MODELING THE
IRS TELEPHONE TAXPAYER INFORMATION SYSTEM

CARL M. HARRIS and KARLA L. HOFFMAN
George Mason University, Fairfax, Virginia

PATSY B. SAUNDERS
National Bureau of Standards, Gaithersburg, Maryland
(Received June 1984; revisions received April 1985, September 1986; accepted April 1987)

The Internal Revenue Service (IRS) toll-free, nationwide telephone system provides prompt tax-information assistance. In 1986, the IRS processed 37.8 million calls from taxpayers at 32 answering sites. This paper documents a critical review of the IRS approach to allocating its staff and equipment. We built a simulation-based model to test various allocation policies for deploying IRS resources. The simulation study included detailed sensitivity analysis on key network variables, and showed the feasibility of modeling a typical IRS location as a multiserver loss/delay queue with retrial and reneging. The second phase of this effort therefore centered around developing a prototype probabilistic model for determining the most effective way of providing service at reasonable levels and at minimum cost. The resulting model allows the IRS to determine from tables the best configuration of people and telephone lines for any expected levels of incoming traffic. In addition, we provided flow balance analyses of the underlying feedback queues that permit the IRS to separate their caller streams into fresh and repeat callers, and thus to estimate actual demand for service.

The studies documented in this paper and in Harris (1982) and Hoffman and Saunders (1981) were undertaken by the Internal Revenue Service as part of a broad effort to learn how best to deploy its resources, within the budget provided by Congress, to meet taxpayer demands for tax law information.

In 1965, the IRS began initial attempts to offer taxpayers toll-free telephone assistance from centralized locations. In 1972, the agency developed guidelines for a nationwide system, and in January 1974, implemented nationwide toll-free service at 135 answering stations (eventually consolidated to 32). In fiscal year 1986, the IRS responded to 37.8 million calls from taxpayers.

The combined results of the sources just mentioned and the research described in this paper have provided the IRS with an automated, rational mechanism for determining the allocation of its scarce resources. The problems the agency faced are not unique to it; other federal, state and local governments have similar information dissemination systems. For example, the Veterans Administration and the Social Security Administration use toll-free telephone systems to provide benefits information.

The telecommunications literature contains tables for calculating the number of trunklines and servers required for a given level of service. However, these tables provide reasonable approximations only when the system is operated at a level at which few calls overflow and waiting times are short. Indeed, when we began our project in 1979, virtually all commercial telephone customer-service systems—for example, airline reservations, customer feedback systems, and mailorder purchasing—operated in such a manner, since the operators of those systems believed that if customers did not receive service immediately, they might call a competitor.

The airline industry is currently struggling to remain profitable in a decontrolled environment, and has reevaluated its policy of instant service. One often finds all circuits busy when calling for airline reservations, and the waiting time before receiving service has increased as well. It has become clear to that industry, as to many others, that staffing and circuiting a reservation system to ideal service levels is extremely costly. We believe, therefore, that the models described in this paper—high utilization, multiserver, low-delay queues with retrial and reneging—represent a major advance in queueing technology and have the potential for wide application.

This project was begun during the Carter presidency and completed during the Reagan administration. The latter, taking the position that taxpayer information should be provided by the private sector and pur-
chased by taxpayers, has initiated major cutbacks in the taxpayer assistance program. Person-to-person contact with taxpayers to provide general tax information has been reduced, and such contact is now limited to the more complicated aspects of the tax code. Currently one can obtain, through an automated (no human interface) telephone system, recorded messages detailing basic aspects of the tax law, or request tax forms by calling various numbers.

Congress has continually resisted such decreases in light of great public interest in the program and a sense of the IRS's obligation to taxpayers. However, because of budget restrictions and administrative uncertainty, the IRS has not yet applied the modeling techniques described in this report as widely as the agency and we had hoped.

Tax reform has created great need for tax information assistance. The model we describe could be extremely useful in situations in which staffing levels must be altered significantly (either increased or decreased) in response to any variation in caller demand. We are confident of the ultimate usefulness of our work in support of IRS resource allocation decisions; discussions are now underway on its eventual implementation.

Meanwhile, one aspect of this project has in fact been completely implemented. The IRS bases its estimates of actual demand on the centers on our queueing flow-balance formula, and the agency uses these projections to justify budget requests to Congress and to measure network quality of service.

1. Introduction to the Problem

The objective of the toll-free telephone system (TFTS) of the Internal Revenue Service (IRS) is to enable citizens to receive tax-information assistance by placing a call, either locally or long distance, for a charge no greater than that for a local call. This system is not a standard reservation type, in which all agents may receive minimal training and generally are expected to possess little technical expertise. In fact, the opposite is generally the case. The range of inquiries is exceedingly vast and requires a level of expertise that ranges from minimal to highly trained and specialized. The telephone system, while nationwide, is not interconnected: each individual answering site, reached by phone numbers advertised to potential callers in a variety of ways, services a prescribed geographic territory.

Each answering site must be furnished with telephone lines (trunks) that are capable of receiving both local and long-distance calls. The number of trunks required depends on the number of incoming calls, the duration of the calls, and, indirectly, the number of servers on duty.

If a taxpayer calls the local answering site while all the trunks corresponding to this type of call (local vs. long-distance) are in use, he or she receives a busy signal. Calls that result in a busy signal are labeled overflows. Such a caller can choose to redial immediately, wait and try again later, or merely decide to find the information elsewhere. Those callers who do not again try to use the IRS telephone system will be referred to as lost demand.

If, however, a line is free, the call is received at an automatic switchboard, called an Automatic Call Distributor (ACD), which distributes the calls to the answering agents. If all of the answering agents are busy when the call reaches the switching device, the caller receives a recorded message that all servers are presently busy, and the call is delayed until an attendant becomes free. "Abandoned calls" or "reneges" are calls which hang up after reaching the ACD but prior to service. A caller who reneges may repeat the attempt to obtain service or may choose to seek assistance elsewhere.

A caller reaching an IRS Taxpayer Service Representative (TSR) may ask for specific tax-law clarifications, may have questions about some correspondence received from IRS, or may be requesting tax forms. After receiving this assistance, the caller hangs up, thereby freeing a trunk line for other incoming calls. The server, however, may have additional work to perform associated with this service (such as filling out a forms-request) and therefore may have some "after-call" (or "after-hang-up") work that must be completed before another caller can be assisted.

The ability of such a system to handle the demand is dependent upon both the number of trunks for each group and the number of attendants. In similar systems operated by commercial enterprises, such as hotel or airline reservation systems, poor service may mean lost business; therefore, both staffing levels and trunking allocations have historically been chosen so that few calls overflow and the average wait before service is short. Only recently have commercial reservation systems altered this policy. In the case of the IRS system, however, a public service is being provided for which there is high demand (during the April 1982 filing season, for example, over 40,000 calls per day were received at the site that serves the Baltimore-Washington area) and for which the cost/benefit relationships of alternative possible levels of service
quality are difficult to ascertain. But, due to limited funds, the centers are not staffed or equipped so amply that the service is comparable to that provided by commercial enterprises such as hotel and airline reservation systems and instant-charge outlets.

The specific objective of this project was to provide the IRS with efficient ways for determining the allocation of staff and equipment to meet as much demand as possible within their budget. They must determine how much of this budget should be allocated to equipment costs and to what extent adding more staff, alternating the schedules of current staff and consolidating offices could alleviate some of the overloading. The IRS understands that, given their current budget, managers of this system must balance two distasteful service characteristics: blocking customers from entering the system or having customers experience long waits. In Sections 2.5 and 2.6, we will further discuss the trade-offs involved in this choice.

We initially chose a simulation model. Its purposes were (1) to describe and understand the system, (2) to answer "what if" questions, and (3) to train managers regarding the scheduling of such centers. This model illustrated that the system could be modelled analytically as a multiserver loss/delay queue with retrial and delay. Sections 5, 6, 7 and 8 describe the queueing model and its uses.

2. The Process Being Modeled

2.1. Reaching the Telephone-Information Center

Calls for taxpayer assistance come to an IRS Taxpayer Service Telephone Information Center from a telephone company central office via bundles of lines called trunks. There are three types of telephone circuits that enable taxpayers to call "toll-free." They are:

(1) *Local Circuits*. Local circuits are those telephone circuits whose use will incur only local-call costs.

(2) *Foreign Exchange (FX)*. These circuits are lines that can be used to extend the IRS local-call-cost concept to population centers distant from the local answering site. The taxpayer dials a 7-digit number and the call is treated on his bill the same way as any local call. The charge to the government agency for these lines is based primarily on the distance between the local calling area and the answering site. (The agency is charged on a per-line basis, not on a per-call basis.)

(3) *Wide Area Telephone Service (WATS)*. This system is a network of long distance circuits that are accessible to all callers in widely defined (state/region) geographic areas. Callers who attempt to access the WATS (800-series prefixed) numbers will have the calls charged to them as if they were local calls. The agency is charged on both a per-line and per-unit-time-of-call basis for these calls. The rate for these calls is significantly higher than for FX lines.

A question of immediate initial interest to the IRS is how many lines of each of these trunk types it must provide in each of the information centers. These lines must be "hard-wired" into the answering site by the telephone company. Local telephone companies have differing regulations governing how often and at what cost these trunk configurations can be changed. For the IRS, they are rarely changed more than once a month during the filing season, and even less often during the remainder of the year.

When a caller dials the IRS taxpayer service number, he or she reaches the local telephone switching center, which connects the call with an IRS trunk line if one of the type the caller needs (local, FX, WATS) is available. The search for a trunk line always begins with the first line of that type and continues in sequence until an available line is found. The lines are indexed or labeled serially; thus, for example, WATS trunk line 11 will never be used unless WATS lines 1–10 are already in use.

Once a call finds a free trunk line, the call is routed to a telecommunication switching device that automatically distributes incoming calls to telephone work stations. There are a variety of Automatic Call Distribution (ACD) devices, which are either mechanical or electronic. Both of these systems can be connected to automated management information systems that present a variety of reports on trunk usage and attendant performance.

2.2. Answering the Taxpayer's Questions

A Taxpayer Service Representative (TSR) who answers these calls will perform one or more of four possible tasks:

(1) Send a tax form,

(2) Explain or clarify some correspondence with the taxpayer,

(3) Retrieve and evaluate the taxpayer's account, and/or

(4) Provide an explanation of one or more elements of the tax code or tax form.
Questions requiring retrieval of a tax account or referral to someone with a more in-depth understanding of the tax law can be handled in three ways: (1) the taxpayer may be placed on hold while the server researches the question; (2) the taxpayer can be transferred to some other server, thereby freeing the original server but maintaining the use of the same trunk line; or (3) the server may phone the taxpayer with the answer at a later time.

Once the caller has completed the conversation and hangs up, the trunk line is again available to receive an incoming call. However, the server may have after-call work related to that call (such as filling out a form request or researching the tax records of the caller). The server will not be available to handle another call until the after-call work is completed.

2.3. Statistics on Call Activity

All of the switching devices used by the IRS have reporting systems that collect (usually hourly) summary data that indicate the level of service being provided. These measures include average length of time a call waits before being connected to a server, average talk duration, number of calls received (number of calls that gained access to the ACD equipment and receive the first ring—this first ring signifies to the ACD that the call is now “in queue” awaiting a server), average holding time (includes both the wait time and the talk duration), number of calls serviced (calls that actually received service), and number of calls abandoned.

The demand to the system varies within any given day, within any week, and also has significant seasonal variations. Within a day, there are usually two peak time periods, one occurring between 9:30 and 11:00 A.M. and the other between 1:30 and 3:00 P.M. The periods of smallest demand are the first half hour (8:30–9:00 A.M.) and last hour (4:00–5:00 P.M.) of the day, with a less pronounced slack period during lunch hour. Within the week, traffic is heaviest on Mondays and declines as the week progresses. Annually, the heaviest volume occur during the “tax season”—January 1 through April 15—with the heaviest weeks occurring at the end of January (when most taxpayers receive information documents such as Form W-2) and the week before April 15 (the last allowable filing date of all tax forms). The slowest period occurs in late fall when demand is typically less than 5% of that during the peak seasons. These trends are, in general, consistent among all TFTS sites.

The questions received by each site range from extremely simple ones to some that require that the respondent have a thorough understanding of complex tax law. Call length varies by type of inquiry, and can be as short as 20–30 seconds or as long as 20 minutes.

2.4. How Can This Complex Process Be Modeled?

Decisions regarding the amount of service to provide, in terms of both Taxpayer Service Representatives and trunk lines, must be made regularly at each of the TFTS sites. However, it is generally impossible to predict accurately when callers will call to request service and how much time will be required to provide that service. The decision process is made even more difficult by budget limitations that further restrict the possible decision alternatives so that any affordable choice involves some balance between two distasteful service characteristics: blocking customers from entering the system, or having customers experience long waits. The major questions relevant to this modeling process include:

(1) How do arrivals occur? Does the distribution of arrivals vary over time? Do many people who have either received a busy signal or have renegoted redial immediately? How do these calls affect the arrival distribution?

(2) What is the distribution of service times? Does its mathematical form vary with the arrival rate?

(3) Does the queue discipline affect the overall service provided? Can a change in the queue discipline reduce costs to the Internal Revenue Service?

(4) How do circuitry configurations affect the overall service provided?

(5) If budget constraints rule out the totally satisfactory situation in which, almost invariably, calls are neither blocked from the system nor have long waits before service, then how should the limited budget be allocated among staff and equipment? Is it preferable to block a high percentage from the system, thereby allowing successful calls to reach an agent immediately, or to block a minimum number of callers, with the result that the waiting times may be extremely long, with lost-call rates correspondingly high?

2.5. Existing Solution Methods

Normally in the telephone traffic literature (Clos and Wilkinson 1952; Finefrock 1974a, b; Goeller 1975;
Gordon and Jewett 1976; Jewett 1976, 1977; Jewett, Shrago and Gordon 1976; Jewett and Holland 1977; Mina 1971a, b, 1972, 1974; Molina 1927), the determination of the number of trunks required for a given level of service uses methodology quite different from that used to determine the number of servers needed. An Erlang-B loss system (i.e., \( M/G/c/c \)) is used to model the queueing process related to trunk-line calculations, while a Markovian delay system (i.e., \( M/M/c/\infty \)), generally referred to as the Erlang-C system, is used to model that process related to server requirements.

The IRS has in fact used the \( M/G/c/c \) loss model to represent callers attempting to reach trunk lines. The calls attempting to enter the system are the inputs, and the distribution over time of call arrivals is assumed to be Poisson. The trunk lines at the IRS office are the "servers"; service is considered to begin when (and if) the call is connected to an available trunk line, and to be completed when the caller hangs up (either because the caller became impatient and chose to leave before being connected to a Taxpayer Service Representative (TSR) or because he or she received the assistance requested). If insufficient trunk lines are available to handle the demand, the caller receives a busy signal (i.e., the caller is blocked from entering the system) and is assumed to redial at some later time when congestion is relieved. The Erlang-B loss formula is used to determine the number of trunk lines needed to reduce the blockage probability (or factor) \( B \) to a prescribed acceptable level. The main problems with the procedure are that it implicitly assumes that the number of trunk lines can be calculated independently of the number of TSRs assigned and that it mostly ignores the effect of customer waiting, which in reality is highly dependent upon the number of servers available. Whittaker (1974, 1975) showed that calculating the number of trunk lines needed to each trunk group type (WATS, FX, and local) separately distorts the predictions for both the number of calls blocked (due to insufficient trunk lines) and the overall average delay.

The use of the formula also implicitly embodies another important assumption: that callers will not redial immediately but will wait until the system is less congested before again attempting to obtain entry. If this is not the case, then the Poisson arrival process on which the formula is based may not properly describe the mix of incoming "first-time" calls and reattempted calls, even though it accurately describes the arrivals of first-time calls alone.

The Erlang-B assumption would be better justified in situations where customers receiving busy signals try their calls later during off-peak times. Unfortunately, the IRS has limited office hours (daytime only) and few time slots during the entire filing period can be considered "off peak." Many taxpayers, therefore, have no choice but to wait or try again during another overloaded time.

In the telecommunications literature, an Erlang-C Model is used to determine the number of TSRs. Here, the "servers" are the TSRs, the queue length is the number of callers connected (assigned to a trunk line) and waiting to talk to a TSR, and service does not begin until the caller is connected to a TSR. In these models, arrivals who find all \( c \) servers busy join a queue with an unlimited number of waiting positions. The arrivals wait in queue as long as necessary for service. The arrival process is again assumed to be Poisson and the service times are exponentially distributed with service time including both talk time and after-talk time. The probability that an arrival is delayed is given by the \( M/M/c/\infty \) delay formula or Erlang-C formula (see Cooper 1981 or Gross and Harris 1985 for a derivation), where \( c \) is the number of servers (TSRs).

The problems with using the Erlang-C formula for calculating the number of servers are more serious than those described for the trunk calculations. The Erlang-C formula is based on the following assumptions, among others: (1) the mean arrival rate \( \lambda \) is strictly less than the mean system service rate \( \mu c \) (for stationarity), and (2) there is an unlimited number of waiting positions (so that no calls are blocked). Obviously, the IRS situation violates both of these assumptions. The number of TSRs is \textit{not} in fact always sufficient to serve all calls, and the load \textit{can} exceed the number of trunk lines so that some calls are blocked. In an effort to correct for these violations, analysts typically use the C formula for this type of problem, with \( \lambda \) replaced by an adjusted arrival rate \( \lambda' = \lambda(1 - B) \), where \( B \) is the blocking factor. Using this adjusted arrival rate in the formula assures that the arrival rate is strictly less than the service rate, and thus avoids the problem of having the queue length build to infinity.

However, to be able to use the Erlang-C formula in this manner, one must already have determined the blocking factor \( B \). But this factor can be calculated only if the number of trunk positions is known, which in turn requires knowing the number of TSRs. Use of this formula also assumes that anyone entering the system is willing to wait as long as is necessary to obtain service, and will not renege.
2.6. Evaluation of Erlang-B and Erlang-C Formulas

Given these limitations and inconsistencies in using the Erlang-B and -C formulas as just described, it is natural to wonder why the industry employed them. The answer, quite simple, is that they are very good approximations for any system that can afford to operate at a level for which few calls receive busy signals (and thus the unlimited queue length assumption holds) and delay times are short (and therefore very few people choose to renege). In this situation, the mean arrival rate is almost exclusively determined by new arrivals and, since the number of redialers, is small, ignoring the redials does not significantly alter the results. Furthermore, easily implemented queueing formulas are rare for more general multi-server systems.

However, the fact is that Congress does not allocate sufficient funds for this service for a staffing level that can provide full, prompt response to nearly all demand. When demand overburdens the staffing level, lost calls and reneging occur and many of these callers redial. If trunking is in excess of that required for a given staffing level, then calls can enter the system whenever a trunk line is available. These calls must then often wait a very long time before being served, all the while tying up trunk lines and causing other calls to be blocked from the system. In this situation, the reneging rate, the lost call rate, and the average wait time for callers are all high.

On the other hand, one can choose to reduce the number of trunk lines and transfer resources into manpower. In the extreme case with only as many trunk lines as servers, no one would wait before being served. However, a high percentage of the callers would be blocked.

Thus, the IRS wishes to find some compromise position between blocking callers from the system and losing them once they reach the system. Originally, the IRS chose to follow the pattern set by telephone-communication engineers, and planned circuitry independently of staffing. The result was that circuitry was provided to allow more calls to enter the system than the staff could answer. The IRS has in recent years chosen to alter this approach, and aims now to build the circuitry system around the staff resources. This change reflects a conscious decision to have callers receive busy signals rather than experience prolonged waits when call demand exceeds available staff capacity. Acceptable service is defined so that callers who enter the system have average waiting times of at most 20 to 40 seconds and an average abandonment rate less than or equal to 5% on an hour-by-hour basis (i.e., 5 reneges/(100 departures = 5 reneges + 95 completions)).

As stated earlier, however, prior to this study, allocation decisions were based on the Erlang-B and -C formulas which involve a number of assumptions that are violated in the IRS situation. In light of these problems, we built a discrete-event simulation model of the IRS telephone information system. It allowed us to investigate the consequences of any violation in these assumptions. Ultimately, the simulation analysis indicated the possibility of constructing a closed-form queueing model, which we present in Section 5.

3. The Simulation Model

The model we developed for studying the IRS telephone system is descriptive rather than prescriptive. Thus, its scope and purpose is confined to measuring the system performance that might be expected if particular options were to be implemented. With such a model, the analyst must design and execute model run sequences that allow the evaluation and comparison of alternative system configurations.

We chose the simulation language SIMSCRIPT II.5 (CACI, Inc. 1976 and Kiviat, Villanueva and Markowitz 1973) for our simulation model. We have divided the process into the following components: the arrival process, the reneging process, the queue, and the service process. We will now describe the events that occur within each of these components and how, in general terms, these events were modeled in the simulation.

3.1. The Arrival Process

Phone calls arriving at the IRS office come from three sources: WATS lines, local lines, and FX lines. First-time call arrivals (all calls except redials) are expected to have a Poisson distribution. Research (beginning with Clos and Wilkinson 1952) as well as intuition substantiate this assumption, since arrivals in a telephone system cannot "see" the other arrivals and therefore should not be affected by the behavior of these arrivals.

However, this assumption seems inappropriate for redialed calls, since the arrival time of such a call is dependent upon the time of the previous attempt to reach the Center. In our model, both calls that are blocked from the system (due to a busy signal) and calls that are abandoned (reneging due to impatience) can redial. We estimated, after conversations with Bell Laboratories staff, that if a caller redials, the time it
takes for him or her to hang up and redial the number will average 200 seconds. Since there was no information about what proportion of the demand represents “redials,” we let the probability of redialing be an input parameter to the simulation model and therefore under the control of the user of this model.

3.2. The Probability of Reneging

Customer abandonment is an important parameter that has a significant effect on model results despite an inherent complexity involved in its estimation. Most typically, customers renege based on their subjective estimate of the amount of time remaining until their service is to begin (partly as a function of any prior call history such as overflow experience). The quality of this estimate is very much a function of the information available to the customer. The usual approach is to base one’s decision on either the size of the queue immediately in front or the delay to date (these numbers are on the average related). The former information is never available to an IRS caller, but of course the latter is. However, human behavior is such that a very long wait often keeps the caller on the line because of an increased expectation and an already incurred overhead (though we know, of course, that future delays in an M/M/c queue are independent of already-spent waiting time).

The alternative approach was to have the simulation program check all callers in the queue periodically to determine whether or not they would remain holding until the next periodic check (assuming they were not served in the interim). Different hold-on probabilities could be assigned for those having waited at least 15, 30, 45, 60, 75, 90, 105, 120, 135, and over 135 seconds, respectively. In the simulation model we implemented both of these approaches (that is, renege based on queue size and renege based on accumulated waiting time), and the runs showed that the system is indeed sensitive to renege probabilities.

3.3. The Queue

Electronic ACD devices allow the IRS the flexibility to choose among a variety of queue disciplines—FIFO, LIFO, WATS before FX, and so forth. The AT&T mechanical device has been designed to provide service that is “approximately FIFO.” A requirement for the approximation to hold is that the three trunk-group types be dealt into positions in the sets alternately: that is, one WATS, then one FX, then one local, then the next WATS, FX, local, and so forth. When there are unequal numbers of trunks in the trunk groups, the dealing should place lines alternately until an entire trunk group is exhausted, then alternately among the two remaining groups, and then all lines of the last group are assigned to the remaining set locations in consecutive line number order.

Since the Baltimore TFFTS site was chosen by IRS staff as the test center for validating our simulation model, and since that site uses an AT&T mechanical ACD, we examined the physical wiring of its trunk lines.

We (and others) were surprised to find that trunk lines in Baltimore are not wired according to the dealing procedure just described. Instead, since trunk lines come cabled, with a number of trunk lines (all of the same trunk-group type—e.g., WATS) residing in a cable, the local telephone company chose to wire the trunk lines as follows: first place all WATS lines sequentially into as many sets as is required, then follow with all FX lines, and finally all local lines.

Having determined that the Baltimore Center’s wiring was inconsistent with the FIFO-approximating wiring directions of Bell Laboratories, we decided to provide the user of our simulation model with the ability to test the effects of using a mechanical device configuration similar to that used in Baltimore.

Alternatively, the user of this model can choose two other queue disciplines: a “pure” FIFO queue discipline or a priority discipline. With the priority option, whenever there are calls in the system that have waited longer than 30 seconds, the call waiting the longest is chosen for service. When there are no such calls, the longest waiting WATS call is chosen for service. If there are no WATS calls waiting, then the longest waiting FX call is handled. Finally, if there are no FX calls waiting, then the longest waiting local is handled. This priority rule for choosing among waiting calls is a constrained cost-minimization approach. One chooses to have the calls that have the highest hang-on cost per second handled first, but only if handling them does not create severe inconvenience by raising the wait of other callers excessively.

Our simulation studies examined each of these queue-discipline options. Subsequent analysis showed that the priority rules were never invoked due to the long waits of all customers. A true FIFO queue discipline does improve service over the mechanically wired “round-robin” approach.

3.4. The Service Distribution

As discussed in Section 2, the “service time” in the Erlang-C delay formula includes both the time the TSR spent talking to the taxpayer (talk-time) and the time spent performing after-call work (after-hang-up time). Furthermore, we have to assume that
this total service time (talk-time plus after-hang-up time) has an exponential distribution.

In order to evaluate the appropriateness of this assumption, we collected data on both talk-time and after-hang-up time of calls received in the Baltimore Center. The data were tabulated on a call-by-call basis for each call received in the Baltimore office during two days of heavy traffic in December. We then analyzed these data to determine what distribution best fit the data. (Available data were not sufficient to determine if there were distributional differences among the three different trunk types.) Since the data were skewed and looked "almost" exponential, we tried) a variety of right-skewed distributional forms. These included the gamma distribution (with the exponential as a special case), extreme-value type I and II distributions, lognormal and Weibull distributions. We used Filliben's DATAPAC (1977) statistical computer package for these analyses. We found the talk-time and after-hang-up time data not to be distributed exponentially. Rather, the distributions that best approximate the data were Weibull distributions with shape parameters equal to 0.8750 and 0.5305 for talk-time and after-hang-up time, respectively.

We obtained the same results when we analyzed each day's data separately. Based on goodness-of-fit tests (minimum chi-square tests and K-S tests—from Chandra, Singpurwalla and Stephens 1981), we do not reject the hypothesized Weibull distributions at the 0.05 level.

Pooling talk-time and after-hang-up time data into one overall service time also yielded a Weibull distribution with shape parameter equal to 0.832 (not very different from that of the talk-time distribution).

One use of the simulation model is to test the effects of violation of the assumptions underlying the Erlang-C formulas. Talk-time can be represented as either Weibull distributed (with shape parameter 0.875) or negative exponentially distributed (with any user-specified positive value of \( \mu \)), and after-hang-up time as a Weibull (with shape parameter 0.5305). To test the effect of combining talk-time and after-hang-up into one distribution (as must be done for the Erlang-C formulas), the user can merely sum the talk-time and after-hang-up time. This total time should then be input as talk-time, the talk-time distribution specified to be exponential, and the after-hang-up time specified to be zero. Thus, the simulation model allows the user to test the sensitivity of model results both to changes in the service-time distributions or to changes in the scale parameter values of a given distribution.

Most important, the simulation results showed a high level of insensitivity to using an exponential distribution with the same mean as the Weibull with shape 0.832. We should not be surprised by this result since the entire system is running in heavy traffic, and the coefficient of variation of the combined Weibull is near one.

This section has presented each of the major components of the simulation model (see Hoffman and Saunders for a complete description of the computer code). The next section presents the results from a number of "runs" of this computerized model. The first part of that section presents our validation tests, while the latter portion presents the results of altering specific input conditions.

4. Validating and Executing the Simulation Model

4.1. The Validation Effort

Law (1979b) stated that "one of the most important problems facing a real-world simulator is that of trying to determine whether a simulation model is an accurate representation of the actual system which is being studied." In this section, we briefly review the techniques he suggested and describe our validation efforts with reference to these approaches.

Debugging the Code. Law began his paper by presenting five techniques for debugging a computer program. They are: (1) write and debug the program modularly, (2) have more than one person read the computer code, (3) develop "trace" routines that allow one to see the program path during execution of the program, (4) run the model with simplifying assumptions for which the true characteristics of the system are known, and (5) when possible, display the simulation outputs on a graphics terminal as the simulation actually progresses.

We attempted to adhere as closely as possible to these guidelines in developing and debugging our simulation model. The code was written and debugged in a modular fashion. Every line of code was read by at least two people, and the code was fully "commented." We wrote debug routines that allow the tracing of every event and caller. SIMSCRIPT II.5 provides additional automatic error-trace information. Finally, we ran the model with simplifying assumptions (adequate trunks, no renegoting, Poisson arrivals and exponential service times) and the simulation results were consistent with analytical results for the M/M/c/\( \infty \) queue.

Validation of the Model. One cannot speak in terms of absolute validity for a model like ours, but rather of the degree to which the model approximates what the IRS believed to be the most significant aspects of
their system. However, since many aspects were approximated (e.g., all TSRs treated as working the entire hour with the number of servers per hour being the rounded average of what actually occurred), and since some aspects of the system remain unknown (e.g., the actual distribution of redials or reneges), the model can at best be only a rough approximation of the system. The question then becomes whether the model is “close enough to reality” to provide the decision maker with the type of information needed. Law presented a staged approach toward “validating” a simulation model. We will discuss our validation efforts with respect to this criteria, the first of which is to develop a model with “high face validity, i.e., a model which, on the surface, seems reasonable to people who are knowledgeable about the system under study.”

We made our most extensive validation efforts in this area. We discussed the system and its decision problems with Taxpayer Service Staff both in the national office and in Baltimore, and included all aspects of the process they believed to be important. We collected and analyzed data on reneges, talk-time, and after-hang-up time from the Baltimore TFTS site and included those approximated distributions in our model. Finally, for system components for which information was scanty or unavailable, the user may vary alternatives so as to make these unknowns consistent with the intuition of the decision makers. By allowing this flexibility, one can test against real occurrences and calibrate the model for different sites or different scenarios.

We next checked the model outputs against those observed in the actual system. To perform this type of testing, we collected data from the Baltimore TFTS site during a key part of the filing period. Since the last week in January and the second week in February were considered similar, we averaged these two weeks on an hourly basis by day of the week and used the results for validation comparisons. (We will refer to this validation as Period 1 data.) Similarly, the weeks of March 7, March 14, and March 28 were averaged and used to compare model outputs to known outcomes. (This data will be referred to as Period 2 validation data.)

Table I presents the validation results for the busiest day (Monday) of Period 1. Similar tables for each of the other days of the two validation periods can be found in Hoffman and Saunders. We note that this output is based on modeling the system under the assumption that no one redials (either after receiving a busy signal or after reneging). The reason for making this assumption is that we had data that specified the number of call attempts. We could use these data as the fresh demand into the system only if no one redialed. If redials did occur, the demand (first-time calls) would have to be adjusted downward according to the redial distribution. Since redials affect only demand—and no other aspect of the outputs of the model—we felt more confident of model output if

| Table I |
| Validation of the Model Using Baltimore Center Data—Period 1 Simulation Model Results<sup>a</sup> (Actual Baltimore Data<sup>b</sup>) |

<table>
<thead>
<tr>
<th>MONDAY</th>
<th>Avg. No. of Call Attempts</th>
<th>Ave. No. of Calls Offered</th>
<th>Ave. No. Calls Abandoned</th>
<th>Ave. No. of Overflows</th>
<th>Ave. No. of Calls Handled&lt;sup&gt;c&lt;/sup&gt;</th>
<th>Ave. Delay Per Handled Call (in seconds)</th>
<th>Ave. Delay Per Abandoned Call (in seconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>752 (745)</td>
<td>745 (668)</td>
<td>29 (78)</td>
<td>7 (77)</td>
<td>703 (549)</td>
<td>17/81 (31.5)</td>
<td>25/25 (15) (35.7)</td>
</tr>
<tr>
<td>2</td>
<td>3463 (3496)</td>
<td>1552 (1336)</td>
<td>207 (104)</td>
<td>1911 (2161)</td>
<td>1316 (1173)</td>
<td>109/105 (83.5)</td>
<td>56/61 (63) (58)</td>
</tr>
<tr>
<td>3</td>
<td>4044 (4087)</td>
<td>1662 (1384)</td>
<td>193 (47)</td>
<td>2382 (2703)</td>
<td>1476 (1337)</td>
<td>80/80 (62.4)</td>
<td>47/47 (34.3)</td>
</tr>
<tr>
<td>4</td>
<td>4661 (4652)</td>
<td>1305 (1199)</td>
<td>226 (125)</td>
<td>3356 (3453)</td>
<td>1067 (996)</td>
<td>130/127 (125.3)</td>
<td>80/80 (85) (109.8)</td>
</tr>
<tr>
<td>5</td>
<td>4641 (4670)</td>
<td>1165 (997)</td>
<td>248 (153)</td>
<td>3476 (3673)</td>
<td>908 (839)</td>
<td>207/209 (208.5)</td>
<td>104/105 (144.1)</td>
</tr>
<tr>
<td>6</td>
<td>4544 (4552)</td>
<td>1459 (1294)</td>
<td>218 (139)</td>
<td>3085 (3258)</td>
<td>1258 (1173)</td>
<td>121/121 (106.8)</td>
<td>64/61 (55) (80.3)</td>
</tr>
<tr>
<td>7</td>
<td>3777 (3737)</td>
<td>1413 (1354)</td>
<td>205 (220)</td>
<td>2364 (2383)</td>
<td>1198 (1089)</td>
<td>134/131 (106.4)</td>
<td>67/69 (60) (49.4)</td>
</tr>
<tr>
<td>8</td>
<td>3420 (3442)</td>
<td>1666 (1523)</td>
<td>173 (183)</td>
<td>1753 (1919)</td>
<td>1511 (1358)</td>
<td>78/78 (54.3)</td>
<td>43/46 (45) (29.4)</td>
</tr>
<tr>
<td>9</td>
<td>1775 (1811)</td>
<td>1012 (1099)</td>
<td>164 (161)</td>
<td>763 (712)</td>
<td>877 (934)</td>
<td>137/141 (131.6)</td>
<td>70/69 (72) (39.0)</td>
</tr>
</tbody>
</table>

<sup>a</sup> The model was run with 10 different pseudo-random-number seeds, and the numbers here are the means of the 10 runs, averaging about 31,000 fresh arrivals per run. One computer run consisted of an entire day.

<sup>b</sup> An “actual Baltimore data” point is an average of the results recorded in the Baltimore Center by the automated information system for each of the three weeks within the period that data were collected.

<sup>c</sup> The reason that number of calls handled does not equal number of calls offered minus number of calls abandoned is that some of the offered calls were not handled during the first hour. End-of-period effects explain these apparent discrepancies.

<sup>d</sup> The number of local, WATS, and FX lines available during this period were 43, 10, and 48 trunks, respectively.
we kept the redialing probabilities set equal to zero. Anyone using this simulation code can, however, alter this input parameter and examine how redials affect demand.

Comparing the simulation output to the actual occurrences indicates that the simulation model's outputs do closely resemble actual behavior. But there are a few noticeable "nonmatches." The simulation model consistently predicts that about 10 to 12 percent more calls will be both offered and handled than actually took place. The IRS staff, when told of this inaccuracy in the model, explained that a mechanical system can take a significant amount of time (averaging around 6 seconds, but possibly as high as 20 seconds) either to switch servers to callers or to search for a free trunk line for an incoming call. The model assumes that these searches will be performed instantaneously.

We asked if the model should be corrected to reflect more accurately the time it takes for a mechanical device to perform these switching operations. The IRS staff explained that they would prefer the model to represent electronic systems (which can search and switch instantaneously) since they intend to use those devices in the future. We recommend that a similar validation effort be performed for a center that uses an electronic device.

As one final validation check, we performed a limited sensitivity analysis by running a variety of scenarios. These led to the following findings for the period tested: (1) the model is rather insensitive to the number of trunk lines available (a 10% decrease in trunk yields only about 2% decrease in the number of calls handled); (2) the model is "moderately" sensitive to the number of servers on duty, the average talk time, and the average after-hang-up time (e.g., increasing the number of servers by 20% increased the number of calls offered and handled by about the same percentage); and (3) the model is sensitive to the probability structure assumed for reneging (e.g., doubling the probability of reneging changes all other output statistics so that the model no longer reflects the behavior observed in the Baltimore office). When we derived the queueing-based model, we therefore performed a more complete analysis of abandonments and reneges (see Section 5.3).

Although this discussion has indicated needs for further validation research, we believe the model does represent the process being studied sufficiently well to warrant its use in investigating certain types of questions. Since simulation models are generally better at comparing alternatives rather than determining absolute answers, a model that is sufficiently valid to be applied to compare, in a relative sense, a number of proposed alternatives may not be accurate enough to determine quite as precisely the actual result of one of those alternatives (Law 1979a, p. 9). The results we present next are those of comparison; the most sensitive parameters (such as redial and renge distributions) are held invariant throughout.

Execution of the Model—Results of Altering Input Specifications. Our next step was to evaluate how model outputs would be affected by alterations in the input conditions. Specifically, we wished to examine a number of assumptions that are typically involved in deriving analytic queueing results but which we know are violated in the IRS system, and to determine whether the model outputs are significantly sensitive to these violations. We omit the validation tables from this paper; all data are presented in Hoffman and Saunders and can be obtained from the authors.

We posed the following questions:

1. Does the system change so dramatically from hour to hour and even minute to minute that the assumption of stationarity is violated?
2. Does the after-call work affect the results of the simulation (i.e., does the fact that lines become available before servers do significantly affect either the time-in-queue or the number of overflows)?
3. Are model outputs sensitive to the presence of nonexponential talk-time and after-hang-up time distributions, or can exponential times be substituted?
4. How do the outputs differ when the queue discipline is altered from FIFO to either the mechanized "non-FIFO" (as used in Baltimore) or the priority queue?

4.2.1. Stationarity. If the queueing system we have described is nonstationary, then there would be no possibility of ever obtaining closed-form analytic queueing formulas. We chose the fourth hour of the day for the investigation of stationarity since it was, for most days, the busiest hour and was also the hour used by the IRS in their decisions regarding the number of trunk lines to employ. Our intent was to estimate the rate of buildup to full capacity and steady-state conditions. Three simulation runs with five replications within a run were executed with a simulation "day" of only two time periods, the first time period (used to build up initial conditions for the second hour) having five, ten, or fifteen minutes, respectively. This first period had input qualities identical to those of the second hour period, i.e., the demand, talk-time, number of servers, and other input
parameters, were characteristic of the fourth hour of the day. The outputs from this limited experiment of three runs were compared to the output from those obtained during the validation stage in order to see if input conditions of the previous hour seem to affect the next hour's output.

The differences among the results of these three runs and that from the validation run are not statistically significant at the 0.05 level although they do exhibit a consistent bias. Thus, it appears that at least for times when the system is congested, it builds to its capacity almost immediately (certainly within 10 minutes) and remains at that capacity-limited level for virtually all of that hour. In light of these results and the fact that traffic overloads the system at all periods of the day, the assumption of stationarity seems valid.

4.2.2. After-Call Work. The queueing models considered for telecommunication systems incorporate all after-call work as part of the talk-time, thereby ignoring the fact that lines become available before servers. This situation led us to pose the following question: how many more calls can enter the system due to the extra time the lines are available and how much additional delay (and, indirectly, reneging) is thereby generated? To answer this question, we added the after-hang-up time to the talk-time and modeled this total time as talk-time, setting the after-hang-up time to zero (keeping all other parameters set at those used in the validation runs).

The data indicate that modeling after-hang-up time as talk-time lowers the number of calls offered and raises the number of calls abandoned, as expected. However, the increase in number of calls offered is only 2%. Since this change is not significant enough to warrant adding that complexity to an analytical model, we recommend using models that have after-hang-up time incorporated into talk-time. We note that research at the Bell Laboratories (Lyon and Rath 1978) confirms this approach. That research led to the recommendation that only when after-hang-up time is at least 50% of talk time should one model the after-hang-up time component as a separate entity. (In the IRS situation, talk-time averages between 160 and 200 seconds per call, while after-hang-up time is on the average only between 8 and 14 seconds per call.)

4.2.3. Nonexponential Service Times. As explained in Section 3, talk-time data and after-hang-up time data collected in the Baltimore office could best be represented by Weibull distributions. We employed the simulation model to test if replacing the sum of those Weibull distributions with an exponential distribution would significantly alter any of the system's performance measures (number of calls handled, delay per handled call, and so forth). The results of using either an exponential distribution to represent the combined talk-time and after-hang-up time, or for the talk and after-hang-up times are similar enough to indicate that the system is not very sensitive to a violation of the exponentiality assumption. Thus we recommended the use of models with one exponential service distribution.

4.2.4. FIFO vs. Priority vs. “Mechanical” Queue Disciplines. Our final test runs examined whether the queue discipline would have any effect on system performance measures. Queueing theory results indicate that the queue discipline will affect only the distribution of the waiting time of calls. Indeed, our simulation results were consistent with this finding. The average number of calls handled, offered, and overflowed all remain unchanged under any of these alternative queue disciplines.

There was virtually no difference between the results for the priority and FIFO queue disciplines. The system we are modeling is sufficiently congested so that calls wait on the average much more than 30 seconds. Therefore, the priority rule is almost never used. There are slight differences between the “mechanical” system and a pure FIFO system. As expected, the pure FIFO queue discipline does have wait times distributed more evenly among WATS, FX, and local calls.

4.2.5. General Conclusions from This Testing. The conclusions from our simulation testing are that a simpler model—one that assumes exponential service, combines talk-time and after-hand-up time and has a FIFO queue discipline—can safely replace the more detailed representation of the system. Can one then use the existing telecommunications formulas or rely on other classical queueing models? Unfortunately, the answer to that question is no. Callers do renge, and callers who either receive a busy signal or renge do often redial. We know, through runs of the simulation model, that model outputs are very sensitive to both reneging and retrial. This conclusion is disheartening in that less is known about these two aspects of the system than about any others.

When this study began, it did not seem at all obvious to us that it might be possible to derive analytic queueing formulas that would accurately represent the IRS system. The analyses from our simulation results have indicated that such a model is possible. Stationarity is a reasonable assumption (at least for a congested system), as are the assumptions of FIFO queue disciplines and of exponential service times which includes after-call work.
5. Building a Queueing Model

The conclusions of the simulation suggest that the underlying complexity of a Taxpayer Service Center may be well represented (in the sense of matching its measures of effectiveness) by a multichannel queueing system with the following basic characteristics. It begins from a Poisson input stream feeding one queue and serviced by numerous independent and identical exponential servers. There is a limited waiting capacity (say, of total size $K$ customers) and any customer finding all waiting spaces filled cannot enter and must either redial or opt not to return. Customers enter service without priority in a first-come, first-served manner (FIFO), and may decide to abandon before service. Any caller temporarily lost because of a full system or as a result of a renge may decide to return after some time has passed, or never to retry (to seek help elsewhere or maybe just forget the problem completely). Since our model permits waiting but does have losses, it is a combined loss/delay system with retrials.

Since service in the midst of congestion is the major problem, it should be understood that we are presenting a model for periods of heavy traffic, though standard heavy traffic results really do not directly apply since this kind of finite-capacity system will have a steady-state limit for any stationary arrival process, independent of relative intensity. Fortunately, as we hoped, this model is indeed solvable, and thus offers an excellent opportunity for a more satisfactory tool for supporting IRS decision making.

5.1. Interarrival Times

Throughout this project, we have assumed first-time arrivals (i.e., all calls except redials) of each of the three types (WATS, FX and local) are generated by separate and independent Poisson point processes (thus the sum of all types is also Poisson). This hypothesis is well-accepted in telecommunications and has been shown to be appropriate in numerous cases (for example, see Riordan 1962). Furthermore, the structural hypotheses leading to the Poisson process (for example, see Gross and Harris) are quite typical of the kinds of telephone call streams arising in these contexts. When the retrial stream is added to the underlying, virgin Poisson ($\lambda$) input of calls, the resultant process in the one the queueing system is actually offered. Hence we see the offered input occurring at rate $\lambda' > \lambda$. The actual carried input $\lambda'$ is less than $\lambda'$ and the actual input stream faced by the servers is no longer Poisson when the retrials are added.

5.2. Service Times

As noted earlier, the exponential distribution can safely be used for the probability structure underlying service times (when treated as talk-time plus after-hang-up time). This conclusion is quite important since it simplifies the analysis.

5.3. Reneging

In the IRS data, there seems to be little or no direct functional relationship between abandonment rates and queue sizes and/or delays. The percentage of reneges to total calls tends to be moderately small anyway. (The reason for this small abandonment rate may indeed have to do with the caller's past experience. Usually the system is sufficiently overloaded so that a caller may experience three or four busy signals before obtaining the prerecorded message. This experience induces the caller to not abandon when hearing the message.) We have therefore decided to assume a constant abandonment rate per customer within the same system configuration. We shall use the letter $g$ in the model to represent the individual reneging rate, and thus $[(m - c)g]$ is the system rate when there are $m-c$ callers in the queue. The actual numbers involved will be displayed in a later section.

5.4. Redialing

As we have already noted, the redialing phenomenon is a vital part of the problem and an issue that makes this problem different from the more usual ones. Our model views all potential retrials as members of a large, invisible population that ultimately feeds back most of its members to the main IRS input stream. The retrial population is a queueing system itself (with infinite number of servers). To specify its behavior fully, we must understand how its input arises, what the service process is, and so forth.

The retrial input stream is generated by overflows and abandonments. We shall assume that each potential retrial makes a decision (to reenter or otherwise) in a random amount of time, distributed exponentially with mean (say) $1/\nu$. The probability that the retrial decides to come back after an overflow or renge is to be denoted by $\alpha$ and thus the probability that he is lost forever is $(1 - \alpha)$, independent and constant across all customer retrials.

The assumption of exponential return times is quite consistent with research findings in the literature. For example, Wilkinson (1956) summarized a detailed study of general calling habits in New York City by saying that "the return times, after meeting 'busy,'
exhibit a marked tendency toward the exponential distribution, after allowance for a minimum interval required for redialing."

The ultimate estimation of the parameters \( \alpha \) and \( \nu \) is of course vital to the solution of the model. The former is estimated by a special analytic procedure from data collected by repeated sampling of a system through hourly periods of comparable configuration. An outline of this process is provided in Section 8 of this paper, and more completely documented in a project technical report (Harris). Unfortunately, values for \( \nu \) cannot be obtained so easily. There are no easy statistical techniques for this problem, so we instead make reference to the literature. Wilkinson (p. 431) claimed that the average return time after overflow was approximately 250 seconds, surprisingly close to the usual levels of mean (total) customer service times apparent in the IRS system. In light of this result and the fact that current dialing and switching is much faster, we have opted to equate the values of \( \nu \) and \( \mu \).

6. Derivation and Solution of the Queueing Model

By virtue of the assumed structure of the answering network, we may proceed to a mathematical representation for the probabilities associated with the queueing structure. This system can be solved for the probabilities, from which the usual measures of system effectiveness are obtained.

Recall that

\[
\begin{align*}
\lambda &= \text{virgin input rate} \\
\lambda' &= \text{actual carried input} \\
c &= \text{number of servers} \\
\mu &= \text{mean service rate} \\
g &= \text{individual renege rate} \\
\nu &= \text{reentry rate of retrials} \\
\alpha &= \text{probability of retrial} \\
K &= \text{system capacity}.
\end{align*}
\]

6.1. The Multiserver Queueing Model

We take advantage of the basic queueing property: when in system equilibrium, the mean flow into a state must equal the mean flow out. This result permits us to obtain system balance equations. The IRS systems are always operating in such heavy traffic that all callers either find all lines busy or at best must be put on hold. Thus the steady-state system size will never dip below \( c \) nor go above \( K \). Similarly, we assume that the retrial population is never nil.

We describe the relevant space of system states by the pair \((m, n)\): \( m \) indicates the number of the queueing system (including those being served) at an instant in time and \( n \) the (invisible) number of potential retrials. The (instantaneous) time rate at which the system leaves state \((m, n)\) is always (noting that \( c < m \leq K, n > 0 \))

\[
\lambda + n \nu + c \mu + (m - c)g,
\]

since (1) virgin arrivals occur at the rate \( \lambda \), either directly joining or balking into the retrial population; (2) retrial population members make their decisions at the rate \( n \nu \); (3) the system service rate is \( c \mu \); and (4) the individual renege rate is \( g \), thus yielding a system renege rate of \((m - c)g\). For \( m = K \), there are four potential nearby states from which \((K, n)\) can be reached, namely, \((K-1, n), (K-1, n+1), (K, n+1)\) and \((K, n-1)\). These four states correspond respectively to: (1) a virgin arrival; (2) arrival by retrial; (3) lost departure from retrial population; and (4) arrival found system full and entered retrial population. The transition rates from each of these are

\[
\begin{align*}
\lambda &\quad (K-1, n) \\
(n + 1) \nu \alpha &\quad (K-1, n + 1) \\
(n + 1) \nu (1- \alpha) &\quad (K-1, n + 1) \\
\lambda &\quad (K, n-1).
\end{align*}
\]

However, when \( c < m \leq K - 1, (m, n) \) can be reached from five adjacent states, namely, \((m-1, n), (m-1, n+1), (m, n+1), (m+1, n)\) and \((m+1, n-1)\). These five states correspond respectively to: (1) virgin arrival; (2) arrival by retrial; (3) lost departure from retrial population; (4) departure after service; and (5) renege that is added to retrial population. The transition rates are then

\[
\begin{align*}
\lambda &\quad (m-1, n) \\
(n + 1) \nu \alpha &\quad (m-1, n + 1) \\
(n + 1) \nu (1- \alpha) &\quad (m, n + 1) \\
c \mu &\quad (m+1, n) \\
(m + 1 - c)g &\quad (m + 1, n - 1).
\end{align*}
\]

Thus, two fundamental difference equations govern steady-state system behavior:

\[
\begin{align*}
(\lambda + n \nu + c \mu + (K - c)g)p_{K,n} \\
= \lambda p_{K-1,n} + (n + 1) \nu \alpha p_{K-1,n+1} \\
+ \lambda p_{K,n-1} + (n + 1) \nu (1- \alpha) p_{K,n+1} \\
+ (m + 1 - c)g p_{K,n-1} \\
(0 < n < \infty)
\end{align*}
\]
and

\[(\lambda + \mu n + (m - c)g)p_{m,n}
\]

\[= c\mu p_{m+1,n} + (n + 1)\rho(1 - \alpha)p_{m,n+1}
\]

\[+ (m + 1 - c)\gamma p_{m+1,n-1} + \lambda p_{m-1,n}
\]

\[+ (n + 1)\omega p_{m-1,n+1}
\]

\[(c < m \leq K - 1, 0 < n < \infty).
\]

We omit the state \(m = 0\) and assume it to occur with probability zero, since that is what happens in the IRS experience under the very heavy loads faced.

We should point out that problems somewhat similar to this one have appeared elsewhere in the open literature. Perhaps the most instructive are an illustrative example in the text by Riordan (pp. 94–96), material in Cooper (p. 82), and a paper by Gaver and Lehoczky (1976).

Riordan described a model of a \(c\)-server pure loss system with both exponential service and interarrival times. Demands not receiving immediate service attempt to reenter randomly at a constant rate, doing so until they are finally able to find an idle server. Riordan could find no simple solution to his less general model which allows neither waiting nor ultimate loss. A few approximations were offered, but no methods were suggested for complete analytic solution. Approaches for parameter estimation were also not discussed. Cooper, in turn, used the Riordan model as an exercise.

The Gaver and Lehoczky problem also did not have reneging or waiting, though it did permit eventual deflection. They used a diffusion approximation for the state probabilities derived as the solution to stochastic differential equations. We perceive this approach to offer no computational advantage over our model of the IRS environment.

The full solution is found by solving the simultaneous system of linear Equations 3 and 4 by varying \(m\) and \(n\), in addition to the observation that the sum of probabilities must be 1. In principle, \(m\) and \(n\) could possibly move over very wide ranges of values. However, the respective sizes of the parameters will force many of these probabilities to be effectively zero, thus considerably reducing the size of the resultant system of equations (certainly making the problem manageable). For example, Figure 1 presents the simultaneous equations in matrix form for a case in which \(m\) is restricted to be no lower than \(K - 1\) and an upper bound of \(N\) exists for \(n\). The structure of the matrix remains essentially the same even as the feasible number of states goes up.

Practical bounds for the ranges of \(m\) and \(n\) can be found by simplifying the actual network. For \(m\), we can approximate the observable system by an \(M/M/c/K\) queue whose arrival rate is given by \(\lambda_t = \lambda + \alpha R\), where \(R\) is the expected size of the retrial population. At first glance, it may seem that these parameters are all unobservable and that it is thus impossible to estimate \(\lambda_t\). Fortunately, such is not the case. First, the product \(R\) (the expected outflow from the retrial queue) may be estimated by the observed sum of overflow and abandonment rates. A later section details the estimation of the parameters \(\alpha\) and \(\lambda\). The originating input rate \(\lambda\) is also one of the three key system variables to be varied parametrically in the determination of a most appropriate local configuration, so we need not always know its value.

To bound \(n\), we have used a birth-death approximation for the retrial queue, with input and service rates given respectively as

\[
\begin{align*}
\lambda_i &= \lambda + i\omega \\
\mu_i &= i\mu
\end{align*}
\]

The \((i, j)\) location in the matrix of the example in Figure 1 is found as follows. (Remember for this case that \(m\) may take on only the values \(K\) and \(K - 1\).) First, we recognize that we need not incorporate any \(p_{m0}\) since it is impossible for the retrial population to be totally vacuous during any period of heavy usage. As noted above, there is a practical upper bound to the retrial population, which we shall call \(N\). Thus we are working with \(2N\) variables \((N\) associated with the calling system size \(K - 1\) plus another \(N\) with size \(K\)).

Hence, we need \(2N\) equations in these unknowns to create a solvable linear system. The first \(N\) of these come from Equation 4, with \(n\) allowed to vary from 1 to \(N\); the second set of \(N\) come from Equation 3. However, to avoid the totally null solution (created by an overdetermined system of equations), we must delete one of the \(2N\) equations (the very last one is usual) and insert the summability-to-one condition. As seen in Figure 1, the system of equations appears to have a very compact and manageable structure, with a relatively large number of zero coefficients. The structure of the original \(2N \times 2N\) matrix of Figure 1 prior to the replacement of the last row may be viewed as a \(2 \times 2\) array of \(N \times N\) submatrices as follows:

\[
\begin{bmatrix}
A_1 & A_3 \\
A_2 & A_4
\end{bmatrix}
\]

When \(m\) may also be \(K - 2\), the problem is expanded to a \(3 \times 3\) array of submatrices and the structure
variable equation for:

<table>
<thead>
<tr>
<th>$P_{K-1,1}$</th>
<th>$P_{K-1,2}$</th>
<th>$P_{K-1,3}$</th>
<th>...</th>
<th>$P_{K-1,N}$</th>
<th>$P_{K,1}$</th>
<th>$P_{K,2}$</th>
<th>$P_{K,3}$</th>
<th>...</th>
<th>$P_{K,N}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$-d_1$</td>
<td>$c_2$</td>
<td>$c_3$</td>
<td>$c\mu$</td>
<td>$G$</td>
<td>$c\mu$</td>
<td>$G$</td>
<td>$c\mu$</td>
<td>$G$</td>
<td>$c\mu$</td>
</tr>
<tr>
<td>$P_{K,1}$</td>
<td>$P_{K,2}$</td>
<td>$P_{K,3}$</td>
<td>...</td>
<td>$P_{K,N-1}$</td>
<td>$P_{K,N}$</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\lambda$</td>
<td>$b_2$</td>
<td>$b_3$</td>
<td>$\lambda$</td>
<td>$\lambda$</td>
<td>$\lambda$</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\lambda$</td>
<td>$\lambda$</td>
<td>$\lambda$</td>
<td>$\lambda$</td>
<td>$\lambda$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\lambda$</td>
<td>$b_N$</td>
<td>$\lambda$</td>
<td>$\lambda$</td>
<td>$\lambda$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\lambda$</td>
<td>$c_N$</td>
<td>$\lambda$</td>
<td>$\lambda$</td>
<td>$\lambda$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\lambda$</td>
<td>$\lambda$</td>
<td>$\lambda$</td>
<td>$\lambda$</td>
<td>$\lambda$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\lambda$</td>
<td>$\lambda$</td>
<td>$\lambda$</td>
<td>$\lambda$</td>
<td>$\lambda$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\lambda$</td>
<td>$\lambda$</td>
<td>$\lambda$</td>
<td>$\lambda$</td>
<td>$\lambda$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 1. Matrix representation of steady-state balance equations, where:

$$a_i = \lambda + iv + c\mu + (K - c)g \quad b_i = (i + 1)\nu\alpha \quad c_i = (i + 1)\nu(1 - \alpha)$$

$$d_i = \lambda + iv + c\mu + (K - 1 - c)g = a_i - g \quad G = (K - c)g.$$

represented as:

$$\begin{bmatrix}
A_1 & A_4 & A_7 \\
A_2 & A_5 & A_8 \\
A_3 & A_6 & A_9
\end{bmatrix}$$

with $A_2 = A_6$, $A_4 = A_8$, $A_3 = A_7 = 0$. Furthermore, the diagonal elements of the submatrices are nearly equal, while their off-diagonal elements are identical. In the event that the matrix should be larger, it is easy to generalize.

6.2. Computational Issues

Our solution method for the steady-state probabilities is most effectively illustrated with the $(2N) \times (2N)$ linear system of equations portrayed in Figure 1. In matrix-vector notation this system may be written as $[A]p = b$.

where $[A]$ is the $(2N) \times (2N)$ matrix of coefficients shown in the figure, $p$ is the vector of $2N$ probabilities we are to solve for, and $b$ is the right-hand-side vector of $2N - 1$ zeros, followed by a 1 in the final position. We outline now the procedure that is used for solution by the computer routine we have used.

First, we know that any matrix such as $[A]$ can be expressed as the product of two matrices in the form

$$[A] = [L][U],$$

where $[L]$ is a lower triangular matrix and $[U]$ is an upper triangular matrix with ones along the principal diagonal (i.e., $u_{ii} = 1$). Thus, $[A]p = b$ can be rewritten as

$$[L][U]p = b.$$ We next introduce the vector $q$ defined by $q = [U]p$. Then

$$[L]q = b.$$ Given the simple triangular structure of $[L]$, we can
easily calculate in succession the values for $q_1, q_2, \ldots, q_{2N}$. Finally, we form in (backward) succession the values of the $p$ vector. It is important to observe that whenever the same matrix has to be repeatedly re-inverted for different parameter values, the form of all these computations is easily recalled by the machine. Thus, subsequent run time is greatly reduced.

6.3. Measures of Effectiveness

This section has presented methodology for computing queue system probabilities subject to the assumption that the values of the parameters ($\lambda$, $\alpha$, $\nu$, $c$, $\mu$, $g$, $K$) are known (see Section 7 for a discussion of how they are estimated). Given queue system probabilities, any appropriate measure of system effectiveness can also be calculated. These would include such considerations as:

1. Mean number of customers actually served;
2. Expected delay for customers connected;
3. Mean total waiting time from call initiation to final completion;
4. Mean server utilization; and
5. Expected system losses.

We note, however, that the actual use of the model as a decision tool requires reversing the process somewhat. We realize that all the measures of effectiveness are themselves functions of the seven parameters. Thus, we generally set one measure at a target value, and then solve for the value of one of the seven parameters deemed adjustable. In principle, this approach can be extended to any number of the parameters, essentially adding one equation for each additional unknown.

The first and last items on the previous list are important measures of the level of service provided. They are indeed related to each other in the sense that the total number of potential customers asking for service is the sum of those ultimately served and those who try but are eventually lost to the system. One additional related and relevant measure of the quality of service is the percentage of calls not serviced (i.e., the overflows denied service by lack of space upon arrival or those who abandon even after waiting some time). These overflows and reneges are to be distinguished from those customers who never call back.

Measures of waiting time (items 2 and 3 on the list) are also important. The calculation of the total mean delay measure (item 3) is very complicated, since successive reattempts by any overflow customer are dependent events. So we shall instead concentrate on measuring system performance using $p_k$ (= probability of retrial), $\alpha$ (probability of retrial), and mean queue delay, $W_q$, of an arbitrary arrival.

We begin with Little's formula. We then obtain the mean delay by dividing $L_q$ by $\lambda'$ to get $W_q$.

The value $\lambda'$ is thus determined by the three parameters $\nu$ (the mean departure rate from the retrial queue), $R$ (the mean retrial population size), and $\alpha$ (the retrial probability). We set $\nu$ by subjective considerations as discussed earlier, and $R$ is found by the calculation.

$$R = \sum_{m=0}^{K} \sum_{n=0}^{N} np_{mn}. \quad (6)$$

The final parameter, $\alpha$, is found from detailed data analyses and a special procedure we developed especially suited for IRS data. Our experience suggests values of $\alpha = 0.7$. We offer an outline of our approach to its derivation in the next section.

7. Parameter Estimation

As should be fairly obvious from the previous discussion, the mathematical model can be used only when a variety of parameters are known. All of the devices used by the IRS have information-reporting systems that collect hourly summary data. These include average time a call waits before connection to a server; average conversation duration; number of calls received (i.e., those that gained access to the ACD equipment and received a first ring); average wait in system (i.e., line delay plus service time); number of
calls that actually received service; and number of calls abandoned. Additionally, most of the electronic switching systems used by the IRS have an overflow-measuring capability.

Thus, the estimation of most of the data needed is somewhat routine. As we have mentioned earlier, the major issues then left are estimating the loss and renge probabilities.

**The Loss Probability**

The key to obtaining an estimate for the loss probability is the steady-state balance of the flow into the main queue with the flow out. If we denote the overflow rate by \( f \) and the abandonment rate by \( a \), then we see that

\[ \lambda + (f + a)\alpha = f + \lambda'. \]  

This expression is essentially Equation 5, rewritten by recognizing that \( R_v = f + a \).

For each location, we collect data on the offered input rate \( \lambda' \), and on the overflow and abandonment rates. We shall further assume that the location operates under a constant configuration over a fixed period of time and that data are collected over periods of constant input rate and service behavior. Equation 7 then gives rise to a technique for estimating \( \lambda \) and \( \alpha \) (on an hour-by-hour basis for any specific location). Fortunately, the resultant estimates of \( \alpha \) consistently hover around 0.7.

The key observation is that Equation 7 implies that any changes in \( f \) and the rate of accepted calls \( \lambda' \) from one period to the next can arise from changes in \( \lambda \) and/or \( \alpha \). But both \( \alpha \) and \( \lambda \) are unknowns in this formulation. In order to initiate the iterative procedure, we have utilized some sample survey information from the IRS to estimate a relationship between \( \lambda \) and \( \lambda' \). Then, our technique is able to focus on \( \alpha \) only, and it proceeds in a fairly routine way.

**The Renenge Probability**

Direct analyses of the raw data for abandonments from a number of key IRS sites shows some fairly definitive patterns. First, there is always a very high rate of reneging over the first and last periods (8:30–9 A.M. and 4–5 P.M.). If we disregard these observations as nonstationary effects not relevant to the resource estimation, then we find that the percentage of waiting customers who renge before completion. This choice in turn implies that the system abandonment rate at full capacity can be written as

\[ (K - c)g = \frac{0.1}{0.9} \mu, \]  

since 90% of the output is now attributable to service completions. Thus, we see that

\[ g = \frac{c\mu}{9(K - c)}. \]  

We shall make special note of Equation 9, for it supplies the relationship between the individual abandonment rate and the system configuration. Since \( \mu \) is a constant, the rate \( g \) is thus directly proportional to the ratio \([c/(K - c)]\). In other words, the individual rate is inversely proportional to the ratio of the number of waiting spaces and directly to the number of servers (i.e., the more waiting spaces per server, the lower \( g \) is). For example, with 100 trunk lines and 65 servers, assuming \( \mu = 1/200 \) sec,

\[ g = \frac{(65)}{(9)(200)}(35) \]

\[ = 0.00103 \text{ reneges/sec} \]

\[ = 3.7 \text{ reneges/hour}. \]

8. **Using the Model**

The ultimate use of the model is for the development of resource tables for decision making. That is to say, given predetermined values (at common levels) for all system parameters, any of the system measures of effectiveness can easily be tabulated. Thus, it becomes possible to find a combination of trunklines and servers to provide any level of service desired.

Within the ranges of potential use, this analytic model appears to be relatively robust with respect to its assumptions. In any event, usage should generally be taken in a differential sense—that is, we want to know what is “best,” relatively speaking, and should not accept a single expected waiting time value as gospel. In fact, model validation strategies can be implemented to make sure that the model and its parameter values stay within the bounds of reasonableness.

It should be noted that any assumption about the loss probability \( \alpha \) combines with the probability of a full system \( p_x \) to give an estimate for the probability that any individual is eventually totally lost. We find
that

$$\text{Prob}[\text{eventual loss}] = 1 - \sum_{n=0}^{\infty} (1 - p_k) p_k^n \alpha^n$$

$$= 1 - \frac{1 - p_k}{1 - \alpha p_k}. \quad (10)$$

Thus, we can obtain an estimator for the true demand rate by combining Equation 10 with the observation that the number of service completions per unit time (call it $d$ for departure rate) must equal the virgin input rate multiplied by the fraction actually completed. So

$$d = \lambda \frac{1 - p_k}{1 - \alpha p_k},$$

and hence

$$\lambda = \frac{(1 - \alpha p_k) \lambda}{1 - p_k}. \quad (11)$$

Equation 11 can in fact be shown to be theoretically equivalent to Equation 5 by recognizing that because of equilibrium

$$\lambda' = d + \text{[abandonment rate]}.$$

In other words, given $\alpha$, either Equation 5 or 11 can be used to estimate $\lambda$. A comparison of the two possible $\lambda$ estimates can be used as a validity check.

**Steps in Use**

To close, we offer an outline of the sequence of operations necessary to establish useful tables, together with a glimpse at the ultimate appearance the tables will have. Recall that there are seven basic parameters for the model (namely, $\lambda$, $\alpha$, $\nu$, $c$, $\mu$, $g$, $K$). As indicated in our earlier discussions, we assume the value of $\nu$ to equal $\mu$. In addition, recall that we found the renege parameter $g$ to be a function of $\mu$, $K$, and $c$, as in Equation 9. Thus, we are left with but five parameters. Each possible set of five numbers leads to a unique value of any measure of effectiveness. In Section 6, we offered five possible measures. To tie the parameters to the measures of service, there would be two basic uses for the tables: (1) to determine the quality of service for an established configuration; and (2) to find the best configuration, subject to service and/or quality constraints.

Though the uses are different, they would come from the same tables, namely, the numerical computation of measures of effectiveness as functions of the five input variables.

The major step in the process then is the solution of the defining system of simultaneous equations as offered, for example, in Figure 1 according to the numerical procedure presented. To do this, we first input the five system parameters:

- $\lambda$: true arrival rate with value obtained from data analysis or prior estimates,
- $\alpha$: loss probability, with value obtained from data, typically between 0.6 and 0.8,
- $\mu$: average service rate, obtained from prior experience,
- $c$: number of servers, and
- $K$: total capacity of system (total number of trunk lines).

Solving the linear equations gives values for the system-state probabilities $[p_n, n = 1, 2, \ldots, K]$. From these, we compute the fundamental measures, $L_q = \text{average queue size and } W = \text{average line wait,}$ as per our earlier discussion. We performed a validation of the queueing model using the same Baltimore data used for validating the simulation model. The input parameters to the model were set as follows:

**Period 1**: $\lambda = 1000$ (virgin input)

- $\alpha = 0.7$ (loss probability)
- $\mu = (200 \text{ sec})^{-1}$ (service rate)
- $\nu = (200 \text{ sec})^{-1}$ (retrial rate)
- $K = 65$ (servers)
- $N = 100$ (trunk lines).

The matrix was fixed at $2,500 \times 2,500$, using $m = K$ and $K - 1$, and $n = 1, \ldots, 1250$.

**Period 2**: All input data are the same as in period 1 except for $\lambda$, which was set at 1,100. (Period 2 is heavier than period 1.)

The actual probability of a full system during Period 1 varied among hours of the day and days of the week, between 0.74 and 0.80. For the fourth period of the day (the heaviest period when $\lambda$ was approximately = 1,000), $p_k$ varied among days of the week between 0.77 and 0.78. The queueing model predicted $p_k$ of 0.784.

Similarly, for Period 2, the proportion of the time the system was full varied between 0.76 and 0.81 (and between 0.79 and 0.81 for the fourth period of the day). The model predicted $p_k$ of 0.793. These results are surprisingly good and indicate that the assumptions imbedded in the model reflect well actual behavior during the filing period for the Baltimore District.
9. Some Concluding Remarks

It is important to remember that the systems we have studied are almost always operating in very heavy traffic conditions. The typical fraction of callers finding all IRS trunk lines in a locale full may vary from a low of 0.4 during the nonfiling season to a high of 0.8 during the filing season, which is much higher than the usual commercial operation might be willing to tolerate. Obviously, this result may well induce strong behavioral reactions that our model has difficulty capturing. But the calculations can easily be repeated frequently, and data can be collected to track system performance. We thus view the queueing model discussed here as providing an important improvement in the service's ability to make appropriate trade-offs.

In light of the results documented in this paper, we made the following recommendations to the IRS:

1. Staffing and trunk configurations should be determined using the analytic model presented in this report.
2. Data on caller radial retrials should be collected and assessed periodically. This assessment is certainly required whenever major changes in tax law or tax administration procedures occur. (Recent demand into TSFS sites has increased due to computer processing problems and the new tax laws. As a result, we were asked to help the IRS perform another estimation study in late 1986 and early 1987. This work was carried out as part of a large system performance evaluation by the IRS, and confirmed the basic results of our first retrial estimation study.)
3. Make all computer programs well understood and operational at the service.
4. Use this package as a budget planning tool to evaluate and explain how minor shifts in the budget allocation can effect overall service in this important taxpayer service.

As we noted in the opening section, budgetary problems and philosophical differences with the Reagan Administration have delayed the complete implementation of our work. But the demand and retrial probability estimation portions have been fully incorporated into IRS decision making, and are now available even in desktop-computer versions. In addition, as stated earlier, we believe that these results may well be applicable to other government agencies and commercial enterprises that provide customer information via telephone lines. For example, the airline industry has recently chosen to allow some loss of service during heavy traffic periods in order to reduce the large costs of staffing and circuiting their telephone reservation systems. The resulting congested environment can be modeled using the kind of queueing approach presented in this paper.

References

JEWETT, J. E. 1977. Network Design for Queueing Situ-
Modeling the IRS Telephone Taxpayer Information System


