APPROXIMATE ANALYSIS OF GENERALIZED BANDWIDTH BALANCING MECHANISM FOR DQDB NETWORK

Sayed A. Nouh and Ahmed R. Khalifa
Computer and Systems Department, Faculty of Engineering,
Al Azhar University, Cairo, Egypt.

ABSTRACT

The Distributed Queue Dual Bus (DQDB) is a protocol specified by the IEEE 802.6 Standard Committee for Metropolitan Area Networks for interconnecting hosts, local area networks and workstations. A potential problem with DQDB is the achievement of fair sharing of bandwidth in terms of the location of the access node for "Queued Arbitrated" (QA) slots (e.g., for data packet and ATM communications). For the sake of compensating this unfairness, the IEEE 802.6 standard recommends to utilize a Bandwidth Balancing (BWB) mechanism with proper value of the system parameter named BWB-MOD. We consider the original DQDB protocol with the bandwidth balancing mechanism operating with asynchronous traffic only and having one level of priority. For the case of two active nodes in the network, an approximate analytic model is developed which completely specifies the nodal throughputs in the steady state. Simulation is used when the number of active stations is larger than two. Our results provide a deep insight into the equilibrium access of data channel by all active stations. They also show that the station throughputs are independent on the state of the network when the overload occurs.

Keywords: Bandwidth Balancing Mechanism, Distributed Queue Dual Bus.

1 INTRODUCTION

The Distributed Queue Dual Bus (DQDB), is a protocol specified by IEEE 802.6 Standard Committee for Metropolitan Area Network for interconnecting hosts, local area networks and workstations. The topology of DQDB is based on two unidirectional buses supporting communications in opposite direction. Nodes can read and write on both buses and communicate to each other by selecting the proper bus. DQDB is a slotted system and has the potential to utilize the whole channel bandwidth independent of the network span, number of nodes, and the transmission rate. The performance obtained with the DQDB access protocol was shown to approach the perfect schedular performance under favorable conditions [1]. However, it has been reported in various publications that large propagation delay to transmission time ratios (high bit rates, large networks) can cause a certain degree of unfairness in access delay which depends on the stations position with respect to the head of bus station in the sense that stations located closer to the head of a bus may experience higher throughput and/or lower message delays and bandwidth can be shared unevenly during long transmission such as file transfer.

More importantly, this unfairness behavior mainly stems from the fact that many slots which carry information on network conditions are in transit when networks span long distances. In particular, distance between the nodes and the initial conditions when nodes start transmitting long messages were reported [2-5], to be the determinants of the throughputs of active nodes will receive under heavy load.

To solve this problem, an enhancement to the DQDB protocol, the Bandwidth Balancing Mechanism [2], was proposed and adopted by the IEEE 802.6 working group. This scheme limits the maximum bandwidth available to each node, in order to leave a portion of bandwidth unused. This unused bandwidth permits the congestion to relax and allows equal sharing of the available bandwidth to be attained eventually.

This paper deals with the implementation of BWB mechanism using Priority Scheduling method. For the case of network having two active nodes, an analytical model based on the Priority Scheduling method is developed. This model completely specifies the nodal throughputs.

In section 2, the DQDB access protocol for asynchronous traffic is reviewed. Section 3 presents the concept of the BWB mechanism. The implementation of BWB mechanism using the Priority Scheduling method is presented in section 4, and we develop analytic model for the case of two active (overloaded) nodes in DQDB network which completely specifies the nodal throughputs. In section 5, the simulation results are obtained for various possible scenarios in the case of two and three active nodes in the network. Section 6 concludes the paper.

2. DQDB MEDIUM ACCESS CONTROL PROTOCOL

DQDB [6] is based on two unidirectional buses which can carry traffic in opposite directions. Each station can read and write on both buses. Information is carried on slots of fixed duration generated by the head of each bus. The duration of a slot, is equal to the size of a data segment of 44 bytes, plus the header of 9 bytes. Each slot contains in its header a busy bit and three request bits, one for each level of priority supported by the DQDB medium access protocol. We are interested in operation at only one level of priority, and hence, will not discuss the behavior of the protocol under multipriority traffic.

The busy bit of a slot indicates whether the slot is empty. The request bit is used for making reservations for data segments in the opposite direction. Since the operation in both directions is identical, we will consider data transmission in one direction only. One of the buses will be called the data channel (bus A) and the other will be called the reservation or request channel (bus B), see Figure (1). Flow of data slots is from upstream nodes to downstream nodes in the data channel. Requests are placed in the reservation channel in the opposite (upstream) direction.

Each station has two counters called the request (REQ) counter and the countdown (CD) counter. When a user is idle, only the REQ counter is operational. While in this state, the REQ counter is incremented by one for each request detected in the reservation channel and decremented by one for each empty slot detected in the data channel. When it becomes active, the content of the REQ counter is transferred to the CD counter and the REQ counter is reset to zero. Although an arriving packet may contain multiple segments, only one segment at a time is admitted per node in the network. For each segment registered, the user sends out a request in the reservation channel by setting to 1 the first free request bit received. Independent from this, the CD counter is decremented by 1 for each empty slot in the data channel. After the CD counter reaches zero, the segment is transmitted in the first free slot observed. During the time the CD counter is active, the REQ counter is incremented for each request slot received from the downstream users. It must be noted that, since access to both buses is independent, it may happen that a segment is transmitted before the corresponding request could be sent out. In such a case, a new segment is registered immediately, if available, and the REQ counter is transferred to the CD counter, a third counter counts the outstanding requests. While it is allowed to have more than one outstanding request, data segments are still not admitted into the CD counter before the previous segment is transmitted.

Several analysis of the effect of propagation delay on fairness in DQDB [7-9], demonstrated that in some cases under overload condition the throughput of the different stations is not equal, with unfair advantage to the station that starts overloading the network. In [10-13], protocol modifications are considered to overcome its inherent unfairness.

3. BANDWIDTH BALANCING MECHANISM

The basic DQDB protocol was extended with Bandwidth Balancing (BWB) mechanism to improve fairness [2]. This mechanism has been chosen by the IEEE 802.6 Standard Committee to incorporate into the DQDB protocol to remedy the unfairness problem. The mechanism intentionally wastes a small amount of bus bandwidth, called the idle capacity, but fairly divides the remaining bandwidth among the heavily loaded nodes, that is, all the lightly loaded nodes take exactly their needed bandwidth, and all the overloaded nodes take their bandwidth equally. So, the throughput achieved by each overloaded node is the same, called the control rate (R).

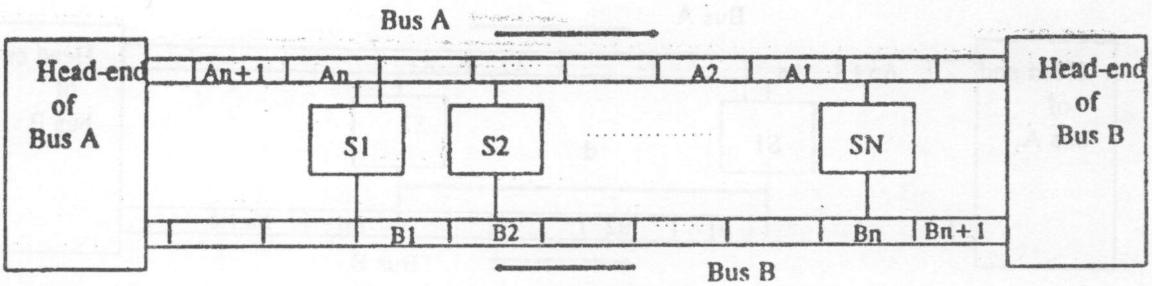

Figure 1. DQDB network.

More specifically, the control rate R is proportional to the idle bus capacity, with a proportionality constant ß (called BWB- Mode) > 1, [2]. Assume that there are N active nodes in the network which may require to send segments onto the same data bus. M of them are heavily loaded, and (N- M) are lightly loaded, Then:

$$R = \beta [1 - (S + M R)] \quad (1)$$

Where S is the total utilization due to (N-M) lightly loaded nodes. Let $\Gamma = R$ denotes the carried load of any heavy loaded node, then:

$$\Gamma_n = \alpha [1 - \{S + (M - 1) R \}] \quad (2)$$

where $\alpha = \beta /(1 + \beta)$

Hence, equation 2 can be implemented by restricting each node from taking more than a fraction α of the spare capacity left over by the other nodes. In general, the control rate and the bandwidth wastage (idle bus capacity) are as follows:

$$R = \beta(1 - S)/(1 + \beta M)$$

$$R = \alpha(1 - S)/[1 + (M-1)\alpha] \quad (3)$$

$$\text{Bandwidth Wastage} = R/\beta = (1 - S)/(1 + \beta M)$$

$$= (1 - \alpha)(1 - S)/[1 + (M-1) \alpha] \quad (4)$$

4. IMPLEMENTATION OF BWB MECHANISM USING PRIORITY SCHEDULING METHOD

The BWB mechanism can be implemented by what is called the Priority Scheduling method, which is simple to implement and easy to analyze. In this implementation, every node has a request counter (REQ-CNT) and a bandwidth balancing counter (BWB-CNT) but no countdown counter (CD-CNT). The REQ-CNT counts unserved requests from downstream nodes as in DQDB. However, these requests are given priority over local data-segments, so that a node only has the opportunity to transmit, when there are no requests and the data bus slot is idle.

The operation of the BWB-CNT is explained as follows:

(i) The REQ-CNT is incremented by one for each request detected in the reservation (request) channel and decremented by one for each empty slot detected in the data channel. As long as the REQ-CNT not equal to zero, a station cannot transmit its own data segment.

(ii) When the REQ-CNT reaches zero and the data bus slot is empty, a station can transmit its own data segment and the BWB-CNT is incremented by one as long as its value is less than the BWB-Mode (ß).

(iii) When BWB-CNT value reaches ß, the counter is reset to zero and the REQ-CNT is incremented by 1. This makes it happen that, the access function to the bus is forced to skip transmitting the segment in the next available empty slot.

Performance Analysis of Two Active Nodes

In this section, we consider simultaneous file transfers by two nodes spaced D slots apart on the bus. As usual, we'll call the upstream node 1 and the other node 2. We show that the BWB mechanism converges to the steady state throughput given by $\alpha/(1+\alpha)$, independent of the initial conditions created by the previous history of the system. The analysis is based on Priority Scheduling method.

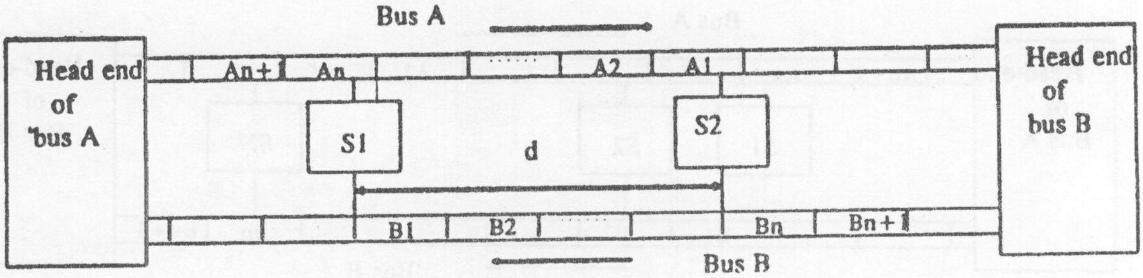

Figure 2. The two-station case.

As reported in [2], we start our analysis at time T_0 at which the following conditions are met:

- The REQ-CNTs of both active nodes have zero value.
- The data channel upstream of node 1 and the request channel downstream of node 2 are empty.
- There is a fraction f_B of the D busy bits in transit between nodes 1 and 2 on the data bus and a fraction f_R of the D request bits in transit between the nodes are set to 1, see Figure (2).

According to the BWB mechanism, each node transmits in a fraction α of the empty slots available to it for its own data transmission. Consequently, in the first D slot times, node 1 will transmit in $\alpha(1 - f_R)$D slots and node 2 will transmit in $\alpha(1 - f_B)$D slots. In the next D slot times, node 1 transmits in $\alpha[1 - \alpha(1 - f_B)]$D slots, while node 2 transmits in $\alpha[1 - \alpha(1 - f_R)]$D slots. The throughput of a node over half a round-trip time depends on the other node's throughput in the previous half a round-trip time.

Let $\Gamma_1(n)$ and $\Gamma_2(n)$ be the fraction of the bandwidth acquired by nodes 1 and 2, respectively, during slots nD to (n + 1)D, where n = 0, 1, 2, ... The analysis for the two nodes are similar and we concentrate on the bandwidth acquired by node 1. Consequently,

$$\Gamma_1(1) = \alpha(1 - f_R) = \alpha - \alpha f_R$$

$$\Gamma_1(2) = \alpha[1 - \alpha(1 - f_B)] = \alpha - \alpha^2 + \alpha^2 f_B$$

$$\Gamma_1(3) = \alpha\{1 - \alpha[1 - \alpha(1 - f_R)]\} = \alpha - \alpha^2 + \alpha^3 - \alpha^3 f_R$$

$$\Gamma_1(4) = \alpha\{1 - \alpha\{1 - \alpha\,[1 - \alpha(1 - f_B)]\}\}$$

$$= \alpha - \alpha^2 + \alpha^3 - \alpha^4 + \alpha^4 f_B$$

By induction

$$\Gamma_1(n) = \alpha - \alpha^2 + \alpha^3 \;........\; + \alpha^n - \alpha^n f_R \quad \text{where n odd} \qquad (5)$$

$$\Gamma_1(n)=\alpha-\alpha^2+ \alpha^3 \;...\; - \alpha^n + \alpha^n f_B \quad \text{where n even} \qquad (6)$$

Eqns 5 and 6 can be written in the form

$$\Gamma_1(n) = A\alpha^n - \sum_{k=1}^{n} (-\alpha)^K \qquad (7)$$

Where

$$A= \begin{cases} -f_R & \text{for n odd} \\ f_B & \text{for n even} \end{cases}$$

Similarly, the throughput of node 2 over half round-trip times is given by:

$$\Gamma_2(n) = A\alpha^n - \sum_{k=1}^{n} (\alpha)^K \qquad (8)$$

Where

$$A= \begin{cases} -f_B & \text{for n odd} \\ f_R & \text{for n even} \end{cases}$$

Let us make a few remarks on eqns (7) and (8). Note that, in the steady state, the nodal throughputs are each $\alpha/(1-\alpha)$ and the amount of system bandwidth wasted is $(1 - \alpha)/(1 + \alpha)$, in accord with eqns. (3) and (4). Note, moreover, that the steady state nodal throughputs are independent of the initial conditions f_B and f_R.

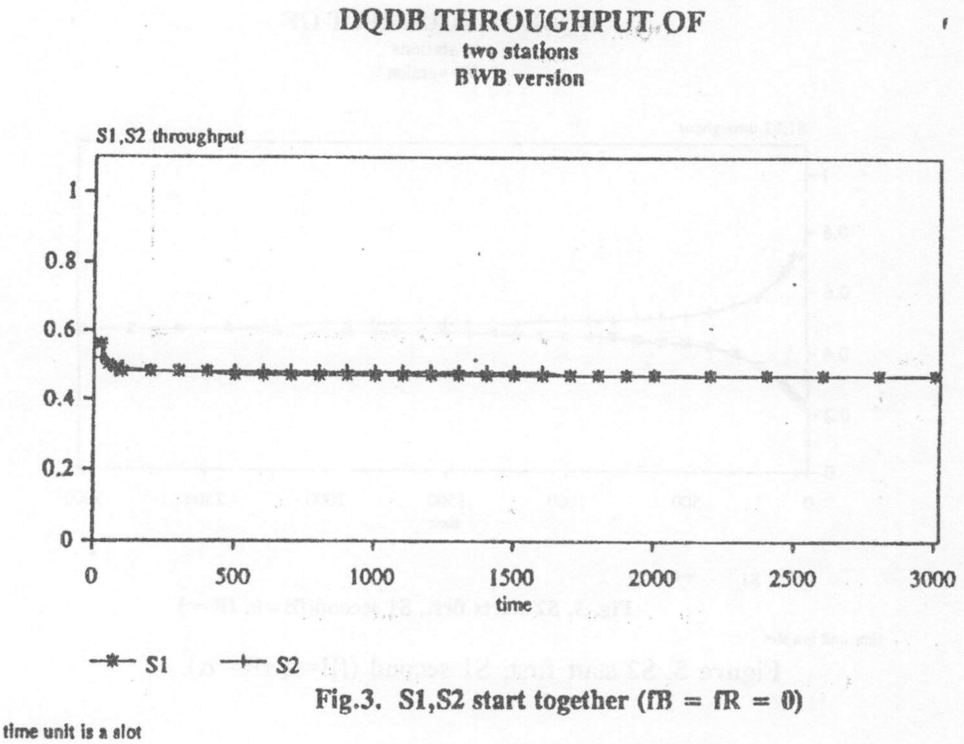

Figure 3. S1, S2 start together (fB=fR=0).

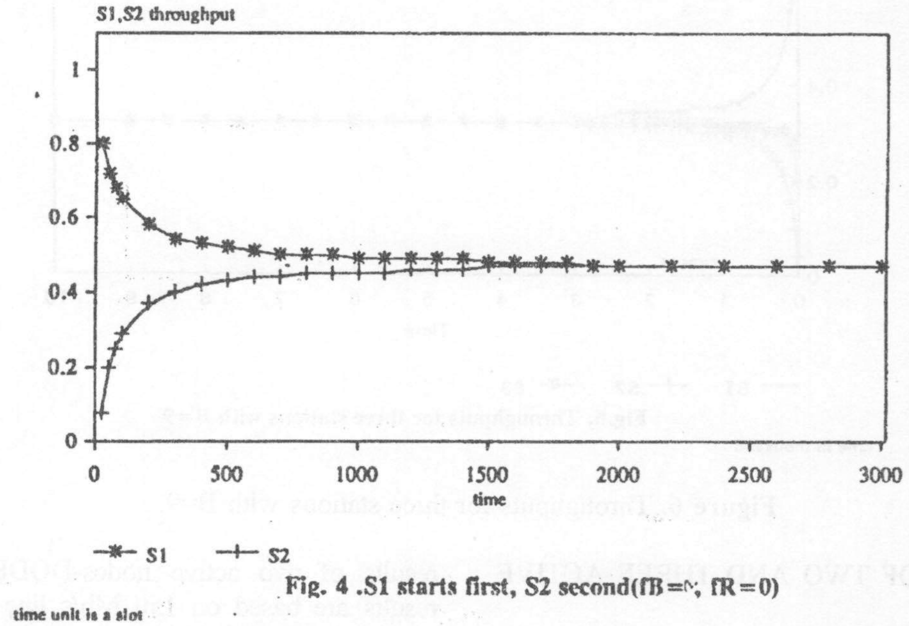

Figure 4. S1 start first, S2 second (fB = α, fR=0).

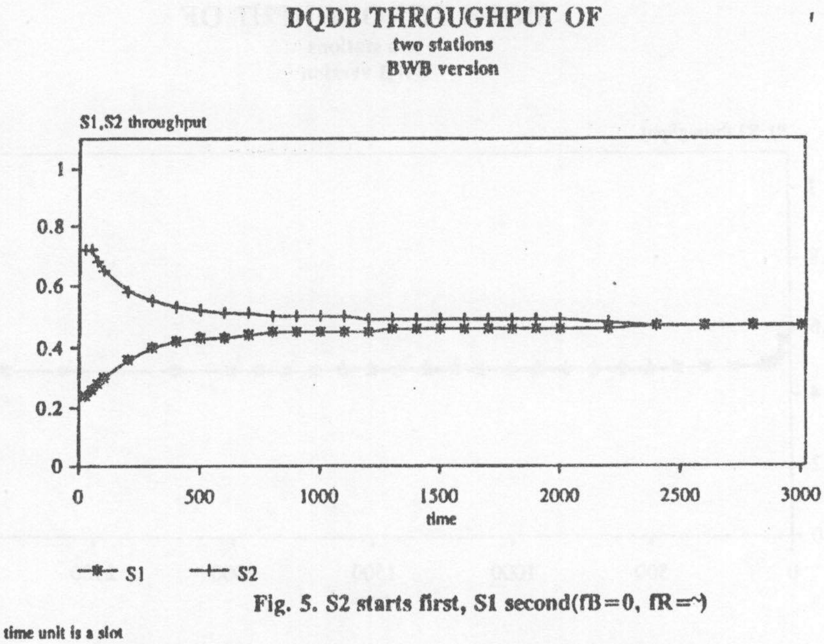

Figure 5. S2 start first, S1 second (fB=0, fR= α).

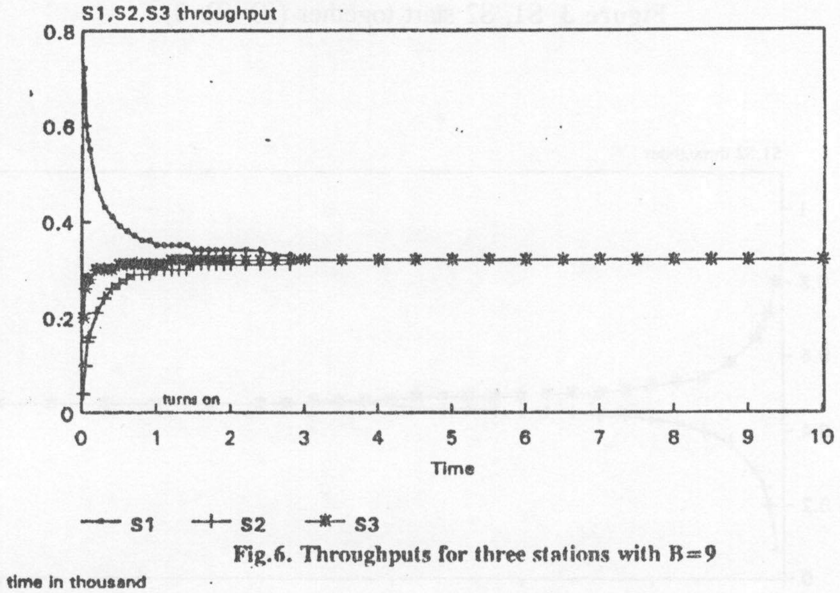

Figure 6. Throughputs for three stations with B=9.

5. SIMULATION OF TWO AND THREE ACTIVE NODES

The approximate throughput expressions, given in section 4, are found to match the simulation results reasonably well. Figures (3-5) represent the simulation results of two active nodes-DQDB network. These results are based on 150 Mb/s line capacity for each bus, slot size is equal to 53 bytes, propagation speed is 200 000 Km/s, the distance between the two nodes is 4 Km, and the BWB-Mode(ß) =3, i.e. $\alpha = 0.75$.

Figure (3) represents the case in which both nodes turn on at the same time: $f_B = f_R = 0$, where Figure (4) represents the case in which the upstream node (node 1), turns on at least half a round-trip time before the downstream node (node 2): $f_B = \alpha$ and $f_R = 0$, and Figure (5) represents the opposite case in which $f_R = \alpha$ and f_B=0. We note that, the steady state throughput of nodes 1 and 2, in the previous three cases is the same, equal to 0.43 and is independent of the initial conditions f_R and f_B. These results are consentaneous to that obtained from eqns. (3) and (4), where S=0, M=2 and ß=3.

Figure (6) depicts simultaneous file transfers by three nodes, with 5 slots (= 4 Km) between successive nodes and ß=9. The plot shows the nodal throughput measured over 10 000 slots. Again, the steady state throughput of nodes 1, 2 and 3 is the same and equal to 0.32, in accord with eqns. (3) and (4), where S=0, M=3 and ß=9.

6. CONCLUSION

In this paper, we concentrated on heavy load conditions in a DQDB network, where a simple implementation of BWB mechanism (the Priority Scheduling method) is introduced. An analytic model, based on the Priority Scheduling method, for two heavy load nodes is developed. This model provides a complete analysis of the DQDB protocol behavior, incorporated with the BWB mechanism under overload conditions. An interesting feature of Priority Scheduling method, is that nodes whose offered load is less than the control rate are not flow-controlled. The analytical model is verified through the simulation results of two and three active nodes. The results show that the node throughputs are independent on the state of the network when overload occurs, but they only depend on the number of active nodes and BWB- Mode (ß).

REFERENCES

[1] R.L. Newman and J.L. Hullet," Distributed Queueing: A Fast and Efficient Packet Access Protocol for QPSX", Proc. 8th ICCC, Munich1986.

[2] E.L. Hahne, A.K. Choudhury and Maxemchuk, "Improving the fairness of Distributed Queue Dual Bus Networks", Proc. IEEE INFOCOM' 90, San Francisco, June 1990, pp. 313-320.

[3] J.W. Wong, "Throughput of DQDB Networks Under Heavy Load ", Proc. EFOC/LAN- 89, Amsterdam, The Netherlands, June 1989, pp. 146-151.

[4] M. Zukerman and P.G. Potter," The Effect of Eliminating the Standby State on DQDB Performance Under Overload ", International Journal of Digital and Analog Cabled Systems, vol.2, 1989, pp.179-186.

[5] H.R. Van As, J.W. Wong and P. Zafiropulo," Fairness, Priority and Predictability of the DQDB MAC Protocol Under Heavy Load", Proc. Internatl. Zurich Seminar on Digital Comm., Zurich, Switzerland, Mar. 1990, pp. 410-417.

[6] IEEE 802.6 Committee, IEEE 802.6 Standard: Distributed Queue Dual Bus(DQDB) Subnetwork of a Metropolitan Area Network (MAN), Inst. of Electrical and Electronics Engineers, New York, Dec. 1990.

[7] M. Zukerman and P. Potter,"The DQDB Protocol and its Performance Under Over-load Traffic Conditions", Computer Network and ISDN, vol. 20 (1990), No. 1-5, Dec. 1990, pp. 261- 270.

[8] V. Phung and R. Breault," On the Unpredictable Behavior of DQDB", Computer Network and ISDN, Vol 24 (1992), pp. 145-152.

[9] M. Conti, E.Gregori and L. Lenzini," A Methodological Approach to an Extensive Analysis of DQDB Performance and Fairness", IEEE Journal on Selected Areas in Communications, vol. 9, No. 1, Jan. 1991.

[10] Flaminio et al, " FQDB: A Fair Multisegment MAC Protocol for Dual Bus Networks ", IEEE Journal on Selected Area in Communications, Vol. 11, No. 8, Oct. 1993.

[11] J. Flipiak," Real Time Network Management ", North-Holland, Amsterdam, Tokyo, New York, 1991.

[12] S.Y. Cheung, "Controlled Request DQDB: Achieving Fairness and Maximum Throughput in the DQDB Network", Proc. IEEE INFOCOM' 92, Florence, Italy, May 1992.

[13] L.N. Kumar and A.D. Bovopoulos, "An Access Protection Solution for Heavy Load Unfairness in DQDB ", Proc. IEEE INFOCOM' 92, Florence, Italy, May 1992.