Fairness of a DTM Dual-bus with Large Inter-node Distances

Csaba Antal
Traffic Analysis and Network
Performance Laboratory
Ericsson Telecommunications Ltd.
e-mail:csaba.antal@eth.ericsson.se

Sándor Molnár
High Speed Networks Laboratory
Dept. of Telecommunications and Telematics
Technical University of Budapest
e-mail:molnar@ttt-atm.ttt.bme.hu

Abstract

Previous studies of Dynamic Synchronous Transfer Mode (DTM), which is based on multi-access dual fiber bus, showed that it could be a viable technology in the future. This paper analyses fairness of the protocol in the case of large inter-node distances, which is especially critical due to the long propagation time of control messages. The study shows that there is significant difference between the fairness of the examined algorithms. It is also shown that even the best algorithm is unfair in certain network configurations. The work is based on simulation, which was carried out by a self-developed simulator.

Keywords: fairness, circuit switching, dynamic synchronous transfer mode

1. Introduction

Dynamic Synchronous Transfer Mode (DTM) [5,10] is an attempt to build the next generation of local and metropolitan area networking technologies on fast circuit switching basis. It is a new broadband network architecture developed at the Royal Institute of Technology in Stockholm (KTH). The technology is in the focus of several Swedish companies, because it can be the competitor to technologies like FDDI or Asynchronous Transfer Mode (ATM) [11] in special application areas.

FDDI is one of the most popular backbone networking technologies in the metropolitan area. It is based on dual fiber ring. DTM, which is also based on dual fiber, can be an upgrade path of FDDI to higher speeds in the future.

ATM transfer capabilities define service classes, which are appropriate for several application types. However, ATM is not the most natural solution for applications with low delay and delay variance requirements due to its inherent cell (packet) switching characteristics. Applications with high bandwidth and low delay variance needs (e.g.: video on demand, video telephony) are more suited to the philosophy of circuit switched networks.

DTM uses dual bus architecture so as with DQDB networks. The main disadvantage of DQDB, which prevented its wide acceptance, was unfair operation in the case of high network loads [9]. This similarity motivates that a sophisticated fairness analysis is to be done for DTM as well. Fairness issues are especially critical if nodes operate at Gbps domain, if inter-node distances are high or if the network uses spatial bandwidth reuse (slot reuse). Providing equal share in throughput and equal access delay for nodes at the same time is another challenge.

DTM is designed for unidirectional medium with multiple access. The total medium capacity is shared by all connected nodes using time division multiplexing. Previous proposals and implementations [5,6] are based on dual-bus topology.

The total capacity of the bus is divided into cycles of 125 microseconds, which are further divided into 64 bit long slots. The sequence of slots at the same position in successive cycles is called DTM channel or simply slot.

There are two types of slots (and so DTM channels): data and static ones. Data slots are used for data transfer. For each DTM channel there is a token, which is assigned to one of the nodes. Both free and used data channels are assigned to nodes. Each channel has exactly one owner at a time. If a node has the right to use a channel (i.e. it has the token), then it has full control on it: it

can setup a connection on it, send data within the connection, release a connection using the channel, or give the channel ownership to another node.

At system start-up, data channels (tokens) are allocated by the nodes but they can be transferred dynamically during operation. *Nodes can ask others for free data channels if they do not have enough to serve a new request.* This procedure is called channel reallocation. To facilitate channel reallocation, nodes maintain status tables that contain the number of free channels at other nodes.

The other type of slot, called static slot, is used for broadcast control channels between the nodes. Nodes send control information in their static slots and listen to all the other static channels to receive control information.

The distributed channel reallocation algorithm of DTM (KTH algorithm) is described in [5,6]. In [3], we studied the fairness of its three variants. We varied the request order of nodes during channel reallocation. The three types of orders are:

- Closest First : KTH-CF algorithm [6]
  It operates according to the closest first rule. That is, the node that received a connection request when having too few free slots sends the first slot request to the closest node out of nodes with free slots according to the status table. If the sum of free slots at these two nodes is not enough for the pending request, the second closest node with free slots is also asked for slots. If further slots are needed the closest non-requested nodes are contacted.
- Logical Ring : KTH-LR algorithm
  Nodes are ordered into a logical ring [2]. The order of channel requests is based on the location in the logical ring instead of the bus. That is, closest node is the first ring neighbour, second closest is the second ring neighbour, etc.
- Random : KTH-RA algorithm [3]
  Nodes choose the next node randomly to ask for channels out of nodes with free slots according to the local status table.

The algorithms were examined in two extreme conditions for short bus-length in [3]:

Model 1 – status tables are *up-to-date* and nodes only try to get free channels from nodes with free channels

Model 2 – status tables are *useless* and nodes try to get free channels from others without any apriori knowledge

We showed in [3] that if model 1 (with up-to-date status tables) is appropriate, then the network is fair independently of request order. According to model 2, KTH-RA was the only fair algorithm.

Model 2 is a coarse model for inconsistent status tables. In this paper, we use an improved inter-mediate model. Basically, we use model 1; but we also assume that inter-node distances are significant, which means that status tables are partly out-dated due to the propagation delay of update messages that are broadcasted after each change in the number of free channels. When stressing the presence of status tables we denote the basic algorithm with KTH-S instead of KTH (e.g. KTH-S-CF instead of KTH-CF).

The purpose of the paper is to analyse the fairness of the DTM dual-bus when status tables become out-dated due to large propagation time of status-table update messages.

2. Methodology

The main assumptions during the simulation study are summarised below.

Traffic generators are based on WWW traffic sources. The model of WWW traffic is based on [7,8]. The inter-arrival time of connections is based on Weibull distribution. The holding time of a request is modelled by Pareto distribution.

We assume that in a given network configuration all traffic generators have the same traffic parameters, and the number of generators connected to the analysed nodes is the same. The *destination* of calls generated by a source, however, can be *different*. We used three network scenarios with different destination distribution:

1. In the "**external**" network configuration each node communicates with a dedicated node at the end of the dual-bus
2. In the "**client-server**" network configuration each node initiates bi-directional connections to the "server" node in the middle of the dual-bus.
3. Each node establishes connection with equal probability to any other node on the dual-bus in the case of the "**peer-to-peer**" model

The basis of fairness analysis is *average connection set-up time, average slot request time and call blocking probability.*

In this paper, we show the simulation study of a long dual-bus, where the inter-node distance is 10 km. The number of nodes is 100 in the first two network configurations and 25 in case of "peer-to-peer" traffic.

3. Simulation Results

3.1 External Network Scenario

Blocking probability of external network configuration is shown in Figure 1. The algorithms are displayed side by side: closest first, logical ring and random request orders (KTH-S-CF, KTH-S-LR, KTH-S-RA). The number of channel reallocation requests is limited for each connection. The channel allocation retry limit (retry limit) is configured to 5, 30 and 50.

If status tables were up-to-date, each 1-slot connection would be established or blocked with at most one slot request. As each node is allowed to ask any other node, there should be no difference between the blocking of different nodes. In this case, however, status tables are outdated. To illustrate it Figure 2 displays the average number of slot requests in *blocked calls*. (It should be below 1 in optimal case.)

The average is above 3 for each node. It shows that there are many inconsistencies in status tables. The other interesting effect is that this value is higher for outer nodes. It indicates that outer nodes have worse status tables, which can be explained with the so-called average distance.

d(i,k) - the distance of node *i* and node *k*

$D(i) = \sum_{k=0}^{99} d(i,k)$ - average distance of node *i*

D(i) decreases when going from the outer part of the bus to the middle node. *D(i)* is minimal for the middle node. The consistence of status tables depends on the delay of administration messages. The average delay of administration messages is proportional to the average distance of the given node, so outer nodes with higher average distance should have more out-dated status tables.

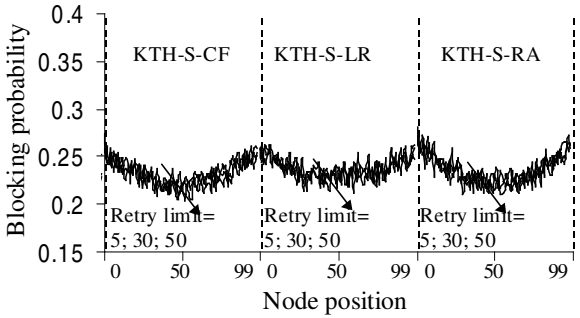

Figure 1 - Blocking probability, external model, one direction

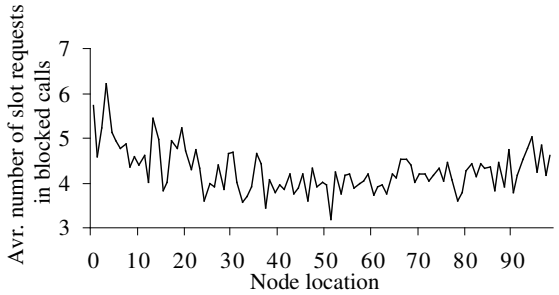

Figure 2 - Average number of slot requests in blocked calls, KTH-S-RA with retry limit of 30

As unfairness is due to the propagation delay of update messages, blocking probability does not depend on the request order (see Figure 1).

The next characteristic is the connection set-up time. It consists of two parts:

- round-trip delay of the set-up message
- delay coming from slot requests

The first part depends on the location of the other party of the connection. As the switching node is at the end of the bus in our scenario, this factor is proportional to the distance measured from the end of the bus. As customers usually respect differences caused by the physical distance of the other party of the connection, we substracted the delay of set-up message and its acknowledgement from the connection set-up time. The new variable is referred to as *average slot request time*. Average slot request time is shown in Figure 3 for each algorithm. Unfairness can be observed at each examined algorithm, but KTH-RA is the worst one.

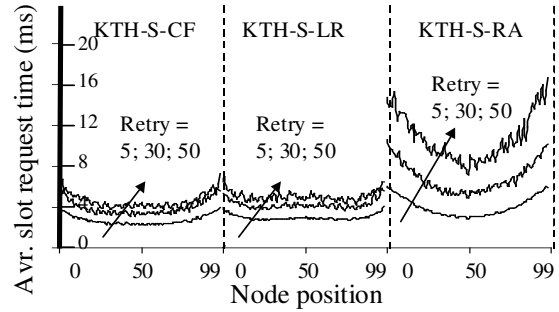

Figure 3 - Average slot request times, external model, one direction

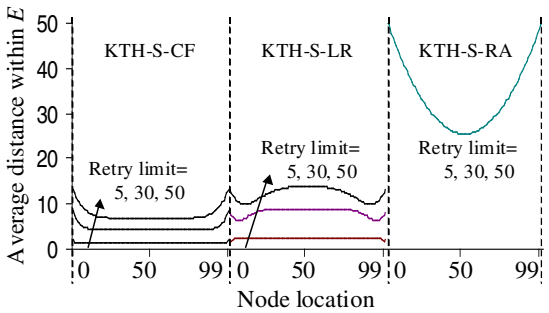

Figure 4 - Average distance within the effective area

We defined a new measure to explain the results:

$D_E(i) = \sum_{k \in E} d(i,k)$ - average distance of node *i* within its effective area *E*.

Effective area is the set of nodes from which node *i* is able to ask slots assuming that every node has free slots.

The effective area depends on the request order and the slot allocation retry limit. If retry limit is *r* at KTH-S-CF and KTH-S-LR algorithms there are only *r* nodes within the effective area. Every node is within the effective area of any node in the case of KTH-S-RA request order, according to the definition. Figure 4 shows the calculated $D_E(i)$ for each request order and for retry limit of 5, 30 and 50. *d(i,k)* is normalized so that *d(i,i+1)*=1, i.e. distance of neighbouring nodes is 1.

In the case of KTH-S-LR and KTH-S-RA algorithms average distance within the effective area and slot request time have the same characteristics. The results of KTH-S-CF algorithm can be explained by the common effect of the difference in the average distance within the effective area and the uneven free slot distribution, which was already shown at short bus [3,2,1]. Because of the shorter average distance of middle nodes, their status tables are more accurate than those of outer nodes. Better status tables yields to shorter slot request times.

Based on Figure 3 it can be said that the best algorithm is KTH-LR in this case.

3.2 Client-Server Network Scenario

Simulation results of the client-server network configuration are displayed in Figure 5 and Figure 6.

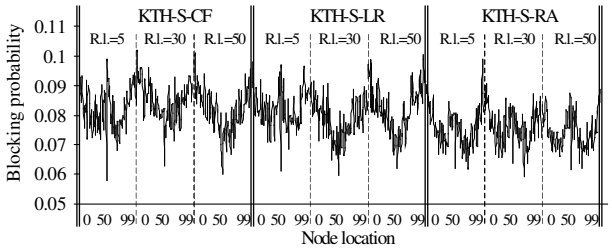

Figure 5 - Blocking probability, client-server model, both directions

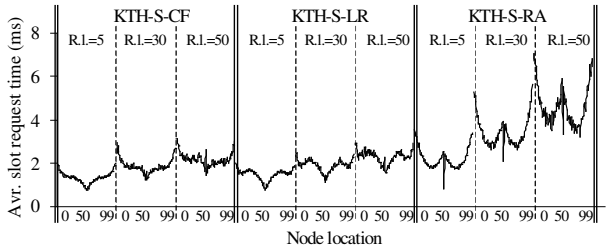

Figure 6 - Average slot request, client-server model, both directions

Both blocking probabilities and slot request times are unfair. It can be observed that blocking probability of the server node (node 50) is smaller than that of the others. It is due to the cache effect [3], i.e. very active nodes have better performance because they can reuse the tokens released by previous local connections.

The resulted curves are due to the following factors:

- Effect of the active node (server) – see [3]
- Effect of idle nodes (upstream nodes of the server) – see [3]
- Effect of propagation times – see "external" scenario

Due to the higher average distance KTH-RA has worse performance than the other algorithms have. As a conclusion it can be stated that there is no winner algorithm in this case: each algorithm is unfair. Difference is only in the extent of unfairness.

3.3 Peer-to-peer Network Scenario

Finally, peer-to-peer load profile is simulated. Figure 7 and Figure 8 show simulation results.

The performance of algorithms in the peer-to-peer model is almost the same.

In average set-up time, the average distance - *D(i)* - determines the shape of the curve, as set-up messages are sent to all other nodes with equal probability. It is the same for each algorithm, so no difference can be observed.

The unfairness of blocking probability is also related to *D(i)* because *D(i)* is proportional to the average propagation delay of status table update messages. It does not depend on the algorithm, therefore blocking probability curves of different algorithms are similar.

The extent of unfairness is the smallest in this network configuration.

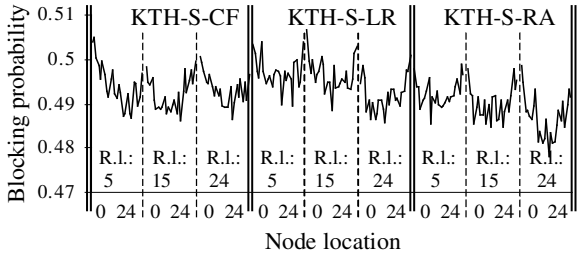

Figure 7 - Blocking probability, peer-to-peer model, both directions

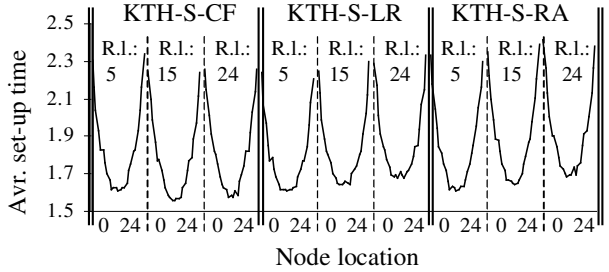

Figure 8 - Average set-up time, peer-to-peer model, both directions

4 Quantitative Comparison

A quantitative analysis is also performed based on the simulation results. We use Jain's fairness index [4] for comparison.

If x_i is the observed characteristics of node i and there are N nodes in the network, then the **Jain's fairness index** can be calculated as:

$$f_{Jain} = \frac{\left(\sum_{i=0}^{N-1} x_i\right)^2}{N \cdot \sum_{i=0}^{N-1} (x_i)^2} \qquad (1)$$

Table 1 shows the indices for each simulation run. The bottom line shows the overall rating of algorithms, which is obtained as the average of the 18 simulation runs.

We defined three categories:

- fair: 0.99 and above
- unfair: 0.97-0.99
- very unfair: 0.97 and below

Note that these are artificial categories. The required level of fairness depends on the application area.

Algorithm		KTH-CF			KTH-LR			KTH-RA		
Retry limit		5	30	50	5	30	50	5	30	50
External	SRT	*0.974*	*0.971*	*0.975*	**0.992**	**0.99**	**0.991**	*0.951*	*0.956*	*0.958*
	BP	**0.996**	**0.998**	**0.997**	**0.998**	**0.998**	**0.998**	**0.997**	**0.996**	**0.996**
Client-server	SRT	*0.965*	*0.973*	*0.981*	*0.976*	*0.982*	*0.984*	*0.96*	*0.962*	*0.964*
	BP	**0.993**	**0.995**	**0.992**	**0.994**	**0.993**	**0.991**	**0.993**	**0.994**	**0.994**
Peer to peer	CST	*0.986*	*0.986*	*0.986*	*0.989*	*0.989*	*0.989*	*0.986*	*0.986*	*0.986*
	BP	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**
Overall		*unfair: 0.987*			**fair: 0.991**			*unfair: 0.981*		
Notation	**fair**	*unfair*	*very unfair*		SRT: Slot request time; BP: Blocking probability; CST: Connection set-up time					

Table 1 – Quantitative analysis: Jain's fairness index

According to the categories defined above, KTH-LR algorithm is "fair"; KTH-CF and KTH-RA algorithms are "unfair".

The main difference between algorithms was observed at "external" and "client-server" network configurations. "Peer-to-peer" scenario was the smallest challenge for algorithms. In "peer-to-peer" scenario, all variants of KTH algorithm had the same performance independently of the request order.

At "external" configuration even the slot request time of KTH-LR was "fair", while that of KTH-RA was "very unfair". At "client-server" configuration average slot request time of KTH-RA was "very unfair", while that of KTH-CF and KTH-LR was only "unfair". Therefore the final preference order of algorithm variants in the case of long inter-node distances is:

1. KTH-LR; 2. KTH-CF; 3. KTH-RA

5 Conclusions

The paper presented a simulation study to analyse the fairness of a DTM dual-bus in case of large inter-node distances are high. Three variants of the basic KTH algorithm were evaluated. Simulations showed that there is significant difference between algorithms with different request orders. The algorithm based on "closest first along a logical ring" (KTH-LR) was the least unfair. According to our fairness categories, KTH-LR algorithm is "fair".

KTH-RA algorithm, which was the best algorithm in the case of short bus-length [3], is the worst choice when the bus is long.

We showed that the main reasons for unfairness are:

- Propagation delay of status table update messages, i.e. difference in consistency of status tables
- Different average distance from other nodes, i.e. difference in the duration of slot requests
- Very active node, i.e. it has better performance due to the "cache effect"

The paper considered only one of the main factors usually resulting unfair operation (large inter-node distance). The fairness study of slot reuse algorithms [10] is one of the directions of our future research.

References

0. Cs. Antal, "Performance Analysis of the DTM Fast Circuit Switched Networking Technology", *Ph.D. dissertation*, Technical University of Budapest, Hungary, 1999
1. Cs. Antal, J. Molnár, S. Molnár, G. Szabó, "Performance Study of Distributed Channel Reallocation Techniques for a Fast Circuit Switched Network", *Computer Communications* Vol. 21 Num. 17, November 1998, Special Issue on the Stochastic Analysis and Optimal Management of Communication Systems, pp. 1597-1609
2. Cs. Antal, J. Molnár, S. Molnár, "Fairness Issues in a New Dual-Bus Networking Technology*"; Proc. 7th IFIP/ICCC Conference on Information Networks*, Aveiro, Portugal, June 1998, pp. 83-98
3. R. Jain, *The Art of computer systems performance analysis:techniques for experimental design, measurement, simulation and modelling*, John Wiley and Sons Inc., New york, 1991
4. C. Bohm, M. Hidell, P. Landgren, L. Ramfelt, P. Sjödin, "Fast Circuit Switching for the Next Generation of High Performance Networks", *IEEE J. on Selected Areas of Communications*, Vol. 14., No. 2, February 1996
5. C. Bohm, P. Lindgren, L. Ramfelt P. Sjödin, "Resource Reservation in DTM", *1st IEEE Symposium on Global Data Networking*, Cairo, Dec. 1993
6. M. E. Crovella, A. Bestavros, "Self-Similarity in the World Wide Web Traffic: Evidence and Possible Causes", *Proc. SIGMETRICS '96*, Philadelphia, May 23-26, 1996
7. S. Deng, "Empirical model of WWW document arrivals at access link", *IEEE International Conference on Communications*, p. 1797-802 vol.3, ICC'96, 23-27 June 1996, Dallas, TX, USA
8. Kumar, L.N.; Douligeris, C. "Demand and service matching at heavy loads: a dynamic bandwidth control mechanism for DQDB MANs", *IEEE Transactions on Communications*, Vol: 44 Iss: 11 p. 1485-95
9. P. Lindgren, "A Multi-Channel Network Architecture Based on Fast Circuit Switching", *Ph.D. thesis*, ISRN KTH/IT/R-96/08-SE, Stockholm, May 1996
10. D. E. McDysan, D. L. Spohn, *ATM, Theory and Application*, McGraw-Hill, 1994.
11. V. Paxson and S. Floyd, "Wide Area Traffic: The Failure of Poisson Modeling", *IEEE/ACM Transaction on Networking*, vol. 3, no. 3, pp. 226-244, 1995.