THE BELL SYSTEM TECHNICAL JOURNAL
Vol. 52, No. 9, November, 1973
Printed in U.S.A.

Overload Model of Telephone Network Operation

By R. L. FRANKS and R. W. RISHEL

(Manuscript received June 5, 1973)

An analytic model for the steady-state behavior of an overloaded telephone network is given. The model includes trunk and machine congestion, retrials, "don't answer and busy," and some network management controls. It is significantly cheaper to use than Monte Carlo simulations for moderate size networks. It compares well with Monte Carlo simulation calculations of point-to-point completion probabilities and the expected number of messages in progress. It compares less well for sender attachment delay and probability of time-out calculations in switching machines.

I. INTRODUCTION TO THE PROBLEM

The purpose of this paper is to develop an analytic model of a telephone network which displays the major steady-state behavior of the network under overload conditions and which is computationally tractable. Besides being used to predict steady-state network operation in the presence of overload, such a model should help in the development of insight into network operation. Also, with an analytic model available, optimization theory can be brought to bear on various problems in network management.

Analytic network modeling seems to have been aimed at the trunking network design problem in the past. The usual approach was to assume that switching machines had enough capacity to have no effect on the traffic through them. Under these conditions, the stream of call attempts on a trunk group may have its distribution changed in two ways: the calls have previously been offered to a different trunk group and overflowed to the present one, or some of the calls in the stream were removed as a result of blocking on a trunk group in series with the present one. These cases have been handled by Wilkinson's Equivalent Random Method[1] and Katz's Carried Equivalent Method.[2]

When a network is overloaded, the effect of machine congestion is not negligible and must be taken into account. Early work on toll

machine congestion was done by Helly,[3] who considered a homogeneous group of identical machines connected by infinite trunk groups. His approach suggested the way we treat sender holding time in the switching machine model. Recently Szybicki[4] gave a model for an overloaded local switching machine.

Monte Carlo call-by-call simulations have been used to study network behavior. Recent examples at Bell Laboratories are simulations by J. A. Kohut[5] and J. M. McCormick. These simulations have the advantage of great flexibility. They also give transient response as well as the steady-state response of the network. Call-by-call simulations may require many runs, or long runs, to obtain reliable statistics for a process under study. They tend to be more expensive to run than analytic models.

II. INTRODUCTION TO A TELEPHONE NETWORK

From a traffic point of view, the network consists of end offices, switching machines, and trunks. The end offices serve as sources and destinations of calls. The trunks are message paths through the network. The switching machines are nodes in the network at which the choice is made of the path to be taken.

To illustrate the important effects in network operation, let us trace the progress of a typical call through the simple network shown in Fig. 1. The call enters the network through end office 1. It finds a free circuit on the trunk group from 1 to 2, attaches to it, and simultaneously bids for a sender in switching machine 2. After a short wait, it is accepted into machine 2. It finds the trunk group from 2 to 4 full, and attempts to attach to the trunk group from 2 to 3. There is a free circuit on that trunk group, so the call attaches to it and bids for a sender in machine 3. This process continues until the call enters end office 5. If the destination telephone is not busy, it rings. If it is answered, the attempted call is successful and becomes a message.

This typical call went through three switching machines. The block

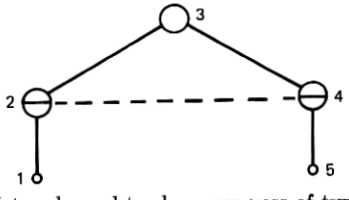


Fig. 1—Network used to show progress of typical call.

diagram in Fig. 2 shows the sequence of operations in a Bell System No. 4A-ETS switching machine.

A call coming into a switching machine enters a queue to wait for a vacant sender. When a call gets a sender, its destination information is impulsed to the sender, and the sender it had in the previous machine is released.

The call then queues for the remaining common control equipment, indicated here as a decoder-marker combination. This decoder-marker decides on the machine the call should be routed to next, tests for a vacant trunk, and sets up the connection, if possible. If there are no vacant trunks to appropriate subsequent machines, a no-circuit announcement is given. After a no-circuit announcement, the trunks on all the links over which the call had progressed are released. If the call is routed to a subsequent machine, it enters a queue for a sender in that machine.

The sender in the current machine is occupied by the call from the time it begins processing the call until the call has transmitted its destination information to the sender which processes the call in the subsequent machine. If a call waits longer than a fixed time to get a sender in the subsequent office, it is timed out. If it is timed out, the call is sent back to the marker-decoder which then connects it to a no-circuit announcement.

Under normal network operation, very few calls receive a no-circuit announcement, and fewer still time out while waiting for a sender. By far the most important causes of a call failing to become a message are for the called telephone to be busy when the call arrives and for the called customer to fail to answer the phone when it rings. When the network is overloaded, the number of no-circuit announcements increases, and time-outs become more frequent. Not only does the percentage of failures increase as the network becomes overloaded, but the number of successful attempts may actually decrease.

The factors underlying the decrease in the number of calls carried by the telephone network as it becomes highly overloaded were already understood in early work, such as Reference 3. As a call is being set up, it uses equipment in one switching machine until the next switching machine on its route accepts the call and receives the destination of the call from the previous machine. If a switching machine becomes overloaded, machines adjacent to it will have to wait longer to have their calls accepted and the destinations passed on. This causes an increase in the service time for putting a call through these machines. This in turn may cause the adjacent machines to

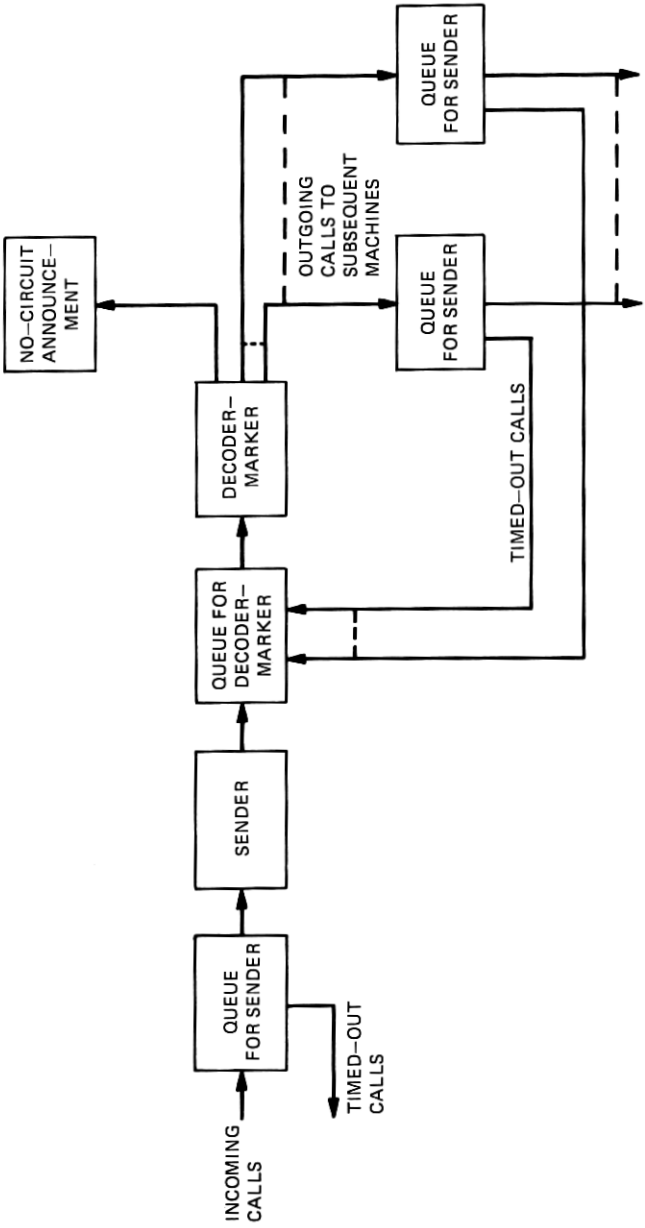


Fig. 2—Switching machine block diagram.

become congested. The time-out mechanism helps to relieve this congestive phenomenon. However, even with time-outs, switching machine congestion can back up throughout the network. Calls being set up occupy trunks on the partial route over which they have progressed. If a large number of calls are attempting on routes which are blocked, a portion of the capacity of certain links could be used by these ineffective attempts trying to set up. This would use capacity that could be utilized by talking calls. These ineffective attempts also use switching capacity in machines preceding the blockage. Most blocked attempts try again; these retrials increase the congestion.

The model to be set up will incorporate the features mentioned above. Based on those observations, the model must take into account both trunking congestion and switching machine congestion. Any call which enters the network will take trunk and machine capacity, even if it fails. For this reason, its effect on the network depends not only on whether a call succeeded or failed, but also on how far it progressed and over which particular route. This information must also be included in the model.

III. INTRODUCTION TO THE MODEL

Our view of modeling the network is probabilistic. We assume that for each trunk group there is a probability that an attempt on it will find a free circuit and for each switching machine there is a probability that an attempt on it will be accepted before timing out. We further assume that these acceptance probabilities are independent of the past history of the attempt.

The model has two conceptually distinct parts. First, the global problem is, given these acceptance probabilities, to find the various quantities of interest such as the expected number of messages in progress between each source-destination pair, the attempt rate on each trunk group and on each switching machine, and the point-to-point completion probability. Second, the local problem is, given those quantities, to find the acceptance probabilities for each switching machine and each trunk group. The local and global problems together give a large number of coupled nonlinear equations which describe the steady-state behavior of the network. These equations form the model.

For the model to be computationally tractable, the number of equations involved must be as small as possible. For this reason we have assumed that each stochastic process in the model can be described by a single parameter. (For example, the call origination process

between a given source-destination pair is assumed to be Poisson.) Even with this assumption, the number of equations involved is very large for any reasonable size network.

It must be emphasized that this is a first-generation model for an overloaded network. While the model does very well at predicting certain statistics, it is relatively poor at predicting others. A model using two parameters to describe each stochastic process could be more accurate than this one. Within the structure of this model, the switching machine treatment could be improved.

The network model given here is in several ways similar to the model used in the optimization problem of Reference 6.

IV. GLOBAL ASPECTS OF THE MODEL

This section deals with global, or network, effects caused by local phenomena. An example of such a global statistic is the mean number of messages in progress between a source-destination pair, which depends on various local effects such as the probability a call offered to each trunk group will be accepted onto it.

Before proceeding further, some definitions are required.

A complete route, $R = (a, b_1, b_2 \cdots b_n, c)$, is a list of the switching machines through which a call may pass in going from the end office connected to a machine a to the end office connected to machine c. For example, in Fig. 1 there are two complete routes, (2, 4) and (2, 3, 4), from end office 1 to end office 5.

A partial route, $r = (a, b_1, b_2 \cdots b_m; c)$ of a complete route R describes the route occupied by a call in the process of being set up and its destination. For example, a call on $r = (a, b_1, b_2, b_3; c)$ started in the end office attached to switching machine a, passed through machines a, b_1, and b_2, has entered (or is waiting to enter) machine b_3, and has as its destination the end office connected to machine c. Define

x_R = Expected rate that calls on complete route $R = (a, \alpha_1 \cdots \alpha_t, b)$ attach to the trunk group from switching machine b to its associated end office.

x_r = Expected rate that calls on partial route $r = (a, \alpha_1 \cdots \alpha_t; b)$ attach to the trunk group from α_{t-1} to α_t.

z_r = Expected rate that calls on partial route r are connected to senders in switching machine α_t.

t_r = Expected rate that calls on partial route r time out while waiting for senders in switching machine α_t.

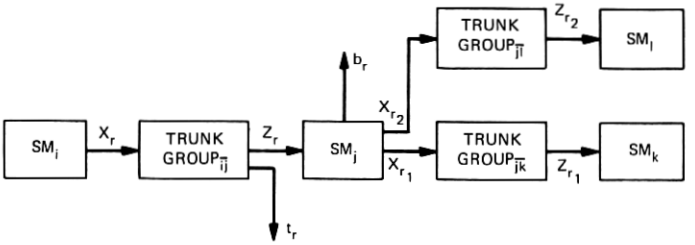


Fig. 3—Relationship of x_r, z_r, t_r, and b_r.

b_r = Expected rate that calls on partial route r which attach to senders in switching machine α_ℓ are blocked because of a lack of outgoing circuits.

SM_i = the ith switching machine.

To make the meaning of these variables more obvious, consider Fig. 3. Calls on partial route $r = (a, i, j; b)$ attach to trunk group $\overline{ij}$, and therefore bid for senders in SM_j at a rate x_r. Some of them are accepted, at a rate z_r, into SM_j and the remaining ones time out at a rate t_r.

Those calls accepted into SM_j then attempt to reach end office b by attaching to link $\overline{jk}$, at rate x_{r_1}, where $r_1 = (a, i, j, k; b)$.

If $x_{r_1} < z_r$, some of the calls alternate route over link $\overline{j\ell}$, with rate x_{r_2}, where $r_2 = (a, i, j, \ell; b)$. If $x_{r_1} + x_{r_2} < z_r$ and there is no further alternate route available, then some of the calls are blocked with rate

$$b_r \triangleq z_r - (x_{r_1} + x_{r_2}).$$

The global aspect of the network relates the x_r's, z_r's, b_r's, and t_r's for all partial routes in the network. The following assumptions are made:

(*i*) The model assumes that every type of call attempting to enter switching machine i has the same probability, P_i, of being accepted. Therefore

$$z_r = P_i x_r \tag{1}$$

for each partial route r entering SM_i.

(*ii*) It further assumes that every type of call attempting to attach to trunk group, $\overline{ij}$, has the same probability $P_{\overline{ij}}$ of attachment. The resulting equations are given below as (2) for the case with no network management control. Appendix A contains the equations that result when network management controls are included.

If a call from a to b in SM_i has a preferred route over link $\overline{ij}$ and a second most preferred route over link $\overline{ik}$, then

$$\begin{aligned} x_{r_1} &= P_{\overline{ij}} z_r \\ x_{r_2} &= P_{\overline{ik}}(z_r - x_{r_1}), \end{aligned} \tag{2}$$

etc., where

$$\begin{aligned} r &= (a,i;b) \\ r_1 &= (a,i,j;b) \\ r_2 &= (a,i,k;b). \end{aligned}$$

(*iii*) It is assumed that any call successfully attached to the trunk group leading to its destination end office, b, has a probability PA_b of being answered and therefore becoming a complete call. This probability depends on the destination. Therefore,

$$x_R = PA_b P_{\overline{bb}} z_r, \tag{3}$$

where

$$\begin{aligned} R &= (a,i,j,k,b) \\ r &= (a,i,j,k,b;b). \end{aligned}$$

(*iv*) The stream of original calls attempting to go from a to b is Poisson with mean $\lambda_{a,b}$. Calls that do not complete retry with probability PR. The expected long run attempt rate for calls from a to b is

$$A_{a,b} = \lambda_{a,b} + PR(A_{a,b} - C_{a,b}),$$

where $C_{a,b} = \sum_{R \text{ from } a \text{ to } b} x_R$, is the completion rate for all calls from a to b. That is,

$$A_{a,b} = \frac{\lambda_{a,b} - PR \sum_{R\ a \text{ to } b} x_R}{1 - PR}. \tag{4}$$

On the link from end office a to its switching machine, SM_a,

$$x_{(a;b)} = P_{\overline{aa}} A_{a,b}. \tag{5}$$

Clearly, with assumptions (*i*)–(*iv*), all the rates involved can be found from the P's. A method for finding the P's in terms of the x_r's and z_r's is discussed in the next section.

V. LOCAL ASPECTS OF THE MODEL

The assumptions made in Section IV for the global portion of the model place few constraints on the local part, requiring only certain acceptance probabilities, P_i, to satisfy those assumptions.

5.1 *Toll Machine Model*

The toll machines to be modeled are No. 4A-ETS switching machines. The statistics of interest in the model are the acceptance probabilities, P, mentioned in the last section, and the expected waiting time, T_w, for a call to get a sender. For relatively light loads, these statistics behave as though the machines selected the next call to be served on a first-come, first-served basis.[7] When the machines are overloaded, they behave as though calls were selected at random for service.[8]

We calculate the expected waiting time based on the former when loads are light and on the latter when they are heavy. The switch from the first to the second method is made where the curves of waiting time versus load cross.

In the light-load case the assumptions are

(*i*) The stream of calls attempting to enter each switching machine is a Poisson stream with mean λ_i.

(*ii*) The time a sender is held in SM_i by a call is an exponentially distributed random variable with mean $1/\mu_i$.

(*iii*) All toll machines have the same time-out interval, T.

(*iv*) The queuing discipline is first-come, first-served.

The problem of finding the probability, P_i, of acceptance of calls into SM_i under these assumptions has been solved.[9] The result is

$$P_i = 1 - \left\{ e^{T(N_{s_i}\mu_i - \lambda_i)} \left[\frac{1}{B(N_{s_i}, \lambda_i \mu_i^{-1})} - \frac{\lambda}{\lambda_i - N_{s_i}\mu_i} \right] + \frac{\lambda_i}{\lambda_i - N_{s_i}\mu_i} \right\}^{-1}, \quad \text{for } \lambda_i \neq N_{s_i}\mu_i \quad (6)$$

$$P_i = 1 - \left\{ \frac{1}{B(N_{s_i}, N_{s_i})} + \lambda_i T \right\}^{-1}, \quad \text{for } \lambda_i = N_{s_i}\mu_i$$

where

N_{s_i} = the number of senders in machine i

$B(n, a) = \dfrac{a^n/n!}{\sum_{j=0}^{n} a^j/j!}$, the Erlang B function.

Under these assumptions Reference 9 also gives the expected waiting time to get a server. The result is

$$T_{w_i}^f = (1 - P_i)\left\{\frac{N_{s_i}\mu_i e^{T(N_{s_i}\mu_i - \lambda_i)} - [N_{s_i}\mu_i + \lambda_i T(N_{s_i}\mu_i - \lambda_i)]}{(N_{s_i}\mu_i - \lambda_i)^2}\right\}, \quad \text{for} \quad \lambda_i \neq N_{s_i}\mu_i \tag{7}$$

$$T_{w_i}^f = (1 - P_i)T\left(1 + \frac{\lambda_i T}{2}\right), \qquad \text{for} \quad \lambda_i = N_{s_i}\mu_i.$$

In the heavy load case, assumption (*iv*) is replaced with

(*iv*)′ The queuing discipline is random.

An asymptotic expression for waiting time under this assumption is

$$T_{w_i}^r = T(1 - P_i/2), \qquad \lambda_i > N_{s_i}\mu_i. \tag{8}$$

Fitting eqs. (7) and (8) together

$$T_{w_i} = \begin{cases} T_{w_i}^f, & \lambda_i \leqq N_{s_i}\mu_i \\ \min\{T_{w_i}^f, T_{w_i}^r\}, & \lambda_i > N_{s_i}\mu_i. \end{cases} \tag{9}$$

To find P_i and T_{w_i}, all that is needed are T, λ_i, and μ_i^{-1}. T is given and

$$\lambda_i \equiv \sum_{r \in I_i} x_r \tag{10}$$

where

$$I_i = \{r | r = (a,\alpha_1 \cdots i; b) \text{ for some } a, b \text{ and } \alpha_1 \cdots\}.$$

To find μ_i^{-1}, consider Fig. 4. Calls on partial route r bid for a sender and connector in SM_i at rate x_r. All calls waiting to enter SM_i have an expected waiting time, T_{W_i}. Calls on partial route r are accepted into SM_i at rate z_r. It takes T_C seconds to connect an incoming trunk to a sender.

Once a sender is connected, it requires T_P seconds to pulse the digits into that sender. It then waits T_{WM} seconds for a translation device. The time to translate the digits, look for an available trunk on an acceptable route, and connect to that trunk is taken as a constant, T_M seconds. If no circuit is available, a call is attached to a no-circuit announcement. Otherwise it is attached to a trunk connected to some switching machine, say SM_j. The call, and the sender in SM_i, wait for a sender and connector in SM_j. If the call is accepted into SM_j, it takes T_C seconds to connect to the next sender and another T_P to pulse its digits into that sender. If the call hasn't been accepted by T seconds, it times out, and must return to the common control equipment to be connected to a no-circuit announcement.

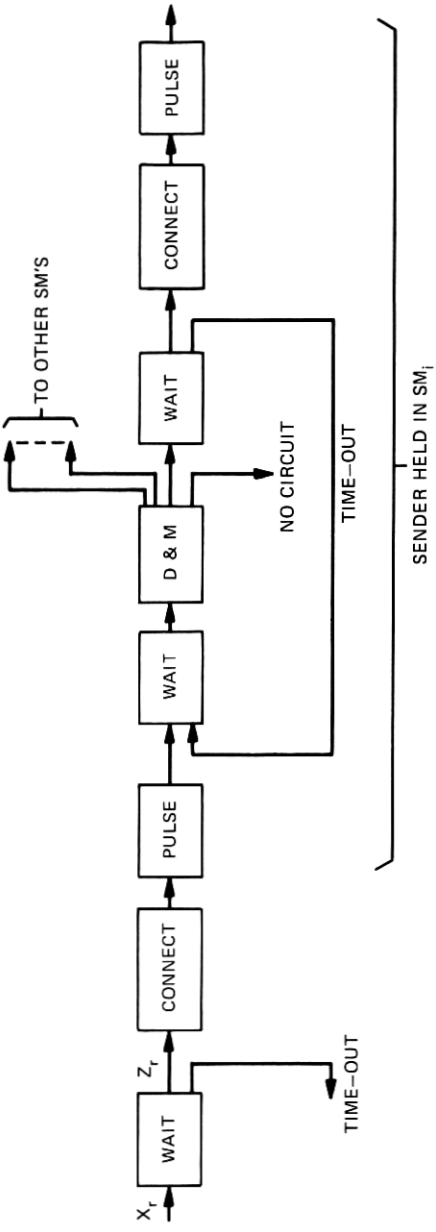


Fig. 4—Flow diagram for expected sender holding time.

The expected time a sender is held by any call accepted into SM_i is

$$\begin{aligned}\mu_i^{-1} = {} & (T_P + T_{WM} + T_M) \\ & + (\text{Expected waiting time for calls leaving } SM_i) \\ & + (T_{WM} + T_M)(\text{Probability a call leaving } SM_i \text{ times out}) \\ & + (T_P + T_C)(\text{Probability a call leaving } SM_i \text{ is accepted into the next machine}).\end{aligned}$$

More explicitly,

$$\begin{aligned}\mu_i^{-1} = {} & [T_P + T_{WM} + T_M] + \left[\frac{\sum_{r\in 0_i} T_{W_{j_r}} x_r}{\sum_{r\in I_i} z_r}\right] \\ & + [T_{WM} + T_M]\left[\frac{1 - \sum_{r\in 0_i} z_r + \sum_{R\in \bar{0}_i} x_R}{\sum_{r\in I_i} z_r}\right] \\ & + [T_P + T_C]\left[\frac{\sum_{r\in 0_i} z_r + \sum_{R\in \bar{0}_i} x_R}{\sum_{r\in I_i} z_r}\right], \qquad (11)\end{aligned}$$

where

$$j_r = \alpha_\ell$$

when

$$r = (a,\alpha_1 \cdots \alpha_\ell ; b)$$

and 0_i and $\bar{0}_i$ are sets of partial and complete routes, respectively, defined by

$$\begin{aligned}0_i &= \{r \mid r = (a,\alpha_1 \cdots i,\alpha_\ell ; b) \text{ for some } a, b, \alpha_1 \cdots\} \\ \bar{0}_i &= \{R \mid R = (a,\alpha_1 \cdots i,b) \text{ for some } a, b, \alpha_1 \cdots\}.\end{aligned}$$

The only symbol in (11) remaining to be explained is T_{WM}, the expected time spent waiting for a decoder-marker. The decoder-markers are modeled as a finite source queue. The assumptions are

(*i*) There are N_m exponential servers.
(*ii*) The queuing discipline is first-come, first-served.
(*iii*) Each N_s sender in the switching machine either is waiting for or receiving marker service or is generating its next marker bid with an exponential interarrival time of mean $1/\gamma$.

This finite source queuing model has been analyzed in Reference 10. The result, in a convenient computational form thanks to D. Jagerman, is

$$T_{WM} = \frac{T_M}{N_m} \frac{N_s - M_m - A[1 - B(N_s - N_m - 1, A)]}{1 + B(N_s - N_m - 1, A) \sum_{i=1}^{N_m} \frac{N_m^{(i)} (N_s - N_m)^{(-i)}}{(\gamma T_M)^i}}, \qquad (12)$$

where the subscripts corresponding to the switching machine have been suppressed and

$$A = \frac{N_m}{\gamma T_m}$$

$$N^{(i)} = N(N-1)\cdots(N-i+1)$$

$$N^{(-i)} = \frac{1}{N(N+1)\cdots(N+i-1)}.$$

To compute T_{WM}, it is necessary to know γ, which depends on the mean rate, m, at which calls arrive at the marker-decoder queue

$$m = \sum_{r \in I_i} z_r + \sum_{R \in O_i} t_r \tag{13}$$

Making use of Little's Theorem[11] and the definitions of m and γ,

$$\gamma = \frac{m}{N_s - m(T_M + T_{WM})}. \tag{14}$$

Equations (13) and (14) have a unique solution for all rates, m, which can be handled by the machines.

5.2 *Comments on the Switching Machine Model*

The interaction of switching machines in the real network is known to be an important cause of congestion. That interaction is included in this model by the waiting time of senders in one machine affecting the holding time of senders in adjacent machines.

This model has some features that appear to be ad hoc. The assumptions, however, are computationally convenient and give results similar to gross machine behavior. The network model has been structured to accept expanded machine models if they are required.

Section VII, on validation, includes a discussion of the accuracy of this switching machine model.

5.3 *Trunk Group Model*

The probability that an attempt will be accepted on a trunk group is found by assuming that the arrival process is Poisson. The required result is the Erlang B function, which depends only on the mean number of calls which would be on the trunk group if it were infinite and the actual number of trunks.

To find $P_{\overline{ij}}$ on the trunk group between i and j, we need the following definitions:

$N_{\overline{ij}}$ = number of trunks between SM_i and SM_j.
$E_{\overline{ij}}$ = expected number of calls on the trunk group between i and j.

$F_{\overline{ij}}$ = expected number of calls that would be on the trunk group if $N_{\overline{ij}}$ were infinite.

n_R = expected number of messages on complete route R.

S_r = expected number of calls on partial route r being processed in SM_ℓ where $r = (a \cdots \ell; b)$.

W_r = expected number of calls on partial route r waiting to enter SM_ℓ where $r = (a \cdots \ell; b)$.

$\frac{1}{\nu}$ = mean holding time for a message.

Then

$$E_{\overline{ij}} = \sum_{R \text{ over } \overline{ij}} n_R + \sum_{r \text{ over } \overline{ij}} S_r + \sum_{r \text{ over } \overline{ij}} W_r$$

$$= \frac{1}{\nu} \sum_{R \text{ over } \overline{ij}} x_R + \sum_{\text{all } TC_\ell} \{T_{W\ell} \sum_{\substack{r \in I_{\ell\_} \\ r \text{ over } \overline{ij}}} x_r + (T_C + T_P + T_{WM} + T_M) \sum_{\substack{r \in I_{\ell\_} \\ r \text{ over } \overline{ij}}} z_r\}$$

$$+ (T_{WM} + T_M) \sum_{\substack{r \text{ over } \overline{ij} \\ r \notin I_i \bigcup I_j}} t_r. \quad (15)$$

To find $F_{\overline{ij}}$, simply notice that $P_{\overline{ij}}$ is a linear factor of every x_r, z_r, and t_r in eq. (15). Given all the P's except $P_{\overline{ij}}$, choose any positive value for $P_{\overline{ij}}$, say $\hat{P}_{\overline{ij}}$, use eqs. (1) through (15) to find $\hat{E}_{\overline{ij}}$, the value of $E_{\overline{ij}}$ corresponding to $\hat{P}_{\overline{ij}}$. Then

$$F_{\overline{ij}} = \frac{\hat{E}_{\overline{ij}}}{\hat{P}_{\overline{ij}}}. \quad (16)$$

Finally

$$P_{\overline{ij}} = 1 - \mathrm{B}(N_{\overline{ij}}, F_{\overline{ij}}), \quad (17)$$

where B is the Erlang B function.

It is unlikely that the arrival process at each trunk group in the network is Poisson. In fact, much of the recent trunking analysis has been directed toward non-Poisson processes. However, the Poisson assumption is a reasonable one to make in a mean value model. The accuracy of the overall model will be discussed in Section VII.

VI. SOLVING THE EQUATIONS

In the last two sections the model was given as a set of equations, (1) through (17). Most reasonable uses of the model require simultaneous solution of the equations and then computation of quantities of interest from this solution.

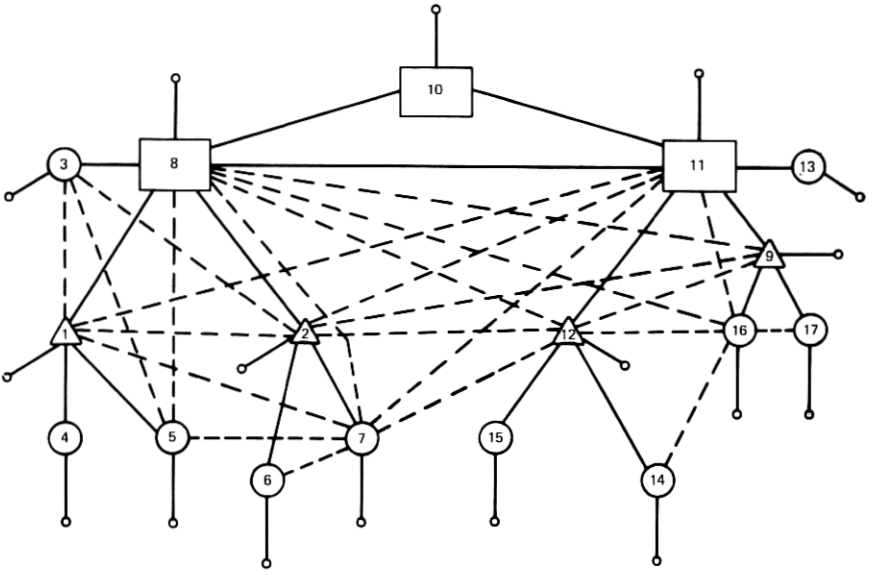


Fig. 5—Network used in point-to-point comparison run.

Solving these equations, at reasonable cost, is essential to the usefulness of the model. As given in Sections IV and V, there are a very large number of equations, both nonlinear and coupled. In the network shown in Fig. 5 are about 50,000 equations and variables. Fortunately, the P's and T_w's calculated in Section V form a fundamental set of variables from which everything else can be calculated. There are 91 of these variables for the network in Fig. 5. In another network of interest there are 240 of these variables.

The approach taken to solving the equations is iterative. Given a set of P's and T_w's, all other variables of Section IV are calculated. Then a new set of P's and T_w's are calculated. If the new set and the old set are the same, a solution has been found. More precisely, let $\mathbf{y}$ be a vector whose components are the P's and T_w's, then the equations of the model specify a function, $F(\mathbf{y})$, which gives the new value of P's and T_w's. In this framework solving the model equation is equivalent to solving

$$F(\mathbf{y}) = \mathbf{y}. \tag{18}$$

To make the solution of (18) easier, the components were normalized to the interval $[0, 1]$. The components corresponding to P's are necessarily in this interval. The components corresponding to T_w's were forced to be in the interval by replacing T_{w_i} by T_{w_i}/T. The re-

sulting function F maps $[0, 1]^n$ into itself, where n is the number of trunk groups plus twice the number of toll machines. The function is continuous so (18) has a solution by Brouwer's Fixed Point Theorem.[12] The question of uniqueness of the solution will be discussed later.

For this model to be a useful tool for network analysis, it is necessary to solve (18) inexpensively. There are two basic problems to be overcome. First, F is so complicated that, for reasonable size networks, its derivative is unavailable. This means that any method which requires F' cannot be used. Second, in these same networks a single evaluation of F costs on the order of \$0.50. The cost of estimating F' by n evaluations of F depends on n, the dimension of y. For the network in Fig. 5 it would cost about \$45 for a single estimate of F'.

In order to solve (18) economically, an algorithm which doesn't require F' or estimates of it was devised. The algorithm adapts the step size on the basis of the last ten evaluations of F.

The algorithm is as follows:

(*i*) Initialize $\mathbf{y}^0 \epsilon [0, 1]^n$ and set $i = 0$.

(*ii*) If $\|F\mathbf{y}^i - \mathbf{y}^i\| < \epsilon$, stop.

(*iii*) Otherwise,

$$\alpha_i = \min_{j \in I_i} \frac{1}{1 + \delta_j}$$

$$\mathbf{y}^{i+1} = \mathbf{y}^i + \alpha_i (F\mathbf{y}^i - \mathbf{y}^i),$$

where

$$I_i = \{j \mid j \text{ an integer} \geqq 0,\ i \geqq j \geqq i - 10\}$$

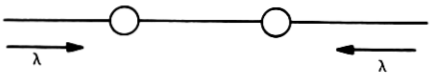
$$\delta_0 = 1$$

$$\delta_j = \frac{\|(F\mathbf{y}^j - \mathbf{y}^j) - (F\mathbf{y}^{j-1} - \mathbf{y}^{j-1})\|}{\|\mathbf{y}^j - \mathbf{y}^{j-1}\|}, \qquad j > 0.$$

(*iv*) Repeat (*ii*) with $i = i + 1$.

A computer program was written to evaluate $F(\mathbf{y})$, i.e., eqs. (1) through (17), and to implement the algorithm for solving (18). Our experience with the program has been that the algorithm usually reaches a satisfactory solution in less than $n/2$ steps. The cost of the program depends on the network size and on the value of various calling parameters. However, the cost for the network shown in Fig. 5 was usually around \$10, with some runs as high as \$40. For a larger network with 240 variables, the cost was usually around \$20.

An important point to mention is that F is not a contraction mapping. This means that the existence of a unique solution cannot be



N_S = 14 λ = 1 CALL/SEC

T_P = 2.5 SEC PR = 0

T = 30 SEC PA = 1

Fig. 6—Example network which has two quasi-stationary solutions.

guaranteed. In fact, in one example, two solutions to (18) were found. The question immediately arises, "Are both of these solutions physically meaningful?" The answer is "Apparently, yes." The existence of two stable operating regimes for the toll network has been suggested before.[3] The argument for their existence is as follows: If large queues form before switching machines, other machines will have their holding times greatly increased and will be able to switch only a fraction of their usual capacity. All other calls will time out. That means, if somehow the queues become large, they might stay large and the network completion rate would be very low, while under the same system parameters if the queues ever became small, they would stay small and the network completion rate would be much larger.

To test this argument, J. A. Kohut's Monte Carlo Simulation[5] of the network shown in Fig. 6 was run. A brief description of this simulation is given in Appendix B. The system started empty and was run for one hour of simulated time, reaching a quasi-stationary condition within the first 10 minutes. Then the offered load was doubled for 10 minutes, causing large queues to form. After that, the simulation was run for one more hour at the original traffic level. Again it reached a quasi-stationary state within 10 minutes. The results are given in Table I. The results show conclusively, for this simulated network, that two quasi-stationary operating regimes exist. For comparison, the results from the analytic model are also included in Table I.

It is possible for this network to be operating in the uncongested regime, receive an unusually large number of calls during some short time interval, and go into the congested regime. In the congested regime, if an unusually small number of calls arrived for a period of time, the system could go into the uncongested regime. It seems reasonable, and the simulation run helps confirm it, that the mean time before spontaneously leaving one of the regimes is quite long. This is the reason for using the term quasi-stationary.

TABLE I—EXAMPLE SHOWING TWO QUASI-STATIONARY SOLUTIONS

Model	Mean Attempts (per 5 min.)	Mean Time-outs (per 5 min.)	Mean Waiting Time (in sec.)
Simulation			
10 to 60 min.	602 ± 9	0	0.17 ± 0.02
80 to 130 min.	605 ± 8	454 ± 11	21.5 ± 0.2
Analytic			
Solution 1	600	10^{-9}	0.22
Solution 2	600	459	22.7

The existence of two quasi-stationary operating regimes apparently has implications for network management. If the network is congested, it may not be due to high calling rates but only due to high sender queues. A control which clears out these queues may be enough to decongest the network. Short sender timing which is currently used in the network is such a control.

VII. MODEL VALIDATION

The model discussed in previous sections contains many important network features. The machine model includes the stochastic arrival of attempts, office work times depending on waiting for adjacent offices, and time-outs. The trunk model includes stochastic arrivals and holding times, with the mean holding times dependent on how far the calls progressed toward becoming messages.

In the final analysis, the model stands or falls by how well it predicts the operation of a real network under overload. While from a validation viewpoint this could best be done by comparing the model with a real situation, there are two good reasons not to do so. First, the data collection problem would be extremely difficult and prohibitively expensive. Second, getting meaningful comparisons would require allowing the network to operate in an unacceptable mode. In addition, any real network would include things not modeled here, such as other types of switching machines and additional network management controls.

An alternative is to compare the model with a Monte Carlo simulation which is currently being used to evaluate network controls. While this comparison cannot evaluate the modeling of effects treated similarly in both models, it does help evaluate the modeling of effects treated differently. We compare our model with Kohut's simulation.

It also contains a simplified machine model, but does not include our simplifying assumptions on how senders wait for senders, that all arrival processes are Poisson, or that machine holding times are exponentially distributed. These assumptions are perhaps the most suspect in our model.

Two kinds of comparison between the two models were carried out. The first compares gross behavior over a very large range of offered loads. The second looks at more detailed statistics under a reasonable overload. In both cases, the results are similar.

The first type of comparison was carried out on the network in Fig. 7. The two models were given exactly the same data on machine sizes, pulsing times, trunk group sizes, etc. A series of runs was made with the only change between runs being the calling rates. The first run used a nominal set of calling rates, the second used twice the nominal calling rates, the third used three times the nominal calling rates, etc. For each run, the Monte Carlo simulation was run until, on the basis of the retrial rates, it appeared to be in steady state. It was then run for another one or two simulated hours to estimate the expected number of messages in progress in the network in steady state. The sample variance was used to estimate the 68 percent confidence interval for the mean. The analytical model was then used to find the expected number of messages in progress for each offered load. Figure 8 shows the curve generated by the analytic model as well as the Monte Carlo simulation's estimates of the corresponding means and confidence intervals.

In Fig. 9, exactly the same runs were made as in Fig. 8, except that switching machines timed out after 5 seconds in Fig. 8 and after 30 seconds in Fig. 9. Monte Carlo runs for more than double the nominal calling rates were not made in the 30-second case, since the simulation was not intended to handle the very large queues that would develop.

From Figs. 8 and 9, it appears that the models predict the same behavior for the expected number of messages in progress over a very large range of calling rates. The numerical values given by the two

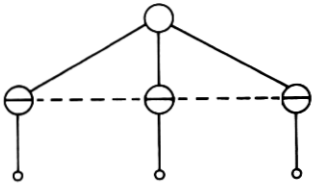


Fig. 7—Network used in massive overload comparison runs.

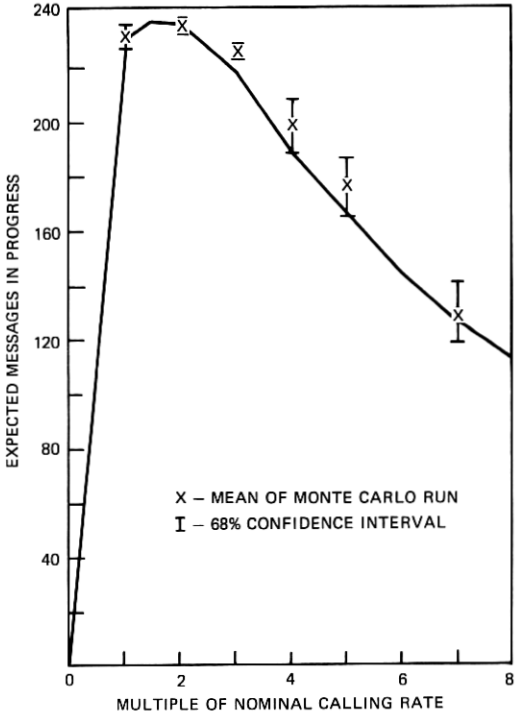


Fig. 8—Carried vs offered load, time-out = 5 seconds.

models also seem to agree well. At three times the nominal calling rate, the two means differ by less than 4 percent in Fig. 8.

The previous examples were generated for the network in Fig. 7. The second type of comparison was made between the Monte Carlo simulation and the analytic model on the network in Fig. 5. This network configuration was used in early network management simulation studies. While it is similar in structure to the toll network, it has one less level of hierarchy.

In order to get reliable statistics, the Monte Carlo simulation was run for three simulated hours. The network appeared to have reached equilibrium by the end of the first hour. Statistics were printed out at 10-minute intervals for the next two hours, and these were used to estimate completion probabilities, expected sender attachment delay, and the probability a call would time out in each switching machine.

Table II shows the comparison of the expected sender attachment delay and probability of time-out given by each model for each of the

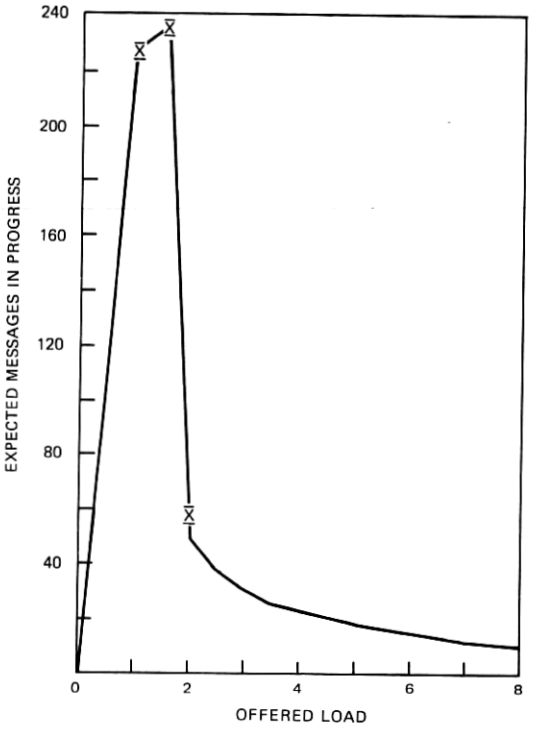


Fig. 9—Carried vs offered load, time-out = 30 seconds.

17 switching machines. For the Monte Carlo run, estimates of the standard deviation in the estimates are also given. It can be seen that the analytic model tends to give larger numbers for both quantities than does the Monte Carlo simulation.

To estimate the completion ratio for each point-to-point pair, the total number of attempts and completions were recorded for each pair for the last two simulated hours of the Monte Carlo run. The estimate of the completion ratio from source i to destination j is

$$\hat{C}R_{ij} = \frac{C_{ij}}{A_{ij}}$$

C_{ij} = number of completions from i to j

A_{ij} = number of attempts from i to j.

This estimate of the completion ratio was used since it has a smaller variance than would result if the completion ratio was calculated for each 10-minute interval and then averaged. The corresponding com-

TABLE II—COMPARISON OF ANALYTIC AND MONTE CARLO MODEL RESULTS FOR SWITCHING MACHINES

Switching Machine	Attachment Delay (in sec.)		Probability of Time-out	
	Analytic	Monte Carlo	Analytic	Monte Carlo
1	0.33	0.19 ± 0.03	0.008	0.008 ± 0.002
2	0.12	0.08 ± 0.01	0.002	0.002 ± 0.001
3	0.30	0.20 ± 0.03	0.016	0.012 ± 0.002
4	0.25	0.16 ± 0.03	0.013	0.007 ± 0.002
5	0.29	0.21 ± 0.03	0.015	0.012 ± 0.002
6	0.18	0.12 ± 0.02	0.008	0.007 ± 0.002
7	0.10	0.04 ± 0.02	0.004	0.001 ± 0.001
8	0.79	0.31 ± 0.03	0.011	0.009 ± 0.002
9	0.49	0.19 ± 0.03	0.016	0.007 ± 0.002
10	0.15	0.07 ± 0.01	0.005	0.003 ± 0.001
11	3.04	2.03 ± 0.07	0.219	0.163 ± 0.009
12	1.83	0.90 ± 0.06	0.085	0.054 ± 0.006
13	0.46	0.32 ± 0.04	0.027	0.020 ± 0.003
14	0.17	0.07 ± 0.02	0.008	0.002 ± 0.001
15	0.18	0.12 ± 0.02	0.008	0.007 ± 0.002
16	0.19	0.08 ± 0.01	0.009	0.003 ± 0.001
17	0.12	0.04 ± 0.01	0.005	0.002 ± 0.001

pletion ratio calculated by the analytic model will be denoted CR_{ij}. To compare $\hat{C}R_{ij}$ with CR_{ij}, it is necessary to have an estimate of the standard deviation of $\hat{C}R_{ij}$. This estimate was made as follows: If the actual completion ratio really is CR_{ij} and if the probabilities of completion are independent for successive ij attempts, then given the number of ij attempts, the number of ij completions is a binomial random variable. Therefore, $\hat{C}R_{ij}$ has mean and standard deviation

$$E[\hat{C}R_{ij}] = CR_{ij}$$

$$\sigma_{ij} = \sqrt{\frac{CR_{ij}(1 - CR_{ij})}{A_{ij}}},$$

respectively. The assumption that successive completion probabilities are independent is not unreasonable, since in this network the trunk groups between a typical ij pair will have an average of 10 to 50 message completions between successive ij attempts.

To conveniently compare CR_{ij} and $\hat{C}R_{ij}$, consider the standardized random variable

$$\xi_{ij} \triangleq \frac{\hat{C}R_{ij} - CR_{ij}}{\sigma_{ij}}. \tag{19}$$

Figure 10 has a histogram of the ξ_{ij}'s for 72 arbitrarily chosen ij pairs.

In the pairs plotted, 68 percent of the ξ_{ij}'s were in $[-1, 1]$, 92 percent were in $[-2, 2]$, and 100 percent were in $[-3, 3]$. This is consistent with the above assumptions. If they were correct, the expected percentages would be approximately 68, 95, and 99.7 percent, respectively.

To get a quantitative estimate of the difference in the completion ratios given by the two models, we require an estimate of $E[\xi_{ij}]$. To get such an estimate, treat the ξ_{ij}'s as independent, identically distributed random variables. Then, using the 72-pair sample to estimate $E[\xi_{ij}]$ and the standard deviation of that estimate gives

$$E[\xi_{ij}] = 0.026 \pm 0.143. \tag{20}$$

This is consistent with $E[\xi_{ij}] = 0$. However, using the estimated mean allows us to estimate the relative error, ϵ_{ij}, between the two models.

$$\epsilon_{ij} \triangleq E\left[\frac{\hat{C}R_{ij} - CR_{ij}}{CR_{ij}}\right]. \tag{21}$$

From (19),

$$\epsilon_{ij} = \frac{\sigma_{ij}E[\xi_{ij}]}{CR_{ij}}. \tag{22}$$

Using $E[\xi_{ij}] = 0.026$, ϵ_{ij} was calculated for all 72 point-to-point pairs. All the calculated values were in (0.0017, 0.0051) and the average was 0.0033. This gives the estimated relative difference in the completion ratios calculated from the two models as 0.33 percent. This error is negligible for practical purposes.

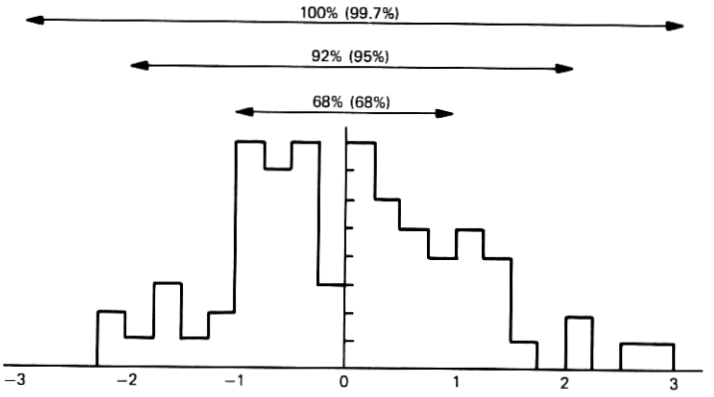


Fig. 10—Relative frequency of ξ_{ij}'s for 72 point-to-point pairs.

From the examples given in this section, it seems reasonable to conclude that

(*i*) The two models agree well when computing network quantities such as point-to-point completion ratios and the expected number of messages in progress.

(*ii*) They show somewhat less agreement in local phenomena such as the sender attachment delay and probability of time-out in individual machines.

VIII. CONCLUDING REMARKS

This paper presented an analytic model of the response of a telephone network to overloads. The model agrees well with a currently used simulation for network quantities such as point-to-point completion ratios and the expected number of calls in progress in the network. It is much cheaper to use, with typical costs being \$20 versus \$200. The model only gives quasi-stationary results. To get the transient response, a Monte Carlo simulation must be used.

The overall model structure permits changes in the model of individual components. Expanding the No. 4A switching machine model, including other types of switching machines, and including additional network management controls might be useful.

IX. ACKNOWLEDGMENTS

Many people have contributed ideas to this work. We wish particularly to acknowlecge contributions by Jack Holtzman, Sheldon Horing, Edwin Messerli, Irvin Yavelberg, Pat Spagon, Dave Jagerman, and George Hallas. We also take this opportunity to thank Elizabeth Murphy for early programming assistance, and Ellen Hill for producing the final program. Finally, we wish to thank John Kohut for help in connection with his Monte Carlo simulation model of the network.

APPENDIX A

Introducing Some Network Management Controls into the Model

For ease of exposition, eqs. (2) omitted network management controls. These equations must be modified to include network management controls corresponding to switching machine code blocking on the basis of destination, skip routing, cancellation of alternate routing from a trunk group, and cancellation of alternate

routing to a trunk group. To see how these controls are included, let

$$r = (a,i;d)$$
$$r_1 = (a,i,j;d)$$
$$r_2 = (a,i,k;d).$$

Partial route r enters switching machine i. Partial route r_1 is the preferred route out of machine i toward the destination d. Partial route r_2 is the alternative to route r_1. Also let

NCB_{id} = one minus the fraction of calls code blocked in switching machine i because they have destination d

NCF_{ij} = one minus the fraction of calls cancelled because they attempt to alternate route from trunk group ij

NCT_{ik} = one minus the fraction of calls cancelled because they attempt to alternate route to trunk group ik

NSK_{ik} = one minus the fraction of calls which skip over trunk group ik when alternate routing.

In terms of these symbols, the replacements for eqs. (2) are

$$\begin{aligned} x_{r_1} &= P_{ij} z_r NCB_{id} \\ x_{r_2} &= P_{ik} NSK_{ik}\{NCT_{ik} NCF_{ij}(z_r NCB_{id} - x_{r_1})\}. \end{aligned} \tag{2'}$$

The interpretation of the equations is as follows: The calls entering SM_i on route r which are not code blocked are offered to trunk group ij. They are accepted with probability P_{ij}. The quantity in parentheses corresponds to calls which are neither code blocked nor accepted into route r_1. The quantity in brackets corresponds to calls which are not cancelled because of alternate routing controls. Of those calls, the ones which do not skip trunk group ik and do find a free trunk enter route r_2. Those calls which do skip trunk group ik or find it full will be offered to the next alternate route, if one exists.

APPENDIX B

A Brief Description of the Monte Carlo Simulation in Reference 5

The Monte Carlo simulation in Reference 5 is a call-by-call simulation in the sense that it generates calls individually and processes them through the simulated network as individual entities. That is, each run of the simulation of Reference 5 produces a realization of the underlying stochastic process as opposed to the model presented here which analytically arrives at statistics for that process. The remainder of this

appendix describes the assumptions and treatments used in the simulation.

The underlying traffic between each source-destination pair is a Poisson stream. Any attempt which reaches its destination end office has a fixed probability of failing because of a "don't answer" or "busy" condition. Any attempt which fails to become a message, for any reason, will retry with a fixed probability. If a failed attempt will retry, the time until retrial is chosen from an exponential distribution. The conversation length for each successful attempt is also chosen from an exponential distribution.

An attempt which arrives at a trunk group can seize a trunk only if one is free at that time. Once a trunk is seized, it is held while the attempt progresses through the network. If the attempt fails, the trunk is released at the time of failure. If the attempt becomes a message, the trunk is also held for the duration of the conversation.

The simulation contains several switching machine models, only one of which was used in the comparisons in this paper. It consists of two groups of parallel servers: the first models the senders, while the second models the common control responsible for translation, trunk testing, and switching. We will refer to the first group as the senders and the second as the markers.

An attempt bids for a sender, if one is seized, then a constant delay is introduced to represent receiving digits. If a sender is not seized within a specified time, the attempt abandons the queue. After a sender has received the digits, a bid is made for a marker. If one is available, it is seized and held for a constant holding time. If one is not available, the sender will wait for a marker. During the marker operation, a test is made for a free trunk. If no trunk is available, the call is immediately blocked and the sender and all prior seized trunks are released. In the simulation, it is assumed that announcements and reorder tone do not extend the holding time of blocked attempts.

After the marker holding time, the sender bids for an attachment to a sender at a distant machine. This bid will result in either an attachment of a sender or an intersender time-out. In the former case, the sender is held for an additional constant length of time which simulates outpulsing the digits. In the latter case, no out-pulsing occurs, but an additional marker usage is required to route the attempt to an announcement.

The queuing discipline for senders and markers is random. When a piece of common control becomes free, a bid is selected at random from the bids waiting. The simulated switch of an attempt through a

switching machine may encounter delay in four different ways. An attempt will be delayed during the sender and marker service times and may be delayed by waiting for these pieces of equipment if they are not available at the time of the bid. The service time delays are fixed, so these delays are equal for all attempts. However, the delays caused by queuing are random and are dependent upon how long an attempt must wait for equipment to become free.

REFERENCES

1. Wilkinson, R. I., "Theories for Toll Traffic Engineering in the U.S.A.," B.S.T.J., *35*, No. 2 (March 1956), pp. 421–514.
2. Katz, S. S., "Statistical Performance Evaluation of a Switched Communications Network," Fifth International Teletraffic Conference, London (June, 1967), pp. 566–575.
3. Helly, W., "Two Stochastic Traffic Systems Whose Service Times Increase With Occupancy," Operations Research, *12* (1964), pp. 951–963.
4. Szybicki, E., "Approximate Method for Determination of Overload Ability in Local Telephone Systems," Seventh International Teletraffic Conference, Stockholm (June, 1973).
5. Kohut, J. A., unpublished work (see Appendix B above).
6. Franks, R. L. and Rishel, R. W., "Optimum Network Call Carrying Capacity," B.S.T.J., *52*, No. 7 (Sept. 1973), pp. 1195–1214.
7. Cardwell, R. H., unpublished work.
8. Hallas, G. A., unpublished work.
9. Gnedenko, B. V. and Kovalenko, I. N., *Introduction to Queueing Theory*, Israel Program for Scientific Translation (1968), pp. 33–39.
10. Descloux, A., *Delay Tables for Finite- and Infinite-source Systems*, New York: McGraw–Hill (1962).
11. Little, J. D. C., "A Proof of the Queueing Formula: $L = \lambda W$," Operations Research, *9* (1961), pp. 383–387.
12. Dunford, M., and Schwartz, J. T., *Linear Operators, Part 1*, London: Interscience (1958), p. 453.