SIMULATION OF A BUSINESS COMMUNICATION SYSTEM SWITCHING NETWORK

James Yan

ABSTRACT

An important objective of teletraffic engineering is to evaluate new telecommunication systems relative to a set of grades of service. To realize this mandate, teletraffic engineers have relied on a variety of analytical and computer-based techniques, of which discrete-event simulation has played a significant role. This paper describes the top-down development of a simulation for evaluating the switching network of a recently developed business communication system, Northern Telecom's SL-1 @ . Simulation is adopted for its total view of the system whereas applicable analytical methods neglect significant interactions. The discussion focuses on the modeling, the design, the implementation and testing of the simulation.

INTRODUCTION

The application of digital technology to switching systems has given rise to a new generation of business communication systems (BCS). These modern BCS, by using stored program control, provide very extensive calling features to enhance the basic telephone service the customer receives. However, in some calling features, multiple speech paths are required in each feature use. Depending on the feature invoked, the paths have to be established either sequentially or simultaneously. The complex connection sequences of some features add another dimension to the traffic analysis of the system's switching network. A challenge teletraffic engineers face is to develop a tool that incorporates the impact of the features in the evaluation of the switching network performance.

Compared to approximate analytical techniques, simulation is a more promising approach for the following reasons:

@ Trademark of Northern Telecom Limited

1) Simulation considers the correlation among the several paths required in the same calling feature. Tractable analytical models ignore the correlation.

2) The interaction among the various call types and calling features is taken into account.

3) Simulation can assess the effects of the different orders of establishing the multiple speech paths.

4) Algorithms for searching network paths can be evaluated.

5) Simulation allows the investigation of a mixed environment with both loss and delay service.

This paper describes the development of a simulator for studying the effects of selected calling features on the switching network of a specific BCS, Northern Telecom's SL-1.

SYSTEM DESCRIPTION

THE SL-1 BUSINESS COMMUNICATION SYSTEM

The hardware architecture of the SL-1 is shown in the functional block diagram of Figure 1. The peripheral equipment consists of the terminals (telephone sets, trunks, attendant consoles, and tone digit receivers) and the circuits needed to transform the terminals' voice signals into 8-bit pulse code modulated (PCM) speech samples. The encoded samples are input into a network pack via a 32-channel time-multiplexed digital loop. Of the 32 channels available , 30 channels are used for voice transmission, one for signalling and one is unused.

The conference circuit is for conversations simultaneously involving more than two parties. The service circuit provides signalling tones and performs the outpulsing

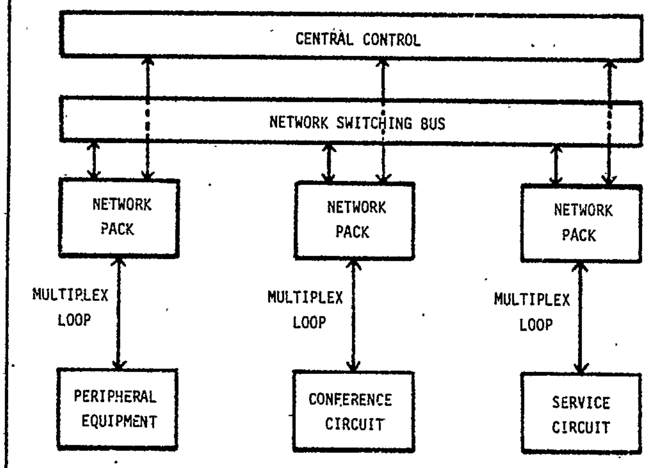

FIGURE 1

of dialing information to outside offices. Both the conference and the service circuits are each connected to a network pack by a 32-channel digital multiplex loop.

A basic SL-1 has one network group consisting of 16 multiplex loops. Twelve of these loops carry voice traffic, two are used for conferencing and the remaining two are connected to service circuits. The loops are interconnected by 16 network switching buses.

THE SWITCHING NETWORK

Within a group, the connection between two terminals, or between a terminal and the conference or service circuits, is established with the network switching bus and the space switches in the network pack. The connection is directly controlled by the network packs, which in turn are directed by the central control. To emphasize the switching function of the network packs, Figure 1 is redrawn and shown as Figure 2 in which the space switch is isolated and the space switch with the time slot interchanger form the switching network. The focus of our simulation study is on the switching network and its queues.

Because the SL-1 employs digital switching, the physical connection between any two loops is shared in time among the 30 channels or time slots dedicated to voice traffic. Instead of being held for the duration of the conversation, the speech path of a call is maintained only during the call's assigned time slots. For each of the 30 time slots, numbered 2 through 31, the space switch in the network pack sets up a new connection. An ordinary phone call requires two time slots, one for each party. The slots have to be an admissible matching pair in the sense indicated in Figure 3. In the figure each double arrow joins the two slots forming an admissible pair. Each party of a call is constrained to receive and transmit information during its time slot. The time slot interchanger (TSI) is used to transfer a call's information from one time slot to its matching slot. Thus the use of the time slot interchangers and the full access nonblocking space switch make it possible to have both intraloop and interloop calls. A connection in a call is blocked only if there is no admissible matching pair of time slots in the originating and the terminating loops.

NETWORK QUEUES

Beside establishing the connections of calls in the talking slots, the switching network also provides access to the system resources needed to set up a call. These resources are the tone circuits and the outpulsers in the service circuit loops, and the tone digit receivers and the attendants spread over the voice loops. Each of these resources is granted on a delay basis. Hence, for each resource or group of resources, there is a queue possibly with its own performance specification. Note that the delay in obtaining a resource may be due to two factors: either the resource is not available, or it is available but there is no network path to the resource's loop(s).

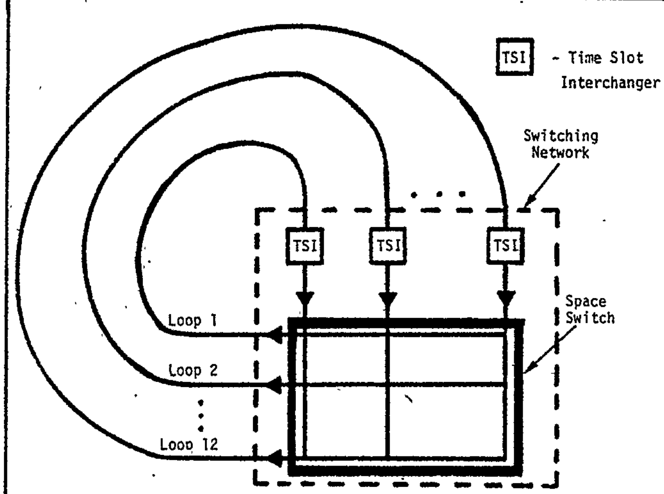

FIGURE 2

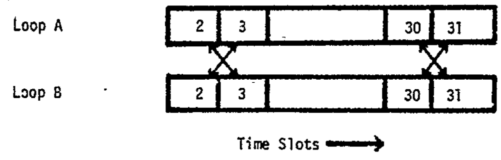

FIGURE 3

DESCRIPTION OF SELECTED SPECIAL FEATURES

In this section the special features included in the simulation program are described. These features are selected solely because of expected impact on the loop traffic capacity of the system.

The features considered may be grouped into two classes: (I) features involving more than two parties per call and (II) features that may be activated upon encountering a busy station. Comprising the first class are transfer calls, consultation calls and conference calls for three to six parties. The second class consists of Hunting, Call Waiting and Ring Again.

A transfer call is executed when a station user of an established two-party call holds the existing call, originates a call to a third party, and upon establishing a three-way connection, transfers the held party to the third party. In a consultation call, a station user, while on an established two-party call, holds the existing call and originates a call to a third party for private consultation. The station user may then return to the held call, make another consultation or add the third party to form a conference call. If the third party does not answer, the station user may hang up or go back to the original call.

Hunting is a system feature that routes a call to another station directory number (DN) of a pre-arranged group when the call directory number is busy.

In Call Waiting, a station user of an established call is informed of another call waiting to be connected. The station user may then place the existing call on hold and answer the waiting call, or he may transfer back and forth between the two calls.

The Ring Again feature permits a calling station user, upon encountering a busy directory number, to request the switching machine to monitor the called number. Once the number is idle, the calling station user is alerted. The station user may then retry the call.

SYSTEM MODEL

MODELING APPROACH

Modeling the usage of the SL-1 switching network involves the definition of six sets of entities: primary transactions, derived transactions, processes, events, facilities and queues. Each primary transaction has a unique process. Given a primary transaction, its associated process is defined by specifying the process' sequence of events and the possible derived transactions. Each derived transaction, in turn, has its own process. Each event indicates the facilities to be seized, the seizure time, the queues to use if queueing is permitted and needed, the logical operations to be executed, and the statistics to be collected.

A top-down approach is used to determine what should be included in the six required sets of modeling entities. All the possible primary transactions and derived transactions are first listed and those of interest are selected. The chosen primary and derived transactions are then analyzed in order to trace their processes. The outcome of the analysis is the identification of the required network facilities, the queues, and the events. The final, most detailed modeling step is to construct the event models.

The modeling procedure just outlined leads naturally to a system model that is sufficiently modular to accept changes with minimal impact on the existing model. The modularity exists in different levels ranging from the most detailed concept, the event, to the least detailed, the primary transaction. The implications of modularity on implementation and testing are discussed in later sections.

Another important consideration in building the system model is the need to strike a balance between model complexity and the human and computing effort required to realize the complexity. To achieve this balance, simplification is necessary. Some effective simplification techniques used are:

- ignoring resources used infrequently,
- simulating resource availability as a random variable,
- replacing random durations with their mean duration

TRANSACTIONS, PROCESSES, AND EVENTS

The primary transactions are the possible call types in SL-1. These calls are classified according to their origin-destination combination and, for calls originating in the SL-1, to the telephone set used. All the calls are modeled as arrivals from a common Poisson source. The originating and terminating loops of a call are randomly selected.

The derived transactions consist of the special features activated by a call. The derived transactions are linked to the primary transactions on a probabilistic basis. Furthermore not all the primary transactions can lead to a particular derived transactions. In Table 1, the primary transactions are listed in the left most column while the derived transactions are listed across the top row. An '*' in a cell indicates that the call type of the cell's row can activate the special feature represented by the cell's column.

SPECIAL FEATURE / CALL TYPE	CALL TRANSFER	CONSUL-TATION	CONFER-ENCE	HUNTING	CALL WAITING	RING AGAIN
Intraoffice, Rotary Dial	*	*	*	*	*	*
Intraoffice, DIGITONE @	*	*	*	*	*	*
Incoming, Attendant - handled	*	*	*	*	*	
Incoming, DID	*	*	*	*	*	
Outgoing, Rotary Dial		*	*			*
Outgoing, DIGITONE @		*	*			*

@ Trademark of Northern Telecom Limited

TABLE 1

The process of a primary transaction is obtained by tracing the time sequence of how a call uses the switching network. Each call type progresses through the following stages: off-hook connect, receiving dial tone, dialing, connecting, talking, and on-hook disconnect. Given a call type, the network resources seized or released in each stage are identified, and the resources' holding times are determined. The holding times are assumed to be exponentially distributed. Their means, however, differ and are input data the user has to supply. In the connecting stage, the called party is either idle or busy. If it is busy, the call can generate a derived transaction selected from the Class II special features. Once a call is in the talking stage, there is a chance that a Class I special feature is activated, producing another derived transaction.

Compared to the primary transactions, the derived transactions, being the system's special features, have less commonality in their processes. Hence, the operations of each special feature of interest have to be analyzed and modeled as a well defined sequence of events. An event can be viewed as either an arrival of a transaction (a new resource demand) or the release of a held resource. Hence, all call originations and activations of special features are events. As a call evolves, its transitions from one stage to the next is also usually an event.

FACILITIES AND QUEUES

The analysis of the processes of both the primary and the derived transactions leads to the facilities and queues listed in Table 2. The list is not exhaustive but contains those more likely to contribute significantly to network congestion. The time slots of the voice loops, service loops and the conference loops are used for establishing network connections. The tone-digit receivers are devices used to interpret the dialing information from the station sets with multifrequency signalling. The attendants' function in the model is solely to handle incoming calls not using the direct inward dialing (DID) feature. Both the tone-digit receivers and the attendants are assumed to be uniformly distributed among the voice loops.

The dial tone queue is for originating calls waiting to receive dial tone. Incoming calls completed by the attendants are held in the attendant queue if no attendant is idle. Since it is unlikely that a caller would wait indefinitely in either queue, the model drops a queued call if the call has waited beyond preset limits. These limits are based on human factor data in the telephony literature.

FACILITIES

1. Time slots of voice loops
2. Time slots of service loops
3. Time slots of conference loops
4. Tone-digit receivers
5. Attendants

QUEUES

1. Dial tone queue
2. Attendant queue

TABLE 2

OVERVIEW OF SYSTEM MODEL

The integrated system model is conceptually shown in Figure 4. The solid arrows trace how a transaction, primary or derived, can load a facility or a queue. The switching matrix is not a facility, but rather a conceptual block to represent the algorithm of routing a transaction to the facilities. The broken arrows indicate the information flow from the call generator and the voice loops to the feature selection generator. For a particular call, the call generator provides the call type while the voice loops give the status of the required network connections. From these two inputs and the model data, the feature selection generator determines which special feature to activate.

SIMULATOR DESIGN

ORGANIZATION OF THE SIMULATOR

The simulator has a two-level hierarchy as shown in the diagram of Figure 5. At the top is the control program. Its logic guides the time evolution of the simulation model. In the common second level are the event models, the resource monitors, and the queue processors. The event models simulate the operations and decisions required in each of the processes. An event model may be common to several processes. The resource monitor determines when a busy resource becomes available so that the resource's queue processor can take over. The rest of this section discusses some of the salient considerations in designing the simulator.

THE CONTROL STRATEGY

Since the SL-1 switching network can be viewed as an event-driven system, a control strategy based on the process interaction approach is adopted. In this strategy, if no call has been queued, the simulation model evolves by realizing the first event selected from an ordered list of scheduled events. The list is generated as the simulation proceeds. But if there are queued calls, the appropriate resource monitors are activated to insure that the queued calls are served at the earliest possible time before the next scheduled event. The details of the strategy are summarized in the flowchart of Figure 6.

MODULARITY IN THE EVENT MODELS

The event models are the building blocks of the simulator. Since these models are shared by the various processes, it is important to make the models as modular as possible. A high degree of modularity can yield the following advantages:

- reduction in the complexity of event modeling,
- basis for modular coding,
- ease of testing,
- flexibility for changes and additions to the simulator

A good degree of modularity is realized by the careful definition of the events' boundary. This is done by a comparative examination of the models of the possible processes. Because the events are linked in the processes, it is essential to have well specified interfaces among the defined events. Modularity is also improved by isolating operations common to some events. For example, scheduling the next event and matching two voice loops are common activities that are modeled independently of the events in which they are used.

RESOURCE MONITORS

A resource monitor is a set of logically linked operations to determine if a resource is available to a queued transaction waiting for the resource. Because a resource monitor can pose a significant overhead in a simulation run, an important parameter that has to be resolved is the frequency of checking the resource availability after the monitor has been activated.

The upper bound on the frequency is the frequency that the real system adopts. However, in a trade-off between cost and accuracy, the sampling frequency implemented can be less without sacrificing too much accuracy. To illustrate this point, consider the dial tone queue. Whenever a call is queued for dial tone, central control would try to provide dial tone several times per second. But the time slots of both the network and the service loops have average holding times in the order of seconds. Hence, in the simulator, it should be sufficient to attempt to match time slots for dial tone once per simulated second.

STATISTICS

Statistics collection is another essential aspect of simulator design. Although Figure 5 shows that updating statistics is separate from simulating an event, the actual collection of statistics is distributed in the sense that each event model includes updating some statistics. The statistical data accumulated in the simulator include:

- Number of call originations,
- Number of calls reaching a busy station or trunk,
- Number of attempts to use each special feature,
- Number of calls according to call type,
- Number of blocked calls according to call type,
- Utilization of the time slots according to loop type,
- Mean waiting time in queues

IMPLEMENTATION

TOP-DOWN IMPLEMENTATION

The modular character of the simulation model leads naturally to a top-down approach in the development, design and testing of the computer program implementing the model. The computer program was developed in three stages. In the first stage, only POTS (Plain Old Telephone Service) calls and special features were simulated. In the second stage, POTS calls and queues were considered. Finally the

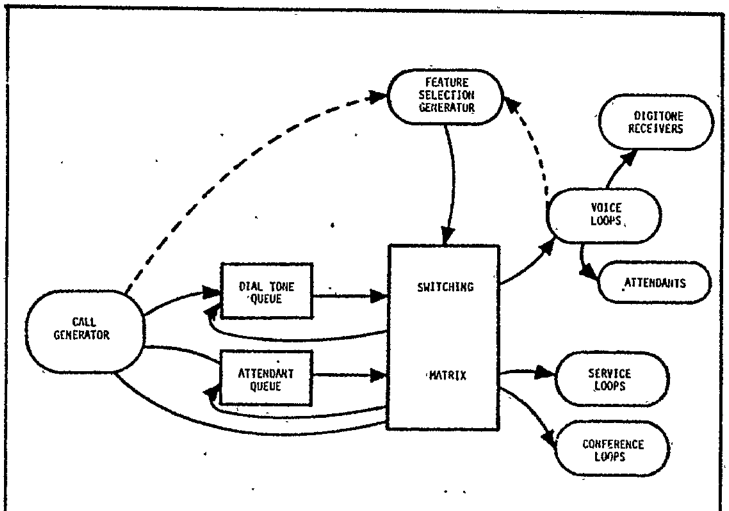

FIGURE 4

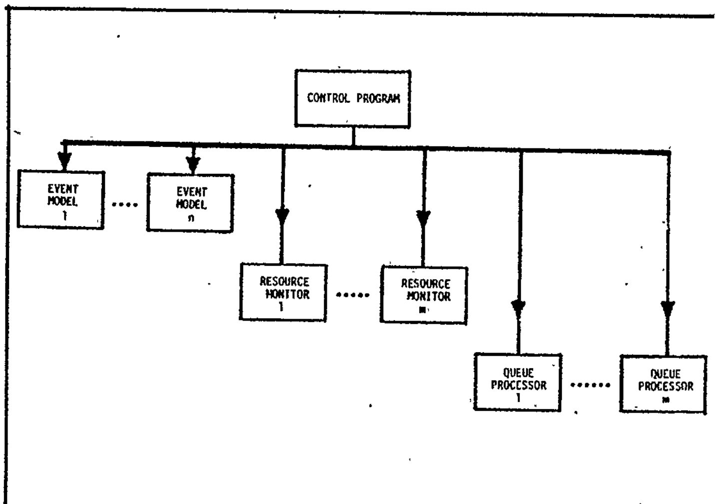

FIGURE 5

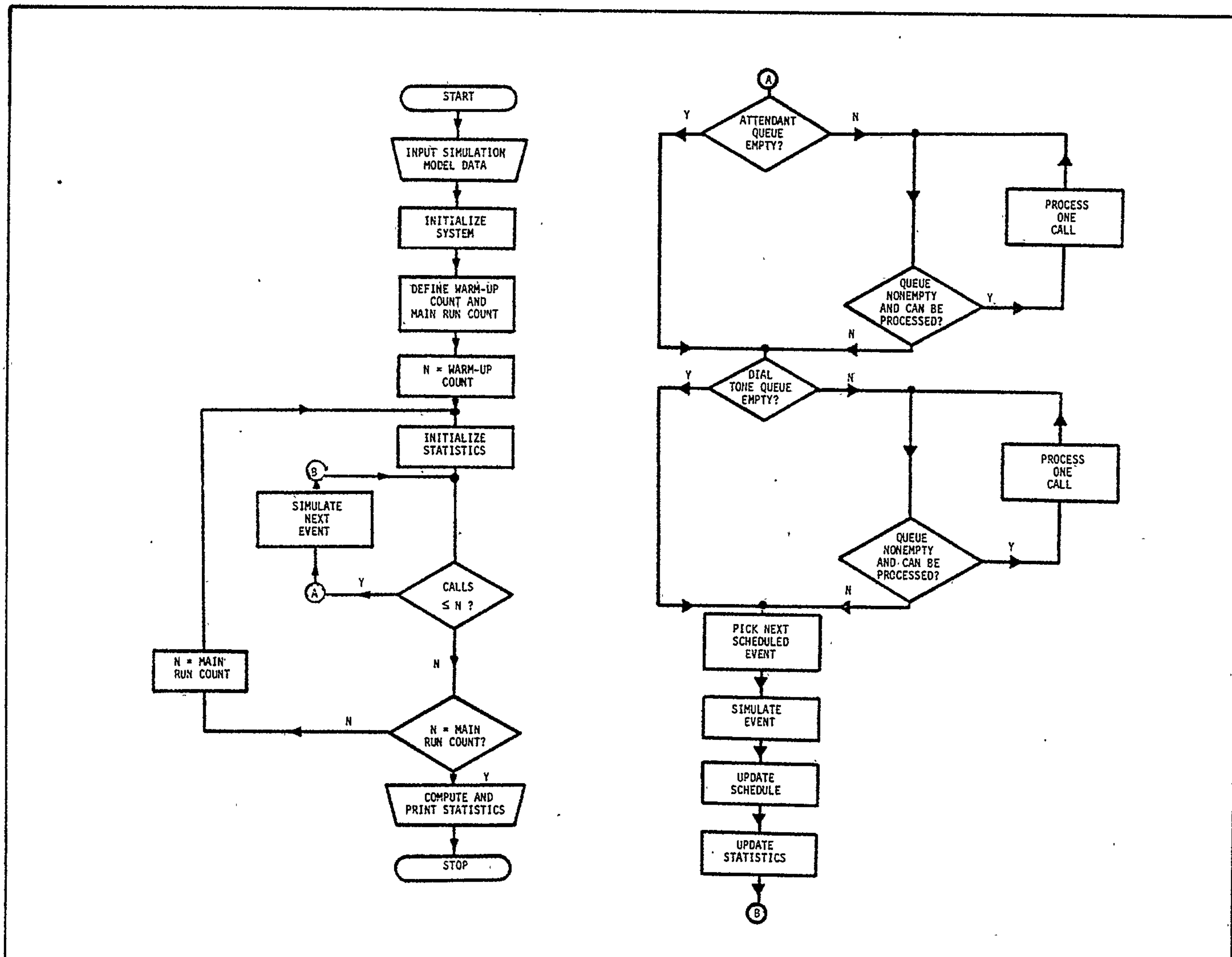

FIGURE 6

POTS calls, queues and special features were merged. In each stage, the program was organized into modules, each corresponding to an event model, a resource monitor, or a queue processor required in that stage of development.

The computer program was fully tested in each stage prior to the subsequent changes. The testing procedure adopted was to first drive the system by generating only one type of call, or allowing only one special feature, or using only one resource type. Then a mix of call types, features and resources which is representative of a typical SL-1 system was used. The advantage of such a procedure is to isolate the program bugs due to modeling and/or programming prior to testing the interactions of the modules.

DATA STRUCTURE MANIPULATION

Data structures are required by the simulation model to provide information on such objects as:

- the calls in progress,
- the scheduled events,
- the status of the facilities,
- the queues of calls waiting for a resource

The data structure manipulations implemented include:

- creating or destroying a record of a call,
- ordering the elements of a list according to an attribute,
- adding an item to a list,
- deleting an item from a list,
- determining the list length

Some of the manipulations also offer an opportunity to improve the efficiency of the program. An example is the data structure on the status of the network loops' time slots. The status is defined as the time slots' last finish time, not as being busy or idle. With such a definition, there is no need to introduce the event of changing the status of busy time slots when they become idle. The operation of matching time slots can be executed simply by comparing the last finish times against the current simulated time. Since idling network connections is a very frequent event, the saving in computer time is considerable.

INPUT-OUTPUT CAPABILITY

The simulator executes in either an interactive or batch environment. Through a list of format-free input data, a user can conveniently specify the particular SL-1 system he wants to simulate. To facilitate the procedure of entering the simulation data, most parameters have been set to default values. Once one simulation run has finished, the user has the option of terminating the simulation or restarting another run using different input information. The user can also specify whether the statistics should be printed at regular user-defined intervals in the course of the measurement run or simply at the end of the run. However, the statistics are always printed at the end of the warm-up period.

CONCLUSION

This paper has traced the development of the simulator of the SL-1 switching network. Basic discrete-event simulation concepts are used to formulate the system model in a top-down fashion. The modularity in the model is then exploited in the later phases of simulator design, implementation and testing.

The applications of the simulator have been numerous and include:

- verification of approximate analytical models,
- capacity determination,
- evaluation of operational procedures,
- testing of provisioning procedures,
- evaluation of new special features.