . To avoid the division by 3, for  $W = W_\alpha$  the contribution of the pel at position 3 to (20) is doubled, and  $n$  is chosen to be 4 for this special case.

## 2.2 Adaptive prediction based on a steepest descent method

The steepest descent<sup>7</sup> is a mathematical method which has been often used for optimization. One advantage of this method is its simplicity. This method has been used frequently for adaptive systems. It is also proposed by Netravali and Robbins<sup>8</sup> and Stuller et al.<sup>6</sup> for motion-compensated prediction. Here it will be applied to adaptive intra-interframe prediction.

Let us assume that the prediction value is a weighted sum of predictor functions as given by (5). Then the prediction error is given by

$$e = s - \mathbf{b}^T \mathbf{f}. \quad (21)$$

In the following, the present pel is denoted by  $s$ , rather than  $s_0$ . The variance of the prediction error  $e$  is a quadratic function in  $\mathbf{b}$ .

$$F(\mathbf{b}) = E[(s - \mathbf{b}^T \mathbf{f})^2], \quad (22)$$

where  $E[\cdot]$  is the expected value. The gradient with respect to  $\mathbf{b}$  is given by

$$\begin{aligned} \mathbf{g} &= \nabla_{\mathbf{b}} F(\mathbf{b}) = -2E[(s - \mathbf{b}^T \mathbf{f})\mathbf{f}] \\ &= -2E[e\mathbf{f}]. \end{aligned} \quad (23)$$

The steepest descent is an iterative method, where starting from an initial guess the vector  $\mathbf{b}$  is modified recursively according to,

$$\mathbf{b}^{(m+1)} = \mathbf{b}^{(m)} - \gamma^{(m)} \mathbf{g}^{(m)}. \quad (24)$$

The adjustment of the vector  $\mathbf{b}^{(m)}$  is made in the direction of the negative gradient. The scalar  $\gamma^{(m)}$  has to be optimized by a one-dimensional search scheme at each step  $m$ . However, real-time appli-

cations are performed with a constant value of  $\gamma$ . The best value of  $\gamma$  depends on the type of data. In addition, the value of  $\gamma$  influences the stability and the speed of convergence of  $\mathbf{b}$ .

From eqs. (23) and (24), it follows that an adaptive prediction scheme which, based on a gradient method, is given by

$$\mathbf{b}^{(m+1)} = \mathbf{b}^{(m)} + 2\gamma E_w[\mathbf{e}\mathbf{f}]^{(m)}, \quad (25)$$

where  $E_w[\cdot]$  is the expected value within a small window of neighboring pels as given by (13), (14), or (15). The coefficient vector  $\mathbf{b}$  is updated on a pel-by-pel basis along the scanning direction, i.e., if  $\mathbf{b}^{(m+1)}$  is the coefficient vector at the present pel,  $\mathbf{b}^{(m)}$  is the coefficient vector at the previous pel. At the beginning of each line, an initial estimate of  $\mathbf{b}$  is used, e.g., the mean of  $\mathbf{b}$  at the previous line. Simulations indicate that because of a fast adjustment an initial vector  $\mathbf{b}$  with elements  $b_i = 1/M$ ,  $i = 1, 2, \dots, M$  is appropriate.

In this study, several modifications of the recursion given by (25) have been investigated. The various algorithms will be compared with respect to prediction gain and cost of implementation. A high prediction gain requires an appropriate value of  $\gamma$  in (25). Simulations with several values of  $\gamma$  indicate that for video signals with normalized range [0, 1] the optimum value of  $\gamma$  is about one. In such a case the adjustment from pel to pel is relatively small, and the transition from one predictor function to another takes several pels. By introducing an additional constraint

$$\sum_{j=1}^M b_j = 1, \quad (26)$$

the value of optimum  $\gamma$  is increased to about 64. The increased value of  $\gamma$  provides a shorter transition from one predictor function to another and the constraint (26) improves the stability of the algorithm.

With the constraint of (26), the steepest descent method has to be modified to minimize the augmented function of (22)

$$\Phi(\mathbf{b}, \lambda) = E[(s - \mathbf{b}^T \mathbf{f})^2] + \lambda(\mathbf{b}^T \mathbf{o} - 1), \quad (27)$$

where  $\mathbf{o}$  is a vector with all elements equal to 1. The coefficient vector  $\mathbf{b}$  is updated recursively by

$$\mathbf{b}^{(m+1)} = \mathbf{b}^{(m)} - \gamma(-2E[\mathbf{e}\mathbf{f}]^{(m)} + \lambda^{(m)} \mathbf{o}). \quad (28)$$

Using (26) to eliminate  $\lambda^{(m)}$  from (28), and replacing  $E[\cdot]$  by  $E_w[\cdot]$ , then

$$\mathbf{b}^{(m+1)} = \mathbf{b}^{(m)} + 2\gamma \mathbf{C} E_w[\mathbf{e}\mathbf{f}], \quad (29)$$

where  $\mathbf{C}$  is an  $M \times M$  matrix given by

$$\mathbf{C} = \mathbf{U} - \frac{1}{M} \mathbf{oo}^T = \begin{pmatrix} 1 - \frac{1}{M} & -\frac{1}{M} & \cdots & -\frac{1}{M} \\ -\frac{1}{M} & 1 - \frac{1}{M} & \cdots & -\frac{1}{M} \\ \vdots & \vdots & \ddots & \vdots \\ -\frac{1}{M} & -\frac{1}{M} & \cdots & 1 - \frac{1}{M} \end{pmatrix} \quad (30)$$

and  $\mathbf{U}$  is the unit matrix. Because of (26), it follows that

$$s - \mathbf{b}^T \mathbf{f} = \mathbf{b}^T (\mathbf{so} - \mathbf{f}) = \mathbf{b}^T \mathbf{d}, \quad (31)$$

where  $\mathbf{d}$  is a vector of differences similar to (17). This leads to an equivalent recursion of (29), given by

$$\mathbf{b}^{(m+1)} = \mathbf{b}^{(m)} - 2\gamma \mathbf{C} E_w[\mathbf{e} \mathbf{d}]. \quad (32)$$

In the recursions given above, the coefficient vector  $\mathbf{b}$  at the previous pel is updated by an adjustment to obtain the coefficient vector at the present pel. However, a picture is two-dimensional in nature, the values of  $\mathbf{b}$  for pels from the previous line in the immediate neighborhood of the present pel are quite close to that of the present pel. This idea results in a modification of (25) which is given below.

$$\mathbf{b}^{(m+1)} = E_w[\mathbf{b}]^{(m)} + 2\gamma E_w[\mathbf{e} \mathbf{f}]^{(m)}. \quad (33)$$

Let us assume that the samples  $s$  and the predictor functions  $f_i$  are represented by 8 bits. In such cases, in the recursions given above at each position within the window, a product of two 8-bit numbers has to be calculated. A reduction in the cost of implementation can be achieved by using the three-level quantizer, shown in Fig. 3, for the prediction error  $e$  and the differences  $d$ . These investigations show that a three-level quantizer with a dead zone is more advantageous than the signum function used by Netravali and Robbins.<sup>8</sup>

The algorithm (29) and (33) for the case of two predictor functions, in combination with a three-level quantizer  $Q_D$ , results in the following recursive scheme,

$$\begin{aligned} b_1^{(m+1)} &= E_w[b_1]^{(m)} + \gamma E_w[Q_D(e) Q_D(f_1 - f_2)]^{(m)} \\ b_2^{(m+1)} &= E_w[b_2]^{(m)} - \gamma E_w[Q_D(e) Q_D(f_1 - f_2)]^{(m)}, \end{aligned} \quad (34)$$

with the constraints

$$\begin{aligned} b_1 + b_2 &= 1 \\ 0 &\leq b_1 \\ 0 &\leq b_2. \end{aligned} \quad (35)$$

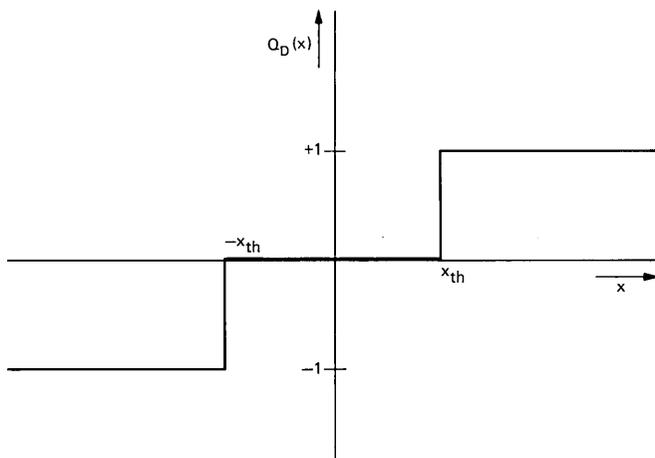


Fig. 3—Three-level quantizer for gradient quantization.

The latter two constraints of (35) were introduced to avoid negative weighting coefficients.

### III. VARIABLE LENGTH ENCODING BY HORIZONTAL RUN LENGTH

An adaptive prediction scheme leads to many predictable pels. A pel is described as predictable if its quantized prediction error is represented by the level zero. To obtain a low transmission rate, the quantized prediction error is coded by a variable length code. There is always a loss in mean transmission rate compared to the entropy if not all of the negative logarithm of the probability of the prediction error representative levels are integer. This loss is especially high if one level has a probability much larger than 0.5. For adaptive prediction schemes, this is true for the quantizer level zero. To overcome this problem, block coding is frequently used. For the application described, a special coding scheme is proposed.

From each frame, a two-level picture is generated which indicates where the pels with zero code words (zCW) and where the pels with nonzero code words (NZCW) are located. This new picture can be coded by known one-dimensional and two-dimensional coding techniques for two-level pictures. The NZCWS are coded in parallel by a variable-length code like a Huffman code and multiplexed with the code words of the two-level picture such that the receiver can decide between the two types of data. A block diagram of such a coder is shown in Fig. 4.

For a horizontal run length code, the set of symbols to be coded is listed in Fig. 5. For each of the sets, i.e., zero runs (ZR), nonzero runs (NZR) and nonzero code words (NZCW), a variable length code can be determined independently and matched to the probability of the

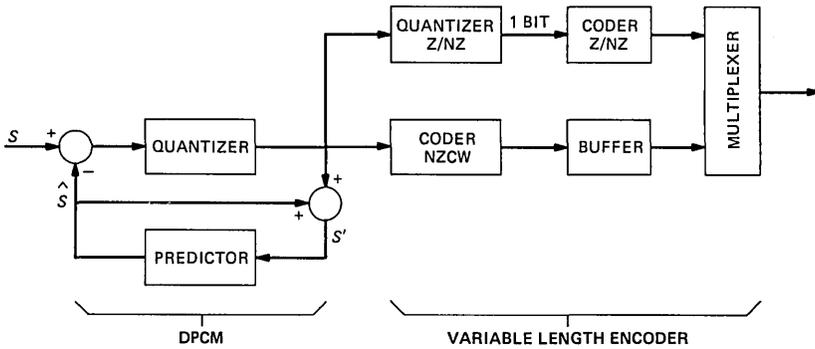


Fig. 4—Block diagram of a new type of variable length encoding.

REGULAR SET OF CODE WORDS

$\{0, 1, 2, \dots, k\}$

NEW SETS FOR CODING

(i) Set of nonzero code words (NZCW)

$\{1, 2, \dots, k\}$

(ii) Set of zero runs (ZR)

$i$	ZR
0	1
1	01
2	001
3	0001
⋮	⋮
$n$	0000...01
$n+1$	0000...00

(ii) Set of nonzero runs (NZR)

$i$	NZR
0	0
1	10
2	110
3	1110
⋮	⋮
$m$	1111...10
$m+1$	1111...11

Fig. 5—Set of symbols for horizontal run length coding.

symbols of that particular set (e.g., Huffman code). The type of runs are chosen so as to allow a wrap-around coding from line to line. Wrap-around coding means that a run is not terminated at the end of a line but continued in the next line. Furthermore, the longest run to be



consists of 60 frames obtained by sampling a video signal of 1-MHz bandwidth, at the Nyquist rate. Each sample was quantized to 8 bits. One frame of each sequence is shown in Fig. 7.

One scene, called Judy, is a head-and-shoulders view of a person engaged in active conversation. The second scene, John and Mike, shows two people entering the camera field of view and walking briskly around each other. The third sequence, Mike and Nadine, is a panned view of two people always in view of the camera.

Even though the quantizer characteristic of a DPCM coder should be designed according to the prediction scheme, for simplification in these investigations, the same 35-level quantizer shown below was used for all simulations. The quantizer has the following positive representative levels: 0, 5, 12, 19, 28, 37, 46, 57, 68, 79, 90, 103, 116, 129, 142, 155, 168, 181. This quantizer was chosen since it gave good picture quality, although the quantization error was visible in specific picture areas under short viewing distance. The decision levels are always in the middle between two succeeding levels. The performance of the predic-



Fig. 7a One frame out of each sequence—Scene Judy.



Fig. 7b One frame out of each sequence—Scene John and Mike.



Fig. 7c One frame out of each sequence—Scene Mike and Nadine.

tion schemes was evaluated by computing the pel entropy, the entropy of a horizontal run length code, and the variance of the quantized prediction error.

For comparison of adaptive and nonadaptive schemes, results for four nonadaptive predictors were obtained. The nonadaptive prediction schemes which were used are given below.

$$\hat{s} = s'_{20} \quad (38)$$

$$\hat{s} = s'_1 - s'_{21} + s'_{20} \quad (39)$$

$$\hat{s} = \frac{3}{4} s'_1 - \frac{2}{4} s'_2 + \frac{3}{4} s'_3 + \frac{3}{4} s'_{20} - \frac{2}{4} s'_{21} + \frac{1}{4} s'_{22} - \frac{2}{4} s'_{23} \quad (40)$$

$$\hat{s} = \frac{7}{8} s'_1 - \frac{5}{8} s'_2 + \frac{6}{8} s'_3. \quad (41)$$

The first predictor (38) is simple previous frame prediction. The prediction scheme given by (39) is frequently proposed for interframe coding.<sup>5,9</sup> The predictor (40) is a three-dimensional predictor proposed by Klie<sup>10</sup> for moving areas of a picture. Equation (41) describes an intraframe predictor which minimizes the variance of the prediction error.<sup>11</sup>

The results of the nonadaptive predictors are shown in the upper part of Tables Ia, b, and c. These investigations show that previous frame prediction (38) is advantageous for sequences with not much motion (Judy), while the intraframe predictor (41) and the predictor (40) are better for sequences with rapidly moving objects (Mike and Nadine). An additional decrease in entropy can be obtained by using the horizontal run length coding scheme. This gain is especially high (16 percent) for the sequence Judy where ZR and NZR are better grouped.

Table Ia—Entropy per pel and variance of the prediction error for nonadaptive and adaptive predictors—Scene Judy.

Entropy in Bit Per Pel		Variance $E[e^2]$	Prediction Scheme
$H_{PEL}$	$H_{RUN}$		
1.035	0.875	16.6	Previous frame, eq. (38)
1.120	0.953	8.5	2-D Interframe, eq. (39)
1.349	1.297	9.1	3-D Interframe, eq. (40)
1.840	1.760	31.7	2-D Intraframe, eq. (41)
0.838	0.765	5.3	Predictor selection, eq. (11), (12), $W_\alpha$
0.781	0.718	4.8	Predictor selection with soft switch eq. (18), (20), $W_\alpha$
0.783	0.730	4.9	Gradient algorithm, eq. (34), $W_\alpha$

Table Ib—Entropy per pel and variance of the prediction error for nonadaptive and adaptive prediction—Scene John and Mike.

Entropy in Bit Per Pel		Variance	Prediction Scheme
$H_{PEL}$	$H_{RUN}$	$E[e^2]$	
2.393	2.190	142.1	Previous frame, eq. (38)
2.400	2.286	114.1	2-D Interframe, eq. (39)
2.154	2.094	61.5	3-D Interframe, eq. (40)
2.397	2.323	88.9	2-D Intraframe, eq. (41)
1.795	1.711	39.6	Predictor selection, eq. (11), (12), $W_\alpha$
1.774	1.687	36.7	Predictor selection with soft switch eq. (18), (20), $W_\alpha$
1.724	1.629	34.2	Gradient algorithm, eq. (34), $W_\alpha$

Table Ic—Entropy per pel and variance of the prediction error for nonadaptive and adaptive predictors—Scene Mike and Nadine.

Entropy in Bit Per Pel		Variance	Prediction Scheme
$H_{PEL}$	$H_{RUN}$	$E[e^2]$	
2.859	2.809	194.9	Previous frame, eq. (38)
3.008	2.982	250.0	2-D Interframe, eq. (39)
2.537	2.504	108.0	3-D Interframe, eq. (40)
2.546	2.506	117.1	2-D Intraframe, eq. (41)
2.385	2.353	87.4	Predictor selection, eq. (11), (12), $W_\alpha$
2.370	2.336	80.8	Predictor selection with soft switch eq. (18), (20), $W_\alpha$
2.325	2.284	77.2	Gradient algorithm, eq. (34), $W_\alpha$

Adaptive prediction schemes as given in Section II were simulated with (38) and (41) as predictor functions. The average bit rate per pel for three schemes are shown in the lower part of Tables Ia, b, and c. The adaptive schemes give an additional decrease in entropy if the horizontal run length coding technique is used; this improvement depends upon the type of picture.

Compared to the case of simple previous frame prediction, the predictor selection in combination with horizontal run length coding results in reductions of 18 to 29 percent. The corresponding reductions for the more sophisticated gradient method are 20 to 32 percent. The minimum and maximum entropy of a single frame within a sequence are reduced by about the same amount as the average entropy of the sequence. This can be recognized for the gradient method in Fig. 8, which shows the entropy per pel of each frame versus frame number.

In Section II, several modifications of the basic methods, to obtain a simpler hardware implementation, were presented. Most of these modifications have only a small influence on the entropy. The basic predictor selection scheme requires the summations of 8-bit numbers for determination of the decision functions (12). A coarse four-level quantizer

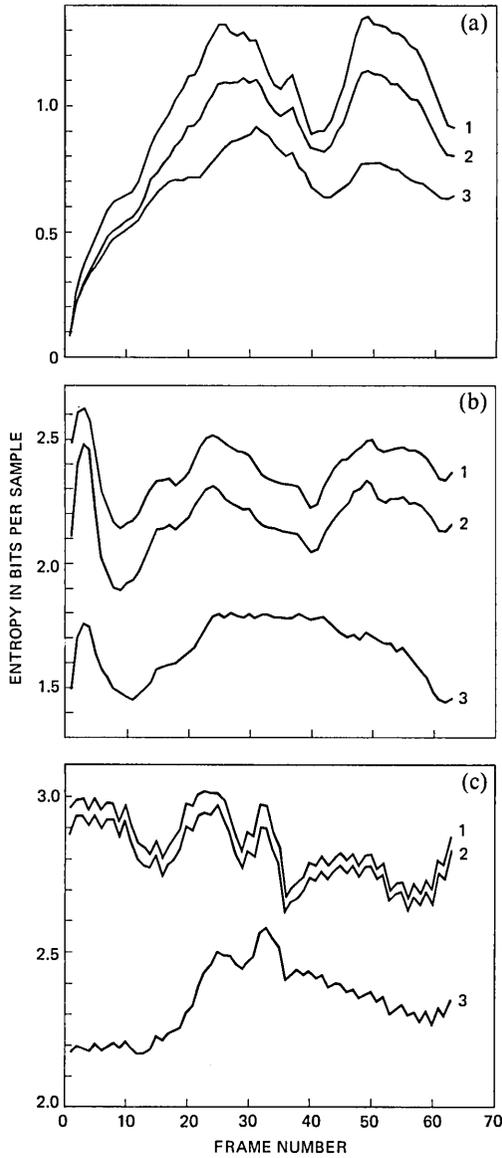


Fig. 8—Plots of entropy per pel versus frame number for each sequence. Configuration one shows the pel entropy  $H_{pel}$  of previous frame prediction; two shows the horizontal run length entropy  $H_{RUN}$  of previous frame prediction; and three shows the horizontal run length entropy  $H_{RUN}$  of the gradient algorithm (33) with the constraint (26). (a) Scene Judy. (b) Scene John and Mike. (c) Scene Mike and Nadine.

$$Q_s(x) = \begin{cases} 0 & 0 \leq |x| < 6 \\ 1 & 6 \leq |x| < 18 \\ 2 & 18 \leq |x| < 36 \\ 4 & 36 \leq |x| \end{cases} \quad (42)$$

for determination of the decision function (16) increases the entropy by about 1 percent.

The use of a binary variable  $v_k$ , equation (18), which indicates which predictor function is advantageous at the position  $k$ , in combination with the soft-switch algorithm of equation (20) is to be preferred. Compared to the predictor selection scheme (11), (12), this algorithm provides a reduction of up to 7 percent in entropy. In addition, it is easier to implement.

For the gradient method the algorithm (34) which incorporates several modifications of the original method is useful concerning cost of implementation and the reduction in entropy. The constraint (26) is especially advantageous. For the algorithm (34), a three-level quantizer with thresholds at  $\pm 4$  was used. The optimum value of  $\gamma$  was found to be  $1/4$ . Each line started with initial values  $b_1 = 1/2$  and  $b_2 = 1/2$  for  $b$ . As long as the weighting coefficients  $b_j$  are represented by more than 4 bits, the gradient method provides a small gain in entropy compared to the predictor selection schemes.

In these investigations, three windows  $W_\alpha$ ,  $W_\beta$ , and  $W_\gamma$  were used. The window  $W_\beta$  provides results very close to that of  $W_\alpha$ , whereas  $W_\gamma$  provides an increase of about 2 percent in entropy.

Further, it was found that using three predictor functions (the intraframe predictor is now split into two functions, one for horizontal prediction and one for vertical prediction) is not better. Besides the intraframe predictor function (39), the predictor function

$$f_2 = \frac{1}{2} s'_1 + \frac{1}{2} s'_3 \quad (43)$$

was also used. This resulted in an increase of 4 to 5 percent in the entropy.

It is of interest to know how these adaptive schemes perform in comparison with conditional replenishment and displacement compensation schemes. The results published in Ref. 6 (Table I, page 1235) based on the same source data are of some interest in this context. Hence, a comparison is possible, but it should be noted that the 35-level quantizer used in this investigation is a modification of the one used in Refs. 6 and 8. Further in this investigation, an additional thresholding of prediction error is not performed.

Compared to conditional replenishment the adaptive schemes provide a reduction in entropy of 19 to 38 percent, depending upon the

scene. For active scenes like John and Mike and Mike and Nadine, the adaptive schemes provide a data rate close to that of displacement compensation (within  $\pm 5$  percent range). The run length coding scheme provides an additional reduction in entropy for sequences with low activity. For the sequence Judy, this reduction is 26 percent compared to conditional replenishment in case of previous frame prediction in combination with run length coding.

## V. CONCLUSION

The performance of two types of adaptive intra-interframe predictors in combination with horizontal run length coding was studied. The gain in entropy of the predictor selection scheme is nearly as high as that of an adaptive scheme which is based on a gradient technique. Various modifications of the two basic methods which were investigated provided only small changes in entropy. Therefore, the adaptive algorithm which has the lowest cost of implementation should be chosen.

Further investigations are necessary for the quantizer design and the buffer control in a fixed rate system. A combination of the described adaptive intra-interframe algorithms with motion compensation will result in a more sophisticated system which provides further entropy reduction.

## REFERENCES

1. F. W. Mounts, "A Video Encoding System Using Conditional Picture-Element Replenishment," B.S.T.J., 48, No. 7 (September 1969), pp. 2545-54.
2. B. G. Haskell, F. W. Mounts, and J. C. Candy, "Interframe Coding of Videotelephone Pictures," Proc. IEEE, 60, No. 7 (July 1972), pp. 792-800.
3. J. O. Limb, R. F. W. Pease, and K. A. Walsh, "Combining Intraframe and Frame-to-Frame Coding for Television," B.S.T.J., 53, No. 6 (August 1974), pp. 1137-73.
4. H. Wendt, "Interframe-Codierung für Videosignale," Internationale Elektr. Rundschau (1973), pp. 2-7.
5. B. G. Haskell, "Frame Replenishment Coding of Television," in *Image Transmission Techniques*, W. K. Pratt, ed., New York: Academic Press, 1979.
6. J. A. Stuller, A. N. Netravali, and J. D. Robbins, "Interframe Television Coding Using Gain and Displacement Compensation," B.S.T.J., 59, No. 7 (September 1980), pp. 1227-40.
7. P. R. Adby and M. A. H. Dempster, *Introduction to Optimization Methods*, London: Chapman and Hall, 1974.
8. A. N. Netravali and J. D. Robbins, "Motion-Compensated Television Coding: Part I," B.S.T.J., 58, No. 3 (March 1979), pp. 631-70.
9. H. Yasuda et al., "Transmitting 4-MHz TV Signals by Combinational Difference Coding," IEEE Trans. Commun., COM-25, No. 5 (May 1977), pp. 508-16.
10. J. Klie, "Codierung von Fernsehsignalen für niedrige Übertragungsbitraten," Ph.D. Dissertation, University of Hannover, West Germany, 1978.
11. P. Pirsch and L. Stenger, "Statistical Analysis and Coding of Color Video Signals," Acta Electronica, 19, No. 4 (1976), pp. 277-87.

## Human Performance Engineering Considerations for Very Large Computer- Based Systems: The End User

By I. S. YAVELBERG

(Manuscript received May 15, 1981)

*Effective Human Performance Engineering for a large-scale, computer-based system involves many complex strategic and tactical decisions regarding the computer system design, the target user's behavior, and the organization/environment. Descriptions of the more important performance considerations are presented. These are based primarily on the experience accrued during the last several years in the building of Bell Laboratories centrally developed computer systems for use by telephone company loop operations personnel in assisting them to do their job. The target population can be broken down into these distinct classes: End Users, Database Maintainers, and the Data System Support Staff. This paper focuses on the End User, and specifically the Bell System Service Representative. Important points include: (i) early emphasis of human performance considerations in the computer system design process can reap valuable benefits; (ii) care must be taken to specify input/output design features which have gone through the human/system engineering step of identifying a favorable payoff versus penalty ratio; and (iii) based on measurement data and user interrogation, computer system availability and transaction failures and response times can seriously damage user performance and system acceptance.*

### I. INTRODUCTION

Effective Human Performance Engineering (HPE) for a large-scale computer-based system involves many complex strategic and tactical decisions with regard to the computer system design, the target user's behavior, and the organization/environment. Descriptions of the more important performance considerations are presented. These are based primarily on the experience accrued during the last several years in

the building of Bell Laboratories centrally developed computer systems for use by telephone company loop operations personnel in assisting them to perform their job. Wherever possible, observations are compared to recent literature associated with other systems or experiments.

The major in-place elements that affect human performance are the organization/environment and the behavioral considerations of the workers themselves. The introduction of a computerized support system adds another element. Presumably, the computer improves the effectiveness of the worker in performing the job. It also provides some design flexibility to counter some of the negative aspects of the other three existing elements. However, the computer system itself introduces serious behavioral effects on its users that may negate its benefits.

The target population can be broken down into these distinct classes: end users, database maintainers, and data system support staff. This paper focuses on the end user with special emphasis on the Bell System Service Representative. Principal points made are as follows:

- An understanding of the target population's attitude and behavior toward computers is necessary.
- Early emphasis of human performance considerations in the computer system design process (workflows) can reap valuable benefits. Unworkable or unreasonable performance requirements on the user can be substantially avoided.
- Care must be taken to specify input/output (I/O) design features which have gone through the human/system engineering step of identifying a favorable payoff versus penalty ratio.
- The introduction of a computer to the existing organization/environment should stimulate some adjustments (job assignments) to optimize the mechanized features. Every attempt should be made to do this as early as possible. However, if workers in one department are asked to increase (or even change) their workload to make it easy to program a computer helping another department, there is extensive resistance.
- Operational—more so than functional—characteristics of the computer system (availability, transaction response time, transaction failures) can have a serious, and often underestimated, negative effect on user performance. For example, many sporadic short outages can have a significantly greater effect on user performance than a single, extended outage.

## II. MAJOR CATEGORIES OF HUMAN PERFORMANCE CONSIDERATIONS

The model that represents the relevant physical and behavioral elements and interrelationships has, as the central element, the collec-

tion of people whose performance is what this paper is all about: the target population or the end user of the computer system. Surrounding the user are three major categories, each of which introduces human performance considerations. The computer system is related to the user by a two-way interface which represents physical, as well as behavioral, interactions. The two in-place categories represent organizational/environmental considerations and the behavioral considerations of the user. These two categories have unidirectional interfaces representing behavioral influences only.

In contrast to other models in the field, I chose to group organization and environment together and add the design and constraints of the mechanized elements (main frame, communications, terminals) under the umbrella category, computer system. I do not advocate waiting for the performance deficiencies to occur before addressing solutions. We can predict potential performance problems and avoid them. This important philosophy is emphasized throughout this paper. Most of the material concerns the human/computer interface: in a positive sense, where interface features relieve user performance problems associated with the precomputer job and, in a negative sense, where operational characteristics of the interface itself introduce new performance difficulties.

### III. TARGET POPULATION

Without question, knowing the target population is fundamental in solving any human performance problems. In particular, it cannot be overstressed how necessary an understanding of the target population's behavior towards computers is with respect to effective performance.

#### *3.1 Population classes*

For many large computer-based systems, the target population really changes with delivery. In the predelivery (usually manual) environment there are generally two classes of users: customer service end users and back-room record maintainers. With the introduction of the computer system, three classes, generally distinguished by their level of expertise with computers,<sup>1</sup> emerge: the unskilled, the semiskilled, and the skilled. In a Bell operating company (BOC) environment, the jobs associated with these levels are often end users and their supervision at the unskilled level; database maintenance clerks and their supervision at the semiskilled level; and data systems support staff at the skilled level. It is important to remember that skill here applies specifically to computer use and not to the whole job. I prefer to identify these classes as end users, maintenance clerks, and operating staff.

Within this target population framework, human/computer inter-

face design should be multifaceted, and groomed specifically for each class.

### **3.2 End user: the Service Representative**

This paper deals with the end user. Most of the field experience, obtained between 1978 and the present, is associated with the Bell System Service Representative (Service Rep). The computer system for which the considerations are based is Premises Information System (PREMIS), which is a Bell Laboratories system designed to provide BOCs with customer-related information to help determine service order information.

PREMIS supports the Residence Service Center Rep on-line during new customer negotiation and also provides some important data that must go on the service order. When a customer calls the BOC to place an order for new residential service, there are a number of tasks that must be performed by the BOC Service Rep taking the order:

- (i) Get the new address correctly.
- (ii) Sell telephone service at the new address.
- (iii) Establish the credit class for the customer to determine if a deposit is required.
- (iv) Quote charges and rates.
- (v) Give the date service will begin.
- (vi) Assign a telephone number.

The Service Rep records all of this information on a service order, which will be used by other departments in the BOC to provide service. PREMIS aids in all of these tasks and replaces the paper records, microfiche, telephone calls, or guesswork used previously. The physical architecture consists of a very large UNIVAC computer, a communications network built around BANCS, another Bell Laboratories product, and 40/4 terminals located at each Service Rep's position. The computer is administered by a data systems group, and the database is maintained by several dispersed maintenance groups. Primarily, PREMIS is an address-keyable system providing the needed address-related data. The Service Rep simply keys the new address into a preformatted mask on the terminal. Usually all that is needed is the house number and street name.

PREMIS responds with information about the geographic area that is needed on the service order. This includes wire center, exchange, rate zone, tax area, directory group, and the service features available for the address. This same display will also include the existing customer's name, telephone number, and presence of an in-place connected circuit or loop from the address back to the central office.

As of December 1980, PREMIS served almost two million residences, accounting for almost half of South Central Bell Telephone Company's

residence service orders. About 1000 Service Reps are now accessing the system.

### **3.3 Behavioral considerations**

Human behavioral factors that are particularly relevant to the introduction of computer support have been widely recognized: short-term memory limits, the need for closure, the desire for control, motivational characteristics, and the fear of computers themselves. Computer system designers—especially those responsible for the human/machine interface—can effectively incorporate design features which protect against the negative behavioral effects of these factors. This paper will cover many of these features. Perhaps, the anxiety or fear of computers is the most underrated and is discussed next.

Extensive research, experimentation, and observations have been noted by Shneiderman showing that user attitudes—fear of computers—can have a major effect on learning and performance.<sup>2</sup> His data survey suggests that “Novices with negative attitudes towards computers learned editing tasks more slowly and made more errors. Anxiety, generated by fear of failure, may reduce short-term memory capacity and inhibit performance.” In a later paper,<sup>3</sup> Shneiderman notes that part of the training requirements must be to overcome a distorted role of a computer as perceived by the potential user. The media and computer manufacturers have unfortunately characterized computer capabilities by life-like behavior which results in potential trainee resistance in the form of increased apprehension, resistance to technology, and anxiety interfering with learning potential. Shneiderman cautions against predictions such as “people and machines are so similar that with a few years effort, they should be able to produce machines that are superior to people.” He states that “this naive view is useless as a goal and harmful in destroying people’s expectations of themselves and how they will use computers.” Experience with new users of PREMIS confirms this apprehensive attitude generally among the most experienced, entrenched, settled-in population. However, the younger, newer employees often showed enthusiasm and genuine wonder at how the computer can help them do their job. In fact, they tended to press for an education into “how the computer works inside,” rather than limit their involvement to the minimal training on transaction I/O.

The negative attitude towards mechanization has sometimes extended in serious directions. While I have never encountered sabotage, evidence of disgruntled employees damaging a delivered computer system exists,<sup>4</sup> apparently out of frustration at not being properly trained. One case resulted in a prison term for consciously sabotaging a system dozens of times during a period of 18 months.

Jones emphasizes the need for building up the confidence of a potential user by incorporating into the design of the computer system features which give the user the feeling that: "his or her commands will be obeyed; the data are in safe hands; a good and thorough job is being done; and the machine is going to help the user."<sup>5</sup>

The desire for control can be best satisfied if the user is the initiator of all human/computer interface sessions. This principle is easy to support if the end user is getting "information retrieval only" support from the computer, as in the case of the Service Rep. For data system support positions where the computer initiates the need for work, this is, of course, not possible.

The psychological needs associated with short-term memory limits and the need for closure (the completion of a task leading to relief), are interrelated. Excesses can lead to delays, forgotten items, and increased error rate. The key computer-related factors affecting these needs are the careful functional design of human/computer interface sessions into discrete subtasks and the transaction response time delays and variations for a variety of operational conditions. Both of these factors will be examined in this paper. The important point, made by Miller, is that "a psychological closure permits at least a partial purging of short-term memory; waiting time invokes stress." It is believed that more extended delays (as in computer response time) "can be tolerated just after closure rather than in the process of obtaining closure."<sup>6</sup> The idea is to have a system design which does not lead to loss of user concentration, that is, maintain continuity of human thought processes.

Finally, certain characteristics associated with the design and delivery of a computer system can seriously damage the motivation of the user. Three such characteristics which have proven important in PREMIS were generally identified in an earlier paper by Hackman:<sup>7</sup> experienced responsibility for work outcomes; experienced meaningfulness of the work; and knowledge of results. These motivational needs, as well as the other psychological considerations discussed above, will be related to PREMIS field experience discussed later in this paper.

#### IV. COMPUTER SYSTEM CONSIDERATIONS

This category is a catch-all for mechanized system elements, hardware, and software. It also includes the physical attributes associated with the human/machine interface design. Discussed below are examples of design, development, and field experience in this category.

Major elements of the physical architecture of PREMIS are as follows:

- Computer—UNIVAC 1100 series.

- Communications Network—BTL/South Central Bell BANCS THP System resident on Control Data Corporation (CDC) Cyber 1000; Connectivity via 50-kb high-speed lines.
- Terminals—Dataspeed Mod 40/4 series; Input device—QWERTY keyboard with function keys; English language; output device—CRT display.

At a lower level, many operational and functional system considerations remain which can fundamentally affect user performance. Before developing some of these specific issues, I offer some recommendations on the staffing and timing of the system design activity.

#### **4.1 Early HPE influence in system design**

I believe human/machine interface issues can be best understood from a user perspective by professionals in the broad HPE field. Early analysis by HPE professionals can help provide a sound basis for system design decisions. By working with hardware and software professionals, they can substantially avoid unworkable or unreasonable performance requirements on the user. Bennett agrees, noting that human/machine interface design “will be most expeditiously advanced if those already trained in human engineering join the design team rather than if software people attempt to learn engineering.”<sup>8</sup> This position is not universally accepted, but certainly trending in this direction. Bennett goes on to paraphrase McCarn: “. . . experts in computers see well-bounded problems with great precision” but have difficulty in accepting the idea “that human communication is much more complex, and its context more extensive, than was initially conceived by computer programming staffs.” I believe this may have been generally true several years ago, but lately, with extensive teamwork among HPEs and computer science types under our belt, it is not unusual for a genuine interest and insight of the human side to evolve in software professionals.

Regarding the detailed human/machine interface design, one key to ultimate high user performance is simplicity in learning. Shneiderman stresses this by noting that simplicity of design can best meet the bottom-line economic objectives of the system (e.g., computer processing efficiency, storage capacity, communication network load) and still generate the desired user performance in terms of reliability, reduced error frequency, and enhanced satisfaction. Closure, which is strongly influenced by short-term memory considerations, must also be considered. The user receives “great relief when information is no longer needed to be retained. There is . . . a powerful desire to complete a task, reduce memory load, gain relief.” In terms of operational design, transaction response time requirements obviously are tightly coupled with this behavioral consideration. From a functional design view, a direct design guideline is to formulate transactions which allow the

same user to complete each task in sequence. These issues are all part of a fundamental principle: A key to influencing ultimate positive acceptability of a large computer system is the initial approach taken in system definition. I am a believer in formally going through these steps:

1. Functional decomposition, analysis, and allocation—This, in effect, divides up the tasks between humans and machines, making the best use of the strengths of each.

2. Work flows—This shows how the new job will get done in a logical sequence.

3. Human/machine interface design—Finally, from the work flows, the interface points can be supported by detailed I/O design.

Martin refers to these three steps as “functional,” “procedural,” and “syntactical.” If Step 1 is not done, the ball game may be lost before it starts. Jordan recognized this many years ago: “Men are flexible but cannot be depended upon to perform in a constant manner, whereas machines can be depended upon to perform consistently but have no flexibility whatsoever.” Martin, more recently, says it best: “The difference in “thinking” talent—the computer being good for ultrafast sequential logic and the human being capable of slow but highly associative thinking—is the basis for cooperation between man (human) and machine. It is because the capabilities of man and machine are so different that the computer has such potential . . . It is important that system designers . . . do not try to make the computer compete with man in areas in which man is superior.”<sup>9</sup>

A very important system design principle which is being used to great advantage in PREMIS is the concept of modular structure. Each separate system function is architecturally built independent of the others. This applies to internals (processing logic and database structure) and externals (transaction groupings). In addition to the obvious benefits of understandability and simplicity, this modular structure allows an upward compatibility in later releases. It also allows the trialing or “soaking” of a new feature in a limited area before general deployment. In a similar way, by working with the target organization staff, the existing target organization framework (e.g., methods, work flows, job, measurements, etc.) can be examined. User performance deficiencies can then be avoided by influencing the change of the organizational structure when mechanized support is introduced. This is covered later in this paper.

#### **4.2 Input/output design considerations**

##### **4.2.1 Interactive transaction characteristics**

PREMIS has incorporated a variety of Service Rep I/O design features which help smooth the interaction from a human point of view. Many

features were part of the original design; others were designed and delivered via field reviews and feedback from the trial users—South Central Bell. A few are in the process of being delivered. Listed below are those believed to influence user performance and satisfaction. Some specific examples can be found in other papers by Hicks<sup>10</sup> and Ferrer.<sup>11</sup>

(i) Input Mode—Inputs can be classified as either “coded” or “prompted.”<sup>12</sup> With coded input, the user supplies the labels and the data values; with prompted input, only the data values need be entered because the labels are supplied by the computer via form-filling (a mask). The latter is recommended for the end user because of its inherent advantages in reducing input errors and work time, while increasing satisfaction. Prompted input can be further divided into “interactive” or “batch.” With interactive prompted, the computer waits for the user to enter a data value after sending each prompt; with batch prompted, the user fills out the entire mask and the computer gets only one transmission. For a computer system without local intelligence at the point of entry, the batch-prompted mode is much more efficient from a communication system overhead and computer usage capacity point of view and was selected for PREMIS.

(ii) Command Structure—Short, simple, consistent structure emphasizing clarity.

(iii) Single Display Frame (Input)—Transaction inputs always limited to single screen (mask); no input paging required. The particularly awkward paging design associated with the MOD 40/4-system software combination made this very important.

(iv) Single Display Frames (Output)—Transaction outputs usually limited to single screen (mask); however, in some cases, in “prompting” mode, multiple output pages can be returned to the user. Because paging commands require full transmission back to the computer, the transaction response and human search time delays are proving impractical for more than a very few pages. A revised design includes providing more limited output via either more selective data returned or asking the user for more input data to narrow the scope of the stored database information. The concept of putting the most likely choices on the first output page has obvious benefit, but as of yet, such an algorithm has not yet been implemented for PREMIS. This issue is covered in more detail later.

(v) Discrete Transaction Per Function Design—Several independent subfunctions requiring computer assist each has their own independent transactions. This is in support of the user’s need for closure. In addition, this allows optimization of transaction queue control to maximize the priority of the end-user transactions.

(vi) Transaction Sequencing—While transactions are discrete,

each output screen is designed so that over-typing a portion of the input control field, leaving the desired data, adding new data and hitting the SEND key can request the next transaction. This achieves a high level of "Reusability"<sup>13</sup> of former inputs and outputs.

(vii) *Cursor Movement*—Nondestructive forward and backward position-by-position movement and programmable tabbing for automatic movement to the beginning of next and previous input fields.

(viii) *Prompting*—Software logic which detects some shortage of input information and requests the user to provide additional input.

(ix) *Menu Selection*—A type of prompting where a limited set of valid responses is presented on the screen and the desired response can be chosen by keying in the choice, the number of the choice, or by positioning the cursor next to the choice.

(x) *Parameter Defaults*—For each community of interest (e.g., city or state), a set of table-driven parameter values which represent an agreement by the user on what are normal, or the most often used, values that can be assumed if the field is left blank.

(ci) *Minimal Input*—The user need enter only a "shorthand" input, and the computer employs some pattern recognition logic to fully interpret the input. (This feature has sensitive system performance implications and is discussed later.)

(xii) *User Control*—The user initiates and controls all human/computer interactions.

(xiii) *User Messages for Transaction Failures*—Currently, the screen goes blank for certain transaction system failures and uninformative messages are returned for others. This is a serious cause of user discomfort. Meaningful messages should be returned in every case, and efforts are underway to make this possible.

#### **4.2.2 Multisystem considerations**

(i) *System-to-System Switching*—Currently, the Service Rep has access to another mechanized support system in addition to PREMIS via the same terminal. During the same work session, one switch (or more) between systems is often necessary. The current log-off-log-on procedure involves several keystrokes and response waits. Serious user dissatisfaction with the procedure has led to the recommendation of a quick-switch procedure of perhaps just one function key. In addition, the entry mask for the appropriate system being logged on to should appear automatically on the screen.

(ii) *System-to-System Consistency*—There exists system-specific function keys and dialogue mnemonics for each of the systems accessed by the same user. Lack of system-to-system consistency is a training problem and a potential source of error. Multisystem agreement on design is currently being negotiated.

### **4.2.3 Screen display physical characteristics**

Most terminals in the field today generally provide acceptable physical characteristics from a human performance point of view. In the case of the Dataspeed MOD 40/4 used for PREMIS, the users reacted favorably when asked about screen brightness and contrast and character size, sharpness, and spacing. The only negative reaction to the screen was the eye strain associated with users whose terminal screen faced a nearby window where the sun glare was strong. This was easily remedied by drapes or blinds.

### **4.2.4 Information display characteristics**

In terms of information display, the PREMIS design was open to human performance considerations. A variety of human-engineered screen information display principles were followed in the PREMIS design.<sup>10</sup> These design features were incorporated to improve the cognitive behavioral response of the user, mostly by minimizing data searching, increasing awareness, and by increasing recognition and distinction. PREMIS users gave very high values on the ease of use of this information display (8 to 10 on a scale of 1 to 10, 10 being the highest). However, no experimentation has yet been done on PREMIS to determine sensitivities to varying display techniques.

(i) Positioning of Input Data on Screen—As close as possible, input data should be entered in a left-to-right and up-to-down sequence consistent with the work operations.

(ii) Input Data Preservation—All input data appears on the output screen completely and precisely in the same position.

(iii) Positioning of Output Data on Screen—All data displayed on an output screen (mask) always appears in the same spatial position in multiple operations to minimize user search time, “reduce disruptive movement and help highlight the impact of the last operation.”<sup>14</sup>

(iv) Highlighting—For key output parameters, where particularly high detectability is desired, highlighting is used.

(v) Data Labeling—All variables on I/O screens appear as pairwise identifiers and values.

(vi) Number Displays—Long number sequences in the same field are broken down into subsets.

(vii) Screen Partitioning—Separate, dedicated areas for input, normal output, and user messages.

(viii) User Message Semantics—All user error control and/or instructional messages are constructive and supportive, not condemning and confrontive.<sup>3</sup>

(ix) User Message Syntax—All user error control and/or instructional messages are in English (no cryptics/codes).

#### 4.2.5 System engineering considerations

While ensuring understandable, simple human/computer interface features, the designer must also consider the effect of these features on the hardware/software capacity of the system. Gilb notes, "Extensive use of humanized input designs can easily result in substantially greater consumption of central processing cycles and secondary storage search time over programs which effectively place the equivalent work processes on human beings."<sup>4</sup> Care must be taken to specify I/O design features which have gone through the human/system engineering step of identifying a favorable payoff versus penalty ratio. This important design activity is not always done in computer systems and, for some features, in the PREMIS project also, subsequent field analysis has sometimes suggested the wrong emphasis.

For example, the misuse of prompting and menu-select features associated with interactive transactions can be a potential source of unexpected (and undesired) induced load. The additional transactions generated from these features can significantly degrade the system load capability. That is, the payoff from reduced training costs, reduced keystrokes, and reduced entry errors may not offset the economic penalty in increased transaction load. A recent example of this is the minimal input feature in PREMIS.<sup>15,16</sup>

This feature allows the Service Rep to key into the computer as few as the first four characters of a street name. The computer searches the database for a match and, if found, returns the desired information. If more than one internal match is found, a menu is returned to the terminal screen and the user, after perusing the menu, selects his or her choice and resends the transaction. Training was provided on how to use this feature but not when to use it. Analysis of audit tapes showed that, for some locations, 50 percent of the time the menu-select response was happening. Not only was this causing an unacceptable additional load on the system, but, the user was paying a penalty in increased time per event. This is evident from Fig. 1. Figure 1a shows that if a second transaction per address is required more than 30 percent of the time, it takes more time on the average to use minimal input than to use full spelling. Figure 1b shows the rate at which transaction volume increases for additional transactions per address beyond the 15 percent error rate assumed for full spelling.

The potential user and system costs of what was intended as a humanized input design was uncovered in an operational review of the system. However, this same review found considerable variability between locations on the percent of time that minimal input required additional transactions. These findings suggested that rather than remove the feature altogether, the objective should be its efficient utilization. Work was undertaken to develop the software analysis and

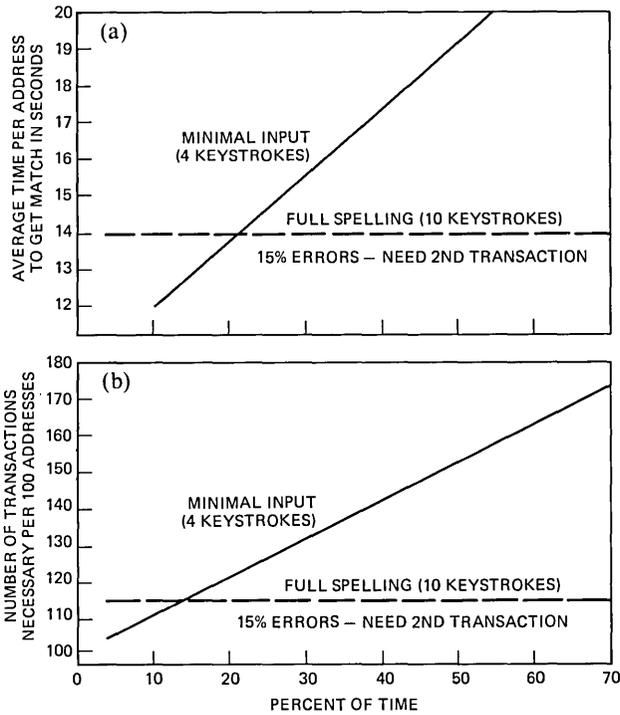


Fig. 1—Impact of minimal input. (a) Service Rep time. (b) Transaction volume. (More than one transaction is required to get a match.)

monitoring tools which could be applied per location, so that usage guidelines per location could be developed. Until these software tools are available, however, the potential cost was deemed too great and, as a short-term remedy, the recommendation was to stop using minimal input.

Another issue which was not properly engineered in the initial delivery of PREMIS was establishing and enforcing limits on the amount of computer processing or output pages which any individual Service Rep transaction could consume. Assumptions were made about real world data which proved not to be true. As a result, information retrieval transactions were allowed which searched vast areas of the database for a "hit." After minutes of chewing up processing time, having failed to find a certain hit, a very large number of possible menu-select choices are output back to the Rep's terminal requiring many pages to go through. In an environment where the Rep is negotiating with a customer on the telephone, these occurrences resulted in Rep frustration leading to a bypass of PREMIS altogether. A careful review of these cases resulted in changes in the software logic and controls to include firm processing and output upper bounds for

certain transactions, while providing user messages and enough information to proceed through the customer contact. For example, it is often possible for the user to change the input in such a way that the search is narrowed and the output limited. (To illustrate, if a Rep is dealing with a customer moving into an apartment complex with many units and the units are identified by a combination of letter and number, say Apartment A-162, the customer would often state their address as 162-A. Rather than output all apartment numbers as possible matches, a quick perusal of a single output page can quickly identify this inconsistency which can be easily handled by overtyping the reversed characters and resending the transaction.

Unfortunately, the software processing safeguards were not applied to all potential overload situations. Providing all possible database matches on the first four characters of the keyed-in street name caused the entire system to go down several times in the first few weeks of 1981. These occurrences were all associated with a geographic area which happened to have many highways. By matching HIGH, all highways were attempted to be sent back to the user. However, the software output buffer could only handle 200 of these in one transaction, and this number was exceeded. The software had no programmed safeguards against this overload, and it caused the whole system to lock up. To make matters worse, no output message identifying the specific transaction or the condition was sent to either the user or the computer center. To relieve this condition, the software processing was changed to limit potential matches to 50 and, if more than 50 occur, send none back to the user; instead, provide a new user output message stating "NO EXACT MATCH ON ADDRESS ENTRY—TOO MANY SIMILAR ADDRESSES TO DISPLAY."

In the same vein, other obvious user-oriented I/O features can cause substantial induced load on the system. Two examples which should not be incorporated into the design without careful analysis are (i) allowing the end user access to a status transaction which provides what state a previously entered transaction is in, and (ii) allowing the end user to enter later transactions without first receiving a response from the first.

Both of these capabilities tend to further degrade an already overloaded condition.

#### ***4.2.6 Novice versus experienced variations***

Several papers in the field today suggest that there be two varieties of I/O design: one for the brand new end user, as part of a training strategy, and another for the expert user. The novice version would have very simplistic inputs and expanded outputs associated with prompting, menus, and full-error messages; the expert version would

be optimized for minimal load on the system, that is, flexible inputs and outputs, but designed to require only a single human/computer interaction per task. The idea is to do a "switch" per terminal or per end-user identification when a certain proficiency level is reached. It has even been suggested that the computer itself track the progress of each user, and via an adaptive switch, change the I/O design from novice to experience accordingly. In PREMIS, only one I/O design exists, but for the new user a Training Subsystem is provided which allows training transactions to be tracked and processed independently from live transactions using a training database. Further field analysis is necessary to determine the cost effectiveness of these various approaches.

#### **4.2.7 User errors**

No matter how hard system designers try to make transaction inputs simple, user errors will always be somewhat of a disappointing surprise when the system is in operation. As part of the system design, there should be easy monitoring (mechanized) of error patterns so that some action can be taken in response to frequent, costly errors. The system modification might be in the semantic or syntactic elements of the input or might be solely associated with training.

A mechanized means of sorting user errors by the Service Rep has recently been designed. The statistical information is conveniently displayed in report format and is available on request by management personnel. This report capability is scheduled to become operational in 1981. It is anticipated that feedback to the Service Reps on their error rates will be a positive influence on their performance and attitude. A user error rate objective standard will be established as soon as a substantial amount of data is collected and analyzed. This standard, which may vary depending on local conditions, should also be made known to the Reps. Experimental results<sup>17</sup> have shown that in a repetitive human/machine interface task, the combination of feedback on both individual performance and an accepted (high) standard can have significant positive effect on reducing user errors, while uplifting individual morale and uplifting the organizational climate. A 15 percent decrease in the average error rate was ultimately achieved. In addition, the results of observations and interviews suggests that the increased pressure on performance did not cause any undue physiological or psychological stress.

#### **4.3 Operational considerations**

In my experience, three computer operational considerations are by far the most significant: availability, transaction failures, and transaction response time.

The users should be (but usually are not) prepared for the effect of these on their job performance. Our experience has shown that these factors have a potentially drastic and traumatic effect on the user, especially when they first use the system. They must be prepared for these effects.

#### 4.3.1 Availability

The availability of interest to the user is the end-to-end variety; that is, the probability that all components of the system are functioning at the time the user desires access.

Repeated availability problems caused by equipment outages can result in long-term attitude problems. If a terminal, for example, goes down often (low mean-time-to-failure), it not only interferes with work performance during the outages, but it is also viewed with distrust when it becomes workable. This results in the user checking and double-checking results constantly, thus, increasing task time.

Often, however, there are socio-technological obstacles in the way of providing realistic availability information to potential users. For example, an on-site review associated with FACS\* uncovered the following:<sup>18</sup>

Overall Application User Opinion: "The system is down on the average of 3 hours per day!"

Operations Staff: "Not so! Availability is in the high 90's!"

How is this widely different view possible? The system administrator is generally measured on how often and for how long the system is down. Therefore, it is highly unlikely that availability—as seen at the user's terminal—is either known or, if known, publicized. The traditional fallacy is the official reporting of "computer main frame downtime during business hours." There are two serious problems with this:

(i) As mentioned earlier, care must be taken to deal with end-to-end availability—the product of on-line hardware/software/communications links reliabilities. That is, the terminal, the transmission lines, the message switcher(s), etc., must all be considered to get a true picture of user availability.

(ii) Often, just as much work is done during off hours to keep the system up to date. Availability can be just as important here.

A classic case of a difference between perceived availability (by the user) and measured availability (by the operations staff) occurred during an on-site operational review of FACS.<sup>18</sup> At the user's terminals, on many occasions during the day, transactions could not be entered

---

\* FACS: Business Information Systems Customer Service/Facility Assignment System, FACS is a Bell Laboratories-developed mechanized support system, an early version of which was trialed in the late 1970s and is now being redesigned.

into the computer. On the other hand, the official availability of the computer for the same time frame was in the high 90s. After intensive investigation, it was found that a major cause of the confusion was tape handling. As part of the recovery procedures, audit tapes are generated saving a transaction history. These tapes would each handle about 40 to 60 minutes worth of activity. After each tape was full, it had to be manually removed from the tape drive and a blank tape mounted. This takes up to 3 minutes per occurrence. During these time intervals, no user transactions could be accepted by the computer; however, they were not considered outages in terms of reporting availability.

A very interesting user behavioral effect results from these multiple, very short outages. Say, a user attempts to enter a transaction during one of these outages. Normally, a "message received" appears on the screen; in this case, the screen goes blank. The user mentally records that the system is not available, but does not know exactly how long this condition has existed. Suppose now, another try is made. Again, a blank screen. The user gives up, walks away, and does something else. Thirty minutes later, the user walks over and tries again. Nothing. Another mental note: "The system's been down anywhere from a half hour to an hour." Two outages of maybe 3 minutes each resulted in a magnified user performance degradation. If nothing else, this phenomenon suggests that many sporadic, short outages can have a significantly greater effect on user performance than a single, extended outage. For extended outages, telephone calls were generally made from the computer center to each user work center informing them of the situation thereby limiting the perceived availability problem to the actual. (The audit tape problem was subsequently relieved by the introduction of a mechanized transfer process.)

During a PREMIS field visit, the negative impact on the Service Rep of many short outages was further magnified. Over a period of 4 hours there were at least 6 system outages of 5 to 20 seconds each. The total actual time the system was unavailable was less than 1 min. (None of these occurrences was officially included in availability statistics because the minimum outage reportable was 5 minutes.) However, every Rep accessing PREMIS when an outage occurs is also on the phone to the customer. There is, of course, no way of knowing when the system will again be available. The Rep can either choose to bypass PREMIS and resort to back-up procedures (extra time, motion, and chance of error) or skip the portion of the customer contact associated with PREMIS and hope the system comes back quickly. The latter was generally done. However, the Rep must formally log back on to PREMIS each time and either call up or clear out of the system any message or partial messages that were in process at the time of the outage. This

might take a minute or two to accomplish. The cognitive load associated with these required actions in concert with verbal continuity of customer negotiations created obvious stress on the Reps. They were visibly disturbed and often made keystroke errors, as well as wrong decisions. Worse, the fact that these outages were not counted (because of their short duration) as official availability measurements put no pressure on the data systems organization to investigate the problem and attempt to resolve it.

Another important issue is the fallacy of partial availability. It has been my experience that the hardware/software portion of the system does not "fail softly" as Martin puts it. No matter what mechanized fall-back processes are built in, the user is usually dramatically affected. What is needed mostly, I believe, are not gradual, stepwise degraded user procedures, but a simple alternate design that bypasses the computer altogether. The user must be able to conduct business for periods of time when the computer is totally down or not available anyway.

Simply waiting for the system to come back up is not acceptable in dealing in real-time tasks with a third party (the customer) such as the Service Rep's job. In effect, the fall-back position closely resembles the procedures and aids which were in the pre-PREMIS environment. After these procedures are designed and tested, they should be included as part of the formal training. In PREMIS, the use of periodic printouts (albeit somewhat out of date) keeps the business going reasonably well.

#### 4.3.2 Transaction failures

Transaction failures are defined to be cases of user-initiated transactions which do not process to successful conclusion. The causes of these failures can be hardware-related or software-related. A high rate of these failures, especially if combined with poor user notification, can seriously affect user performance. An example of such a situation was observed with FACS. Table I shows representative failure statistics.<sup>18</sup>

The causes of these failures were spread among management soft-

Table I—Failure data

Month	Total Transactions	Total Failures	Failure Rate (Percent)
August, 1978	140,770	2873	1.8
September, 1978	135,720	2992	2.2
October, 1978	158,175	1987	1.3

(For comparison purposes, several other large-scale systems manage to maintain failure rates of less than 0.1 percent.)

ware bugs, application software bugs, and database logical and physical inconsistencies. The effect on all three classes of users was serious: the operations staff received all the failure messages independent of the transaction entry source. They were inundated with paper each day and generally were unable to analyze, pattern, or react in any effective way. The paper continued to stack up each day so there was virtually no real-time support provided. This left them frustrated at their obvious ineffectiveness. Each aborted transaction was taking up to eight days to evaluate. The database maintenance clerks were continuing to do their database changes generally unaware which of their transactions “bombed out,” because they were not notified by output messages or calls from the support staff. Often, they would discover these events by later transactions not processing properly, that is, the database change may have tried to load some inventory but since it never worked, later accesses against this inventory would fail. This caused frustration on their part because they would receive calls from the application users when their transactions did not work. The application users were negatively affected because their ability to serve the customers’ needs in a timely way was degraded. All of the users who initiated transactions that failed were further frustrated because failure messages were not returned to the entry terminal! The lack of job continuity and efficiency, because of these transaction failures, was having a measurable effect on the acceptability of the system from a user point of view. A variety of fixes were designed—the prime effort, of course, focused on improving the quality control of the transaction processing. In addition, the transaction failure messages (in English!) were returned to the initiator’s terminal, and extensive training and documentation was developed to help people cope with these events, including straightforward error correction procedures.

#### **4.3.3 Transaction response time**

Transaction response time is defined to be the interval between sending the transaction until the first element of the response appears. It has many of the same characteristics as availability and transaction failures in trying to get a true handle. The software designers are sensitive to poor transaction response time perceptions because these perceptions imply less than optimal programming in many cases, or the system administrator may have installed a queueing structure that is inefficient from a user point of view or the communications manager may have installed a poorly balanced communication layout.

It should be noted at this point that fast response time is not cheap. Neither computer processing power or telecommunications network capacity can be indiscriminately increased for fear of negating the economics of the system itself. Therefore, extreme care must be taken

in specifying response time objectives. In PREMIS, the various transactions are grouped according to specific users and discrete response time requirements are established for each grouping. A table-driven queuing structure built into the computer management system supports this structure. In addition, communication lines vary from dedicated to a single terminal to heavy time sharing with others. Martin points out—and I agree—that, in practice, justifying response time based on pure time economic considerations is virtually impossible. It is the psychological factors which can affect user performance. These factors include “expectance,” “chunking,” “short-term memory,” “closure,” etc. For example, experiments have shown short-term memory limits are severely stressed when the response time increases. This stress results in user discomfort, as well as increased error rate. Experiments have also shown that user annoyance is very low if the response time is under 8 seconds; some annoyance appears from 8 to 13 seconds, and the user becomes very annoyed if the response time is greater than 13 seconds.<sup>19</sup> This is very consistent with Service Rep reaction on PREMIS. Williams reports on experiments which show that for fairly routine data entry tasks “User performance deteriorates with system delays of greater than 15 seconds.”<sup>20</sup> Other experiments have shown a discontinuity at 15 seconds. It becomes more than an annoyance and disrupter—it becomes a demoralizer and reduces motivation.<sup>6</sup> I do not have statistical data on user error rates in PREMIS as a function of transaction response time. However, a recent study by Barber,<sup>21</sup> showed that for a response time of, on the average, 8 seconds, the subsequent transaction had an error rate of about 0.10; when the response time was increased to 20 seconds, the error rate increased to 0.25. For the case of the Service Rep position, Table II gives recommendations based on qualitative behavioral observations (mostly my own judgmental estimates of disturbance levels). These values are also based, unfortunately, on systems without local terminal intelligence so that every user action requires full transmission to the computer: The recommendations given have the built-in presumption that the

Table II—Response time objectives

Action	During Busy Hour (90 Percent Confidence) (Seconds)
Bring-up screen	4
Paging	4
Input error (syntactic only)	5
Input error (all others)	7
Simple transactions (direct access key to data)	7
Simple transactions (limited search in data base)	10
Complex transactions (extensive search in data base)	15
Transactions allowing parallel tasks	25

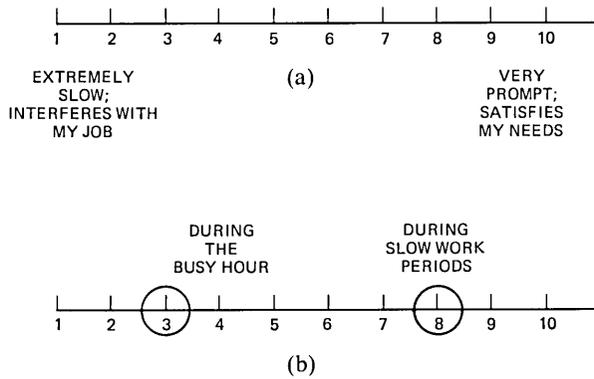


Fig. 2—(a) Original scale for Service Reps opinion of response time. (b) Revised scale.

Rep is willing to wait somewhat longer if they perceive the computer task to be more difficult (simple versus complex transactions) or the system is in a high load condition (busy hour). In other words, the user discomfort level is related to the extent to which the real delay exceeds expectation.

Dealing with response time as seen by the user can have some very interesting and important quantitative and behavioral effects. Recently, the PREMIS users in Memphis reported the response time for simple transactions to be about 30 seconds, while users in Birmingham and Jackson reported a 7-second average. An examination of the terminal and communications network layout revealed that there were excessive terminals in Memphis competing for the same line.

A 1979 PREMIS field review in Jackson, from the Service Rep's perspective, yielded the following:

The Service Rep's attitude towards response time was obtained via formal questionnaires. However, a real-time adjustment to the questions had to be made. Originally, the Service Reps were asked to indicate their opinions of response time. They were asked to choose an answer based on the scale shown in Fig. 2a.

It quickly became apparent that the Reps were having great difficulty with this request. Informal probing led to a decision to pose the request in two parts (See Fig. 2b.), which ultimately provided a true picture when the stop-watch measurements were analyzed (See Fig. 2b.)

Stopwatch measurements were extensively taken at three different user positions over a one-week period. In addition, the users' behavior was observed, and they were requested to complete questionnaires.

Tables IIIa, IIIb, IIIc, and IIId provide some key statistical summary data on response time for the most important transactions. Referring to the table, a success, or "hit" condition refers to a user transaction

Table III—Response time characteristics

(a) Variations across locations (means)				(c) Variations across transaction outcomes							
	Terminal Locations	Computer Time (Mean)	Stopwatch Time (Mean)	Stopwatch Times							
				Mean	Mode	Ranges					
Hits (second)	A	4.39	8.87	Hits (second) No-hits (second) Outcomes	10.68	6.0	5-60				
	B	3.90	12.09								
	C	4.80	10.38								
	Average	4.27	10.20								
No-hits (seconds)	A	5.93	8.15					1	9.61	9.3	5-19
	B	6.62	13.63					2	9.33	6.0	6-19
	C	8.45	11.35					3	9.38	8.0	5-29
	Average	6.56	10.65					4	11.68	8.0	5-21
								5	11.86	7.0	6-27
								6	9.65	7.0	6-18
				7	11.43	11.0	7-26				
				8	10.00	6.5	5-23				
				9	29.64	26.0	22-54				
				10	11.23	7.0	6-30				

(b) Variations across locations (details)					(d) Variations across business day (means)								
	Terminal Locations	Stopwatch Timings			9-10 a.m.	10-11 a.m.	10-12 a.m.	12-1 p.m.	1-2 p.m.	2-3 p.m.	3-4 p.m.	4-5 p.m.	
		Mean	Mode	Ranges									
Hits (second)	A	8.87	6	5-29	6.3	10.2	12.5	7.0	8.7	11.5	7.5	8.1	
	B	12.09	11	5-60									
	C	10.38	6	5-39									
No-hits (second)	A	8.15	7	5-39	10.6	17.5	12.2	8.8	9.3	9.8	9.9	9.1	
	B	13.63	11	6-54									
	C	11.35	12	5-35									

which returned the requested information to the user; a “no-hit” condition implies an undesirable result usually accompanied by an error message. The response time objective was 7 seconds. Table IIIa shows the variation in mean response time across users’ locations for both user response time (measured by stop watch) and computer response time (automatically measured by the computer, representing only the main frame residence time). Note that the use of the computer response time as the only official reportable number had provided a false sense of security. The variations between user sites is significant; therefore, a more detailed statistical breakdown is shown in Table IIIb. (It turned out that the computer polling algorithms were out of balance causing these variations). The response time variability between transaction outcomes is shown in Table III (one specific result was the redesign of the software processing for the ninth no-hit condition). Finally, Table IIIc shows the variation in response time across the business day. It showed the longest response times before and after lunch time which is consistent with the known customer activity level variations (this information resulted in recommending to the maintenance and computer center staff that offline transaction activity be avoided during these time periods).

Based on my observations and later interviews, I believe the wide variability in response time for different outcomes was apparently at least as troublesome to the users as the mean response time. This is no longer surprising based on several recent experiences and experiments appearing in the literature. First of all, what variability is detectable by the users? A recent experiment by Butler<sup>22</sup> showed that for a response time in the 8-second range, a 2-second increase was consistently detectable by 75 percent of the subjects. Now, in terms of discomfort, Miller notes “Increasing the variability of the output display rate produces both significantly decreased human performance and a poorer attitude toward the system and the interactive environment.” The results of this experimentation showed that user satisfaction, measured on a relative scale, decreased by 25 percent as the response time variability went from low to high. In addition, the amount of time required by the user to analyze the output and take action increased by 15 percent under the same conditions.<sup>6</sup> In addition, not preparing the user for this compounds the negative effect. Shneiderman, in reviewing several experiments, states “If the variance of response time is small, users incorporate the waiting into their work patterns by pre-planning future queries or attending to other functions, but if the variance is large, users must maintain a continued high level of awareness and become anxious if response time grows.”<sup>22</sup> Quantitatively, Martin recommends as a rule of thumb that the standard deviation of response time should not be more than half the mean

response time. Therefore, for an 8-second mean, two-thirds of the real response time should be less than 12 seconds.

Finally, a condition with both flaky transaction failures and slow and erratic response time is more damaging to user performance than the sum of each effect. A recent review of PREMIS, which had such a combined condition, disclosed high-user frustration at not knowing if a blank screen meant something wasn't working or the system was just slow. On-site observations showed that they would often send the same transaction several times in succession, thereby compounding an already overloaded input queue. This phenomena of an induced load was supported by data on audit tapes. An analysis of the data<sup>16</sup> revealed that as the system approached the busy-hour, users repeatedly hit the send key in their frustration at the slow response time. The Service Reps openly admitted that they were doing this. Figure 3 shows the measured response time characteristic versus offered load for the system as it existed in September, 1980. The effect of the induced load on response time is shown for a particular Monday morning busy hour, where a serious computer problem caused repetitive entry of very long running transactions. In these cases, each transaction was, in effect, worth many normal transactions. It is estimated, based on audit tape data analysis, that this user-initiated phenomena combined with transaction processing difficulties increased the average response time from a normal 10- to 20-second range to a 60- to 70-second range.

## V. TARGET ORGANIZATION/ENVIRONMENT CONSIDERATIONS

It should not be surprising that the performance of a very large computer-based application can be greatly affected by (and can greatly affect) the target organization/environment. Every attempt should be made to analyze this impact up front and force appropriate adjust-

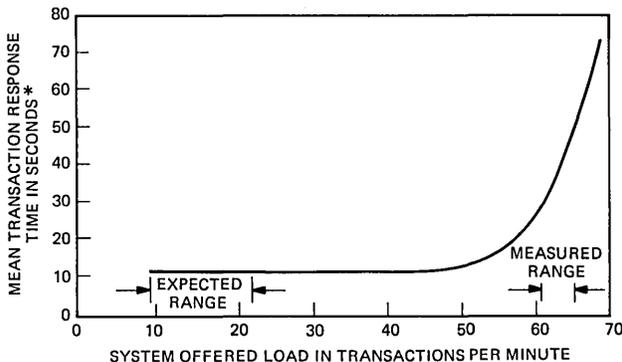


Fig. 3—Load characteristics.

ments in the organization or environment to allow a graceful introduction of the computer system. The first consideration—the relationship of the computer tasks to their appraisal through existing job performance measurements—is extremely important and often underestimated, especially for the application user. This can be the focal point of serious user performance problems. The Service Rep job is one such case.

### ***5.1 Relationship of existing user job performance measurements to use of the computer***

There are three major possibilities here:

(i) Does the Rep perceive the computer system to have a positive effect on their performance measurements?

(ii) Are the measurements themselves tied to bottom-line system performance objectives?

(iii) Are the Reps provided feedback on how well they are doing?

Assume a Service Rep is negatively measured, at least in part, if the customer contact is broken off for any reason and the customer is put on hold. In a pre-PREMIS environment, paper records are accessed during the contact; with PREMIS, the computer is accessed via the terminal. The computer-based records may be more accurate than the paper records. This guarantees better service to the customer which presumably has positive satisfying value to the Rep. But the more significant behavioral influence on the Reps is the effect of the computer on their job performance measurement. So, for example, if the system response time is so slow that Reps believe that customer contact breaks will increase, they are going to be unhappy.

An on-site review of the Service Rep's use of PREMIS was held in Jackson, Mississippi, in 1979. Under the existing measurement structure at that time, breaks in customer contact played an important role. That is, Reps with frequent and lengthy contact-time breaks would be negatively effected. Personal interviews with the Reps included the question, "How do you perceive the effect of PREMIS on your average contact time? Their verbal responses generally showed a keen sensitivity to the issue and a uniform distress over the transaction response time degradation in the busy hours (the observation of response time as a sensitive behavioral characteristic will be further amplified in the next section). There is no easy way out of this without cooperation of the target organization. A slight increase in contact-time breaks and break durations may, of course, be a small price to pay if a substantial reduction in key information errors occurs which ultimately translates into reduced total work or some other bottom-line measure. One such measure is input error frequency.

A data collection summary<sup>23</sup> of the effect of PREMIS on the frequency

of errors in an important parameter makes this point. One of the items that a Service Rep must determine is the address of a customer who wants service so that the electrical connections can be assigned and hooked up to the right house, as well as provide the downstream accounting people with the right place to mail bills. The Service Rep places the address on a contact memo which later is used to generate the formal service order. If the address is incorrect, a variety of difficulties can develop, depending on how long the error goes without being detected. If the error is caught early, the order is sent back to the Rep to change and re-submit; if the error succeeds in surviving the entire service provisioning process, billing discrepancies may occur. In any case, there is substantial cost associated with each of these errors. One objective of PREMIS is to provide the Rep with on-line address verification support during the customer contact, thereby avoiding address errors up front. Table IV summarizes the history of address error statistics since PREMIS was cut live in Jackson, Mississippi. The pre-cut-live error rates were on the order of 10 percent in 1978.

It is generally acknowledged that PREMIS—both from an ease-of-use and a database-accuracy point of view—allowed this dramatic improvement.

The key is to revise the job performance measurement closer to the bottom line instead of to a secondary parameter. This is easier said than done, especially if this condition is not recognized before the computer system is installed.

Recently, the Service Rep job performance measurements were changed. The emphasis is now on customer service rather than on the contact time. That is, questionnaires on how gracefully and efficiently the job is being done, from both a behavioral and efficiency point of view, are now being sent directly to customers on a per-Rep basis. If an address error was not caught early and proliferated all the way to the installation step, the customer would be negatively affected (by the missed appointment) and would so report. This change of emphasis on individual performance measurements was implemented totally

Table IV—Address error trend

Quarter	Total Service Orders	Orders With Address Errors	Percent
1Q79*	58,707	3,919	6.5
2Q79	77,746	4,570	5.9
3Q79	80,800	4,154	5.1
4Q79	69,217	3,305	4.8
1Q80	68,717	2,730	4.0
April-May	47,167	1,388	2.9

\* Premise begins live operation.

independent of PREMIS; however, the beneficial effects to PREMIS are obvious.

Of utmost importance is the direct feedback to the user of both the performance objective and the user's personal performance. In the famous Hawthorne investigation on worker productivity, the investigators found that the experimental subjects demonstrated improved performance regardless of the variables. Many years later, H. M. Parsons concluded that the two major influences on the workers were (i) they were given frequent feedback on their performance, and (ii) they were aware that their performance was directly tied to their salary increases.<sup>24</sup>

In addition, the Service Rep is now handling new functions on a PREMIS environment. The associated new tasks, of course, take additional time; but from a total organization point-of-view, the functions are being performed more efficiently. This very important consideration is discussed next.

### ***5.2 Relationship of the existing organization-to-function mapping to an optimal strategy***

In an organization/environment where the work is broken down into several discrete functions performed by discrete work groups, the flexibility to re-distribute certain tasks across traditional boundaries can often improve the effective use of a computer system. A case in point are the traditional organizational entities in the BOC, where several vertical management structures each are involved in the processing of a horizontal customer service order. To illustrate the example, consider the roles of three of these organizations: The establishment of the service order (Service Reps); the assignment of facilities (assigners) and; the installation of the facilities (installers). PREMIS provides on line, up-to-date information up front to the Service Rep associated with the service history at a particular address. Now, one specific issue for which PREMIS can provide benefits is the interfering station condition. Simply put, it is a request for new telephone service at an address where previous service has not been terminated. This most often occurs when the new customer's request for service is processed before the processing of the disconnect service order of the existing customer at the same address. Before PREMIS, this condition was often not detected until the installer goes to the address, thereby causing significant extra work by the Rep, assigner, and installer. With PREMIS, the information about the existing customer is displayed on the screen while the Rep is negotiating with the new customer. Now, if the Rep's job can be expanded to resolve this condition right up front, significant wasted work can be avoided downstream; that is, a modest increase in the Rep's work load is certainly justified from a

Table V—Interfering station condition (isc) workload analysis. Assumptions: Percent of inward service orders for which an ISC exists is 7.3%; percent ISC identified by Service Rep with PREMIS is 92.0%.

Workload Characteristics	Service Rep	Assigner	Installer	Other
Average wasted work effort per ISC occurrence <i>not</i> detected by Service Rep	12 min	20 min	45 min	10 min
			ISC Identified by:	
		Assigner	Installer	Other
Annual benefit per line with PREMIS		4.7¢	4.4¢	0.56¢
Annual expenses per line with PREMIS because of increased Service Rep work efforts	(1.6¢)			
Annual benefit* (cost) per year with PREMIS for 4 million line entity	(\$64,000)	\$188,000	\$176,000	\$22,400

\* Annual benefits per line are linear.

corporate point-of-view if a much larger savings is realizable by the assigner and installer. An economic study by Hafelfinger,<sup>25</sup> based on on-site observations, interviews, and several months' worth of data, convincingly demonstrated this. South Central Bell initiated new work flows in a PREMIS environment across the relevant organizations, thereby allowing pre and post data to be gathered. Table V summarizes the key results.

Thus, per PREMIS system, almost a quarter of a million dollars can be saved yearly by optimizing the task requirements across organizational boundaries. (It is believed that these figures are conservative and much more savings is likely.)

It is important that the system design and development plan include an activity associated with the optimal re-association of functions across organizational boundaries. The best vehicle and methodology to accomplish this depends on the relationships of the system design and customer organization and the complexity and breadth of the mechanized functions. A vehicle that has proved effective for PREMIS is the formal establishment of a working committee of several BOC representatives working under the auspices of the AT&T Residence Service Center, with consulting support from the Bell Laboratories PREMIS HPE group. The committee recommended a generic approach to the changes in existing job functions in the Residence Service Center to best use PREMIS. Detailed before and after work flows formed the basis of the method used in this activity and proved to be an effective device.<sup>26</sup> An adjunct method—to be used in conjunction with work flows—has also shown some advantage for certain environments, i.e., the formal groupings of like tasks into a series of work modules.<sup>27</sup> These modules have characteristics as follows:

- (i) Not splittable between people.
- (ii) One (or more) make up a person's job.
- (iii) Each can be independently assigned to the appropriate group.

The strategy is to define the work flows (and work modules) as part of the system design process early enough to influence the adjustments in the target organization's job structure. The work modules can also be used as the basis for organizing procedural user documentation and training, independent of job definition.

Make no mistake, there is a real threat to the success of a computer system if the organizational dynamics are not well understood and dealt with. The worst possible condition exists when the computer system is almost totally supporting one department but dependent on another to increase its workload to make the system work effectively in a mechanized environment. To ensure an effective working system, an early corporate, multidepartmental commitment is vital. It has been my experience that when it is shown that additional human effort in one department can save much more human effort in another department and improve the corporate bottom line, with the organization doing the extra work getting the credit and a good public relations job is done, then the chances of acceptance will be improved. This was the case of the ISC discussed earlier. On the other hand, if workers in one department are asked to increase (or even change) their jobs to make it easy to program a computer helping another department there is extensive resistance.

### **5.3 Customer interface**

Another potentially important environmental influence on user performance is the customer interface. This is appropriate when the performance of the user tasks includes interfacing with the customer of the service being provided. This condition exists for the case of the Service Rep where the customer is initiating action with respect to obtaining (or changing) telephone service.

An earlier example discussed the potential problem of the target organization using customer contact time as a measure of a Service Rep's job performance. This presumes direct involvement with the customer over the telephone.

In the telephone work mode, the Service Rep is visually interpreting screen responses from the computer, while simultaneously listening to the customer provide additional data all under contact-time pressure. We know from experimentation<sup>28</sup> that there are load stress bounds beyond which there can be some degradation of human performance if, during a person's job, he or she is time-sharing between auditory and visual inputs.

PREMIS is also being used in an environment where the Rep is face-

to-face with the customer. Now the behavioral damage to the Rep of a slow computer response time is compounded. The visibility of the interaction to the customer has been dramatically extended, thereby introducing a "fish-bowl" effect on the user. Over the phone, perhaps the Rep can disguise the delay by gathering additional information for a time, and then, out of desperation, discuss the weather or last Saturday's football game. In a face-to-face contact, I have seen the Rep and the customer staring nervously at a blank screen for a time that seemed like forever, but which was only for about 20 to 25 seconds. This was very disturbing to the Rep, especially if the response time increased during the busy period, with other customers waiting in line (also watching the blank screen).

Corrective strategies to minimize the pain in this environment had to do with adjusting the work station layout and changing the workflow steps. When possible, screens were not made visible to the customer. This was relatively easy to do when the customers and Rep were separated by a counter or barrier. For layouts where there is free flow of customers throughout the work area (as in many Phone Center stores), the terminals were put off in a corner and accessed privately by the Rep after he or she had collected the pertinent information from the customer on a scratch pad.

## **VI. SUMMARY**

The introduction of a very large computer-based application system to an existing work environment provides the opportunity for enhancing the application user's job performance. However, an awareness is necessary of the special computer considerations associated with the organization/environment pressures, the interface with the customer, and the user's own behavior. The features of the mechanized support should be carefully tailored to make the human/machine interface as effective as possible. Principles used in the functional and operational design of PREMIS—with the Service Rep as the user—emphasize simplicity, consistency, and clarity.

The field experience gathered over the last two years has shown that certain operational aspects of PREMIS can have a cumulative, very potent negative effect on user performance. As reported by the Reps, these include "flaky" availability, slow response time, and a cumbersome log-on procedure. Also, the Reps report that the time to complete the functions and the level of task difficulty were not decreased by the introduction of PREMIS. Balancing this, the Reps feel that the system has allowed them, as well as the rest of the workers, to serve telephone customers more effectively and efficiently with higher confidence and less errors. The general positive influence of PREMIS—as seen by the

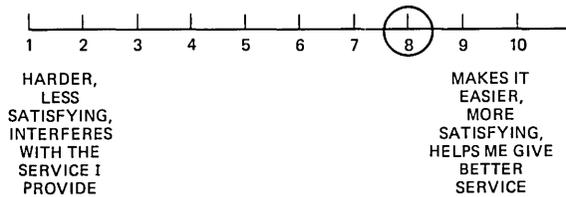


Fig. 4—Service Reps response to “How has PREMIS affected your job?”

Service Reps—was apparent in the final question of their most recent field review questionnaire, “How has PREMIS affected your job?” with the average score circled (see Fig. 4).

In my judgment, this positive reaction is in large part because of a systematic influence in the system design decisions of conscious human performance engineering principles. In addition, a total system engineering view must be taken of all design features so that the user benefits are not outweighed by hardware/software costs.

The time is past where large-scale computer systems can be initially delivered, found to be unacceptable from a user point of view, and then virtually re-done. The applied technology of human performance engineering has reached the stage in maturity and know-how to influence the a priori system design and help achieve a successful initial product.

## VII. CONCLUSION: A SYSTEM DESIGN PERFORMANCE AID

The following techniques work when designing computer systems that are easy for people to use:

(i) Have HPE people perform an early task analysis. This can substantially avoid unworkable or unreasonable user performance requirements. Divide tasks between people and machines, making the best use of the strengths of each; for example, allocate simple, repetitive jobs to the computer and complex, judgmental jobs to people.

(ii) Develop work flows early to ensure that the work will get done in a logical, workable sequence.

(iii) Design each system in an incremental fashion—both from a software-architecture (internals) and user-transaction (externals) point of view.

(iv) Involve the existing target organization in the system design process to help determine appropriate work shifts to achieve optimum use of the mechanized assist features. If certain work groups had added tasks, make sure they will get credit.

(v) Understand the impact of the computer system on the existing work performance measurements of individuals and groups to avoid behavioral resistance.

(vi) Design the human/computer interface and transaction command structure to be consistent and simple—simple to use and simple to learn.

(vii) Allow the user to initiate and control all human/computer interactions.

(viii) For a computer system without local intelligence, use a batch-prompted (with mask) input mode.

(ix) Limit inputs and outputs to a single display frame (page).

(x) Design output screens to allow over-typing of input commands to request the next transaction.

(xi) Return meaningful messages in English to the user for all possible transaction outcomes.

(xii) Incorporate computer-assist features (menus, prompts, defaults, minimal input) but only after task analysis is done to ensure that these features do not induce unnecessary transactions causing computer overload, as well as forcing users to take longer to complete their tasks. Perform iterative analysis of alternate designs to establish a quantitative computer/human performance balance.

(xiii) Make system-to-system switching from the same terminal an easy operation.

(xiv) Make terminal function keys and dialogue pnenomics consistent among systems.

(xv) Incorporate ease of use principles in information display design (data displayed in actual work sequence, data preservation across outputs, labeling, etc.).

(xvi) Make all user messages constructive and supportive, not condemning and confrontive.

(xvii) Establish quantitative system performance objectives and computer costs for

- a. Transaction Response Time by transaction class, processing type, and outcome
- b. Total System Availability number of occurrences, as well as total time
- c. Transaction Failure Rate by transaction class.

(xviii) Establish quantitative human performance objectives and costs for

- a. Human error rate by transaction class
- b. Training time
- c. Work time.

(xix) **SELL THE SYSTEM.** Use any means—user group meetings, presentations, deliverable documents, informal visits—to convince the target population, top down and bottom up, that the addition of the computer to the work environment is positive in terms of job effectiveness and user satisfaction.

## VIII. ACKNOWLEDGMENTS

Several colleagues supplied many useful suggestions. In particular, D. F. Lee, L. Bernstein, S. R. Fagan, and M. W. Soth, by their reviews, provided valuable guidance in the organization of this paper; P. R. Hawkins and M. P. Tarka helped with data analysis; and H. Dravis was involved in most of the editing and revision work. I thank them all.

## REFERENCES

1. M. J. Heffler, private communication.
2. B. Shneiderman, "Human Factors Experiments in Designing Interactive Systems," *Computer*, December, 1979, pp. 9-18.
3. B. Shneiderman, "Only Non-Human" *Datamation*, Forum (May, 1980) p. 295.
4. T. Gilb and Gerald M. Weinberg, *Humanized Input: Techniques for Reliable Keyed Input*, Englewood Cliffs, N.J.: Winthrop Publishing, 1977, pp. 629-33.
5. R. F. Jones, "Four Principles of Mass-Computer Dialogue," *IEEE Trans. Professional Commun. PC-21*, No. 4 (December, 1978), pp. 154-9.
6. L. H. Miller, "A Study in Man-Machine Interaction," *National Computer Conference*, 46, AFIPS Press, 1977 p. 409-21.
7. J. R. Hackman and G. R. Oldham, "Development of the Job Diagnostic Survey," *J. Appl. Psych.* (April, 1975), pp. 159-80.
8. J. L. Bennett, "The User Interface in Interactive Systems," *Annual Review of Information Science and Technology*, 7, ed., C. Cuadra Washington, D. C.: Amer. Soc. Inform. Sci., 1972, pp. 159-96.
9. J. Martin, *Design of Man-Computer Dialogues*, Englewood Cliffs, N.J.: Prentice Hall, 1973.
10. B. L. Hicks, private communication.
11. N. L. Ferrer, T. G. Hicks, and N. E. Lyness, private communication.
12. M. J. Rothkind, private communication.
13. F. Gebhardt and I. Stullmacher, "Design Criteria for Documentation Retrieval Language," *J. Amer. Soc. Inform. Sci.* 29, No. 4 (July, 1978), pp. 191-9.
14. W. J. Hansen, "User Engineering Principles For Interactive Systems," *Proc. Fall Joint Computer Conf.*, 39, AFIPS Press, 1971, p. 528.
15. P. R. Hawkins, unpublished work.
16. M. Tarka, private communication.
17. B. Das, "Consequences of Production Feedback and Standards in a Human-Machine Task," *Proc. Human Factors Soc.*, 1980, 24th Annual Meeting, pp. 629-33.
18. AT&T, BISCUS/FACS Review, 1978. Dallas, Texas, Section on System Administration.
19. M. S. Schoeffler, private communication.
20. J. D. Williams, private communication.
21. R. E. Barber, Experimental Results from 1979 Ph.D. Dissertation, N.Y.U.
22. K. Butler, private communication.
23. I. S. Yavelberg, private communication.
24. H. M. Parsons, "What Happened at Hawthorne?" *Service Magazine*, March, 1974.
25. J. Hafelfinger, private communication.
26. HOMAC Committee, "PREMIS Work Flows," AT&T Residence Department, Basking Ridge, New Jersey, 1980.
27. M. W. Soth, private communication.
28. Mowbray, private communication.



## Graphic Displays of Combined Presentations of Acoustic and Articulatory Information

By J. E. MILLER and O. FUJIMURA

(Manuscript received August 17, 1981)

*Articulatory data have been collected by a computer-controlled microbeam system. The locations of lead pellets placed on a speaker's articulators are recorded as  $x, y$  coordinates along with the acoustic signal and a train of timing pulses which synchronize the frames of X-ray data with the acoustic signal. Some 830 utterances have been recorded to date, resulting in a database of nearly 120 million bytes. We describe the graphic techniques by which those data are examined; namely, a simultaneous display of articulatory features such as velum lowering, lip and jaw closure, and tongue motion, together with the spectral information of the corresponding speech. These displays are also annotated with phonetic transcriptions obtained automatically by pattern-matching techniques. Acoustic representations of speech signals, in a simplified spectrographic format, are contrasted with articulatory measures, and implications are discussed concerning the difficulty of the automatic recognition of speech via conventional input processing.*

### I. INTRODUCTION

Articulatory data specified by  $x, y$  coordinate positions of lead pellets located on the velum, tongue, lip and jaw, all in the midsagittal plane, have been collected by a computer-controlled X-ray microbeam system. The corresponding speech signal and a sequence of timing pulses which synchronize the frames of articulation data with the acoustic data are also recorded. This system, which operates at the University of Tokyo, has been described in detail by Fujimura, Kiritani, and Ishida<sup>1</sup> and Kiritani, Itoh, and Fujimura.<sup>2</sup> In a typical data collection session an utterance lasts about six seconds, and approximately 700 frames of coordinate data on eight pellets are obtained. If fewer pellets are used, the frame rate will be higher since pellet positions are

determined sequentially for each frame. The waveform of the acoustic signal is quantized into 12-bit samples at a 10-kHz sampling rate, and each frame of pellet data is marked by an integer pointing to the corresponding acoustic waveform sample. All quantities, i.e., pellet coordinates, samples of the acoustic wave, and frame pointers, are stored as 16-bit quantities, and thus a 6-second utterance requires approximately 144,000 bytes. To date there have been some 830 utterances in two languages—American English and Japanese by a total of six different speakers (one female). The resulting database totals nearly 120 million bytes.

## II. ON-LINE CRT DISPLAYS

It is important in dealing with a large database that it be possible to retrieve and examine the data easily. The initial methods employed have been of an on-line, interactive nature. Utterances are identified by a 5-digit number, which indicates the data-collection session and position within the sequence of utterances recorded. Specification of this ID number at the computer console results in the retrieval from disk storage of all data pertinent to the utterance. Then a time window is selected under knob control for a CRT display of the time traces of the pellets, together with the RMS envelope of the speech wave. A copy of such a display is shown in Fig. 1. The ID number and the first and last frames of the window are specified on the top line. The text of the utterance is printed below, and average values of the pellet traces, together with their two-letter designators, such as BY for the *y* (up-down) coordinate of the pellet on the tongue blade, are listed on the right for up to six coordinates. The RMS envelope of the acoustic signal is shown at the bottom. Alternatively, an *x-y* coordinate map showing trajectories of movement in the pellet positions is displayed for the frames included in the time window. (See Fig. 2.) The window size has been cut down to the 45 frames spanning the portion of the text indicated by the bracket. The trajectory of each pellet begins at the point marked by an arrow and the final position is marked by an *X*. This type of display very clearly reveals the velum lowering for the nasalization, as well as the complex activity of the tongue associated with this portion of the sentence. The vertical dots represent the position of a cursor, which can be moved by a knob through the windowed frames. On-line digital-to-analogue facilities permit listening to the associated sound for time intervals of varying lengths. Generally, these display features have proved very useful in studying the data, but there is an inherent inadequacy in the representation of the speech signal by waveform envelopes. To combat this shortcoming, we have supplemented the output facilities with a spectrographic display of the acoustic information.

BEN ANNOUNCED THAT AN INNOCENT-SEEMING INFANT HAD NIMBLY NABBED MOST

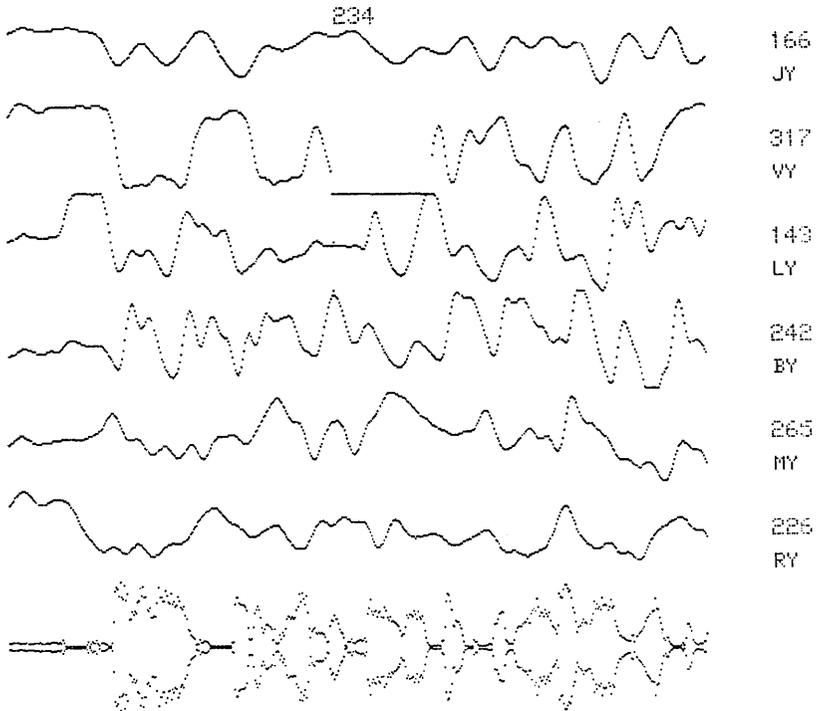


Fig. 1—On-line display of time traces.

### III. HARD COPY DIGITAL SPECTROGRAMS

To obtain the resolution necessary to generate an eight-level gray scale, we have employed a Versatec plotter having 200 rasters per inch. We define a super pel or picture element as a matrix of 4 by 4 rasters and let the number of black rasters out of the sixteen in each super pel be a nonlinearly increasing function of the gray level. Specifically, for levels 0 through 7 the number of black rasters is 0, 1, 2, 3, 4, 7, 10, 16. (See Fig. 3.) The original resolution of 200 rasters per inch is thus decreased to 50 pels per inch, but the resulting gray scale proves fairly uniform and is quite sufficient to display signal intensity in the frequency bands of the digital spectrograms. Figure 4 illustrates an example that adopts the time-frequency proportions of the traditional spectrograms.<sup>3</sup>

These displays are produced by using a raised cosine on a time window of 100 samples, or 10 milliseconds, and computing a spectral slice by fast Fourier transform techniques every 40 samples, or 4

BEN ANNOUNCED THAT AN INNOCENT-SEEMING INFANT HAD NIMBLY NABBED MOST

163

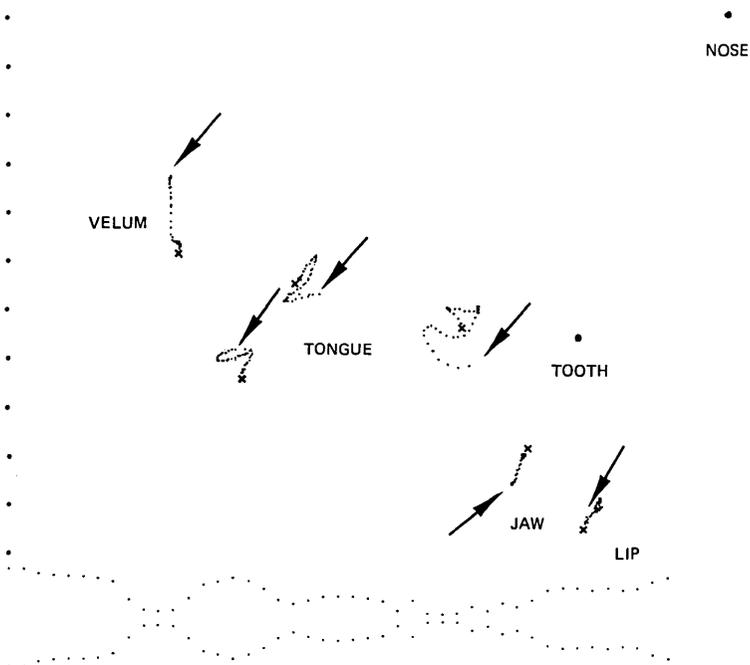
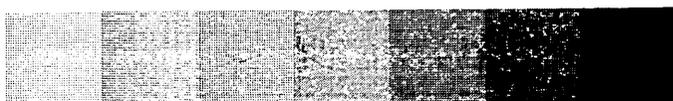


Fig. 2—On-line display of x-y trajectories (lateral view).



LEVEL:	0	1	2	3	4	5	6	7
BITS:	0	1	2	3	4	7	10	16
	0000	0000	1000	0001	0010	1001	0110	1111
	0000	1000	0000	0000	1000	0110	1011	1111
	0000	0000	0001	1000	0001	0010	1101	1111
	0000	0000	0000	0010	0100	1001	0110	1111

GRAY-LEVEL RESOLUTION — 50 PELS PER INCH.

ONE PEL = 4 x 4 RASTERS.

Fig. 3—Gray scale.

milliseconds. Each slice is “boosted” by a linear function to raise the high frequencies, and all amplitudes above an adjustable threshold are linearly mapped onto the eight gray levels. Having a resolution of 50 pels per inch, a sampling rate of 10 kHz and taking slices every 40

samples results in a time scale of 0.2 second per inch. In Fig. 4, one inch of height (on the actual computer output) represents 1000 Hz of frequency. It has been found, however, that great economies in paper space can be achieved by shrinking the frequency (vertical) scale, while leaving the time (horizontal) scale unchanged, and this deviation from the standard visible speech format does not adversely affect the usefulness of the spectrogram. (See Fig. 5 in which there are four 2-second lines of spectral data placed on one page as the result of such frequency scaling.) We find that if we cut the height down to as little as 0.8 inch, we are then able to display a speech spectrogram, together with as many as twelve time functions of pellet coordinates on a hard copy output. (See Fig. 6.) Such a display is substantially more useful than the RMS envelop (as in Fig. 1) for identifying phonetic events in the utterance being examined.

A display of this type can handle at maximum a time window of two seconds, which, at typical frame rates of 120 per second, is approximately 240 frames. The time scale is marked off both by frames (every tenth) and by milliseconds (every 200). Although computed while running the display program on-line (at about 100 times real time with our present CPU), this output is generated off-line on the Versatec plotter.

#### IV. OFF-LINE OVERVIEW DISPLAYS

It is frequently of considerable convenience to both prepare and examine the data away from the computer, and in so doing, it is advantageous to have a complete representation of as much as an entire utterance on the same sheet of output. The example shown in Fig. 7 meets these requirements. As an off-line or "background job," we are able to prepare on the Versatec plotter an 8-1/2 by 11 inch

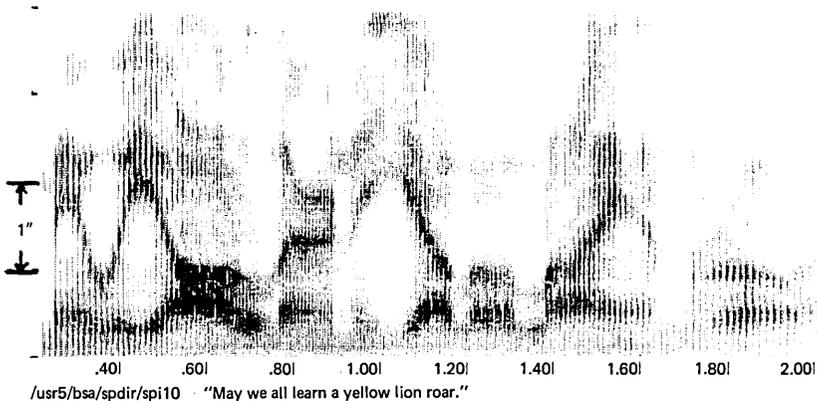


Fig. 4—Digital spectrogram in traditional format.

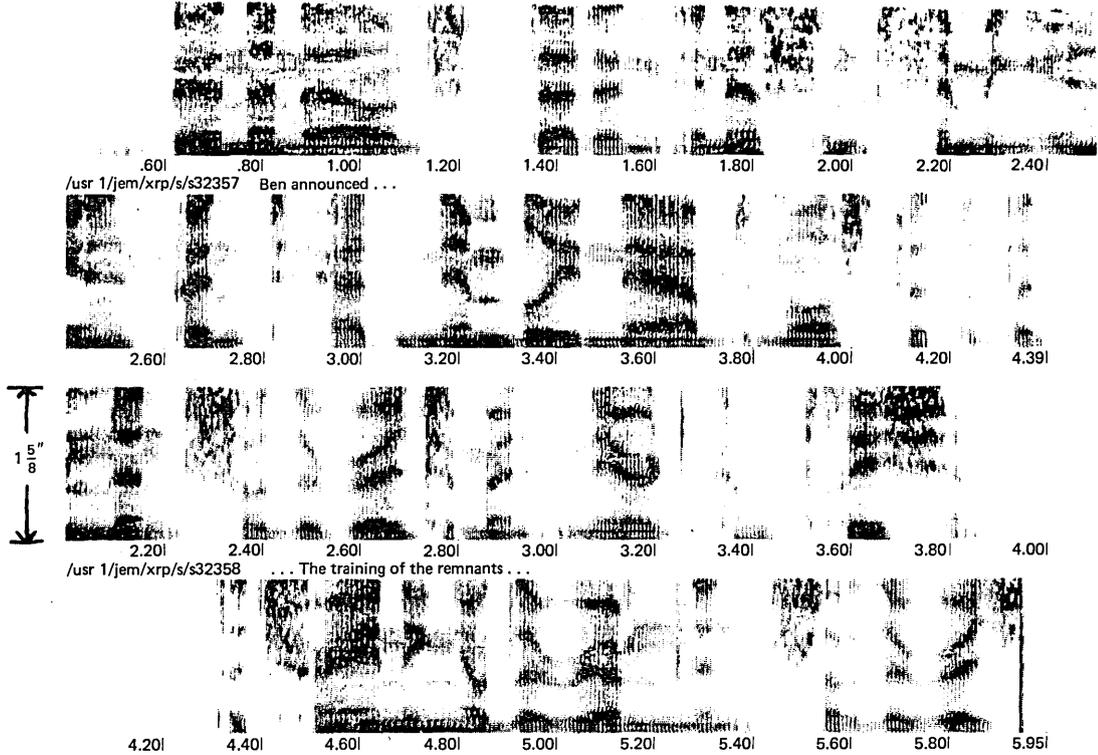


Fig. 5—Digital spectrograms with reduced frequency scale.

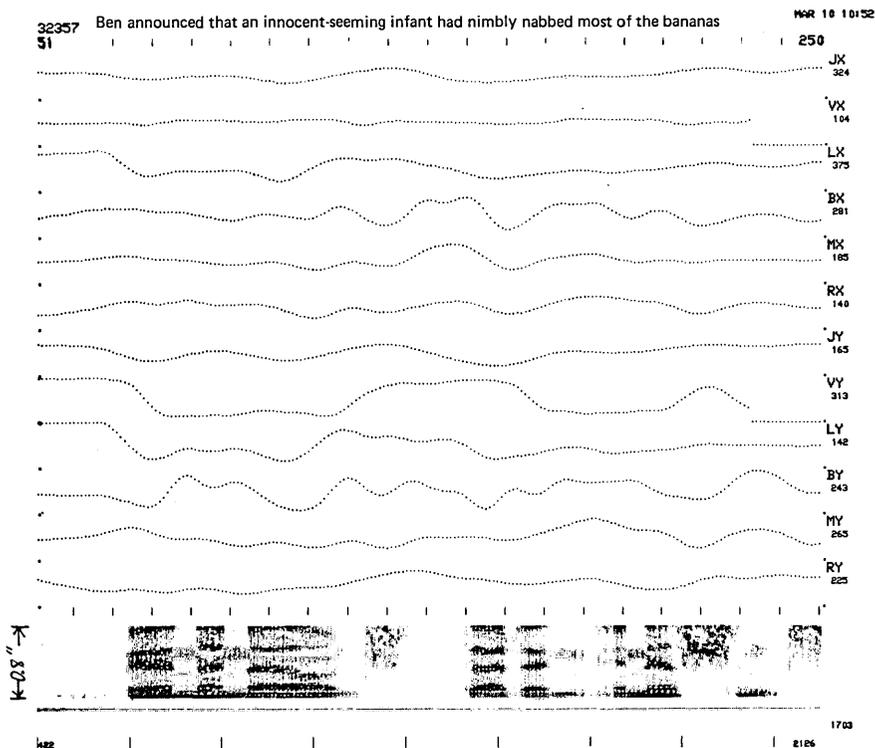


Fig. 6—Time traces and digital spectrograms.

page divided into three horizontal sections, each of which spans two seconds, plus 0.1 second of overlap into the next section. The sections contain ten stripes representing articulatory time functions coded in the eight-level gray scale and plotted in time synchrony with a spectrographic display of the acoustic data. Timing marks at 200-millisecond intervals are also indicated.

The ten gray-level stripes which describe the articulatory events are arranged as follows: There is a top set of four stripes indicating front-back or horizontal motion for, from the top, the lower lip and three tongue pellets (blade, mid and posterior position on the tongue surface), and a lower set of six which portray vertical motion for lower lip, mandible, three tongue pellets, and the velum. The correspondence between stripes and articulators may be determined by associating the stripes ordered from top to bottom on the page with the articulators ordered from front to back in the speaker's head. The range of coordinate values for each pellet is linearly mapped into eight levels of gray such that the minimum is the first level or white, the average value is level five or medium gray, and large values are level eight or

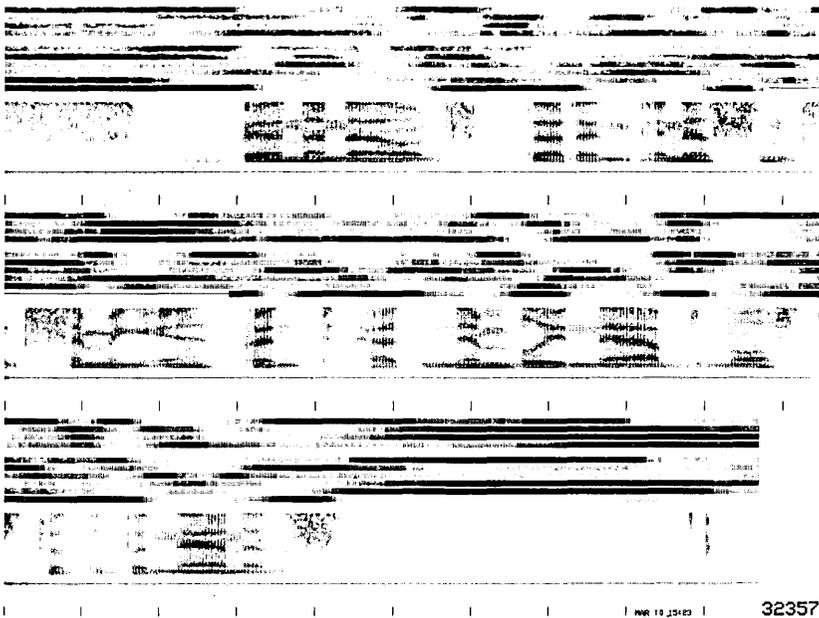


Fig. 7—Off-line display of articulatory and acoustic data.

black. That is, in Fig. 7 black corresponds to up for vertical motion and front for horizontal motion.

## V. FEATURE-ENHANCED DISPLAYS

As an extension of the display of the type shown in Fig. 7, which serves as a record and check on the database, we produce others of the type shown in Fig. 8, designed to emphasize articulatory features and contrast articulatory and acoustic events. In these, we reverse the mapping onto the gray scale for vertical motion of both the velum and jaw pellet and horizontal motion of the lip. As a result of these reversals, for example, the nasalization produced by a lowered velum, stress correlated with the minima in the jaw position, and the consonantal gesture indicated by lip constriction and a raised or advanced position of the tongue are easily spotted as black regions in the stripes (see below). It is an option of the program to produce the output with the gray scale for the stripes used in reverse so that when the displays are reproduced as transparencies the stripes can be individually colored. Thus, the features can be readily identified by a color coding.

It will be immediately noticed that the spectrographic representation of the acoustic information has been substantially modified. It has a checker-board appearance due to the frequency range having been

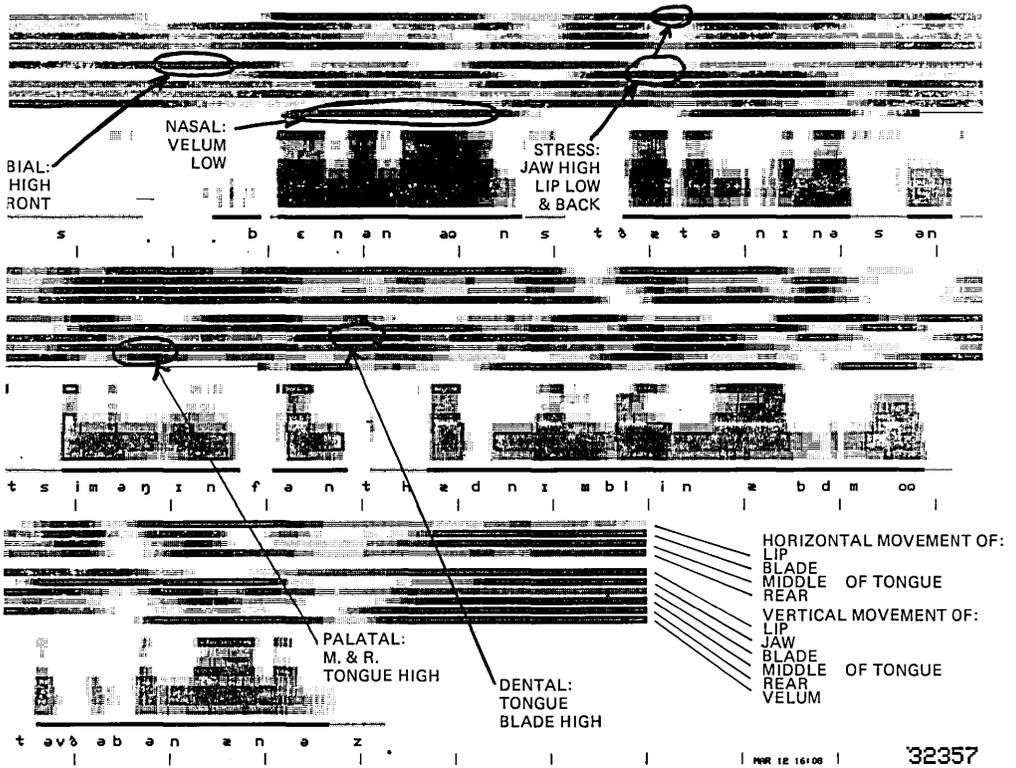


Fig. 8—Feature-enhanced display of articulatory and acoustic data.

grouped into eight bands. This output was produced by calculating a Fourier transform every two milliseconds and taking the maximum amplitude values obtained from three adjacent spectral slices. The maxima are used as an estimate of the spectrum at time points spaced every four milliseconds. The time window for the Fourier transform is five milliseconds; hence, the frequency resolution is 200 Hz, and the first twenty frequency components span a usable frequency range of 4000 Hz. These components are grouped into eight bands centered around 200, 400, 600, 900, 1300, 1800, 2500, and 3500 Hz, respectively. The average output magnitude in each of these bands is then mapped onto the gray scale. It will be noted that with such a grouping, the first formant and the nasal formant will be found in bands 1 to 4, the second in 4 to 6, and the third in 6 and 7. Voiceless fricatives and stops show up often in bands 7 and 8.

The black, gray, and white stripe plotted below the spectral display indicates the voiced, unvoiced, and silence (VUS) coding of the acoustic signal as determined by the Atal-Rabiner algorithm.<sup>4</sup>

Finally, a processing due to Nelson<sup>5</sup> utilizes vus coding and information on jaw minima to locate phrase boundaries, and then identifies occurrences of specific articulatory events that are predicted from a feature interpretation of a sequence of phonetic symbols obtained from a manual transcription of the utterance material. Each symbol is positioned automatically, according to these pattern identifications, and the resulting symbol alignment is shown in Fig. 8. A vus stripe and phonetic annotations will also appear on displays of the type shown in Fig. 6 and 7, provided the utterance has been processed with the coding algorithm and the Nelson pattern-matching scheme.

The display of the acoustic data in Fig. 8 represents a reduction by a factor of five over the traditional spectrogram on information to be stored; that is, from 40 harmonics to eight bands per spectral slice (albeit at the expense of more computation for the particular methods we use). But more important is its very "block-like" appearance, which in contrast to the smoothly fluctuating "grayness" of the articulation stripes, reveals apparent "segmentation" of the time events. These blocks make it easier to evaluate the success of the phonetic alignment. Also, since the frequency bands are selected considering the human auditory characteristics, the spectral representation may be similar to what the human perceives.

The marginal notations in Fig. 8 point out examples of the feature enhancement mentioned above. The utterance in this case is "Ben announced that an innocent seeming infant had nimbly nabbed most of the bananas." The nasalization occurring in "Ben announced" is indicated by the black on the bottom of the pellet stripe set. There are numerous other occurrences of nasalization in the sentence also marked by black in the bottom stripe, except where pellet tracking failed during the recording session. The missing pellet data are indicated by the thin line in an interval of approximately 600 milliseconds beginning about two seconds into the utterance (end of first section—beginning of second). Stressed words are revealed by black in the jaw stripe—second from top in a group of six. The feature of stress is also accompanied by a lowering of the lip which is indicated by white in the top stripe of the six, unless the vowel shows lip constriction. Hence, in most cases the white-black combination of these two stripes serves as a very clear cue for the stressed sounds.

Since the horizontal motion of the lip is plotted in the top stripe of the set of four, with black indicating retraction of the lip in this type of display, and since vertical motion is plotted in the top stripe of the set of six, labials such as /b/ and /m/ have black only in the set of six. However, the fricative /v/ in which the lower lip is retracted would be identified by black in both stripes. Color coding in which the same color is used for each articulator in both horizontal and vertical dimensions is helpful in directly identifying such articulatory features.

Apical consonants, namely, /t/, /d/, and /n/, are produced with the tongue tip and blade high behind the teeth and, hence, are revealed by black in the stripe third from top in the set of six.

The palatal sounds produced with a high mid-position of the tongue are similarly marked by black in the fourth stripe. All these articulatory characteristics are utilized in Nelson's automatic annotation.

We can with these displays go beyond the mere identification and location of these articulatory events and observe interesting time relations between the movements of the articulators and the resulting temporal patterns of the acoustic output. For example, we observe the very early onset of the lip closure for /b/ at the beginning of the sentence. Nasalization for the nasal consonants in the beginning phrase extends continuously over several syllables. We can also see how the labialization trails the vowel glides that cause the gesture as in "announced" and "most," where the lip constriction extends well beyond the bounds of the vocalic segment. Palatalization can be seen to extend through all the front vowels in the words "seeming infant," unaffected by the intervening consonants. We note finally the state of the articulators at the conclusion of the sentence and contrast the slow but still changing activity during this silence with the fast, yet smooth, movement during the sound production.

## VI. CONCLUDING REMARKS

In conclusion, we point out that the gray scale facilities have made it possible to reveal a great deal of information on a single page of output and to be freed in many cases from the necessity of on-line use of the computer as a means of data inspection. Furthermore, adequate use of the gray scale has made it possible to detect important features in the events and interesting relationships between the acoustic and articulatory events. Such spectrographic information as we observe in Fig. 8 is typical of the type of input used by an automatic speech recognition system; although, the coarse sampling used by most systems often eliminates the sharp discontinuities we see here. In this connection, it is interesting to note that the acoustic information as shown here is in many ways complementary to the articulatory information shown in comparison. Individual consonantal gestures, particularly in terms of place distinction (e.g., /p/ as opposed to /t/ or /k/) are obvious by articulatory measures, but often very difficult to identify via acoustic signals. We need more complex details, such as continuous formant transition patterns seen in Fig. 7, or some ad hoc and context-sensitive feature detection strategies based on such details, for acoustic identification of phonemic characteristics. While the articulatory gestures are sluggish and asynchronous, their phonemic correlates are, for the pertinent dimension, reproducible and often nearly invariant.<sup>6</sup>

After having identified the real invariant characteristics in the articulatory aspects that are only indirectly observable in the acoustic signals, we can ask proper questions as to how we might process such acoustic signals most effectively to derive the necessary phonetic information. To the extent we are impressed by an acoustic representation as in Fig. 8, we can identify the phonetic message by the present machines. To the extent we have to supplement with the other representations, such as direct articulatory or abstract pattern matching of more detailed acoustic information, effective speech recognition schemes remain to be devised.

## REFERENCES

1. O. Fujimura, S. Kiritani, and H. Ishida, "Computer Controlled Radiography for Observation of Movement of Articulatory and Other Human Organs," *Comp. Biol. Med.* 3 (1973) pp. 371-84.
2. S. Kiritani, K. Itoh, and O. Fujimura, "Tongue-pellet Tracking by a Computer Controlled X-ray Microbeam System," *J. Acoust. Soc. Am.*, 57, No. 6, Part II (June 1975), pp. 1516-20.
3. R. K. Potter, A. G. Kopp, and H. C. Green, *Visible Speech*, New York: Van Nostrand, 1947.
4. B. S. Atal and L. R. Rabiner, "A Pattern Recognition Approach to Voiced-Unvoiced-Silence Classification with Applications to Speech Recognition," *IEEE Trans. Acoustics, Speech, and Signal Processing*, *ASSP-24* (June, 1976) pp. 201-12.
5. W. L. Nelson, "Automatic Alignment of Phonetic Transcriptions of Continuous Speech Utterances with Corresponding Speech-Articulation Data," *Speech Communication Papers*, J. J. Wolf and D. H. Klatt, eds., New York: Acoust. Soc. Amer. (June 1979), pp. 63-6.
6. O. Fujimura, "Temporal Organization of Articulatory Movements as a Multidimensional Phrasal Structure," *Phonetica*, 38, Nos. 1-3 (1981), pp. 66-83.

## **1A Voice Storage System:**

# **Voice Storage in the Network—Perspective and History**

By E. NUSSBAUM

(Manuscript received November 17, 1981)

In mid-1976, Bell Laboratories undertook development of the then radical concept of introducing new customer services via a voice-storage capability in the network. With four systems installed and ready for service, the project was terminated in October, 1981, as a result of Federal Communications Commission (FCC) actions stemming from the Computer Inquiry II decisions declaring voice storage to be an enhanced service that could not be offered by the regulated network.

The following five papers, describing the services, architecture, and technology of the 1A Voice Storage System (vss) implementation of this Custom Calling Services (ccs) II offering were written two years ago and held until the regulatory outcome was settled. This brief introduction is intended to provide some background and perspective on the intervening time period.

The basic design concepts behind the ccs II offering and its associated 1A vss serving vehicle can be categorized into four major elements:

(i) Provision of a new class of flexible stored program controlled customer services involving storage, for later delivery, of the customer's voice messages under either called party control (Call Answering services) or calling party control (Advance Calling services)—to be known collectively as Custom Calling Services II. The design of ccs II included careful attention to interaction with previously existing ESS-based services (Custom Calling Services I) and to software flexibility of feature definitions, as has historically been the case in other stored program controlled systems.

(ii) For widespread availability and easy deployment, and for minimum costs, a system configuration consisting of shared voice storage nodes, each subtending multiple local ESS offices via dedicated trunk groups to provide access for customers. Appropriate traffic engineering allows for rapid changes to meet changing demand needs and forecasts.

(iii) Extensive use of digital technology to assure optimum opportunity to ride the silicon learning curve, including disk storage for digitally encoded speech, stored program control with distributed peripheral microprocessors, and the use of coder/decoders (CODECS), digital buffering, and custom large-scale integration (LSI) at the interfaces.

(iv) Emphasis on overall integrated network operation between the ESS and 1A vss entities, including a formalized signaling plan, joint operation on billing functions and service orders, and the use of a common operations support system network.

Based on these concepts, the 1A vss was designed and programmed for providing CCS II as described in the five papers that follow. The first system was installed in Philadelphia, Pennsylvania, and the first calls were processed there in early 1979. Initial experiments indicated that the human factors aspects of the design (customer interaction and perception of the services) were generally good and required only very minor changes. However, the reliability of the service, in terms of lost calls and integrity of the long-term voice message database, was below expectations, as was the total throughput capability of the centralized vss. During 1979, required software changes were undertaken and retested and three additional systems were installed in New York, Dallas, and Chicago. In March, 1980, full-time (24 hours a day) "friendly user" service was established with the Philadelphia 1A vss, starting with 35 Bell of Pennsylvania employees using the service from their residence phones, and subsequently progressing to 150 employees. Similar friendly user service was also initiated at other sites with the largest activity taking place at the Dallas site where friendly users numbered 450 people by late 1980.

As a result of these successful tests, a tariff was filed for the Philadelphia offering with the Pennsylvania Public Utility Commission (PUC) in May, 1980, for planned July, 1980, service. Plans were made for filing tariffs for the other sites shortly thereafter. The tariff was not approved by the PUC because of a pending antitrust suit filed by the Associated Telephone Answering Exchanges, Inc. (ATAE) and related PUC proceedings. Viability of the service offering was further complicated by the uncertainty created by the May, 1980, FCC order under its Second Computer Inquiry findings that enhanced services encompassed the area of voice storage, which could therefore not be offered as part of the regulated telephone network.

By early December, 1980, the Pennsylvania PUC's Administrative Law Judge recommended rejection of the AT&E position, but service continued to be deferred because of FCC rulings. A late December, 1980, FCC order specifically reaffirmed the earlier definition of voice storage as an enhanced service, but provided AT&T the option of requesting waivers under certain circumstances. The Bell System filed such a petition for waiver for CCS II in March, 1981. During the intervening period, all four systems continued to provide friendly user services to Bell System employees, and considerable data on customer reaction and system performance continued to be recorded.

In October, 1981, the FCC rejected AT&T's petition for waiver to allow the offering of CCS II as part of a regulated network service. At that point, the offering was withdrawn and the project terminated. At the time of the withdrawal of the service, measured data at the four sites and on laboratory support monitoring equipment indicated that the system met or exceeded all of its original design requirements on customer machine interactions, maintainability, reliability, and throughput.

The following five papers document the architecture of the service and of the serving vehicle. The confluence of these customer needs with the rapid progress in voice-storage technology will undoubtedly result in numerous future applications.



## **1A Voice Storage System:**

### **Prologue**

By G. D. BERGLAND, W. T. HARTWELL, and G. W. SMITH, JR.

(Manuscript received July 31, 1979)

#### **I. INTRODUCTION**

##### **1.1 Motivation**

In modern times, distant voice communication in the United States has been predominantly by telephone. There are some applications for which voice is recorded, transmitted (perhaps by mail), and replayed, but this form of "taped letter" is the exception rather than the rule.

A constraining factor when communicating by telephone is that both the calling and called parties must simultaneously be available. The probability of a successful call completion is approximately 0.75. Unfortunately, the successful completion is often to the desired line but not to the desired terminating party and, thus, additional calls are required to complete the communication.

In many cases, the required communication need be only one way. When this is true, it would be convenient to leave a voice message on the first calling attempt. Such a capability is being introduced with the 1A Voice Storage System (vss) and Custom Calling Services (ccs) II.

##### **1.2 Elements of voice communication**

Clearly, when people are together in the same place at the same time, they can simply talk. If they are available at the same time, but are in different places, they need some sort of transmission path between them. This path can be as exotic as a fiber optic cable carrying millions of conversations simultaneously in a digital bit stream or it can be as simple as a string between two tin cans.

If people are in the same place at different times, some storage media is required. For text, the medium could be paper on a bulletin board, a paper note, or a message scrawled on the washroom wall. For

storing voice communication, some sort of a sound recording mechanism is required.

The 1A vss brings together a unified communication system incorporating the required transmission paths and storage media to permit voice communication between individuals who do not coexist in either space or time.

### **1.3 History**

Early in the history of the telephone, telephone operators served as receivers and transmitters of messages for their customers. In many small towns, they simply jotted down notes, and periodically tried to deliver them whenever they had time available.

In all but the smallest exchanges, this quickly became an untenable situation. In self-defense, the operators quickly reverted back to simply making connections. The message storage function had to be provided in some other way.

Eventually, telephone answering bureaus sprang up, and these have been available for more than 50 years. Later, an attempt was made to mechanize the storage and playback of one-way voice messages through customer premises answer-and-record devices. These have been provided by the Bell System and several other companies since the early 1950s.

Initially, answer-and-record devices were quite expensive and their major use was limited to providing universal announcements which could be accessed by many people. They proved to be more economical when a single device was used to store the messages of many people than when they were used for individual customers.

The concept of a centralized vehicle to implement voice storage features had its origin as a Bell Laboratories research concept during the 1940s. On-going research into specific component technologies, control architectures, service definitions, and system implementations continued until the early 1970s when the required technologies had matured sufficiently to enable a cost-effective realization of a vss. During 1975, the development of a vss began, with the specific design for the 1A vss reaching completion during 1976. The first 1A vss was shipped to The Bell Telephone Company of Pennsylvania in August, 1978, with the expectation that it would be placed in service with CCS II features in 1980.

Many people have viewed the concept of storing and forwarding voice messages with almost as much enthusiasm as the concept of storing and forwarding text messages via so-called "electronic mail." Just as predictions were made in the 1930s that newspapers would soon be delivered by wire, many predictions have been made about the utility of delivering stored voice messages by wire. While the

technology for providing both of these services has existed for some time, the projected costs have always been prohibitive. With this thought in mind, the next section will describe some of the components which were required to build the 1A vss with the attributes discussed above.

## **II. THE 1A VSS IMPLEMENTATION**

### ***2.1 Requirements***

The 1A vss must meet a host of access, cost, maintenance, reliability, and capacity requirements. Access to the beginning of any message must be provided within seconds. The costs must compare favorably with those of customer premises equipment.

The storage system should be as reliable as the transmission path connecting to it. It must provide sufficient capacity to ensure service when needed. Typically, the total storage capacity requirements for a shared facility are substantially less than for per-customer facilities.

The system must be engineered, taking into account estimates of expected holding time, average storage time, probability of message left, and the fact that 25 percent of normal "plain old telephone service" (POTS) calls terminate in "busy" or "do not answer."

The cost of storage is a function of both message length and storage time. While acceptable costs may be achievable through the use of analog recording equipment, we have found that access, maintenance, reliability, and versatility considerations all tend to favor a digital approach.

### ***2.2 A new node in the stored program control network***

As with many new telecommunication services, the cost of a 1A vss must initially be spread over as wide a customer base as possible. This implies that a single 1A vss system should provide storage capabilities for several switching systems rather than providing the required logic and storage in each electronic switching system (ESS) office.

The connection between stored program control (SPC) switching offices and the 1A vss is represented schematically in Fig. 1. Note that the 1A vss system looks much like an independent office. It receives inputs over special voice access trunks, it has its own control processor, and it directs traffic in and out of a storage subsystem directly instead of through the processor to the storage subsystem. From a control point of view, the ESS systems have ready access to the line state of customers, whereas the 1A vss knows about the stored messages; therefore, control of the services requires close cooperation between the software in ESS and in 1A vss.

To make CSS II features simultaneously available to hundreds of thousands of ESS customers on the basis of casual (i.e., daily) activation

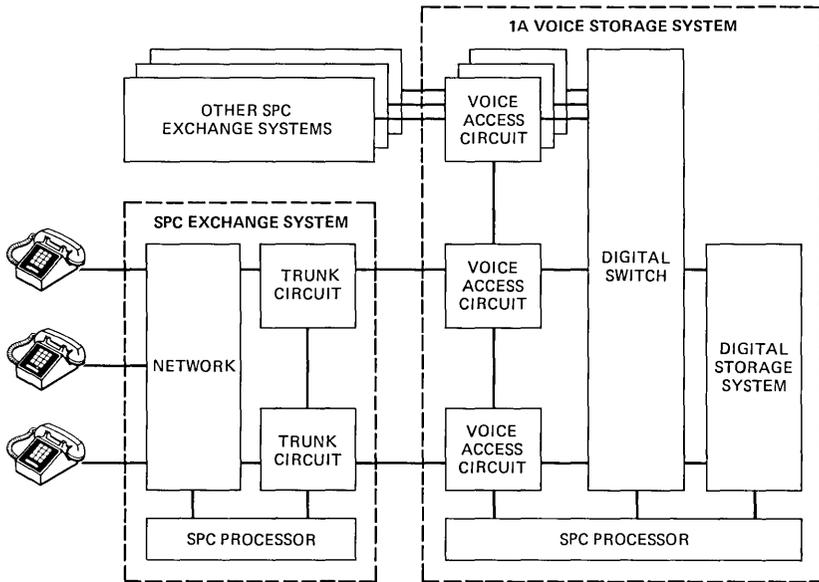


Fig. 1—The 1A vss in the spc network.

and deactivation of service, the use of dedicated, per-line equipment was quickly ruled out in favor of trunk-interfaced equipment. The requirement to enable any customer's rapid access to messages stored on his behalf similarly ruled out the use of sequential storage media, such as analog or digital tape. Therefore, it is apparent that the storage subsystem is a major cost item and as such deserves considerable attention. Even though memory costs continue to go down between 20 and 30 percent per year, the amount of storage required is still large enough to keep the memory subsystem costs significant. This problem is made worse when reliability requirements force the duplicate storage of some messages.

If an encoding rate between 32 and 64 kb is required, bubble and charge-coupled devices (ccds) become prohibitively expensive. Moving-head disks currently appear to provide the most viable solution from both economical and performance standpoints.

The advent of economical high-capacity disk storage has made the 1A vss economically feasible. The description of the implementation of the 1A vss is covered in the companion article "Architecture and Physical Design," by R. G. Cornell and J. V. Smith, in this issue of *The Bell System Technical Journal*.

### 2.3 Custom calling services II

With the availability of the 1A vss as a flexible node in the spc

network, numerous imaginative and useful services can be provided. The initial CCS II services were chosen after extensive study of customer requests and market surveys to determine utility and convenience. The CCS II services are described in detail in the companion article "New Custom Calling Services," by D. P. Worrall, also appearing in this issue.

#### **2.4 Call answering service**

Answer-and-record services provided centrally can be considerably enhanced over those provided solely via customer premises equipment. For example, the customer's phone can be answered on "busy," as well as on "no answer." During an office busy hour, the number of busys and no answers are approximately the same.

With the use of the remote access options, the 1A VSS provides the answer-and-record service to customers who are away from their home telephone.

#### **2.5 Advance calling services**

The CCS II entry into the voice store and forward market is called Advance Calling. Advance Calling allows the calling party to record a message to be delivered at a future time. The customer may specify a future time or have the 1A VSS deliver the message immediately. Messages may be sent to most telephones which can be dialed directly, even to an individual's phone as a reminder or a wake-up call.

### **III. SUMMARY**

Installation of the 1A VSS, as an extension of the transmission and switching facilities already provided by the telephone network, leads to a new dimension of utility and convenience for the telephone customer. For many people it will remove the frustration of trying to get through. For others, it will provide convenient access to information of special interest.

The technology is here today. The companion papers in this issue define the CCS II features, the implementation of the 1A VSS hardware, operational software, maintenance software, office engineering, and testing considerations.

The 1A VSS will provide a new dimension in voice communications which we have just begun to exploit.



## **1A Voice Storage System:**

# **New Custom Calling Services**

By D. P. WORRALL

(Manuscript received October 24, 1979)

*In creating new customer services, systems engineers traditionally conduct market needs analysis, engineering feasibility studies, and project economic analysis. Finally, they prepare development requirements. In addition, systems engineering designs the service from the customer's point of view. This article describes new custom calling services as the customer sees them and highlights some interesting features of the services.*

### **I. INTRODUCTION**

The introduction of No. 1 ESS in 1965 signaled the beginning of new telephone services to the residence and business customer. Some of these services, called Custom Calling Services (ccs), were made available to customers served by ESS central offices and allowed customers to make more efficient use of their telephone service. Custom Calling Services include Call Waiting, Call Forwarding, Three-way Calling, and Speed Calling.

Call Waiting, the most popular service, notifies customers of a second call when they are already on a call. The customer has the option of answering the second call and placing the first call on hold. The service has been very popular with customers who use the telephone frequently.

Another popular service is Call Forwarding, which allows customers to "transfer" all of their incoming telephone calls to another telephone number. When the service is activated, calls to the customer's number will automatically be forwarded to another number which the customer specified when the service was activated. Originally, calls could only be forwarded within the local dialing area of the customer, but cus-

customer reaction to that limitation soon caused a change so that calls can now be forwarded anywhere. The service is popular with small business customers who forward their calls between their business locations and their homes.

Three-way Calling, previously called Conference Calling, allows customers to add on a third party to an established telephone call and then talk with both parties at the same time in a full three-way conversation. Customers involved in community affairs use Three-way Calling to help arrange group activities.

Speed Calling, previously called Abbreviated Dialing, offers customers the capability to dial frequently called numbers with only one or two digits. Customers may choose an eight-number list and/or a 30-number list. Initially, customers could only change their list of numbers via a telephone company service order. This process was time consuming and expensive and detracted from the value of the service. The service has been upgraded to allow customers to change the contents of their Speed Calling lists directly from their own telephone. This service improvement has increased the value of Speed Calling to customers. Customers not only enter frequently called numbers into the list but also special numbers, such as fire, police, first aid, and poison control, for easier and faster access to emergency services.

At the end of 1978, CCS services were available to 15 million customers served by ESS central offices. Customers currently subscribe to over 4 million CCS services. Since 1965, there has been a rapid growth and wide distribution of ESS central offices, allowing marketing organizations to more effectively advertise and sell these services to our customers. The results of these efforts can easily be seen in Fig. 1.

But today's communications-oriented society requires still more advanced telecommunications services. Market studies have indicated customer interest in a wide variety of new and modern telephone related services. Some of this interest stems from the limitations of today's CCS. These services require the physical presence of the customer at a telephone to actually use or obtain value from the services. But in today's very active and mobile society, customers desire control over their communications services even when they are not at their telephones.

Several of these service needs, which are the subject of this paper, concern the ability of a customer:

(i) to receive a message from someone when the customer is not at home or is too busy to answer the telephone,

(ii) to send a message to someone when the customer cannot, or chooses not to, reach them directly,

(iii) to give information to callers, and

(iv) to control services when the customer is not at home. The

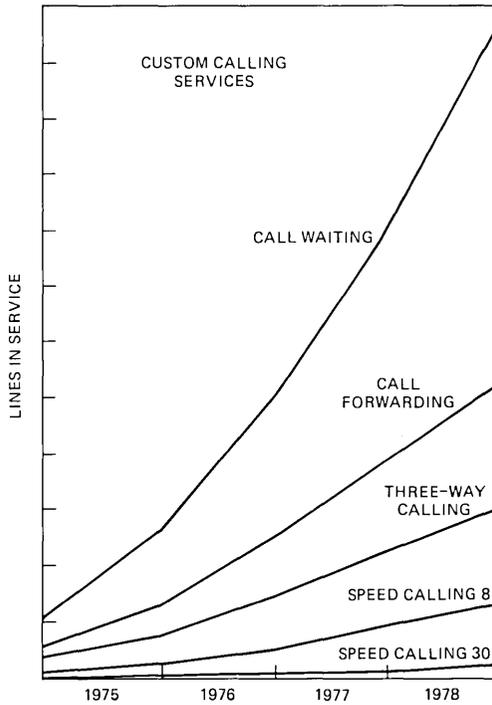


Fig. 1—Bell System sales.

evolution of these “customer needs” into actual telephone services has resulted in the addition of Call Answering, Advance Calling, Custom Announcement, and Remote Access services to the existing Custom Calling Services.

Call Answering provides telephone answering and message recording service for both residence and small business customers. The service offers the general capability to answer a call, deliver to the caller a customer-recorded greeting, and then record a message from the caller. Messages are accumulated and delivered to the customer upon request. The service is under control of the customer, may be used from rotary dial or dual-tone multifrequency (DTMF) signaling telephones, and requires no special equipment on the customer’s premises.

Advance Calling service adds a new dimension to telephone service. Advance Calling offers customers the capability to record a message and have it sent to a designated telephone number at a future time. It is for use by customers when direct communications is not desirable or not possible because lines are busy or the call spans large time-zone differences.

Custom Announcement Service allows customers to record an informational announcement, which is delivered to anyone who calls the

customer's telephone number. It is for use by small businesses, schools, social groups, and churches as an information distribution service.

Remote Access will allow customers to access and control these new services from another telephone when they are not at the telephone with the service.

The chart below summarizes the original and new CCS and divides them into terminating and originating categories.

Each of the new CCS services will be described in more detail in the following sections.

<i>Terminating Services</i>	<i>Originating Services</i>
Call Waiting*	Three-way Calling*
Call Forwarding*	Speed Calling*
Call Answering†	Advance Calling†
Custom Announcement†	

## II. SERVICE DEVELOPMENT

Detailed service specification work began with an in-depth analysis of various market studies, which were done by AT&T to identify the demand for new services. Combining these market data with an understanding of the technology to provide these new services resulted in a general outline of a service, with many possible service options. Potential customers were interviewed individually and in groups to further understand customer needs and interests in the services and options.

When a form of a service already existed, studies were conducted to determine their basic capabilities, their strengths, and their weaknesses. Users of these services were questioned to determine their likes and dislikes about the services. Human factors studies were done to optimize the human interface to the proposed new services.

All of this information was used to generate service descriptions and detailed specifications for service operations. The services were developed to offer infrequent users the basic capabilities. Also the services contained enough flexibility and options to satisfy a wide variety of needs for frequent users. Many unique service capabilities are incorporated which are not available from existing alternatives.

The following sections describe in detail the new service offerings. Since the development of new services is an ever-evolving process, the actual services designed may differ slightly from the following descriptions. Customer reactions to the services will be evaluated and changes and enhancements will be designed to meet customer needs.

---

\* Original Custom Calling Services, now called CCS I.

† New Custom Calling Services, now called CCS II.

### III. CALL ANSWERING SERVICE

Call Answering (CA) service is a sophisticated telephone answering and message storage service which provides a variety of service capabilities and options to meet the needs of different users. The service answers calls, delivers a customer-recorded greeting to the caller, records a message from the caller, and then stores the message for later retrieval by the customer. Initially, two forms of CA service will be offered. First, a Daily Call Answering (DCA) service is designed for customers with an infrequent or occasional need for an answering service (e.g., night out, weekends, vacations, etc.). This service is simple to access and control and may be activated at any time from the customer's telephone *without* prior arrangement with the Bell operating company (BOC). A maximum number of messages stored is established for DCA customers. The service is available to customers with rotary dial or DTMF signaling. The customer will be charged for the duration of service activation, plus a charge for each call received.

The second CA service, Monthly Call Answering (MCA) has been designed for the user who has a regular need for CA service. It provides greater message storage capacity and message handling capability than the DCA service. A customer desiring MCA service must subscribe to the service by contacting the BOC Business Office and will be charged a monthly fee, plus a usage charge based on the number of calls received.

The CA service can be regarded as having four phases: service activation; call answering and message recording; message retrieval; and service deactivation. Before describing each phase, several terms needed to be defined. A *greeting* is the recording which is delivered to the calling party by the CA service. The greeting may be either a standard BOC-provided message or one which has been personally recorded by the customer. A *message* is a recording made by a caller (in response to a customer greeting), which is stored for later retrieval by the customer. *Prompting announcements* or *prompts* are a special set of CA service announcements that will assist a customer in using the service.

### IV. CALL ANSWERING—SERVICE ACTIVATION

To activate DCA service, a customer dials the service activation code (Table I) and a prompt asks the customer to select either the standard or personal greeting by dialing either 2 or 7. A customer who chooses the personal greeting is instructed to start recording after the record tone. If the customer stays on the line after finishing the recording, the personal greeting is replayed for the customer's approval.

To activate MCA service, a customer dials the service activation code.

Table I—Custom Calling Services II Access Codes

	Code <sup>†</sup>
Call Answering	
Activation	1151
Message retrieval	1152
Deactivation	1153
Monitor	1154
Advance Calling	
Message recording	1141
Status check	1145
Custom Announcement	
Activation	1158
Deactivation	1159
Remote Access	Seven-digit telephone number
Privacy Code Change	1161

<sup>†</sup> Customers with DTMF signalling telephones may use \* instead of the digits 11 (i.e., \*51 instead of 1151).

(Note, the service access codes are the same for DCA and MCA service.) The service identifies the customer as a monthly subscriber. The customer then receives prompting instructions to enter a service control code. An MCA customer may record a new greeting, replay an existing greeting, choose the standard greeting, or choose the number of rings before CA answers. An MCA customer may also hang up during this prompting announcement, in which case the CA service will be activated with the existing greeting that is on file for that customer.

For both DCA and MCA customers, CA service is activated when the customer hangs up. Customers may verify activation of the service by dialing their own telephone numbers and having the call answered by CA service. A customer wishing to change or modify a greeting may do so at any time by simply performing the activation procedure again.

#### V. CALL ANSWERING—CALL ANSWER AND MESSAGE RECORDING

Once CA service has been activated, the service begins to answer calls to the customer's telephone. The most typical call type to be answered by CA service is the "no-answer" call. If a DCA customer does not answer the call within three ringing cycles (approximately 18 seconds), the call will be switched from the customer's line and connected to CA service. For MCA, the customer selects the number of rings (from 0 to 7) during activation. If 0 rings is selected, the call is immediately answered by CA service, regardless of the busy/idle status of the customer's line.

When a call is answered by CA, a CA service Logo (a short series of tone signals uniquely identifying CA service) and the customer's greeting are delivered to the caller. The CA service automatically provides a record tone at the end of the greeting and switches to a recording

mode to record a message from the caller. A message is considered received if the service detects the presence of voice signals for a duration of at least one-half second. After the message is recorded, it is duplicated for reliability, and stored in that customer's message file for later retrieval. Voice signals shorter than one-half second are erased and not stored for the customer. This prevents customers from hearing disconnect tones and noises from callers who did not wish to leave a message.

A caller has ten seconds from the record tone to begin a message. Recording does not start until the service detects the presence of audio signals after the record tone. Thus, messages stored for a customer do not include the silent interval prior to the first voice signal. Also, the trailing silence at the end of a message is erased, so customers need not listen to silence during message retrieval. The service stops recording if 3 seconds of continuous silence is detected or if the maximum message length is reached. The service automatically disconnects the call at the end of message recording.

In addition to answering a call when the customer does not, CA service provides the unique capability to answer calls when the customer's line is busy. Furthermore, CA service has the capability to handle several calls at the same time. Typically, the basic service will answer only two simultaneous calls, but monthly customers have the option (at an additional charge) to specify up to 15 simultaneous calls to be answered by CA service. Additional calls, exceeding a customer's simultaneous calls limit, would receive normal busy tone.

Another unique aspect of CA service is that whenever CA service is answering a call and/or taking a message, the customer's line is free to handle other calls. The customer may answer other calls or place an outgoing call. The line is not tied up while a message is being recorded.

Because CA has been designed to supplement the existing Custom Calling Services, its interactions with Call Waiting (CW) and Call Forwarding (CF) have been carefully designed. If a customer has both CW and CA, during the activation phase of CA, the CW service is momentarily inhibited so that the CW tone sequence will not be recorded into the customer's greeting. During a period when CA is active, CW and CA interact to maximize the opportunity for the customer to receive a call. When the customer is dialing or the customer's phone is ringing, CW tone cannot be given so the call is answered by CA service. When the customer is busy on an established call and another call arrives, the normal CW tone is provided. The customer may flash the switchhook and answer the second call. If the customer does not answer the CW call, normal procedure for CW service is to provide a second CW tone 10 seconds after the initial CW tone. If CA service is active, the second CW tone is not provided, but at the time the tone

would have occurred, the call is answered by CA. This procedure gives the customer with CW and CA service a great deal of flexibility in handling incoming calls.

With the above arrangement for CW and CA service, a customer can “answer” at least four calls at the same time. The first call is answered directly, the second could be answered by CW, and the third and fourth calls could be answered by CA service.

A customer may have CF service, as well as CA service. In this case, both services *cannot* be active at the same time. One service must be deactivated before the other service can be activated. A call can be either forwarded to another telephone number or answered by CA service but not both.

After CA service answers a call, MCA customers may screen incoming calls by listening to the message which is being left. Customers dial the monitor access code (Table I), and CA service bridges the customer onto the call which has been answered. The bridge is a one-way bridge, allowing the customer to only listen to the message being left. If the customer desires, he may be connected directly to the caller by flashing the switchhook. The caller is switched off of CA service and is connected to the customer. Any message which may have been recorded at that time is saved and stored away with all other messages for that customer.

To protect customers and the BOC from abusive use of CA service, several constraints are placed on message recording capabilities. These constraints include limits on (i) the length of an individual message—the service stops recording when the maximum message length for that customer has been reached; (ii) the number of messages stored for a customer at any one time—when the limit is reached, the service automatically turns itself off until some of the messages are retrieved; and (iii) the amount of time a message may remain in storage—the service will erase any message exceeding this storage limit. Typically, the limits are 30-second messages, 30 messages in storage at one time, and seven days storage. Each of these limits may be specified for individual MCA customers.

Another way has been provided to establish the limits. The system has a capability which will provide customers with a fixed amount of storage. The service will answer calls and take messages of any length until the storage for that customer is full. The limit on message length or number of messages, or both, is thereby removed with this option.

## VI. CALL ANSWERING—MESSAGE RETRIEVAL

Whenever messages reside in a CA customer's message file, two message waiting signals are given to the customer to notify him of the existence of these messages. Whenever a customer goes off-hook to

place a call, the initial 2 seconds of dial tone is interrupted (i.e., turned on and off rapidly). A customer may dial during the interrupted dial tone. The second signal, a short burst (500 ms) of ringing is applied to the line whenever a customer disconnects from a completed call. The customer may, at his discretion, retrieve messages or ignore either signal.

The message retrieval procedure is basically the same for both DCA and MCA customers. The customer dials a retrieval code (Table I) and is connected to CA service. The CA service responds to the customer with a prompt stating "You have had  $M$  calls since you last played back your messages and you have  $N$  messages waiting," where  $M$  equals the number of calls answered by CA and  $N$  equals the number of messages for the customer. Each message recorded for that customer is now delivered in the same order in which the messages were received. A unique feature of CA service is that just prior to returning *each* message, CA service automatically tells the customer both the day of the week and the time of day when that message was recorded—for example, "Thursday—9:15 a.m." The message is then delivered to the customer. Messages are separated by a short tone and approximately 3 seconds of silence.

Daily CA customers have very basic message handling capabilities during retrieval. Messages for a DCA customer are delivered once in chronological order. After complete delivery of all messages, a service prompt indicates that if the customer continues to listen, the messages will be repeated two more times. During any of the message deliveries, the DCA customer may "skip" the delivery of a message by dialing the digit 4. The message is skipped and the next message in the sequence is started. Whenever an DCA customer hangs up during retrieval, all messages which were delivered completely at least once (or skipped) are automatically erased from the message file. All other messages are retained.

Monthly CA customers have complete message handling capabilities including *save*, which saves a message until the next message retrieval request; *repeat*, which immediately begins delivery of the current message again; *pause*, which causes a 5-second hold on message delivered; and *skip*, which causes the current message to be stopped and the next message in the sequence to begin. Successive dialing of *skip* or *repeat* causes MCA service to sequence ahead one message at a time or to sequence backward one message at a time.

Whenever an MCA customer hangs up from message retrieval, all messages delivered completely at least once (or skipped) and not specifically saved, are erased from the message file. All other messages are retained until the next retrieval request.

When messages are erased from storage, various information is

tabulated and stored for billing and service analysis purposes. This information includes length of message, duration of storage, date and time of message recording and message retrieval, number of messages, and the busy/idle status of the customer's line when call was answered. This information will be analyzed in detail to better characterize customer answering service needs.

#### **VII. CALL ANSWERING—SERVICE DEACTIVATION**

A CA customer may turn off or deactivate the service at any time. The procedures are the same for DCA and MCA customers, but the action taken by CA service is different. The procedure is for the customer to dial the deactivation code (Table I) and listen to a service prompt indicating that the service will be turned off. If one or more messages are still in storage, they will be delivered if the customer continues to listen. Whenever a DCA customer disconnects from the deactivation request, all of the customer's messages, the personal greeting, and the service active indicators are erased. Necessary billing information, such as duration of service activation and the number of calls, is collected and an accounting entry is made to bill that customer for the service.

Whenever an MCA customer disconnects from a deactivation request, only the CA service active indicator is removed. The customer's personal greeting and any messages in the customer's message file are retained. The MCA customer will still receive message waiting indicators until the message file is empty.

#### **VIII. CALL ANSWERING—SERVICE OPTIONS**

Several CA service options exist that will allow the BOC to customize the service parameters of CA service to fit the needs of individual customers or groups of customers. Each of the following service parameters may be adjusted to fit the needs of the market place: length of personal greeting, maximum length of messages, number of messages stored at one time, length of time messages remain in storage, and number of simultaneous calls answered for each customer. In addition, a Remote Access option can be applied to CA service. Remote Access will be described in more detail later.

#### **IX. ADVANCE CALLING SERVICE**

Advance Calling (AC) is a message-sending service available to all customers served by properly equipped ESS central offices. It offers customers the capability to send a recorded message to a designated telephone number at a future time. Considerable interaction between the customer and the service is required to allow customers to specify

the telephone number, to specify the time of delivery, and to actually record the message. Extensive studies were done to develop easy and logical input procedures for AC.

The basic service operation is for the casual or infrequent user of the service who may not remember the operating procedures. Special user prompting announcements guide and assist customers through the entire procedure. However, frequent users need not be burdened by instructional prompts. For instance, when the service is prompting for digits to be entered, either time or telephone number, the reception of any digit cancels the prompt and the service begins collecting digits. Thus, a customer may “dial thru” prompting announcements. In addition, customers with DTMF signaling may dial the digit to cancel prompting announcements preceding a message recording operation to speed up service operation.

Since there is a significant amount of information to be entered into AC service, the chances for dialing mistakes is higher than normal. The AC service allows certain errors to be corrected. For instance, if the service detects an invalid area code or an invalid time of day, the service asks the customer to re-enter the information. If the customer detects a mistake while dialing, customers with DTMF signaling may dial the digit \* to signal the service that an error has been made. The service then prompts the customer to enter the correct information. Furthermore, all information dialed by the customer is repeated to the customer for verification.

The AC service operates in three distinct phases. The first phase, called message recording, allows the customer to select the telephone number to which the message should be delivered, specify the time for delivery (optional), and record the message. The second phase, called delivery, is automatically performed by AC service. It places calls to the specified telephone number and upon answer, delivers the customer's message. The third phase, called delivery status check, allows the customer to determine if and when a message has been delivered. If a message has not yet been delivered, a customer may cancel any future attempts to deliver it. Each phase of AC service will be discussed in more detail.

## **X. MESSAGE RECORDING PHASE**

Any residence or small business customer may access AC service by dialing the service access code (Table I) on a DTMF signaling or rotary dial telephone. The customer is greeted with a prompting announcement indicating that AC service has been reached and the customer should dial the telephone number where the message should be delivered. The prompt states, “Please dial the telephone number to which

you want your message delivered. Dial the entire number exactly as if you were dialing it directly." The customer then enters the telephone number, and it is spoken back to the customer for verification. The verification response states, "The number to which you want your message delivered is (number entered)." If incorrect, the customer may reenter the telephone number by following error correction procedures.

The customer is then asked to choose between Customer Specified Delivery (CSD) or Service Specified Delivery (SSD) by entering the appropriate control digit. The announcement states, "If you wish to select the time you want your message delivered, please dial the number 9. If not, please dial 7." If CSD is selected, the service prompts the customer to enter the time of day (hours and minutes and a.m./p.m. indication) at which the first attempt to deliver the message is to be made. The service asks the customer to, "Please indicate in your local time when you want your message sent. First dial the hour and the minutes. Then dial A for a.m., or P for p.m." (Twenty-four hour time notation is not used by the service since tests show a very large error rate with that time convention.) The time entered by the customer is spoken back for verification. The announcement states, "Thank you. The time you want your message sent is (day of week) (time) (a.m., p.m., noon, or midnight)" for example, "Thursday—3:15 p.m." Customers may only enter delivery attempt time within 24 hours of the current time of day. If a customer does not wish to specify the time (i.e., chooses SSD), the first attempt to deliver the message is made 15 minutes after the message is entered, but no attempts are made during the late night hours.

The customer is then prompted to record the message. The service prompt states, "When you hear the tone please record your message. You will have (number) minutes of recording time. After recording your message, please dial 2 to approve it for delivery." The actual maximum length of a message is a service parameter to be adjusted as experience is gained with the service. Initially, a 1-minute message is planned for AC service. The message must exceed a 1/2-second minimum message length, and the recording will end if the maximum message length is reached or the customer remains silent for 3 seconds (whichever occurs first). If the customer dials 3, the message is then played back for verification. The service prompt states, "(tone) Thank you. Your message will now be played back for your approval. After hearing your message, you may approve it for delivery by dialing the number 2. If you want to re-record your message, dial the number 6 (tone)." The playback of the recorded message begins within 1 second. When the customer dials the number 2, the message is scheduled for delivery.

Whenever a customer makes a mistake detectable by the system, an appropriate prompting announcement informs the customer of the mistake made and voice-responds any information entered. For instance, if a customer entered only six digits of a telephone number, the service would respond with, "We're sorry. You have not dialed enough digits. The numbers you dialed are (numbers)."

The service duplicates the recorded message for reliability, schedules the delivery of that message at the appropriate time, and records necessary information for future billing purposes.

## XI. DELIVERY PHASE

If the customer specified the delivery time, (Customer Specified Delivery, CSD), the first attempt to deliver that message is made at that time. If the customer selected System Specified Delivery (SSD), the service will schedule the first attempt 15 minutes after the message is entered, unless the time falls into a nondelivery time period. Selected hours during the day may be specified as nondelivery periods for SSD messages. Typically included are the late night hours (10 p.m. to 7 a.m.) in the time zone where the message will be delivered. In addition, retry attempts for SSD messages will not be made during central office busy hours where the call originates. For both CSD and SSD, no delivery attempts will be made during an administrative period lasting 10 minutes at midnight.

The tentative time for the first delivery attempt,  $t_1$ , is given by:

$$t_1 = \begin{cases} T + 15 \text{ min} & (-0, +15 \text{ min}) & \text{SSD} \\ T^1 & (-0, +5 \text{ min}) & \text{CSD} \end{cases},$$

where

$T$  = time of message entry into AC

$T^1$  = customer-specified delivery time.

As referred to above,  $t_1$  is only a tentative first delivery time because it may be necessary to adjust  $t_1$  to avoid nondelivery periods. The exact time for the first delivery,  $T_1$ , is determined from  $t_1$  by the algorithm in Fig. 2.

At the time of delivery, the service places a call, over the regular telephone network, to the telephone number specified by the customer. Voice-presence circuits will "listen" to the call progress tones associated with the call attempt, and the service will be able to determine the status of the call. It will know that a busy or reorder condition was reached or that the called number is ringing. Any other call condition that might be reached is treated as a ringing connection. If the call attempt reaches busy, the attempt ends and a record is made of the busy condition. If a ringing connection is reached, the service will wait

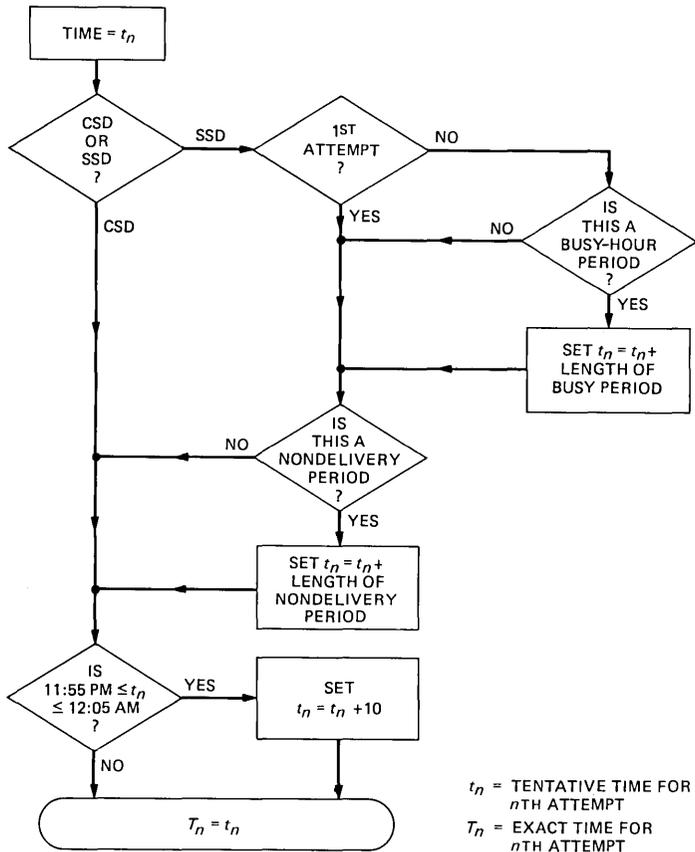


Fig. 2—Delivery time algorithm (find  $T_n$  from  $t_n$ ).

for approximately 30 seconds (five ringing cycles) for answer before abandoning the attempt and records the no-answer call condition. The time of the next attempt is then calculated.

The algorithm for determining the next attempt time considers how many attempts have already been made, whether the previous attempt reached busy or a no-answer condition, and whether CSD or SSD was selected. In general, the time between attempts gets longer as the number of attempts increases. Advance calling will make from six to ten attempts in no more than 13 hours for CSD and in no more than 24 hours for SSD. It was felt that if a customer specified the time of delivery, the message was important and, therefore, delivery should be attempted more frequently.

To determine the tentative time of the next delivery attempt,  $t_n$ , when  $n \neq 1$

$$t_n = T_{n-1} + \Delta t_n,$$

where  $\Delta t_n$  is found in the appropriate Delivery Time Table in Tables II and III. Based on the call condition reached on the previous attempt and whether a CSD or SSD message is being processed, select from the Delivery Time Table the appropriate time interval and add it to  $T_{n-1}$ . The Delivery Time Tables also include a range adjustment factor to allow the service to smooth out traffic loads over time periods to prevent peaking of call attempts.

In addition to the time-table calculations for  $t_n$ , several other conditions must be evaluated:

(i) The total number of attempts,  $N$ , for any message cannot exceed 10.

(ii) If attempt  $n$  encounters a busy after attempt  $n - 1$  encountered a no answer, set  $n = 2$ . This allows for a faster retry on the next attempt to increase the probability of delivery since it appears that the customer is now home but busy on the telephone.

(iii) When the above adjustment has been made, but the next call attempt encounters a no-answer condition, set  $n = N - 1$ , allowing the service to extend the time of the next attempt since it appears that the customer is really not available.

(iv) If there have been three consecutive no-answer conditions *and* the last attempt occurred between 11 a.m. and 2 p.m., add an additional 3 hours to  $t_n$  calculated from the table. This allows the service to move several call attempts into the evening hours to maximize the probabil-

Table II—Delivery Time Table—CSD

$t_n$ if attempt $n - 1$ reached (minutes)		
$n$	Busy	No Answer
2	5 (-0,+5)	30 (-0,+15)
3	10 (-0,+5)	60 (-15,+15)
4	15 (-5,+5)	60 (-15,+15)
5	20 (-5,+10)	90 (-15,+30)
6	25 (-5,+10)	120 (-15,+45)

Table III—Delivery Time Table—SSD

$t_n$ if attempt $n - 1$ reached (minutes)		
$n$	Busy	No Answer
2	15 (-0,+15)	60 (-15,+30)
3	30 (-15,+15)	120 (-15,+30)
4	60 (-15,+15)	180 (-15,+45)
5	75 (-15,+30)	180 (-15,+45)
6	90 (-15,+30)	240 (-15,+60)

ity of delivery to customers who work during the day and are home during the evening.

Once  $t_n$  is determined from the Delivery Time Tables and the additional constraints above,  $T_n$  is calculated from  $t_n$  by the algorithm in Fig. 2.

The generation of the delivery algorithm for AC was based on evaluations of network completion studies and studies of calling habits of telephone customers. But at best, the delivery algorithm is a "best guess." As AC service is used by customers, detailed data on call attempts, time of day, and busy/idle conditions reached will be recorded and analyzed. Results of that analysis will be used to modify the delivery algorithm to maximize the probability of delivery with the fewest number of attempts.

When the called telephone is answered on any delivery attempt, the answering party is greeted by a service Logo and a standard announcement identifying the call as AC service and explaining that a recorded message will be delivered if the customer will continue to listen. The announcement states, "This is the Bell System Advance Calling Service with a personal message recorded earlier for delivery to you. If you stay on the line, your message will be played three times." The message is then delivered for the customer.

The message is considered delivered if the customer hangs up during the introductory greeting or the delivery of the message. No further attempts will be made. The service will record, for billing purposes, the length of the call made to deliver the message.

## XII. DELIVERY STATUS CHECK

A unique aspect of AC service, is that customers may check on the delivery status of their messages up to 48 hours after the message is entered into the system. A customer may check the status of a message or messages by dialing a Delivery Status Check access code (Table I) from the same telephone where the message was originated. If a message or several messages exist for a single destination, the status of each message will be returned with an indicator of the day and time each message was entered into the system (i.e., "your message of Thursday 3:15 p.m. was . . ."). If messages have been recorded for delivery to several telephone numbers, the customer is asked to enter the telephone number of the message for which the status is desired. The appropriate status is then returned.

For each message status requested, one of five possible responses is given:

(i) The message could not be delivered (the service has stopped all further attempts);

- (ii) The message has not been delivered, but further attempts are continuing;
- (iii) The delivery call was completed, but the called party hung up before the message was delivered;
- (iv) The message was partially delivered when the called party hung up; or
- (v) The entire message was delivered.

When a delivery call is completed, the day and time of the completion is included in the status report.

Another feature of AC is that when a customer receives a status report that a message has not yet been delivered, the customer may cancel any future delivery attempts for that message by dialing the cancel code, 0. The service then verifies the cancellation of a message. The status of messages remains available to customers until a delivered or undelivered status is reported to the customer or until 48 hours have elapsed since the message was recorded. The actual message is erased when delivery is made or when no further attempts to deliver are to be made.

### **XIII. BILLING**

There are two parts of the billing for AC service. Customers are charged for entering a message into the service, regardless of the outcome of the delivery attempts. This message recording charge covers the cost of recording, all delivery attempts, and one status check. Additional status checks are charged on a per-call basis. Customers are also billed at standard call rates in effect at the time of day the message is delivered for any local message unit charges or toll charges associated with any delivery attempt that is completed.

### **XIV. CUSTOM ANNOUNCEMENT SERVICE**

Custom Announcement Service (CAS) will provide customers, mainly small businesses and community organizations, with a low traffic announcement service. The service operations are identical to CA service, except that callers cannot leave messages and the customer's telephone is never rung while the service is active. Customers may activate or deactivate the service at any time by dialing the proper access codes (Table I). During activation, a customer records the announcement desired.

The CAS offers the customer the advantage of having the telephone completely free for making outgoing calls. It also allows customers to specify how many simultaneous calls would be answered by CAS. When CAS is combined with Remote Access (to be described next), a customer may set up a recorded announcement service on a telephone number

without actually equipping that number with a telephone. The service would be controlled from any other telephone using remote access procedures.

## XV. REMOTE ACCESS

The above description of CA, AC, and CAS has assumed that the customers only access the service from their homes or business telephones. But there is an identified need for customers to be able to access their services when they are away from their regular telephones. A Remote Access (RA) feature is available for CA, AC, and CAS, which will permit customers to access their services from any DTMF signaling telephone.

Each customer who selects the RA feature must use a customer-changeable, variable-length (maximum nine digits) Privacy Code (PC) so that the service can identify the customer. In addition, customers are given a special seven-digit telephone number which they can call when RA is desired. All customers in the same central office code (the first three digits of a telephone number) will use the same RA telephone number.

When a customer wants to access a service from a telephone other than the telephone on which the service is active, a call must be placed to the special RA telephone number (plus area code if needed). This call will be a charged call with the appropriate local message unit or toll charges applied. Upon terminating to the service, customers will be prompted to enter their home or business telephone number and their PCs. This must be done using DTMF signaling. The service will check the telephone number and the PC entered. If a match exists, the customer is prompted to enter the access code for any service desired (see Table I).

If a match does not exist between the telephone number and the PC, the caller is given a second try. If the second try fails, the caller is disconnected from the service. To deter attempts to "break" a PC (over one billion combinations possible) and gain unauthorized access to a customer's service, records are kept on the number of error attempts made over some time interval. When a preset threshold is reached, RA capability is turned off for that telephone number for some period of time and is then turned on again.

Once customers properly complete the PC check, they enter service access codes and have complete control over their services just as if they were at their base telephones. The only restriction imposed is that customers must use DTMF signaling to control the services via RA. A customer may access CA service to turn it on or off, retrieve messages, or change a personal greeting; AC service to send a message, check on the status of a message or cancel a message; or CAS to turn it on or off

to record a new announcement. Note, that a customer must subscribe to RA on a monthly basis, but via RA, a customer could access and use DCA service.

When an RA customer first subscribes to the service, the BOC assigns a null PC. This permits the customer to gain initial access to the service. When customers pass the PC check using the null PC, they can dial the Privacy Code Change (PCC) access code (Table I) and input any new PC desired. By using the PCC feature regularly, customers can repeatedly change their PCs. Further, no BOC records are made of customer's PCs.

For all services, RA may be assigned to a telephone number without actually having a telephone equipped on that line. This is very attractive for AC since it allows customers in areas where AC is not directly available to send AC messages via RA. To allow customers to make efficient use of RA for AC service, an additional "recycle" capability was designed. At the end of the message recording phase, when a customer would normally have been disconnected, provisions have been made to allow customers to enter the AC access code again to "recycle" back to the beginning of the message recording sequence so that a second message could be sent via the same RA call. Recycle is also available during other phases of AC service. Thus, with the recycle capability a customer could, on a single RA call, send several AC messages, check the status of previous messages, cancel previous AC messages if not yet delivered, and retrieve CA service messages from storage.

As experience is gained with customers using RA, the remote access procedures described here may be extended to other CCS services, e.g., Call Forwarding.

## **XVI. THE FUTURE**

The new services described in this paper offer the telephone customers new and unique communications services. They allow customers to make more efficient use of telecommunications. These initial services establish a solid foundation on which to build future improvements and additional capabilities. The extent to which these services are enhanced will be directly determined by the customers themselves. Many enhancements have already been studied and defined for CA and AC services. The development only awaits customer feedback from the initial services.

One thing is certain for the future. The technology used to provide these new services offers potential for other services that will have an impact on the way people will communicate with each other.



## **1A Voice Storage System:**

# **Architecture and Physical Design**

By R. G. CORNELL and J. V. SMITH

(Manuscript received July 31, 1979)

*To provide new basic capabilities associated with the Custom Calling Services II (CCS II) feature package, a new system has been developed. This system, the 1A Voice Storage System (1A VSS), is a generalized resource with the ability to receive, store, administer, compose, and deliver voice messages or announcements. The capacity of this system is such that it can provide CCS II services on hundreds of simultaneous calls and serve tens of thousands of concurrent subscribers. In this article, the circuit architecture and physical design of the 1A VSS is described.*

## **I. INTRODUCTION**

### **1.1 Background**

Custom Calling Services now available to electronic switching system (ESS) customers have been implemented through software features and the existing switching periphery associated with the ESS system itself. However, the new Custom Calling Services II (CCS II) feature package for No. 1/1A ESS customers has created the need for a new functional capability not available in the existing plant: high-capacity, high-availability, and rapid-access voice storage. Additionally, the complexity and real-time processing demands of CCS II creates the need for auxiliary processor support for the implementation of these services. In response to these needs, a new node in the Stored Program Control (SPC) network has been developed: the 1A Voice Storage System (1A VSS). The 1A VSS is a generalized resource with the ability to receive, store, administer, compose, and deliver voice messages or announcements. The capacity of the 1A VSS is such that it may be deployed in a centralized fashion, each system serving tens of ESS

Central Offices, hundreds of simultaneous calls, tens of thousands of concurrent subscribers to CCS II features, and a total community (via subtending ESS offices) of hundreds of thousands of lines.

## II. SYSTEM ARCHITECTURE

### 2.1 Basic structural concepts

The basic attributes of the 1A vss are that it is a self-contained stored-program-controlled entity in the SPC network, that it interconnects with subtending No. 1 ESSs via trunks, and that it provides random access from these trunks to a massive voice storage medium. Special new services utilizing these resources are implemented by SPC within the 1A vss. It follows that 1A vss embodies many of the structural and operational attributes of an ESS. The basic components of the 1A vss architecture (Fig. 1) are a switching network, highly specialized trunk circuits called voice access circuits (VACs), an ensemble of high-capacity storage devices and their controllers, a central processor and program/data store, a peripheral controller, and an ensemble of service circuits and data set controllers.

Although 1A vss interfaces to No. 1 ESS via analog trunks, all internal switching and storage of voice is implemented digitally. To support this, each VAC is equipped with a Coder/Decoder CODEC which transforms analog voice to/from a digital representation. The digital bitstreams associated with encoded voice are switched via the digital network to storage media controllers, and ultimately to the media itself. Voice storage is implemented digitally via high-capacity moving-head disk transports. Moving-head disk transports provide high-speed random access to stored messages such that the delays experienced by humans after requesting message playback are barely perceivable.

A major objective associated with the 1A vss development has been to minimally impact the design of the ESS offices to which the 1A vss interconnects. The 1A vss and each subtending ESS must intercommunicate frequently to coordinate their efforts toward the implementation of service. Rather than defining the need for a new data link to be established in the ESS to effect this communication, interoffice signaling is implemented via multifrequency (MF) tone packets. The switching network (Fig. 1) is used to establish a path between the relevant trunk and one of the 1A vss service circuits. Once the interoffice signaling phase of a call is complete, the network is used to reswitch the trunk to an appropriate storage controller.

Because the customer may control [via dial pulse or dual tone multifrequency (DTMF) signaling] the operation of 1A vss at any time during a call, a DTMF/dial pulse receiver is permanently assigned to each VAC. This implies a tremendous scanning and control load associated with

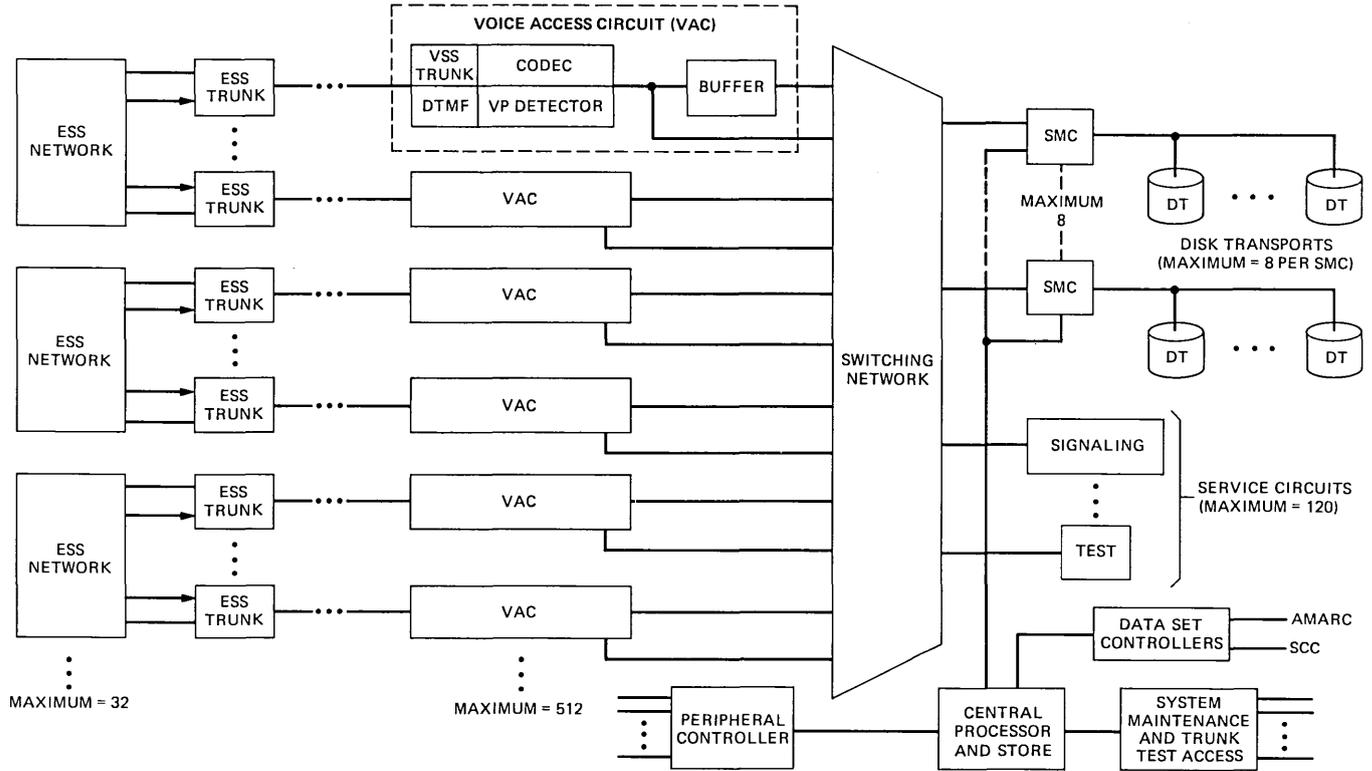


Fig. 1—No. 1A vss architecture.

the 1A vss periphery. Much of the burden of peripheral control has been implemented via microprocessors distributed throughout the periphery itself. Each storage controller is microprocessor controlled, as is the peripheral controller. Each of these processors communicates with the central control via a functional command language which minimizes both the frequency of communication and the amount of central control real time devoted to peripheral control. This provides the central control with the additional real time required to implement the complex CCS II services.

The 1A vss exists within the SPC network as a self-contained office. As such it has been designed to meet the reliability standards established for ESS offices. These standards include reliability and maintainability features in both hardware and software that ultimately limit system downtime to mere minutes per year per system. Specific hardware features include duplication of critical units, each implemented with self-check logic design to enable fault detection. The SPC of 1A vss is the Auxiliary 3A Processor which was originally developed for the No. 3 ESS and No. 2B ESS systems. All signal paths experience an automatic loop-around test immediately before they are switched into a voice path. All programs and data stored on processor memory or disk are duplicated. Voice messages are also duplicated.

At the system level, 1A vss incorporates data links to the Switching Control Center (scc) for remote fault monitoring and control, to the automatic message accounting recording center (AMARC) for automatic message billing features, and to Engineering and Administrative Data Acquisition System (EADAS) for traffic analysis purposes. The 1A vss trunks are automatically diagnosed via the centralized automatic reporting on trunks/remote-office test line (CAROT/ROTL) system.

## **2.2 The voice access circuit**

Each trunk appearance at the 1A vss terminates a VAC. The VAC (Fig. 2) consists of a trunk circuit, a DTMF and dial-pulse receiver, a voice-presence detector, an automatic gain control (AGC) circuit, an analog to/from digital CODEC, and a high-capacity, two-port FIFO buffer. Each function of the VAC is under the control and surveillance of the central processor via a microprocessor-based peripheral controller (PC). The VACs and PC communicate via a synchronous, bit-serial message protocol through transmission over an internal distribution network.

The 1A vss VACs utilize standard Type II E&M trunk circuitry and are available in 2- and 4-wire versions.

An AGC circuit is included to provide for constant playback sound level for both prerecorded announcements and prompts and for messages recorded directly from customers. The AGC circuit also meets a

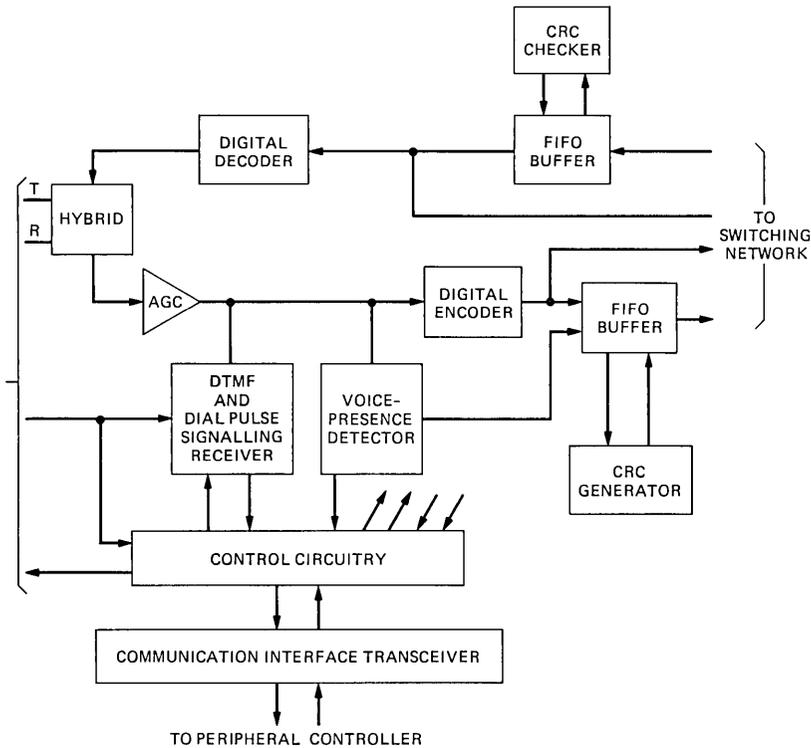


Fig. 2—No. 1A vss voice access circuit.

transmission requirement that all messages be played back at  $-20$  vu (measured at the VAC trunk port) to assure that customers hear 1A vss playback at a pleasing level, regardless of variations in level at the time of message recording. The AGC circuit is under program control in that its function may be defeated at the command of the PC. In the disabled mode, the AGC circuit provides a fixed gain, enabling trunk maintenance testing.

At the time of the message's recording, the speaker may leave periods of silence preceding and following his message. These periods of leading and trailing silence are detected and removed by 1A vss to provide for a pleasing and efficient playback. The voice-presence detector in the VAC seeks to distinguish between signals with the characteristics of voice and signals with the characteristics of noise. Additionally, the voice-presence detector reports "voice present" for continuous signals at high levels to allow for the storage of frequency shift keying (FSK) encoded data. The voice-presence detector functions by observing the rate of change of the envelope surrounding the incoming analog waveform. A nonvarying or slowly varying envelope

indicates steady-state background noise or line noise. A rapidly varying envelope indicates voice. The output of the voice-presence detector is used for the deletion of leading and training silence from messages as will be described later.

Because CCS II customers may signal VSS for control purposes by dialing digits at any time during a call, a DTMF and dial pulse signaling receiver is permanently assigned to each VAC. To overcome the cost penalties of a per-trunk assignment, a new low-cost integrated circuit DTMF and dial pulse receiver has been developed for 1A VSS. The receiver is based on charged-couple device (CCD) filter technology and eliminates the need for bulky RC networks. This new receiver performs at a level of quality equivalent to that of standard central office receivers in approximately an order of magnitude less space and at significantly lower cost. The logic circuitry associated with this new receiver also receives as an input the incoming E-lead signal over which dial pulses from the subtending ESS are repeated. The receiver, upon detecting E-lead activity begins to count dial pulses. Once a valid dial pulse digit is received, the receiver reports the digit as though it were a DTMF digit. Thus, the VAC, PC, and central processor are insulated from the distinction between customer DTMF signaling or dial pulse control. The receiver queues two received digits until the PC interrogates it in order to lessen the scanning load on the PC. The receiver also provides an "early detect" output signal when it begins to observe a DTMF or dial pulse digit as an aid to the record/playback control over messages, as will be described later in greater detail. Once a digit is fully verified, a second "digit present" output is raised to inform the PC that the receiver should be read.

The VAC receives its control via a bit-synchronous serial data transceiver. Each transmission by the PC consists of a 27-bit word, containing 11 address and parity bits and 16 command and status bits. Each transmission to the VAC is followed by a return transmission to the peripheral controller. The bits associated with commands to the VAC are retransmitted back to the peripheral controller in the follow-on reply to allow for transmission error checking by the peripheral controller.

Voice signals to be stored are encoded digitally into a bit serial format at 32 kb/second by an Adaptive Delta Modulation (ADM) CODEC. This CODEC, originally designed for the Subscriber Loop Carrier-40 system, is driven at its level of optimal performance by the AGC circuit which precedes it. Extensive subjective testing has shown that this ADM CODEC with AGC performs at a level of quality comparable with that specified as the Bell System standard for transmission. Upon recording, digitally encoded voice is loaded into a two-port FIFO buffer at the CODEC bit rate. When a sufficient number of bits have been

stored to fill the message storage portion of a disk track (155,078 bits) the disk's controller establishes a path through the network between the buffer and a disk transport and transfers the track's worth of bits at the rate of approximately 10 Mb/second. The buffer must be serviced only once every 4.85 seconds, and hence the disk's controller must be connected to the buffer relatively infrequently. This frees the disk's controller to simultaneously serve the needs of many buffers.

The buffer circuit organizes incoming digitally encoded voice into 1024-bit blocks. Each block consists of 1007 bits of digitized voice, 16 bits generated by the buffer's cyclic redundancy (CRC) generator for the purpose of error detection over the block, and one "voice presence in this block" bit as received from the voice-presence detector. The CRC character is checked by the disk's controller upon transmission to the disk as an error check over the switching path. Upon playback, the disk's controller again checks the CRC characters to verify the integrity of the disk medium. As the message is sent from the buffer to the CODEC, the CRC characters are again checked and stripped out of the bitstream. This again checks the integrity of the transmission from disk controller to buffer, plus the buffer's memory.

Once the buffer is told by the system to begin recording a message sent from the CODEC, it will actually enter a "pseudo record" mode until the voice-presence detector indicates "voice present." In this mode, it records bits but discards them after approximately 120 ms. When the voice-presence detector indicates the presence of voice, the buffer ceases to discard bits and records continuously. This feature eliminates "clipping" of the message potentially introduced by the operational delay of the voice-presence detector.

In the event that a customer should seek to control 1A vss through DTMF or dial pulse signaling while recording or playback is in progress, the DTMF/dial pulse signaling receiver sends a "freeze" command to the buffer. This prevents the customer from losing any of his message while signaling. The receiver's "early detect" output is sent to the buffer and, when active, causes the buffer to suspend its activity. If a "digit present" signal does not follow, the buffer is immediately "thawed." If a digit is detected, the PC and central processor may elect to force the buffer into the "thawed" mode if appropriate (the customer's digit may, however, be a request for some other action on the part of 1A vss).

Because of return loss in the trunk circuit and because of transmission echos, a recorded DTMF digit may be fed back into the receiver and be perceived as a customer-initiated signal. To prevent this, the "early detect" signal freezes the buffer, while the receiver continues to listen for DTMF signals. If a DTMF signal is detected while the buffer is "frozen," the signal must have originated from the incoming line and

not from the message being played back. If the signal originates from the recorded message, the receiver suspends its operation for several hundred milliseconds to lessen the frequency of interruption of the recorded DTMF signals.

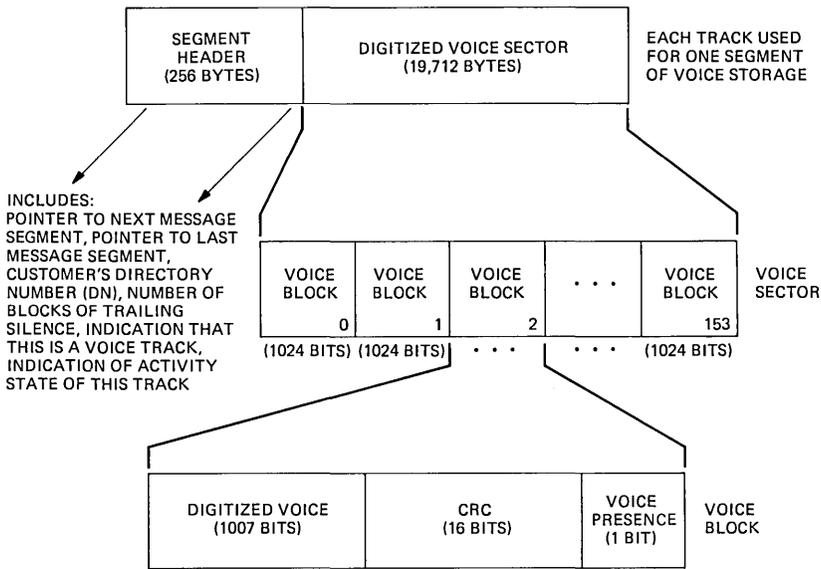
All VAC circuitry, including the buffer, is implemented on two 8- by 13-inch circuit packs. Thus, the plug-in VAC circuit pack represents the growth circuitry on an engineering basis for expanding offices.

### **2.3 The storage media controller**

The functional responsibilities of the disk's controller clearly go beyond that of a computer disk controller. This device, named the Storage Media Controller (SMC), exerts control over the disk media attached thereto, over buffer circuits, and over the electronic switching arrangement that interconnects buffers and SMCs. The SMC is a micro-processor-based controller endowed with considerable intelligence and autonomy. The SMCs communication with the central processor is at a high functional level designed to minimize the frequency of communication with central control and to minimize the real time invested by central control over the detailed operation of its periphery. In this realm, the SMC might receive a work order from the central processor requesting that it play the message beginning at physical location  $x$  of its disk media to customer  $y$ , who is connected to VAC  $z$ . The SMC then accesses the data associated with that message, performs security checks to assure that proper data are being accessed, verifies the integrity of the signal path to the required VAC, and begins to transfer digitally encoded voice to that VAC. This digitally encoded voice is physically realized as many segments of data (see Fig. 3), each segment consisting of 4.85 seconds of recorded voice and 256 bytes of control data. These data include the identity of the customer to whom the message belongs, forward and backward linked list pointers to the next and previous segment of the message, and miscellaneous data regarding the status of the segment. The SMC uses the linked list pointers to manage the playback of an entire message autonomously, and reports back to the central processor when it detects "end of message."

A similar function is executed upon message recording, whereby the central processor furnishes a set of disk storage locations that is available for recording purposes. The SMC manages the building of the data portion of each segment as it is recorded.

The SMC performs somewhat differently in order to compose prompts or system announcements. Digitally encoded voice fragments, each consisting of 63 ms of sound, are "stitched" together by the SMC to create complete phrases. The central processor sends the SMC an ordered set of pointers to many fragments that, taken together, forms a particular phrase. The SMC then seizes the appropriate VAC circuit,



VOICE IS PARTITIONED INTO SEGMENTS, EACH CONTAINING 155,078 BITS (4.85 SECONDS) OF THE TOTAL MESSAGE. EACH SEGMENT IS STORED SEPARATELY ON THE DISK MEDIUM ALONG WITH A DATA HEADER CONSISTING OF A DESCRIPTION OF THE MESSAGE'S SEGMENT POSITION, THE MESSAGE OWNER, THE TYPE OF MESSAGE, AND THE STATUS OF THE MESSAGE.

Fig. 3—No. 1A vss disk data format.

accesses the fragments from among its disk population, and stores the fragments contiguously in the VACS buffer circuit. The short duration of the speech fragment, taken together with the vast number of fragments that can be stored in the SMCs disk population enable an automatic message composition capability of exceptional quality, variability, and scope. For example, all prompts could be duplicated in several languages, with the target language selected independently for the requirements of a particular call.

The SMC executes trailing silence deletion from recorded messages as a post-processing operation. After a message has been completely recorded, the SMC is instructed by the central processor to autonomously begin with the last segment of a message, inspect it via the stored "voice-presence" bits for the presence of voice, and to delete segments that contain no voice. When the last segment containing voice is identified, its data field is updated to reflect this fact, and to indicate the exact location within the segment of the "end of voice." Upon message playback, the SMC will transmit all bits in segments until the last segment is reached. At that time, only those bits preceding end of voice are transferred to the VAC buffer.

The SMC has the capacity to accept work lists from the central processor to conduct up to 80 concurrent record, playback, or compose

operations. To implement this, the SMC scans the number of VACS assigned to it at any point in time and schedules disk data transfers for those buffers currently in need of service. The VACS that are assigned to a particular SMC at any point in time is a function of system load only. These VACS may be any subset of the total VAC population, and the members of this subset change dynamically with time. For example, if a customer is hearing the playback of six messages in sequence, each of these six messages may have been sent to the VAC associated with that customer by six different SMCs. Thus, that VAC moves from the work list of one SMC to the next as playback proceeds.

The order in which the SMC serves VACS is determined by the locations on disk of the stored segments of data associated with those VACS. The SMC schedules data transfers according to an algorithm which minimizes disk head travel and, thus, maximizes the number of transfers that can be conducted in a given period of time.

In addition to managing the transfer of digitally encoded voice to and from VACS, the SMC serves as a traditional disk controller to serve the central processor's bulk data needs. Data are stored on the same disk media used for voice messages, with special track header fields used to distinguish between locations used for data and locations used for voice. The amount of storage space allocated for voice and for data changes dynamically as a function of system operation and load.

Messages are duplicated to ensure message reliability in the event of a disk transport failure. A message is duplicated after it has been completely recorded and all silence-deletion post processing is complete. A message is always duplicated onto a disk transport associated with an SMC different from that of the original recording. Message duplication is initiated by the central processor causing the initial SMC to seize an idle VAC and to move the first segment of the message to that VAC's buffer. A second SMC is then directed to read the VACS buffer as though it were an incoming message. This process is repeated until the complete message is duplicated. Because the buffer is filled and emptied at the high bit rate associated with buffer to/from SMC transmission, the duplication process transpires at a fraction of the time that was required for the initial message recording.

Each SMC equipped in the 1A vss may control up to eight disk transports. The fully equipped 1A vss has eight SMCs associated with it. The disk transports will be described in detail later in this article.

## **2.4 Switching within 1A VSS**

There are two electronic space division switching entities associated with the 1A vss as shown in Fig. 4. The first is a nonblocking Time-Multiplexed Space Division Switch (TMSDS) which serves to interconnect VACS with SMCs. The second is an engineered blocking electronic

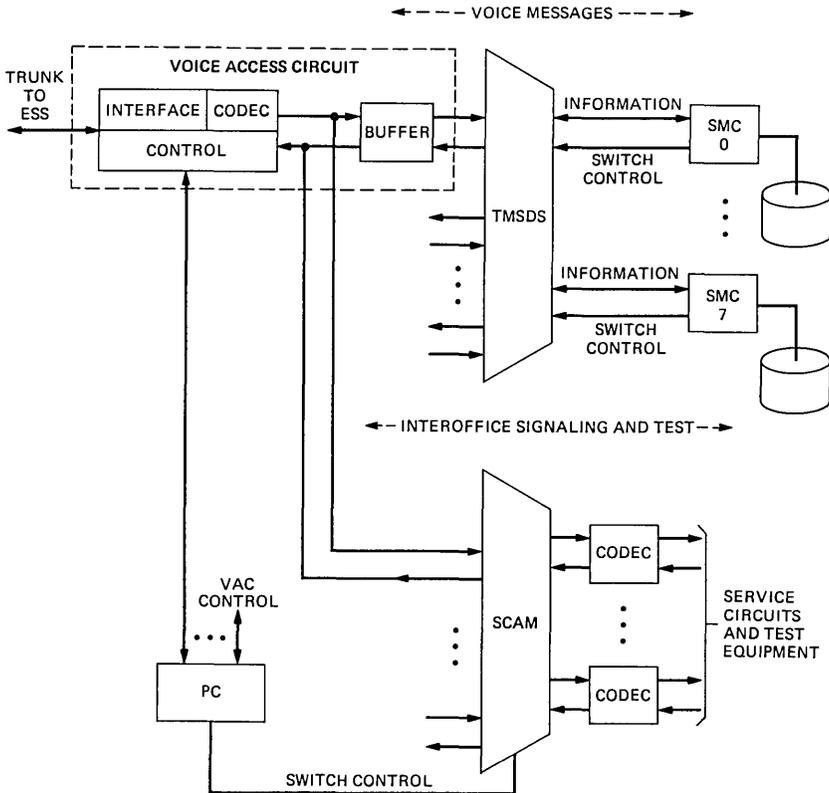


Fig. 4—No. 1A vss network implemented.

space division switch, called the Service Circuit Access Matrix (SCAM), which serves to interconnect VACs with service circuits and test circuits.

The TMSDS is under the combined control of the ensemble of SMCs. Each SMC utilizes the TMSDS as a time-multiplexed switch with regard to data transfer to/from up to 80 VACs on its work list. Each SMC continuously and sequentially steps through paths to those VAC's buffer to conduct a transfer of digitally encoded voice. When a buffer is in a state of readiness, the SMC schedules the transfer according to the current location of the disc track to be used, relative to the current position of the disk's moving head carriage. When the disk is ready for data transfer, the SMC uses the TMSDS to establish a path to the buffer and transfers 157,696 bits (including encoded voice, check bits, and voice-presence bits) in approximately 17 ms. Completing this transfer, the SMC progresses to its next active VAC to effect a similar transfer.

The TMSDS is utilized simultaneously in a manner as described above by as many as eight SMCs. The SMCs operate completely asynchro-

nously with respect to each other and, thus, the TMSDS is under the control of as many as eight independent entities. To prevent collisions of control and/or "lock-up" situations, the TMSDS has integral control circuitry which adjudicates conflicts of control among the SMCs, such as may occur during circuit failures, or during the duplication of voice messages.

A request for access to a VAC by one SMC, while a second SMC is using the VAC causes the TMSDS to put the second request into a "wait" state. The second SMC may wait until an internal time-out occurs at which time it either retries the attempt or determines that a fault has occurred in the TMSDS.

The SCAM is a two-stage space division electronic switch engineered for 0.001 probability of blocking under peak load. The SCAM is controlled by the microprocessor-based peripheral controller in a manner analogous to network control in an ESS. All paths in the SCAM are automatically verified after set-up but before the signal path is completed by a signal loop-around test conducted between the VAC and a "hair pin" in the selected service circuit port.

### **2.5 The peripheral controller**

A microprocessor-based PC has been included in the 1A VSS architecture for the purpose of relieving the central processor of the responsibility for scanning. This peripheral controller also is responsible for the SCAM as previously noted. As described earlier, the PC communicates directly with VACs via a bit-serial communication protocol utilizing a 27-bit word transmitted to and/or from each VAC or SCAM. This word includes commands to the VAC (or SCAM) and received status from the VAC (or SCAM). Additionally, each command sent by the PC to the VAC is acknowledged by the VAC in a response transmission. This provides a checking function to ensure reliability of transmission.

A loop-around transmission is conducted every 40 ms by the PC with each VAC. This serves a basic scanning function associated with trunk status and DTMF signals or dial pulse digit reception. (The DTMF receiver of the VACs queues digits and E-lead signals to permit this relatively slow scanning rate.) Additionally, the PC may be directed by the central processor to perform special communication with VACs at any time in order to effect changes in the state of the VAC (e.g., wink to the ESS, disable the AGC circuit, freeze the recording process). Certain "histories" of events are recorded by the PC for periodic transmission to the central processor. This is exemplified by a history of voice-presence activity seen by each VAC to simplify "time-out-on-no-voice" processing within the application software. Associated with this function, the PC samples the voice-presence detector of an active VAC once each 40 ms and builds a file which documents voice-presence

activity over intervals of hundreds of milliseconds. The central processor periodically interrogates the PC to access voice-presence history files and, thus, is not forced to scan the PC at a high rate to implement a time-out-on-no-voice function.

The PC is duplicated for reliability purposes with one unit active and the other unit in a standby state. The PC incorporates a master clock which controls the population of CODECS. The master clocks of each PC are interconnected in a special arrangement which causes high-speed autonomous switch-over from the clock of the active PCs to the clock of the stand-by PCs upon the failure of the active clock of the PCs. This switch-over precedes the follow-on switch of PCs themselves, which is initiated by the central processor. Thus, a master clock failure does not cause even a momentary system outage because of loss of the CODEC clock. A PC failure is detected by self-check circuitry in the PC and reported to the central processor. The central processor, thus, directs the standby PC to become active. The newly active PC then determines the state of the periphery and assumes control.

## **2.6 Central processor**

The central processor used in the 1A vss is the Auxiliary 3A Processor (AP). Various versions of this processor are also used in No. 2B ESS, No. 3 ESS, Transaction Network System,<sup>1</sup> and No. 5 Crossbar Electronic Translator System. The AP has the general attributes of a minicomputer with the added reliability of duplicated processors, memory, I/O controllers, and communication buses. Self-checking circuit techniques enable the AP to have rapid fault detection and reconfiguration.

Three types of frame, the AP frame, the supplementary main store frame, and the maintenance frame comprise the central processor for 1A vss. The AP frame consists of duplicated 3A central controls, duplicated semiconductor memories and I/O controllers. The supplementary main store frame consists of additional semiconductor memory. The 1A vss uses 768K 18-bit words of memory with the capacity to grow to 1024K words of memory. The maintenance frame consists of a system status panel, two cartridge tape units used for loading programs, and teletypewriter equipment.

## **2.7 Moving-head disk transports**

The storage medium employed by the 1A vss consists of an ensemble of moving-head disk transports, each with a storage capacity of 300 million bytes of digital memory. This translates into a total voice storage capacity per disk of approximately 21 hours.

The disk media itself consists of 11 platters mounted on a single rotating shaft. Nineteen of the surfaces of these platters are useful for

data storage and each is served by a read/write transducer. The many transducers, or heads, are mounted on arms which extend into the assembly of platters, and the arms are fastened to a common moving-carriage assembly which positions the heads radially on the surface of the platters. The access time to a particular datum is a function of the current position of the carriage with respect to the physical location of the datum on the platter. It is to minimize track access response time that a minimal head travel algorithm was selected to dominate control over job sequencing within the SMC.

The reliability of the 1A vss is influenced by the reliability of the alternating current power which serves the disk transport community. To provide virtually disturbance-free power in the face of the uncertainties of commercial power service, a new uninterruptible power supply (UPS) has been developed for 1A vss. This system, called the TRIPORT,<sup>2</sup> after its internal structure, will react to power failure within a cycle of the disturbance and smoothly convert its own battery-supplied inverter for backup without creating significant power waveform discontinuities.

### **III. PHYSICAL DESIGN**

#### **3.1 1A VSS physical design**

The 1A vss equipment uses a standard set of devices, apparatus, and design tools known as the 1A technology. The 1A technology hardware is used in the No. 4 ESS<sup>3</sup> and the 1A Processor,<sup>4</sup> as well as various other ESSs now being manufactured by Western Electric. The use of this technology allowed 1A vss to take advantage of the present manufacturing capabilities of Western Electric. The AP frame, the supplementary main store frame, and the maintenance frame were under manufacture by Western Electric prior to the development of 1A vss.

#### **3.2 Circuit packs**

Circuit packs used in 1A vss are FB-, FC-, and FE-coded packs. The FB- and FC-type packs are approximately 4 by 7 inches, with 40 pinouts and 80 pinouts, respectively. The FE-type packs are approximately 8 by 13 inches, with either 80 or 160 pinouts. These packs use the standard WE 946/947-type of connector.

The majority of the circuit packs designed for 1A vss are six-layer multilayer boards with path widths of 8 mils. Where the wiring density of multilayer boards was not required, double-sided and single-sided boards were designed.

Great attention was paid to maximize the density of circuitry per board. For example, the 1A vss VAC is implemented as two circuit packs—the trunk access circuit (TAC) pack and the buffer pack. The

functions located on the TAC pack include the trunk circuit (Type II E&M, 2- or 4-wire), a DTMF receiver, an AGC circuit, a voice-presence detector circuit, a CODEC, and interface and local control circuitry. To achieve this level of density, a new DTMF receiver based on CCD filter technology has been designed for the 1A vss. Additionally, most digital logic has been implemented as custom-integrated circuits using the new low-power Schotcky gate array technology. Figure 5 is a photograph of the 1A TAC pack.

The buffer circuit pack contains 327,680 bits of random-access memory (RAM) organized as a two-port FIFO memory. Each port may operate asynchronously with respect to the other at speeds defined by the connecting circuits. The 16K RAM used on the buffer pack is the Western Electric coded 28A device. Again, to achieve high-circuitry densities, a large portion of the control logic on the buffer pack is implemented using the gate array technology.

Before the multilayer art masters or the integrated circuit masks for the gate arrays were produced, machine wire-wrapped models were built to test the logic design. Figure 6 shows the buffer pack both in its wire-wrap version and in its final printed wire-board version using gate arrays. Notice that without the use of the custom-designed gate arrays to replace a large number of individual IC devices, the buffer circuitry would have required two packs and connector positions instead of one pack. Because a large number of TAC packs and buffer packs are

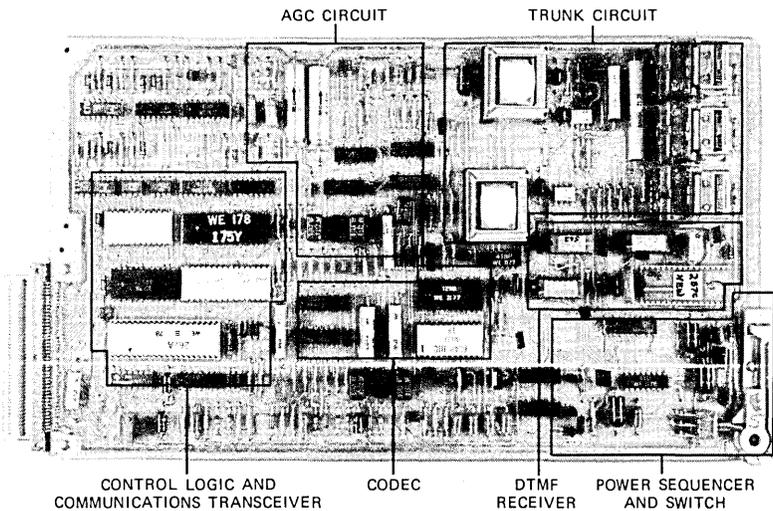
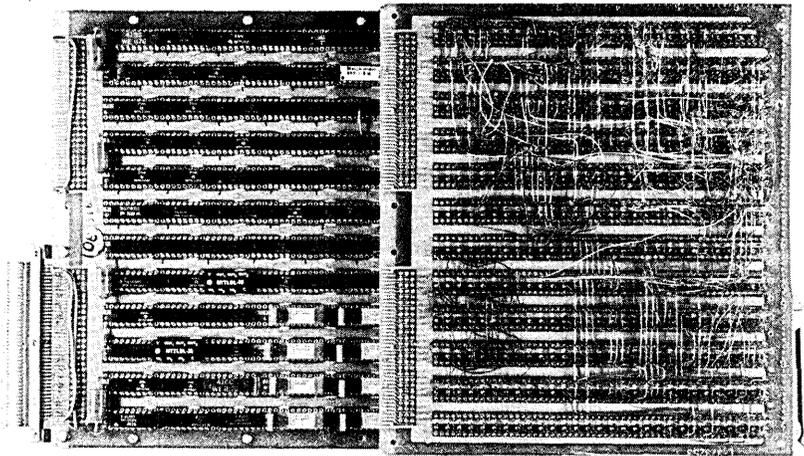
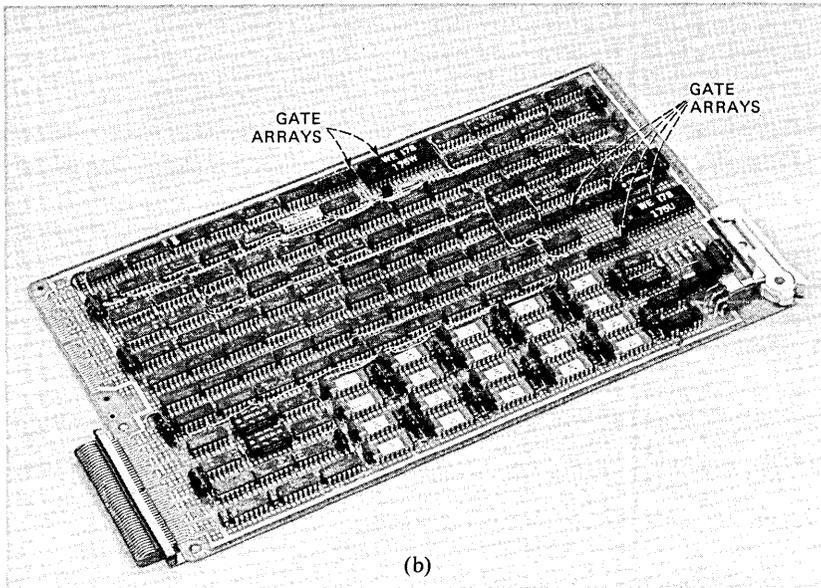


Fig. 5—Trunk access circuit.



(a)

Fig. 6a—Wirewrap buffer.



(b)

Fig. 6b—ML PWB buffer.

required in a large 1A vss office, the savings in space and cost were considerable.

All 1A vss circuit packs associated with “growable” functions (for example, the TAC and buffer packs) are designed to be plugged into

and removed from their units without removing unit power. To achieve this, a power switch has been integrated into the circuit pack handle to allow power sequencing of individual packs to be effected automatically. Figure 5, the TAC pack, shows this switch.

### 3.3 Unit and frame designs

The 1A vss has been partitioned into functional units, each completely self-contained including power converters and alarm circuitry. Figure 7 is a photograph of the SMC unit used to control the moving-head disk transports. This unit, as well as all of the basic units of the 1A vss, was designed using the 1A technology. The mounting plate, apparatus mountings, circuit pack connectors, backplane boards and backplane wiring, and designation strips are all standard apparatus used in other Western Electric manufactured units.

The units are mounted on standard 1A-type equipment frames which are 7 ft high by 2 ft 2 in. wide. All frames are 18 in. in depth. Table I lists all of the frames required for a 1A vss office. The vAC frame is the only frame added to operational systems to serve growth needs. Each vAC frame has a capacity of 32-voice access circuits to the No. 1/1A ESS offices. The minimum number of vAC frames is two to ensure reliability of trunk groups. The vAC frames do not have to be fully equipped. The maximum number of vAC frames is sixteen which provides a total system capacity of 512 TACS. Figure 8 is a photograph of the vAC frame.

Other self-contained equipment provided on an engineered basis are the moving-head disk transports (minimum 3) and the TRIPORT

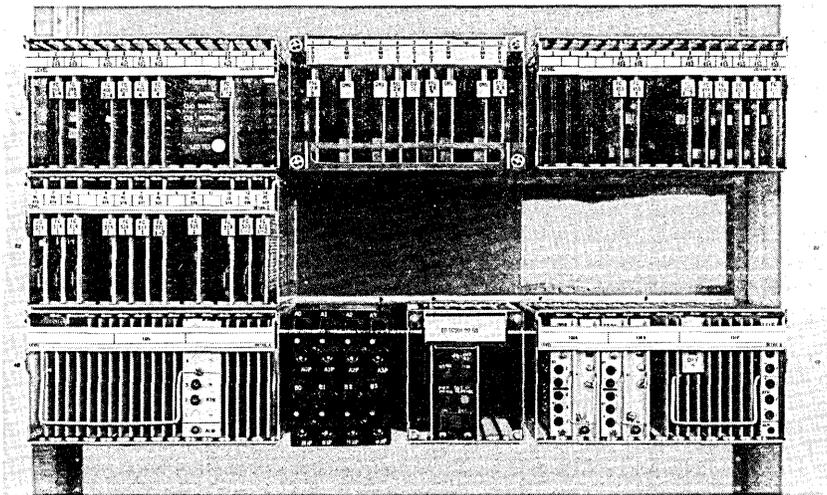


Fig. 7—Storage media controller.

Table I—1A VSS frames

	Size	Number Required Per Office
<b>EXISTING FRAMES</b>		
Auxiliary 3A processor frame	4'4" (double bay)	1
Supplementary main store frame	2'2"	2
Maintenance Frame	2'2"	1
<b>FRAMES DESIGNED FOR 1A VSS</b>		
Voice access circuit frame	2'2"	2-16
Storage media controller frame	4'4" (double bay)	1
Peripheral control frame	4'4" (double bay)	1
Service circuit frame	2'2"	1
Test frame	2'2"	1
Miscellaneous frame	2'2"	1
Power control and distribution frame	2'2"	1
<b>OTHER EQUIPMENT</b>		
TRIPORT cabinet (located in office power plant area)	2'2"	3-8*
Disk transport	23" x 36" x 40" (maximum dimension)	3-24*

\* Based on CCS II engineering requirements.

cabinets (minimum 3) which provide the uninterruptible ac power for the disk transports. The TRIPORTS are usually located with the office power plant.

Almost all cabling between 1A vss units and frames is connectorized. The objective of connectorization is to enable the 1A vss to be fully assembled, wired, and tested at the factory using the same cables that will be used at site. The only cables not connectorized are the scan and distribute leads associated with alarm circuitry.

### 3.4 Floor plan

The 1A vss requires approximately two building bays (approximately 800 square feet) for the maximum-size office. A fixed floor plan is specified in order to use predesigned cabling. This not only ensures cable lengths to meet electrical requirements but also eliminates the line engineering of each cable. Figure 9 shows the floor plan layout of the 1A vss. As mentioned earlier, the only equipment provided on a capacity basis are the vac frames, the disk transports, and the TRIPORTS. The office power plant and TRIPORT frames are usually located in a separate power room. Figure 10 is a photograph of a 1A vss model.

The wiring aisles of the frame line ups are 3 ft wide as compared to a typical 2-ft width in most electronic switching systems. This allows normal office cooling methods to be used, even though a fully equipped vac frame dissipates approximately 1200 watts.

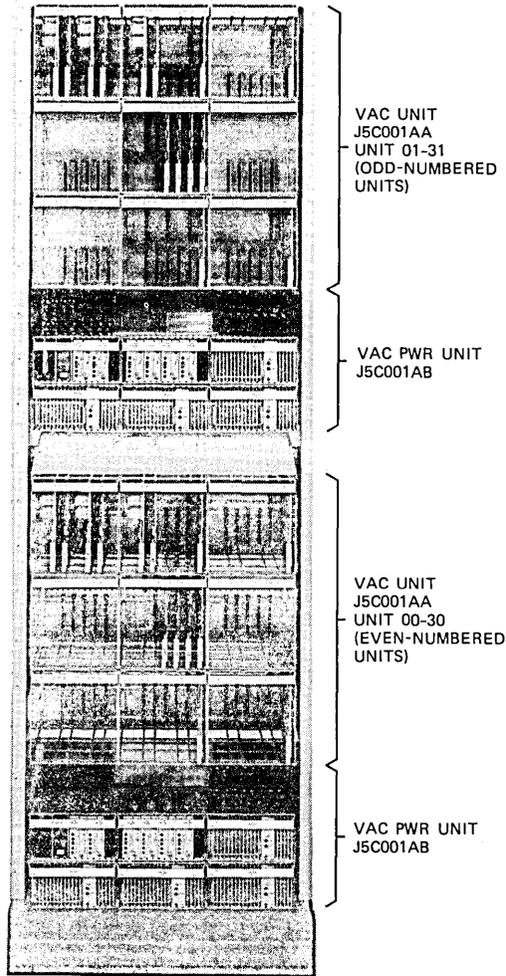
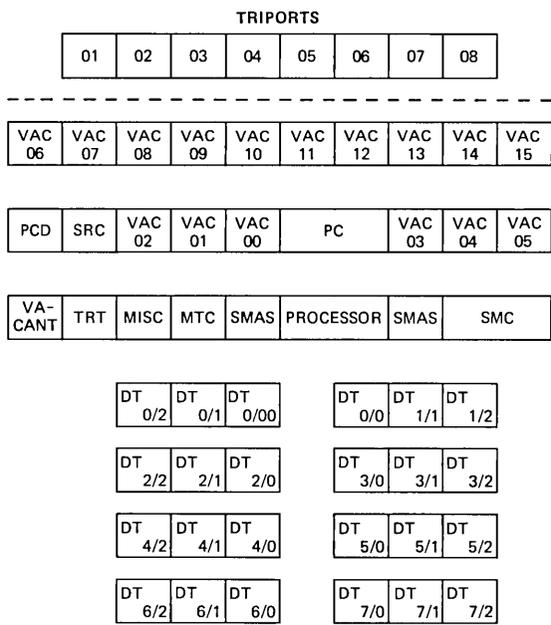


Fig. 8—Voice access circuit frame.

The commercial disk transports used in 1A vss are designed so that cabling enters the cabinet from the bottom. This is consistent with the normal application of disk transports in computer rooms which use raised floors with all cabling being done under the floor. The 1A vss has been designed to be located on either a raised floor or on a regular floor. Apparatus is available to support either cabling to the disk transports under a raised floor or in overhead cable raceways.

#### IV. CONCLUSION

A new centralized system has been developed for the purpose of



DT - - - DISK TRANSPORT  
MISC - - MISCELLANEOUS FRAME  
MTC - - MAINTENANCE FRAME  
PC - - - PERIPHERAL CONTROL FRAME  
PCD - - POWER DISTRIBUTION FRAME  
PRCC - - PROCESSOR FRAME  
SMAS - - SUPPLEMENTARY MAIN STORE FRAME  
SRC - - SERVICE CIRCUITS FRAME  
TRT - - TRUNK TEST FRAME  
VAC - - VOICE ACCESS CIRCUIT FRAME  
SMC - - STORAGE MEDIA CONTROLLER FRAME

NOTE: TRIPORTS ARE SHOWN AS PHYSICALLY LOCATED IN VSS. TRIPORTS ARE USUALLY LOCATED IN THE POWER ROOM.

Fig. 9—Voice storage system floor plan.

adding high-capacity voice storage and voice storage processing capabilities to the SPC network. This system has been developed using a mixture of new periphery design and use of an existing processor complex. The system has been implemented in a generalized fashion

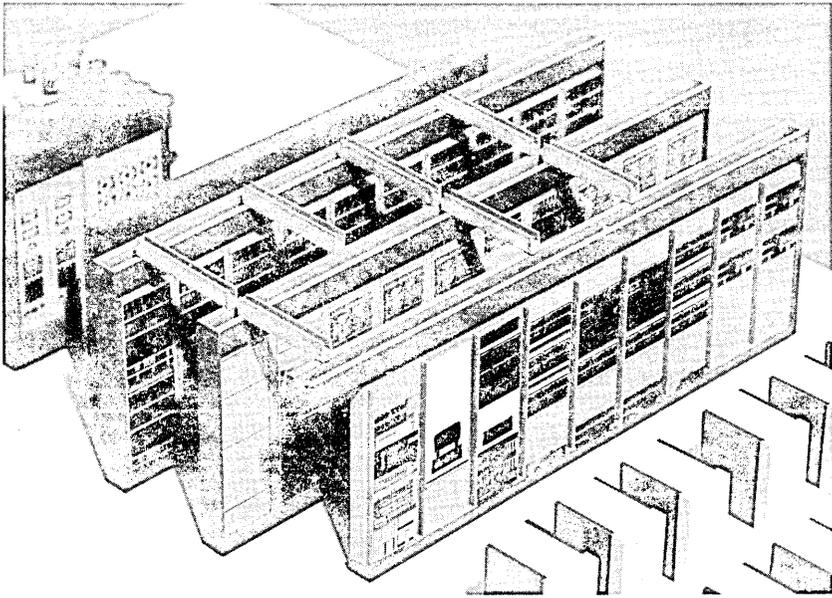


Fig. 10—Model of 1A vss.

which allows the addition of future storage services with minimal (if any) impact upon the system hardware architecture.

#### REFERENCES

1. Transaction Network, Telephones, and Terminals, *B.S.T.J.*, 57, No. 10 (December 1978), pp. 3325-539.
2. R. Ostapiak and R. Rando, "Triport a Low Cost Source of Continuous AC Power," *Bell Labs Record* (July-August, 1978), p. 175.
3. J. H. Huttenhoff et al., "No. 4 ESS Peripheral System," *B.S.T.J.*, 56, No. 7 (September 1977), pp. 1029-55.
4. J. O. Becker et al., "1A Processor: Technology and Physical Design," *B.S.T.J.*, 56, No. 2 (February 1977), pp. 207-36.



## **1A Voice Storage System:**

# **Software**

By G. W. GATES, R. F. KRANZMANN, and L. D. WHITEHEAD

(Manuscript received July 31, 1979)

*The new Custom Calling Services II (ccsII) have been provided by adding a 1A Voice Storage System (1A vss) as a new node in the Stored Program Control network. Software and a new trunk circuit are required in the No. 1/1A ESS to provide call control, call filtering, and routing to a 1A vss. The 1A vss accepts the call and provides the package of voice services known as ccs II. The software required to provide these services is described.*

### **I. OVERVIEW**

#### **1.1 Design considerations**

The software required to implement the new line of Custom Calling Services (ccs II) being developed for the Bell System exists in both No. 1/1A ESS and in the 1A Voice Storage System (1A vss). The software in No. 1/1A ESS is required for call screening, for determining which calls should receive service by 1A vss and for dealing with the interaction of existing services with this new class of services.

There is a strong interaction between the ESS and 1A vss in providing ccs II services. To the extent possible, all service control has been placed in the 1A vss. In addition, all customer data and control data are maintained on the disks in 1A vss. In a general sense, as soon as ESS determines that the calling party requires a 1A vss-provided service, the call is routed to 1A vss. All control data are permanently maintained in 1A vss and required data are sent to ESS when it is needed. When the data are no longer needed, as when a call answering customer deactivates,<sup>1</sup> ESS destroys the data so as to regain the

memory space; the data are retransmitted when the customer next activates.<sup>1</sup>

### **1.2 The SPC network**

The partition of function between the ESS and the 1A vss subsystem is an example of the growing trend in the Bell System to specialize functions in the various nodes of the Stored Program Control (SPC) network. The SPC network is the name applied to the collection of stored program controlled systems which provide customer services and Bell operating companies (BOCs) administrative services. These systems are interconnected by trunks and data links and, hence, are referred to as a network. The network includes the increasing number of Electronic Switching Systems (ESSs), the Operations Support Systems (OSSs), the Traffic Service Position Systems (TSPSS), and 1A vsss. This growing network of stored program controlled systems permits increased sophistication in customer features and in techniques for providing these features. The 1A vss services are an example of the concentration of customer feature implementation in a specialized type of node in the SPC network. Such a node can provide its specialized features to many class 5 offices by having calls routed to it for service.

Figure 1 illustrates the role of 1A vss as a node in the SPC network. The connection to class 5 No. 1/1A ESSs and to OSSs is shown.

## **II. SOFTWARE IN NO. 1 ESS**

### **2.1 Partition of functions between the host office and 1A VSS**

An early objective in the design of 1A vss features was to minimize the impact on the interconnecting (host) ESS and to place the major burden of responsibility on the 1A vss itself. There were several reasons for this important principle:

(i) The 1A vss processor and system software were new and, hence, provided flexible vehicles which could more readily support functional changes as the system matured.

(ii) Although the 1A vss was initially to serve the No. 1/1A ESS host office, other host systems such as No. 2B ESS, No. 3 ESS, No. 5 ESS, and No. 5 Cross-bar Electronic Translator System were also considered as potential candidates for the future. Any function performed within the 1A vss would need to be developed only once, whereas each host office function would require separate development on each system.

Figure 2 depicts the major functions performed by the host ESS and the 1A vss. The 1A vss itself provides the capability for recording, storing, and returning voice messages and announcements, and for interacting directly with the customer to provide the services described in the companion article.<sup>1</sup> For the customer to make use of these

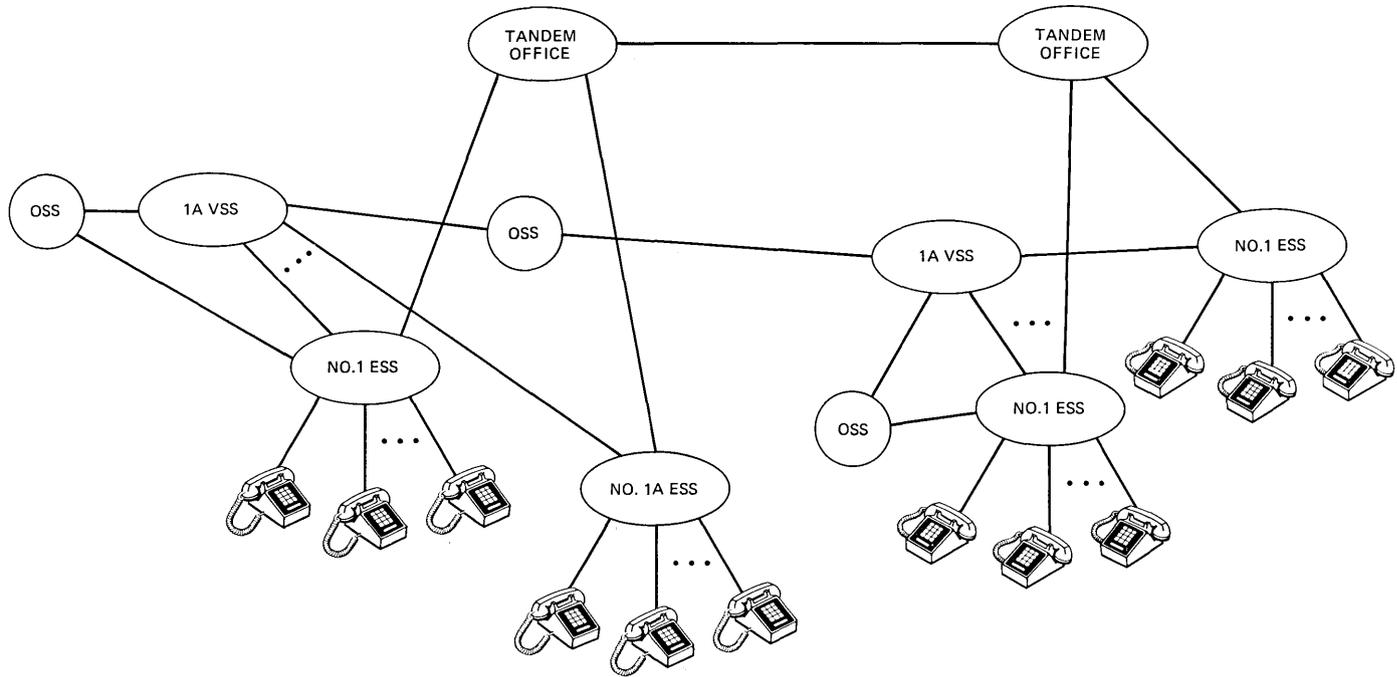
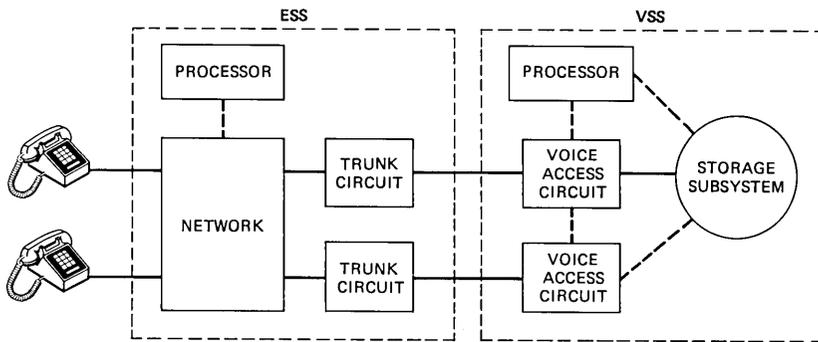


Fig. 1—1A vss: a new node in the SPC network.



**ESS**

- DETECT REQUESTS FOR VSS SERVICE
- PROVIDE CUSTOMER ACCESS TO VSS
- MAINTAIN STATUS OF ACTIVE CUSTOMERS
- INTERCEPT CALLS AND ROUTE TO VSS
- PROVIDE MESSAGE WAITING ALERTING
- PROCESS SERVICE ORDERS AND PASS DATA TO VSS
- PROVIDE LOCAL VSS SERVICES
- PROVIDE SMOOTH INTERACTION WITH EXISTING SERVICES
- MAINTAIN ESS/VSS TRUNKS

**VSS**

- ANALYZE AND PROCESS SERVICE REQUESTS
- STORE, RETRIEVE, AND PLAYBACK MESSAGE SEGMENTS
- PROVIDE FLEXIBLE ANNOUNCEMENT CAPABILITY
- MANAGE DISK SPACE
- PROVIDE FOR INTERACTIVE CONTROL BY CUSTOMER
- ACCUMULATE DATA FOR BILLING AND TRAFFIC
- INITIATE CALLS THROUGH ESS
- MAINTAIN VSS HARDWARE

Fig. 2—ESS/1A vss software functions.

services, the host ESS provides several capabilities, some new and some extensions of existing capabilities. Essentially, these capabilities are of two types: those that are general and those that are related to specific features. General capabilities are as follows:

(i) Customers gain access to 1A vss by dialing the special service prefix (\* or 11) plus two digits, or by dialing seven digits.

(ii) ESS and 1A vss processors communicate using an expanded form of multifrequency signaling.

(iii) Service orders are entered from the ESS with subsequent transmission of service order data to 1A vss.

(iv) Audits of new and modified data are performed to assure the integrity of transient data.

(v) Maintenance of the ESS trunk circuit and the transmission facility to the 1A vss is provided.

(vi) Resource usage counts are recorded for the software resources used.

Capabilities that are related to specific features include:

(i) Terminating calls are intercepted and rerouted to the 1A vss. Intercept is of three types: immediate, busy, and don't-answer. Call Answering service typifies the use of the intercept capability within the ESS.

(ii) The customer is provided an indication that the 1A vss has voice messages waiting to be retrieved by the customer.

(iii) Voice messages are delivered from the 1A vss office through the host ESS for the Advance Calling feature. Also included is the handling of a special AMA billing message sent from 1A vss for billing the terminating portion of the Advance Call.

(iv) Capability is provided for a customer to monitor a call being recorded and to answer the call personally if desired. This feature is called Monitor/Cut-Through.

(v) Coordinated interaction of the new CCS II features with existing features, including those of CCS I, is provided.

The following section presents an overview of the software required for the host ESS in order to implement these capabilities.

## **2.2 ESS software design overview**

The host ESS for the initial implementation of CCS II is the 1/1A ESS. It is beyond the scope of this paper to discuss the detailed structure of the 1/1A ESS implementation for 1A vss services since it would require substantial background in the design of 1/1A ESS software and hardware. Hence, the design will be presented conceptually, allowing the reader to mentally apply it to any familiar switching system, as appropriate.

The ESS software for implementing the new Custom Calling Services can be viewed as a set of new and modified capabilities, each providing a particular part of the service. Figure 3 illustrates those capabilities and the control flow among them. The following conventions apply in Fig. 3:

- Circles represent input/output devices as follows:
  - SO TTY—Service Order (so) Teletypewriter.
  - LINES—Both 1A vss customer lines and others.
  - TRUNKS—Both interoffice trunks to the 1A vss and to other switching offices.
- Rectangles represent the various functional capabilities.

The primary function of each 1A vss-related capability is discussed below.

The Service Order Handler accepts messages from the SO TTY, screens messages to ensure that they are syntactically correct and that the subscriber specified by the message is permitted access to the services specified. It also assembles appropriate SO messages to send to the 1A vss via the Data Message Sender. All 1A vss subscriber service orders pass through a host ESS prior to transmittal of the service order data to the 1A vss processor. This is done primarily to consolidate the administrative aspects of service order processing and to allow the ESS to make appropriate screening checks. Thus, a service

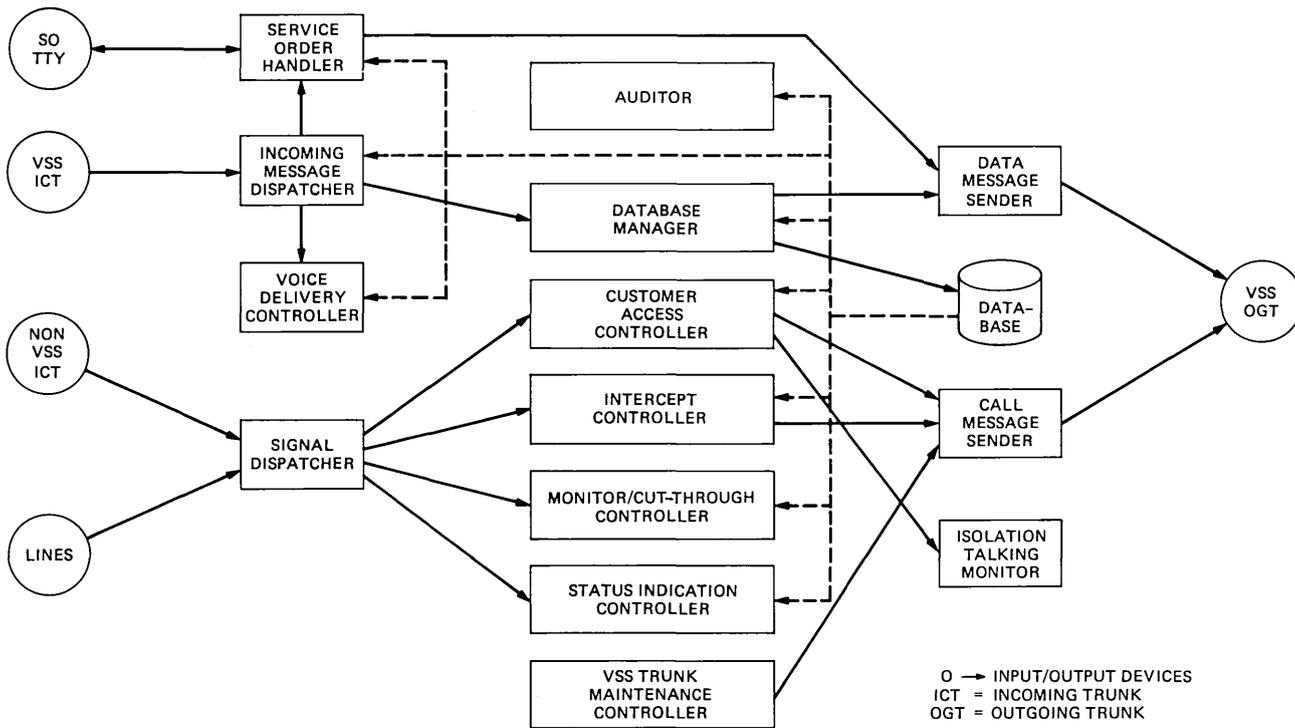


Fig. 3—Host ESS conceptual design.

order describing a new customer for the ESS can simply identify the 1A vss services that the customer wishes to purchase as part of a single order. Note that the entire customer profile describing all aspects of the 1A vss service purchased is maintained in the 1A vss processor and that those items required by the ESS processor are sent to the ESS from 1A vss when the subscriber activates the service.

The Service Order Handler also is responsible for processing 1A vss customer-related VERIFY messages which allow the ESS craft person to check the content of the customer profile in the 1A vss processor. This requires the transmittal of a VERIFY request from ESS to 1A vss and a VERIFY RESPONSE message in the opposite direction.

The Incoming Message Dispatcher receives all MF message packets from 1A vss trunks, analyzes the dispatch code, and distributes the data to the appropriate client program.

The Voice Delivery Controller provides functions required for the delivery of Advance Calling messages. This essentially resolves into a terminating call (if the destination telephone is in the host ESS) or into a tandem call (when the destination is not in the host ESS), with the 1A vss incoming trunk serving as the originating terminal in both cases.

The Signal Dispatcher is a collection of routines which interprets signals from lines and trunks to decide which other processes should be requested to further process the signals. For example, when a customer dials the Call Answering access code \*51, the Signal Dispatcher interprets the \*51 as a request to activate Call Answering service and passes control to the Customer Access Controller.

The Auditor represents the collection of audit programs which assesses the consistency of data for 1A vss-related features. Audits in ESS systems form a powerful force to maintain the stability of the host ESS. Several new data structures for 1A vss services have required corresponding new audit software, while extensions of existing data structures required modification of extant audits.

Note that only the Database Manager is allowed write access to the database representing the pertinent customer profile data within the ESS for active subscribers, whereas many other capabilities are given read-only access. The database for an active Call Answering subscriber includes such items as message-waiting tone and message-waiting ring indicators, Monitor/Cut-Through feature allowed, and identity of the trunk group to the 1A vss. Customer service requests that require access to the 1A vss are handled by the Customer Access Controller. It screens the request to assure that the request is allowed within the ESS, formats an appropriate MF packet, selects a trunk to the 1A vss and passes control to the Call Message Sender. Since some CCS II services are offered on a usage-sensitive basis within the host ESS,

potentially all lines within the office can request activation of these services. However, some combinations of usage-sensitive services, such as Call Answering on COIN lines, are inconsistent or confusing. Therefore screening, based on the originating and terminating major class of the line, is used to control such situations so that specific services may be denied to particular line classes.

Whenever a CCS II subscriber has an active intercept feature, such as Call Answering the Intercept Controller assumes control of any call that would normally terminate to the subscriber's lines. Its function is to perform the actions necessary to route the call to the 1A vss.

The Monitor/Cut-Through Controller processes requests for this subfeature of Call Answering service and is illustrated in the sequence shown in Fig. 4. The MONITOR function is accessed when the Call Answering subscriber dials the appropriate two-digit access code. A check of the subscriber's database is first made to determine whether the Monitor/Cut-Through feature is allowed.

The Status Indication Controller provides for Message Waiting Tone (MWT) and Message Waiting Ring (MWR) to alert a Call Answering subscriber that messages are waiting to be retrieved from the 1A vss. Both status indications are under control of the customer's database profile established by the service prototype (see section on the 1A vss Feature Processor Subsystem). The MWT is provided when applicable on all call originations using software control of dial tone through standard digit receivers. The MWR is a short burst of ringing applied to the customer's line following disconnect from stable network connections involving that line.

The Call Message Sender and Data Message Sender perform the task of transmitting MF packets of information to 1A vss for various software clients as described in the previous section. These modules perform functions, such as seizing appropriate memory resources, a 1A vss trunk, and an MF transmitter, as well as implementing the inter-processor communication protocols.

The Isolation Talking Monitor disables the Call Waiting feature and any similar features that may apply tones or switching noise to a customer's line. This capability is invoked principally, while the customer is in the process of recording a greeting or message on the 1A vss, but it may be utilized in other circumstances to prevent adverse interaction of CCS II with other services to which the customer may have access.

The 1A vss Trunk Maintenance Controller is responsible for providing new diagnostic software to verify the operation of the new two-port trunk circuit, as well as the standard end-to-end operational test provided on many interoffice trunks. Additionally, provision to allow the standard transmission quality tests both automatically and man-

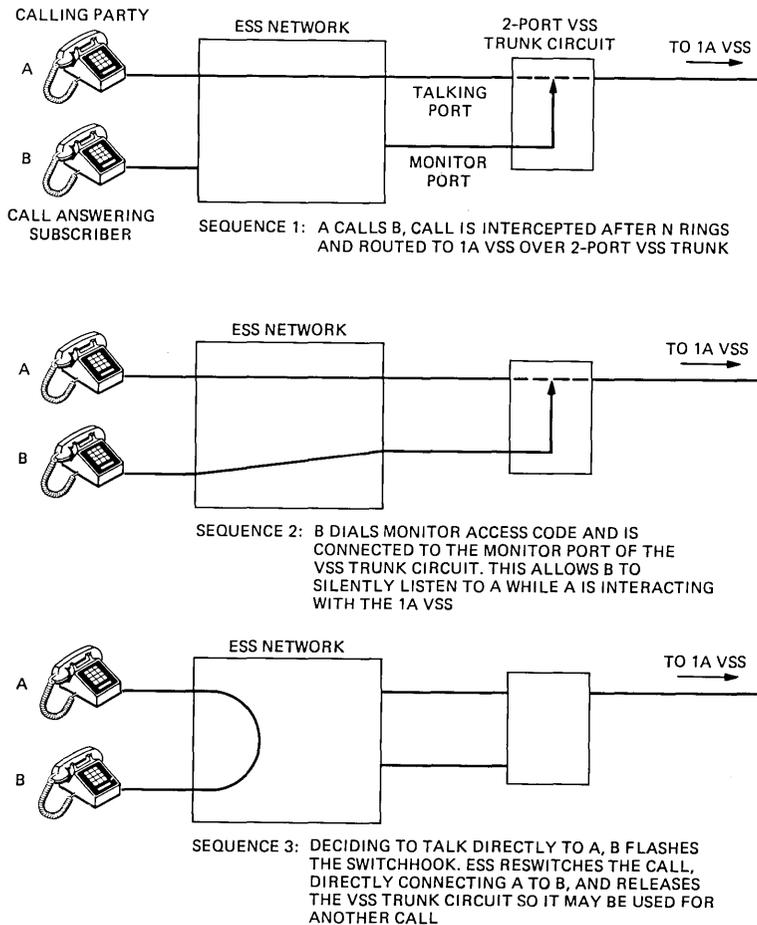


Fig. 4—Monitor/Cut-Through capability.

ually from the various test panel configurations is contained within the Trunk Maintenance Controller conceptual model.

### 2.3 Interaction with existing customer services

Great care was taken to assure that the new CCS II services mesh well with the many customer services already provided by the 1/1A ESS. Two examples are given here to illustrate this interaction.

#### Example 1—Call Answering interaction with Call Forwarding.

Both of these services allow the customer to accomplish a similar goal, namely, when activated, each will result in calls that would normally terminate at the subscriber's line being routed to an alternate

destination. For Call Forwarding, the customer specifies the destination; for Call Answering, the new destination is the 1A vss. Obviously, both cannot be active at the same time or the intent would be ambiguous. Hence, only one type of intercept service is allowed to be active at any one time. If one service is active, subsequent attempts to activate the other result in reorder tone being applied to the customer line.

*Example 2*—Call Answering (busy line) interaction with Call Waiting.

In this case, a priority of action is used. Call Waiting is useful in informing a customer whose line is busy that another caller is trying to reach the customer. This is done by applying a short beep-tone to the customer's line at intervals of 10 seconds. The customer, wishing to answer the new incoming call, verbally informs the original party of his intent to do so. He then flashes the switchhook which results in the customer being connected to the new party, while the original party is placed on "hold." A subsequent switchhook flash will bring back the original connection. However, if the customer also has Call Answering service active and elects not to answer the new caller, the latter will be intercepted after the first 10-second period and will be routed to the 1A vss. In this way, the Call Waiting and Call Answering services conflict minimally and provide enhanced call control capability for the 1/1A ESS customer.

Feature interactions of this sort are implemented wholly within the host ESS.

#### **2.4 The Interface between VSS and ESS**

As shown previously, 1/1A ESS customers gain access to the 1A vss via two-way voice frequency trunks interconnecting the two systems. Signaling associated with the use of these trunks is accomplished via an expanded form of E and M, Multifrequency (MF), wink-start signaling, which is still the most common interoffice signaling arrangement in the Bell System. This communication arrangement was selected since it was most easily adaptable to other existing switching systems.

Communication messages can be divided into two main categories: (i) those messages which will normally proceed to a talking connection between a customer and the 1A vss—Call Messages, and (ii) messages which involve transmission of call control data only—Data Messages. Signaling protocol for Call Messages is quite standard, except for the content and amount of information to be transmitted. Typical information in an MF Call Message packet specifies the type of call message (e.g., Call Answering activation, or Call Answering intercept), the restriction class, the 1A vss customer identity (by Directory Number),

and the billing specifications. Standard Call Message protocol is illustrated in Figure 5, Section A.

Consider for example the Call Message MF packet requesting activation of Call Answering service for a casual user. The MF packet sent from the host ESS to 1A vss would be of the form

MESSAGE RESTRICTION BILLING  
 (KP) DISPATCH CUSTOMER DN CLASS DN (ST),  
 CODE

where

- KP =KEYPULSE DIGIT
- Message Dispatch Code =2-digit code identifying the receiving client program in 1A vss.
- Restriction Class =For example, "complaint observing requested by this customer."

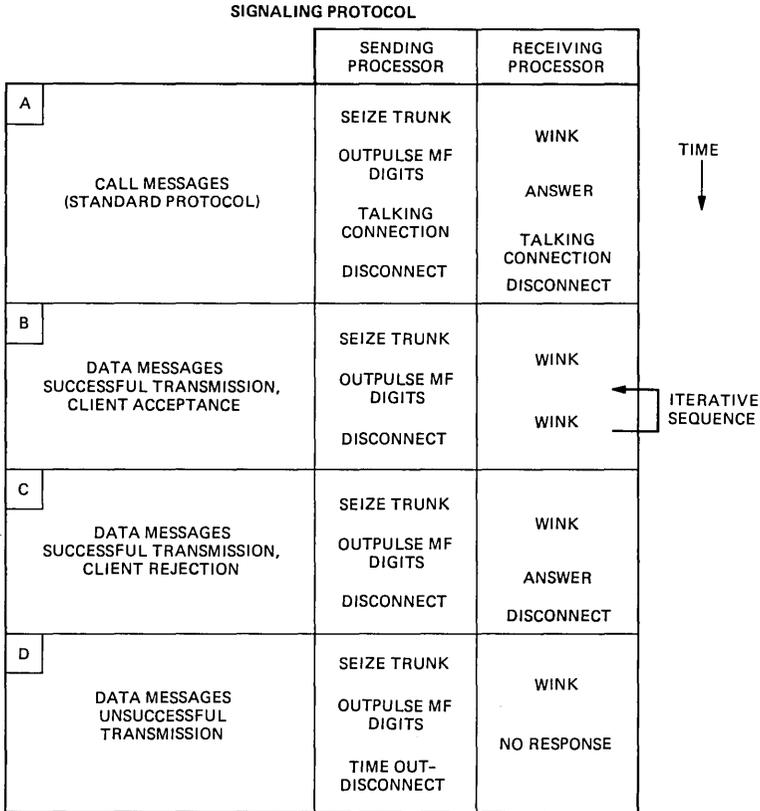


Fig. 5—ESS/1A vss interprocessor communication.

Customer DN	=4-, 5-, or 7-digit form of the customer directory number (DN).
Billing DN	=Directory number to which 1A vss should bill the charges for this use of Call Answering service (optional).
ST	=“START PROCESSING” digit (end-of-message).

There are 99 two-digit message dispatch codes. Data Message MF packets have much the same format as Call Message packets. The dispatch code distinguishes the packet as a Data Message and identifies the client program in the receiving processor. Examples of Data Messages are (i) the activation message sent from 1A vss to ESS in response to an ESS customer's activation request, and (ii) service order messages and their replies.

Signaling protocol for Data Messages is necessarily more complex than the Call Message protocol since it involves interprocessor communication without the presence of the customer to detect the success or failure of the communication. Figure 5, Sections B, C, and D depict the signaling protocol for Data Messages. Note that one of three responses is expected from the receiving processor:

**WINK**—Implies successful transmission of the message and acceptance by the client program.

**ANSWER**—means the MF packet was received but was rejected by the client program, for example, because of incorrect format.

**TIME-OUT**—implies unsuccessful transmission in the same sense as standard interoffice MF signaling.

To increase trunk usage efficiency for data transmission, capability to batch Data Messages is provided. Batching means that multiple independent Data Messages may be transmitted over a single trunk. After receiving the WINK acknowledgment, the transmitting processor will either disconnect, signifying end of transmission, or commence sending the next Data Message.

### III. SOFTWARE IN THE VOICE STORAGE SYSTEM

#### 3.1 *Software techniques*

Control in 1A vss is distributed among the central processor and several microprocessors which control the periphery. The microprocessors provide the necessary low-level repetitive control and relieve the central processor of this workload.

A major goal in the design and implementation of the central processor software was that it be easily modifiable. Several techniques and rules were used to achieve software that allows new features to be added easily.

(i) The fundamental technique is embodied in the software architecture. The software architecture was designed to partition the complexity of the system so that designers and programmers have to concern themselves with only a subset of the total problem.

(a) Software that requires knowledge about the detailed hardware characteristics is concentrated into a few subsystems and the need for this knowledge is eliminated from other subsystems. This technique makes feature development easier by controlling the need for detailed hardware knowledge and makes it easier to change the hardware and firmware. It also results in less overall software impact when hardware and firmware are changed.

(b) The software that controls feature operation is concentrated in one subsystem. Other subsystems provide high-level service functions to the feature subsystem. This effectively creates a language of functions which can be used to build and expand services.

(ii) Within the feature subsystem each feature is implemented as independent software. To do otherwise would mean that each time one feature was changed, other features could be affected.

(iii) Separate data structures are built for each feature. Shared data structures for one customer seems natural, but if the features are completely disjoint, then the data are kept disjoint to avoid interaction effects.

(iv) A high-level language with data structure definition capability is used.

(v) Many characteristics of 1A vss operation are implemented as system parameters so that they will be easy to change as experience is gained from early customer use. Some of these parameters will require software recompilation to change, while others can be changed by modification to the system in the field. Examples of system parameters include: (a) number of seconds of silence before time out during recording, and (b) the time to return answer supervision during a call. The concept of parameters is oriented toward overall system characteristics. An analogous concept of options on specific customer features is used and is discussed in the section on feature implementation.

(vi) The software was built with the rule, "Design it correctly, build it, then tune it." Tuning a system too early can destroy its structure and, hence, destroy its modifiability.

(vii) The call-related portions of the software were designed and implemented using finite-state-system concepts.

The finite-state-system design technique consists of partitioning the software into functional models where each model is viewed as a finite state automaton. The model consists of states, signals, and transition

routines. The occurrence of an event causes a signal to be sent to a model which is in a particular state. The model executes particular transition routines as a function of its state and the received signal. It then enters another state to wait for another signal. Figure 6 illustrates the diagram of such a model.

Finite-state design techniques provide a good structure for the implementation of call processing. They produce software which is self-documented by the state diagrams of each model. Because of this documentation and the intuitive naturalness of the structure, the resulting software is easy to read and understand.

These techniques have produced call processing software which should be easy to modify as new features are added to the 1A vss. Initial indications are that this goal has been met, but several years of experience will be required before a final judgment can be made.

### 3.2 Software architecture overview

The 1A vss software runs under control of the Extended Operating System (EOS), a real-time control system developed for Auxiliary 3A Processor applications. Call handling software runs in a single EOS task and is controlled within the task by the State Table Controller (STC). The STC provides the structure necessary to process signals and to

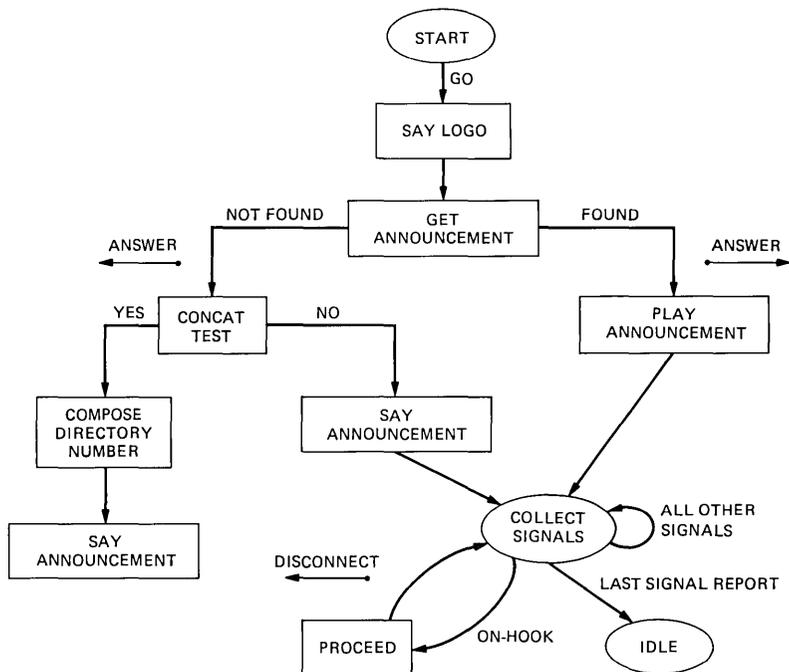


Fig. 6—Example of finite state model.

control models as required for the vss finite-state design techniques. The STC schedules models, queues signals and maintains state control for each model.

The overall structure of the call processing software is shown in Fig. 7. The software is divided into six major subsystems. The Feature Processor controls the actual customer features. It calls on the Recording Trunk and the Database Manager for services; these systems in turn request lower-level services from the Input/Output Processor, the Voice Manager, and the File System. Support services are provided by device handlers, disk memory allocation software, a message duplication service, and disk erasure software.

### 3.3 The feature processor subsystem

The characteristics of a customer feature are incorporated in the Feature Processor subsystem. Like all call processing software, the Feature Processor is a collection of finite state models which performs transitions from state to state as the various call events occur. Events such as "off-hook," "customer dialed a 6," or "message playback complete" cause signals to be sent to the appropriate model. The model executes transition subroutines, sends signals if required, and enters another state to await the next signal. Each call in the system creates an "instance" of each model as the call progresses (similar to a software process). Multiple call capability comes implicitly from the collection of all instances of these models.

The set of Feature Processing models and associated transition routines orchestrates the handling of calls but does little of the actual work. The work is done by calling on the Recording Trunk and the

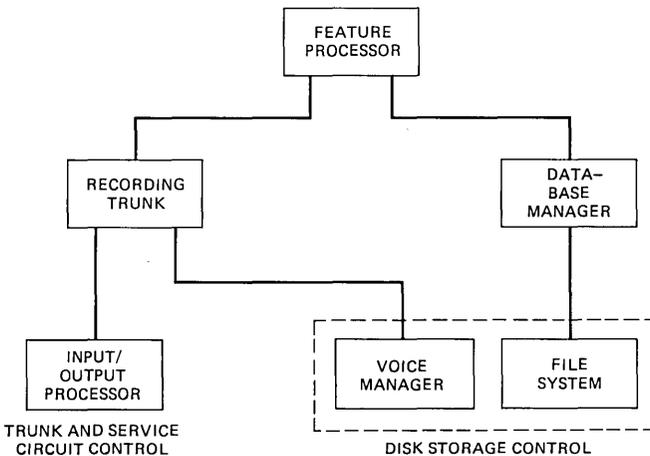


Fig. 7—Basic structure of call processing software.

Database Manager. These two subsystems provide an extensive set of high-level functions which constitutes a primitive language for building customer services.

Examples of the type of functions provided include:

- (i) Report origination
- (ii) Report on-hook
- (iii) Return answer
- (iv) Get dialed digits
- (v) Disconnect
- (vi) Seize trunk
- (vii) Send data
- (viii) Say a system announcement
- (ix) Play a customer message
- (x) Compose an announcement from fragments
- (xi) Record a message
- (xii) Erase a message
- (xiii) Secure customer data
- (xiv) Release customer data
- (xv) Modify customer data.

Many options have been incorporated into each service in order to be responsive to the changing needs of the telephone customer.

The solution to handling the changing needs of the Call Answering customer was to implement essentially all the options which were considered useful and to provide a way to define a customer feature as a collection of these options. Additionally, several packages of Call Answering services can be marketed by defining several collections of options. A set of options is called a prototype, thus, a package of Call Answering options is defined by defining a prototype. The three packages of Call Answering to be marketed initially, i.e., Daily, Monthly, and Deluxe, are created by defining three prototypes with the associated, required sets of options. Further flexibility was gained by providing the capability, through customer service orders, to modify each of the options for the individual customer. Thus, a deluxe customer can have the maximum length of a message extended from 30 to 60 seconds by a service order indicating such a change for that one customer.

#### IV. THE RECORDING TRUNK SUBSYSTEM

The Recording Trunk Subsystem is an abstraction of an "idealized voice storage trunk." Such an "idealized" trunk in 1A vss would be capable of recording and playing messages and handling timed sequences in an autonomous manner. By abstracting these characteristics and incorporating them into a software system, feature designers

are given a powerful capability for building voice features which do not require intimate knowledge of the complex 1A vss architecture. The Recording Trunk Subsystem provides the feature programmer the ability to:

- (i) Record a message
- (ii) Play a message
- (iii) Erase a message
- (iv) Return answer supervision
- (v) Control silence timeout
- (vi) Acquire allowable digits
- (vii) Control digit timing
- (viii) Recognize flash signaling
- (ix) Send messages to the ESS office
- (x) Receive messages from the ESS office.

The Recording Trunk calls upon the Voice Manager (VM) and the Input/Output Processor (IOP) in providing functions to the Feature Processor.

## V. THE DATABASE MANAGER AND FILE SYSTEM

Data services are provided to the system by the Database Manager and the File System. The 1A vss Database Manager was designed and tuned to fit the type of support needed by vss features. Rapid access to the customer database is provided by the physical clustering of logically adjacent data. Flexible database services are achieved by basing the design on the general concepts of the relational model of data structures.

The File System provides for the random access storage and retrieval of variable length records. To provide the required reliability, each record is duplicated when written. The File System and the Database Manager are designed to specifically complement each other so as to meet the objective of minimization of data storage and transfer costs. The File System also provides storage services directly for administrative data such as billing and traffic data and the collection of data on equipment failures.

## VI. INPUT/OUTPUT PROCESSOR

The IOP provides functional control and error detection for the 1A vss trunks and service circuits. In this capacity, it receives requests for service from the Recording Trunk and system maintenance software. Control is achieved through interaction with the microprocessor-based peripheral controller, with responses from the periphery distributed to the requesting subsystem. High-level functional device requests are

accepted by the IOP and transmitted to the periphery as a sequence of device commands.

The IOP also receives notification of autonomous events from the periphery, e.g., a trunk seizure. These are distributed, as appropriate, to the associated Recording Trunk, maintenance or error control software. Timing control and error recovery are also provided by the IOP.

## VII. THE VOICE MANAGEMENT SUBSYSTEM

### 7.1 Voice manager

The Voice Manager encompasses all software for control and manipulation of stored voice. The operational unit of access is the message. Messages may have a variable length and are comprised of one or more fixed-length segments. These segments are the fundamental units of storage allocation and deallocation. Each message has a unique owner. The owner may be either a customer, designating a message entered in conjunction with the customer's service, or the owner may be the system, designating a system announcement identified with a particular vss service.

The Voice Manager provides the Feature Processor three basic capabilities for manipulating messages: the ability to record a message, the ability to play a message, and the ability to erase a message when no longer needed, thus, releasing the space for other uses. Each service is defined as a sequence of these operations with appropriate system announcements played to prompt the customer.

Because it is impractical to store certain system announcements in prerecorded form, e.g., "You have *seven* messages," the VM provides the capability to play such messages from a small set of prerecorded fragments. This allows the Feature Processor to specify the phrase "You have seven messages" as follows:

Specification	Fragments
Fragment identifier	"you have"
Decimal number	seven
Fragment identifier	"messages."

To allow a reasonable range of numbers, individual fragments are recorded. The following are examples of such fragments: the numbers 1 through 19, plus 20, 30, such phrases as a.m., p.m., and the names of the days of the week.

### 7.2 Storage media controller handler

The Storage Media Controller (SMC) handler provides the communication path between the Voice Manager and the microprocessor-based peripheral device which performs the voice commands.

The structure of the handler is governed by two characteristics of the SMC: its ability to service many active calls simultaneously and its highly autonomous operation. The handler defines four phases of operation, described below, and provides all necessary synchronization. These phases are staggered for the equipped SMCs to smooth the main processor load.

**Command Phase**—The handler transmits all voice commands accumulated during the last cycle.

**Voice Phase**—The handler is dormant. The SMC schedules and performs the commands it has been given, signaling the handler when complete.

**Reply Phase**—The handler reads the status replies indicating the disposition of the commands performed during the voice phase.

**Disk Controller Phase**—The SMC performs the functions of a conventional disk controller, e.g., data read/write. This state holds until the start of the next cycle.

### **7.3 Storage allocator**

The Storage Allocator is charged with managing the storage available for digitized voice in the SMC community. This storage is addressed by specifying the SMC, Disk Transport, and segment. The number of SMCs and Disk Transports varies with office configuration; segment numbers are a property of disk geometry. The basic strategy of monitoring the idle/busy status of segments and providing rapid allocation/deallocation of resources employs a combination of main-memory-resident segment address lists and disk resident maps. The memory lists provide for normal, rapid resource allocation/deallocation, while the disk maps maintain the idle/busy status of all resources and provide a permanent record of disk configuration.

## **VIII. SYSTEM MEASUREMENTS**

Since 1A VSS is a switching-type entity, it requires traffic and billing data and interfaces to both the Engineering and Administrative Data Acquisition System (EADAS) and the Automatic Message Accounting Recording Center (AMARC) to make these data available to the BOC. However, 1A VSS provides an entirely new class of vertical services, CCS II, without precedent. Hence, little information exists upon which to gauge, for example, customer response to the new services or traffic engineering rules.

A single, unified data collection mechanism is provided to meet data collection requirements. For each system activity of interest a unique action is defined with regard to data content, reasons for collection, method of collection, and intended uses. Because of the different uses

of the data, two primary collection methods are provided: the peg count, typically used for traffic actions which appear on the quarter hourly reports, and the transaction file, in which the action, and associated parameters, are written into a disk file.

#### **IX. SECURITY AND RELIABILITY**

A major concern during the design of 1A vss has been the protection of customers' messages. The two aspects of this concern are that a message should be returned only to the correct customer and that messages should not be lost. To guarantee the correctness of delivery, the identity of the customer owning the message is stored in the control portion of each message segment. During playback, the identity of the requesting customer is passed to the SMC and each message segment is validated before being played. During recording, each message segment is checked before being written to verify that its current owner is the Storage Allocator. These checks protect against several types of failures that might cause a message or a message portion to be played to the wrong customer.

Messages are protected against loss by replicating each one. System announcements and voice fragments are replicated on each SMC and are accessed via translators. This is done both for reliability and availability. Because of the lower access rates, customer messages are duplicated for reliability. To accomplish this, the following deferred duplication scheme is used. As the customer speaks the message, one copy is recorded in real time. Upon completion of this recording, a duplicate command identifying this newly recorded message is placed on a queue. A background program serves this queue as processor and SMC time permits, and records the second copy of the message. To provide the desired quality of service, a system parameter specifies the maximum tolerable elapsed time to duplication. If this value is exceeded, duplication takes precedence over other new work, until the desired level of service is restored.

#### **IX. SUMMARY**

ccs II services are provided jointly by software and hardware enhancements in No. 1/1A ESS and by a new voice processing system, 1A vss. The software for these services has been provided in such a way as to provide economical service and to permit straightforward expansion to new voice services in the future. The 1A vss software contains the basic voice control software functions needed for many types of voice services. These building block functions permit expansion of ccs II to meet marketing requirements.

## REFERENCES

1. D. P. Worrall, "1A, Voice Storage System: New Custom Calling Services," B.S.T.J., this issue.
2. R. P. Almquist et al., "Software Protection in No. 1 ESS," Proc. Int. Switching Symposium, 1972, pp. 565-9.



## **1A Voice Storage System:**

# **Office Engineering, Maintenance, and Reliability**

By P. W. BOWMAN, J. C. KENNEDY, and G. A. VAN DINE

(Manuscript received July 31, 1979)

*The quality of service that will be delivered by the Voice Storage System (1A vss) is influenced by a number of diverse factors which are addressed in this paper. High intrinsic reliability is designed into the system at every level; it is built from system redundancies, defensive software strategies, hardware self-checking, and manufacturing quality control. Good service also requires a complement of equipment which can adequately handle peak traffic loads; the sizing of this complement is the job of Office Engineering. Effective maintenance software will minimize, and in most cases obviate, the impact on service of circuit component failures. And finally, effective plans and test equipment are required to ascertain that a newly installed 1A vss properly performs its intended functions.*

## **I. INTRODUCTION**

The 1A Voice Storage System (1A vss) comprises a new functional node in the telephone network, which will provide a number of new services classified generally as Custom Calling Services II (ccs II). These services provide the means for conveying voice messages between customers who do not happen to be at their telephones at the same time.

A conventional telephone switching office mediates communication between two parties by setting up a connection over which they may converse. Unlike a switching office, the 1A vss takes on the role of one of the conversing parties. In the course of handling a typical call, the 1A vss receives input control signals at various times, outputs system announcements in response, and either records or retrieves customer messages.

The quality of service is important; yet, this type of communication is complex; there are many opportunities for system malperformance to disrupt the call. The following paragraphs introduce several factors that contribute to high quality service; they are expanded upon in the remainder of this article.

Office Engineering, which is covered in Section II, is the process through which a Bell operating company (BOC) decides specifically what equipment to order to provide CCS II.<sup>1</sup> Inherent in the process are estimates of the amount of traffic to be expected as a function of time. This information is derived from market studies which project the expected usage of specific services in specific customer sectors.

Once a system is put into service, however, it is important to obtain feedback which will allow the early usage estimates to be examined in light of actual market acceptance. With such information, intelligent decisions can be made for office growth, which tracks increasing service demands. The collection and utilization of these usage data is covered under Traffic Measurements in Section 2.4.

With 1A vss services in use, the system is being entrusted with large numbers of customers' messages. Extensive precautions have been taken in all phases of the system design and manufacturing process to keep these messages secure and to assure their delivery. The design and manufacturing strategies employed to assure dependable 1A vss performance are covered in Section III under the heading of Reliability.

Although the best quality obtainable with present technology is built into the 1A vss components, precautions are taken against the occurrence of component failures by providing spare equipment and maintenance software which can locate the malfunctioning component. This software then responds to system troubles by configuring a working system that excludes the faulty unit. The philosophies and methodology employed to keep 1A vss operational and to facilitate repair are presented in Section IV.

As a final step in office installations, many system tests are performed by the Western Electric installation organization. These are followed by BOC tests which verify the system's interfaces with other systems. Associated feature tests verify that the 1A vss works properly when accessed through the connecting ESS offices. This testing is covered in Section V.

## **II. OFFICE ENGINEERING**

### ***2.1 Planning***

The engineering of a 1A vss office and of the associated equipment in 1/1A ESS is performed to determine the specific equipment that is required to provide CCS II. Initially, trial office engineering is performed

for economic feasibility study purposes. Such trial economic studies result in a decision as to whether or not to proceed with a local offering of CCS II. After a decision is made to proceed with 1A vss, an order is placed for the required equipment.

To engineer a 1A vss office and the equipment in the associated 1/1A ESS offices, CCS II marketing information is necessary. Estimates of the rate of penetration and the ultimate market penetration for the services being provided are required for three types of stations: residential, small business, and large business. Providing this essential input is the responsibility of the BOC marketing organization.

The schedule for introducing CCS II in the 1/1A ESS offices must also be determined. For each 1/1A ESS that will provide these services, the number of residential, small business, and large business stations is required. The estimated annual growth rate for each of these categories, and the period of time for which the equipment is being engineered, must also be specified.

The above marketing and 1/1A ESS information comprise the required inputs to the worksheets which guide BOC engineering personnel through the numerous calculations required to determine the amount of equipment required to provide CCS II services and to support their growth.

## **2.2 Storage and offered load**

Storage time is a new traffic parameter associated with 1A vss. Storage time is defined as the time a voice greeting or message is stored in 1A vss before it is erased. The average storage time multiplied by the number of greetings and messages recorded during this time gives the number of simultaneous greetings and messages in the system. This, in turn, multiplied by the average length of a voice greeting or message determines how much storage is required, and hence the number of disk transports in a 1A vss office.

The two main parameters that must be determined to engineer a 1A vss office are (i) the offered load (erlangs) between each 1/1A ESS and 1A vss, and (ii) the total storage (disk capacity) required in the 1A vss. These are calculated from numerical arrays which contain market penetration estimates and traffic characteristics tabulated by service and customer types. The two-way trunks between 1/1A ESS and 1A vss are engineered for 0.01 probability of blocking. The disk transports are engineered for more than an order of magnitude better blocking performance than the trunks.

## **2.3 Equipment engineering**

For the No. 1/1A ESS offices, CCS II services require new dial pulse repeating/monitor trunks, additional program store and additional call

store. The increase in program store is required to hold the CCS II feature; the additional call store holds the service translations. Since additional ESS capacity is required for 1A VSS-handled calls and for advance calling delivery, some 1/1A ESS processor capacity must be allocated for these services. The CCS II services have a minimal effect on the 1/1A ESS network and service circuits.

The 1A VSS office equipment<sup>2</sup> (see Figs. 1 and 2) can be divided into three categories. The first category is comprised of a minimum basic set of common equipment required for all offices. Included in this first category are the two Auxiliary 3A Processors (labelled generically as the Central Processor in Fig. 1), main memory storage, the two peripheral controllers, three storage media controllers (SMCs) and a service circuit frame.

The second category covers major frames or units of equipment which are either traffic sensitive themselves or which must be provided to support traffic-sensitive equipment. Frames or units in this category must be provided by the supplier during the system installation, or at major growth periods. Typically, this equipment is engineered for a two-year period. Included in this second category are voice access circuit frames and units, SMCs, and uninterruptible power equipment called triports. The voice access circuit frames in this category commonly contain additional unequipped circuit pack locations to accommodate equipment in the third category.

The third category is comprised of traffic-sensitive plug-in equipment, which includes trunk access and buffer circuit modules, plus disk transports. These units are procured as required and may be installed and turned up for service by the BOC maintenance personnel. Such growth results from adding 1/1A ESS connecting offices, penetration into the potential user market anticipated in the market forecast, or a rise in actual trunk group and storage usage based upon traffic measurements. Equipment in the third category is typically engineered for a 6-month period.

#### **2.4 Traffic measurements**

Traffic measurements are provided on periodic reports referred to as C, H, Q, and D schedules. The C schedule lists measurements of trunk utilization, while the H schedule provides measurements of internal subsystem utilization. Measurements which are taken to reflect the load on internal subsystem resources include counts of disk transport, service circuit and disk subsystem usage, total calls handled, processor real time, processor main memory usage, and a record of overload control actions. Both the C and H schedules are half-hour reports. The quarter-hourly Q schedule is a subset of the H schedule that provides a quick look at a few key measures of system performance

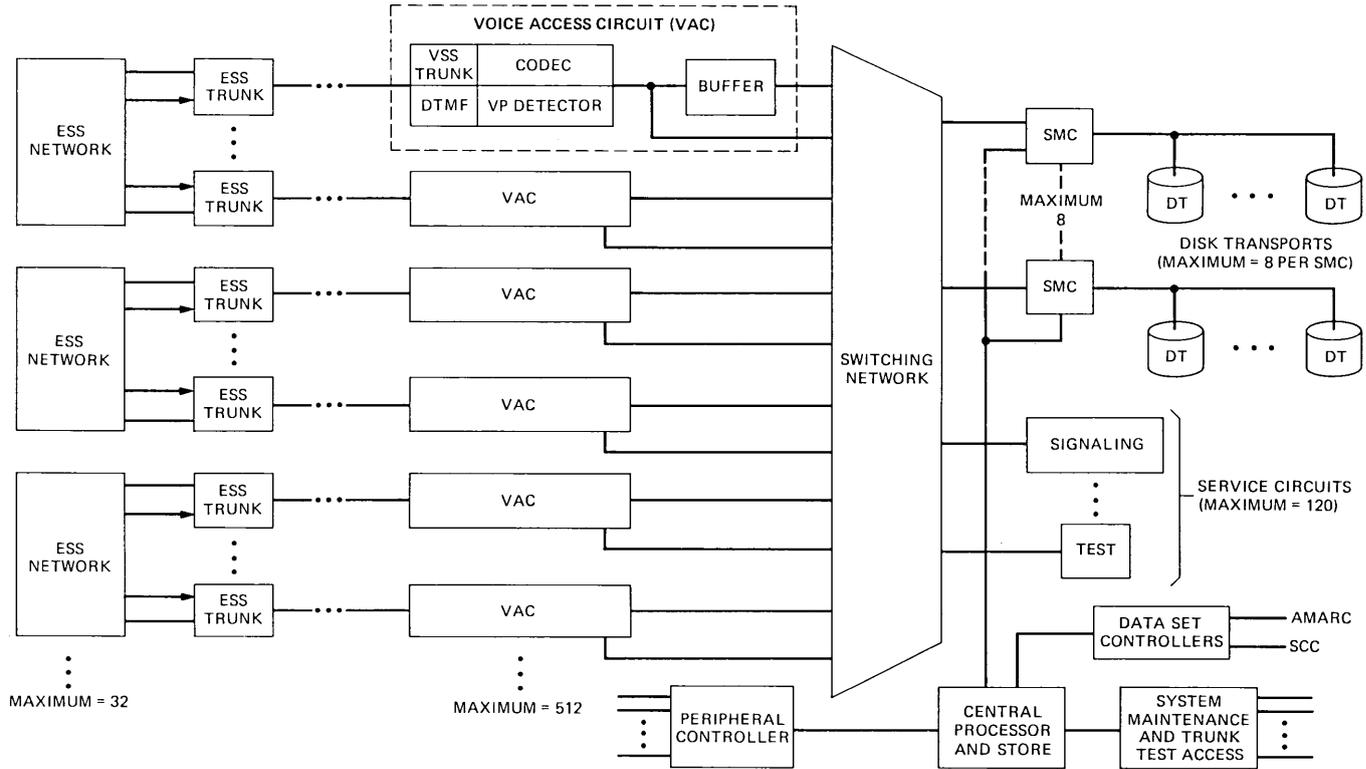


Fig. 1—No. 1A vss architecture.

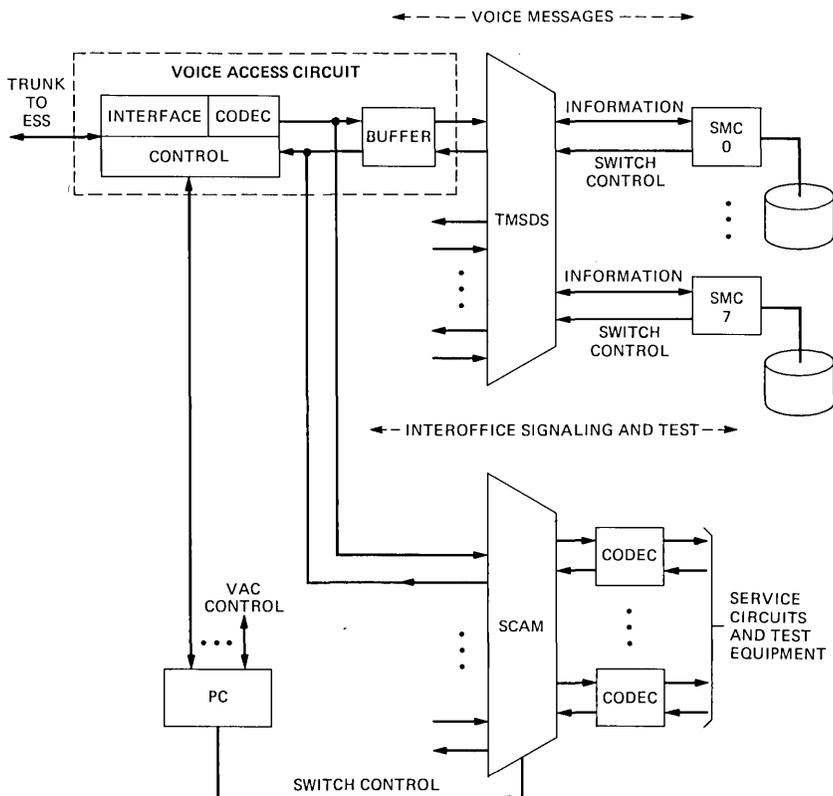


Fig. 2—No. 1A vss network partitioning.

that indicate the proper functioning of the vss office. The D schedule gives the daily totals of the C and H schedules. Also, the traffic measurements on trunks, storage, and usage are enumerated by service on the D schedule for specific half-hour periods.

### 2.5 Interface to the total network data system

The 1A vss has two independent data links which comprise the interface to the Total Network Data System (TNDS). One connection is to an Engineering and Administrative Data Acquisition System (EADAS), and the second is to the Network Administrative Center (NAC).

The EADAS polls vss for the H, C, and D schedules. Within EADAS, sets of traffic measurement outputs are triggered when preset threshold criteria are exceeded to indicate heavy traffic conditions. The EADAS also supplies the vss traffic measurements to other TNDS systems for additional traffic processing and analysis.

The NAC channel is an interface to the No. 2 Switching Control

Center (SCC) system over a dedicated facility. It is used to transmit the Q schedule. In the event of an EADAS data link or machine failure, the NAC channel serves as the backup for all traffic data (H, C, and D) in addition to the normally received Q schedule.

### **2.6 Transaction file**

The transaction file is a mechanism for the collection, storage, and retrieval of data generated by the vss application software. Raw data concerning the software activity or state at given times during a program's execution are written to the disk file. The transaction file is processed to produce a detailed engineering, service cost, and human factors characteristics of vss services. Included in this information are distributions of holding times and storage utilization. The traffic measurements give the average values of the traffic parameters, but the transaction file analysis gives the complete distributions.

## **III. RELIABILITY**

### **3.1 Overview**

Although experience with previous ESSs may be drawn upon in establishing reliability criteria for 1A vss, there are important differences. For 1A vss, the term "call" has a special meaning: In a switching machine, the primary job is to establish a path (talking connection) over which two parties may converse. It does not matter whether the parties are human or computer; in either case, once a connection is made, the burden of the information exchange is the responsibility of the parties, not the switching machine. The 1A vss, however, handles one end of the information-exchange transaction; furthermore, information flows alternately in each direction as is typical of a conversation. In the course of handling a typical call, 1A vss receives input control signals at various times, outputs system announcements in response, and either records or retrieves customer messages. This type of communication is complex. The system is continuously involved in a dynamic interaction with the customer during the entire course of the connection, and there are many opportunities for a malfunction to disrupt the call.

At the first level of analysis, the 1A vss may be viewed as two primary subsystems: (i) a large disk storage community, and (ii) an access system which handles the storage and retrieval of disk information. These two subsystems are quite different in nature: the disks are electromechanical moving-head storage devices, while the access system consists of program-controlled digital circuitry. Hence, the approaches taken to achieve acceptable system reliability must take these characteristics into account.

The first approach used, as a defense against occasional failure of

disk transports, is to create duplicate copies of stored messages on two different disks. The second approach is to engineer the remainder of the system, via which the disks are accessed, to a failure rate sufficiently low so that it does not contribute significantly to the probability of losing a message. The manner in which this is accomplished is discussed later. The net result is that the overall system reliability is presently dominated by the failure rates of commercial disk transports. The 1A vss is expected to provide a grade of service comparable to that of ordinary telephone service.

### **3.2 General design approach for reliability**

Reliable operation of 1A vss, as with other large systems, depends upon numerous design and manufacturing factors. The high probability of long-term reliable operation is built into the system at every level—from system design through hardware, firmware and software designs, carrying through quality control in the manufacturing process.

At the system level, extensive redundancy is used. Key units (e.g., the Auxiliary 3A Processor) are fully duplicated, and are provided with automatic protection switching so that a failure of either unit will still leave the system fully functional. Engineered units (e.g., SMC and disk units), whose quantity is traffic dependent, have on-line spares. The system design also encompasses power backup, communication channel redundancy, alternate external interface channels, and trunk group diversity arrangements to obviate or minimize the impact of component failures. At the hardware level,<sup>2</sup> self-checking arrangements are extensively used in the design. Cyclic redundancy check (CRC) circuitry monitors all message data at several points along the storage and retrieval path. Parity circuitry checks data transmissions along communication buses into and out of main memory and over disk storage and retrieval paths. An alarm system of ferrod scan points keeps watch over power and other hardware conditions. Firmware in the intelligent controllers [SMC and Peripheral Controller (PC)] routinely performs a number of checks to assure proper setup of switch and matrix connections, as well as validity of the transmitted data.

Furthermore, the analog path over the trunks which connect 1A vss with ESS offices is automatically checked by use of an interoffice communication scheme which uses MF signalling over these same trunks.

At the software level, 1A vss incorporates an extensive system of maintenance, diagnostic, and audit programs which can detect and locate a wide variety of hardware faults and transient errors. The maintenance system will perform appropriate system reconfigurations if necessary.

It should be noted that 1A vss reliability, and hence availability, is

influenced by the time required to repair certain faults. If one of a duplicated pair of units has failed, the system still functions normally. However, it is now vulnerable to a total outage which would occur if the mate unit should fail before the first one is repaired. The probability of this happening is proportional to the repair time. The 1A vss design incorporates a number of features which aid craft personnel to minimize repair time. These include automatic trouble-locating circuits, firmware and software, trouble-locating reference material to interpret diagnostic results, modes of manual diagnostic program execution, built-in test facilities, and a physical arrangement in which circuit boards and 95 percent of the interunit cabling are connectorized.

The quality of the 1A vss equipment itself is controlled by tests and inspections at each assembly level through the manufacturing and installation process. Basic components are either manufactured by Western Electric or purchased under rigid specification to guarantee the incoming quality. Once components are assembled onto circuit modules, these are thoroughly tested on computer-driven test facilities. When the modules are assembled into functional units and frames, these are, in turn, tested at the factory prior to shipment. Since a standard floor plan is used by 1A vss, the connectorized cables which will be used in the field to interconnect frames are included in the factory test and shipped along with the frame.

Final testing occurs at the 1A vss office site where, as part of the installation process, the frames are interconnected and operated as a system.

### **3.3 Hardware redundancies**

#### **3.3.1 Duplicated units**

On 1A vss (see Figs. 1 and 2), there are two unit types which perform common-control functions on which system operation depends.<sup>2</sup> These are the Auxiliary 3A Processor (AP) and the PC. Each is fully duplicated.

Associated with each AP is a complete set of communication buses over which it exchanges data with other units. Communication with the SMCS is via a direct-memory access (DMA) parallel channel. The even-numbered SMCS connect via one set of bus hardware, while the odd-numbered SMCS connect via the second set. This arrangement is replicated for the second AP. The *A* and *B* power buses similarly supply odd- and even-numbered SMCS. Taking this arrangement into account, the 1A vss software always places the duplicate of a message on a disk subsystem (SMC and its disks) of the opposite group (even or odd). Therefore, even if an AP should fail, and then one of the DMA channels on the good AP should also fail, the system will be able to retrieve a copy of every stored message. Since each AP also has a separate DMA

bus connection to each PC, the system can survive a failure of any PC, AP, or interconnecting parallel channel.

Another characteristic of these redundant bus arrangements is that a failed peripheral unit can always be diagnosed from the active AP. This means that the standby AP can remain at all times in the update mode; in this way, the system is better prepared to handle an AP failure as well.

Each of the PCs, which controls the per-trunk circuitry and service circuit access via the Service Circuit Access Matrix (SCAM), has its own communication channel and interface circuit in each Voice Access Circuit (VAC). Hence, failure of either a PC or its communication with a VAC can be bypassed by switching to the other PC.

### **3.3.2 Engineered units— $M$ plus $N$ sparing**

Several unit types are engineered to a quantity determined by the traffic to be handled by the office. This, of course, includes the VAC units which contain the per-trunk circuitry for up to 16 trunks, and the service circuits, such as MF transmitters and receivers. The disk subsystems (SMCs and their disks) are also engineered, although the quantities can also be influenced by other factors such as reliability considerations or traffic measurements. For example, the number of disks provided will depend on the average length of time that messages are left in the system. In like manner, the Time Multiplex Space Division Switch (TMSDS), which interconnects the trunk and the disk equipment communities, is designed in a modular fashion for convenient growth.

A failure of one of these engineered units could degrade service, but it could not take the entire 1A VSS out of service; hence, the number of spares provided is less than full duplication. The sparing philosophy used has come to be known as  $M$  plus  $N$  sparing where, for  $M$  in-service units, a number of spares  $N$  is provided where  $N < M$ . Full duplication is where  $N = M$ . An additional philosophy applied for VSS is that the spares are kept in service and in use. In this arrangement, no unit can be identified as the spare; however, should a failure occur, the number remaining in service can carry the traffic at the engineered level.

There are two main advantages to keeping spares on line: (i) they contribute to a better grade of service during the large fraction of the time when there are no failures, and (ii) they are periodically tested by both routine diagnostics and by the operational checks which go on all the time in in-service equipment.

### **3.4 Automatic trouble detection and protection switching**

As alluded to earlier, a key element in the graceful failure mechanisms of 1A VSS is the action taken by the maintenance and diagnostic

software. These programs react to either hardware or operational software indications of trouble by testing suspect units or communication paths, isolating the trouble area and reconfiguring the system so as to bypass the faulty equipment. The diagnostics are also routinely called in by the maintenance software for testing of the system each night during hours when the system is relatively idle. The maintenance software is described in Section IV.

### **3.5 Storage duplication**

Whereas 1A vss disks hold programs and data used by the processor, the bulk of the storage media is required for messages and announcements. In general, when any information is stored, it is then scheduled for duplication on another disk associated with another SMC. This arrangement allows outage of a disk or of other system components without loss of any data.

Although processor information duplication is scheduled with a high priority, analysis shows this to be unnecessary for voice messages. That is, as long as messages are duplicated within approximately one hour, the delay contributes little to the probability of lost messages since this delay is still very short relative to the mean time between failures of disk transports.

However, the advantage of the delayed-duplication philosophy, is that priority may then be given to more urgent service-affecting activity, such as the handling of calls during a temporary peak traffic period. During much of the time when the system is not heavily loaded, messages will be duplicated quickly.

## **IV. MAINTENANCE SOFTWARE**

### **4.1 Overview**

A complex maintenance software system is required to enable the 1A vss to meet the high reliability objectives described in Section III. The primary responsibility of this maintenance system is to accept error indicators from the operational and administrative software, reconfigure the 1A vss such that the suspected faulty unit is isolated, diagnose the isolated unit without affecting normal 1A vss functions, and help resolve exactly where the fault is in terms of replaceable circuit modules. The main interface with the operational and administrative software is the error control subsystem. This subsystem receives the error indicators and determines the corrective action. If the error control subsystem is unable to maintain a working configuration, the system recovery subsystem is called into action. Other parts that make up the maintenance software system are (i) The trunk maintenance subsystem, which includes the ability to perform automatic tests between the 1A vss and connecting client offices, manual

tests from either end, and automatic trunk administration functions. (ii) The routine diagnosis subsystem, which is responsible for automatically testing the entire 1A vss periodically. (iii) The power/alarm subsystem, which monitors the system's power and alarm indicators, controls the system status panel, and operates all the other audible and visual indicators. (iv) The maintenance administrative subsystem, which accumulates hourly, daily, and monthly maintenance status and reports. All of the above functions are controlled and administered by the maintenance control subsystem. This control subsystem coordinates all maintenance activities in the 1A vss, maintains the vss configuration database, initiates all diagnostic executions, and administers system reconfigurations. The remainder of this section will describe these subsystems in greater detail.

#### 4.2 Error control and error history analysis

The error control subsystem receives error indicators from operational and administrative software. It is responsible for evaluating these error inputs and determining what action to take. Figure 3 shows the sequence used to maintain a working configuration when the system experiences errors. Error messages sent to error control can be classified into the following types: (i) A device handler has lost the ability to communicate with its hardware. (ii) An error occurred but

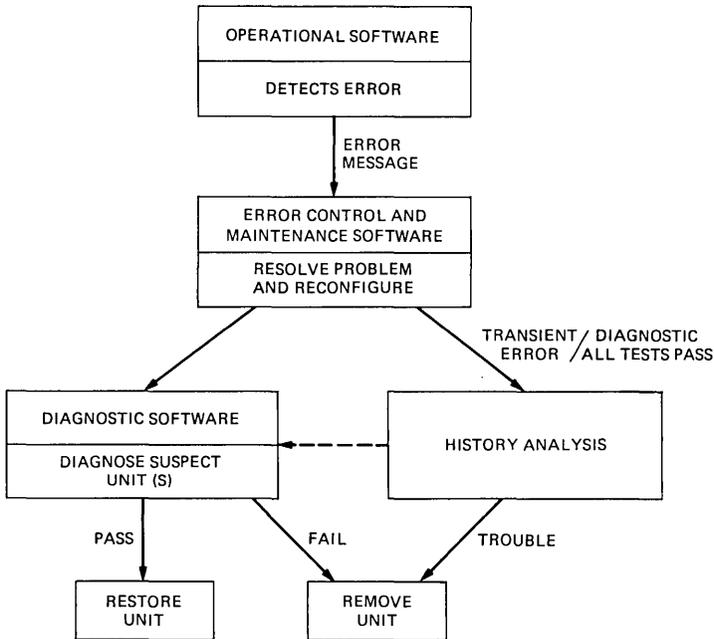


Fig. 3—Error detection/fault recovery.

the operation was successful because of a retry strategy. (iii) An error occurred and either no retry strategy was employed or it was unsuccessful.

When an error message of the first type is received, the unit involved is already out of service since fault recovery code is designed into the device handlers to cause an immediate reconfiguration if communications are lost with the unit. In this case, error control does not have to resolve the problem. The action taken is to request a conditional restore of the unit identified in the error message. The unit is already out of service so this request results in a diagnosis of the unit. If the resulting diagnosis fails, appropriate failure information used to resolve the fault is displayed on the maintenance teletypewriter (TTY) and the unit is left out of service. Manual action will then be required to restore the unit. If the unit passes diagnosis, it is put back in service and the history analysis part of error control is informed. In certain instances, it is possible for a unit to fail but pass diagnosis and be restored. To prevent an endless cycle of this type of occurrence, each time the unit is restored, history analysis is informed. If this cycle repeatedly occurs in a short time span, history analysis will order the unit to be removed from service and to be put in a trouble state. Manual action will then be required to restore the unit to service.

An error message of the second type is called a transient error. No attempt to resolve the problem is made by error control. Instead, the transient error is sent to history analysis. If many transient errors for a particular unit occur in a short time span, history analysis will cause the unit to be diagnosed by requesting a conditional restore. Since the unit is in service at this time, it will be removed from service when the restore request is received (diagnostics can be run only on an out-of-service unit). Once the unit has been removed from service, the actions taken are identical to those in the first error-type sequence, including the feedback to history analysis if the diagnostic passes.

Error messages of the third type trigger the greatest amount of activity. Some of these messages are such that a specific unit is immediately implicated. For these cases, a conditional restore of the unit is requested. Other type-three error messages do not allow error control to make an exact determination of where the fault lies. These messages result in conditional restorals of some suspect units, and implication of other units via transient error messages sent to history analysis. In the former case, the resulting diagnostic will either find a fault or exonerate the units. The action in the latter case is identical to that taken for transient error messages sent by operational software.

It is expected, in the cases where the determination is not clear, that the unit implication lists and history analysis thresholds will be optimized as field experience becomes available.

### 4.3 System recovery

The system recovery subsystem has the responsibility for recovering a working 1A vss when the system must be reinitialized (similar to phasing the ESS machines) because of processor problems, when catastrophic peripheral hardware failures occur, or when 1A vss software insanity occurs. When the system is reinitializing because of processor problems, system recovery has little control over the situation. If sufficient time does not elapse between successive reinitializations, the reinitialization level escalates. At each level, more drastic software initialization occurs until the bootstrap level is reached. Once this occurs, system recovery takes its first positive action—it triggers a memory reload of all resident memory programs and the office database.

As specified in Section III, some of the 1A vss periphery is duplicated. If a duplex failure occurs, the failure is catastrophic and 1A vss ceases to function operationally. When the duplex device handlers recognize this occurrence, they send a message to system recovery and stop functioning. System recovery will attempt to find a working combination of one of these devices and one of the processors by trying all possible combinations and by driving the initialization level higher and higher each time. Unlike the escalations caused by processor problems, if the system recovery subsystem is unsuccessful at finding a working combination, the escalations can be stopped manually by pressing the manual stop key on the system status panel. When the manual stop key is pressed, the 1A vss will stop thrashing through initializations and will settle into a quiescent nonoperational state where diagnostics can be run manually on the duplex-failed unit until the faulty component is found and repaired. When one of the units is functional, the manual stop key is released and a manual bootstrap will restart the system.

The system recovery subsystem provides an outlet for software modules (including the system recovery modules) that find themselves in untenable states from which further processing would cause further system software failures.

When this occurs, a module can send a message to system recovery which causes an initialization of the module, and perhaps all other modules depending on the level of initialization. This could cause an escalation to the memory reload bootstrap level. The manual stop key has no effect on software sanity initializations.

The last interface to system recovery is utilized when the 1A vss configuration has deteriorated below a predefined threshold. For threshold analysis, the 1A vss is divided into the storage subsystem, the service circuits subsystem, and the voice access (per trunk) circuit subsystem. When 50 percent or less of the units in any of these

categories are in service, system recovery is notified. The first notification will cause a switch of the processors which will raise the level of initialization by one. The next unit removed from service in that group will trigger conditional restores of all the out-of-service units in the affected subsystem. Further reports of units being removed from service in this category will be ignored until all the units in the category are out of service. If this occurs, all the units in the category will be unconditionally restored to service (unconditionally means that no diagnostic is run), and the threshold recovery algorithm is reset. The algorithm is also reset if sufficient time elapses between any of the levels of recovery. The manual stop key functions as described in the duplex failure case.

#### **4.4 Trunk maintenance**

The 1A vss office connects via trunks to the 1/1A ESS connecting offices. The main philosophy of the 1A vss trunk maintenance plan is that the connecting offices, regardless of type, are the controlling offices for the interconnecting trunks. This means that the 1A vss will have supporting automatic trunk test equipment, but will not have responsibility for initiating trunk facility maintenance. The connecting offices execute and evaluate end-to-end operational and transmission tests. Nevertheless, the 1A vss does contain a substantial amount of trunk maintenance software which is used for trouble detection, trouble verification, sectionalization, repair, repair verification, service protection, and new circuit installation or circuit rearrangement testing.

The 1A vss will accommodate the execution of end-to-end operational tests initiated from the connecting office. This test, which is run whenever the trunk diagnostic is executed at the connecting office, will validate the interoffice signaling capability and continuity over the transmission path, but will not test transmission quality. This will occur automatically at least on a daily basis. The 1A vss will also accommodate a Remote Office Test Line/Centralized Automatic Reporting on Trunks (ROTL/CAROT) automatic transmission test. Only terminating test equipment exists at the 1A vss; therefore, the transmission test must be initiated at the ESS connecting office end either manually or automatically through a CAROT facility. This test, which includes a verification of transmission quality, will be run automatically at least once a day.

The 1A vss provides extensive trunk service protection on its own. Whenever an internal 1A vss problem exists which affects one or more vacs, the corresponding trunks at the vss end are put in the reverse-make-busy state. A trunk is in this state when it is seized from one end, but no signaling is sent or accepted by that end. Viewed from the

opposite (ESS) end, the trunk is said to be high and wet. This will keep the connecting offices from using the affected trunks at the price of making them execute some software which deals with the high-and-wet state. When the internal 1A vss problem is cleared, the trunk circuit will be put back into a normal idle state.

All carrier groups connected to a 1A vss have hardware Carrier Group Alarms (CGAs). A CGA is reported to a trunk maintenance module which immediately removes all associated trunks from service and releases any service circuits tied to these trunks. This protection is required because a 1A vss has only a small number of service circuits which could all be occupied by a faulty carrier that causes all the associated trunks to be seized. When the CGA is cleared, all the associated trunks will be put back in service automatically.

Whenever a vAC is put back in service, an operational end-to-end test is run on the corresponding trunk from the 1A vss end. This test does not require any special software at the ESS end. It is used to determine if the trunk is high and wet (seized permanently from the ESS end), or if it is otherwise faulty so that it should be locked out at the 1A vss end. The locked-out condition occurs if the ESS does not respond to a trunk seizure from the vss end. A locked-out trunk will not be used operationally by the 1A vss but will be treated normally if seized by the connecting office. Trunks in the locked-out state will be automatically and periodically tested by the 1A vss end-to-end test. If it passes, the trunk will be restored to a normal condition. A trunk found to be high and wet will be restored to service automatically when the permanent seizure from the connecting end is dropped.

The trunk maintenance subsystem also provides a manual trunk maintenance capability from either end of the trunk. The craft personnel at the connecting end can request several test signals from the 1A vss, or can be connected to the trunk test panel at the 1A vss. By using various TTY input messages and the trunk test panel, craft personnel at 1A vss can call test signal generators or the trunk test panel at the connecting office end.

The key points of the trunk maintenance plan for the 1A vss are that the extensive trunk maintenance software, along with the error control/history analysis software described earlier, provide a powerful sectionalization tool for all trunk-related problems. Consequently, almost no end problems will require manual restorals of trunks at the end where the problem did not exist.

#### **4.5 Maintenance control**

The maintenance control subsystem orchestrates all maintenance activity in the 1A vss periphery. The processor contains its own maintenance system. The maintenance control subsystem gets re-

quests to remove units from service, to diagnose units, restore units to service, and to switch duplex units. These requests come from the error control, system recovery, trunk maintenance and routine diagnosis subsystems, and from manual inputs via the TTY from craft personnel. Maintenance control is responsible for prioritizing these requests and ensuring that no interference occurs between concurrent maintenance activities.

Requests to remove units from service will be evaluated to determine the effect of this action. Removing a unit from service may affect several other units, or it could trigger particular alarm conditions or even system recovery action. Maintenance control must make these determinations and take appropriate action. One example of an interactive condition is the removal of an SMC from service, which will necessitate the removal of all disks connected to it, along with its associated second-stage switch (one of the modules in the two-stage TMSDS shown in Fig. 2). All three of these unit types are grouped into a family, and anything that affects one member of the family is evaluated to determine its affect on other family members. There are several other family groupings in the 1A vss.

Another situation arises when the removal from service of an SMC would leave the system below a critical minimum number of operational SMCS (as detailed in the system recovery section). In this case, a routine remove request would be denied.

Requests to diagnose units will cause these units to be removed from service if they are active, and then helper units will be selected as required before the request is passed along to the diagnostic control subsystem. Requests to restore a unit to service will first cause it to be removed from service if required, and then diagnosed. If the unit passes the diagnostic tests, it will be initialized and restored. A unit can also be unconditionally restored which will cause it to be initialized and put back into service without being diagnosed. When a unit is restored, a family evaluation will take place. For example, this occurs when all the disks connected to a particular voice message controller are taken out of service; then the storage media controller and its associated second stage switch are also taken out of service but are put in a nonfault stage. When any one of the disks is restored to service, the associated SMC and second-stage switch are also restored automatically.

#### **4.6 Diagnostics**

The 1A vss peripheral diagnostic programs are used to detect faults in their respective units; the resulting failure data are then used to identify any of the replaceable circuit modules which are faulty. A table-driven diagnostic design is used, and a high-level-macro approach

facilitates the diagnostic development. The same diagnostics are used for frame testing in the factory, for installation testing at the field site prior to cutover, and for on-line testing programs, while the system is operational. The diagnostics are triggered either manually, by craft personnel using a TTY, or automatically as was discussed earlier. The manually initiated diagnostics may be originated from either the on-site maintenance TTY or remotely from a Switching Control Center (SCC). The trouble location capability is an integral part of the diagnostics and will be described later. In general, the location capability is at its maximum when a unit contains a single fault; if it contains multiple faults, the location resolution is reduced.

Each unit diagnostic is a collection of macros used to test the unit. These macros expand into data table words when assembled. Each macro expands into an OP-CODE or INDEX word, plus one or more data words. These data table words are grouped into one or more segments each containing less than 2K words. When a diagnostic is triggered automatically, all the segments in a particular unit diagnostic are run on the unit under test. If the diagnostic is manually triggered, all segments or selected segments can be run. A diagnostic segment can also run in an interactive mode where execution proceeds to a selected point in a segment, or proceeds from a selected point to another, or loops over one or more tests.

Figure 4 shows the structure of the diagnostics. The data tables reside in auxiliary memory in the 1A vss disks. When executed, the segments are overlayed, one segment at a time, into a 2K paging buffer in main memory and are used to drive the diagnostic control program which resides in main memory. A simple model of this control program is included in Fig. 4. After overlaying a segment of data tables, a task dispenser examines the first data table word and extracts the OP-CODE or INDEX information. This is used to pass control to the appropriate task routine. Each macro type has a corresponding task routine which fetches the data words following the OP-CODE or INDEX word and executes accordingly. Most of the task routines build commands for the 1A vss peripheral devices and then send these commands to the appropriate device handlers. These handlers send the commands to the devices and they in turn cause the execution of one or more device-resident firmware routines. The result of this execution is sent back to the task routine via the device handler. If this sequence was triggered by a test-type macro, the task routine will compare the results with an expected result (supplied in the data words of the macro expansion) and will determine whether the test passed or failed.

Of special interest is the fact that the diagnostic structure resides in three different memory media. The data tables are in auxiliary mem-

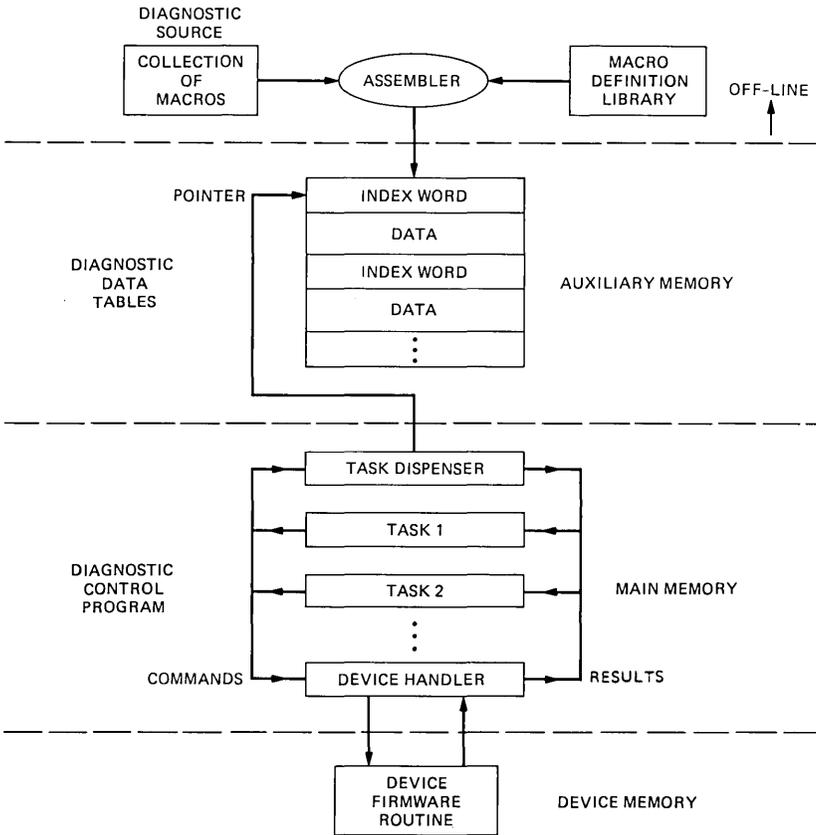


Fig. 4—Diagnostic design.

ory, the control is in main memory, and the work routines are in device memory (firmware). This table-driven, high-level-macro approach to designing diagnostics has been proven very successful in other ESS developments.<sup>3</sup> It has permitted the diagnosticians to concentrate efficiently on the functional requirements of the hardware.

#### 4.7 Trouble location

Trouble location uses all available information to deduce where a fault exists in the system. Once the resolution is made, the fault is corrected and the out-of-service unit is restored to service. Trouble location may require analyzing error conditions, plant measurements, maintenance reports, and any other indications of atypical system behavior. No attempt will be made here to describe all possible approaches. However, the great majority of all faults will be resolved by the unit diagnostics.

The trouble location information is built into the diagnostic source.

A trouble location macro follows each test-type macro in a diagnostic segment. This macro requires the diagnostician to comment on what the test was checking for, to list the suspected replaceable circuit modules in a prioritized order, and to specify any special instructions required for the repair process. When the diagnostic source is assembled into data tables, these macros are expanded into readable repair information in the diagnostic listing. The listings are processed off-line where the trouble location information is extracted and used to form a separate Trouble Location Manual (TLM). When a diagnostic detects a test failure, it terminates at that point. Craft personnel can use the test number and segment number to find the appropriate diagnostic listing or the TLM where the trouble location information will be found.

#### **4.8 Routine diagnosis**

Periodically, the routine diagnosis subsystem will test the 1A vss in an attempt to provide automatic preventive maintenance. Periodic testing will detect a fault soon after it occurs, which reduces the probability of the occurrence of a second fault before the first one is repaired. This is desirable since, as previously mentioned, the power of the diagnostics is reduced if the unit under test contains multiple faults. Daily routine diagnosis will attempt to remove, diagnose, and restore all the 1A vss units that are in service. If the diagnostic fails, the unit will be left out of service.

There are several special considerations that the routine diagnosis subsystem must observe. It will run only during the intervals when the system has little or no customer load. It will cause the processors to switch only after first triggering a diagnosis of the processor that was the standby. It must take hardware family associations into account to minimize the effect on the rest of the system.

Routine diagnosis also has the responsibility for performing certain test sequences that are not executed when the diagnostics are run as previously described. These test sequences can be run manually if the appropriate diagnostic parameters are used, but they seldom will be executed this way because of their long execution times. Two examples of such sequences are the multiple path select tests in the switch and matrix diagnostics, and the complete media test and initialization portion of the disk diagnostic. These sequences will be run periodically, but not on a daily basis.

#### **4.9 Power/alarm control**

The power/alarm subsystem monitors the system's power and alarm indicators (with the exception of the carrier group alarms described in Section 4.4). Examples of these are unit power indicators, bus power alarms, and building alarms. The power/alarm software also provides routines for other maintenance software to trigger minor, major, and

critical alarms in the system. These power and alarm inputs will result in appropriate TTY output messages, audible and visual indications, and corresponding indications on the System Status Panel (SSP). All of these alarms will be sent to the connecting SCC system using another port on the maintenance TTY channel and telemetry. An interesting feature of the 1A VSS maintenance philosophy is that the power and alarm indicators which provide input to the power/alarm subsystem will not trigger any maintenance activity other than the indicators described above. The reasoning behind this is that all such power/alarm problems will result in operational problems with the unit or units involved and will be handled by the error control subsystem.

Most 1A VSS peripheral units have power and status indicators built into the power switch module visible on the unit's front panel. These indicators show if the unit is operating normally, is out of service, is powered down or is in a power alarm state. The indicators are controlled by the power/alarm subsystem. The status of the 1A VSS units is also displayed on the SSP. If all the members of a particular unit type (e.g., SMCs) are in service, the corresponding light on the SSP is extinguished. If one or more units are out of service, the light is lit. Minor, major, or critical alarms result in output messages describing the problem, appropriate indicators on the SSP, audible alarms and aisle pilot lights indicating which aisle or equipment is experiencing the problem. If the 1A VSS does not have resident craft personnel, the audible alarms can be shut down at the site and monitored only via remote connections to the SCC system.

#### **4.10 Administrative maintenance function**

A great deal of effort was put into the man-machine-interface design of the administrative maintenance capability. This capability consists of:

(i) Reporting features which allow the craft personnel to ask for the maintenance status of various system components via TTY input messages.

(ii) Automatic outputs of maintenance information on an hourly basis.

(iii) Automatic outputs of detailed plant measurement data on a daily basis.

(iv) Automatic outputs of a summary of the plant measurement information on a daily basis (used by the management in charge of the system).

(v) Monthly measurements and summary records.

The hourly maintenance information can be tailored by the individual system managers. The default case is to report all available information. The entire report or selected categories can be printed

out or inhibited as desired. Examples of items comprising this report are a list of units out of service, entire trunk groups that are inoperative, and carrier facilities in the alarm state. The detailed plant measurements show which units had transient error conditions, which were automatically removed from service, which were found faulty as a result of the automatic removal, and the measurements also show the length of time each remained out of service. Other plant measurements indicate the frequency of system reinitializations and the levels at which the reinitializations occurred, and provide a calculation of the message reliability based on the Mean Time to Repair (MTTR) and the Mean Time Between Failure (MTBF) of the storage subsystem.

## V. INSTALLATION TESTING AND ACCEPTANCE TESTING

### 5.1 Overview

As the final phase of the installation, a new vss is put through an extensive series of tests to verify proper operation. Most of these tests result in TRV printouts of system actions that can serve as a permanent record of test results. In addition, a high-temperature test may be run at the option of the BOC. The BOC personnel either participate in these tests or subsequently review the results. At this point, the system is turned over to the BOC for final acceptance testing prior to cutting the system into service.

Acceptance tests are run by a BOC to assure itself of the proper operation of a newly purchased system. It is highly likely that no two vsss will ever be exactly alike in terms of installed office configuration, trunking arrangements, connecting office number and types, and the mix of customers served by these offices. Nevertheless, the BOC has extensive standard test documentation and the expertise of the Western Electric installation organization to draw upon in tailoring testing to the specific vss office configuration.

Acceptance tests may include:

- (i) an audit of the installation tests,
- (ii) tests which check the interaction of the 1A vss with other systems to which it connects (e.g., ESS client offices and operational support systems).

The installation tests are described in Section 5.3. In these tests, the system's ability to handle a heavy calling volume is verified by the application of a simulated traffic load. The equipment which generates these test calls is described next.

### 5.2 Call simulation equipment

The equipment used to provide a simulated traffic load to 1A vss is different than that used in ESS offices because the requirements of 1A vss are unique. The processor on either an Electronic Switching

System (ESS) or on 1A vss can be presented a load by generating a large number of short-holding-time calls with computer-driven test facilities. In the ESS case, lengthening the holding time on calls would additionally load down the network. This is not generally done—it is considered unnecessary because networks are well understood and can be engineered to a highly predictable level of performance. Long calls would, incidentally, also load down the test facility; a much larger computer would be required to manage the much greater number of calls concurrently extant within the switching system.

For 1A vss, both a high calling rate and long call holding times are required. It is true that the processor, which handles the call setup and tear-down, is loaded by a high calling rate. However, unlike the situation in an ESS, the 1A vss processor also has work to do during the course of a call to handle the storage and retrieval of messages. Furthermore, the SMCS and disks are loaded most heavily by lengthy messages which must be stored and retrieved.

The corresponding throughput capacity requirement on the call simulation equipment is large. This is handled by a number of microprocessor-driven call simulators which provide a distributed load. In each VAC unit (of which there are two in each VAC frame) there is a normally unoccupied circuit module location having backplane wires to each Trunk Access Circuit (TAC). A call-simulator board may be plugged into this location to provide a call load on any or all of the 16-trunk circuits in this VAC. Simulated traffic is applied to selected trunks by the insertion of a strap plug (in lieu of an actual trunk connection) for each such trunk on the VAC backplane. Each call simulator can generate traffic autonomously, or can be monitored and controlled from a common facility.

The call simulators are capable of generating any or all of five call types associated with the Call Answering service.<sup>1</sup> They produce messages comprised of a pulsating tone that is encoded to contain a check number; message length may be set from 10 to 80 seconds in 10-second increments. Front panel switches (see Fig. 5) or external control may select the call type and message length parameters. Additionally, the E&M lead signalling is handled as is the generation of multifrequency (MF) digit strings which contain call type and directory number information for the vss. Retrieved messages are decoded and matched for a correct check number. Counts of total calls and of errors in each of five categories are recorded by the microprocessor. They are also selectively displayed on a three (hexadecimal) digit read-out on the front panel.

At the shortest (10-second) message length, a call simulator is capable of generating in the vicinity of 160 calls/hour/trunk depending, of course, on the response time of the vss. This substantially exceeds

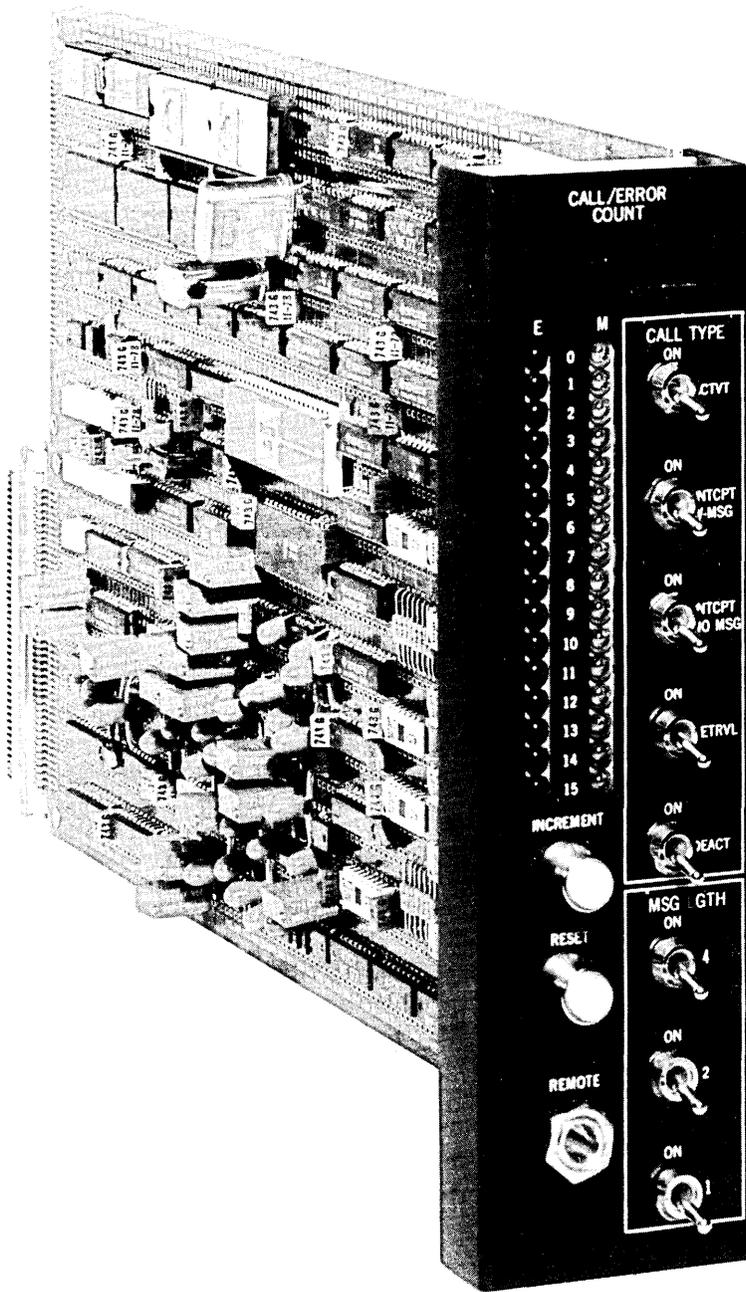


Fig. 5—Call simulator.

anticipated per-trunk calling rates which are predicated on longer call holding times. Therefore, in doing volume tests, message lengths are chosen appropriately to obtain the calling rate for which an office has been engineered.

A jack on the front panel provides a 300-baud RS 232 communication interface to a common monitor/control unit. This unit contains a minicomputer controller and a teletype for output. A maximum of 32 call simulators, each located in a VAC unit, would be concurrently connected to this monitor/control unit in a maximum-size 1A VSS. In the normal monitor mode, a full report is issued every 15 minutes. Others modes are available for hourly, daily, or on-demand reports. For each active trunk in the system, a printed line is produced that contains counts of total calls and of each of the five error types. Control and monitoring of all call simulators from this central point greatly facilitates the administration and recording of the integrated and maintenance volume tests, as discussed in the next section.

### **5.3 Installation tests**

Various final checks performed by Western Electric installation are discussed in the following:

(i) Idle System Tests are those in which the system is caused to sequence through the routine diagnostics which are normally run once a day. No traffic load is applied. This test verifies that all units and subsystems are operating properly.

(ii) Integrated Volume Tests involve a simulated traffic load being applied to the system. In this test, the 1A VSS is required to carry the level of traffic (calls per hour) for which it was engineered, for a 24-hour period, during which full grade-of-service requirements must be met.

(iii) Maintenance Volume Tests verify the system's ability to recover properly from hardware troubles and software initializations. Simulated traffic is applied during the test. The grade of service delivered by the 1A VSS during induced disturbances is monitored by the call simulation equipment and printed out on an associated TRV. The printout provides per-trunk counts of the number of calls handled properly, and of call-handling errors in different categories. The number of such errors, if any, which are allowed depends upon the severity of the disruption induced in the system. For example, the manually initiated removal from and restoral to service of any units in the 1A VSS should be accomplished without perturbation of call-handling activity. On the other hand, the disabling of an active controller will abort a call being set up at that instant, while a major software reinitialization could suspend call handling for a matter of seconds. A failure of commercial ac power is induced to verify the system's ability

to transfer smoothly to battery-backup power. In general, the system design is such that the voltages supplied to vss operational equipment are not disturbed by such a failure.

Also, various additional tests are run to cover interfaces or test equipment not checked by diagnostics (e.g., trunk test facilities).

#### **5.4 Acceptance tests**

During the interval between turnover and cutover of a system, the BOC performs acceptance tests to verify that the system will perform as engineered. Aside from the optional repetition of any of the installation checks, these tests deal primarily with interfaces to other systems. For vss this includes the connection to the automatic message accounting recording center (AMARC) system, which receives billing information; to EADAS (discussed in Section II); which collects traffic data; and to the SCC for maintenance monitoring. However, the largest interface is made up of the trunk groups, which connect to ESS client offices.

A complete test of vss features is made at the option of the BOC. Since the presently planned services and features can be thoroughly tested over the interconnecting trunk groups by several hundred phone calls, the development of special field test facilities has not been warranted. These intersystem tests are spelled out in a "script" which defines the specific actions (and their timing) to be taken in placing each call. A grouping of test telephones has been defined, each of which is provided with specific ESS features (e.g., one- or two-digit speed calling, call forwarding, call waiting, etc.). Appropriate calls placed on these phones can then verify that the combination of the ESS with its software, and the 1A vss with its software, are interacting correctly so as to produce the expected announcements or other responses.

## **VI. SUMMARY**

A conventional telephone switching office mediates communication between two parties by setting up a connection over which they may converse. Unlike a switching office, the 1A Voice Storage System (1A vss) takes on the role of one of the conversing parties. In the course of handling a typical call, 1A vss receives input control signals at various times, outputs system announcements in response, and either records or retrieves customer messages. Since this type of communication is complex, offering many opportunities for a system malfunction to disrupt a call, the 1A vss design incorporates a large number of failure-defense strategies.

The quality of service is of utmost importance. This article has presented the underlying philosophies and the approaches used to

obtain this quality. Several contributing factors which have been covered are adequate office engineering to assure sufficient equipment to handle peak traffic, high intrinsic reliability based upon system redundancies and defensive software strategies, and an effective maintenance software system to minimize the effect of failures on system service. The incorporation of these system features has provided a robust 1A vss system which is flexible and reliable.

## VII. ACKNOWLEDGMENTS

The work described in this article could not have been accomplished without the combined efforts of numerous system designers, as well as the contributions of the Western Electric manufacturing and installation organizations.

## REFERENCES

1. D. P. Worrall, "1A Voice Storage System: New Custom Calling Services," B.S.T.J., this issue.
2. R. G. Cornell and J. V. Smith, "1A Voice Storage System: VSS Hardware Architecture," B.S.T.J., this issue.
3. P. W. Bowman et al., "1A Processor: Maintenance Software," B.S.T.J., 56, No. 2 (February 1977), pp. 264-71.



## ACRONYMS AND ABBREVIATIONS FOR VOICE STORAGE SYSTEM

AC	advance calling
ADM	adaptive delta modulation
AGC	automatic gain control
AMARC	automatic message accounting recording center
AP	auxiliary 3A processor
ATAE	Associated Telephone Answering Exchanges, Inc.
BOC	Bell Operating Company
CA	call answering
CAROT/ROTL	centralized automatic reporting on trunks/remote office test line
CAS	custom announcement service
CCD	charge-coupled device
CCS II	custom calling services II
CF	call forwarding
CGA	carrier group alarm
CODEC	coder/decoder
CRC	cyclic redundancy check (or code)
CSD	customer specified delivery
CW	call waiting
DCA	daily call answering
DMA	direct memory access
DN	directory number
DT	disk transport
DTMF	dual tone multifrequency
EADAS	engineering and administrative data acquisition system
EOS	extended operating system
ESS	electronic switching system
FCC	Federal Communication Commission
FIFO	first-in-first-out
FSK	frequency shift keying
IOP	input/output processor
LSI	large-scale integration
MCA	monthly call answering
MF	multifrequency
MISC.	miscellaneous
MTBF	mean time between failures
MTC	maintenance frame
MTTR	mean time to repair
MWR	message waiting ring

MWT	message waiting tone
NAC	network administrative center
OSS	operations support system
PC	privacy code
PCC	privacy code change
PCD	power control and distribution frame
POTS	"plain old telephone service"
PUC	Public Utility Commission
RA	remote access
RAM	random-access memory
RC	resistance-capacitance
SCAM	service circuit access matrix
SCC	switching control center
SMC	storage media controller
SO	service order
SO TTY	service order teletypewriter
SPC	stored program control
SRC	service circuit
SSD	service specified delivery
SSP	system status panel
STC	state table controller
T	TRIPORT cabinet
TAC	trunk access circuit
TLM	trouble location manual
TMSDS	time multiplexed space division switch
TNDS	total network data system
TSPS	traffic service position system
TST	test frame
TTY	teletypewriter
UPS	uninterruptible power supply
VAC	voice access circuit
VP	voice presence
VSS	voice storage system
VM	voice manager
VU	volume unit

## CONTRIBUTORS TO THIS ISSUE

**Glenn D. Bergland**, B.S.E.E., 1962, M.S.E.E., 1964, Ph.D. 1966, Iowa State University; Bell Laboratories, 1966—. Mr. Bergland began working at Bell Laboratories in the Military Systems Research area. He did early work in the discovery, application, and hardware implementation of several new Fast Fourier Transform (FFT) algorithms which were applied to speech and signal processing. From 1968 to 1971, he supervised a group doing research in the design and application of highly parallel computer architectures and became project engineer for the final development of the Parallel Element Processing Ensemble (PEPE) system. In 1972, Mr. Bergland became head of the Advanced Switching Architecture Department in Naperville, Illinois, where he proposed the initial Voice Storage System (vss) concept. In 1974, he became head of the Software Systems Department, where he started the vss feature development for the No. 1 Electronic Switching System (ESS). Since 1977, he has been head of the Digital Systems Research Department in Murray Hill, New Jersey. His principal research areas are in software design methodologies, telecommunications terminals, and telematics services. Member, IEEE, ACM, Sigma Xi, Eta Kappa Nu, Phi Kappa Phi, Tau Beta Pi, AAAS. Honorable mention Eta Kappa Nu Outstanding Young Electrical Engineer.

**Peter W. Bowman**, B.S.A.M.P., 1968, University of Wisconsin; M.S., 1969, University of California-Berkeley; Bell Laboratories, 1968—. Mr. Bowman has worked on specification and diagnostic design for the 1A Processor auxiliary data system. He designed a performance and evaluation system to be employed at the field trial for the High Capacity Mobile Telephone System—now known as Advanced Mobile Phone Service. He had responsibility for testing and integrating the 1A Processor diagnostics into the initial No. 4 ESS generic. He had design responsibility for the diagnostic, fault recognition and recovery software for the Voice Storage System and supervised the implementation phase. He supervised the Voice Storage System integration and system test group during the final development phase of the system. Mr. Bowman currently supervises the Advanced Mobile Phone System-Cell Operational Software group. This group is responsible for call processing and operational software.

**Ronald G. Cornell**, B.S.E.E., 1969, M.S.E.E., 1971, Ph.D. (Electrical Engineering), Cornell University; Bell Laboratories, 1973—. Upon joining Bell Laboratories, Mr. Cornell worked on multiprocessor architecture and digital signal processing features for ESS. From 1974 to

1976, he helped develop the input/output channel system for the Auxiliary 3A Processor. In 1976, he became supervisor of the Voice Storage System Group, responsible for architecture and circuit design of the 1A Voice Storage System. From 1978 through 1980, Mr. Cornell was responsible for exploratory studies into new service capabilities made possible through digital local switching systems. He is currently Head, Advanced Mobile Phone Service (AMPS) Software Design Department, responsible for software development associated with the AMPS system. Member, IEEE, Eta Kappa Nu.

**Paul E. Fleischer**, B.E.E., 1955, M.E.E. 1956, Dr. Eng. Sc. (E.E.), 1961, New York University; Instructor, New York University, 1956-1961; Bell Laboratories, 1961—. Mr. Fleischer's earlier work was concerned with the application of analog, digital, and hybrid computation to the design and optimization of electric networks. More recently his work has been in active filters and equalizers. During the past few years he has been engaged in the design of switched capacitor filters, as well as other integrated circuits. Member, IEEE, Eta Kappa Nu, Tau Beta Pi, Sigma XI.

**Osamu Fujimura**, B.S. (Physics), 1952, D. Sc., 1962, University of Tokyo; Kobayashi Institute of Physical Research, 1952-1958; University of Electrocommunications, Tokyo, 1958, 1961, 1965; M.I.T., 1958-1961; Royal Institute of Technology, Stockholm, 1963-1964, 1965; University of Tokyo, 1965-1969, 1973; Bell Laboratories, 1964-1965, 1973-1976. Mr. Fujimura has been engaged in speech research at Kobayashi Institute of Physical Research at M. I. T., and at the Royal Institute of Technology. At the University of Electrocommunications, he served as Assistant Professor, and at the University of Tokyo, he was Professor at the Research Institute of Logopedics and Phoniatics; later, he was appointed Director of the Institute. At Bell Laboratories, Mr. Fujimura served as consultant in speech research and later became Head, Linguistics and Speech Analysis Department.

**Apparajan Ganesan**, B. Tech. (Electrical Engineering), 1975, Indian Institute of Technology, Madras; M.S.E.E., 1976, State University of New York; Manager, Computer Aided Design, General Instruments Corporation, 1977-1980; Bell Laboratories, 1980-1981. At Bell Laboratories, Mr. Ganesan worked on the application of integrated circuits to telecommunications. Member, IEEE.

**Geoffrey W. Gates**, B.S.E.E., 1967, M.S.E.E., 1968, Ph.D. (Computer Science), 1973, Michigan State University; Bell Laboratories,

1973–1980. Early in his career at Bell Laboratories, Mr. Gates was involved in exploratory switching studies. He subsequently worked on the extended operating system and auxiliary 3A processor development. From 1977 to 1979, he was supervisor of the VSS Software Architecture Design Group.

**Walter T. Hartwell**, B.S.E.E., 1961, M.S.E.E., 1968 Newark College of Engineering; Bell Laboratories 1956–1959, 1961. Mr. Hartwell initially worked on the exploratory development of data modems including FSK, SSB, and Quadrature PSK from 1956 to 1959. From 1961 to 1962, he worked on the Earth Station's Antenna Control equipment for project TELSTAR. From 1963 to 1971, he worked in military digital systems research with efforts in hybrid computation, underwater sound signal analysis and detection, complex signal analysis, picket fence reduction, the first FFT processing system, and the Parallel Element Processing Ensemble (PEPE) system. Since 1971, he has been engaged in Telecommunications Switching and Service systems development. He proposed the architecture and did early economic studies of the vss system. In 1977, he became a supervisor in the Local Switching Exploratory Studies Department with responsibilities in the areas of storage systems and services, network access, signal processing and low-bit rate speech. He is currently a supervisor in the Interactive Voice Systems Department. Mr. Hartwell holds patents in the areas of control systems, signal processing, and system architectures. Member IEEE, Eta Kappa Nu, Tau Beta Pi.

**Harry Heffes**, B.E.E., 1962, City College of New York; M.E.E., 1964, Ph.D., 1968, New York University; Bell Laboratories, 1962—. Mr. Heffes has worked in the areas of control and filtering theory. More recently, he has been concerned with modeling and analysis of teletraffic and computer systems. He is currently Adjunct Associate Professor of Electrical Engineering and Computer Science at Stevens Institute of Technology. Member, Tau Beta Pi, Eta Kappa Nu, American Men of Science, ORSA.

**James C. Kennedy**, B.S.E.S., 1967, Pennsylvania State University; M.S.E.E., 1968, Stanford University; Ph.D. (Electrical Engineering), 1971; Bell Laboratories, 1967—. Mr. Kennedy first worked on the planning and design of cost-reduced No. 1 ESS periphery, including the evaluation of alternative scan elements and control schemes. He evaluated hardware architectures to implement new No. 1 ESS switching features. As a supervisor from 1976 to 1980, Mr. Kennedy was responsible for the following vss activities during different intervals: system

design, interface specifications, test utilities, program administration, testing, traffic analysis, administration software, and maintenance software. From March 1980 to February 1981, he was supervisor of the No. 5 ESS Message Switch Group. Since February 1981, he has been responsible for the evolution of No. 5 ESS. Member Phi Beta Kappa, Tau Beta Pi, Sigma Xi, Pi Mu Epsilon.

**Richard F. Kranzmann**, B.S. (Electrical Engineering), 1960, Union College; M.S. (Electrical Engineering), 1962, New York University; Bell Laboratories, 1960—. Since joining Bell Laboratories, Mr. Kranzmann has worked on software design for several switching systems including the UNICOM (Universal Integrated Communications) system, AUTOVON (Automatic Voice Network), and No. 4 ESS. After serving as supervisor of the Indian Hill Technology Education Group from 1967 to 1969, he worked on 1A Processor peripheral diagnostics, and later managed the team responsible for designing the 1A Processor automatic trouble locating system. He joined the vss project in 1976 with responsibility for the interface of the 1A vss to the No. 1 and 1A ESS host offices, and since 1979 has supervised the design of fundamental system software necessary to support Common Channel Inter-office Signaling in No. 1 and 1A ESS.

**Kenneth R. Laker**, B.S.E.E., 1968, Manhattan College, M.S.E.E., 1970, Ph.D. (E.E.), 1973, New York University; Captain, USAF, Air Force Cambridge Research Laboratories, 1973–1977; Bell Laboratories, 1977—. Mr. Laker has worked in the areas of surface acoustic wave devices and active networks. Since joining Bell Laboratories, he has been engaged in various development and exploratory activities associated with active-RC and switched capacitor filters. He is presently supervisor of the Signal Processing Subsystems Group, concerned with codecs, filters, and signaling subsystems. Mr. Laker is co-author of *Modern Filter Design: Active-RC and Switched Capacitor*, Prentice-Hall, 1981, and co-editor of *Modern Active Filter Design*, IEEE Press, 1981. Member, Eta Kappa Nu, Sigma Xi, Senior Member, IEEE, President-elect, IEEE Circuits and Systems Society.

**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 com-

puter networks, and computer performance evaluation. He is Head, Mathematics of Physics and Networks Department.

**Joan E. Miller**, B.A. (Mathematics), 1953, Mount Holyoke College; M.A., 1956, Indiana University; Ph.D., 1971, Columbia University; Bell Laboratories, 1957—. At Bell Laboratories, Miss Miller's research activities have focused on speech analysis, musical acoustics, computer text editing, and graphics. Member, Acoustical Society of America.

**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. He is a supervisor in the Mathematics of Physics and Networks Department. Member, IEEE, SIAM.

**Eric Nussbaum**, B.S. (Electrical Engineering), 1955; M.S. (Electrical Engineering), 1956, Columbia University; Bell Laboratories, 1959—. Since joining Bell Laboratories, Mr. Nussbaum has been active in various development and exploratory areas of switching involving hardware, software, and systems. He is currently Director of the New Switching Services Laboratory with responsibilities for advanced studies of new types of telecommunications systems and services encompassing voice, data, and video. Member IEEE, Communications Society, Tau Beta Pi, Eta Kappa Nu.

**Peter Pirsch**, Dipl. Ing., 1973, Dr. Ing., 1979, University of Hannover, Hannover, West Germany; Telefunken, Hannover, Television Department, 1966-1973; University of Hannover, 1973—; Bell Laboratories, 1979-1980, and summer 1981. At Bell Laboratories, Mr. Pirsch was engaged in DPCM encoding of TV signals, and he also worked on various aspects of picture coding and signal processing.

**George W. Smith, Jr.**, B.S.E.E., 1952, North Carolina State University; M.S.E.E., 1958, Stevens Institute of Technology; M.A., 1961, Ph.D. (Electrical Engineering), 1963, Princeton University; Bell Lab-

oratories, 1952—. Early in his career at Bell Laboratories, Mr. Smith's activities concerned work on military systems such as the Nike-Zeus and UNICOM. From 1963 to 1968, he was supervisor of No. 1 ESS ADF System Design and Coordination. In recent years, he has served as Head of various departments, namely, No. 1 ESS Call Program Design; Design Automation Department; Auxiliary Processor Systems Department; Voice Storage Systems Development Department; and the Advanced Switching Technology Department. Currently, he is Head, Information Network Development Department. Member, IEEE, Tau Beta Pi, Eta Kappa Nu, Sigma Xi.

**Jay V. Smith, B.E.E.,** 1959, University of Dayton; M.S.E.E., 1962, Ohio State University; Bell Laboratories, 1959—. Mr. Smith's experience at Bell Laboratories has included work on No. 5 Crossbar, No. 3 ESS, No. 2 ESS, and the Voice Storage System. Currently, he is supervisor of the Local Data Transport (LDT) Project Control Group.

**Gilbert A. Van Dine, B.S.E.E.,** 1956, Purdue University; Bell Laboratories, 1956—. Mr. Van Dine initially worked on digital circuit designs for NIKE missile systems, the UNICOM, and the No. 1 ESS ADF systems. He later worked on CRT display techniques. Since 1967, he has supervised the development of test facilities for 1A Processor frames, and the utility system used for monitoring and control of the 1A Processor in the system laboratory environment. Just prior to his work on the Voice Storage System, he supervised a group responsible for No. 4 ESS reliability studies.

**Lonnie D. Whitehead, B.A. (Psychology),** 1958, M.A. (Experimental Psychology) 1964, Emory University; M.S. (Computer Science), 1964, Stevens Institute of Technology; Bell Laboratories, 1965—. After joining Bell Laboratories, Mr. Whitehead worked on MULTICS, an experimental time-sharing system developed jointly by MIT, General Electric, and Bell Laboratories. In 1968, he became supervisor of a group which developed a computer operating system for an experimental radar project. From 1969 to 1974, Mr. Whitehead worked on real-time operating systems for naval applications. From 1974 to 1975, he worked on the No. 1 ESS, and from 1976 to 1981, he was supervisor of the Voice Storage System Feature Software Design Group. Mr. Whitehead assumed his current position as supervisor of the Voice Systems Architecture Group in July, 1981. Member, Association for Computing Machinery, IEEE Computer Society.

**David P. Worrall**, B.S.E.E., 1963, Penn State University; M.S.E.E., 1965, New York University; Bell Laboratories, 1963—. After joining Bell Laboratories, Mr. Worrall worked on the SAFEGUARD System, designing the prelaunch radar-to-missile communications link and the prelaunch microwave link for the remote launch configuration. In 1974, he joined the Customer Services Planning Department and later was appointed supervisor of the Stored Voice Feature Planning Group with responsibility for systems engineering for the 1A Voice Storage System. In 1981, Mr. Worrall assumed his current position as supervisor of the Direct Service Dialing Feature Planning Group, which is responsible for system engineering, planning, and requirements for new network service capabilities. Member, Tau Beta Pi.

**Irvin S. Yavelberg**, B.S.E.E., 1960, University of Arizona; M.S.E.E., New York University, 1962; M.S. Operations Research, New York University, 1973; Naval Ordnance Test Stations, China Lake, California, 1960; Bell Laboratories, 1960—. At the Naval Ordnance Test Station, Mr. Yavelberg worked on missile electronic component design. At Bell Laboratories, he initially worked on the development of guidance equations for launch vehicles whose objective was to place communication satellites, like TELSTAR, into orbit. Mr. Yavelberg then worked on various system engineering assignments associated with the SAFEGUARD Anti-Ballistics Missile System and the APOLLO APPLICATIONS Space Program. Following this, he worked on simulation modeling of the telephone network and helped design administrative traffic controls. Since 1973, Mr. Yavelberg held supervisory positions in the system engineering, requirements, and design areas of the Facility Assignment and Control System (FACS) project. He is currently the Human Performance Engineering supervisor for the Premises Information System (PREMIS) project. Member, National Society for Performance and Instruction, Human Factors Society.



## PAPERS BY BELL LABORATORIES AUTHORS

### COMPUTING/MATHEMATICS

**Approximation and Algorithms for Bin Packing Problems: A Survey.** M. R. Garey and D. S. Johnson, *Analysis and Design of Algorithms in Combinatorial Optimization* (Proc School on Combinatorial Optimization), edited by G. Ausiello and M. Lucerfina, New York: Springer-Verlag, 1981, pp 147-72.

**Color Induced Distortion in Graphical Displays.** W. S. Cleveland and R. McGill, *The Harvard Library of Computer Graphics*, 17 (1980), pp 7-10.

**Comment on Friedman and Rafsky.** C. L. Mallows, *J Amer Statist Assoc*, 76, No. 374 (June 1981), p 294.

**Connected Digit Recognition From Using a Level Building DTW Algorithm.** C. S. Myers and L. R. Rabiner, *IEEE Trans Acoustics, Speech, and Signal Processing*, ASSP-29, No. 3 (June 1981), pp 351-63.

**Global Approximate Newton Methods.** R. E. Bank and D. J. Rose, *Numer Math*, 37 (1981), pp 279-95.

**A Group Testing Problem on Two Disjoint Sets.** G. J. Chang and F. K. Hwang, *SIAM J Algebraic and Discrete Math*, 2, No. 1 (1981), pp 35-8.

**Hierarchical Addressing Scheme in Bell Labs Network.** S. Kho, *Proc 15th Annual Hawaii Int Conf Syst Sci* (January 8, 1982).

**Human-Interface Design of the Display Text Editor TED.** D. K. Sharma and A. M. Gruchacz, *Proc Natl Telecommun Conf* (November 29-December 3, 1981), pp C6.5.1-C6.5.6.

**The "Icicle Plot" Display of a Hierarchical Clustering.** J. B. Kruskal and J. M. Landwehr, *Amer Statist Assoc 1980 Proc Statist Computing Section* (August 11-14, 1980), p 302.

**Lowess: A Program for Smoothing Scatterplots by Robust Locally Weighted Regression.** W. S. Cleveland, *Amer Statist*, 35, No. 1 (1981), p 54.

**Matrix Correspondences of Plan Partitions.** E. R. Gansner, *Pacific J Math*, 92, No. 2 (1981), pp 295-315.

**MML CCITT Man-Machine Language.** B. H. Hornbach, *Natl Telecommun Conf Record 1981*, November 29-December 3, 1981.

**Node-Deletion Problems on Bipartite Graphs.** M. Yannakakis, *SIAM J Comput*, 10, No. 2 (May 1981), pp 310-27.

**On the Complexity of Testing Implications of Functional and Joint Dependencies.** D. Maier, Y. Sagiv, and M. Yannakakis, *J Assoc Comput Mach*, 28, No. 4 (October 1981), pp 680-95.

**An Optimal Hierarchical Procedure for a Modified Binomial Group Testing Problem.** F. K. Hwang, C. G. Pfeifer, and P. Enis, *J Amer Statist Assoc*, 76 (1981), pp 947-9.

**Scoring Rule and Majority Agreements for Large Electorates With Arbitrary Preferences.** W. V. Gehrlein and P. C. Fishburn, *Math Soc Sci*, 2, No. 1 (1981), pp 23-33.

**Solving Nonlinear Systems of Equations Arising from Semiconductor Device Modeling.** D. J. Rose and R. E. Bank, *SIAM J Numer Anal*, 17, No. 6 (December 1980), pp 806.

**Some Graphical Procedures for Studying a Logistic Regression Fit.** J. M. Landwehr, D. Pregibon, and A. C. Shoemaker, *Amer Statist Assoc 1980 Proc Bus and Econ Statist Section* (August 11-14, 1980), pp 15-20.

### ENGINEERING

**Criteria for Continued Inspection During the Discontinue Phase of MIL-STD-105D.** B. S. Liebesman, *J Quality Tech*, 14, No. 1 (January 1982), pp 34-9.

**Erratic Welding in Low-Carbon Steels Caused by Variations in the Electrical Resistivity.** R. L. Cohen and K. W. West, *Tooling and Prod* (February 1981), pp 94-6.

**Experimental and Theoretical Results on Fine-Line P-Channel Devices.** W. Fichtner, R. M. Levin, K. K. Ng, and G. W. Taylor, IEDM Technical Digest (December 7-9, 1981), pp 546-9.

**Experimental Studies of a Nonlinear Interface.** P. W. Smith, W. J. Tomlinson, and P. J. Maloney, IEEE J Quantum Elect, *QE-17*, No. 3 (March 1981).

**Fasnet: Proposal for a High Speed Local Network.** J. O. Limb, INRIA Proc Int Workshop of Office Inform Syst, Session II (October 13-15, 1981).

**A Fourth-Order Homogeneous Random Field in Oceanography.** T. T. Kadota and F. M. Labianca, IEEE Trans Inform Theory, *IT-27*, No. 5 (September 1981), pp 656-8.

**A Functional Technique for Decomposing the Complexity of Requirements Analysis.** P. Zave, Proc 15th Hawaii Int Conf on Syst Sci (January 6-8, 1982), pp 295-303.

**Geomagnetic Induction on a Trans-Atlantic Communications Cable.** L. V. Medford, A. Meloni, L. L. Lanzerotti, and G. P. Gregori, Nature, *290* (1981), pp 329-30.

**A Hierarchical Optimization Method of Large-Scale Time-Delay Systems.** M. Jamshidi and M. Malek-Zavarei, Proc 20th IEEE Conf Decision and Control (December 18, 1981), pp 1303-4.

**Hybrid Adaptive Procedure for Estimation of Psychometric Functions.** J. L. Hall, J Acoustical Soc of Amer, *69*, No. 6 (June 1981), pp 1763-9.

**Performance of Frequency-Hopping Multiple-Access Multi-Level FSK Systems With Hard-Limited and Linear Combining.** O. Yue, IEEE Trans Commun, *COM-29*, No. 11 (November 1981), pp 1687-94.

**Real-Time Simulation of Adaptive Transform Coding.** R. V. Cox and R. E. Crochiere, IEEE Trans Acoustics, Speech, and Signal Processing, *ASSP-29*, No. 2 (April 1981), pp 147-54.

**The Rotation of Hydromagnetic Waves by the Ionosphere.** M. K. Andrews, L. J. Lanzerotti, and C. G. MacLennan, *ULF Pulsations in the Magnetosphere*, edited by D. J. Southwood, Dordrecht, Holland: D. Reidel, 1981, pp 141-5.

**A Self-Detecting Microwave Spectrometer Based on the BARITT Diode.** W. M. Walsh and L. W. Rupp, Rev Sci Instr, *52* (1981), p 1029.

**Study of Microwave Self-Detecting Oscillators.** L. W. Rupp, Jr., Rev Sci Instr, *57*, No. 7 (July 1981), pp 1032-6.

**Transversely Excited Waveguide Lasers.** O. R. Wood, II, and P. W. Smith, Conf Lasers and Electro-Optical Systems (1980), pp 34-6.

**Wave-Scattering Properties of Interdigital Saw Transducers.** R. L. Rosenberg, IEEE Trans Sonics and Ultrasonics, *SU-28*, No. 1 (January 1981), pp 26-41.

## MANAGEMENT/ECONOMICS

**Is Telephone Service an Economic Right?** E. E. Zajac, Proc Conf Inst Pub Util, Williamsburg, Va (December 10-12, 1979), pp 94-109.

**Majority Committees.** P. C. Fishburn, J Econ Theory, *25*, No. 2 (October 1981), pp 255-68.

**Temporal Variability of United States Consumer Price Index.** W. A. Gale, J Money, Credit, and Banking, *13*, No. 3 (August 1981), pp 273-97.

## PHYSICAL SCIENCES

**Adsorption Sites and Bard Lengths of Iodine on Cu(111) and Cu(100) from Surface Extended X-ray Absorption Fine Structure.** P. H. Citrin, P. Eisenberger, and R. C. Hewitt, Phys Rev Lett, *45* (1980), p 1948.

**Amorphous Semiconducting Ag-Te Films.** J. J. Hauser, J de Physique, *42*, C-4, Supp to No. 10, Proc 9th Int Conf Amorphous and Liquid Semiconductors, edited by B. K. Chakraverty and D. Kaplan, Les Ulis, France: Les Editions de Physique (October 1981), pp 943-6.

**Analytic Evaluation of Interference Effects on Laser Output in a Fabry-Perot Resonator.** G. P. Agrawal and M. Lax, J Opt Soc Amer, *71* (May 1981), pp 515-9.

- Band Structure and Optical Properties of  $\text{In}_{1-x}\text{Ga}_x\text{As}_y\text{P}_{1-y}$ .** S. M. Kelso, D. E. Aspnes, C. G. Olson, D. W. Lynch, R. E. Nahory, and M. A. Pollack, *Proc 4th Int Conf Ternary and Multinary Compounds, Japanese J Appl Phys, 19* (1980) Supplement 19-3, pp 327-31.
- Buckling Reconstruction on the Laser Annealed Si(111) Surface.** Y. J. Chabal, J. E. Rowe, and D. A. Zwemer, *Phys Rev Lett, 46*, No. 9 (March 2, 1981), p 600.
- Calculations on the Effects of the Surface Potential Barrier in LEED.** E. G. McRae and M. L. Kane, *Surface Science, 108*, No. 3 (July 1981), pp 435-45.
- The Challenge of Biological Membranes: The Ultimate in Microelectronics.** B. Chance, P. Mueller, D. DeVault, and L. Powers, *Phys Today* (October 1980), pp 32-8.
- Channeling Analysis of CW-Argon-Laser-Annealed, As<sup>+</sup>-Implanted Silicon.** J. S. Williams, W. L. Brown, and J. M. Poate, *Conf Proc Symp Laser-Solid Interactions and Laser Conf Processing, Amer Inst Phys Conf No. 50* (1979).
- Characterization of InGaAs Photodiodes Exhibiting Low Dark Current and Low Junction Capacitance.** R. F. Leheny, R. E. Nahory, M. A. Pollack, E. D. Beebe, and J. C. DeWinter, *IEEE J Quantum Elect, QE-17*, No. 2 (February 1981), pp 227-31.
- Characterization of the Hexatic B and Crystal B Phases By Optical Microscopy.** J. W. Goodby and R. Pindak, *Molecular Crystals Liquid Crystals, 75* (1981), p 233.
- Chemisorption and Surfaces Studied by Nuclear Magnetic Resonance Spectroscopy.** T. M. Duncan and C. R. Dybowski, *Surface Science Reports, 1*, No. 4 (November 1981), pp 157-250.
- Coding Dilute <sup>3</sup>He-<sup>4</sup>He Mixtures To 0.6 mK.** D. D. Osheroff and L. R. Corruccini, *Phys Lett, 82A*, No. 1 (February 3, 1981), pp 38-40.
- Coexistence of Superconductivity and Long-Range Antiferromagnetic Order in  $\text{Gd}_{1.2}\text{Mo}_6\text{Se}_8$ .** M. B. Maple, L. D. Wolf, C. F. Majkrzak, G. Shirane, W. Thomlinson, and D. E. Moncton, *Phys Lett A, 77*, No. 6 (June 23, 1980), p 487.
- A Compound Exhibiting a Crystal B to Hexatic B Phase Change.** J. W. Goodby, *Molecular Crystals Liquid Crystals Lett, 72* (1981), p 95.
- Conductivity Cusp in a Disordered Metal.** T. F. Rosenbaum, K. Andres, G. A. Thomas, and P. A. Lee, *Phys Rev Lett, 46*, No. 8 (February 23, 1981), pp 1111-406.
- A Continuous X-ray Study of the Interfacial Reaction in Au-Al Thin Film Couples.** J. M. Vandenberg and R. A. Hamm, *J Vac Sci Tech, 19* (1981), p 84.
- Cosmic-Ray-Power Spectral Variations.** I. Simulation Study. L. J. Lanzerotti, C. G. MacLennan, S. P. Agrawal, and D. Venkatesan, *J Geophys Res, 86* (1981), pp 6951-6.
- Crystal Growth and Impurity Incorporation by Transient Laser and Electron Beam Heating of Semiconductors.** W. L. Brown, *Proc 15th Int Conf Phys of Semiconductors, 49* (September 1-5, 1980), pp 1271-80.
- Current Techniques in Muon Spin Rotation Experiments.** W. F. Lankford, K. G. Lynn, W. J. Kossler, A. T. Fiory, R. P. Minnich, and C. E. Stronach, *Nucl Instr and Meth, 185* (1981), pp 469-80.
- Degradation of  $\text{LaNi}_5$  Hydrogen Absorbing Material by Cycling.** R. L. Cohen, K. W. West, and J. H. Wernick, *J Less Common Metals, (Proc Int Conf Hydrides), 73* (1980), pp 273-9.
- Disorder: A Frontier of Theoretical Physics.** P. W. Anderson, *Nature Journal, Shanghai, China* (February 1981), No. 2, p 83.
- Distribution of Amplitudes of Geomagnetic Variations and Relationships to the K Index.** M. L. Kane, L. J. Lanzerotti, and C. G. MacLennan, *J Geophys Res, 86* (1981), pp 5889-94.
- Effect of Configuration Cross-Over on the Electronic Raman Scattering by 4f Multiplets.** G. Guntherodt, A. Jayaraman, E. Anastassakis, E. Bucher, and H. Bach, *Phys Rev Lett, 46* (1981), p 855.
- The Effect of  $\alpha$ -Particle Damage on the Superconducting and Magnetic Transition Temperatures of Ternary Borides.** R. C. Dynes, J. M. Rowell, and P. H. Schmidt, *Ternary Superconductors* (Proc Int Conf Ternary Superconductors), Amsterdam: Elsevier North Holland Publishing, 1981, pp 169-73.
- Electron-hole Plasma on Photoexcited  $\text{Al}_x\text{Ga}_{1-x}\text{As}$ .** E. Cohen, M. D. Sturges, M. C. Olmstead, and R. A. Logan, *Phys Rev B, 22*, No. 2 (July 15, 1980), p 771.
- Electronic and Geometric Structure of the Point Defect GaP:O.** G. A. Baraff and M. Schluter, *Proc on Defects and Radiation Effects in Semiconductors*, 1980, Inst

- of Phys Conf Series No. 59, edited by R. E. Hasiguti, Inst of Phys, Bristol and London (1981), pp 287-92.
- Electrophysiological Evidence for the Development of Stereopsis in Human Infants.** B. Petrig, B. Julesz, W. Kropfl, M. Anliker, and G. Baumgartner, *Science*, *213* (1981), pp 1402-5.
- Energy Transfer and Anderson Localization in Ruby.** S. Chu, H. M. Gibbs, S. L. McCall, and A. Passner, *Phys Rev Lett*, *45* (1980), pp 1715.
- Extended X-ray Absorption Fine Structure—Its Strengths and Limitations as a Structural Tool.** P. A. Lee, P. H. Citrin, P. Eisenberger, and B. M. Kincaid, *Reviews Mod Phys* (1981), pp 769-806.
- Fluorimetric Study of the Binding of Protoporphyrin to Haemopexin and Albumin.** A. A. Lamola, I. Asher, U. Muller-Eberhard, and M. Poh-Fitzpatrick, *Biochem J*, *196* (1981), pp 693-8.
- Generation of Surface Gratings with Periods <math><1000 \text{ \AA}</math>.** L. F. Johnson and K. A. Ingersoll, *Appl Phys Lett*, *38*, No. 7 (April 1, 1981), pp 532-4.
- Gravitational Lensing and the Relation Between QSO and Galaxy Magnitude-Number Counts.** J. A. Tyson, *The Astrophys J*, *248* (1981), p 689.
- Growth of Doped Silicon Layers by MBE.** J. C. Bean, *Impurity Doping Processes in Silicon, Materials Processing: Theory and Practices Series*, edited by F. F. Y. Wang, Amsterdam: North-Holland Publishing, 1981.
- Hydromagnetic Wave Observations in the Vicinity of a Magnetospheric Plasma Density Gradient.** L. J. Lanzerotti and A. Wolf, *J Geophys Res*, *86* (1981), pp 2447-50.
- Ice in the Polar Regions of the Moon.** L. J. Lanzerotti, W. L. Brown, and R. E. Johnson, *J Geophys Res*, *86* (1981), pp 3949-50.
- Identification and Assay of Synchrotron Radiation-Induced Alterations on Metalloenzymes and Proteins.** B. Chance, P. Angiolillo, E. K. Yang, and L. Powers, *FEBS Lett*, *112* (April 1980), pp 178-82.
- In<sub>0.53</sub>Ga<sub>0.47</sub>As/Si<sub>3</sub>N<sub>4</sub> N-Channel and P-Channel Inversion Mode MISFETs.** A. S. H. Liao, R. F. Leheny, R. E. Nahory, J. C. DeWinter, and R. J. Martin, 1981 IEDM Technical Digest (December 7-9, 1981), pp 637-40.
- Ion and Electron Angular Distributions in the IO Torus Region of The Jovian Magnetosphere.** L. J. Lanzerotti, C. G. MacLennan, T. P. Armstrong, S. M. Krimigis, R. P. Lepping and N. F. Ness, *J Geophys Res*, *86* (1981), pp 8491-6.
- Light Scattering Spectroscopy of Two Dimensional Electron Gases in Semiconductors.** A. Pinczuk, J. M. Worlock, H. L. Stormer, R. Dingle, W. Wiegmann, and A. C. Gossard, *Proc 15th Int Conf Phys of Semiconductors*, *49* (1980), pp 1025-8.
- Lithium Doping of Polycrystalline Silicon.** G. L. Miller and W. A. Orr, *Appl Phys Lett*, *37* (1980), p 110.
- Magnetic Order and Energetics of Dipolar Coupling in Magnetic Superconductors.** M. Redi and P. W. Anderson, *Phys Sci Sect Proc of Natl Acad Sci*, *78*, No. 1 (January 1981), pp 27-30.
- Magnetic Order in Superconductors.** D. E. Moncton, *Proc Int Conf Phys of Transition Metals*, in *Inst Phys Conf Ser No. 55*, Chapter 10 (1981), p 537.
- The Metal-Insulator Transition in Disordered Germanium-Gold.** B. W. Dodson, W. L. McMillan, J. M. Mochel, and R. C. Dynes, *Phys Rev Lett*, *46* (1981), p 46.
- The Metal Insulator Transition in Granular Aluminum.** R. C. Dynes and J. P. Garno, *Phys Rev Lett*, *46* (1981), p 137.
- MeV Ion Scattering for Surface Structure Determination.** L. C. Feldman, *CRC Critical Reviews in Solid State and Materials Science*, *10*, No. 2, Boca Raton: CRC Press (May 1981), pp 143-67.
- Microfabrication and Basic Research.** W. F. Brinkman, *VLSI Electronics: Microstructure Science*, *2*, New York: Acad Press (1981), pp 149-69.
- Multiplet Splittings and Jahn-Teller Energies for the Vacancy in Silicon.** M. Lannoo, G. A. Baraff, and M. Schluter, *Phys Rev B*, *24* (July 15, 1981), pp 955-63.
- The Negative U Character of Silicon Vacancies.** G. A. Baraff, E. O. Kane, and M. Schluter, *Proc 11th Int Conf on Defects and Radiation Effects in Semiconductors*, 1980, edited by R. R. Hasiguti, Bristol and London: Inst of Phys, 1981, pp 19-31.
- A Neutron Diffraction Study of the Coexistence of Antiferromagnetism and Superconductivity in GdM<sub>2</sub>S<sub>8</sub>.** C. F. Majkrzak, G. Shirane, W. Thomlinson, M. Ishikawa, O. Fischer, and D. E. Moncton, *Solid State Commun*, *31* (1979), pp 773-5.

- New Method for a Scaling Theory of Localization.** P. W. Anderson, D. J. Thouless, E. A. Abrahams, and D. S. Fisher, *Phys Rev B*, 22 (1980), pp 3516-9.
- Nonlinear Microwave Resonance Behavior of Itinerant Holes in an Organic Conductor.** W. M. Walsh, L. W. Rupp, F. Wudl, D. Nalewajek, P. A. Lee, and F. J. DiSalvo, *J Appl Phys*, 52 (1981), p 2031.
- Nuclear Structure Study of  $^{24m}\text{Na}$  Using Laser Induced Nuclear Orientation.** P. G. Pappas, M. S. Feld, D. E. Murnick, H. M. Gibbs, and O. R. Wood, II, *Hyperfine Interactions*, 9, Amsterdam: North-Holland, 1981, pp 187-97.
- Observation of Gas Absorption in Evaporated Amorphous Silicon Films Using Secondary Ion Mass Spectrometry.** C. W. Magee, J. C. Bean, G. Foti, and J. M. Poate, *Thin Solid Films*, 81 (1981), pp 1-6.
- Optical Recording in Hydrogenated Semiconductors.** M. Bösch, *Appl Phys Lett*, 40, No. 1 (January 1, 1982), pp 8-10.
- Oscillatory Magnetic Fluctuations Near the Superconductor-to-Ferromagnet Transition in  $\text{ErRh}_4\text{B}_4$ .** D. E. Moncton, D. B. McWhan, P. H. Schmidt, G. Shirane, W. Thomlinson, M. B. Maple, H. B. MacKay, L. D. Woolf, Z. Fisk, and D. C. Johnson, *The Amer Phys Soc*, 45, No. 25 December 22, 1980, p 2060.
- Particle Diffusion in the Geomagnetosphere: Comparison of Magnetic and Electric Field Fluctuations.** L. J. Lanzerotti and A. Wolfe, *J Geophys Res*, 85 (1980), pp 2346-8.
- Physiologically Determined Receptive Fields in Monocular and Binocular Texture Perception.** B. Julesz, *Proc 28th Int. Cong Phys Sci* (1980), pp 309-14. *Advances in Physiological Sciences*, Elmsford, N. Y.: Pergamon Press, 1981.
- Picosecond Nonequilibrium Carrier Transport in GaAs.** C. V. Shank, R. L. Fork, B. I. Greene, F. K. Reinhart, and R. A. Logan, *Appl Phys Lett*, 38, No. 2 (January 15, 1981), pp 104-5.
- Pinned Vorticity in Rotating Superfluids With Application Neutron Stars.** D. Pines, J. Shaham, P. W. Anderson, and A. Alpar, *Supplement of Progress of Theoretical Phys, Japan*, No. 69 (1980), pp 376-96.
- Polarized Low-Energy Diffraction Study of the Mechanism of Electron Reflection from  $W(001)$  at Low Energies.** E. G. McRae, D. T. Pierce, G.-C. Wang, and R. J. Celotta, *Phy Rev B* 24 (1981), pp 4230-9.
- Possible Coexisting Superconducting and Magnetic States.** H. S. Greenside, E. I. Blount, and C. M. Varma, *Phys Rev Lett*, 46 (1981), p 49.
- Pressure-Induced Phase Transformations in the Chalcopyrite Structure Compounds,  $\text{CuGaS}_2$ - $\text{AgGaS}_2$ .** A. Werner, H. D. Hochheimer, and A. Jayaraman, *Phys Rev B*, 23 (1981), p 3836.
- Pressure-Induced Valence Change in  $\text{Y60}$ .** A. Werner, H. D. Hochheimer, A. Jayaraman, and J. M. Leger, *Solid State Commun*, 38 (1981), p 325.
- The Radio Optical Emission in 3C33 South.** L. Rudnick, W. C. Saslaw, T. Crane, and J. A. Tyson, *Astrophys J*, 246 (1981), p L647.
- Rapid Changes in Plasma Flow at the Dawn Magnetopause.** L. J. Lanzerotti, C. B. MacLennan, R. P. Lepping, and S. M. Krimigis, *Planetary and Space Sci*, 28 (1980), pp 1163-9.
- Role of Dislocations in the Electrical Conductivity of  $\text{CdS}$ .** A. R. Hutson, *Phys Rev Lett*, 46, No. 17 (April 27, 1981), pp 1159-62.
- Sea Water Corrosion of Two Phase  $\text{Cu-10-Ni-8Sn}$ .** T. J. Louzon, *Environmental Degradation of Engineering Materials in Aggressive Environments*, edited by M. R. Louthan, R. P. McNutt, and R. D. Sisson, Bracksburg, Va.: Pub. Lab. for the Study of Environmental Degradation of Engrg. Matls. (September 21, 1981), pp 15-23.
- Self Consistent Calculations of the Electronic Structure for Ideal Ga and As Vacancies in GaAs.** G. B. Bachelet, G. A. Baraff, and M. Schluter, *Phys Rev B*, 24 (July 15, 1981), pp 915-25.
- SEXÁFS: New Horizons in Surface Structure Determinations.** P. Eisenberger, P. H. Citrin, R. Hewitt, and B. Kincaid, *CRC Critical Reviews in Solid State and Materials Sciences*, 10 (1981), p 191.
- Sharp Metal-Insulator Transition in a Random Solid.** T. F. Rosenbaum, K. Andres, G. A. Thomas, and R. N. Bhatt, 45, No. 21 (November 24, 1980), pp 1111-408.
- Single-Particle Energy Levels in Doped Semiconductors at Densities Below the Metal-nonmetal Transition.** R. N. Bhatt and T. M. Rice, *Phys Rev B*, 23 (February 15, 1981), pp 1920-34.

- Smectic Polymorphism and Molecular Shape—The Orthogonal Phases.** J. W. Goodby, *Molecular Crystals Liquid Crystals*, 75 (1981), p 179.
- Spin Dependence of Electronic Surface Resonance Scattering From W(100).** D. T. Pierce, R. J. Celotta, G.-C. Wang, and E. G. McRae, *Solid State Commun*, 39 (1981), pp 1053-5.
- Star Interpolation and High-Energy Expansion for Evaluating Solid-State Greens Functions.** G. A. Baraff and M. Schluter, *Phys Rev, B*, 23 (April 15, 1981), pp 4256-9.
- Summary Abstract: MeV Ion Scattering Studies of the (111) Si/SiO<sub>2</sub> Interface.** R. Haight, W. M. Gibson, and T. Narusawa, *J Vac Sci and Tech*, 18, No. 3 (April 1981), p 973.
- Systematics of the Chemisorption Energy of Hydrogen and Oxygen on Transition Metals—I.** C. M. Varma and A. J. Wilson, *Phys Rev*, 22 (1980), p 3795.
- Textons, the Elements of Texture Perception, and Their Interactions.** B. Julesz, *Nature*, 290 (1981), pp 91-7.
- Theoretical and Clinical Advances in Global Stereopsis.** B. Julesz, *Advances in Physiological Sciences, Proc 28th Int. Cong Phys Sci* (1980), pp 451-9.
- Theoretical Considerations on the Superconductor-Ferromagnet Transition Region.** E. I. Blount, *J Appl Phys*, 52 (1981), p 2169.
- Thermodynamics of Positronium Thermal Desorption From Surfaces.** S. Chu, A. P. Mills, and C. A. Murray, *Phys Rev A*, 23 (1981), p 2060.
- The Thouless Conjecture for a One-Dimensional Chain.** P. W. Anderson and P. A. Lee, *Special Issue of Progress of Theoretical Phys, Japan, Supplement No. 69* (1980), pp 212-9.
- Transitions of Superconductor to Ferro- and Antiferromagnets.** C. M. Varma, *Proc Third Int Conf on f- and d- Band Superconductivity* (June 21-23, 1979), pp 391-402.
- Workshop on Crystals and Focusing.** D. W. Berreman, 1980 AIP Conf Proc Laboratory EXAFS Facilities, No. 64 (1980), pp 100-3.
- X-ray Photoemission from the Creutz-Taube Mixed Valence Complex: A Reassessment.** P. H. Citrin and A. P. Ginsburg, *J Amer Chem Soc*, 103 (1981), p 3673.
- X-ray Studies of Smectic-B Liquid Crystal Films.** D. E. Moncton and R. Pindak, *Proc of the Ordering in Two Dimensions Conference*, Amsterdam: Elsevier North Holland, Inc, 1980, p 83.

## SPEECH AND ACOUSTICS

- An Improved Endpoint Detector for Isolated Word Recognition.** L. F. Lamel, L. R. Rabiner, A. E. Rosenberg, J. G. Wilpon, *IEEE Trans Acoustics, Speech, and Signal Processing, ASSP-29*, No. 4 (August 1981), pp 777-85.
- Isolated and Connected Word Recognition—Theory and Selected Applications.** L. R. Rabiner and S. E. Lewinson, *IEEE Trans Commun, COM-29*, No. 5 (May 1981), pp 621-59.
- A Level Building Dynamic Time Warping Algorithm for Connected Word Recognition.** C. S. Myers and L. R. Rabiner, *IEEE Trans Acoustics, Speech, and Signal Processing, ASSP-29*, No. 2 (April 1981), pp 284-97.
- Performance of Transform and Subband Coding Systems Combined With Harmonic Scaling of Speech.** D. Malah, R. T. Crochiere, and R. V. Cox, *IEEE Trans Acoustics, Speech, and Signal Processing, ASSP-29*, No. 2 (April 1981), pp 273-83.
- Prosodic Aspects of American English Speech Rhythm.** L. H. Nakatani, K. D. O'Connor, and C. H. Aston, *Phonetica*, 38 (1981), pp 84-106.

# CONTENTS, JULY–AUGUST 1982

## Part 1

Electromagnetic Propagation in Homogeneous Media With Hermitian Permeability and Permittivity

E. E. Bergmann

A Primal Algorithm for Finding Minimum-Cost Flows in Capacitated Networks With Applications

C. L. Monma and M. Segal

Improved Reconstruction of DPCM-Coded Pictures

A. N. Netravali

An Improved Training Procedure for Connected-Digit Recognition

L. R. Rabiner, A. Bergh, and J. G. Wilpon

Simulation and Design Studies of Digital Subscriber Lines

S. V. Ahamed

## Part 2

### AUTOMATED REPAIR SERVICE BUREAU

Preface

L. Schenker and S. J. Barbera

Evolution

P. S. Boggs, M. W. Bowker, E. A. Overstreet, and R. W. Vetter, Jr.

The System Architecture

R. L. Martin

Data Base System

C. M. Franklin and J. F. Vogler

The Trouble Report Evaluation and Analysis Tool

S. P. Rhodes and L. S. Dickert

The Front-End System

S. G. Chappell, F. H. Henig, and D. S. Watson

Software Tools and Components

R. F. Bergeron and M. J. Rochkind

The Context-Sensitive Switch of the Loop Maintenance System

J. P. Holtman

**Mechanized Loop-Testing Strategies and Techniques**

**F. J. Uhrhane**

**Mechanized Loop-Testing Design**

**O. B. Dale, T. W. Robinson, and E. J. Theriot**

**Second-Generation Mechanized Loop-Testing System—A Distributed  
Microprocessor Application**

**H. Rubin**

**Cable Repair Administrative System**

**P. S. Boggs and J. R. Mashey**

**Human Performance Design Techniques**

**G. H. Leonard and J. E. Zielinski**

**Two Examples of Human Performance Analysis and Design in Plan-  
ning the ARSB**

**R. F. Gauthier and W. A. Harris**

**On-Line Documentation: Mechanizing Development, Delivery,  
and Use**

**R. J. Glushko and M. H. Bianchi**

**Economic Evaluation**

**E. A. Overstreet**

## B.S.T.J. BRIEF

### A Josephson Parallel Multiplier

By T. A. FULTON and L. N. DUNKLEBERGER

(Manuscript received January 14, 1982)

This report describes the operation of an 8- x 12-bit parallel multiplier which employs Josephson tunnel junctions (Fig. 1).<sup>1\*</sup> The device contains 548 junctions used in two-junction "Jaws"-type logic elements.<sup>2</sup> Ninety-six of these logic elements function as AND gates to form the partial products. Their outputs are fed into a Wallace-tree<sup>3</sup> arrangement of 89 full adders, each having one Jaws for the CARRY computation and one for the SUM. The thirteen most significant bits of the product are returned to the outputs.

A latching-logic mode of operation is used.<sup>1</sup> This employs a five-phase ac current supply for power and timing, provided by unipolar pulses from a room-temperature word generator. The minimum cycle time achieved is 75 ns, with a latency time delay between input and output of 30 ns measured at the room-temperature connectors. Both times are within design specifications. The latency time is determined primarily by the time required for settling of the power-supply pulse amplitudes (20 ns total) and by the round-trip cable delay (8 ns) from the room-temperature connectors to the chip immersed in liquid He. The worst-case logic delay for the Jaws elements (ripple carry through 18 stages) is estimated as <2 ns, based on the <100 ps computer-simulated Jaws delay. In initial tests after preliminary adjustment of bias levels (chiefly the five-pulse amplitudes from the word generator), the multiplier operates correctly on all of several hundred selected combinations of inputs, as judged by oscilloscope monitors. Both exhaustive testing and optimization of bias levels remain to be carried out.

---

\* (See Ref. 1 for two recent publications devoted to reviews of the state of the art in Josephson digital circuits.)

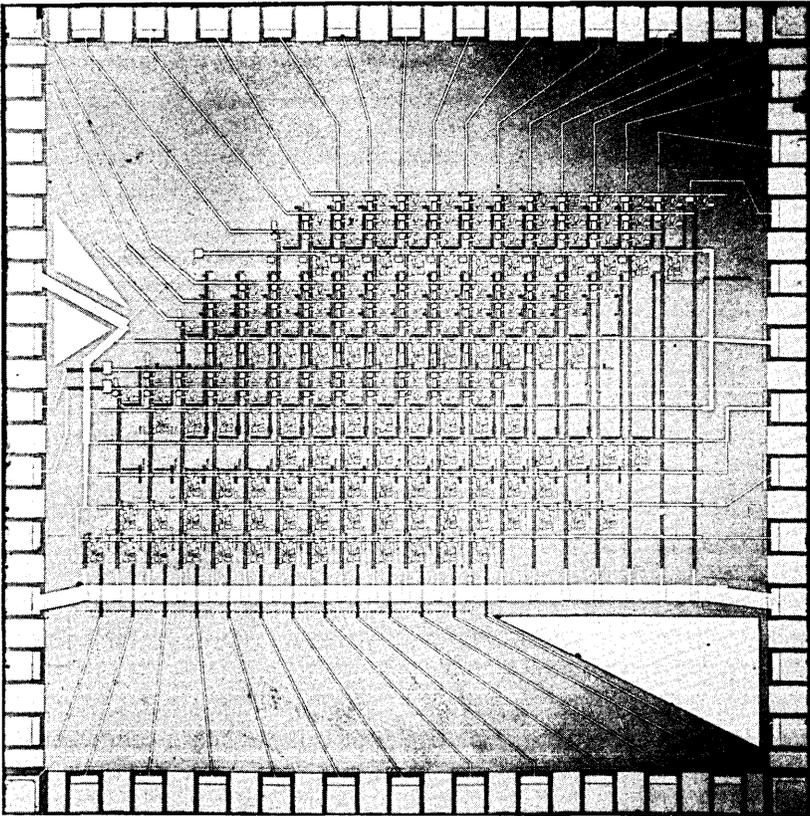


Fig. 1—A photomicrograph of the multiplier chip.

Circuits were fabricated on 2-inch-diameter oxidized Si wafer substrates having 12 chips per wafer. The circuit has seven levels. These are the Nb ground plane, two levels of ground-plane insulation (anodized Nb and an evaporated Ge-SiO sandwich), the Cu-Ge resistors, the Pb-alloy y-direction wiring and junction-base electrodes, the Ge-SiO crossover insulation, and the Pb-alloy x-direction wiring and junction-upper electrodes. Linewidths of  $10\ \mu\text{m}$  are used predominantly in the wiring and junctions, while some resistors are of  $2.5\text{-}\mu\text{m}$  linewidth.

Fabrication was performed in a small class-1000 clean-room facility. Failures were caused primarily by lithographic or other fabrication-related defects and, secondarily, by nonuniformities in junction-critical currents. Estimates for the yield for defect-free fabrication are roughly 25 percent.

The very-high-speed capabilities of Josephson junctions are not well utilized in the particular design chosen for the multiplier. It was

decided to emphasize simplicity at the expense of performance (within the speed specifications mentioned previously) to minimize fabrication difficulties. Nevertheless, to our knowledge the circuit is the most complex, fully working Josephson circuit described to date.

We wish to acknowledge substantial contributions to this work by J. N. Hollenhorst, S. S. Pei, and S. K. Tewksbury. We have benefited by continued support from R. E. Slusher and J. A. Giordmaine.

## REFERENCES

1. IBM J. of Res. and Dev. *24* (March 1980) and IEEE Trans. Electron Devices, *ED-27* (October 1980).
2. T. A. Fulton, L. N. Dunkleberger, and S. S. Pei, "A Simple, High-Performance Current-Switched Josephson Logic Gate," Appl. Phys. Lett., *34* (1979), pp. 709-11.
3. C. S. Wallace, "A Suggestion for a Fast Multiplier," IEEE Trans., *EC-13* (1964), pp. 14-7.







**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.

CONTENTS (continued)

<b>Software</b>	<b>863</b>
G. W. Gates, R. F. Kranzmann, and L. D. Whitehead	
<b>Office Engineering, Maintenance, and Reliability</b>	<b>885</b>
P. W. Bowman, J. C. Kennedy, and G. A. Van Dine	
<b>Acronyms and Abbreviations</b>	<b>913</b>
CONTRIBUTORS TO THIS ISSUE	<b>915</b>
PAPERS BY BELL LABORATORIES AUTHORS	<b>923</b>
CONTENTS, JULY-AUGUST ISSUE	<b>929</b>
<b>B.S.T.J. Brief:</b>	<b>931</b>
<b>A Josephson Parallel Multiplier</b>	
T. A. Fulton and L. N. Dunkleberger	