University of Qatar

ENGINEERING JOURNAL OF QATAR UNIVERSITY

VOL. 1 1988

PUBLISHED BY THE FACULTY OF ENGINEERING
QATAR UNIVERSITY

Engineering Journal of Qatar University, Vol. 1, 198—.

"PERFORMANCE EVALUATION OF A CLASS OF ADAPTIVE BIFURCATED ROUTING ALGORITHMS"

By

N.M. Ayad & Dr. F.A. Mohammed
Atomic Energy Establishment,
Inshas, Egypt.

M.S. Metwally,
Electrical Engineering,
Qatar University, Doha,
Qatar, Arabian Gulf.

ABSTRACT

With the cost of computation decreasing, packet-switched computer communication networks are becoming increasingly cost effective. In this paper a class of bifurcated routing algorithms is considered. One of the algorithms uses distributed control and routing decisions based on stochastic measures. The second one is a deterministic routing algorithm that uses localised deterministic routing decisions based on centralised measures. The third algorithm uses "Learning Automata" principle which is a promising technique because of its simplicity and ease of implementation.

The three algorithms have been modelled and simulated to evaluate their performance under network conditions. A quantitative investigation of the three algorithms under various traffic conditions has been carried out. The overall average time delay, the average retransmission probability, and the average response time have been taken as a common basis of the comparative study. Regarding these measures, the learning automata has proved to be the best.

1. INTRODUCTION

For packed switched networks, it seems that significant performance advantages are associated with those systems that use advanced routing techniques, the study of which is the object of this paper. Three adaptive routing algorithms have been modelled, evaluated, and modified. The algorithms are the distributed computation, the deterministic rule, and the learning automata algorithms. The three algorithms employ the bifurcated technique, but they are totally distinct regarding their control features, principles of operations, and initialisation procedures.

First each of the three algorithms is briefly described (1, 2). Then, quantitative investigations of the performance of the algorithms under different affic conditions have been performed. Finally, conclusions are illustrated.

2. DISTRIBUTED COMPUTATION ALGORITHM

The distributed computation routing algorithm depends initially on the existence of the shortest path algorithm. The shortest path provides a loop free route to start with, while the distributed computation algorithm equalises the incremental or marginal delay over each link of the network.

The algorithm is applied independently at each node and successively updates the routing table at that node based on information communicated, under protocol guarantee, between adjacent nodes about the node marginal delay to each of its destinations. The communication protocol together with the loop freedom property insures fast, proper, and generally good communications between network nodes. Also, these properties make the algorithm look attractive for both congestion measure and control.

The greatest advantage of this algorithm is in its distributed computation nature, which makes it favourable for large and diversely distributed networks. The model structure can be viewed as if each node individually bifurcates its coming traffic over all its outgoing links. Routing bifurcation ratio control constraints prevent isolated loops and prohibit traffic from looping back into the network after it reaches its destination.

The traffic bifurcation can be updated, either in a periodic manner or when necessary, to permit routing adaptation with varying network conditions. The updating is incremental, decreasing the volume of traffic directed to the links having larger values of marginal delay. This link marginal delay is simply calculated by linearly accumulating the times between each packet departure and arrival on that link. Thus, the algorithm breaks into two parts, a protocol between nodes to calculate the marginal delay and an algorithm for modifying the routing bifurcation.

3. DETERMINISTIC RULE ALGORITHM

The deterministic rule algorithm depends extensively at its start on the existence of a stochastic algorithm known as the flow deviation. The stochastic algorithm insures a near minimum average time delay, while the deterministic one improves this minimum by maximising the traffic bifurcation in the network (4).

In this algorithm, an independent routing decision sequence based on information about the delays on the lines connecting to the network, is made for each node. The delay on a particular line is measured at the nodes attached to the lines. Practically, there must exist a network control centre (NCC) to recalibrate the optimum flow distribution whenever there are significant changes in the external flow pattern at the local nodes. The maximum traffic bifurcation of each commodity at each node is determined. The patterns of traffic bifurcation are sent to the respective nodes, either periodically or whenever found necessary according to the adopted criteria. The decision sequences resulting from the bifurcation patterns received are generated from each node. The messages after being classified by commodities, are then routed to the decision sequences associated with their commodity.

The complex computation overhead, encountered at initialisation in this algorithm, is really a drawback since it represents an added delay. However, the most important advantage of this algorithm is that the local nodes need send their estimates of the external arrival rates to the NCC and receive the bifurcation traffic patterns whenever there is a significant change. Thus, if the traffic update is not too frequent, the traffic overhead is minimum.

This model logically combines the traffic flow deviation method with the deterministic routing of Yum. The flow deviation is substituted for by the shortest path algorithm (5) and the minimum delay. The model starts by determining the shortest paths from each source node to its final destination. Then a total traffic flow assignment is carried out with the objective of minimising the average time delay. Finally, traffic is split over outgoing links to improve the overall average time delay.

The proposed algorithm splits the total traffic allowed to be bifurcated over the available route. The bifurcated traffic is routed over assigned appropriate paths in the form of iterative deterministic routing sequences. The sequences are generated according to a dictated routing probability (5). The routing probabilities are updated, depending on traffic delay measurements, at slower rates than the routing decisions.

The general structure of the model can be viewed as an initialisation part followed by a deterministic routing sequence generation algorithm. The initialisation part determines the traffic flow of each link that minimises the average time delay of the computer network. The deterministic decisions are taken on the basis of specified maximum traffic bifurcation with the objective of improving that minimum average time delay. This approach being adaptive, shows its capability to operate with comparatively reasonable time delays under varying traffic conditions.

4. THE LEARNING AUTOMATA ALGORITHM

The learning automata model provides a novel, computationally attractive model for monitoring the routing procedure in the PSN. The routing complexity, which depends on the network scale and its topological nature, will be greatly reduced. On the other hand, since the model is dependent only on the local data it is more realistic due to the distributed nature of the computer networks.

The model is a logical evolution combining the advantages of distributed computation, the deterministic rule, and the learning automata features. It appears to be a simple system that is capable of solving problems with a high degree of uncertainty. The algorithm uses Yum's technique (4) to spread traffic. However, it does not do so by splitting the traffic randomly. Traffic splitting is achieved by, for the traffic to be routed, using routing decision sequences generated at successive time steps with the aid of Yum's algorithm. The distributed algorithm technique (3) is used to calculate routing decisions at each node. Since the desicions are being viewed as automata actions, the whole network will resemble an environment whose reaction is either to penalise or reward a decision by increasing or decreasing its blocking probability. Thus, equalising the blocking probability around its minimum, to ensure absolute expediency for the automata behaviour, is the object of the reinforcement scheme that acts on the routing bifurcation strategy. Such calculations are less frequent than those for the distributed routing decision sequences. An expedient linear reinforcement scheme with scaler parameters has been used. To satisfy the criteria, accurate stochastic measurements have been performed on the traffic flow at successive periods or epochs. These epochs are of course larger than the decision time periods.

As with all decentralised adaptive policies, oscillatory behaviour was expected in some cases. The initially supplied fixed reasonable routing table was a good solution. However, those first choices do not affect the asymptotic behaviour of the model.

5. SIMULATION RESULTS

The three algorithms are implemented on a simple network of 10-nodes, Figure (1), using the standard simulation techniques (6), and the main simulation program. The flow chart of the simulation program is shown in Figure (2). The delays encountered by each packet are recorded and averaged over respective time steps and epochs. Other measures which are useful from the user point's of view are the average transit time, the average retransmission probability, and the network average utilisation. These are provided by all simulations. The common parameters taken as a basis of comparison are:

1. Retransmission Probability

The average values of the retransmission probability, averaged over one epoch or modification period, are plotted separately in Figures (3), (4) and (5). When the first replications, which are subject to the simulator transit conditions are neglected, the three algorithms converge in a comparable time. Figure (6) illustrates that the retransmission probabilities are averaged over all the experiment period i.e. 10 epochs or modification periods each of 32 s, at an average network load of 7200 messages/hour/node. For the learning automata case, the retransmission probability reaches its maximum faster than in the other two cases. In addition, it has the smallest absolute minimum despite the initial oscillatory period. On the other hand, the distributed computation gives the highest retransmission probability values.

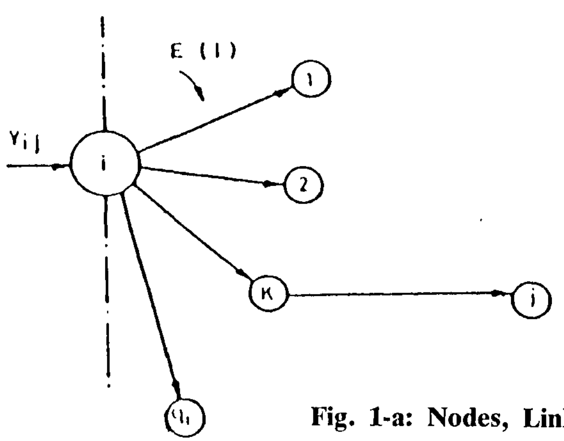

Fig. 1-a: Nodes, Links & Traffic Representation.

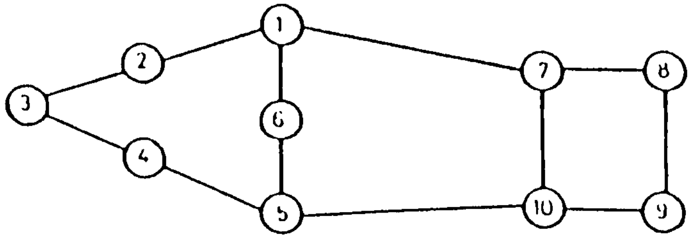

Fig. 1: Sample Network (10 Nodes, 24 Links).

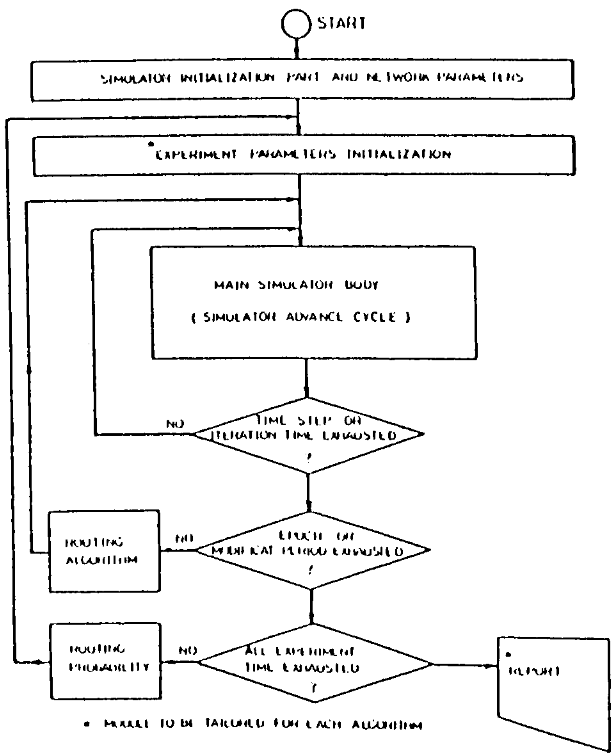

Fig. 2: The Simulation Program Flow Chart.

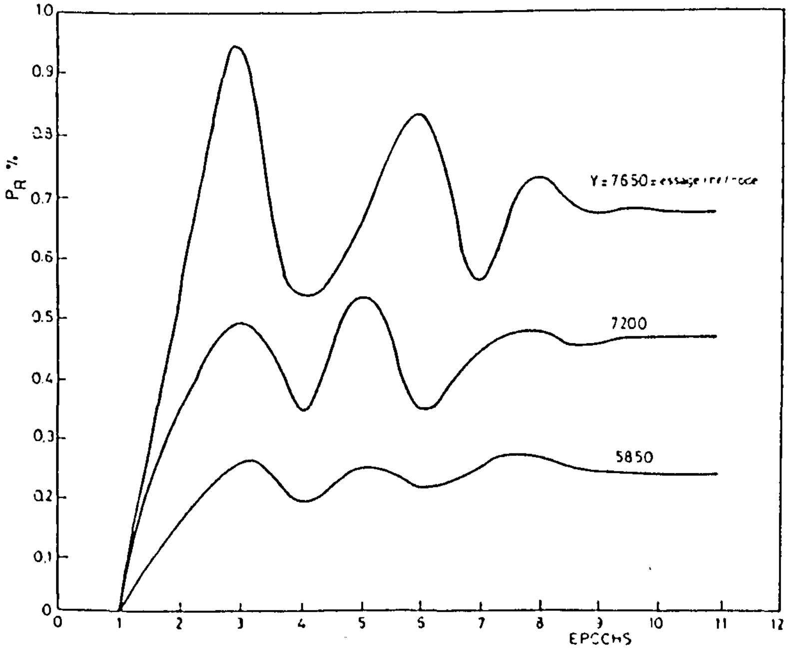

Fig. 3: Average Overall Retransmission Probability for L.A..

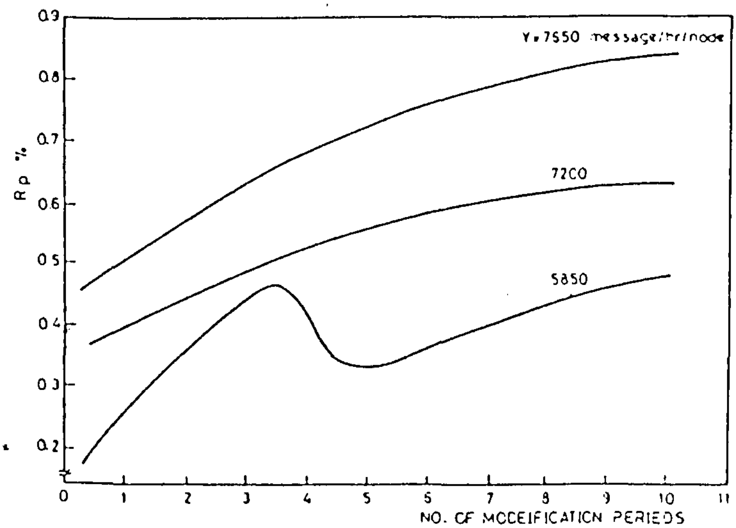

Fig. 4: Average Overall Retransmission Probability for YUM.

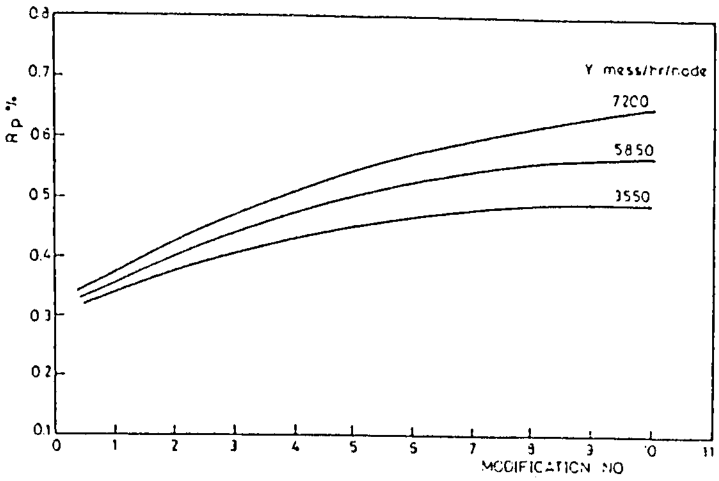

Fig. 5: Average Overall R_p for D.C. Algorithm.

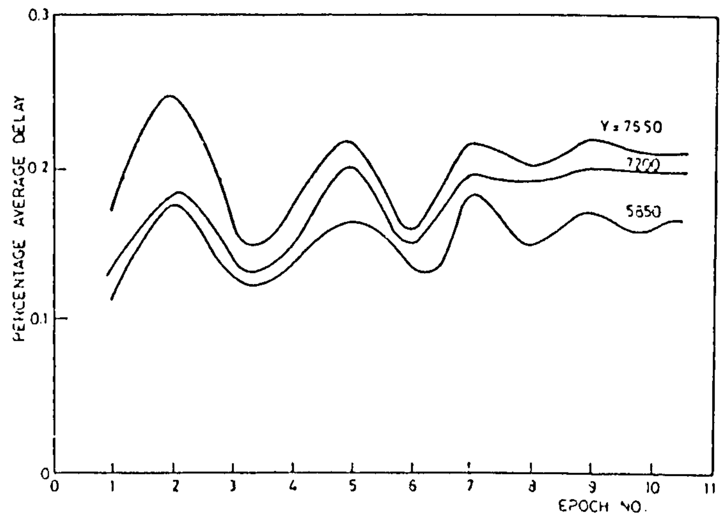

Fig. 6: Average Overall Time Delay for L.A..

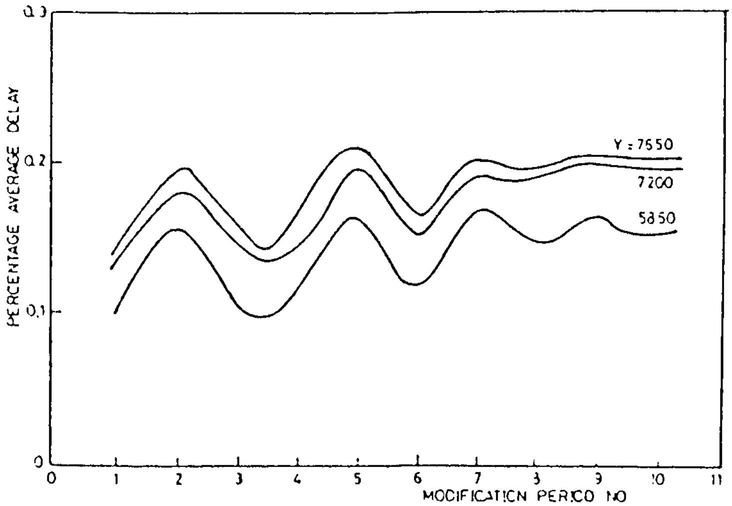

Fig. 7: Average Overall Time Delay for YUM.

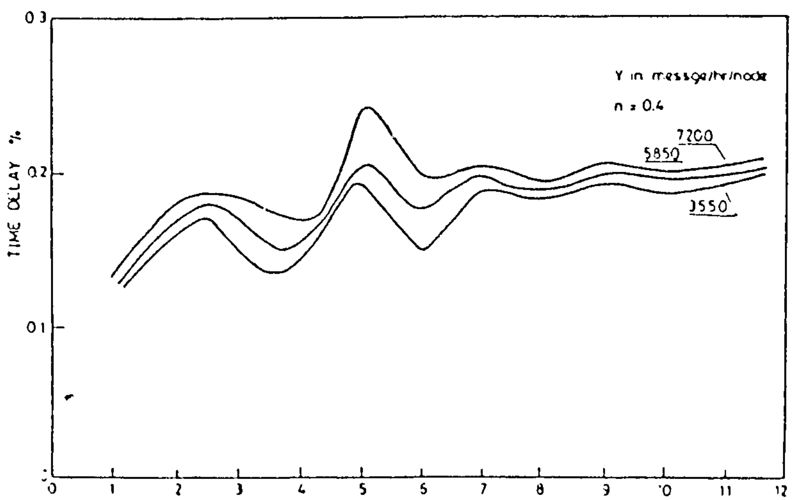

Fig. 8: Average Overall Time Delay for D.C. Algorithm.

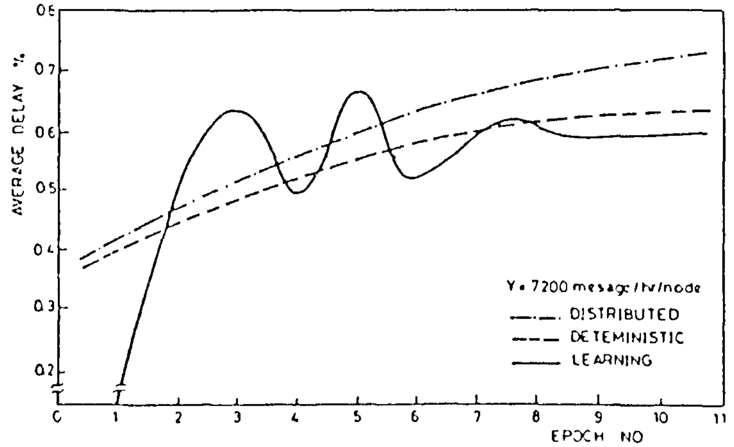

Fig. 9: Average Overall Retransmission Probability.

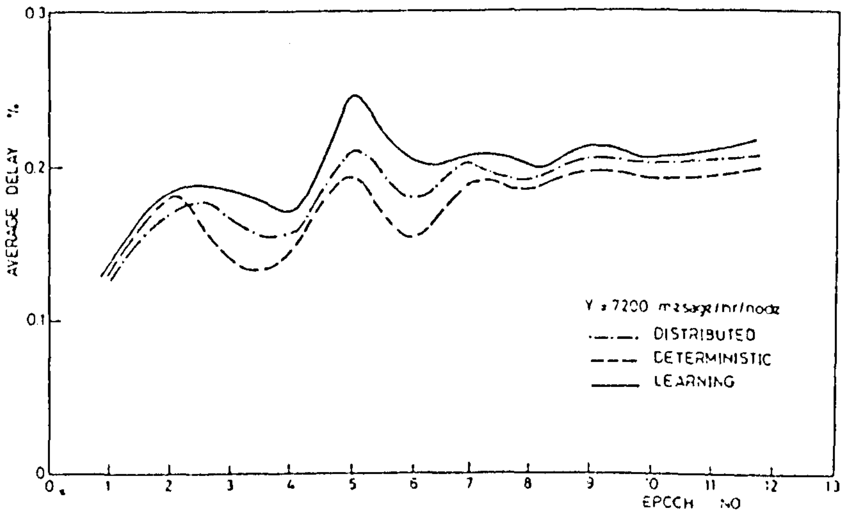

Fig. 10: Average Overall Delay.

2. Average Time Delay

The delay encountered by each packet is recorded and the average over respective time steps is calculated. These results are plotted versus respective load values for each algorithm in Figures (7), (8) and (9). It has been noted that the time delay varies significantly with the offered traffic. Figure (10) shows the time delay averaged over all the experiment time. It is clear that the deterministic algorithm gives the best average delay performance. However the relative differences between the three algorithms are small.

3. Response Time

The average delay transmission time, i.e. the network response time is related to the packet delay. This measure is provided by the simulation experiment.

6. CONCLUSION

Despite its initial oscillation, the learning automata can be considered to be the best of the three algorithms regarding its overall average, followed by the deterministic algorithm then the distributed one.

It is interesting to observe that the estimated 95% confidence intervals were found to be 3.323%, 3.12% and 2.95% of the sample mean for the retransmission probability, the time delay, and the response time respectively.

REFERENCES

1. **Ayad, N.M.A.:** "Performance Evaluation of Routing Techniquenor Packet-Switched Computer Networks", Ph.D. Thesis, Cairo University, 1984.

2. **Ayad, N.M.A., Mohammed, F.A., Madkour, M.A.** and **Metwally, M.S.:** "Performance Comparison of Quasi-static Routing Algorithms for Packet-Switched Computer Networks", Computer Communications Review, Fall, 1984.

3. **Gallager, R.G.:** "A Minimum Delay Routing Algorithm Using Distributedommunication", IEEE Trans. on Comm., Vol. Com-25, No. 1, Jan. 1977, pp. 73-85.

4. **Yum, T.R.:** "The Design and Analysis of a Semidynamic Deterministic Routing Rule", IEEE Trans. on Comm., Vol. Com-29, No. 4, April 1981, pp. 43-59.

5. **Elbaz, D.** and **Authi, G.:** "Distributed Algorithms for Optimal Routing in Packet-Switched Computer Networks: Shortest Path and Nonlinear Flow Control", First International Workshop on Methodologies and Applications of Complex System Theory (FIWMACST), Cairo, Nov. 1983, pp. 15-17.

6. **Pool, T.G.** and **Szymankiewicz, J.S.:** "Using Simulation to Solve Problems", McGraw-Hill (U.K.), 1977.