ITC 11, Minoru Akiyama (Editor)
Elsevier Science Publishers B. V. (North-Holland)

A Universal Environment Simulator for SPC Switching System Testing

Wolfram LEMPPENAU
Phuoc TRAN-GIA

Institute of Communications Switching and Data Technics
University of Stuttgart, Fed. Rep. of Germany

ABSTRACT

Besides of queueing system analysis and system simulations, the environment simulation provides a realistic test technique for communication systems. In this paper the concept of a universal environment simulator is presented, whereby realistic customer behaviour as well as subscriber-system interaction are considered. Implementation and performance aspects are discussed. The simulator is implemented by means of a multiprocessor structure operating in a function sharing mode according to a distributed control strategy. The Universal Environment Simulator UNES provides a tool to investigate switching system performance, e.g. call handling capacity under designed load as well as under overload , or the effectivity of overload control strategies. The performance of the environment simulator itself is investigated using a queueing network model, where system characteristics like message transfer or message circulation delay caused by the simulator are discussed.

1. A UNIVERSAL ENVIRONMENT SIMULATOR CONCEPT

During the past decade, a number of digital, stored program controlled (SPC) switching systems have been developed and introduced, which replace the electromechanical system generation. This tendency can also be recognized in current developments of private automatic branch exchanges (PABX's). In accordance with advances in hardware and software methodologies, the architectural complexity of this system generation increases rapidly. As a consequence, more powerful performance investigation methods are required in order to support the system design and to ensure a proper system performance.

Fig.1 illustrates a systematical overview of various established performance evaluation methods, where the importance of system simulations can be clearly recognized. Opposite to the two model oriented and queueing system oriented simulation levels with different degrees of abstraction and complexity, the environment simulation provides the most realistic test technique for telecommunication systems.

In the literature a number of environment simulators have been presented. Most of them are designed for specific systems to be tested [5-10]. Thus, they are system-dependent and can only be applied to the dedicated systems. Other approaches [1-4] deal with more system-independent concepts; they are designed for use in simulations of a relatively small number of subscribers. Most of known environment simulators do not take into account the dependency of the subscriber behaviour on system reactions (feedback effects, e.g. repeated attempts, subscriber impatience, etc.) as well as subscriber-system interactions.

In this paper the concept of the universal environment simulator UNES will be presented. The simulator is designed for telephone switching systems for up to one thousand connected subscribers. The interconnection and the communication to an arbitrary switching system is realized by an interface which is independent of the target system. The random subscriber behaviour is described in terms of arbitrarily chosen distribution functions and includes also real-time system reactions. Therefore, realistic system loads, which model stationary as well as nonstationary overload conditions, can be generated and offered to the system for test purposes. In order to characterize and to simulate overload traffic streams, e.g. for performance investigations of overload control mechanisms, the offered traffic can be realized by means of short-term, nonstationary load patterns (c.f. [12]), whereby realistic effects like repeated attempt phenomena can be taken into account.

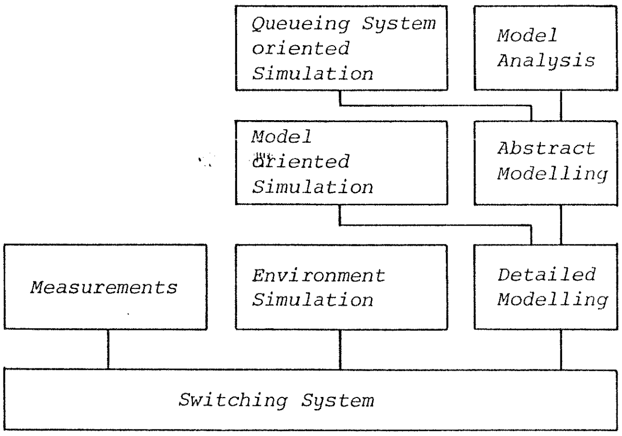

Fig. 1 *Performance Investigation Methods for Switching Systems*

The main features of the developed universal environment simulator will be briefly outlined. The subscriber behaviour is modelled and specified in the form of a SDL diagram (SDL: CCITT Functional Specification and Description Language) and is embedded in a multiprocessor structure environment. Thus, the subscriber behaviour model is programmable depending on the desired application. The interface to the target system, i.e. the switching system to be tested, is designed in a system-independent manner, represented by a set of telephonic events in conjunction with a messaging system.

In order to generate the random subscriber reactions, e.g. to simulate effects like dialling before dial tone conditions, incompleted dialling, call abandonments, subscriber impatience, etc., a large number of programmable distribution function types by means of a hardware random number generator are provided.

2. SYSTEM DESCRIPTION

2.1 System Overview

The functional structure and the software hierarchy of the environment simulator UNES are illustrated in Figs. 2 and 3, respectively. The simulator consists of three functional modules :

- System Control Module (SCM)
- Subscriber Behaviour Module (SBM)
- Target System Interface (TSI).

The simulator modules are implemented by means of microprocessor-based control units. Intermodule communication is done by message interchanging via the system bus.

2.1.1 System Control Module (SCM)

The System Control Module supervises the whole activities of the environment simulator. It will be distinguished between the configuration phase and the simulation phase. During the configuration phase, the environment simulator can be accessed and programmed by the user on the SCM. The configuration data to be stored can be either entered interactively through the man machine interface or loaded from mass storage or host computers. These data are subdivided into program parts for the modules TSI and SBM and tables, which are dedicated for use in the Random Number Generator (RNG), Message Transformer (MTR) and Subscriber Finite State Machines (FSM). During the simulation phase, the Simulation Control (SIC) monitors the system activities, starts and stops simulation runs by activating/deactivating the SBM and accepts messages from the SBM for measurement and statistic purposes. Messages from the SBM can be stored in a trace buffer or saved simultaneously in mass storage and host computer.

2.1.2 Subscriber Behaviour Module (SBM)

The main function of the Subscriber Behaviour Module is to model the designed number of subscribers and trunks connected to the system. It generates telephonic events (e.g., off-hook, on-hook, digits, etc.) for the simulated subscriber groups according to subscriber behaviour models and the reactions of the system to be tested.

The module controls also the signalling activities to the TSI during the simulation phase. Messages generated by a simulated subscriber are transmitted to the TSI and messages from the Target System (TAS), received via the TSI, are directed to the addressed subscriber,

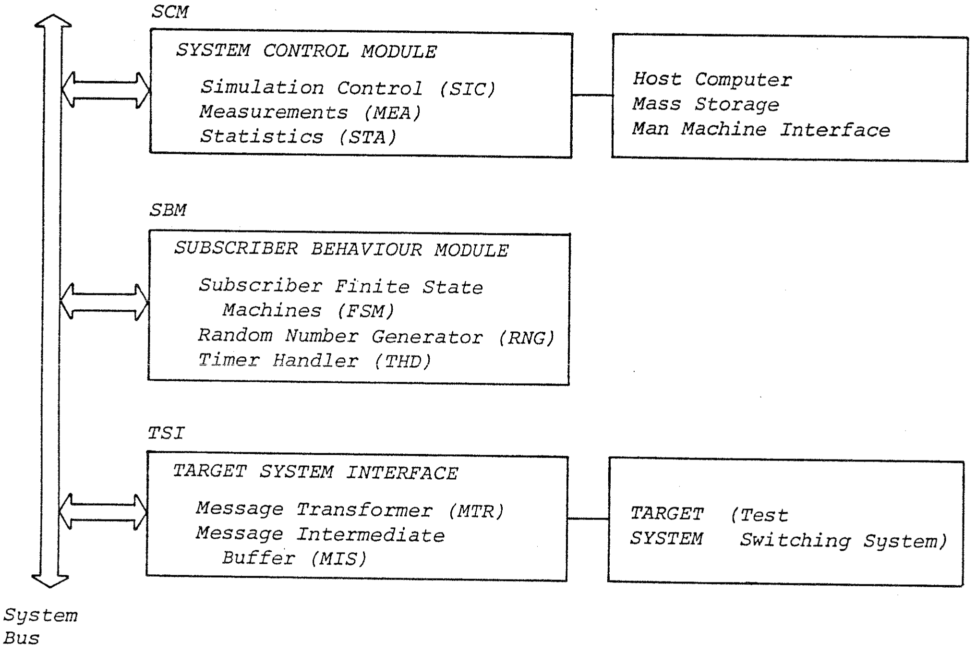

Fig. 2 *Functional Structure of the Universal Environment Simulator UNES*

e.g. to the appropriate FSM. The FSM accepts the message and stops the current subscriber-individual software timer by sending out a stop-timer message to the Timer Handler (THD). In order to model a subscriber in conjunction with his individual behavioural timing properties (impatience interval, wait for reattempt time, interdigit intervals, etc.), a software-timer is allocated to each simulated subscriber process, controlled by the THD. The subscriber process, which defines the behaviour of the subscriber in detail, is described by means of finite state machines in the form of SDL diagrams containing random branching probabilities and subscriber-oriented timer length distribution functions.

The Random Number Generator (RNG) is implemented by means of a multiplicative congruential random number generator providing uniformly distributed random variates in the interval (0,1] (c.f. [13]) in conjunction with a table-driven distribution function transformer. The transformer, which contains the desired and programmable distribution function types, inverts the uniformly distributed random numbers into required random numbers according to predefined distribution functions. Based on random numbers obtained from the random number generator, the FSM determines the action to be done. This action can be either an immediate or a delayed reaction from the subscriber to the test switching system. In the case of an immediate reaction, the corresponding message to the TSI will be sent. Otherwise, the delay is realized by starting an appropriate software-timer according to the corresponding random variable with a given distribution function. At the timeout epoch, the THD will inform the FSM by messaging. Based on the current state and on the obtained random numbers, the FSM sends a message to the TAS and plans the next subscriber action by starting the software timer. These facts will lead to decomposition and branching of messages, which will be modelled in detail in section 3.

Inside of the FSM, each subscriber or trunk is represented by an individual random-driven process. The actual state of a process is located in the individual data area of the simulated subscriber, where references to the specific data for a subscriber-type (e.g., behaviour-oriented time periods, probabilities for actions/reactions, facilities, etc.) are also stored.

2.1.3 Target System Interface (TSI)

The Target System Interface controls the communication with the Target System (TAS) connected to the environment simulator. Functionally, it consists of a Message Transformer (MTR), which transforms the message alphabets (e.g., coding and numbering of subscribers, digits, etc.) used internally in the simulator and in the test system. On the other hand, the TSI has to perform the flow control function. On the physical level the transmission of messages on the bidirectional data link is protected by parity- and timeout-mechanisms. In the case of a transmission error, the simulation is stopped and the actual state of the entire configuration is frozen. In order to enable more accurate measurements of message delay and system reaction times, the TSI is allowed to report the sending and receiving instants of messages to the system control module through the subscriber behaviour module.

2.2 Software Hierarchy

Based on the functional structure described above, the software structure of the environment simulator is built up in three levels (Fig.3). Level 1, represented by the SCM, includes the software for the overall system control and statistic evaluations. By entering the configuration phase, level 1 takes control over the whole system and initiates directly the data and program areas of levels 2 and 3, which stand for

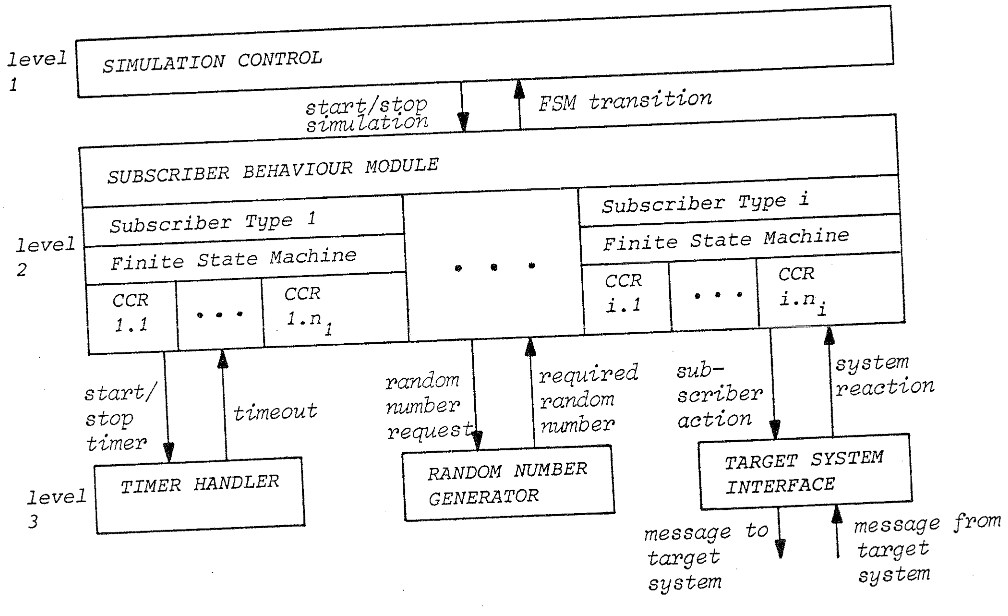

Fig. 3 Software Structure of the Universal Environment Simulator UNES (CCR : Call Control Record)

the SBM and TSI, respectively. During the configuration phase, the operating control units of SBM and TSI are deactivated. This enables level 1 to address the local program/data storage of levels 2 and 3 directly as data area.

During the simulation phase, there is no direct access to the local storage areas of levels 1, 2 and 3. By starting a simulation run, levels 2 and 3 will be activated and the interlevel communication has to be executed via message interchanging. Level 1 deactivates its overall system control, activates its statistic evaluation task and transfers the control over the system and messaging processes to level 2. Additionally, level 2 supervises the individual subscriber processes realized as Finite State Machines (FSM) in accordance with their appropriate data and program areas. The simulated subscribers are divided into groups of subscriber type (numbered from 1 to i, c.f. Fig. 3); each subscriber type represents a particular behavioural environment (telephone lines, trunk groups, subscribers with extended facilities, etc.) with their different attributes. The actual state of a subscriber and his attributes are stored in a Call Control Record (CCR). The states, their transitions and their attributes are linked together to form a universal software interface up to level 1, which is provided to simulator users.

Thus, a desired subscriber type behaviour can be programmed in terms of state transition diagrams, which consist of behaviour-dependent branching probabilities, random variables for timer periods in conjunction with programmable distribution functions. The simulation of the time-dependent and random-driven subscriber behaviour gets assistance of level 3. This level includes the Timer Handler (THD), the Random Number Generator (RNG) and the Target System Interface (TSI). For the definition of a subscriber reaction, the FSM requests and receives random numbers from the RNG. Actions initiated by subscriber processes (in terms of timeouts) are predefined by the FSM by starting the dedicated software timer. Since there is always a subscriber action planned for the future, the subscriber-individual software timers are always active. The determination of a timeout event is based on the simulation time supervised by the THD. The whole message flow between the three software levels during a simulation run is controlled by the FSM.

In order to estimate the performance of the described environment simulator, the functional modules and the messaging structure will be considered and mapped into a queueing model. The detailed model and its investigation are the subject of the next section.

3. PERFORMANCE OF THE ENVIRONMENT SIMULATOR

In order to estimate the performance and the maximum call throughput capacity of the environment simulator UNES, a detailed queueing model is developed, which will be investigated by means of computer simulations. In the model, both traffic levels, the call level and the message level (subcalls, telephonic events [11, 12]), are taken into account.

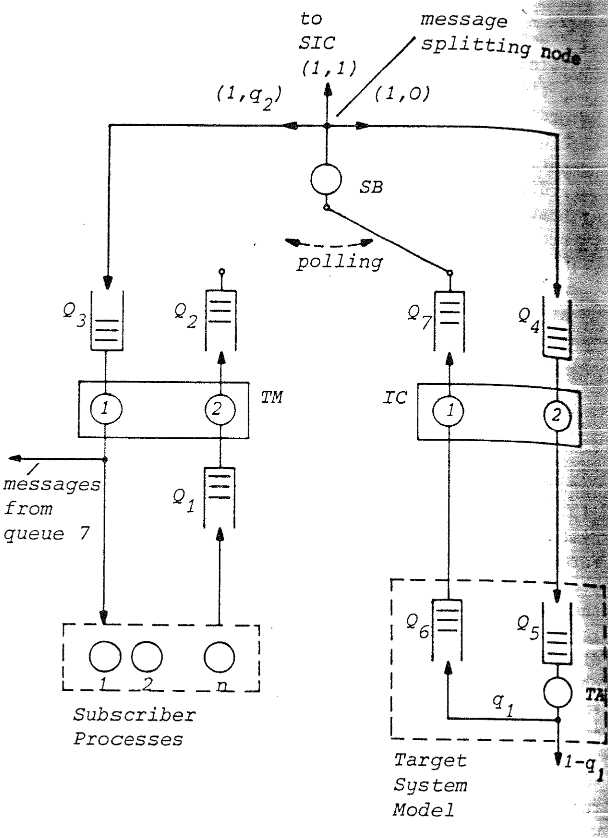

Fig. 4 Queueing Model for the Universal Environment Simulator UNES

3.1 Model Description

The detailed model of UNES is depicted in Fig. 4 in the form of a queueing network. In the following, the model components will be briefly described.

3.1.1 Server Stations

The server stations correspond to the functional units presented in section 2. The timer handler control unit is modelled by the server station TM (Timer Manager), which is connected to the multi-server group of subscriber processes. This server group consisting of n servers stands for the subscriber-individual software timers as described in section 2.1.2. The departure process of this server group forms the timeout message traffic, which initiates subscriber activities.

The server station SB represents the message handling activities corresponding to the state transitions of the subscriber behaviour in the FSM.

The server IC models the control unit of the TSI module, which supervises the transfer protocol of telephonic messages to the target system as well as the intermodule communication towards the SBM.

reaction time of the test system is proximately described by means of the server

1.2 Traffic Sources

ice the queueing system is a closed queueing twork, there exist no external traffic sources. wever, due to the splitting of messages after ing served in the subscriber behaviour module, essage generation in the sense of a branching rocess is taken into account.

t the message splitting node (c.f. Fig. 4), epending on the origination i (i=1,2 for queues ,7 respectively) and the destination j (j=1,2,3 for server stations TM, SIC and IC), a message coming from i will generate a group of messages of size g_{ij} routed to the destination j with the probability q_{ij} according to the group size atrix

$$\underline{G} = \{g_{ij}\} = \begin{pmatrix} 1 & 1 & 1 \\ 2 & 2 & 0 \end{pmatrix} \qquad (3.1)$$

and the message branching matrix

$$\underline{Q} = \{q_{ij}\} = \begin{pmatrix} 1 & 1 & 1 \\ q_2 & 1 & 0 \end{pmatrix}. \qquad (3.2)$$

Considering the message traffic generated by subscribers during the set-up phase, the conversation phase and the releasing phase of a call, the message traffic generated by a particular subscriber is modelled by means of an interrupted Poisson process (IPP, c.f. [14]); the "on"-phase of the process corresponds to the call set-up phase, the "off"-phase stands for the conversation and the idle phases of the considered subscriber. Thus, the intermessage intervals of a subscriber, which are approximated by the timer period lengths, is described in the model by the random variable T_{SUB}. In general, T_{SUB} is assumed to be distributed in accordance with the interarrival time distribution of a generalized interrupted Poisson process [14]. By t[illegible] choice of Markovian "on"- and "off"-phases f[illegible] the IPP process, T_{SUB} follows a 2nd-order hyperexponential distribution function.

3.1.3 Traffic Flows

Timeout messages, which represent subscriber-oriented telephonic events to be handled with, are offered to queue 1 and subsequently served by the timer manager. After service in the TM station, messages will wait in queue 2 for the next polling instant initiated by the subscriber behaviour control unit SB. Reaching the message splitting node, the observed message originating from queue 2 will branch into three messages, one for each direction towards the TM, SIC and IC servers, as described with the matrix notation above.

The subsequent message offered to queue 3 corresponds to another start-timer message, while the message representing the telephonic event to be transmitted to the target system will wait for transmission in queue 4. Since every activity inside level 2 has to be reported to the SIC, a subsequent message must be generated and sent to the module SCM.

A message received from the target system will be processed in the server station IC (Interface Control Unit) and subsequently waits to be polled in queue 7. Based on the current state of the addressed subscriber process, the branching procedure will take place after service in SB, according to eqns. (3.1) and (3.2).

Messages with origination queue 7 and destination server TM will leave the system after service in TM. Messages sent to the simulation control module will not further affect the model.

3.1.4 Server Scheduling

The server stations operate according to the following schedules :

i) Servez TM
While queue 1 is served exhaustively in a first-in, first-out order, a nonexhaustive service is implemented for queue 3.

ii) Server SB
A cyclic nonexhaustive service in conjunction with a polling mechanism is applied.

iii) Server IC
Nonexhaustive servive is implemented for incoming messages (queue 6) and exhaustive service is designed for the outgoing direction (queue 4).

3.1.5 Target System Modelling

The target system, i.e. the switching system to be tested is modelled by means of a single server station with the service time T_{TAR}, which can be thought of as the reaction time of the test system. After being processed in the switching system, a system reaction upon a message will be created and sent back to the simulator with the probability q_1.

The Target System Model can be changed to adequately model specific target system structures, e.g., using an infinite server model, or a more complex queueing network.

3.2 Results and Discussion

Simulation is provided to determine the delays of messages whereby the following delays are considered (c.f. Fig. 4)

T_{IN} Input delay :
Receiving and recognizing delay of input messages, i.e. system reactions, caused by the message processing structure in UNES. This delay is accounted from the receive instant at IC until the departure time in server SB.

T_{OUT} Output delay :
Sending delay for messages to be transmitted to the target system. This delay is accounted from the departure instant in the subscriber process until the completion instant of the transmission at server IC.

T_{CIR} Circulation delay : Time interval from the timeout occurance of a subscriber process until the next timer of the considered process is activated. This turn-around time is measured by observing a message from departure instant in the subscriber process server group (timeout), via the following model components: queue 1, server TM, queue 2, server SB, queue 3, server TM.

3.2.1 Simulation Parameters

In order to estimate the performance limitations of UNES, the parameters of the model shown in Fig. 4 will be chosen in the following for worst-case considerations. Thus, all service phases, which are likely to have coefficient of variations less than unity (i.e. rather have deterministic or hypoexponential distribution), are assumed to be of Markovian type with means of one millisecond. In the simulative investigations presented below, the probabilities are chosen as $q_1=0.5$ and $q_2=0.75$. Due to observations done in test systems, the target system reaction time is modelled to be negative exponentially distributed with mean 50msec.

The random variable T_{SUB} for server processes follows a hyperexponential distribution of 2nd order. In conjunction with the standard balancing equation for this type of distribution function, T_{SUB} is determined by the mean $E[T_{SUB}]$ and the coefficient of variation $c[T_{SUB}]$.

Simulation results will be depicted in the following with their 95% confidence intervals.

3.2.2 Results

In Figs. 5 and 6 the mean values of input, output and circulation delays are depicted as functions of the number n of simulated subscribers and the subscriber traffic intensity $\alpha_{SUB} = 1/E[T_{SUB}]$, respectively. In Fig. 5 it can be clearly seen that for low subscriber traffic intensities, say $\alpha_{SUB} < .2$/sec, no queueing delays have been registered. The expected values of delays, especially the circulation delay, increase rapidly at higher subscriber traffic range, dependent on the number of simulated subscribers. The rapid increase of the circulation delay is caused by the waiting time in queue 3 (c.f. Fig. 4) in conjunction with the scheduling of server TM and the branching of additionally superposed messages from the target system. It should be noted here that the characteristics shown in Fig.5 correspond to worst-case parameters; the subscriber traffic intensities to be simulated are normally given at $\alpha_{SUB} = .01$/sec ($E[T_{SUB}]$ = 100sec).

The dependency of average delays on the number of simulated subscribers is illustrated in Fig. 6, where the assumption of an extremely high subscriber traffic intensity is made ($E[T_{SUB}]$ = 4sec, i.e. one message per 4 seconds from each subscriber on average, including his conversation and idle times). It can be recognized in this diagram, that even with this assumption, the delay characteristics of the environment simulator are in a reasonable range, which has no strong influence on the simulation accuracy, for up to one thousand simulated subscribers.

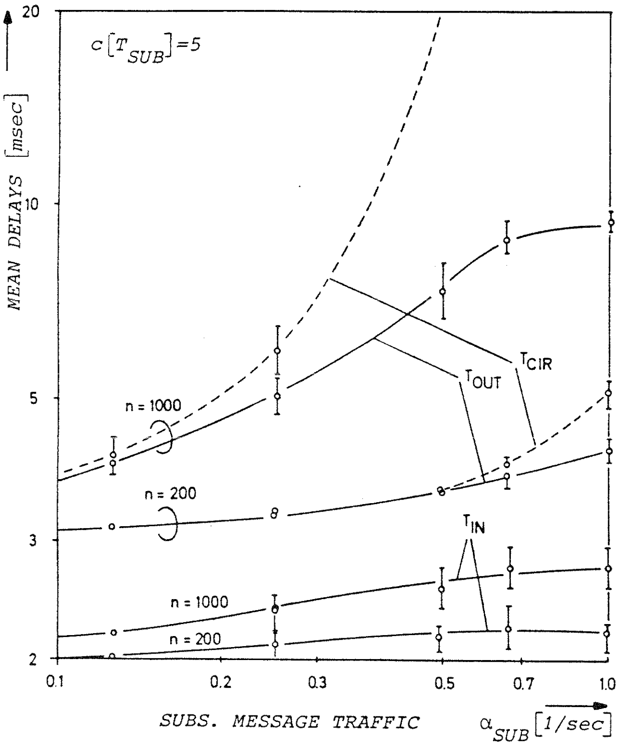

Fig. 5 Mean Delays vs Subscriber Message Traffic Intensity

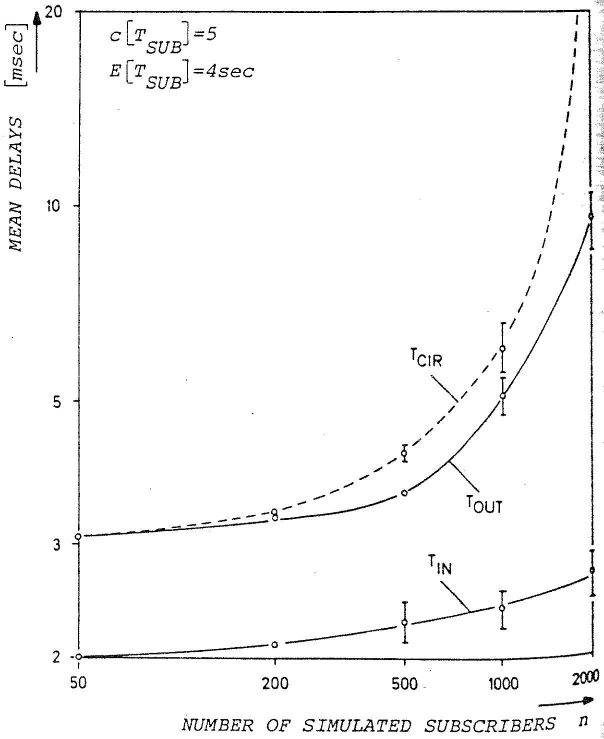

Fig. 6 Mean Delays vs Number of Simulated Subscribers

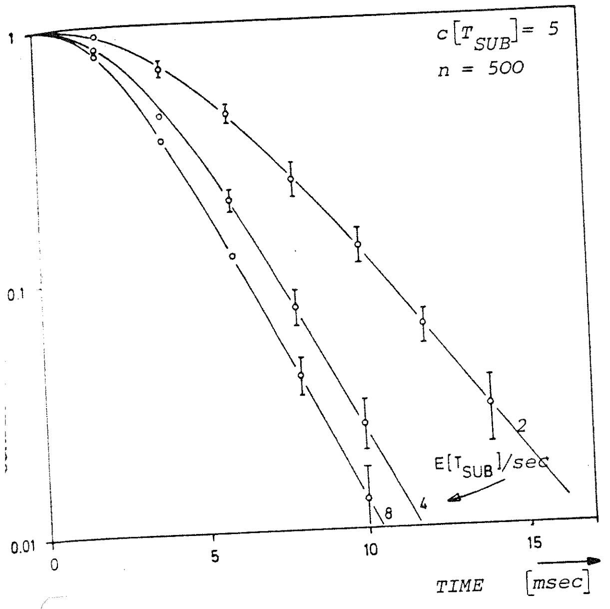

Fig. 7 *Complementary Circulation Delay Distribution Function*

Fig. 7 depicts the complementary distribution function of the message circulation time for different values of $E[T_{SUB}]$, where the influence of the subscriber traffic intensity on the delay characteristics is illustrated.

4. CONCLUSIONS AND OUTLOOK

In this paper, the concept as well as implementation and performance aspects of the universal environment simulator UNES have been presented, which is developed for telephone switching systems. The simulator provides a universal performance evaluation tool for switching systems, where realistic phenomena, which strongly affect the switching system performance, like subscriber impatience or repeated attempts can be considered and investigated.

The environment simulator allows us to investigate switching system performance under designed load and overload conditions. Furthermore, it can be applied to evaluate the effectivity of overload control strategies in switching systems, whereby detailed modelling of subscriber behaviour including considerations of the system feedback is arbitrarily programmable and adaptable. The performance of the simulator is investigated by means of a queueing network model. By means of the investigation, performance measures like the limiting load generation capability under a given message traffic characteristic and message delays are obtained. The concept can be extended to model subscribers operating with new services, which will be provided in current and future system developments, e.g., in ISDN-featured systems (ISDN: integrated services digital network).

ACKNOWLEDGEMENT

The authors would like to express their thanks to Prof.P.J. Kuehn for his interest in this project, to J. Saegebarth (University of Dortmund) for helpful discussions on random number generation, to R. Lehnert (TeKaDe - Philips Kommunikations Industrie, Nuremberg) for stimulating discussions on test requirements for switching systems and to T. Raith for helps with simulations.

REFERENCES

[1] Johner W., "LCS - ein Anrufsimulator für Ortsvermittlungsstellen", Elektrisches Nachrichtenwesen 55(1980)3, 217-220.

[2] Metzger R.M., Staber E., "Anrufsimulator UCS für die Prüfung von Fernleitungsausrüstungen", Elektrisches Nachrichtenwesen 55(1980)3, 210-216.

[3] Dael G., Amarger D., Marrois C., "AROMAT Telephone Call Monitoring and Recording Equipment", Comm. & Transm. 1983, pp. 39-52.

[4] Jansson B., Johansson K., Ostlund T., "LPB 110, A System for Controlled Test Traffic", Ericsson Review No.1 1984.

[5] Haugk G., Tsiang S.H., Zimmerman L., "System Testing of the No. 1 Electronic Switching System", Bell Syst.Tech.J. (1964)9, 2575-2593.

[6] Gruszecki M., Cornelis F., "Anwendung des Umweltsimulators zur Bestimmung der Leistungsfähigkeit von rechnergesteuerten Vermittlungssystemen", Elektrisches Nachrichtenwesen 51(1976)2, 119-123.

[7] Foucault C.C., Cerny D.J., Ponce de Leon L., "TCS-Umweltsimulation", Elektrisches Nachrichtenwesen 48(1973)4, 461-465.

[8] Del Coso Lambreabe M.A., "Umweltsimulation für Echtzeit-Software", Elektrisches Nachrichtenwesen 54(1979)2, 132-138.

[9] Grantges R.F., Sinowitz N.R., "NEASIM: A General-Purpose Computer Simulation Program for Load-Loss Analysis of Multistage Central Office Switching Networks", Bell Syst.Tech.J. (1964)5, 965-1005.

[10] Gruszecki M., "ENTRASIM: Real Time Traffic Environment Simulator for SPC Switching Systems", Proc. 8th ITC, Melbourne 1976.

[11] Dietrich G., Salade R., "Subcall-Type Controlled Simulation of SPC Switching Systems" Proc. 8th ITC, Melbourne 1976.

[12] Tran-Gia P., "Subcall-oriented Modelling of Overload Control in SPC Switching Systems", Proc. 10th ITC, Montreal 1983, pp.5.1.1.

[13] Ide H.D., Sagebarth J., "On Properties of Random Number Generators and their Influence on Traffic Simulation", Proc. 10th ITC, Montreal 1983, pp.2.4.6.

[14] Tran-Gia P., "A Renewal Approximation for the Generalized Switched Poisson Process", Proc. Int. Workshop on Applied Math. and Perf/Rel. Models of Comp./Comm. Systems, Pisa, Italy 1983 (North-Holland 1984) 167-179.