An Effective Scheme for Pre-Emptive Priorities in Dual Bus Metropolitan Area Networks *

Jörg Liebeherr * *Ian F. Akyildiz* ** *Asser N. Tantawi* ***

* Computer Science Division
University of California - Berkeley
Berkeley, CA 94720
U. S. A.

** College of Computing
Georgia Institute of Technology
Atlanta, GA 30332
U. S. A.

*** IBM Research Division
Thomas J. Watson Research Center
Yorktown Heights, NY 10598
U. S. A

Abstract

The IEEE 802.6 standard for Metropolitan Area Networks does not provide multiple priority traffic for connectionless data services. A priority mechanism that was considered for the standard showed to be not effective. As of now, there exists no protocol for multiple access dual bus networks that is able to implement pre-emptive priorities and, at the same time, can satisfy minimal fairness requirements for transmissions at the highest priority level. In this study, a protocol with strictly pre-emptive priorities, i.e., a protocol that does not admit low priority traffic if the load from high priority traffic exceeds the capacity of the transmission channel, is presented. The protocol is derived from a unique bandwidth allocation scheme with a full utilization of the bus capacity, with a fair distribution of bandwidth respective to traffic from a particular priority level and with pre-emptive priorities. The performance of the presented protocol is compared to a priority mechanism that is based on the bandwidth balancing mechanism. It is shown that adopting the new protocol results in shorter access delays for high priority transmissions.

1 Introduction

In July 1991, the Distributed Queue Dual Bus (DQDB) protocol was released as the IEEE 802.6 standard for Metropolitan Area Networks [8]. The standard left the protocol without an effective mechanism to support multiple priority traffic. Even though IEEE 802.6 supports the assignment of priorities, all connectionless data traffic must be sent at the lowest priority level ([8], p. 46). However, it is widely acknowledged that the support of multiple priority levels is needed to provide a variable quality of service to the stations of the network. Support of high priority traffic is especially needed for network control and management.

It is agreed upon that a satisfactory media access scheme for dual bus networks with multiple priority traffic should satisfy the following requirements [5]:

1. The bandwidth allocated to high priority traffic is independent from low priority traffic.
2. Within any priority level, the maximum bandwidth that can be allocated is equal for all stations.
3. Bandwidth is never wasted.

As of now, a priority mechanism is missing that satisfies all three requirements.

In this study we present a protocol for dual bus networks with multiple levels of priorities that satisfies all requirements listed above. We present a formal characterization of multiple-priority access schemes in dual bus networks. This allows us to show the deficiencies of existing priority mechanisms. We present the unique solution to a bandwidth allocation scheme with a pre-emptive priority mechanism that does not waste bandwidth and satisfies fairness conditions for transmissions within each priority level. We develop a new protocol that is based on this unique solution. The new protocol uses results from [11] where we presented a uni-priority access protocol for dual bus networks that does not waste bandwidth and guarantees a fair distribution of bandwidth to the stations.

The paper is organized as follows. In section 2 we review the priority mechanism of the DQDB protocol. We discuss recent proposals that attempt to improve the priority mechanism of the DQDB protocol. In section 3 we categorize bandwidth allocation schemes for dual bus networks with multiple priorities, and derive a bandwidth allocation scheme that agrees with above mentioned requirements 1−3. In section 4 we present a new protocol that implements the concept of a so-called *strongly fair* and *waste-free* bandwidth allocation with pre-emptive priorities. We compare the performance of our protocol with an implementation of a priority mechanism that satisfies above listed requirements 2 and 3. We conclude our results in section 5.

COMM'92-8/92/MD,USA
© 1992 ACM 0-89791-526-7/92/0008/0161...$1.50

2 Media Access Protocols for DQDB Networks with Multiple Priorities

A *DQDB* network consists of two unidirectional buses with data flow in opposite directions. One bus is denoted by bus A and the other by bus B as shown in Figure 1. A slot generator at the head of each bus emits empty fixed sized slots at a constant rate. Each station is connected to both buses. A station transmits data by filling in an empty slot on a particular bus. Note that due to the topology of the dual bus each station has to make a routing decision whether to use bus A or bus B for transmission dependent on the physical location of the destination station. Since the architecture of a dual bus network is symmetric we will focus on data transfer on bus A.

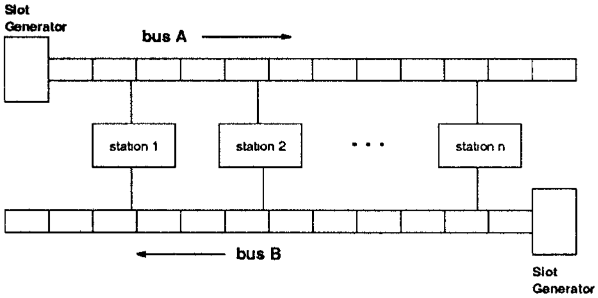

Figure 1: *DQDB* Network.

In subsection 2.1 we describe the media access protocol of the DQDB network with multiple priority levels of the IEEE 802.6 standard [8]. In subsection 2.2, we discuss proposals from the literature that attempt to enhance the priority mechanism in DQDB networks.

2.1 Media Access in a DQDB Network with Multiple Levels of Priorities

The *DQDB* protocol prevents the stations close to the head of a bus from acquiring all empty slots by implementing a reservation scheme. A station having a segment ready for transmission on bus A notifies the stations closer to the the head of bus A by sending a reservation request on bus B.

In a DQDB network with P priority levels, each slot contains $(P+1)$ access fields: a busy bit and P request bits, one request bit for each priority level. A slot with the busy bit set indicates that the slot contains data. The request bit of priority p set indicates a reservation request at priority level p $(1 \leq p \leq P)$. If a station writes data into an empty slot it sets the busy bit. A reservation request of priority p is submitted by setting the priority-p request bit.

For each priority level, a station keeps a queue of untransmitted segments. Only the segment at the head of a queue is allowed to submit a reservation request at the particular priority level. At each priority level, a station determines its turn to transmit a segment of priority p with two counters, the request counter (RQ_p) and the countdown counter (CD_p) $(1 \leq p \leq P)$.

If a station does not have segments of priority level p queued for transmission, it increments its RQ_p counter for each passing slot on bus B having the request bit set at equal or higher priority levels $(q \geq p)$. It decrements RQ_p foı each empty slot the station detects on bus A. Upon arrival of a segment with priority p to the station, RQ_p is copied to CD_p and then set to zero. Then RQ_p is incremented for slots on bus B having the priority-p request bit set. CD_p is decremented for each empty slot passing by the station on bus A and incremented by one for each slot on bus B having the request bit set at higher priority levels $(q > p)$. When CD_p reaches zero the segment of priority level p is allowed to take the next empty slot for transmission.

The so-called *bandwidth balancing mechanism* [4] was included into the standard to achieve a fair distribution of bandwidth within a single priority class if the network is heavily loaded. The bandwidth balancing mechanism enforces at each priority level that a station uses only a fraction of the available bandwidth for transmissions. This is achieved by incrementing RQ_p each time after a fixed number of β [1] transmissions of priority-p segments.

2.2 Enhancements to the Priority Mechanism

The priority mechanism as described in the previous subsection (including bandwidth balancing) is not effective. It was shown that it merely guarantees that stations with high priority traffic do not obtain more bandwidth than stations with low priority traffic [14, 15, 17]. Without the bandwidth balancing mechanism it is possible that stations with low priority traffic obtain more bandwidth than stations with high priority traffic [18].

Non-unity ratio bandwidth balancing [16] enforces that high priority traffic is assigned more bandwidth than low priority traffic by using different values of β for traffic from different priority levels. Higher values for β are used for high priority traffic. However, high priority traffic is not independent from low priority traffic, i.e., increasing the amount of low priority traffic results in decreased traffic at higher priorities.

In *symmetric bandwidth balancing* [3], stations with low priority traffic leave an additional empty slot that is otherwise used for transmission each time after receiving a fixed number of high priority requests.

In [2], a priority mechanism with pre-emptive priorities is presented. Using additional bits in the slot header, stations notify each other about the highest priority level currently active on the network. A station refrains from transmitting if the highest active priority level is higher than the priority level of segments stored at the station. A fair distribution of bandwidth to stations transmitting at the highest priority level is not guaranteed.

In *bandwidth balancing with global priority information* [5], the slot header carries information on the priority level of transmitted data. The so-called bandwidth balancing modulus (β) is set equal for all priority levels. Under heavy load this priority mechanism distributes the bandwidth equally among transmissions at the same priority level. In addition, high priority traffic is independent from traffic at lower priorities. However, the scheme never utilizes the entire bandwidth of the bus. Variations of this scheme can be found in [6, 9, 10]. We refer to [13] for a detailed discussion of the literature on *DQDB*.

As mentioned before, none of the above schemes achieves at the same time the independence of high priority traffic from low priority traffic, an equal maximum bandwidth allocation for stations transmitting at a particular priority level and a full utilization of the bandwidth of the dual bus.

---

[1] $\beta = 8$ is the default value in [8]

3 Properties of Bandwidth Allocations with Multiple Levels of Priorities

In this section we formally define properties of bandwidth allocation schemes for dual bus networks with multiple priorities. Because of the symmetry of the dual bus topology we only consider transmissions on one bus. Formally, a bandwidth allocation maps the traffic load from all stations into individual portions of the bandwidth that can be used for transmission.

Let $\mathcal{N} = \{1, 2, \ldots, N\}$ be a set of stations and let $\mathcal{P} = \{1, 2, \ldots, P\}$ be a set of priority levels. A high priority index denotes a high priority level. Let λ_{ip} ($\lambda_{ip} \geq 0$) denote the load and γ_{ip} ($0 \leq \gamma_i \leq 1$) denote the allocated bandwidth for station i ($1 \leq i \leq n$) at priority level p ($1 \leq p \leq P$). Both load and allocated bandwidth are normalized over the total bandwidth. Let the ($N \times P$) matrices Λ and Γ be given by:

$$\Lambda = \begin{pmatrix} \lambda_{11} & \lambda_{12} & \cdots & \lambda_{1P} \\ \lambda_{21} & \lambda_{22} & \cdots & \lambda_{2P} \\ \vdots & \vdots & & \vdots \\ \lambda_{N1} & \lambda_{N2} & \cdots & \lambda_{NP} \end{pmatrix}$$

$$\Gamma = \begin{pmatrix} \gamma_{11} & \gamma_{12} & \cdots & \gamma_{1P} \\ \gamma_{21} & \gamma_{22} & \cdots & \gamma_{2P} \\ \vdots & \vdots & & \vdots \\ \gamma_{N1} & \gamma_{N2} & \cdots & \gamma_{NP} \end{pmatrix}$$

We define $\|\Gamma\|$ and $\|\Lambda\|$ as:

$$\|\Lambda\| = \sum_{i=1}^{N}\sum_{p=1}^{P}\lambda_{ip} \qquad \text{and} \qquad \|\Gamma\| = \sum_{i=1}^{N}\sum_{p=1}^{P}\gamma_{ip}.$$

Further we define Λ_p, $\|\Lambda_p\|$, Γ_p and $\|\Gamma_p\|$ ($1 \leq p \leq P$) as:

$$\Lambda_p = \begin{pmatrix} \lambda_{1p} \\ \lambda_{2p} \\ \vdots \\ \lambda_{Np} \end{pmatrix} \qquad \text{and} \qquad \|\Lambda_p\| = \sum_{i=1}^{N}\lambda_{ip}$$

$$\Gamma_p = \begin{pmatrix} \gamma_{1p} \\ \gamma_{2p} \\ \vdots \\ \gamma_{Np} \end{pmatrix} \qquad \text{and} \qquad \|\Gamma_p\| = \sum_{i=1}^{N}\gamma_{ip}.$$

Definition 1 A *bandwidth allocation for multiple priority levels* is defined as a relation $\Omega \subseteq (\Lambda \times \Gamma)$ such that $\gamma_{ip} \leq \lambda_{ip}$ and $0 \leq \|\Gamma\| \leq 1$ (for all $1 \leq i \leq N, 1 \leq p \leq P$).

Although relation Ω does not necessarily determine Γ uniquely for given Λ we will use the notation $\Gamma = \Omega(\Lambda)$. We denote the element in row i and column p of $\Omega(\Lambda)$ by $\Omega_{ip}(\Lambda)$. We denote the pth column of $\Omega(\Lambda)$ by $\Omega_p(\Lambda)$. We use $\|\Omega(\Lambda)\|$ and $\|\Omega_p(\Lambda)\|$ to denote the sum of all elements in matrix $\Omega(\Lambda)$ and vector $\Omega_p(\Lambda)$, respectively.

We define the following fairness criteria for traffic of a particular priority level.

Definition 2 Ω is *fair* if for all Λ

1. $(\forall p)(1 \leq p \leq P)(\forall i,j)(1 \leq i,j \leq N):$
$$(\lambda_{ip} < \lambda_{jp} \rightarrow \Omega_{ip}(\Lambda) \leq \Omega_{jp}(\Lambda)),$$
and
2. $(\forall p)(1 \leq p \leq P)(\forall i,j)(1 \leq i,j \leq N):$
$$(\lambda_{ip} = \lambda_{jp} \rightarrow \Omega_{ip}(\Lambda) = \Omega_{jp}(\Lambda)).$$

The *fairness* conditions guarantee that within each priority level a station does not obtain more bandwidth than a station with a higher arrival rate, and stations with the same arrival rate obtain the same bandwidth.

Definition 3 Ω is *strongly fair* if for all Λ $(\forall p)(1 \leq p \leq P)(\exists \alpha_p^* > 0)(\forall i)(1 \leq i \leq N):$

$$((\lambda_{ip} \leq \alpha_p^* \rightarrow \Omega_{ip}(\Lambda) = \lambda_{ip}) \wedge (\lambda_{ip} > \alpha_p^* \rightarrow \Omega_{ip}(\Lambda) = \alpha_p^*)).$$

Strong fairness guarantees for priority level p that for given load Λ each station cannot obtain more bandwidth than a given threshhold value α_p^*. Stations with a load less than the threshhold value obtain all the bandwidth they need. All stations with a load exceeding the threshold value obtain the same bandwidth. Note that the condition for *strong fairness* implies *fairness*.

Next we formally describe bandwidth allocations that are able to utilize the entire bandwidth of the communication channel.

Definition 4 Ω is *waste-free* if for all Λ

(i) If $\|\Lambda\| < 1$, then $\|\Omega(\Lambda)\| = \|\Lambda\|$, and

(ii) If $\|\Lambda\| \geq 1$, then $\|\Omega(\Lambda)\| = 1$.

If the traffic load from all stations is less than the capacity of the bus, a *waste-free* bandwidth allocation guarantees that all stations can transmit their load. If the traffic load exceeds the capacity, the entire bandwidth can be used for transmission, i.e., no bandwidth is wasted.

The quality of service of the priority mechanism of a bandwidth allocation is categorized as follows.

Definition 5 Ω implements *pseudo-priorities* if for all Λ $(\forall p, q)(1 \leq p < q \leq P):$

$$(\|\Lambda_q\| > \|\Lambda_p\| \rightarrow \|\Omega_q(\Lambda)\| \geq \|\Omega_p(\Lambda)\|)$$

Pseudo-priorities just guarantee that a station with high priority traffic does not obtain less bandwidth than a station with low priority traffic.

Definition 6 Ω implements *weak priorities* if for all Λ $(\forall p, q)(1 \leq p < q \leq P):$

$$(\|\Lambda_q\| > \|\Lambda_p\| \rightarrow \|\Omega_q(\Lambda)\| > \|\Omega_p(\Lambda)\|)$$

and
$(\forall p, q)(1 \leq p < q \leq P)(\exists \epsilon)(\epsilon > 0) : (\|\Omega_q(\Lambda)\| <$

$$\|\Omega_q \begin{pmatrix} \lambda_{11}+\epsilon & \cdots & \lambda_{1p}+\epsilon & \lambda_{1q} & \cdots & \lambda_{1P} \\ \lambda_{21}+\epsilon & \cdots & \lambda_{2p}+\epsilon & \lambda_{2q} & \cdots & \lambda_{2P} \\ \vdots & \vdots & & \vdots & & \vdots \\ \lambda_{N1}+\epsilon & \cdots & \lambda_{Np}+\epsilon & \lambda_{Nq} & \cdots & \lambda_{NP} \end{pmatrix} \|)$$

Weak priorities guarantee that high priority traffic is allocated more bandwidth than low priority traffic. However, if the traffic load from low priority levels is increased by a constant ϵ the allocated bandwidth to the high priority stations is decreased. Therefore, *weak priorities* allow that stations with low priority traffic obtain a portion of the bandwidth even though the arrival rate of high priority traffic exceeds the total capacity of the bus.

Definition 7 Ω implements *strong priorities* if for all Λ $(\forall p, q)(1 \leq p < q \leq P):$

$$(\|\Lambda_q\| > \|\Lambda_p\| \rightarrow \|\Omega_q(\Lambda)\| > \|\Omega_p(\Lambda)\|)$$

and
$(\forall p, q)(1 \leq p < q \leq P)(\forall \epsilon)(\epsilon > 0) : (\Omega_q(\Lambda) =$

$$\Omega_q \begin{pmatrix} \lambda_{11} + \epsilon & \cdots & \lambda_{1p} + \epsilon & \lambda_{1q} & \cdots & \lambda_{1P} \\ \lambda_{21} + \epsilon & \cdots & \lambda_{2p} + \epsilon & \lambda_{2q} & \cdots & \lambda_{2P} \\ \vdots & \vdots & & \vdots & & \vdots \\ \lambda_{N1} + \epsilon & \cdots & \lambda_{Np} + \epsilon & \lambda_{Nq} & \cdots & \lambda_{NP} \end{pmatrix})$$

Strong priorities assign more bandwidth to high priority traffic than to low priority traffic. In addition, the bandwidth allocated to high priority traffic is independent from the load of low priority traffic.

Definition 8 Ω implements *pre-emptive priorities* if for all Λ

$$(\forall p)(1 \leq p \leq P) : (\sum_{q>p} \|\Lambda_q\| > 1 \rightarrow \|\Omega_p(\Lambda)\| = 0)$$

An allocation with *pre-emptive* priorities does not admit low priority traffic if the traffic demand from higher priorities exceeds the total bandwidth. Note that *strong priorities* imply *weak priorities*, and *weak priorities* imply *pseudo-priorities*. In addition, we can follow from Definitions 4 and 7 that a *waste-free* bandwidth allocation Ω with *strong priorities* implements *pre-emptive* priorities, and *pre-emptive* priorities require a *waste-free* allocation with *strong priorities*. In [11] we showed the following properties for dual bus networks with a single level of priorities:

Lemma 1 *Uni-priority DQDB without bandwidth balancing [7] is waste-free, but not fair.*

Lemma 2 *Uni-priority DQDB with bandwidth balancing [8] is strongly fair, but not waste-free.*

With definitions 5 - 8 we can show the following properties of priority mechanisms for dual bus networks:

Lemma 3

- *(i) The priority mechanism of IEEE 802.6 [8] implements* pseudo-priorities.
- *(ii) The priority mechanism of non-unity ratio bandwidth balancing [16] allows to implement* weak *but not strong priorities.*
- *(iii) The priority mechanism of DQDB with bandwidth balancing and global priority information [5] has strong, but not pre-emptive priorities.*

Proof:

(i) Since the bandwidth balancing mechanism ensures *strong fairness* [11], it clearly guarantees ***pseudo-priorities.*** To show that the conditions for *weak* priorities are not satisfied assume a network where each station transmits at one priority level and all stations are heavily loaded. Then the allocated bandwidth to a station i transmitting with priority p is given by [5]:

$$\Omega_{ip}(\Lambda) = \frac{\beta}{1 + \sum_{r=1}^{P} \beta \cdot K_r} \tag{1}$$

where K_r denotes the number of stations transmitting at priority level r. Since $\Omega_{ip}(\Lambda) = \Omega_{jq}(\Lambda)$ (for $1 \leq i, j \leq N, 1 \leq p, q \leq P$) regardless of the load at different priority levels, the condition for *weak priorities* is clearly violated.

(ii) We consider the heavy load scenario as given above. In non-unity ratio bandwidth balancing each priority level p has a different value for the bandwidth balancing modulus, with $\beta_p > \beta_q$ if $p > q$. The bandwidth allocated to a station is calculated by [16]:

$$\Omega_{ip}(\Lambda) = \frac{\beta_p}{1 + \sum_{r=1}^{P} \beta_r \cdot K_r} \tag{2}$$

Then, $\Omega_{ip}(\Lambda) > \Omega_{jq}(\Lambda)$ if $p > q$. However, *strong priorities* are not satisfied, since the bandwidth allocation to stations with high priorities is dependent on low priority traffic.

(iii) Again we consider the heavy load scenario. The allocated bandwidth to station i with priority-p traffic is given by [5]:

$$\Omega_{ip}(\Lambda) = \frac{\beta}{\prod_{q=p}^{P}(1 + \beta \cdot K_q)} \tag{3}$$

Weak priorities are satisfied since $\prod_{q=p}^{P}(1 + \beta \cdot K_q) < \prod_{q=p'}^{P}(1 + \beta \cdot K_q)$ for $p > p'$. The conditions for *strong priorities* are satisfied since the allocated bandwidth Ω_{ip} is independent from priority-q traffic with $q < p$ (equation (3)). To show that the allocation is not *pre-emptive*, we show that it is not *waste-free*. *Waste-freedom* is not satisfied since

$$\sum_{p=1}^{P} \sum_{k=1}^{K_p} \frac{\beta}{\prod_{q=p}^{P}(1 + \beta \cdot K_q)} < 1. \tag{4}$$

An ideal bandwidth allocation with multiple priority traffic combines *strong fairness*, *waste-freedom* and *strong priorities*. The following theorem states that such an allocation can be found.

Theorem 1 *There exists exactly one bandwidth allocation Ω^* that is strongly fair, waste-free and has strong priorities for all Λ. Ω^* is determined by the unique solution to the following system of equations:*

$$\Omega^*_{ip}(\Lambda) = \begin{cases} 0 & \text{if } (\exists q)(q > p) : (|\mathbf{O}_q| > 0) \\ \min\{\lambda_{ip}, \dfrac{1 - \sum\limits_{q \geq p, j \in \mathbf{U}_q} \lambda_{jq}}{|\mathbf{O}_p|}\} & \text{otherwise} \end{cases} \tag{5}$$

with

$$\begin{aligned} \mathbf{U}_p &= \{j \mid \lambda_{jp} = \Omega^*_{ip}(\Lambda)\} & (6) \\ \mathbf{O}_p &= \{j \mid \lambda_{jp} > \Omega^*_{ip}(\Lambda)\} & (7) \end{aligned}$$

$$(1 \leq i \leq N, 1 \leq p \leq P)$$

$\mathbf{U}_p$ denotes the set of *underloaded stations* with priority-p traffic, i.e., stations which can satisfy their bandwidth demand of priority-p traffic. $\mathbf{O}_p$ denotes the set of *overloaded stations* with priority-p traffic, i.e., stations with a load at priority level p exceeding the allocated bandwidth. Note that no bandwidth is allocated to a station if the set of overloaded stations at higher priority levels is nonempty. A complete proof of Theorem 1 is given in [12].

In the remaining part of this section, we show that the *strongly fair* and *waste-free* bandwidth allocation with *strong priorities* can be obtained in a distributed way. Recall that a *waste-free* bandwidth allocation with *strong priorities* is *pre-emptive.*

Definition 9 *For each (i,p) $(1 \leq i \leq N,\ 1 \leq p \leq P)$ define $\mathbf{O}_p^{(>i)}$ by:*

$$\mathbf{O}_p^{(>i)} = \{j \mid j \in \mathbf{O}_p \wedge i < j \leq N\} \tag{8}$$

$\mathbf{O}_p^{(>i)}$ denotes the set of stations in the overload set of priority level p with higher index than i.

Theorem 2 *Given a strongly fair and waste-free bandwidth allocation Ω^* with strong priorities. Then $i \in \mathbf{O}_p$ if and only if*

$$\lambda_{ip} > \delta_{ip} \tag{9}$$

with

$$\delta_{ip} = \begin{cases} 0 & \textit{if } (\exists q)(q > p) : (|\mathbf{O}_q^{(>i)}| > 0) \\ \dfrac{1 - \sum\limits_{q \geq p, j < i} \Omega^*_{jq}(\Lambda) - \sum\limits_{q > p} \Omega^*_{iq}(\Lambda) - \sum\limits_{q \geq p, j > i, j \in \mathbf{U}_q} \lambda_{jq}}{|\mathbf{O}_p^{(>i)}| + 1} & \textit{otherwise} \end{cases} \tag{10}$$

We prove the theorem in [12]. From Theorem 2 we additionally obtain:

Corollary 1 *Given a strongly fair and waste-free bandwidth allocation with strong priorities. For all $i \in \mathbf{O}_p$*

$$\Omega^*_{ip}(\Lambda) = \frac{1 - \sum\limits_{q \geq p, j > i, j \in \mathbf{U}_q} \lambda_{jq} - \sum\limits_{q > p} \Omega^*_{iq}(\Lambda) - \sum\limits_{q \geq p, j < i} \Omega^*_{jq}(\Lambda)}{|\mathbf{O}_p^{(>i)}| + 1} \tag{11}$$

Corollary 2 *The strongly fair and waste-free bandwidth allocation with strong priorities can be implemented if each station i with $\lambda_{ip} > 0$ knows the following set of parameters:*

$$\Big(|\mathbf{O}_p^{(>i)}|,\ \sum_{j > i, q \geq p, j \in \mathbf{U}_q} \lambda_{jq},\ \sum_{q > p} \Omega^*_{iq}(\Lambda),\ \sum_{j < i, q \geq p} \Omega^*_{jq}(\Lambda)\Big).$$

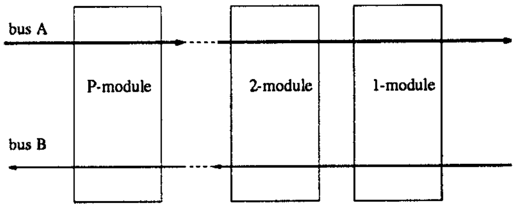

Figure 2: Priority Modules of a Station.

In the following section we present a media access protocol for dual bus networks with multiple priority levels that uses the results from Theorem 2 and Corollaries 1 and 2.

4 A Strongly Fair and Waste-Free Multiple Access Protocol with Pre-Emptive Priorities

We only consider data transmission on bus A. We will use the station index to denote the relative physical distance to the slot generator of bus A. So station 1 will denote the station closest to the slot generator of bus A, station 2 the second closest station, etc. . The stations with greater index than station i are referred to as the downstream stations of station i, stations with smaller index are referred to as the upstream stations.

4.1 Design Concepts

Transmission of traffic from a station is handled independently for each priority level. Each station consists of so-called *modules* that control the transmission of traffic from a particular priority level. The modules of a station are organized such that modules for high priority traffic are upstream from the modules for low priorities (Figure 2). We denote the module that controls traffic of priority level p as the p-module. Each module is considered as either *underloaded* or *overloaded.* An underloaded p-module can satisfy its bandwidth requirements at priority p, an overloaded p-module is characterized by an offered load that exceeds the allocated bandwidth. Both underloaded and overloaded modules use bus B to send reservation requests to upstream stations. Underloaded p-modules send a priority-p reservation request for each segment. If an underloaded p-module becomes overloaded, it stops sending reservation requests, and sends a signal on bus B to notify the upstream stations that it is overloaded. Once the signal is sent, no more reservations requests are submitted. If an overloaded p-module becomes underloaded, it sends a signal on bus B to indicate to upstream stations that it became underloaded. Then, the p-module resumes sending priority-p reservation requests, one for each segment of priority p.

Before a p-module is allowed to transmit a segment of priority p, it has to consider all reservations from downstream modules with equal or higher priority. For each reservation request of priority $\geq p$ and for each overloaded module of priority p downstream, the station has to leave an empty slot on bus A. If a station receives an overload signal of priority $> p$, it ceases transmission until the opposite signal is received.

From Theorem 2 we know the necessary and sufficient overload conditions for the p-module of station i. Since we use the station index to denote the relative position of a station and since the modules of a station are ordered as given in Figure 2, the values in equation (10) can be calculated by:

λ_{ip} : load of priority-p-traffic to p-module of station i,

$\sum_{q\geq p, j<i} \Omega^*_{jq}(\Lambda) + \sum_{q>p} \Omega^*_{iq}(\Lambda)$: rate of busy slots on bus A from priorities $\geq p$ seen by p-module of station i,

$\sum_{q\geq p, j>i, j\in \mathbf{U}_q} \lambda_{jq}$: rate of reservations requests on bus B of priorities $\geq p$ received by p-module of station i,

$|\mathbf{O}_p^{(>i)}|$: number of overloaded p-modules downstream on bus A.

4.2 Implementation

Each slot carries the following bits in the slot header: one request bit and one busy bit for each priority level, a *plus bit* and a *minus bit*. The pth busy bit is set by a module of priority level p when inserting a segment of priority p into a slot. Each underloaded p-module sends one reservation request for the segment on top of the queue of untransmitted priority-p segments. This is done by setting the request bit of priority p in a slot on bus B. If an underloaded module becomes overloaded, it sets a plus bit and a priority-p request bit in a slot on bus B. After setting the plus bit, no more reservation requests are transmitted. If an overloaded p-module becomes underloaded, it sets a minus bit and a request bit of priority p in a slot of bus B, and resumes setting priority-p request bits, one for each segment. Note that neither plus nor minus bits can be set in slots that have the request bit set at any priority level. Note however, that request bits in a slot can be set at all priority levels.

A p-module determines its turn to transmit a (priority-p) segment with five counters, the request counter (*RQ*), the countdown counter (*CD*), the overload request counter (*ORQ*), the overload countdown counter (*OCD*) and the stop-transmission counter (*STOP*). An idle p-module, i.e., a p-module that does not have a priority-p segment queued for transmission, increments *RQ* for each passing slot on bus B with the request bit set at priority level p or higher. *ORQ* is incremented when a slot on bus B passes by with both the plus bit and the priority-p request bit set. *ORQ* is decremented by one for each slot with both the minus bit and the priority-p request bit set. *STOP* is incremented by one if a slot passes by with both the plus bit set and a request bit at a priority $> p$ set, and *STOP* is decremented by one for each slot with both the minus bit and the request bit at a priority $> p$ set.

At an idle p-module, *RQ* is decremented for each empty slot and for each busy slot with the busy bit set at priority $< p$ that passes by on bus A (*RQ* is not decremented if $RQ = 0$).

When a priority-p segment arrives at an idle module, the contents of *RQ* and *ORQ* are copied to *CD* and *OCD*, respectively, and *RQ* is set to zero. Now, *RQ* is incremented for each slot with set request bit at priority p on bus B. *CD* is incremented for each slot on bus B with the request bit set at priority $> p$. *ORQ* is incremented for each slot on bus B with the plus bit and the priority-p request bit set, and *ORQ* is decremented for each slot on bus B with both the minus bit set and the priority-p request bit set. For each empty slot and for each busy slot of priority $< p$ on bus A, *CD* is decremented by one. If *CD* is zero the module decrements *OCD* by one. If an empty slot arrives at the p-module and *CD*, *OCD* and *STOP* are zero, the empty slot is used for transmission of the segment. If the p-module has more segments waiting for transmission, *RQ* and *ORQ* are copied to *CD* and *OCD*, and *RQ* is set to zero.

With Theorem 2, each p-module can determine by itself whether it is overloaded or underloaded. The rates needed to calculate equation (9) are obtained from the values of counters. Most of the required information is already stored in counters *RQ*, *CD*, *ORQ*, and *STOP*. Three additional counters are needed. *NoSeg* contains the total number of segments (of priority p) queued for transmission at the p-module, *SlotCtr* is incremented for each arriving slot on bus B, and *Bsy* is incremented for each busy slot read on bus A with priority p or higher. A p-module determines its state each time *Basis* [2] slots have passed by on bus B ($SlotCtr = Basis$). Then it calculates:

$$Quota = \begin{cases} 0 & \text{if } STOP > 0 \\ \dfrac{Basis - Bsy - RQ - CD}{ORQ + 1} & \text{otherwise} \end{cases} \tag{12}$$

and sets counters *SlotCtr* and *Bsy* to zero. *Quota* provides the maximum number of slots a module is allowed to transmit during a period of *Basis* slots. If $NoSeg > Quota$, the p-module is overloaded, otherwise the p-module is underloaded.

4.3 Evaluation

In order to show that our protocol achieves the objectives of *strong fairness*, *waste-freedom* and *pre-emptive priorities*, we execute simulation runs of file transfer scenarios [3]. We compare our protocol with the priority scheme presented in [5] to our knowledge is the only (verified) bandwidth allocation scheme that satisfies the conditions for both *strong priorities* and *strong fairness*.

We study a dual bus network with four stations that start file transfers on bus A at different times. Each station transmits at a particular priority level. Starting time and priority level of the file transfers for all stations are shown in Table 1 [4]. We assume that station 4 transmits a file with a length of 5,000 segments, the files transmitted by

Station	Start of Transmission	Priority Level
station 1	$t = 1,000$	1
station 2	$t = 4,000$	2
station 3	$t = 9,000$	2
station 4	$t = 17,000$	3

Table 1: Simulation Parameters.

---

[2] We set *Basis* to the round-trip slot delay of the bus, i.e., the sum of the slot lengths of bus A and bus B [11].

[3] The simulations were implemented using the simulator for dual bus network protocols presented in [1]

[4] The time unit is one slot The simulation starts at $t = 0$

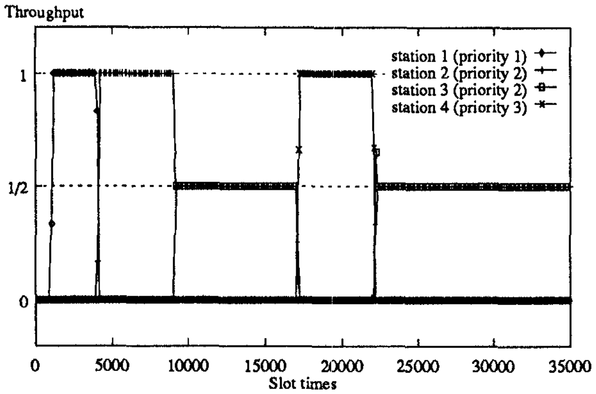

Figure 3: File Transfer with New Priority Scheme (*Basis* = 150; Round-trip delay = 150 slots)

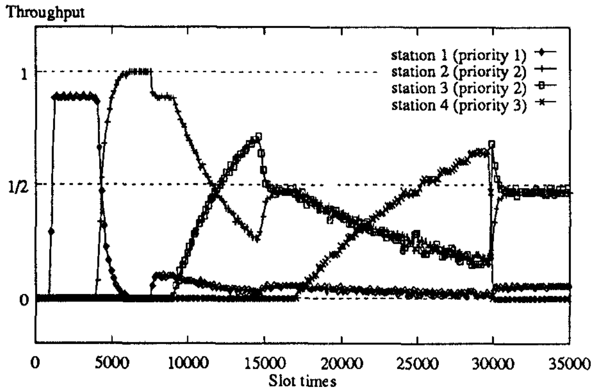

Figure 4: File Transfer with Priority Scheme in [7] ($\beta = 8$; Round-trip delay = 150 slots)

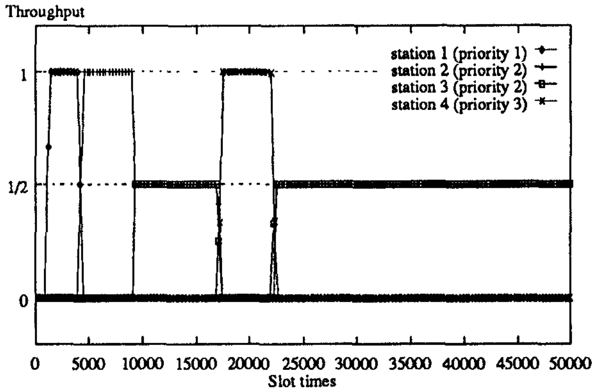

Figure 5: File Transfer with New Priority Scheme (*Basis* = 300; Round-trip delay = 300 slots)

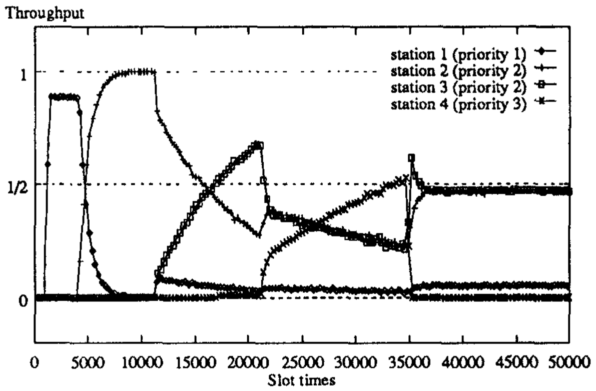

Figure 6: File Transfer with Priority Scheme in [7] ($\beta = 8$; Round-trip delay = 300 slots)

other stations are assumed to be significantly larger. We set the distance between adjacent stations to $\Delta = 25$ slots [5]. The total round-trip delay of the bus is therefore given by 150 slots. Once every round-trip delay we measure the number of segments that each station was able to transmit. The total observation period is set to 35,000 slots. Figure 3 shows the results of the simulation for the new protocol with *Basis* = 150. Each point in Figure 3 gives the percentage of the bandwidth on bus A that is used by a station for transmission, i.e, the throughput of the station, in an interval of one round-trip delay. When station 1 starts transmission (at priority level 1) it seizes the entire bandwidth. As soon as station 2 with priority-2 traffic becomes active, it immediately pre-empts transmissions from station 1. When station 3 begins transmitting with priority 2, it shares the bandwidth with station 2 such that both stations 2 and 3 obtain half of the total bandwidth. At $t = 17,000$, station 4 with traffic at priority level 3 pre-empts any traffic with a lower priority. When all 5,000 segments of station 4 are transmitted, stations 2 and 3 again share the available bandwidth. Note how quickly the new protocol can adapt to changes in the network load.

For comparison, in Figure 4 we present a simulation of the same scenario with the priority scheme given in [5]. As mentioned in section 2.2, the scheme given in [5] is based on the bandwidth balancing mechanism. We use the default value for the bandwidth balancing modulus ($\beta = 8$). Figure 4 shows that each time the load of the network changes, it takes considerable time to adjust to the new network load. Because of the long convergence time, station 4 is not able to pre-empt the traffic from lower priority stations. This results in significantly higher transmission times for high priority traffic compared to the new protocol.

The advantages of the new protocol become even more apparent when the slot distance between the stations is increased. Increasing the slot distance corresponds to increasing the physical distance between stations, or equivalently, increasing the transmission speed of the network. We present the same simulation scenario as in Table 1 for a dual bus network with a slot distance of $\Delta = 50$ and $\Delta = 100$ slots between two adjacent stations. Again we measure the throughput once every round-trip delay. For $\Delta = 50$ slots we present simulation results for a total observation period of 50,000 slots. The results are shown in Figure 5 for the new priority scheme (with *Basis* = 300) and in Figure 6 for the priority scheme from [5]. The new priority scheme is insensitive to doubling the slot distance between the stations, i.e., doubling the transmission rate of the bus. However, using the priority scheme from [5], the convergence time after

---

[5] At a transmission rate of 155 Mb/s one slot corresponds to a distance of about 500 m and to a transmission delay of about 2.8 μs

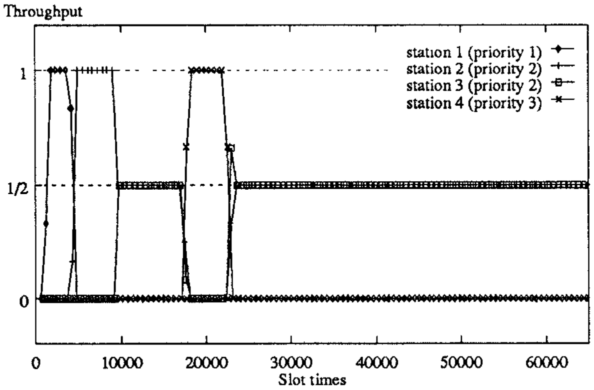

Figure 7: File Transfer with New Priority Scheme (*Basis* = 600; Round-trip delay = 600 slots)

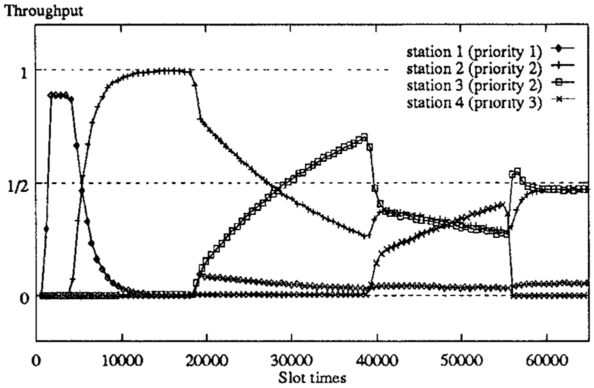

Figure 8: File Transfer with Priority Scheme in [7] ($\beta = 8$; Round-trip delay = 600 slots)

load changes increases significantly compared to the previous simulation.

For $\Delta = 100$ we choose an observation period of 65,000 slots. Again the priority scheme of our protocol is effective as shown in Figure 7. In Figure 8 we show that the bandwidth balanced priority scheme [5] is almost ineffective. Although station 4 (with priority 3) becomes active at $t = 17,000$, it achieves a non-zero throughput for the first time at about $t = 40,000$.

	$\Delta = 25$	$\Delta = 50$	$\Delta = 100$
New Protocol	5,076	5,152	5,302
Protocol in [5] with $\beta = 8$	12,732	17,785	38,440

Table 2: Transmission Delay for File from Station 3 (File Length = 5,000 Segments).

In Table 2, we present the exact transmission time of the priority 3 file transmitted from station 4 for all simulation runs. The transmission time is the interval in slot times (see footnote 5) from the time station 4 becomes active (at $t = 17,000$) until the last of its 5,000 segments is transmitted. The table clearly shows the superiority of our new protocol.

5 Conclusions

We provided a unified view on priority mechanisms for dual bus networks by formalizing properties of bandwidth allocation schemes. We showed deficiencies of existing protocols that support multiple priority traffic. We presented a new priority mechanism that does not waste bandwidth, provides a fair distribution of bandwidth within each priority level, and provides *pre-emptive* priorities. We proved the uniqueness of the priority mechanism. We introduced a media access protocol that is able to provide the unique priority mechanism. We showed that the new protocol achieves the implementation of *pre-emptive* priorities. The performance of the protocol was compared to an implementation of a priority mechanism that provides ***strong priorities*** and ***strong fairness*** (according to our terminology used in section 3). Our protocol adapts quicker adapt to changes in the network load. We achieve a significantly lower transmission delay for high priority traffic compared to other priority schemes.

References

[1] C. Bisdikian. DQDB Delay Analysis: The BWB Mechanism Case. In *4th IEEE Workshop on MANs*, pages 5.1.1 - 5.1.28, Captiva Islands, FL, November 1990.

[2] C. Bisdikian and A. Tantawy. A Mechanism for Implementing Preemptive Priorities in DQDB Subnetworks. In *Proc. ICC 1991*, pages 1062–1067, Denver, CO, June 1991.

[3] R. Breault. Enhancement to the Bandwidth Balancing Mechanism. In *Contribution No. 802.6-90/03 to the IEEE 802.6 Working Group*, January 1990.

[4] E. L. Hahne, A. K. Choudhury, and N. F. Maxemchuk. Improving the Fairness of Distributed Queue-Dual-Bus Networks. In *Proc. INFOCOM '90*, pages 175 - 184, San Francisco, CA, June 1990.

[5] E. L. Hahne and N. F. Maxemchuk. Fair Access of Multi-Priority Traffic to Distributed-Queue Dual-Bus Networks. In *Proc. INFOCOM '91*, pages 889 - 900, Bal Harbour, FL, April 1991.

[6] K. Han and I. Hyun. The Dynamic Bandwidth Balancing Mechanism for Improving DQDB Performance. In *Proc. ICC 1991*, pages 1345 - 1349, June 1991.

[7] IEEE. 802.6 Working Group, Proposed Standard: Distributed Queue Dual Bus (DQDB), Subnetwork of a Metropolitan Area Network (MAN), February 1990. P802.6/D12, Unapproved Draft.

[8] IEEE. Std 802.6-1990, IEEE Standards for Local and Metropolitan Area Networks: Distributed Queue Dual Bus (DQDB) of a Metropolitan Area Network (MAN), July 1991.

[9] D. J. Jeong, C. H. Choi, and W. S. Jeon. Fairness Improvement in DQ Protocols With Multiple Priorities. In *Proc. ICC 1991*, pages 1340 - 1344, June 1991.

[10] S. Y. Kim and K. J. Han. An Enhanced Bandwidth Balancing Mechanism of the DQDB MAN. In *Proc. ICC 1991*, pages 423 - 427, Denver, CO, June 1991.

[11] J. Liebeherr, I.F. Akyildiz, and A.N. Tantawi. Distributed Fair and Waste-Free Media Access Protocols for Dual Bus Metropolitan Area Networks. Technical Report RC 17489, IBM Research Division, December 1991.

[12] J. Liebeherr, I.F. Akyildiz, and A.N. Tantawi. An Effective Scheme for Pre-Empitve Priorities in Dual Bus Metropolitan Area Networks. Technical Report RC 17720, IBM Research Division, February 1992.

[13] B. Mukherjee and C. Bisdikian. A Journey Through the DQDB Literature. Technical Report RC 17016, IBM Research Division, July 1991.

[14] M. Spratt. A Problem in the Multi-Priority Implementation of the Bandwidth Balancing Mechanism. In *Contribution to the IEEE 802.6 Working Group*, November 1989.

[15] M. Spratt. Implementing the Priorities in the Bandwidth Balancing Mechanism. In *Contribution No. 802.6-90/05 to the IEEE 802.6 Working Group*, January 1990.

[16] M. P. Spratt. Allocation of Bandwidth in IEEE802.6 using Non-Unity Ratio Bandwidth Balancing. In *Proc. ICC 1991*, pages 729–735, Denver, CO, June 1991.

[17] H. R. van As. Performance Evaluation of the Bandwidth Balancing in the DQDB MAC Protocol. In *Proc. 8th Annual EFOC/LAN Conference*, pages 231 – 239, June 1990.

[18] H. R. van As, J. W. Wong, and P. Zafiropoulo. Fairness, Priority and Predictability of the DQDB MAC Protocol under Heavy Load. In *Proc. 1990 International Seminar on Digital Communication*, pages 410 – 417, Zurich, Switzerland, March 1990.