REPRINTED FROM:

PERFORMANCE OF COMPUTER-COMMUNICATION SYSTEMS

Proceedings of the IFIP WG 7.3/TC 6 Second International Symposium on the Performance of Computer-Communication Systems
Zurich, Switzerland, 21-23 March, 1984

edited by

Harry RUDIN
and
Werner BUX

IBM Zurich Research Laboratory
Rüschlikon, Switzerland

PARTICIPANTS EDITION

1984

NORTH-HOLLAND
AMSTERDAM • NEW YORK • OXFORD

PERFORMANCE OF COMPUTER-COMMUNICATION SYSTEMS
H. Rudin and W. Bux (editors)
Elsevier Science Publishers B.V. (North-Holland)
© IFIP, 1984

THE NEW ROUTING ALGORITHM ON THE SITA DATA TRANSPORT NETWORK

Jean Michel Kaliszewski

S.I.T.A.
112 ave Charles de Gaulle
92522 Neuilly sur Seine
France

This paper is devoted to the topic of Routing Algorithms in packet switched networks. After a short review of papers and work in this area, we make a brief presentation of the SITA Data Transport Network. The decision to implement a Predictive Routing rather than an Adaptive Routing is discussed. The new Routing Algorithm is then presented as well as the algorithms and off-line computer programs, which are necessary for generating routing tables. A possible extension of the algorithms is finally presented.

1. INTRODUCTION

Routing Algorithms in Computer Communication Networks have been given much attention in recent years (see ref. /1/,/2/,/3/,/4/ for instance) and have been subject to theoretical investigations, experiments and effective realisations. A key issue of this topic is the adaptability of the Routing Algorithm to changing conditions (failure of network components and traffic flow variations, in order to optimize some criteria (minimize delay, or maximize throughput). Optimal adaptive routing algotithms have been proposed, see /5/,/6/; however, they raise implementation problems, such as, traffic flow measurement, or load-splitting (solutions to transient looping have been proposed, see /7/,/8/). Nevertheless, adaptive routing algorithms have been implemented. In some cases the routing algorithms only adapt to topological changes (e.g., DECNET see /3/, CERNET see/9/, DATAPAC see /10/). The ARPANET, Routing Algorithm called SPF, see /11/, provides adaptability to topological changes and to load variations also; it does not attempt to globally optimize the routing, but rather to optimize transmission delay of each flow individually.

Apart from other considerations, such as, complexity of development, stability under heavy load and reliability of adaptive routing algorithms, their optimality is a controversal matter (see ref. /12/,/13/) and could not be established, neither theoretically, nor practically.

Other routing algorithms, which might be called dynamic allocation, have been designed and implemented in several networks, such as, TYMNET, TRANSPAC and SNA (see /14/,/15/, /20/. This type of routing is associated to a specific data transport service : the session service (or switched virtual circuit service) : the route is computed at the time of session opening by the user. The calculation is based on user traffic type and user requirements, as well as the state and load of the network at connection time.

This route remains unchanged, as long as the session remains open. In case of link, of node failure, resulting in the unavailability of the route, the session is generally broken and a new route is allocated when the user re-opens his session.

Another class of routing algorithms, which we designate as Predictive Routing, has also been developed. In this type of routing a set of routes is off-line computed for every pair of nodes, based on predicted traffic flow values. These routes are then stored in tables in the switching nodes and in a given node, the proper route for every destination is chosen as a function of the network state. In case of change in network state (wether failure, or recovery) a new route is automatically selected. As this last process is purely deterministic and independant of actual network load, given the network state, one can predict which routes are used, hence, the name of this type of routing. The routing algorithm in use on the SITA Network has been subject to change, but has always been of the predictive type.

Although this type of routing has not been subject to many published studies, it gives rise to some interesting problems, which we wish to expose, taking the SITA Network as an example.

In the next section we briefly present SITA's Data Transport Network. In section 4 we analyze the design decisions concerning the SITA new routing algorithm. The routing algorithm is the subject of section 4 and associated computer tools are presented in section 5. Section 6 is devoted to a summary of the routing concepts introduced before, and in section 7 we make a brief comparison with other routing algorithms.

2. SITAs DATA TRANSPORT NETWORK (DTN)

SITA Network and Services have been presented several times, see /1/, /16/. Two different traffic types are transmitted over the SITA Network :

- Type A Traffic

  This is a conversational query-response traffic between CRTs and Airline Computer Systems (ACS). For this traffic, transmission delay is the most important objective.

- Type B Traffic

  Telegraphic one-way traffic, exchanged between TTYs (and eventually ACSs).

  For type B traffic, the most important objective is a very low loss probability.

Both traffic types are transmitted over the network with the packet-switching technique, but type B traffic undergoes additional processing for message switching and traffic protection.

Several years ago, SITA defined a new network architecture named SITA Advanced Network, the implementation of which began at the end of 1981.

The Advanced Network (AN) comprises of four specific systems, which are the following :

- The Data Transport Network (DTN
- The Network Control System (NCS)
- The Message Storage and Handling System (MSS)
- The User Interface System (UIS)

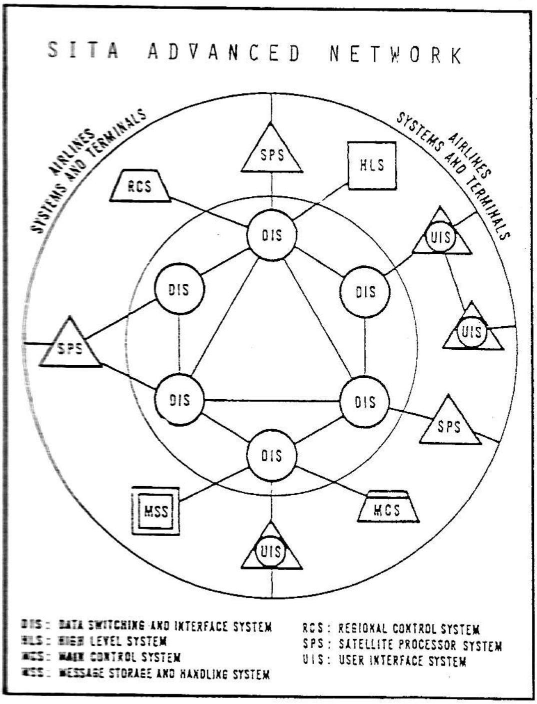

Figure I

Figure I shows a functional lay-out of this network and exhibits the following :

- An HLS, which is a system used as switching node in the former SITA High Level Network and relieved from type A and transit traffic processing in this new architecture.

- SPSs, which are the presently used access systems performing traffic concentration and protocol adaptation between user protocols and SITA internal protocols.

Specialized type A or type B concentrators (TACs and TBCs) are also used in the Network.

Airlines Computer Systems (ACS) may be connected either directly to the DTN, or through an Access System (SPS, UIS).

The different systems of the AN may be described as follows :

2.1. User Interface Systems

The UIS will replace the SPS as concentrators and Access Systems. They will perform the general functions of traffic and circuit concentration, as well as local switching and will offer new protocol interfaces, such as, BSC/SDLC, X25. They will also support service enhancements, such as, Multi-Host Access.

2.2. Message Storage and Handling Systems

This new family of processors is designed to store and process messages, data requiring high protection (e.g., type B), on mass storage and to support new telecommunications services. With the installation of DISs (see below) and MSSs, HLSs are progressively phased out (the target is a total of 4 message switching systems).

2.3. Network Control System

The NCS is aimed at providing a man-network interface environment for network operations and network management.

The NCS is a hierachical system composed of 4 Regional Control Centres (RCC) and a Main Control Centre (MCC).

2.4. Data Transport Network

The Data Transport Network is a packet-switching network offering a datagram like service for data transmission.

The DTN nodes are called DIS and perform all processing for type A and transit traffic and from an ISO Model point of view, the DTN will offer all transport functions (i.e., level 4) necessary for existing and new traffic types. HLSs are thus relieved from packet switching by the DISs and are devoted to message (i.e., type B) switching until replacement by MSSs.

DIS machines are Univac DCP40 mini computers. DTN links are either medium speed (9600 to 19200 bps) or wide band (48 Kbps). The link protocol is a superset of LAP-B. The present Routing Algorithm is an extension of the previous SITA HLN Routing. In order to avoid intricate problems in running two different routing algorithms in the network, a new routing procedure will not be introduced, as long as some HLSs are used as packet-switching nodes.

This routing works as follows :

Every system (DIS, UIS, MSS, RCS, HLS, SPS, TAC etc...) is given a two octet, topology-independent address, called High Level Designator (HLD).

This HLD is included in the routing header of every packet and is used for the routing which is based on destination addresses only.

In Every DIS a table (Access Handling Table AH) contains an entry for every HLD. This entry gives the outgoing-link number.

In case of topological change (link or node failure, or recovery) the A.H. Table is updated. The updating mechanism works as follows :

The local network of every DIS is divided into several destination groups for routing purposes, in order to split important traffic flows into smaller ones easier to reroute. For each destination group * of a DIS, in every other DIS, a table of Alternate Routes, called the Best Routes Table, is defined. This table contains a sequential list of route descriptions giving :

. the outgoing link of that route for that destination group;
. a set of link-up conditions associated to that route.

This table is scanned sequentially each time a change in network state occurs and the first route is selected, so that its "link-up conditions" are verified by the current network state; then entries in the A.H. Table are updated with this new information.

* Each system being identified by its HLD, a destination group is thus a list of HLDs.

Mainly because routing is based on destination information only, the Best Routes Tables are highly constrained.

A detailed analysis of these constraints may be found in /17/.

These tables are generated by off-line computer programs. New sets of tables are generated periodically to take into account traffic increase, connection of new systems and changes in topology (new nodes, new links).

3. CHOICE OF A NEW ROUTING ALGORITHM

In this section we want to address two questions :

i) Why should one choose a predictive rather than an adaptive routing algorithm? (Dynamic allocation algorithm, as briefly defined earlier, are related to session-oriented networks, rather than datagram networks. They present interesting similarities with the new routing algorithm, as we shall see in section 6).

ii) What improvements should the new routing algorithm bring to the existing one?

Let us first recall the technical requirements.

3.1. SITA Network Requirements

The main requirements concerning the DTN are :

i) Efficient use of network resources; this technical economical requirement implies a high average network load.

ii) High availability of routes between switching nodes, implying Alternate Routing in case of link or node failure.

iii) Average transmission delay of every flow must be below a specified threshold (0.5 sec).

iv) High reliability of network functioning (robust algorithms).

v) Ease of operation (World Wide Network).

vi) Data service with : - average loss probability (= 10-3)
- no sequence scrambling.

3.2. WHY NOT AN ADAPTIVE ALGORITHM ?

A general statement can be made about routing algorithms : from a performance point of view, designing a routing algorithm is always a matter of trade-off, as it implies using a fraction of network resources (line bandwidth and CPU time) to optimize the use of the remaining portion of network resources by user traffic.

In an attempt to provide better optimization, adaptive routing algorithms tend to use a higher portion of network resources for the routing algorithm itself.

However, recent adaptive algorithms, such as, ARPANETs SPF (ref. /11/), show good results in terms of CPU and bandwidth utilization, and we focus our analysis on the following points : performances, complexity, reliability and vulnerability.

3.2.1. Performance

We will not consider algorithms adapting to topological changes only, as they cannot have acceptable performance with heavily loaded networks, which is the case for the SITA Network.

Simulation studies (ref. /12/) have shown that unless the traffic is highly unpredictible and chaotic, adaptive routing does not show any significant superiority, in terms of transmission delays, over a static scheme.
Another study (ref. /13/) shows that adaptive routing provides a lower throughput than (static) scheme for a highly loaded network.

These results are particularly relevant to the SITA Network, for which we have the following :

i) High average load (usually from 30% to 70%).
ii) Good predictibility and regularity of traffic flow.

The second property is a consequence of the fact that the traffic flow is the summation of many individual flows.

These considerations do not favour an adaptive routing. In fact, there is little concern for us in getting the optimal transmission delay. What is really saught after, is keeping the transmission delay of every flow below a given threshold.

3.2.2. Complexity

In terms of software, undoubtedly, adaptive routing is more complex. The behavior of the mechanism in the adaptive case is also more complex and this makes it difficult to put limits on network size, which can be accommodated by such an algorithm. Trouble-shooting also appears more difficult in an adaptive-routing environment.

Conversely, network changes (new nodes, topological changes) appear to be easier with an adaptive routing, as they can be completely automatic (ref. /11/) instead of requiring generation and loading of a new set of tables in the predictive routing case.

3.2.3. Reliability and Vulnerability

Reliability of adaptive routing algorithms has improved (e.g., SPF as compared to the older routing algorithm in ARPANET). However, a potential risk of unstable routing exists and this risk increases with the load of the network (ref. /10/). To prevent this unstability it is necessary to have a positive bias in the measurement of delays and this leads to a decrease in optimality.

Adaptive algorithms, being distributed among the switching nodes, are more vulnerable to hardware or software errors of one of the nodes. (See ref. /19/ for a report on such a problem).

Predictive routing imbeds de facto a static security and is thus highly reliable. However, we should mention that errors may occur in the content of routing tables, which might create errors in the network (these errors will not affect the entire network, but only a part of it).

3.2.4. Conclusion

It appears from the above analysis that a predictive routing is better suited to SITA network requirements according to the reasons below :

- It is well adapted to traffic characteristics.

- Its on-line software is simpler.

- It is highly reliable.

3.3. The Reasons for a New Routing Algorithm

Present routing mechanism is based on the address (i.e., HLD) of the destination system. The main advantage of this solution is its simplicity. However, it has the following drawbacks :

i) Switching is based on the HLD, which is a two octet field in the packet header. This implies that access to the switching table (the A.H. Table) is done by Hash coding. Using a smaller address for routing would improve switching time and switching capacity of DISs.

ii) The partition of the local network of a destination DIS is used to split important traffic flows into smaller ones easier to route.

This method suffers from two defficiencies :

a) It often happens that several DISs have important traffic flows for the same destination DIS. In that case, it is difficult to find a partition of its local network so that all flows are satisfactorily split. Moreover, for those nodes which have small traffic flows for that destination, they must perform that splitting also, which results for them, in additional Best Routes Tables, i.e., additional memory requirements.

b) In some cases it would be preferable to split traffic flows according to the local network of the origin DIS to achieve the desired result.

iii) Mainly because routing is based on destination information only, strong constraints apply to Best Routes Tables (see ref. /17/ for more details). These constraints, which tie all the tables together, imply that the whole set of tables is generated simultaneously and, also introduce strong limitations in route optimization.

Moreover, as new traffic types are foreseen in the near future (e.g. file transfer) the need arose for a new routing algorithm, which would allow easier extension of DTN transmission services from the pure datagram service presently offered.

4. THE NEW ROUTING ALGORITHM

The main entities of the new routing algorithm are described hereafter :

4.1. Logical Routes (LRs)

An LR is a full-duplex logical connection between any two DISs, on which multiplexing of traffic flows between pairs of systems, (e.g., SPS-ACS) is performed.

Several LRs are defined between two DISs and every DIS may have a maximum of 512 LRs (see Figure II).

Every LR offers :
- 3 droppability levels,
- 3 priority levels,

which can be used simultaneously by user traffic.

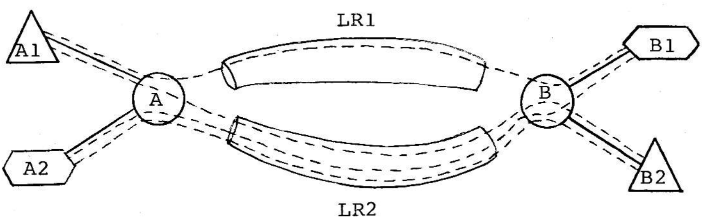

Figure II - 2 LRs between DISs A and B

An LR may be in normal or congested state and in the latter case end-to-end congestion control may be applied.

In every DIS each LR is given an identifier, which is a number between 0 and 511. Thus, every LR has two identifiers, one in each direction.

The mapping of traffic onto LRs is done normally on the following basis :

(HLDin, HLDout, [Traffic Class]) ⟶ LR number.

HLDin : HLD of origin system.
HLDout : HLD of destination system.
Traffic class : optional parameter.

Between two DISs, one LR (and only one) has an apparently different, but functionally equivalent mapping on it :

HLDout ⟶ LR number.

(The use of this LR will be seen in the next section).

LRs are established at configuration time and are permanent.

4.2. Routing Paths

- A Routing Path is an ordered sequence of links establishing a non-looping physical path between an origin node and a destination node.

- On every link belonging to a Routing Path, a Channel Number (CN), ranging from 0 to 255, is associated to that Routing Path. (See Figure III).

- Two routing paths having the same destination node and sharing the same set of links from a transit node up to that destination node, may have the same Channel Number on that set of links.

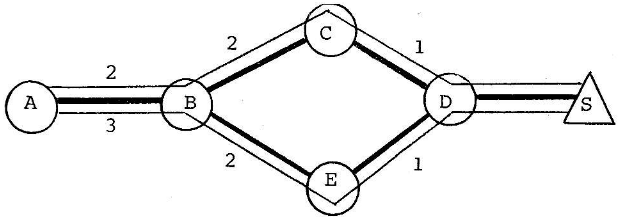

Figure III - Routing Path A-B-C-D has Channel Numbers, 2 on link A-B, 2 on link B-C and 1 on link C-D.

Routing Path B-C-D may share the same Channel Numbers as Routing Paths A-B-C-D, on links B-C and C-D.

- Channel number 1 is always used for the last link of a Routing Path.

  On every DTN link, channel numbers are assigned independently in each direction. Physical Routing on the DTN is not symmetrical. Two routing paths are associated to every LR, one in each direction. When a packet is out on a given Routing Path, its header contains the Channel Number of that Routing Path on every link it uses. When that packet reaches a DIS, the channel number is picked up from the header and used as an entry index to the Switching Table (ST) associated to the incoming link. This table gives the outgoing link number and channel number to be used on that link and included in the header. The packet is then output on that link.

4.3. Table of Alternate Paths (TAP)

- A Table of Alternate Paths contains an ordered sequence of Routing Paths, which can be used by a given LR.

- Two TAPs are associated to every LR, one at each extreme node : they are called peer TAPs.

- The same set of TAPs may be used by different LRs belonging to the same pair of DISs.

- Each entry in a TAP contains the following information :

  - First link and Channel Number on that link of the Routing Path.

- List of links belonging to the Routing Path.

- Each DIS maintains a Network Status Vector which has a binary position associated to every link of the DTN.

- In case of topological change in the network, each DIS, after having updated its Network Status Vector, scans the TAP associated to every LR originating from it and selects the first Routing Path, the links of which are all in up condition.

  If no Routing Path can be found, the LR is declared down, and this information is transmitted to the users on that LR. Packets sent on that LR are discarded by the Routing Algorithm (but may be stored by higher level functions). Once a new Routing Path has been selected, the DIS updates the Path Selection Table (PST) at entry corresponding to the LR number, with new outgoing link number and Channel Number on that link.

  The PST is used to switch input packets (i.e., coming from Local Network Nodes) on currently used Routing Path.

5. ROUTING CONFIGURATION GENERATION PROGRAMS

A Predictive Routing Algorithm is really composed of two-parts :

- The On-line part, i.e., software switching, which actually performs routing of traffic on the network.

- The Off-line part, i.e., Routing Configuration Generation Software, which prepares all tables needed by the first part.

The flow chart of Routing Configuration Generation Programs (Figure IV) shows the functional modules which we describe next.

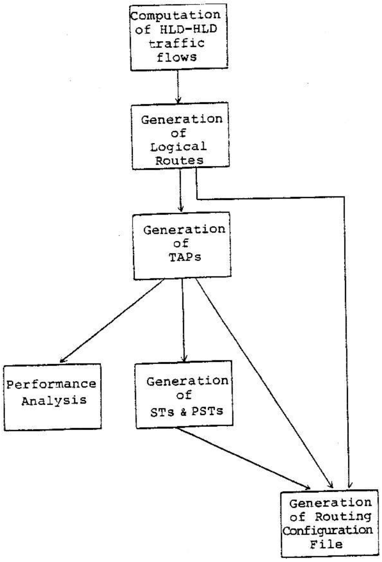

Figure IV - Routing Configuration Generation Program

5.1. Computation of HLD-HLD Traffic Flows

Forecast files give traffic information in terms of number of terminals connected to the network and monthly traffic figures (messages and characters) generated per terminal per geographical area and per company. In order to obtain traffic flows between access systems (i.e., HLD-HLD traffic flows), it is necessary to take into account the time difference between sources of traffic.

This is done as follows :

- A traffic intensity function is measured for every source access system (SPS, TAC etc...) for a 24 hour period.
- Every individual traffic flow is multiplied by the value of its corresponding traffic intensity function at the GMT time chosen for traffic flow computations.

On the SITA Network, three different peaks of traffic are observed : European, Asian and American.

Thus, for every HLD pair, traffic intensity is computed for each of the 3 periods. As these values are expressed in bps and in blocks/second, this makes a total of 12 traffic figures per HLD pair (6 in each direction).

5.2. Generation of Logical Routes

Total traffic flows between DISs often exceed the capacity of a medium-speed circuit and may represent more than 50% of a wide-band link capacity.

In order to have an easy rerouting of traffic in case of link failure, it is necessary to split DIS-DIS flows into several parts. This can be achieved by generating several LRs between the DISs. As generating extra LRs increases memory requirements, this process has to be done optimally.

Given a threshold tf on LR traffic, the algorithm works as follows :

i) For every HLD pair consider the maximum traffic figure for the three peak hours and the two directions and order HLD pairs according to these values.

ii) At some stage of the process, LR1, LR2, ... LRp, have been created between the two DISs and we are examining. HLD pair (i,j) :

   - Assign pair (i,j) to LR1 and compute its traffic for every period and direction,
   - if the threshold is exceeded, reassign pair (i,j) to LR2, then LR3 etc.,
   - if no LR can be found, create LR p+1 and assign (i,j) pair to it.

   Once all the HLD pairs have been mapped on an LR, it may happen that the last created LR has a very small traffic on it. In this case, that LR is eliminated and the HLD pairs previously mapped on it, are reassigned as evenly as possible on the other LRs.

As it happens, for many HLD pairs there is no traffic exchanged, It would be uneconomical (in terms of memory requirements) to map these pairs explicitly. Instead, an additional LR is created for every destination Local Network, named the default LR.

If no mapping exists for a given pair (HLD1, HLD2) it suffices to determine the Local Network of HLD2 and route traffic on the default LR, which is attached to it.

5.3. Generation of TAPs

This step is the most important as regards performance optimization and it is also the most complex, thus, we will restrict ourselves to the basic principles.

The overall procedure is divided into two parts :

i) Generation of a Base of Routes,
ii) Generation of TAPs.

This second step is itself divided into two parts :

i) First path optimization.
ii) Generation of complete TAPs.

5.3.1. Generation of a Base of Routes

The idea of this preliminary step is two-fold.

i) Restrict the optimization process to those routes which are worth being considered for Routing Path optimization (two paths correspond to a route : one for each direction).

   The optimization process maximizes residual capacity, i.e. favours shorter routes, in order to provide a congestion free rerouting.

ii) All routes are generally considered several times before inclusion in a table, therefore, it is computationally efficient to construct this route only once and to store it.

   The algorithm generates recursively :

   - All routes of length one.
   - Then all routes of length two.
   - etc...

   To a given pair of nodes, if the algorithm has to generate routes of a given length, it generate all routes of that length; it stops generating longer routes for that pair, if more than Nmin routes have already been generated. (Nmin = 8 in practice).

5.3.2. TAP Generation

TAP generation is based on a heuristic procedure, which tries to :

i) Optimize path selection for each flow individually, over the different peak periods.

ii) Maximum unused capacity in the network, under the condition that the network remains uncongested.

Path selection is based on a bottleneck analysis over the different peak periods and on path length (the shortest uncongested path is chosen).

A more formal description of the algorithm is as follows :

a) Order pairs of nodes, such as, neighbouring nodes with high traffic flows came first and distant nodes with small traffic flows come last.

b) Route one traffic flow at a time according to the above defined order. (For the two traffic flows corresponding to the same LR, routing is not necessarily symmetric) :

- For every element of the network (node or link) assign a maximum load level lmax.

- Evaluate each path which may be found in the base of routes, as follows :

  i) Route the flow corresponding to a given peak period on that path.

  ii) Compute the load lij of every element j of path i and compute the following quantity :

$$Ni = \max_{j} \frac{lij}{lijmax}$$

  iii) For path i consider the peak-period for which Ni is maximal and compute the quantity Qi :

  $Qi = Ni + Z1*sup(Ni-1,) + Z2*Li$

  where $Z1 = 100$, $Z2 = 10$ and

  Li = Length of Path i.

  iv) Select the path io, such that $Qio = \min_{i} Qi$

  v) Update the load of links and nodes of Path io.

Remarks

a) Because $Z1 >> Z2$, this procedure selects a path with minimal length Li, unless it is congested ($Ni > 1$).

b) Links corresponding to satellite circuits may be given a longer length to take into acount the propagation delay.

c) The load of the elements is updated with only a fraction of the traffic flow. This fraction is decreasing as the rank of the path to be added to the table is increasing.

5.4. Performance Analysis

As seen previously, only first Routing Paths are explicitly optimized for a given network state for all peak periods, therefore, it is necessary that subsequent paths give satisfactory results over all peak periods and for the most probable degraded modes. (1 link down, 2 links down, 1 node down).

This is done in the program which generates degraded modes and analyses the performances to be expected with the generated set of TAPs.

5.5. Generation of STs

This program generates channel numbers necessary to encode every Routing Path to be used in the network and stored in Switching Tables.

It minimizes the number of channel numbers to be used and works backwards on Routing Paths, starting with shorter routing paths first.

5.6. Generation of Routing Configuration File

This set of programs prepares from previously generated files, the routing configuration file in the proper format for direct load in the DISs.

These programs allow some modifications in the configuration and also perform consistency checks.

Since several peaks of traffic are observed, we may think of designing a Predictive Routing Algorithm, which would change its routes automatically at different hours. This would be difficult to realize with the old routing algorithm on the SITA Network, as routing is only destination dependant, routes changes have to be made synchronously in all switching nodes.

However, with the new routing algorithm, every node can change its TAPs independently of other nodes, as long as all Routing Paths have been properly encoded in the switching tables. Thus, time-dependent Predictive Routing is easy to implement.

Once there are Routing Paths for every peak period, several solutions are possible.

The most straightforward way would consist of generating one set of TAPs per peak period to load all of them in the DISs, which would address different TAPs at different hours. Another solution would consist of generating only one TAP for all peak periods, but this TAP would contain different Routing Paths corresponding to various peaks : to each Routing Paths corresponding to various peaks : to each Routing Path a "Time-up" condition would be attached in addition to "Link-up" conditions. These "Time-up" conditions would be locally generated by each node.

This second solution is less complex, in terms of on-line software and has less memory requirements than the first one, however, the off-line software may be a little more complex. This solution is presently under investigation for future implementation.

6. SUMMARY OF SITA ROUTING ALGORITHM

The algorithm can be summarized as follows :

- Assign an address, the HLD, to every system.
- Map every HLD pair on a Logical Route, which is a full duplex logical connection between DISs seen as entry/exit nodes of the DTN.

  This mapping is done so as to meet minimum and maximum traffic constraints on every LR.
- Associate an ordered set of Routing Paths to every LR, in each direction, called a TAP and store it in the origin DIS of that LR for the corresponding direction.
- Allocate a channel number on every link part of a Routing Path, to identify it.

  Switching of packets is done according to :

  . the LR number in the origin node,
  . input link and Channel Number on that link, in a transit node.

7. COMPARISON WITH OTHER ROUTING ALGORITHMS

In section 3 we have focussed our attention on the predictive versus adaptive issue. We will now make interesting connections between SITA Predictive Routing Algorithms and Dynamic Allocation Algorithms.

The main difference between these two algorithms is that the process of computation and generation of routing elements (LRs, RPs, CNs etc.) for the SITA algorithm is an off-line process, whilst the correspond-ing processs in these algorithms is, at least, partly an on-line process performed by a central facility (super-visor) and partly distributed among the switching nodes.

SITA PREDICTIVE ROUTING	SNA ROUTING
LINK	TRANSMISSION GROUP
CHANNEL NUMBER	EXPLICIT ROUTE NUMBER
ROUTING PATH	EXPLICIT ROUTE
TABLE OF ALTERNATE PATH	
LOGICAL ROUTE (+ PRIORITY LEVEL)	VIRTUAL ROUTE
	LIST OF VIRTUAL ROUTES

Figure V - SITA versus SNA Routing Concepts

Figure V shows a correspondence between SITA Predictive routing and SNA routing.

The link concept in SITA Network and the Trans-mission Group Concept in SNA are very similar entities; in both cases these entities may be composed of several independent physical circuits, and there may be several of these entities between two adjacent nodes.

The Channel Number Concept can be compared to the Virtual Circuit Number in Transpac, the Logical Record Number in Tymnet, or the Explicit Route Number (ERN) in SNA.

However, in each case, differences exist. For instance in SNA, forwarding of packet implies the use of Destination sub-area address plus the ERN, while in SITA routing, the Channel Number is used in conjunction with input link number to get output link number and Channel Number.

A Routing Path is pretty much the same concept as that of an Explicit Route, although the same Explicit Route is used in both directions (symmetric routing) and no such constraint exists for Routing Paths.

Only one Explicit route is associated to a Virtual Route, thus, in SNA, there is no entity corresponding to a TAP.

The LR entity is close to the Virtual Route Concept. For both routings we have the same number of priority levels, but in the LR case, different priorities exist on the same LR while they correspond to distinct Virtual Routes.

In SITA routing, at the time being, nothing similar to the list of Virtual Routes exists.

The failure of an Explicit Route causes the Virtual Route to be declared inoperative in SNA; and with the session reopening, a new Virtual Route with a new Explicit Route, it will be allocated to that session. Thus, Alternate Routing is performed at Virtual Route level in SNA and at Routing Path level in SITA routing.

In conclusion, we should say SNA routing is close to Predictive type, as Explicit Routes are off-line computed and that in SITA routing Dynamic allocation of LRs is foreseeable in the future.

8. CONCLUSION

While this article is being written, both the On-line software and the Off-line computer programs are under development.

Generation of life-size configurations have shown that memory requirements will be reasonable.

It has not been possible to present all the work and results on Predictive Routing here. There are some important possible improvements still under investigation. However, we hope to have raised some interest and provided an insight into Predictive Routing.

REFERENCES

/1/ Mischa Schwartz : Computer Communication Network Design and Analysis. Prentice Hall 1977.

/2/ Andrew S. Tanenbaum : Computer Networks. Prentice Hall 1981.

/3/ Mischa Schwartz, Thomas E. Stern : Routing Techniques used in Computer Communication Networks. IEEE Transactions on Communication Networks, Volume COM-28, No. 4, April 1980.

/4/ J.L. Grange, M. Gien eds. : Flow Control in Computer Networks. North Holland Publishing Company 1979.

/5/ Robert G. Gallagher : A Minimum Delay Routing Algorithm Using Distributed Computation. IEEE Transactions on Communications, Volume COM-25, No. 1, January 1977.

/6/ A. Segall : The Modeling of Adaptive Routing in Data Communication Networks. IEEE Transactions on Communications, Volume COM-25, No. 1, January 1977.

/7/ A. Segall : Failsafe Distributed Algorithms for Routing in Communication Networks. In /4/, pages 235-240.

/8/ J.M. Jaffe, F.H. Moss : A responsive Distributed Algorithm for Computer Networks. IEEE Transactions on Communications, Volume COM-30, No. 7, July 1982.

/9/ R. Fjellheim : Cernet Packet Routing Method. Computer Communications, Volume 3, No. 4, August 1980.

/10/ D.E. Sproule, F. Mellor : Routing, Flow and Congestion Control in the DATAPAC Network. IEEE Transactions on Communications, Volume COM-29, No. 4, April 1981.

/11/ J.M. McQuillan, I. Richer, E.C. Rosen : The New Routing Algorithm for the ARPANET. IEEE Transactions on Communications, Volume COM-28, No. 5, May 1980.

/12/ W. Chou, J.D. Dowell, A.W. Bragg : Comparative Evaluation of Deterministic and Adaptive Routing. In /4/, pages 257-285.

/13/ H. Rudin, H. Muller : On Routing and Flow Control. In /4/, pages 241-255.

/14/ La Roy W. Tymes : Routing and Flow Control in Tymnet. IEEE Transactions on Communications, Volume COM-29, No. 4, April 1981.

/15/ J.M. Simon, A. Danet : Controles des Ressources et Principes du Routage dans le Reseau TRANSPAC. In /4/, pages 33-44.

/16/ G.J. Chretien, W.M. Konig, J. Rech : The SITA Network in Computer Communication Networks edited by R.L. Grimsdale and F.F. Kuo. NATO Advanced Study Institute Series, Noordhoff-Loyoben, 1975.

/17/ J.M. Kaliszewski : Routing Algorithm and Route Optimization on the SITA Data Transport Network. Proceedings of GLOBECOM 83 Conference San-Diego 1983.

/18/ J.M. McQuillan, I. Richer, E.C. Bertsekas : ARPANET Routing Algorithm Improvements. Second Semiannual Report, October 1978, Report No. 3840, BBN.

/19/ E.C. Rosen : Vulnerabilities of Network Control Protocols : An Example. ACM SIGSOFT, Software Engineering Notes, Volume 6, No. 1, January 1981.

/20/ J.D. Atkins : The Transport Network of SNA. IEEE Transactions on Communications Volume COM-28, No. 4, April 1980.