FACTA UNIVERSITATIS (NIŠ)
Series: Electronics and Energetics vol. 8, No. 1 (1995), 103–118

THE ALTERNATE ROUTING – PROS AND CONS FOR AN APPLICATION IN COMPUTER COMMUNICATION NETWORKS

Tadeus Uhl

Abstract. This paper provides an investigation of the routing rules with shortest paths and alternative routes. In the study the average delay in the network, the average accessibility in the network and the implementation effort for the routing protocol are assumed to be the main performance values. The advantages and disadvantages of the rules mentioned above will be presented and discussed. The study demonstrates that the choice of a routing rule in computer networks represents an optimization problem. The obtained results of this paper yield an important contributions to the solution of this problem.

1. Introduction

The routing of transmitted data blocks, usually packets, is one of the fundamental control problems in computer communication networks. During the last years several different routing rules have been proposed and new ones are still under development, but only a few of them have gained in importance for practical applications [1].

For routing rules attempting to optimize network performance several different performance criteria have already been considered. Among others these are i.e. the number of hops between source and destination, the coast of all involved facilities, the delay to reach a certain destination, and the network throughput. Further criteria for an assessment of the properties of routing algorithms are: computational simplicity, adaptiveness to changing traffic and topologies or robustness, stability, fairness and optimality [2]. Yet there is no overall optimal algorithm fulfilling all these criteria in a best way.

Manuscript received February 4, 1995.
The author is with Institute for Telecommunications, Fachhochschule Flensburg, D-24943 Flensburg, Germany.

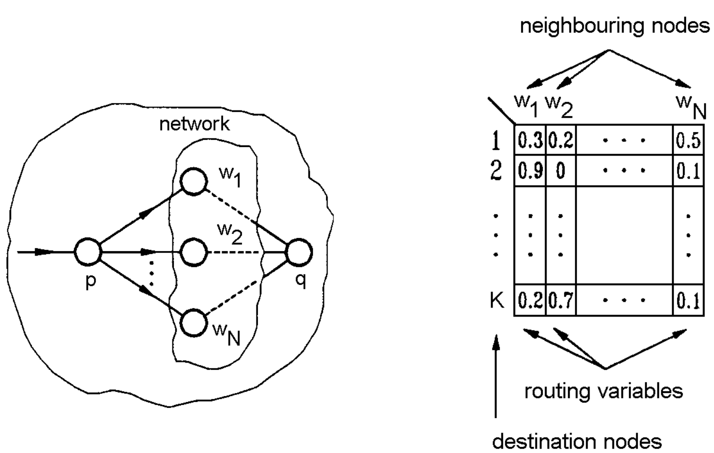
In fact, there are some contrary effects between specific criteria in certain implementations of routing schemes.

In numerous cases one has to ask the important question, wether a routing rule with single routes or a routing scheme with multiple paths (the so–called alternate routing) should be used. For an answer to this question, several of the criteria mentioned above have to be taken into consideration. In the context of this work an attempt is made to give a proper solution to this problem.

First, the routing problem will be described by a detailed definition. Then two distinct groups of algorithms for the determination of routing tables will be presented. Chapter 3 comprises the description of the model under study and a discussion of the obtained results. The paper is closed with a summary.

2. Routing Problem

The main task of the routing rule is to find a path through the network on which the data blocks (named as packets in the sequel) are transmitted from the source to the sink. Seen from current communication node this means to choose the next neighbouring node to which the packets have to be sent. The problem is illustrated in figure 1.



a) b)

Fig. 1. Definition of the routing problem.

Figure 1. also shows the so-called routing table. Elements of this table, usually described as routing variables, indicate through which path the packets are to be sent. For an efficient determination of the routing table entries special algorithms must be applied.

2.1 Classifcation of routing rules

The characteristic of routing rules can be described by following parameters:

1) number of possible routes:
   - non alternate (single routes),
   - alternate (multiple routes);
2) topicality of the update information on the network state:
   - fixed (independent from time),
   - adaptive (dependent from time);
3) amount of the update information on the network state:
   - local (information extracted from a part of the network),
   - global (information collected the whole throughput network);
4) update interval of information on the network state:
   - rhythmic (periodic update),
   - arhythmic (asynchronous update);
5) place of routing decision:
   - centralized (in one special node),
   - distributed (in every node);
6) objective of the routing strategy:
   - system optimization (minimisation of the overall cost function),
   - user optimization (minimisation of the cost function for a certain connection between a pair of users).

As an example for this classification the new routing rule in the ARPANET can be considered [3]. With regard to this classification, the routing rule can be described as adaptive, non alternative, distributed, arhythrmic with global information on the network state.

2.2 Algorithms for determination of paths

Currently several algorithms for the generation of shortest paths in computer networks are existing. The so–called Bellman–Ford–Moore algorithm [4], [5] is based on Bellman's principle of optimality [6]. This algorithm iterates on number of hops in a path. The Dijkstra algorithm is another popular *shortest–path algorithm* (SPA) [7]. This algorithm iterates on the length of a path. Both algorithms allow to seek the shortest paths from a given source node to all other nodes. The Floyd–Warshall algorithm computes shortest paths between all pairs of nodes [8]. It iterates on nodes allowed as intermediate nodes in paths. All three algorithms are implemented as a labelling algorithm. A description of technical details is omitted here, the interested reader is referred to [2].

In case of small and dense networks Dijkstra algorithm is faster than the Bellman–Ford– Moore algorithm. At the moment it is used in the routing protocol of the ARPANET [3]. The Floyd–Warshall algorithm is implemented in some networks with centralized routing, for instance the TYMNET network [9].

The number of algorithms for a generation of alternate paths in network is decisively smaller. Here, as the main difficulty, the so–called loop problem has to be regarded. Some algorithms for the computation of loop–free routing tables with alternative paths are described in detail in [10]. A brief summary is given below.

There are three different classes of loop-free routing tables.

Class I: No additional information is used (e.g. shortest path routing).

Class II: All information available at the switching node is used (e.g. identification of the node from which the packet has been received).

Class III: Additional information is transmitted with the packets (e.g. a trace of visited nodes).

In [10] it has been shown that algorithms of the second class yield best results. Within class II it is possible to describe a so-called *iterating optimizing algorithm (IOA)* which is able to provide a satisfactory number of alternate paths at each node. As an incoming link must not be used again for routing the received packets, the number of alternate paths is sometimes decreased by one in order to avoid ping–pong loops.

Because the Dijkstra algorithm is very efficient and the iterating optimizing algorithm provides a satisfactory number of loop–free alternatives at every node, both will be used to calculate routing tables within this study.

3. Model Under Study

3.1 Network topology

The investigations of this paper have been carried out for the network structures shown in figures 2–4. The first network is rather small but fully connected. The second topology is asymmetric and only connected to a small extent. The network from figure 4 is characterized by a high density of connections and includes a quite large number of nodes and channels.

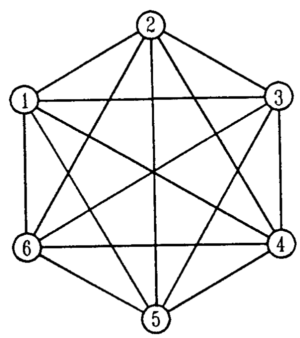


Fig. 2. Topology of network 1.

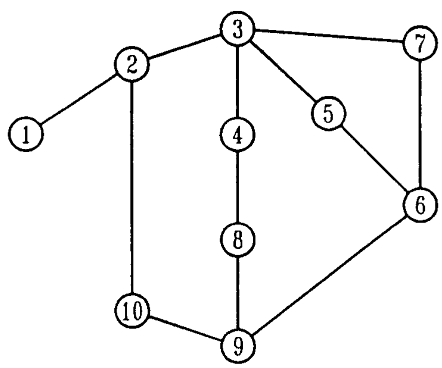


Fig. 3. Topology of network 2.

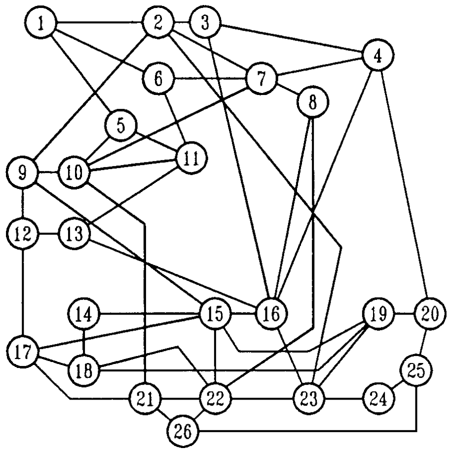


Fig. 4. Topology of network 3.

3.1.1 The obtained routing tables and their accessibility

In this study the shortest path algorithm by Dijkstra and the iterating optimizing algorithm by Mersch/Uhl are applied (see chapter 2.2). Tables 1 and 2, respectively, give examples for possible routing tables in case of network 2.

Table 1. Routing table for node 2 in case of SPA

	1st route		2nd route		3th route	
destination node	neighbour node	route length	neighbour node	route length	neighbour node	route length
1	1	1	–	–	–	–
2	–	–	–	–	–	–
3	3	1	–	–	–	–
4	3	2	–	–	–	–
5	3	2	–	–	–	–
6	10	3	–	–	–	–
7	3	2	–	–	–	–
8	10	3	–	–	–	–
9	10	2	–	–	–	–
10	10	1	–	–	–	–

Out of the length of the paths shown in the tables above the corresponding routing variables (named as α, $0 \leq a \leq 1$, in the sequel) can be determined by using a suitable algorithm.

Table 2. Routing table for node 2 in case of IOA

	1st route		2nd route		3th route	
destination node	neighbour node	route length	neighbour node	route length	neighbour node	route length
1	1	1	10	5	–	–
2	–	–	–	–	–	–
3	3	1	1	3	10	5
4	3	2	10	4	–	–
5	3	2	10	4	1	4
6	10	3	3	3	1	5
7	3	2	10	4	–	–
8	10	3	3	3	1	5
9	10	2	3	4	1	4
10	10	1	3	5	–	–

The average accessibility in the network $E[AC]$ is defined as follows

$$E[AC] = 1 - E[FA] \tag{1}$$

where: $E[FA]$ describes the average fault probability of the network.

The average fault probability of the network will be determined under use of the modified algorithm by Gaudreau presented in [11]. Within a new modification now the fault probability of channels will be combined with routing variables in the networks. The obtained results for the networks of figure 2-4 are shown in table 3 (fault probability of the channel is set to $10^{-5}s$.

Table 3. Average accessibility as a function of routing rules

	average accessibility		
algorithm	network 1	network 2	network 3
SPA	1 - 1.20E-4	1 - 2.11E-4	1 - 2.47E-4
IOA	1 - 9.00E-9	1 - 7.33E-5	1 - 1.57E-5

It is remarkable that the alternate routing scheme leads to a considerably better accessibility in the network compared to routing with single paths.

3.1.2 Implementation aspects

A distributed, adaptive routing scheme typically consists of a number of separate processes including:

a) a measurement procedure determining pertinent network characteristics,
b) a protocol for disseminating information about these characteristics,
c) a procedure for the determination of the routing tables, and
d) a method for the decision (based on routing tables) which channel should be chosen: in other words - the mechanism for packet distribution.

The following chapter provides a presentation on aspects of implementation methods, cf. d).

The routing variables $\alpha_i,\ i = 1, 2, \dots, N$ (elements of routing tables) yield the optimum distribution (with respect to a given criterion). All kinds of implementation schemes are bearing a common problem. The actual routing leads to an empirical distribution $\alpha_i^* = pa_i/pa$ resulting as the quotient between the number pa_i of packets routed through link i and all packets pa. Due to the fact that the number of distributed packets within one update period is limited and the mechanisms itself show several weaknesses, the distribution a_i^* differs more or less from the intended optimum α_i. As a consequence, the average transmission delay ET^* is unable to reach the minimum value ET - it is larger.

The classical method for the distribution of packets uses the random number generators. It is a stochastic mechanism which works quite simple. In practice, the unity interval [0,1) is divided into subintervals of length α_i $(i = 1, 2, \dots, N)$, so that the generator offers N different random events with probabilities α_i. After the calculation of a new number and the determination of the relating random event, the packet will be sent to the corresponding route.

The second implementation method is the so–called deterministic routing scheme [12]. In contrast to the first method this method is a deterministic mechanism for packet distribution. Its principal idea is to calculate a routing sequence S_p for node p by

$$S_p = \{s_1, s_2, \dots s_k, \dots, s_m\} \tag{2}$$

whereas $s_k = i$ means a decision to send the k-th packet to output link i $(i = 1, \dots, N)$.

Assuming that the routing variables α_i are given, each decision for an outgoing link is counted in one of N decision counters d_i. It is the aim to minimize the sum of squared differences

$$\sum_{i=1}^{N}(\frac{d_i}{m} - \alpha_i)^2 \tag{3}$$

between the fraction of packets routed to link i in a total of m packets and the probability α_i. Both should be as close as possible to each other.

For the routing of a certain packet (e.g. packet m) one has to calculate all errors $e_i,\ i = 1, 2, \ldots, N$, that would occur in case of sending the packet to output i,

$$e_i = \sum_{\substack{j=1 \\ j \neq i}}^{N}(\frac{d_j}{m} - \alpha_j)^2 + (\frac{d_i + 1}{m} - \alpha_i)^2. \tag{4}$$

The resulting link k then is determined by the error e_k,

$$e_k = \min_i[\min_{e_i}[e_1, e_2, \ldots e_N]],\ i = 1, 2, \ldots N, \tag{5}$$

which is the minimum value e_i with smallest index.

There are several problems related to the implementation of the deterministic routing sequences. Due to the fact that all errors must be calculated again for each routing decision, the method implicates a high computational complexity and calculations require a large amount of time. Another problem affects the limited capacity of decision counters. In case of a long term operation of the network these counters may run into an overflow. One solution to that conflict is the reset of all counters at the beginning of a new update period.

The third and last method which has taken into consideration for a comparison is the so–called splitting scheme that was suggested in [13]. Like the sequence algorithm it is a deterministic method, but it only comprises a small implementation complexity. Once more one single update period be observed with given values α_i for the optimal packet distribution. An index m be introduced counting the total number of decisions. The scheme components b_i are initialized at the starting point of the algorithm as indicated in the following equation

$$b_i(0) = \alpha_i(m)\ ,\ i = 1, 2, \ldots, N. \tag{6}$$

It is, however, possible to divide the described algorithm into two distinctive parts:

1) choice of an outgoing channel,
2) correction of the vector $\underline{b}$ using a correction coefficient q $(0 < q < 1)$.

It is important that the condition

$$\sum_{i=1}^{N} b_i(m) = 1 \tag{7}$$

must always be fulfilled.

To find a decision for the routing of the m-th packet it is necessary to calculate a new vector $\underline{\omega}(m) = (\omega_1, \omega_2, \dots \omega_N)$ containing the differences

$$\omega_i(m) = \alpha_i(m) - b_i(m) \ , \ i = 1, 2, \dots N. \tag{8}$$

The outgoing link for which $\omega_i(m)$ yields the absolute maximum value is chosen. If there are several equal maximum values, it is necessary to choose the direction for which the corresponding α_i is maximum.

After the choice of a certain direction k for packet m the algorithm recalculates $\underline{b}(m+1)$ from the old $\underline{b}(m)$. This is performed with the aid of q. For each link i except k $b_i(m+1)$ can be obtained by

$$b_i(m+1) = q \cdot b_i(m) \ , \ i = 1, 2, \dots, N \ , \ i \neq k. \tag{9}$$

The value of $b_k(m+1)$ has to comply with

$$b_k(m+1) = 1 - \sum_{\substack{j=1 \\ j \neq i}}^{N} b_i(m). \tag{10}$$

$\underline{b}(m)$ can be interpreted as a vector that reflects the choice frequencies. $\underline{\omega}$ represents the deviations between these frequencies and the theoretic probability α_i. The algorithm tries to minimize $\underline{\omega}$. The effectiveness of the splitting scheme depends on an appropriate choice of q.

The implementation methods mentioned above have been compared in [14]. All methods show critical behaviour for small arrival rates (low number of packets). The sequence procedure gives best results (smallest deviations from the optimum delay), but the method is extremely slow. Random distribution (as the classical method) appears to be a fast, but rather inaccurate scheme. An implementation should only be taken into consideration in case of high arrival rates. The application of the splitting scheme leads to fairly good results concerning the discrepancy between theoretical and empirical packet distributions, though it is much faster than the sequence mechanism. However, the accuracy of the method strongly depends on the appropriate choice of the correction coefficient.

In contrast to adaptive routing the implementation method for routing with single paths is very simple. It is sufficient to detect which element of the routing table contains a value of 1.

3.2 Node model

The model of the communication node is shown in figure 5.

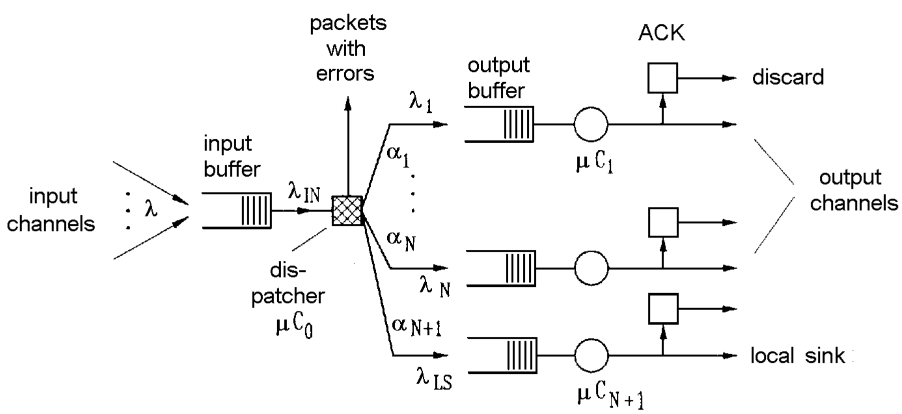


Fig. 5. Node model.

In the presented model a Poisson arrival process with the parameter λ [packets/s] will be assumed. The lengths of message packets (denoted as $1/\mu$) are negative–exponential distributed with a mean value of 1024 [bit/packet]. The acknowledgement packets have a length (denoted as $1/\mu_{ack}$) of 64 [bit/packet]. All buffers are of infinite capacity. For every channel a speed of 19,6 [Kbit/s] is assumed. The dispatcher speed be assumed by C_0 =100 [Kbit/s].

3.2.1 Analytical performance evaluation tool

A network consisting of K nodes and I channels is considered.

For the investigations in this paper the analytical method for performance evaluation of loop–free routing rules in computer networks has been used as presented in [15]. A short description of the method is given below.

Step 1: Set the length of each channel in the network to value 1. Determine routes for every pair of nodes by the use of the SPA and IOA scheme.

Step 2: Determine the flow in the channels $(\lambda_1, \lambda_2, \dots, \lambda_I)$ according to the procedure in [15].

Step 3: Calculate the load factors $\rho_i = \lambda_i/\mu C_i$ for the channels $(i = 1, 2, \dots, I)$.

Step 4: Modify the average packet length $1/\mu_i$ according to the formula (consideration of the acknowledgement procedure)

$$\frac{1}{\mu_i} = \frac{1}{\mu} + \frac{1-\rho_i^2}{\mu_{ack}} \ , \ i = 1, 2, \dots, I. \tag{11}$$

Step 5: Calculate the delays in the input and output area of nodes by use of the known formulas for $M/M/1$ systems, e.g. $\tau = (\mu C - \lambda)^{-1}$.

Step 6: Calculate the total delay τ_G in the network according to the following relation

$$\tau_G = \frac{1}{\lambda_G}\Big\{\sum_{i=1}^{I} \lambda_i \cdot \tau_i + \sum_{k=1}^{K} \Big[(\lambda_{IN}^{(k)} \cdot \tau_{IN}^{(k)}) + (\lambda_{LS}^{(k)} \cdot \tau_{LS}^{(k)})\Big]\Big\}. \tag{12}$$

where are $lambda_G$ total average arrival rate of the network, $\lambda_{IN}^{(k)}$ average arrival rate in the input area of the k-th node, $\lambda_{LS}^{(k)}$ average arrival rate in the output area of the k-th node for local sink and $\tau_x^{(y)}$ average delay according to the formula for $M/M/1$ systems.

This analytical method has been implemented and tested in many examples (cf. [15]). The analytical results have also been validated by simulations. The tests have shown that the method gives a good accuracy and is suitable for solving different problems in computer networks.

3.2.2 Obtained results

The analytical study has been carried out for the networks shown in figures 2–4. In the investigation the average delay in the network has been

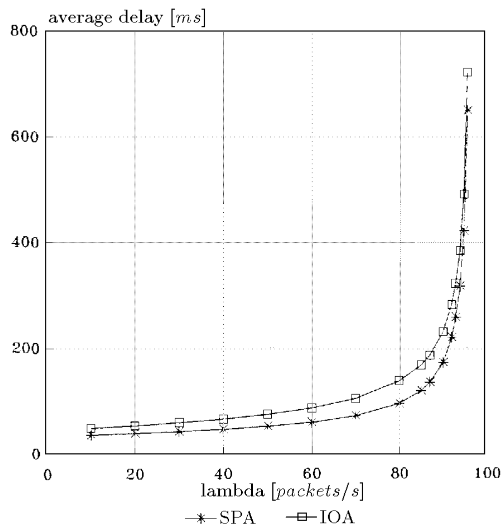


Fig. 6. Average delay for network 1 as a function of a network load and different routing rules

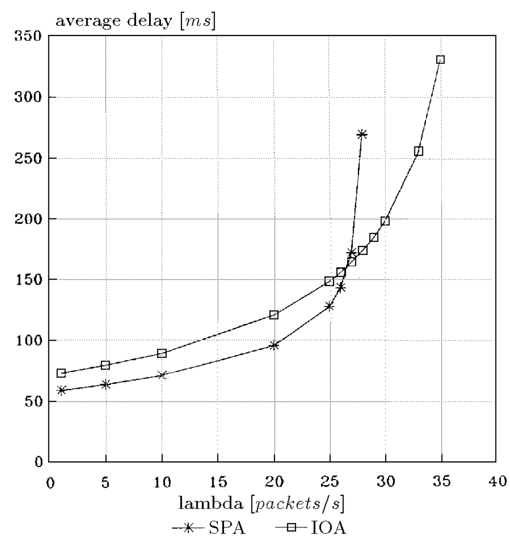


Fig. 7. Average delay for network 2 as a function of a network load of and different routing rules

assumed as the main performance value. The obtained results are displayed in figures 6–8.

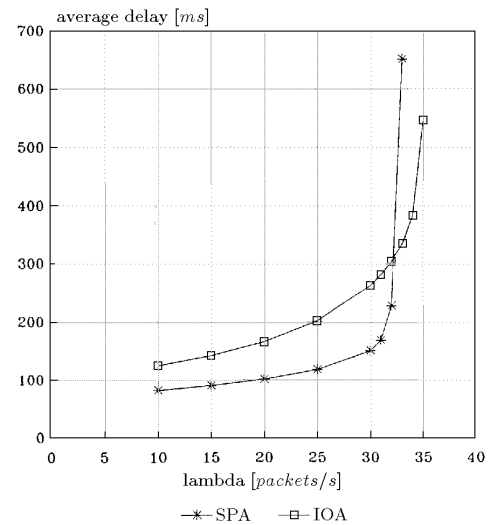


Fig. 8.Average delay for network 3 as a function of a network load and different roufing rules

The curves shown that in case of a light load the routing with shortest paths obviously is the best rule independent of the network topology. In the case of high traffic load the alternate routing shows main advantages. The only exception of this rule occurs when fully connected networks are present. Then the alternate routing do not achieve better results than routing with shortest paths for every network load.

4. Conclusion

This paper has presented and compared routing rules with single and multiple routes. In the investigation the average delay in the network, the average accessibility and the implementation effort of the routing protocol have been used as the main performance parameters. In detail the following results can be concluded.

The algorithms for determination of routing tables in case of the shortest path first scheme are quite uncomplicated and require only a small computing time. The implementation procedure of the SPF rule is rather simple. The latter routing method yields the best results in case of low and average load situations. The accessibility onto the network with SPF routing turns out to be very low which is a main problem. That's why this routing rule should only be used in networks with small average fault probability of the channels.

The algorithms for the determination of routing tables in case of alternative routes are complicated and need a large amount of computing time. The practical implementation of alternate routing is accompanied by a considerable computing effort. According to the analysis here the implementation methods with random number generators are a tolerable compromise between velocity and accuracy. The alternate routing is the bes strategy in case of overload situations in the network. The accessibility onto the network with alternative routes is very large. This is a important aspect for practical applications.

In summary it can be finally stated that the choice of a routing rule in computer networks represents an optimization problem. The obtained results in this paper yield important contributions to the solution of this problem.

REFERENCES

1. Schwartz, M. and Stem, T.: *Routing Techniques Used in Computer Networks.*. IEEE Trans. on Commun., vol. COM–28, no. 4 (April 1980) pp. 539–551.
2. Spragins, J., Hammond, J. and Pawlikowski, K.: *Telecommunications: Protocols and Design.* Addison–Wesley Publishing Company, Inc. (1991).
3. McQuillan, J., Richer, 1. and Rosen, E.: *The New Routing Algorithm for the ARPANET.* IEEE Trans. on Commun., vol. COM–28, no. 5 (May 1980) pp. 711–719.
4. Bellman, R.: *On a Routing Problem.* Quarterly of Applied Mathematics, no. 16 (1958) pp. 87–90.
5. Moore, E.: *The Shortest Path Through a Maze.* Proc. of Inter. Symposium on the Theory of Switching, Part II, Cambridge: Harvard University Press (1959) pp. 285–295.
6. Bellman, R. and Dreyfus, S.: *Applied Dynamic Programming.* Princeton: Princeton University Press (1962).
7. Dijkstra, E.: *A Note on Two Problems in Connexion with Graphs.* Numerische Mathematik, no. 1 (1969) pp. 269–271.
8. Floyd, R.: *Algorithm 97.* Shortest Path. Communications of the ACM, no. 5 (1962) pp. 345–350.
9. Tymes, L.: *Routing and Flow Control in TYMNET.* IEEE Trans. on Commun., vol. COM-29, no. 4 (April 1981) pp. 392–398.
10. Mersch, N. and Uhl, T.: *Algorithms for the Determination of Loop-Free Alternarive Routes in Communication Networks.* AEÜ, no. 2 (March 1992) pp. 93–101.

11. Gaudreau, M.: *Recursive Forinulas for the Calculation of Poin-to-Point Congestion.* IEEE Trans. on Commun., no. 3 (March 1980) pp. 313–316.

12. Yum, T.: *The Design and Analysis of a Semidynamic Deterministic Routing Rule.* IEEE Trans. on Commun., vol. COM–29, no. 4 (April 1981).

13. Parshenkov, N. and Barannik, K.: *Performance Evaluadon of Probabilistic Algorithm for Adaptive Routing with Flow Splitting in Packet Switching Networks.* Internal Report of the Institute of Problems of Information Transmission of the Academy of Sciences of the USSR, Moscow (1988).

14. Mertsch, M. and Uhl, T.: *Implementation Methods for Alternate Routing Strategies.* Proc. of 2nd Gennan-Soviet Seminar on Flow Control, Academy of Sciences of the USSR, Moscow (1989).

15. Uhl, T.: *An Analytical Method for Performance Evaluation of Loop-Free Routing Rules in Computer Networks.* Wirtschaftsinformatik, vol. 32, no. 4 (August 1990) pp. 352–358.