A COMMUNICATIONS SYSTEM SIMULATION FOR A NETWORK DISPLAYING CIRCUIT-SWITCHING, MESSAGE-SWITCHING, AND PACKET SERVICE

Paul P. Boehm
Jeffrey J. Guzy
Richard K. Paetzold

ABSTRACT

This paper describes a computer based simulation which models a network comprised of switching nodes capable of supporting circuit-switching, message-switching and packet-switching. The paper contains three major elements: description of the variable parameters which can be exercised to give insight into the simulated network performance, and some conclusions of the simulation.

The specific model is structured into three elements:
(1) Traffic Generator,
(2) Network Simulator, and
(3) Statistics Report.

The Simulation model, written in GPSS can represent an hierarchical network or, alternately, a non-hierarchical network. Each model is comprised of all nodes serving as circuit-switches, with a small number of arbitrarily selected nodes acting as message-switch facilities and/or packet-switch facilities. Traffic introduced is comprised of voice calls, message traffic, and packet traffic. Signalling and supervision, and routing plans are evaluated as part of the simulation analysis. Insight into network control information, throttling points, relative queue lengths, and call/message connect - delivery times are derived.

INTRODUCTION

This simulation is a tool developed to aid in the design or modification of a communications network. During the simulation design phase particular emphasis was placed upon flexibility so at later times modifications would require minimal programming and yet yield reliable data for further analysis. The areas of particular concern were protocols, network configuration, network structure, and routing plans.

The sizing of nodes and trunks, and node connectivity are of particular concern if a network is being designed prior to any actual implementation. This program provides a great degree of latitude to the user when defining the network.

The simulation provides two types of routing, Deterministic and DART (Deterministic and Adaptive Routing Technique). The Deterministic routing doctrine simulates the function of a deterministic routing table at each node; each node retains a table which contains a primary, secondary and tertiary trunk route to be utilized for each destination node in the network. Thus, for a given destination node, the primary trunk route is selected if available. When this trunk is not available, the secondary trunk is selected. Finally if that trunk is unavailable, the tertiary trunk is selected. When none of these trunk routes are available, the message must either be terminated or queued for later re-transmission.

The DART plan allows a fourth connection attempt from the last node the message could enter. If this fourth attempt fails, as a failure of the third attempt for Deterministic, the message must be terminated or queued. DART also allows message responsibility to be distributed in the network.

A special path selection algorithm called 'Calculated Path' is used in arriving at the route candidate.

INPUT INFORMATION

Input to the simulation program is composed of three parameters: the network, traffic statistics and system control mechanisms.

o Network

The information required to define the network includes node and trunk capacities, node connectivity, node function, and node type.

Node and Trunk Capacities

Node capacity relates a node's servicing capability to a quantity establishing the maximum number of simultaneous messages or calls that a node can handle. The maximum quantities to be defined include circuit-switch, message-switch, packet-switch function, and signalling and supervision messages.

Trunk capacity defines the number of simultaneous messages or calls which can be transmitted between two defined nodes into various channels. Between any two nodes three channel subsets (data, circuit-switch, and signalling and supervision) are provided. An additional requirement in defining channels is to specify transmission mode. This is

Comm System Simulation (continued)

used to distinguish the capability of transmitting in only one direction (simplex) from channels capable of transmitting in either direction (full-duplex).

Node Connectivity

Node connectivity is specified by inputting the complete list of nodes and nodes which are one trunk away. The total number of nodes must be specified. One degree of flexibility permits a disconnected network where no connectivity is provided between two or more sets of nodes.

Node Functions

As a basic premise of the program, all nodes have circuit-switch capability leaving message-switching and packet-switching service to the user's discretion. All nodes capable of servicing message-switch messages are labelled N/R (narrative/record) while those capable of packet-switching are labelled P. A node capable of all these services is labelled P/N/R. This information, unique for each node, must be supplied to the program.

Node Types

By specifying node types, a non-hierarchical or hierarchical network can be modeled.

Within a hierarchical structure, the nodes functioning as tributaries are at the bottom level, their function is to accept and deliver messages or calls to the individual subscribers. The next level is at a Regional node whose function is to route messages from tributaries over the backbone network or high-density trunks. The highest level of nodal types is the Gateway. Its task is to interface this network to other networks. Two other node types remain, Mobile Regionals and Mobile Tributaries respectively. The 'Mobile' tag signifies at some later time, a second program run, that these nodes may be connected to some other portion of the network.

Network Example

To illustrate the network concepts, a network, Figure 1, Sample Network is shown. Node type is indicated by the shape of the node, node function is indicated by the node label. The node connectivity, node functions and node types are specified arbitrarily.

o Traffic Statistics

The messages created by the Traffic Generator are controlled by user inputs which define message distributions, priorities, classification, destination, length, and interarrival time distributions. Messages are created in a probabilistic manner in compliance with the users' specification of the interarrival time and the type of distribution. A destination is chosen to meet the distribution defined by input. The message is then

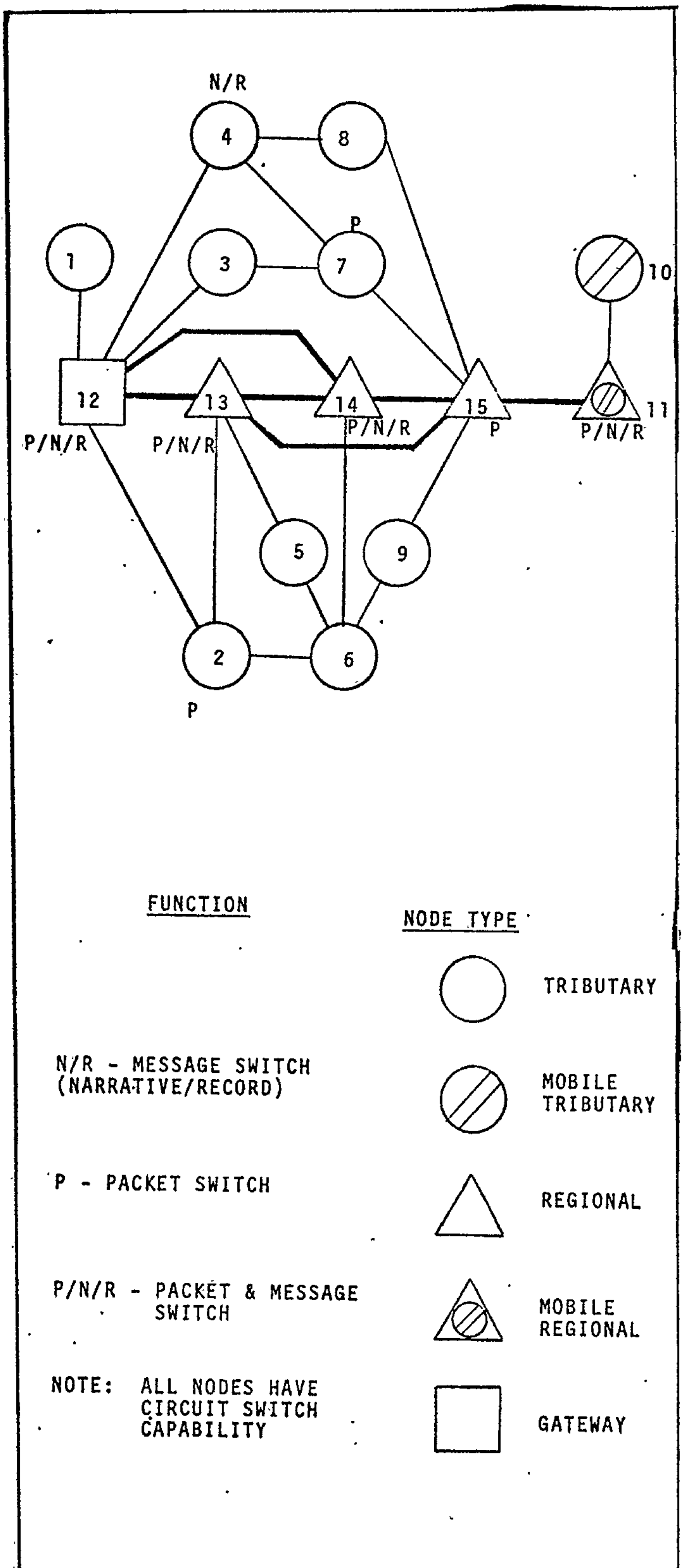

FIGURE 1 - SAMPLE NETWORK

Comm System Simulation (continued)

assigned a type, priority, and classification in accordance with the user specifications. Input characteristics include:

a. Message Types

    Narrative/Record
    Voice
    Packetized Data

b. Message Priorities

    5 levels provided

c. Message Classification Levels

    Specat
    Top Secret
    Secret
    Classified
    Unclassified

d. Destinations

    Local Traffic
    Adjacent
    To Mobile Nets
    Intra-Net Messages
    Inter-Net Messages

o System Control Mechanisms

In any network, blocking conditions are generated which tend to control message flow. Conditions initiated by an overloaded network are deterministic in nature and are discussed later. Several other types of blocking conditions are created probabilistically; security mismatch, subscriber busy, and negative message acknowledgment are all probabilistic in nature.

Subscriber busy conditions are related to the Originator Traffic Density expressed in erlangs. The probability of reaching a busy condition is twice the density level (P(Subscriber Busy) = 2 x .25 = .5, since the density is .25). This parameter can be controlled by changing the Originator Traffic Density as part of the input data. A busy condition causes a termination of a circuit switch message and a re-queueing of all other messages.

Negative message acknowledgments (NACK) can only occur for message-switch and packet-switch traffic when being transmitted from a storage node. After the data transmission, if a NACK is received at the transmitting node, it shall retransmit the information. This procedure can continue until three successive NACKs are received which shall result in a re-calculation of the transmission path.

Simulator Model Description

The Model consists of three program modules: traffic generator, network simulator, and statistics reporter. See Figure 2.

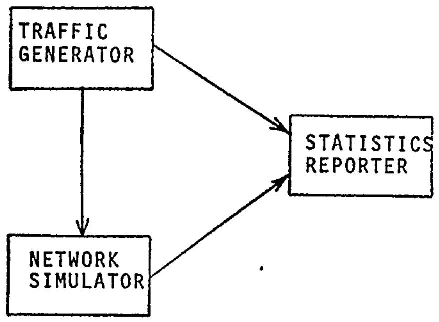

FIGURE 2 - MODEL PROGRAM STRUCTURE

o Traffic Generator

Traffic is initiated, labelled, and time-stamped with a network path in the Traffic Generator. The inputted traffic specifications are utilized here to obtain the traffic mix desired by the operator.

Once the traffic mix has been initiated, a path is calculated for each message (before network travel). The path from origination to destination is based upon four criteria:

1. message type,
2. network connectivity,
3. network hierarchy, and
4. node capabilities.

Message Type.

A path is calculated for a message dependent upon the way the message shall be delivered on the network. A circuit-switched message gets the shortest path with no storing of the message in its network travel. It is the shortest path possible because every node in the network has the capability to handle circuit-switched messages. A message-switched message must undergo store-and-forward service at responsible nodes in the network. The entire message is stored at any given time at one of its responsible nodes. These responsible nodes (a minimum of one) must be included in the path.

While a message-switch message cannot be subdivided, a packet-switch message could be considered a subdivision of a message-switch message and undergo store-and-forward service between the responsible nodes. Each packet of information is stored at a responsible node in the message's path; however, the packet can be stored at different locations in the network. These responsible nodes (a minimum of one) are also included in the packet-switch message path.

Comm System Simulation (continued)

Network Connectivity

The algorithm to calculate a path (either with responsible nodes or without) uses the nodal connectivity from the input information. As part of the traffic generator, a network directory matrix and connectivity matrix are established defining the inter-connectivity of nodes. These matrices impart the information about the network structure.

Network Hierarchy

Network structure has two methods of interconnectivity; hierarchical and non-hierarchical. A hierarchical network has a super structure of high-density (backbone) trunk groups between selected nodes. In relation to the average trunks and nodes, information can be transmitted more efficiently on the high-density trunk groups and nodes. The algorithm to calculate a path always attempts to utilize the high-density trunks.

A non-hierarchical network uses a homogeneous interconnection of nodes; one node type is the same as all others. Therefore, the shortest path is obtained in the path calculation.

Node Capabilities

The node capability is derived based on input information. When a path is calculated, the node capability is taken into account. If a responsible packet node is needed in a path, a responsible packet node is added to the path during the calculation.

The traffic has now been prepared for network travel. All messages are written to a magnetic tape, and stored for use by the network simulator.

o Network Simulator

The network simulator, by reading the messages stored on tape, uses the path on each message to route it through the network. Routing entails four phases of message delivery, Figure 3. Each phase is a specific network signalling and supervision message, which controls the network, allowing propagation of traffic to the destinations.

1. Connection request; requests a connection from the origination to the destination or its responsible node.
2. Lockin; acknowledgment from the destination to the origination that a connection has been set up (or from responsible node to origination). As the Lockin signal moves through the network, it actually acquires the trunks to be used.
3. Information; transmit the information from the origination to the destination on the trunks acquired (if to a responsible node, store at that node).
4. Disconnect; release the trunks, the message transfer is complete.

The switching of messages use the four phases as described above. The network simulator accounts for all the activity in the network and keeps track of the network status; node by node, trunk by trunk, and message by message.

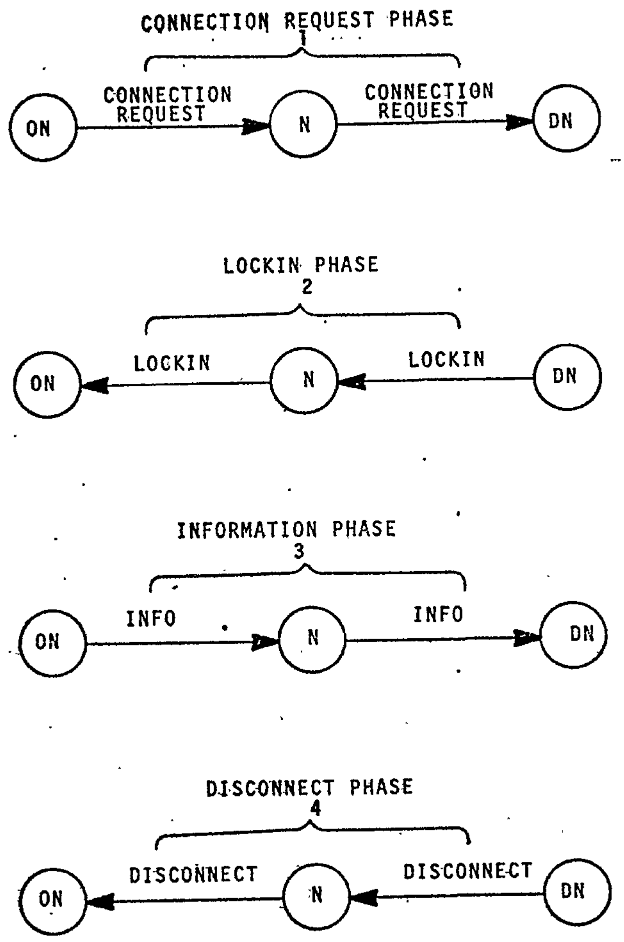

FIGURE 3 - FOUR PHASES OF MESSAGE DELIVERY

Once the network approaches blocking conditions, the network simulator can alter a message's path and re-route around a busy condition. The re-routing can be caused by one of the following:

1. node busy; re-route around the busy node
2. trunks busy; re-route around the full trunk group

3. pre-emption; a higher priority message seized an in-use channel, causing the pre-empted message to re-route.

4. subscriber busy; data can be re-queued at a responsible node till the subscriber is available.

Figure 4 shows how and when re-routing occurs. Each of the above re-routings is a network signalling and supervision message which instructs the affected node to take action.

The routing doctrine, defined as an input characteristic, determines the number of attempts of re-route, and the actual alternate path, if any. The blocked traffic is then delayed while an alternate path is obtained and re-queued for the next attempt at the message delivery.

All network action is driven by a logic state-table. Its construction is in modular form for ease of understanding and operation. Questions are asked at each node; a specific action is taken as a result.

o Statistic Reporter

The Statistics Reporter monitors major events in the simulation, from the creation of the traffic mix to the queueing at each node in the network. The statistics are basically categorized into four units:

1. Input traffic distributions
2. Output traffic transit time distributions
3. Blocking frequencies
4. Unique time distributions.

The distributions are printed out in tabular form with mean and standard deviation calculated for each table. These tables are the 'product' of the simulation and must be carefully analyzed in comparing alternate routing schemes, various network configurations, and changes in traffic mix.

Due to the large amount of output data available, a subset of output data is used in evaluating alternate routing schemes. This subset consists of:

1. Call-handling time (total time required by a call independent of information duration).
2. Connect-time (time expended in establishing an end-to-end connection).
3. Signalling and Supervision Queue times (queuing for signalling and supervision messages).
4. Message Connected Statistics (measure of network throughput).

Not only are distributions gathered for the above subset, but time-dependent graphs can be prepared from gathered statistics for this subset. Thus system status can be viewed at various times; from initial transient time, through stability, and into blocking conditions.

The alternate routing schemes can be visually compared when similar curves are prepared on a single graph. This technique is used in the network analysis as well.

Summary and Conclusion

The simulation consists of controlling traffic on a network with multiple signalling and supervision messages. Statistics gathered, as the simulation is running, are later examined to obtain a measure of performance. The purpose of the simulation is to observe and compare the performance of different protocols, network configurations, network structures, and routing plans.

All network data is presented to the program as input data. Thus, the data required to specify the number of nodes in the network, the type of function performed at each node, the nodal interconnectivity and capacity of the trunks are provided as program input parameters. Also provided as program input parameters is all the data required to specify packet-switching traffic, message-switching traffic and circuit-switching traffic. Thus, both the network and traffic are completely arbitrary (but completely described for a given run).

Output time delays and queue statistics are gathered and later utilized to provide the measure of performance.

The fundamental conclusion reached is that the simulation approach is a very viable method of obtaining system performance, commonly unattainable in any other manner. After performing the simulation, it is possible to select the input information which gives the best results within the scope of the simulation. The results of the simulation also provide a firm logical basis on which further improvements can be considered.

Several things were learned (or re-learned) which apply to this effort but which also apply quite generally.

An extremely important first step in any simulation is to properly scope the effort within the time, and money bounds. Simulation seems to suffer a common problem; most simulations are underestimated in design costs and necessary run times. It is important at this time to determine if, in fact, the required results can be obtained within an acceptable budget.

Having determined an acceptable (within budget) scope for the simulation, it is extremely important to provide a specification detailing the scope of the simulation itself and the scope of the results desired.

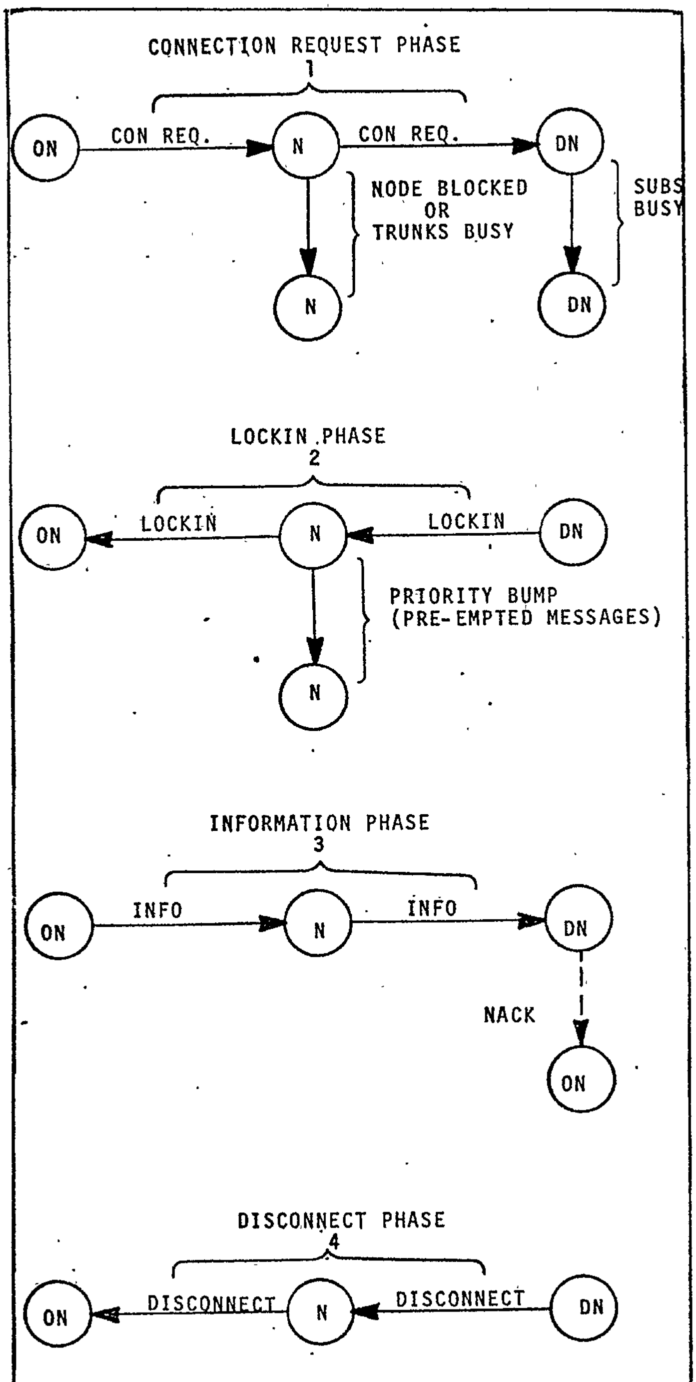

FIGURE 4 - MESSAGE DELIVERY PHASES AND RE-ROUTING CONNECTIONS

Selection of the language to be used in the simulation should be based on a trade-off of run-efficiency vs programmer efficiency. That is, a higher level simulation language (say GPSS) contains many instructions which are very powerful as opposed to some lower level languages. The powerful instructions permit a programmer to produce a more powerful simulation for a given (programmer) effort but at the eventual cost of more machine time. This trade-off must be considered when deriving the detailed scope specification.

Run costs must be considered in a more general way, as well. Most simulations develop in an evolutionary manner, i.e., examination of the results of one run suggest modifications to be tried in the next run. Thus, it is important to keep the per-run costs at a manageable level. Essentially, this means that the simulation must be as short as possible and still produce the required results.

Test and verification generally are difficult in simulations. Some simulations are run to verify an operation performed elsewhere. However, many simulations are performed because there is no other economical manner to obtain the results. The results of these simulations must be approached with caution since a universal approach to test and verification is not apparent.

A final item of consideration relates to the internal timing of the simulation. With a simulation of a network loaded with traffic, two critical periods become important when attempting to minimize run times. The first is in the minimum duration of time required to "load" a network, i.e., more generally, the transient time required for a simulation to achieve a steady-state performance level. The second is in the minimum time required for the network (simulation) to properly perform and to acquire statistically significant results. Thus, results can only be gathered after each period has elapsed.

Since several points of the above discussions might be discouraging, it is important to recognize that simulations are powerful tools and have been effectively utilized in many areas.

If you're considering simulation - do it!

This work was accomplished under AF Contract No. F30602-74-C-0189