Université d'Ottawa · University of Ottawa

Connection Oriented Data Service in DQDB Metropolitan Area Network

by

Xiaojing Tao, B.A.Sc.

A thesis submitted to the
School of Graduate Studies and Research
in partial fulfillment of the requirements for the degree of

Master of Applied science
in electrical Engineering

Ottawa-Carleton Institute of Electrical and Computer Engineering
School of information Technology and Engineering

Faculty of Engineering
University of Ottawa

August, 1998

0-612-38769-0

Abstract

This thesis examines and suggests solutions to some of the technical problems and performance problems encountered when designing, modeling, and evaluating a Connection-Oriented (CO) data service in Distributed Queue Dual Bus (DQDB) Metropolitan Area Network. The CO data service primarily supports applications with high Quality of Service (QoS) requirements, such as video and audio. These applications usually are time-sensitive so that a guaranteed bandwidth and limited access delay should be warranted. The challenges in designing the CO service will involve issues such as performance, reliability, evaluating of the system, availability, etc.

We present and study a CO service in the base of a former proposed theorem. Our research emphasizes on the ability of CO service to support various types of applications and source sharing among CO and other services. The proposed CO service is designed following the standard distributed queue principle of providing a moderate fairness of utilization. It provides scalability while maintaining the requirement of the two most important performance metrics: CO service throughput, and CO service response time.

Acknowledgements

I would like to express my sincere gratitude to my supervisor, Dr. Luis Orozco-Barbosa. Thank you for giving me the opportunity to participate and cooperate in the MAN Connection-Oriented service research project. I really enjoyed working under your guidance, leadership and supervision.

Many thanks to Dr. N. D. Georganas (Director of the MCRLab) and Dr. Luis Orozco-Barbosa, for procuring me with more than adequate research facilities. Dr. Orozco-Barbosa was the principal investigator of the MAN Connection-Oriented service project.

I am also grateful to the Natural Sciences and Engineering Council of Canada (NSERC), for given me the opportunity to study, learn and acquire knowledge in the field of MAN. (This work was supported by the Natural Sciences and Engineering Council of Canada (NSERC) under grant number OGPIN 334.)

I would like to thank my parents, Guosen Tao and Yaoren Chen, for providing me with full support and love throughout my studies.

Many thanks to some of my working colleagues, especially: Dr. Nasir Uddin Ahmed (professor at the University of Ottawa), Mr. Francois Malric (the MCRLab manager), Mr. Jeff Brinskelle (formerly the MCRLab manager). Thank you all for your comments, help, exchange of knowledge and friendliness throughout the course of my studies. It was really appreciated.

Table of contents

List of Acronyms

ACF	Access Control Field
ANSI	American National Standards Institute
ATM	Asynchronous Transfer Mode
AU	Access Unit
B-ISDN	Broadband Integrated Service Digital network
BWB	Bandwidth Balancing
BWBM	Bandwidth Balancing Modulus
CATV	Cable Television
CC	Credit Counter
CCITT	Consultative Committee on International Telegraphy and Telephony
CD	Countdown Counter
CO	Connection Oriented
CRC	Cyclic Redundancy Check
CRM	Maximum Credit
DMPDU	Derived MAC Protocol Data Unit
DQDB	Distributed-Queue Dual-Bus
DQSM	Distributed Queue State Machine
FDDI	Fiber Distributed Data Interface
GBW	Guaranteed Bandwidth
HOB	Head of Bus
IMPDU	Initial MAC Protocol Data Unit
INC	Income per Slot
ISDN	Integrated Service Digital Network
ISO	International Standards Organization
LAN	Local Area Network
LLC	Logical Link Control
LME	Physical Layer Management Interface

MAC	Medium Access Control
MAN	Metropolitan Area Network
MID	Message Identifier
MSDU	MAC Service Data Unit
NMP	Network Management Process
OCD	Overload Countdown Counter
ORC	Overload Request Counter
OSI	Open System Interconnection
PA	Pre-Arbitrated
PCM	Pulse Code Modilated
PDU	Protocol Data Unit
PLCP	Physical Layer Convergence Function
QA	Queued Arbitrated
QoS	Quality of Service
QPSX	Queued Packet Synchronous Exchange
RC	Request Counter
SAP	Service Access Point
SGC	Segment Cost
SMDS	Switched Multi-Megabit Data Service
SONET	Synchronous Optical NETwork
TRC	Trigger Counter
VCI	Virtual Channel Identifier
WAN	Wide Area Network

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

Communication networks are becoming commonplace nowadays. The local area network (LAN) [29, 30] is used in connecting equipment owned by the same organization over relatively short distances such as campus and institute. The wide area network (WAN) provides long-haul communication services to various points within a large geographical area, e.g., a continent. As the performance of LANs degrades when the area of coverage becomes large, LANs have limitations in geography, speed, traffic capacity, and the number of stations they are able to connect. Meanwhile, WANs are not proper to serve for a modest community of users within a 25-km radius due to such factors as history, design, cost, performance, and service types.

There is a need for an intermediate network that operates over a metropolitan area at comparatively high data rates and with reasonable protocol complexity [31, 32]. With some characteristics of LANs and some of WANs, the metropolitan area network (MAN) technology embraces the best features of both [1, 33, 34, 35, 36, 37]. The interconnection of LANs, MANs, and WANs is shown in Figure 1.

Metropolitan area networks are basically an outgrowth of LANs. The MAN effort was started in 1982, about two years after IEEE project 802 was initiated. The functional goals were to provide for interconnection of LANs, bulk data transfer, digitized voice, and video, as well as conventional terminal traffic. It was assumed that cable television (CATV), fiber optics, and radio were valid choices of media.

The concept of a MAN is modeled after that of a LAN, operating at speeds of at least 10 Mbps, but applied to a larger geographical area (nominally 50 kms in diameter). Typically, a MAN spans an entire office park, an entire campus, or an entire city and its suburbs. A MAN is therefore a high-speed network that interconnects a number of LANs, thereby allowing the sharing of resources and information within a metro community.

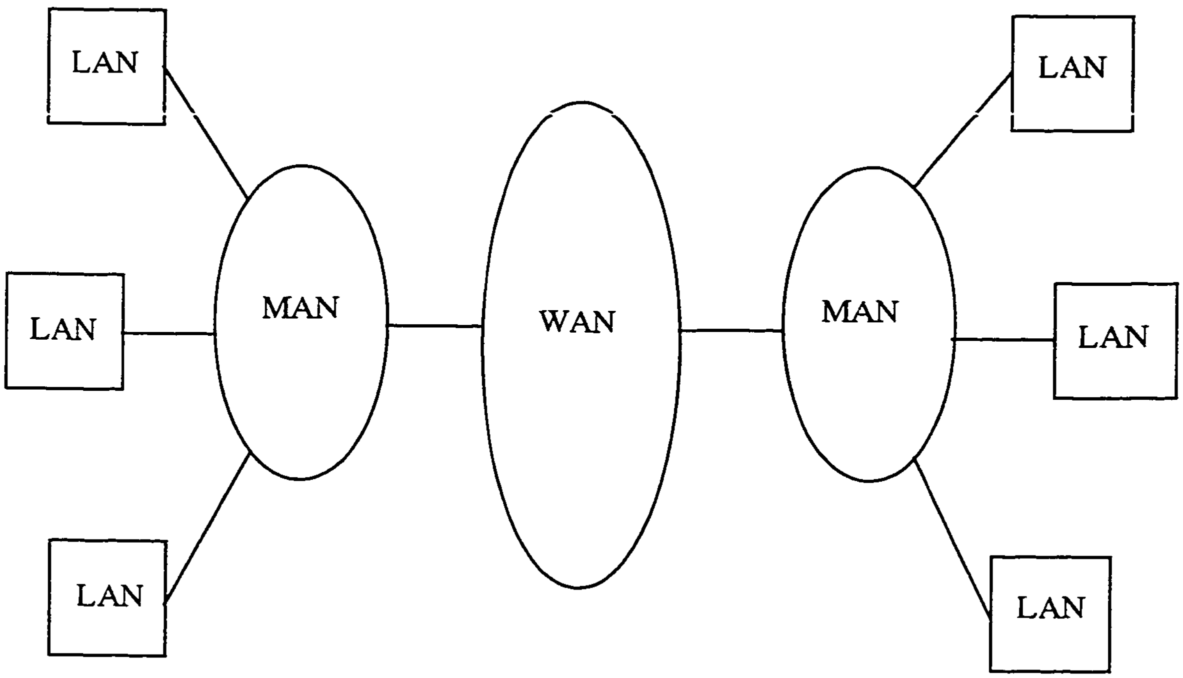


Figure 1. Interconnection of LANs, MANs, and WANs.

After many years of research on MANs, a number of alternatives have been explored and rejected. One approach has emerged that has received widespread support from providers and users, and has been standardized by the IEEE 802 committee as IEEE 802.6. It is useful to look at their definition:

> A MAN is optimized for a larger geographical area than a LAN, ranging from several blocks of buildings to entire cities. As with local networks, MANs can also depend on communications channels of moderate-to-high data rates. Error rates and delay may be slightly higher than might be obtained on a LAN. A MAN might be owned and operated by a single organization, but usually will be used by many individuals and organizations. MANs might also be owned and operated as public utilities. They will often provide means for internetworking of local networks. Although not a requirement for all LANs, the capability to perform local networking of integrated voice and data (IVD) devices is considered an optional function for a LAN. Likewise, such capabilities in a network covering a metropolitan area are optional functions of a MAN.

There are two kinds of services MANs can provide. The first type is a broadband nonisochronous service over a metropolitan area. This service, known as switched multi-megabit data service (SMDS), has already been defined and deployed. The second type of service is to provide isochronous service (video and voice).

The IEEE 802 committee has reached a consensus to use the distributed-queue dual-bus (DQDB) protocol as the standard MAN. Besides the IEEE DQDB MAN, other types of MAN have been proposed. Expressent and Fasnet are first successful two sets of protocols to be proposed in MANs [38, 39, 40, 41, 42]. The fiber distributed data interface (FDDI) [43, 44, 45, 46], the most popular example of an extended LAN, is also proposed as a dual token ring to support MAN. One may classify the different proposed non-DQDB MANs into three categories: (1) tree-based MANs, (2) toroid-based MANs, and (3) LAN-based MANs.

Although the metropolitan area network is closer to implementation today than it was several years ago, it has not yet realized its full potential because metropolitan area networks need to address problems that local area networks do not face. The key issues facing MANs includes: (1) cost, (2) security, (3) reliability, (4) compatibility with current and future networks, (5) human factors, and (6) network management. Although some of these issues are interrelated, they are the major factors that can influence the choice of schemes. The issues are holding back MANs from realizing their full potential and must be addressed in implementing MANs. Any MAN service must offer capabilities to address these and other related issues.

1.2 Research Contributions

With respect to proposed mechanism and performance of the DQDB network, this thesis adds new contributions and results to the existing literature. The contributions and results are mostly focused on the full definition of connection-oriented (CO) service in DQDB network and its performance analysis. They are:

1. To develop the CO service in a simulated DQDB network;
2. To analyze and improve some existing problems for the implementation of the CO service;
3. To provide efficient methods on solving the interference between the different service;
4. Description of more complicate mechanism to realize multiple services in each node;
5. To analyze the system with known traffic model;
6. Reusability of simulation for future research.

During the course of this research, the following conference paper was published:

X. Tao, L. O. Barbosa, *Tuning-up the DQDB Connection-Oriented Data Service*, Proceedings of IEEE CCECE'98, pp.137-140, Waterloo, Canada, May 23-26 1998.

1.3 Outline of the Thesis

Chapter 2 presents an overview of the DQDB network. Important notions such as characteristics, physical and functional architectures, frame structure, access unit and access mechanism are described. The bandwidth balance (BWB) mechanism is also mentioned as it is already standardized by the IEEE 802.6 committee for the queue-arbitrated (QA) service.

Chapter 3 gives an introduction to applications that are suitable to be supported by the CO service. Two main problems, the interference between two CO service nodes and disturbance from QA service, are mentioned and analyzed. Several methods are suggested to solve these problems. A dynamic adjustment mechanism is introduced to set up the CO system parameters. Furthermore, a detailed flow chart of CO service is given with analysis of access delay and priority mechanism in integrated service node.

Chapter 4 considers more advanced DQDB network configuration. Self-similar and MPEG video traffics are used to evaluated the CO service and compared with the results obtained using traditional Poisson traffic. Some strategies are included to alleviate the problems encountered in simulation. The performance metrics observed include access delay, throughput, bandwidth utilization, etc.

Chapter 5 gives a conclusion with respect to the topics investigated in this thesis.

Chapter 2

Distributed-Queue Dual-Bus

The IEEE 802 committee perceived the need for high-speed services over wide areas and formed the IEEE 802.6 Metropolitan Area Network (MAN) committee in 1982. Several systems were proposed to the IEEE 802.6 committee, but in November 1987 the committee reached a consensus to use Distributed-Queue Dual-Bus (DQDB) as the standard medium-access-control (MAC) protocol. DQDB is modeled after the Queued Packet and Synchronous Exchange (QPSX) protocol, developed at the University of Western Australia and proposed by Telecom Australia to the IEEE 802.6 committee [47]. Besides being a MAC standard for MAN, DQDB is also regarded by the ANSI T1S1.1 committee as the major component in the broadband ISDN user network interface standard.

Based on DQDB, vendors can build components that will allow communication services within a metropolitan area not exceeding 50 km in diameter.

2.1 Basic Features

The DQDB standard is both a protocol and a subnetwork. It is a subnetwork in that it is a component in a collection of networks to provide a service. For example, multiple DQDB subnetworks could be interconnected by bridges, routers, or gateways to form MANs and WANs. The term distributed-queue, dual-bus refers to the use of a dual-bus topology and a MAC technique based on the maintenance of distributed queues. In other words, each station connected to the subnetwork maintains queues of outstanding requests that determine access to the MAN medium. The DQDB MAC protocol may be regard as a hybrid multiplexer managing both circuit-switching and packet-switching traffic.

The DQDB subnetwork has many features, some of which make it attractive for high-speed data services:

- Shared Media: It extends the capabilities of shared media systems over large geographical areas.
- Fault Tolerance: It is tolerant to transmission faults when the system is configured in a loop.
- Congestion Control: It is based on a distributed queuing algorithm as a way of resolving congestion. This protocol ensures that a high utilization of the transmission resources and their fair allocation are possible under overload conditions. Also, a priority mechanism is provided that enables a network operator to provide preferential performance to some types of traffic over other types.
- Connectionless Service Provision: It provides a connectionless services at the MAC layer as other IEEE LANs.
- Segmentation: Its use of ATM technique allows long, variable-length packets to be segmented into short, fixed-length segments.
- Flexibility: It is able to utilize a variety of media, including coaxial cable and fiber optics. It can simultaneously support both circuit-switched and packet-switched services.
- Compatibility; It is compatible with current IEEE 802 LANs and future networks such as B-ISDN. In fact, many view DQDB as an early implementation of B-ISDN.

The DQDB standard allows a MAN to be formed by interconnecting DQDB subnetworks via bridges, routers, and gateways. DQDB may also serve as a backbone network. A major difference between DQDB and most of earlier slotted networks is that a reverse channel is used to reserve slots. Also, DQDB supports traffic of multiple priority levels. The network span of about 50 km, transmission rate of about 150 Mbps, and slot size of 53 bytes allow many slots to be in transit between the nodes.

2.2 Architecture

2.2.1 Physical Architecture

The DQDB dual-bus topology is identical to that used in Fasnet. As both buses are operational at all times, the capacity of the subnetwork is twice the capacity of each bus. In this network, nodes are connected to two unidirectional buses, which operate independently and propagate in opposite directions as shown in Figure 2. The nodes are attached to both buses via a logically OR writing tap and a reading tap that is logically placed ahead of the writing tap. Every node is able to send information on one bus and receive on the other bus. The head station (frame generator) generates a frame every 125 μs to suit digitized voice requirement. The frames are continuously generated on each bus so that there is never any period of silence on the bus. The frame is subdivided into equal-sized slots. The empty slots generated can be written into by the other nodes. The end station (slave frame generator) terminates the forward bus, removes all incoming slots and generates the same slot pattern at the same transmission rate on the opposite bus [48]. (The node at the head of the bus is sometimes called the head of bus.) The slots are 53 octets long, the same as ATM cells, to make DQDB MANs compatible with BISDN.

The subnetwork operates in one of two topologies: open bus (Figure 2) or looped bus (Figure 3). The two topologies provide basically the same service. The looped bus topology is well suited for network configuration in case of physical break or failure.

A single node is the head of both buses in a looped bus topology, whereas two stations take on the HOB functions in open bus architecture.

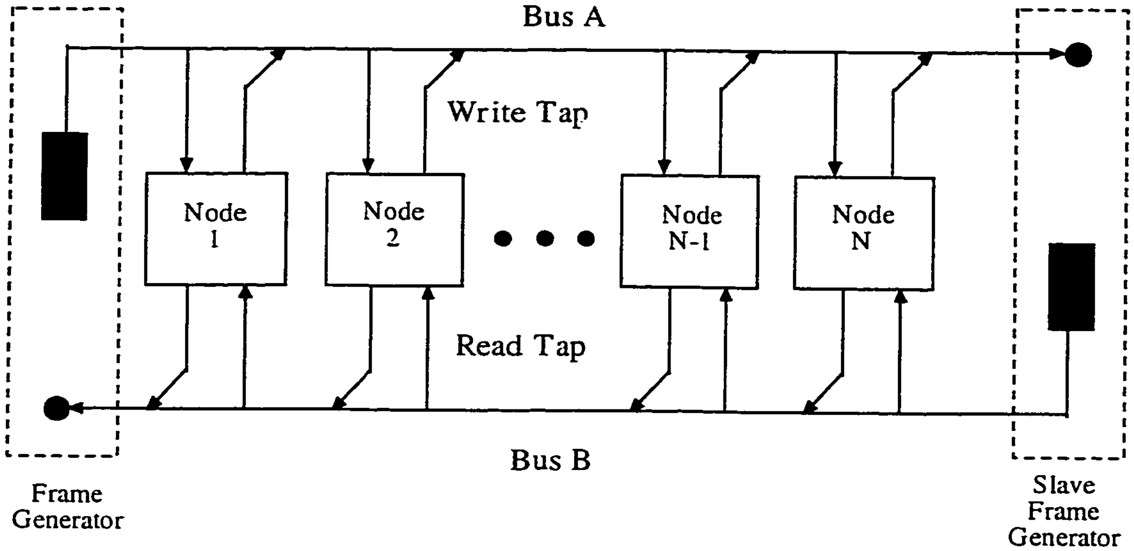


Figure 2. Open bus topology of DQDB network

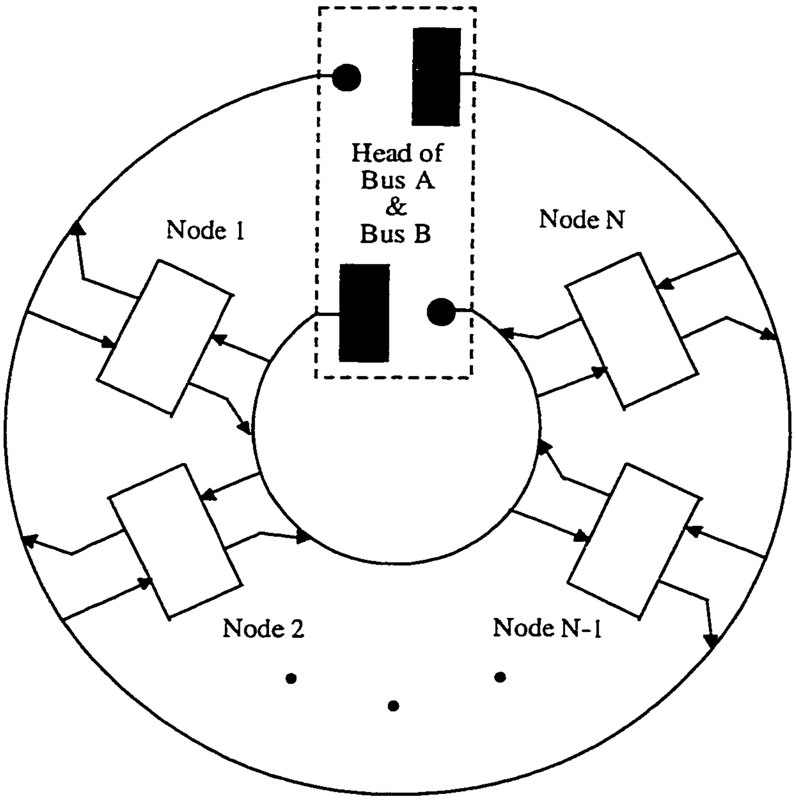


Figure 3 Looped bus topology of DQDB network.

2.2.2 Functional Architecture

The functional architecture of a typical DQDB node consists of the physical layer and the DQDB MAC layer, as shown in Figure 4. The physical layer, which corresponds to the OSI physical layer, is responsible for timing and synchronization issues and monitoring the error rate on the incoming transmission links to ensure quality service. The layer incorporates a convergence protocol, which provides a consistent service to the DQDB layer regardless of the transmission system. The three transmission systems referenced in the standard are DS3 transmitting at 44.736 Mbps over coaxial cable or fiber, SONET transmitting at 155.2 Mbps over single-mode fiber, and G.703, which transmitting at 34.368 Mbps and 139.264 Mbps over a metallic medium.

The DQDB layer is equivalent to the MAC sublayer. It is responsible for addressing, framing, sequencing, error detection, medium access control, fragmenting messages into individual slots, and reassembling slots into messages. It is intended to provide three major services:

1. A connectionless data service, which provides support for connectionless MAC data service to LLC
2. An isochronous service, which provides support for users which require a constant interarrival time. This allows the transfer of data between two isochronous service users over an already established isochronous connection.
3. A connection-oriented data service, which supports the transfer of segments between nodes sharing a virtual channel connection

2.2.2.1 Physical Layer Functions

The generic physical layer contains three functional entities: the transmission system, the physical layer convergence function, and the layer management functions. Each of these functions is described in the following subsections.

The Physical Layer service is provided to the DQDB Layer entity at a node through two SAPs. Each SAP is associated with one duplex transmission link connecting the node to an adjacent node.

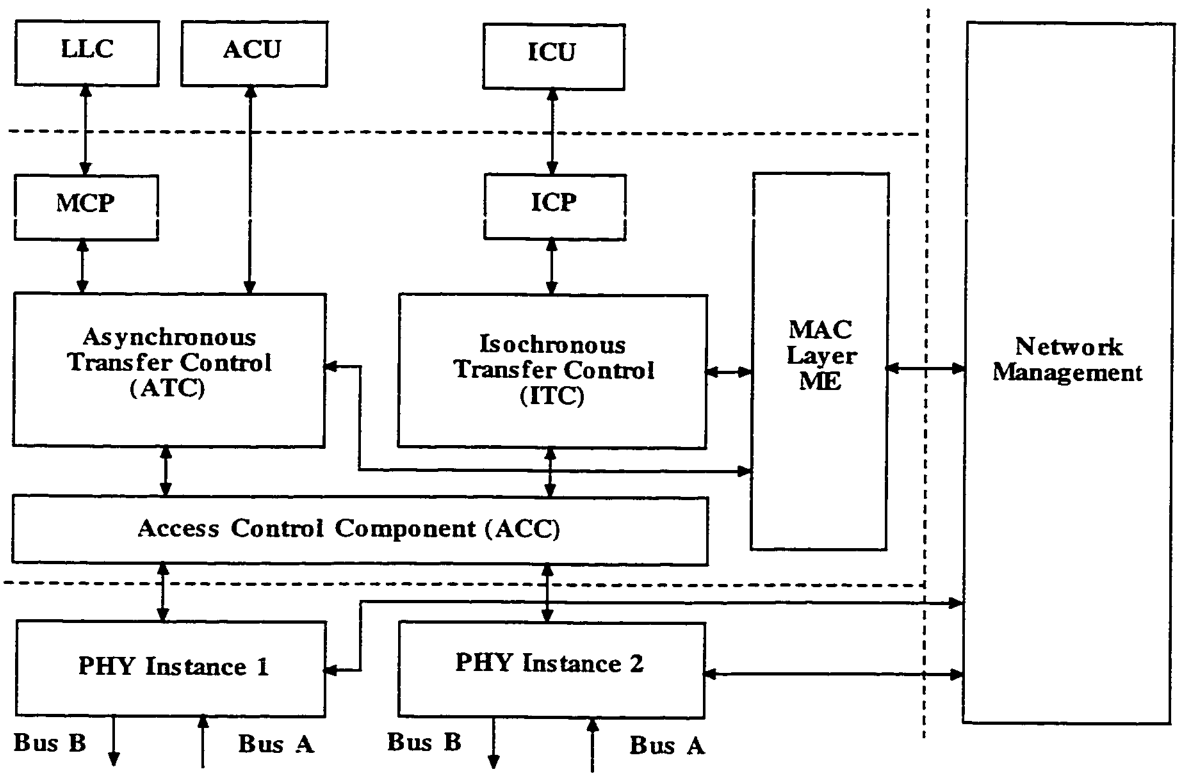


Figure 4. Functional architecture of a DQDB node.

Transmission System

The transmission system function provides a standard transmission interface used to access the transmission link between adjacent nodes. Standard public network transmission systems identified as suitable to provide this function are those specified in CCITT Recommendation G.703 [54], the DS3 rate specified in T1.102 [55], T1.107 [56], and CCITT Recommendation G.707 [57], G.708 [58], G.709 [59] SDH. Other transmission systems may also be supported in the future.

Physical Layer Convergence Function

In order to allow the DQDB Layer to operate independently of the nature of the transmission system, a physical layer convergence function is used. This function provides the standard Physical Layer to DQDB Layer service, irrespective of the nature of the underlying transmission system. Therefore, each different transmission system will require the definition of a unique Physical Layer Convergence Procedure (PLCP). This standard will define the physical layer convergence procedure for each standard transmission interface supported. If the transmission system already provides the defined Physical Layer service, the PLCP would be a null function.

In the future this standard may also define a PLCP that does not use an existing standard transmission system, but which defines this function as part of the protocol. This approach may be suitable for private backbone networks using the DQDB access mechanism.

Physical Layer Management Entity (LME)

The Physical LME performs management of the local Physical Layer Functions. The Physical LME also provides the Physical Layer Management Interface (LMI) to the

Network Management Process (NMP) for the remote management of the local physical subsystem.

2.2.2.2 DQDB Layer Functions

Within the DQDB Layer there are four principal types of functions: the common functions, the access control functions (Queued Arbitrated (QA) and Pre-Arbitrated (PA)), the convergence functions, and the layer management functions. Each of these functions is described below.

Common Functions

The Common Function block acts as a DQDB Layer relay for the transfer of slots and management information octets between the two SAPs to the local Physical Layer entity. Thus, the Common Functions block allows the QA Functions block and PA Functions block to gain read and write access to the QA and PA slots.

The Common Functions block also operates on the slots and management information octets to support functions that operate at some or all of the nodes in the subnetwork. These functions include the head of bus function, the Configuration Control Function, and the MID Page Allocation Function.

The Configuration Control and MID Page Allocation Functions manage DQDB Layer objects necessary for nodes to communicate on the subnetwork. Hence, these functions are part of the DQDB LME. However, there is a requirement for coordination of the DQDB LMEs at all nodes to support these functions. The Common Functions block of the DQDB Layer supports the DQDB Layer Management Protocol for coordination of the Configuration Control and MID Page Allocation Functions.

Queued Arbitrated (QA) Functions

The QA functional entity provides an asynchronous data transfer service for 48-octet segment payloads. The QA functional entity accepts the segment payloads from a convergence function, and adds the appropriate segment header, including VCI, to the segment payload to create a QA segment. The QA segment is queued for access to the dual bus by use of the Distributed Queueing Function. QA segments received by the QA functional entity are stripped of the segment header and the payload is passed to the correct convergence function, based on the VCI value in the segment header.

Pre-Arbitrated (PA) Functions

The PA functional entity provides access control for the connection-oriented transfer over a guaranteed bandwidth channel of octets that form part of an octet stream. The operation of the PA functional entity requires the previous establishment of a connection. As a result of the connection establishment, the PA functional entity will be informed of the VCI value for segments used in the connection and the offset of the octets to be used for reading and writing within the multiple user PA segment payload.

The PA entity accepts the single octets from a convergence function and writes them into the pre-allocated positions within the payload of PA segments with the appropriate VCI value in the segment header. The VCI value will have been set by the Slot Marking Function at the head of bus.

To receive an octet stream, the PA functional entity, on receiving a PA segment with the correct VCI value, will copy octets from the pre-allocated positions within the segment payload. The octets are passed to the correct convergence function, based on the VCI value in the segment header and the offset of the octet in the PA segment payload.

Convergence Functions

It is intended that the DQDB Layer will provide a range of services including connectionless data transfer, isochronous data transfer, and connection-oriented data transfer. These services are provided by convergence functions placed above the QA and PA functional entities. This standard specifies the convergence function to support the connectionless MAC data service to LLC and gives guidelines for the provision of an isochronous service.

DQDB Layer Management Entity (LME)

The DQDB LME performs management of the local DQDB Layer Functions. It also communicates with the DQDB LMEs at other nodes to provide distributed management of the DQDB Layer resources. The communication uses a DQDB Layer Management Protocol supported by the DQDB Layer Common Functions block. The DQDB LME also provides the DQDB Layer Management Interface to the Network Management Process for the remote management of the local DQDB Layer subsystem.

2.3 Frame Structure

In a DQDB subnetwork, the frame (and consequently the transmission bandwidth) is segmented in slots. The frame structure and slot format are shown in Figure 5. Each slot consists of a 53-byte transmission unit for isochoronous and asynchoronous traffic, comprised of a 1-byte acess control field (ACF), a 4-byte segment header, and a 48-byte segment payload.

The ACF (the first byte of each segment header) is composed of 8 bits (1 octet or 1 byte) whose functions are as follows:

1. BUSY: This bit indicates if the slot has data (BUSY=1), or can be accessed (BUSY=0).
2. TYPE: There are two slot types, and this bit indicates whether the slot is prearbitrated (PA) for transfering isochoronous segments, TYPE=1 or is queue arbitrated (QA) for carrying asynchronous segments, TYPE=0. The prearbitrated access mode is reserved for isochoronous services, that is, services that require a fixed but smooth bandwidth, such as voice and video services.
3. RESERVED: This bit is for future use and is presently set to 0.
4. PSR: This indicates whether or not the previous slot has been received.
5. REQUEST: There are four REQUEST bits indicating if a slot contains a request for transmissions. The four bits offer four priority levels.

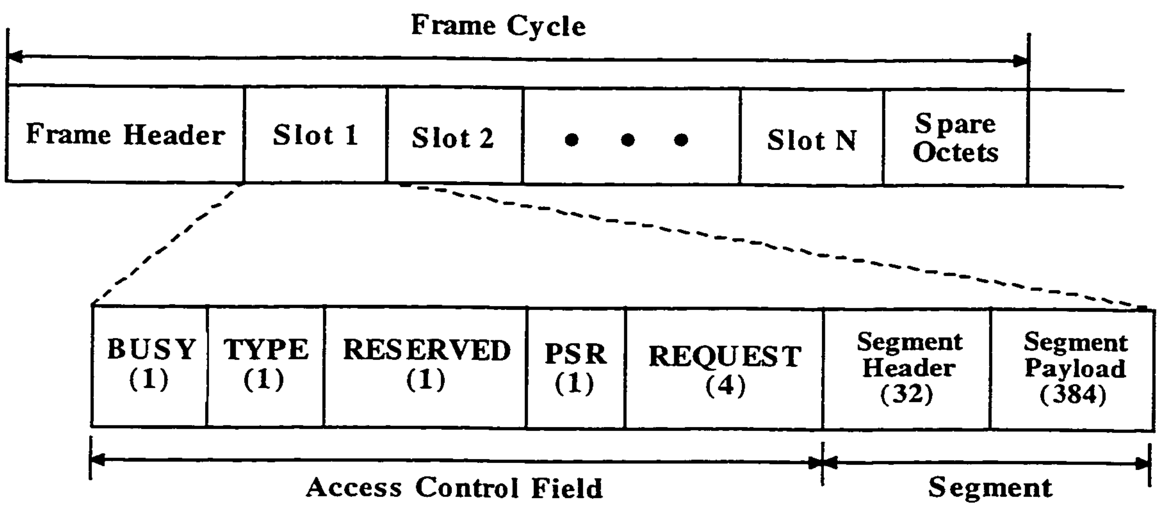


Figure 5. Frame structure and slot format of a DQDB node.

A segment consists of a header and a payload as shown in Figure 6. The segment payload carries the 48 bytes of data. In PA segments, the segments may be shared by a number of isochronous service users. In QA segments, the contents of the payload are not restricted. The segment header field consists of the following:

1. Virtual channel identifier (VCI): This is a 20-bit field used in identifying the virtual channel to which the segment belongs. (A virtual channel is a logical connection.) Since there is no destination or source address associated with an isochronous connection, the VCI and the byte offset determine the byte in a PA slot that has been preassigned to a node. The VCI value of all ones (hexadecimal FFFFF) corresponds to the connectionless service; other nonzero VCI values are for other services.
2. Payload type: A 2-bit field indicating the type of data being transmitted. It is presently set to 00 for user data.
3. Segment Priority: Another 2-bit field presently set to 00 for a multiport bridge, which connects many subnetworks.
4. Header checker sequence: This 8-bit field is meant for detecting error in the segment header field. The generating polynomial used for this purpose is

$$G(x)=x^8+x^2+x+1$$

2.4 Access Unit (AU) and attachment

The nodes on the DQDB subnetwork consist of an access unit and the attachment of the access unit to the two buses. The access unit performs the node's DQDB Layer Functions and attaches to a bus via one read and one write connection. The writing of data to the bus is a logical OR of the data from upstream with the data from the access unit. The read connection is placed logically ahead of the write connection and allows all data to be copied from the bus unaffected by the unit's own writing.

The read and write functions can be implemented using either passive or active techniques. An example implementation of the AU connection is shown in Figure 7. With this connection arrangement, an AU can copy data that passes on the bus, but can never remove it, and only alter data when it is permitted by the access protocol.

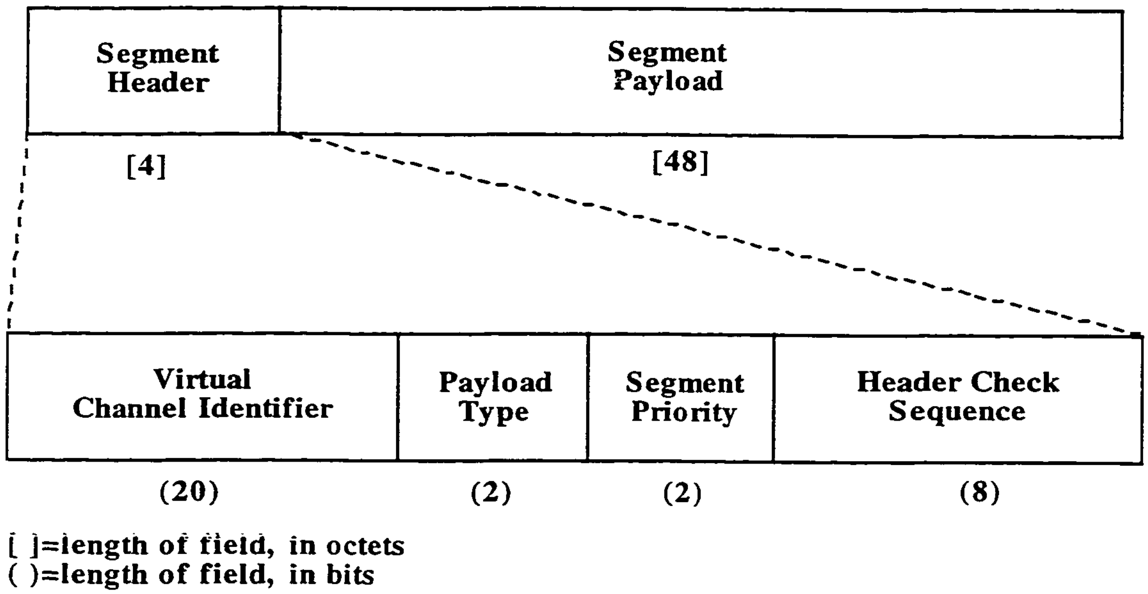


Figure 6. DQDB segment and segment header format.

With the exception of AUs performing the head of bus functions, the operation of the bus is only loosely coupled to the operation of each AU; an AU can fail (provided that it does not destructively write to the bus), or even be removed from the DQDB subnetwork, with no consequence on the operation of the rest of the subnetwork.

Under normal conditions there is a single source of slot timing for the subnetwork, which is used to ensure that all nodes in the subnetwork transfer data (i.e., slots and management information octets) at the same average rate. This is necessary to ensure stable operation of the Distributed Queue access mechanism and to ensure that isochronous communications can be supported without slips being introduced by different timing sources at the two communicating entities.

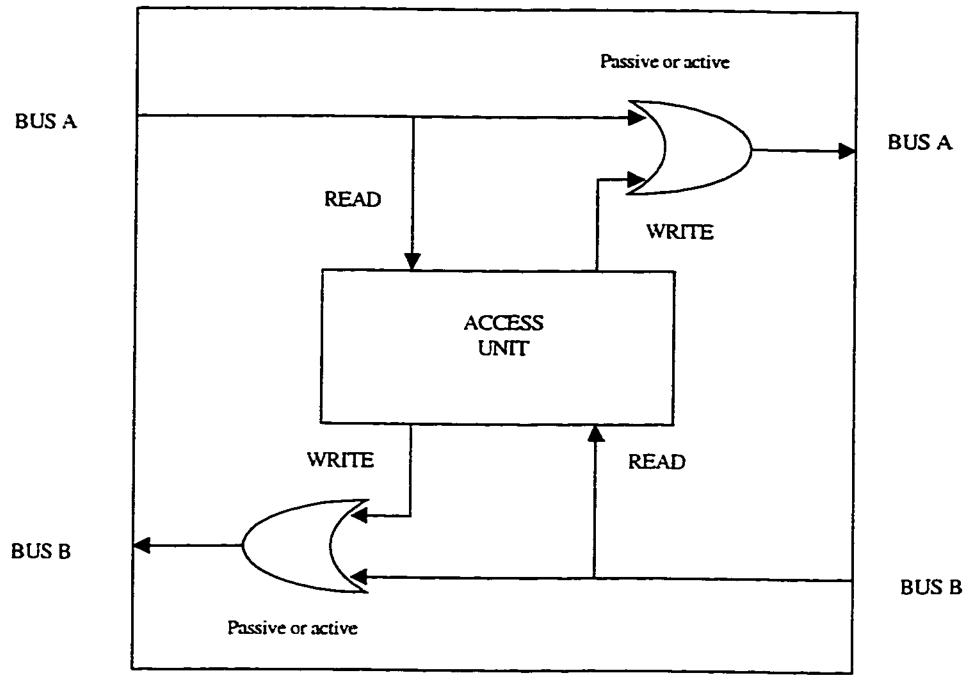


Figure 7. Example Access Unit Attachment.

The timing source can be traceable to an external timing source, such as that provided by a public telecommunications network. Any node or nodes can be designated by network operations procedures to provide the External Timing Source Function. If the subnetwork is supporting certain isochronous services and is connected to a public network, then the external timing source may be required. However, in the absence of an external timing source, one node with the capability specified below will be selected by subnetwork configuration control procedures to be the source of subnetwork timing. In this case, the selected node must have available and operate a local clock that has a nominal period of 125 µs.

2.5 Access Mechanism

Unique to the DQDB subnetwork is the concept of distributed queues (Dqs). Unlike the local queues, the distributed queues are merely logical queues. The service mechanism within each distributed queue is nonpreemptive priority queueing of segments with FCFS for each priority. The distributed queue is implemented by having stations keep track of segment requests from stations downstream, as observed on the reverse bus. To archieve this, each station has a set of two physical counters per priority: the request-bit counter (RC) and the countdown counter (CD), as shown in Figure 7. The RC keeps track of current empty-slot requirements of the downstream stations, whereas the content of the CD indicates the number of empty slots to be passed before the station's own data transmission takes place. Access to the buses is controlled by these counters.

As mentioned earlier, two types of slots are carried on the bus: queued arbitrated (QA) and pre-arbitrated (PA) slots. Corresponding to the two kinds of slots are two different access methods: QA and PA.

2.5.1 Queued Arbitrated Access

The QA access method supports services that are not time-sensitive, such as asynchronous data transfer. Nodes gain access to QA slots by using the distributed-queue access protocol augmented by two features designed to optimize the protocol: bandwidth balancing and priorities. This access protocol is a distributed reservation scheme. This implies at least three things: (1) time on the medium is divided into slots; (2) a node that has data to transmit must reserve a future slot; (3) the nodes collectively determine the order in which slots are granted.

When a node has a QA segment to transmit, it chooses the bus that leads to the intended destination (forward bus) and then sends on the other bus requests for free slots. Assume, without loss of generality, that a node is transmitting on bus A and making a request for transmission on bus B. (The same discussion holds true for transmission on bus B because the operation of both buses is identical.) The node joins the distributed queue by putting its request on bus B (thereby informing upstream nodes) and waits for its turn. Each node maintains the state of the distributed queue in a distributed queue state machine (DQSM), which monitors the transmission procedure through the request counter and the countdown counter. The DQSM can be in either of two states: idle or countdown. When a station has no segments to transmit, the DQSM is in the idle state. In that state, the RC is incremented by 1 for every slot with a reserved request (REQUEST=1) passing by on bus B, and decremented by 1 for every empty slot (BUSY=0) passing by on bus A. When the queue has segments to transmit, it enters the countdown state. In that state, the content of RC is copied into CD and RC is reset to 0, then the CD is decremented by 1 for every empty slot passing by on bus A. Through the CD, each node knows how many empty slots it must let pass by before its turn. When the value of the CD reaches 0, the station grabs the next empty slot, sets the BUSY bit to 1 and writes its data on the payload field. After the station accesses the slot, it enters the idle state again and monitors the requests as before. Figure 9 gives the state transfer chart of QA service. Here busy state is countdown state mentioned above.

A station is allowed to send only one request for every transmission and cannot send another request before it transmits the segment corresponding to its previous reservation. Only after a request has been serviced can the next request be sent. Hence, for two buses and four priority levels, there can be at most eight pending requests for each station. To summarize, to gain access on a bus using the QA protocol, a node

- Sends a request for transmission to the upstream nodes
- Joins the distributed queue by using the countdown counter

- Maintains the distributed queue with the request counter
- Allows the passage of enough empty slots to satisfy the request of the nodes ahead of it in the distributed queue
- Sends its own data when its turn comes

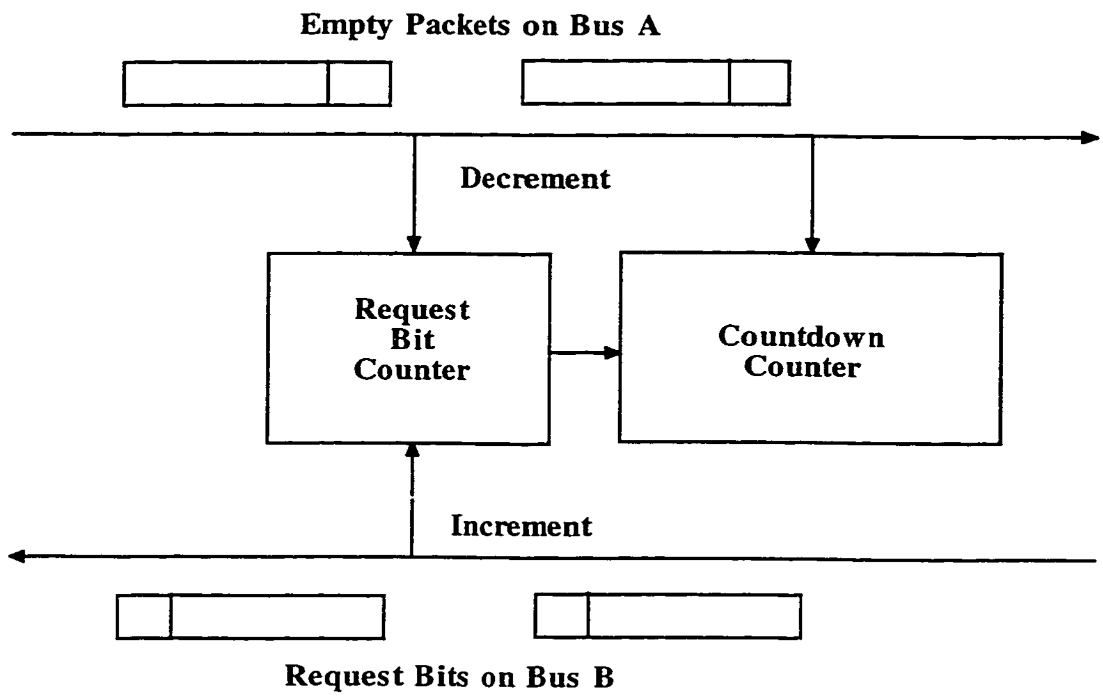


Figure 8. The request and countdown counters.

Ideally, priority queueing enables selective degradation, under overloaded conditions, of lower priority services, allowing preferential access to higher priority delay-sensitive services. The distributed queue mechanism with priority, specified in the standard, can be relied upon to provide fair sharing of capacity and preferential access if the active stations accessing a given bus span a distance that is less than the equivalent of one 53-octet slot, i.e., approximately the following:

2 km	at	44.736 Mb/s
546 m	at	155.520 Mb/s
137 m	at	622.080 Mb/s

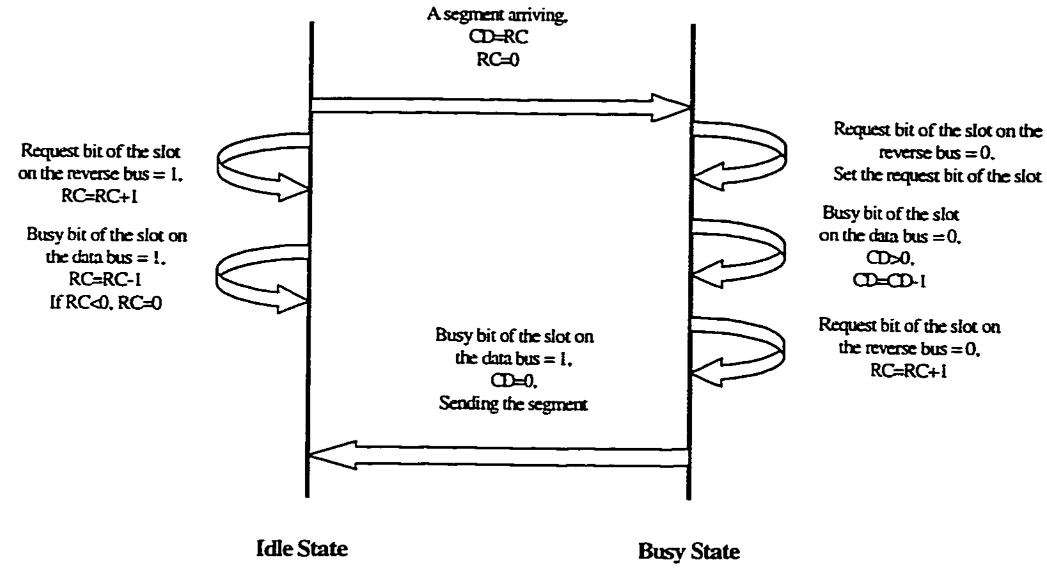


Figure 9. State transfer chart of DQDB QA service.

Beyond these distances, particularly under overload conditions, the effectiveness of the priority queueing mechanism decreases with increasing number of slots on the bus between contending stations.

2.5.2 Pre-Arbitrated Access

The PA access method supports time-sensitive applications such as isochronous connection-oriented services. Nodes gain access to the PA slot by following a request-and-assign procedure. A node establishes connection with another node by requesting bandwidth from a call/connection control entity. The HOB is responsible for marking the slots and placing a VCI (virtual channel identifier) in the slot header. The HOB also sends a sufficient number of PA slots to meet the requested bandwidth. Each node examines the VCI in passing slots and maintains a table indicating each VCI value it must access and the position(s) within the slot for reading and/or writing. If the VCI matches one listed in the node's table, the node reads from and/or writes into the appropriate position(s) in the slot.

Unlike in a pure ring architecture, data does not pass through each station. A station on the bus reads the addresses of passing information slots and copies the data if the destination address in the information slot matches that of station.

DQDB is able to integrate traffic with two different priorities, where the lowest priority is subdivided into four priority levels. The high-priority slots are allocated by the master station to be shared between a number of selected stations; the low-priority slots can be used by all stations.

The major advantages of the network are full bandwidth utilization and low waiting delays. The main cause of performance degradation in DQDB is due to the packet segmentation and the resulting accumulated slot overhead.

2.6 Provided services

The Queued Arbitrated and Pre-arbitrated Functions of the DQDB layer provide access control to the dual buses. These access control functions are used by a range of convergence functions to create the DQDB Layer services. The standard specifies the convergence function for the provision of the IEEE 802 MAC Sublayer service to the LLC Sublayer (ISO 8802-2). Also specified is a general isochronous service.

At no time do two convergence functions simultaneously use same VCI at a given node. Hence, if a node is to receive a particular segment, the VCI can be used to direct the segment to the appropriate convergence function at the node

2.6.1 Provision of MAC Service to LLC

The provision of MAC service to LLC consists of the segmentation of the MAC Service Data Unit (MSDU) at the source into fixed-length units and the transfer of these fixed-length units to the destination, which reassembles them into the MSDU.

The segmentation process follows the formation of an Initial MAC Protocol Data Unit (IMPDU) by the addition of an IMPDU header, an optional Header Extension, an optional 32-bit CRC, a Common PDU trailer, and a variable-length PAD Field to the MSDU. The PAD Field ensures that all of the fields added to the MSDU are 32-bit aligned. The IMPDU is fragmented into fixed-length segmentation units for transfer in QA segment payloads. There may be padding of the IMPDU with trailing zero octets to ensure complete filling of the last segment unit.

All segment payloads that support the MAC service are called Derived MAC Protocol Data Units (DMPDUs) and consist of a header field and a trailer along with the segmentation unit. The DMPDU header field consists of three subfields. The first is the Segment Type Subfield. The second is the Sequence Number subfield. The third subfield is the Message Identifier (MID). The DMPDU trailer field consists of two subfields. The first is the Payload Length Subfield. The second subfield is the Payload CRC.

The MID is used to provide the logical linking between segmentation units derived from the same IMPDU, and should be unique on a subnetwork while the IMPDU is being transferred. Each AU will have at least one unique MID. The allocation of the MID numbers is controlled by the MID page allocation scheme, which is a distributed method for claiming and keeping MID values that are unique across the whole subnetwork. The MID identifies all of the DMPDUs derived from a single IMPDU, and is used in the reassembly of the segmented IMPDU at the destination.

2.6.2 Provision of Isochronous Service

The Pre-Arbitrated access control mechanism does not necessarily accept or deliver isochronous service octets in an isochronous fashion. The variation from isochronous delivery occurs when the node at the head of bus does not generate Pre-Arbitrated slots in an isochronous fashion. If the user of the isochronous service requires the DQDB Layer to accept and/or deliver the octets isochronously, there is a requirement for some buffering. The nature of this buffering depends on the isochronous service user and is performed as part of an isochronous convergence function.

2.6.3 Provision of Connection-Oriented Data Service

DQDB provides both connectionless and connection-oriented telecommunication services. Connectionless services are asynchronous and correspond to the datagram form of packet switching. Connection-oriented services involve a call setup and correspond to virtual-circuit packet switching, and circuit switching.

Connection-oriented services can use the DQDB network after an initial call setup procedure until the connection is broken. As a result, of the call setup process a virtual channel identifier (VCI) is allocated to the call, which uniquely identifies it amongst all calls in progress at the same time. The VCI is carried in the header of the DQDB slot. Standards for call setup itself are available for use in conjunction with DQDB.

There are two types of connection oriented services offered by DQDB: isochronous and asynchronous. Isochronous services handle a regular stream of octets seperated by equal time intervals and are designed for pulse code modilated (PCM) voice and fixed-bit-rate video services. The essential requirement of these services is that there be very little mean end-to-end delay and very little variability of end-to-end delay. On the other hand, asynchronous services operate on a virtual channel and are designed for users with asynchronous data, which can tolerate some delay in transmission.

In the case of connectionless services, data packets containing a destination address are transmitted on to the DQDB network without any call setup procedure. In this case, a special VCI is used to indicate that the destination of the packet is given in the data portion of the DQDB slot rather than by the VCI itself. This is the basis of the switched multimegabit data service.

2.7 Bandwidth Balancing

It has been observed that early versions of DQDB were unable to provide a fair distribution of bandwidth to stations under overload traffic conditions, because upstream stations can use a larger transmission bandwidth than downstream stations, and stations using a large amount of bandwidth tend to prevent other stations' access. One way of explaining this unfairness issue is that the DQDB protocol pushes the system too hard. The problem was overcome by adding the bandwidth balancing (BWB) mechanism to the protocol [49, 50, 51]. The main purpose of the modified protocol is to compel heavy users not to take unfair advantage of their position and monopolize the network. The BWB mechanism leaks some bandwidth in order to prevent the hogging of bandwidth under overload conditions. It does this by forcing nodes whose offered load is larger than the available bandwidth to be rate controlled. Each of these nodes is allocated a fraction R of the available bandwidth. The value of R is

$$R = \frac{\beta(1 - S)}{1 + N\beta} \tag{1}$$

where S is the total bandwidth used by the nodes that are not rate controlled, N is the number of nodes that are rate controlled, and β is a proportionality constant known as the BWB modulus. With a startup value (or default value) of 8, the value of β varies from 0 to 64 and is set according to traffic conditions.

To implement the BWB mechanism, one more counter for each direction at each node is required. This counter, known as the trigger counter (TRC) is increased by 1 every time the node transmit a data segment. When the value of the TRC reaches the value of β, the RC is increased by 1 and the TRC is set to 0. This artificial increment of the RC forces the rate-controlled node to let one empty slot pass by on the node after every β slots, even though no request was sent by the downstream nodes. The downstream nodes may not use this empty slot, resulting in bandwidth wastage. Thus, the bandwidth balancing mechanism creates some other problems. It forces the network to leave a certain percentage of the available bandwidth unused, so full bandwidth utilization is given up to provide fairness to all stations. Also, the four levels of priority given by the original protocol are eliminated.

Since its adoption by the IEEE 802.6 committee, the DQDB subnetwork has generated a significant interest due to the simplicity of its medium access control protocol. This intense interest has led to a literature explosion on DQDB.

2.8 New Connection-Oriented Service Standard on DQDB

IEEE Standard Board has presented the standardized Connection-Oriented Service on a Distributed Queue Dual Bus (DQDB) Subnetwork of a Metropolitan Area Network (MAN). Enhanced Queued Arbitrated (QA) Functions, which can support applications requiring bandwidth guarantees and delay limits on a DQDB subnetwork, are specified. Connection-Oriented Convergence Functions (COCFs) using the enhanced QA Functions, which are necessary to support connection-oriented service, are also specified.

Different COCFs optimized for different types of applications are envisaged on top of the QA Functions block, using for example, AAL3/4 or AAL5. The enhanced QA functions are necessary for providing services with bandwidth guarantees and delay limits. The enhanced QA functions are compatible with the QA functions as defined in ISO/IEC 8802-6: 1994. Due to the layered approach, the enhanced QA functions can be used by different convergence functions.

Applications that are expected to be supported by this services fall into the category of low to medium bandwidth compressed digital video, i.e., 348 kbits/s to 44.2097 Mbit/s. Other applications such as time-critical data communication or frame relay interworking are not precluded.

Chapter 3

Connection Oriented Data Service

The Connection Oriented Data Service (CO service) has been defined by the IEEE 802.6 committee to support variable-bit-rate (VBR) data services over DQDB MANs. The main aim of the CO service is to be able to guarantee the QoS required by a wide variety of VBR services, such as real-time computer interconnection, interactive CAD/CAM, realtime control, telemetry, video conference, video-telephony, video mail service, video and audio distribution, telemedicine, electronic publishing, TV, videography and many others.

The CO service under study herein is different from the connection-oriented service described in the previous chapter. The latter makes use of pre-arbitrated (PA) slots to convey the information among the communicating nodes. Furthermore, it requires the establishment of a connection between the peer entities using a signalling protocol similar to Q.931. On the contrary, the CO service to be introduced in this chapter makes use of the QA access mechanism and it does not require the establishment of a connection between the peer entities. Otherwise, the CO service makes use of the segmentation and reassembly procedures also used by all the other services (connectionless and connection-oriented). In particular, it uses the same format for the Common PDU header and Common PDU trailer.

The CO service has been designed around a control mechanism known as the *guaranteed bandwidth* (GBW) protocol. Many studies have been conducted on the performance of the GBW protocol being used as the underlying mechanism of the CO service of DQDB MANs. However, none of these studies has clearly specified the requirements for the effective implementation of the GBW protocol. In this thesis, we undertake a detailed study of the implementation issues for the effective operation of the overall CO service.

From now on, we will refer to the connection-oriented data service simply as the CO service and to the connectionless service as the QA service. We have preferred to keep the name QA service since this term has been widely used in the literature. For simplicity, we will refer to the nodes implementing the CO or QA services simply by CO nodes and QA nodes, respectively. Furthermore, we will often refer to node 1 as the node placed at the upstream end of the data bus and all subsequent nodes as nodes i, for i = 2, 3, ..., n with node n being the one placed at the downstream end of the data bus.

3.1 The GBW protocol

In [52], Karvelas has described the operation of the Guaranteed Bandwidth (GBW) protocol. Under the GBW protocol, a Credit Counter (CC) and three parameters: the Segment Cost (SGC), the Income Per Slot (INC), and the maximum credit (CRM) are used by each node for supporting the CO data service. A node increases its CC by INC for every slot passing by on the reverse bus, also called the request bus, i.e., the bus used by the node to set a request for a free slot on the data bus. Whenever the node has a segment ready to transmit for which a request has not been already sent and if CC > SGC, the node issues a request and decreases CC by SGC. By requiring each node to transmit only on reserved slots and by appropriately selecting the values of INC and SGC, the GBW protocol can guarantee the bandwidth required by a CO node. For instance, if the channel bandwidth is 155 Mbps by setting INC=30 and SGC=155 a bandwidth of 30 Mbps is guaranteed. This can also be explained as follows. Assume that CC has been set initially to 0. CC will be increased by INC for every slot passing by on the reverse bus. CC will only become greater than SGC after having seen 6 slots on the reverse bus. At this time the node can issue a second request while setting CC=25. The node is not allowed to place another request but 5 slots after while setting CC to 20. Following this procedure, CC will be reset to 0 again (one period), 31 slots from the initial time during which a total of 6 requests have been placed. Therefore, in one period, 6 out of 31 slots will be used by the node. In the case of a channel rate of 155 Mbps, the guaranteed bandwidth is $155 \times 6/31=30$ Mbps.

In order to prevent CC for increasing indefinitely during the periods when the node is idle, a maximum value of credit CRM is introduced. That is, if the value of CC exceeds CRM, the node will not continue to increase CC although it will observe slots passing by on the reverse bus. It is evident that the minimum value of CRM that will provide the node with the guaranteed throughput is the one that satisfies the following inequality:

$$CRM \geq \lceil SGC/INC \rceil \bullet INC \tag{2}$$

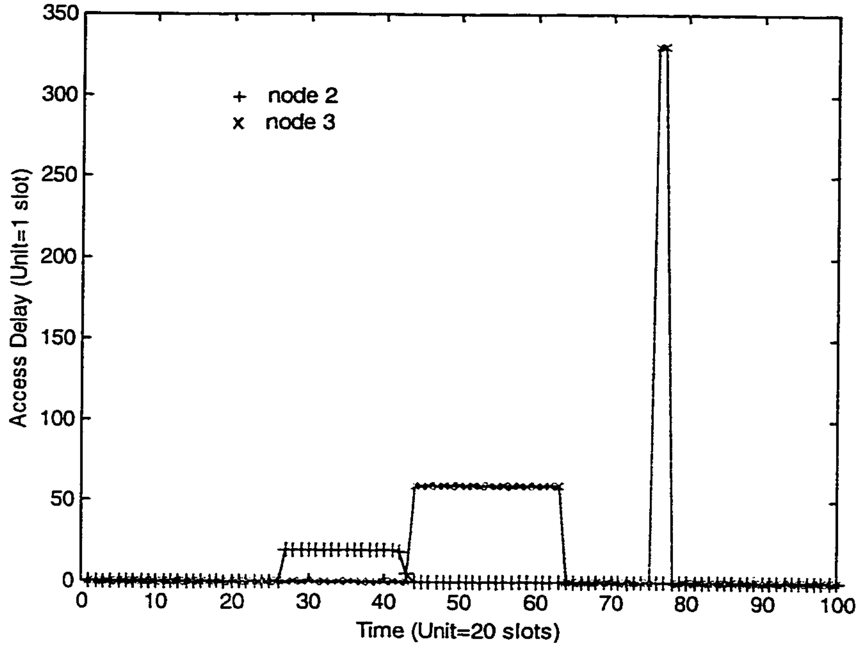
where $\lceil x \rceil$ is the minimum integer which is greater than x.

The GBW protocol is intended to be used by the nodes implementing the CO service. Some performance studies have shown the effectiveness of the GBW in providing the guaranteed bandwidth under various scenarios [52][53]. However, there are still many issues to be addressed. In this study we undertake the analysis of the following main issues: 1) implementation requirements of the CO service, 2) integration of QA and CO services, 3) tuning of the CO parameters, 4) fairness, and 5) operation of hybrid (CO/QA) nodes.

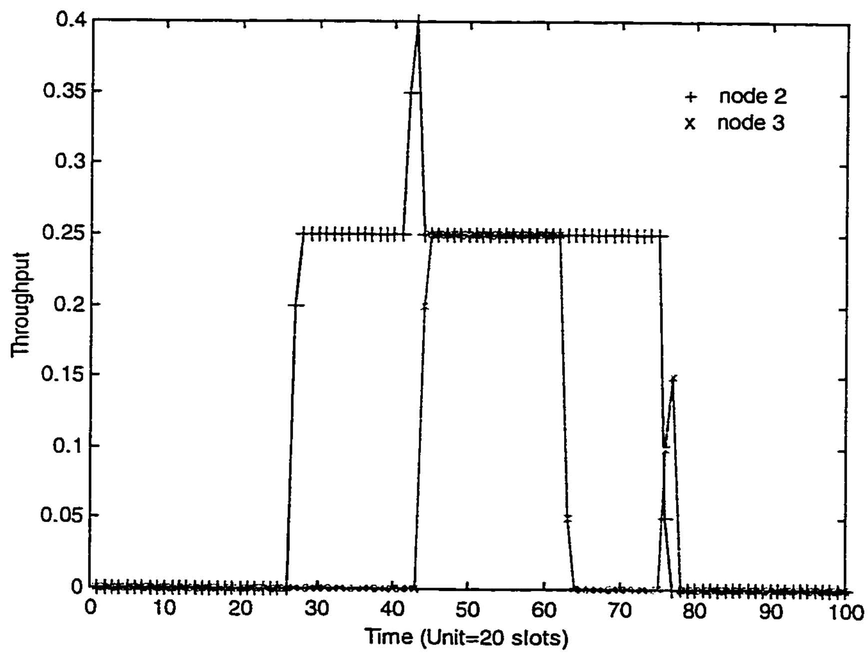
3.2 Multiple Requests

According to the specifications of the GBW mechanism, a node implementing the CO service can issue multiple requests. It is therefore important to analyse the requirements to implement effectively this feature. To do so, we conducted a set of simulations. We considered a system consisting of three nodes with the most upstream node, node 1, implementing the QA service while the two other nodes, node 2 and node 3, implement the CO service. At the beginning of the simulation, the QA node occupies the whole channel capacity. Each one of the two CO nodes requires 25% of the total channel capacity. Both nodes implement the CO service by using the GBW procedure, but they do not make use of a RC nor of a CD counter. The distance between two adjacent nodes has been set to 10 slots. The access delay for both CO nodes as a function of the running time is shown in figures 10a and 10b. The access delay is defined as the time elapsed between the instant when a node places a request and the instant when the node initiates the transmission of the segment associated to the request.

Under this scenario, Figure 10a, the node 2 becomes active before node 3. When node 3 becomes active, node 2 remains active and ends its transmission after node 3 has finished transmitting. As shown in Figure 10a, when the node 2 becomes active and as long as node 3 remains inactive, the access delay for node 2 remains constant to 20 slots. This delay corresponds to the round-trip delay between node 2 and 1. After node 3 starts to send its segments, the access delay of node 2 decreases to zero and node 3 experiences a longer access delay (60 slots). This delay becomes ever longer for some segments of node 3 because they can not be sent before node 2 stops sending even though no more segments come to node 3 for a long time. Those segments will experience very long access delay (330 slots) as they have to wait for node 2 finishing its segments sending. If we switch the active time of node 2 and 3, the access time of node 3 is 40 slots at first. After node 2 becomes active, its access delay increases to



(a) The overall access delay of two CO service nodes



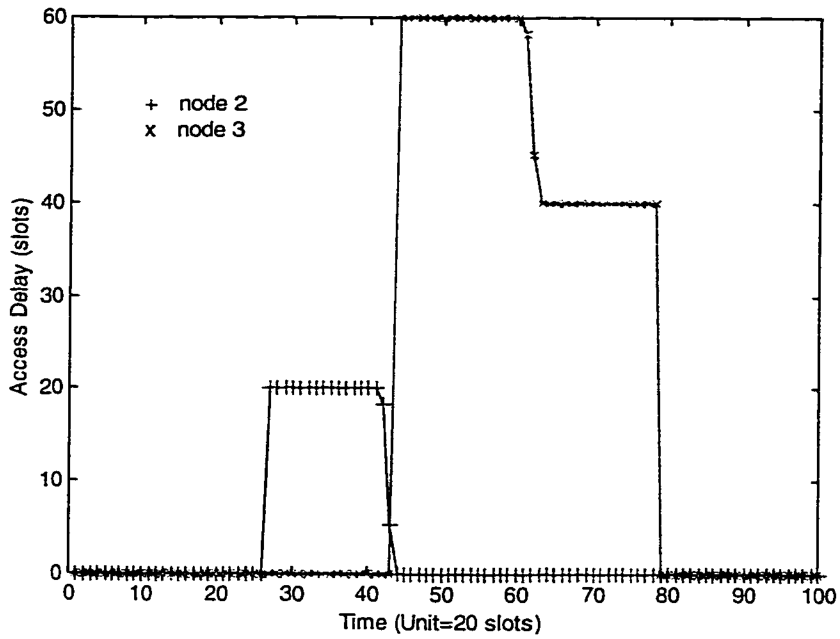
(b) The throughput of two CO service nodes

Figure 10. The access delay and throughput of CO service nodes without usage of RC and CD (Node 2 starts first and ends last.).

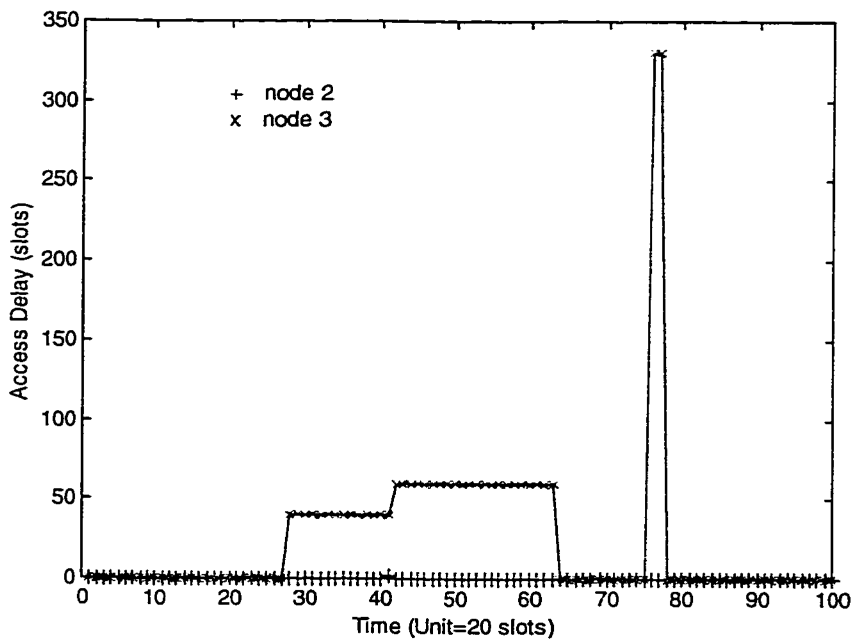
60 slots while node 2 will not experience any access delay. The access delay of node 3 returns to 40 slots only after node 2 stops sending. Similar situation will happen if we change the start and stop times of these two CO service nodes. It is not avoidable that upstream CO service nodes occupy the slots originally for downstream CO service nodes under overload situation. Figure 11 indicates all the other situations that the violations happen. Only the access delay diagram is given.

The reason is that CO service node sets request for each segment it wants to send and multiple requests are allowed. Under overload situation, only the number of free slots which equals to the number of CO service requests will pass downstream. Thus, the upstream CO service node has chances to use the slots originally supplied to downstream CO node because the upstream CO node does not know there are downstream CO service nodes. Seeing figure 10(b), the throughput of node 2 increase when node 3 becomes active. That indicates the node 2 uses the free slot for the requests of node 3 until all queued segments of node 2 with requests set access the network. A throughput pulse of node 3 happens when node 2 ends its transmission. Thus, some segments of downstream CO service nodes may be blocked for a very long time. The longer the upstream CO nodes work, the longer the block time will be. This sometimes will let the segments and whole packet nonsense because CO service mainly supports the application required high QoS. This shortcoming is not tolerable for CO service.

To solve this problem, we suggest adopting RC and CD from QA service to realize a distributed queue for CO service. Then each CO node will record the downstream CO service requests and let certain number of free slots passing by to satisfy the requirement of downstream CO node before its segment is sent according to its local CO service queue. As CO node can send multiple request, for each segment with request sending a CD is needed to hold the number of downstream CO request sent before it. Therefore, an array of CD is required. The number of CD needed depends on how much bandwidth the CO node requires and how far the

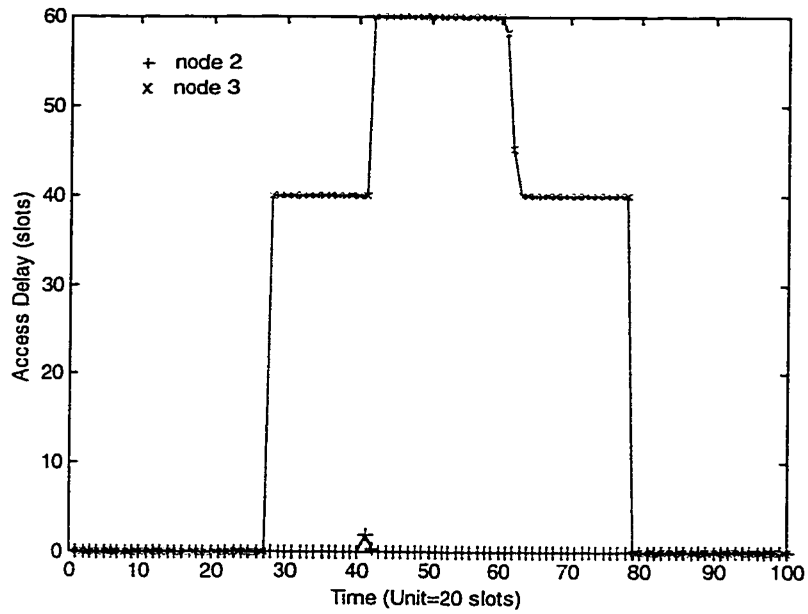


(a) Node 2 starts first and ends first.



(b) Node 2 starts last and ends last.

Figure 11. Other violation situations appeared in the two CO service nodes under overloaded condition.



(c) Node 2 starts last and ends first.

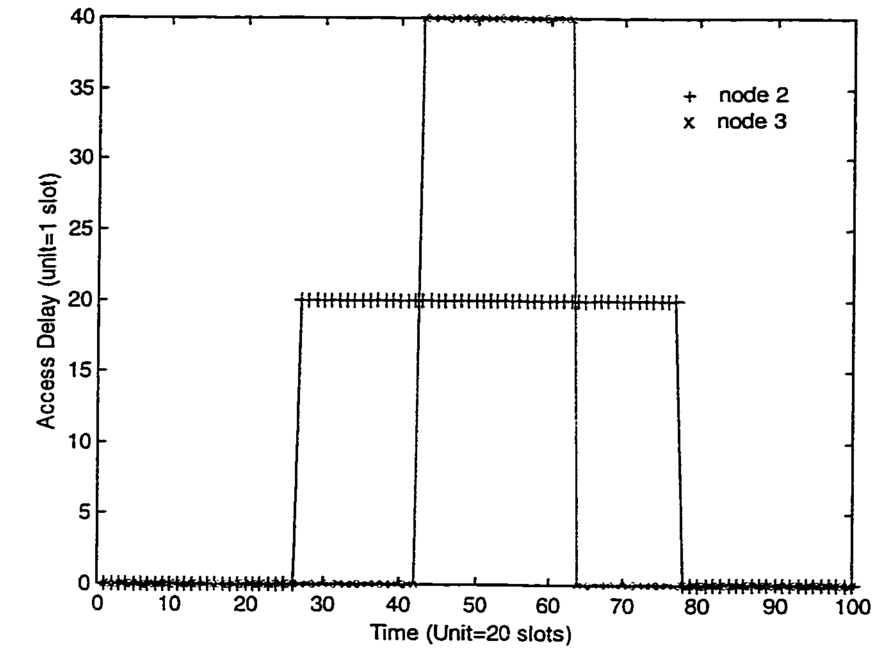
Figure 11. Other violation situations appeared in the two CO service nodes under overloaded condition.

distance between the CO node and the most upstream node is. This value (n_{CD}) is bounded by the following formula:

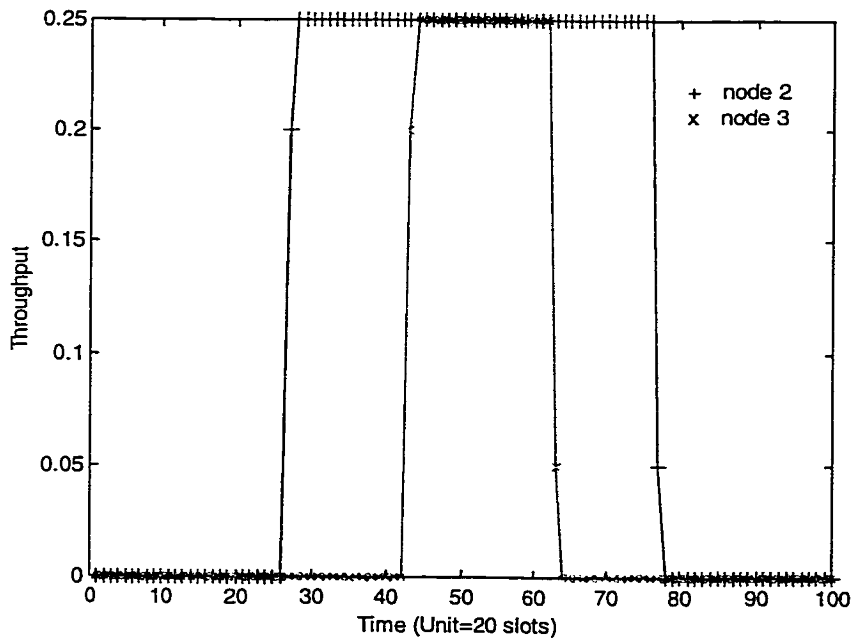
$$n_{CD} \leq \lceil 2d_i \bullet B_i / B_t \rceil \tag{3}$$

where d_i is the distance between CO node i and the most upstream node, B_i is the required bandwidth of CO node i and B_t is the total bandwidth of DQDB network

Fig. 12 gives the results for same configuration as Fig. 10 with RC and CD array for each CO node. Due to RC recording the CO requests of node 3, node 2 will not occupy the free slots belonging to node 3 and let free slots passing for the downstream CO segment with request sent earlier than its own segment. Thus, no

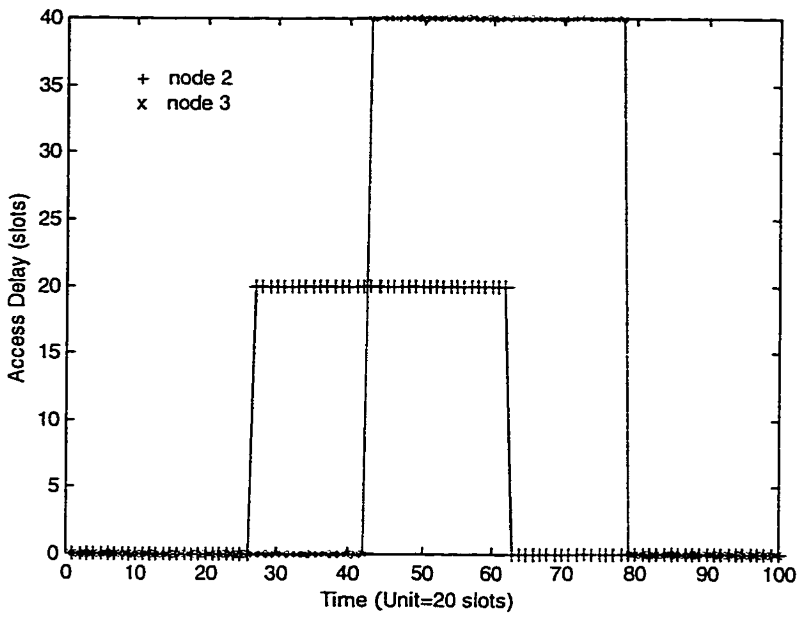


(a) The overall access delay of two CO service nodes

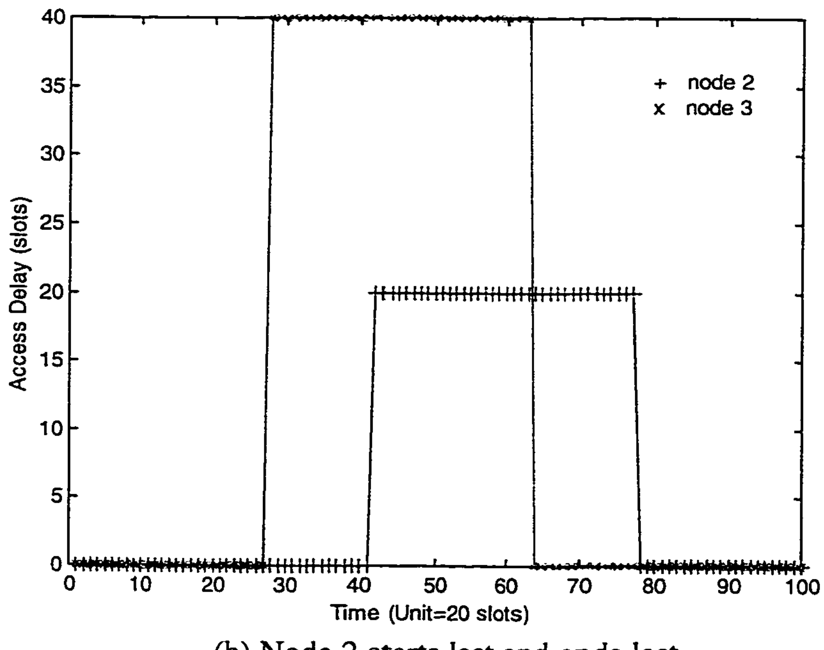


(b) The throughput of two CO service nodes

Figure 12. The access delay and throughput of CO service nodes with usage of RC and CD (Node 2 starts first and ends last.).

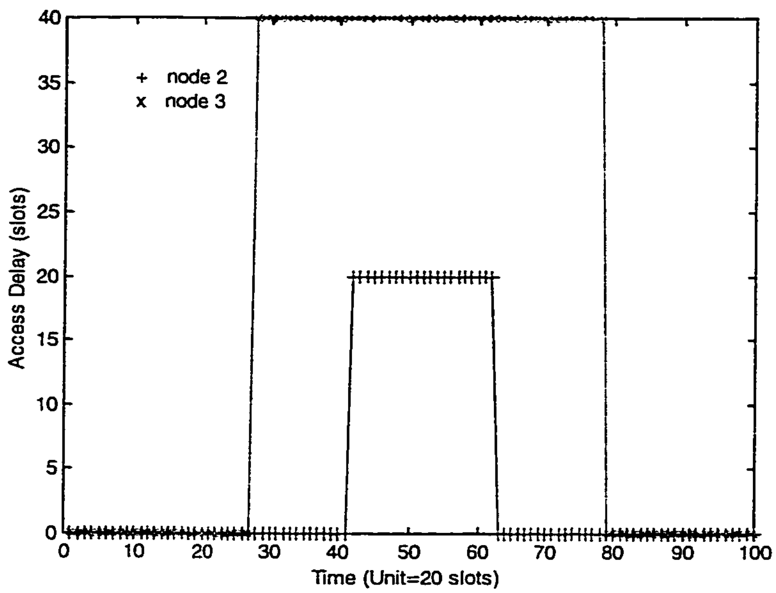


(a) Node 2 starts first and ends first.



(b) Node 2 starts last and ends last.

Figure 13. Recover of other violation situations appeared in the two CO service nodes under overloaded condition.



(c) Node 2 starts last and ends first.

Figure 13. Recover of other violation situations appeared in the two CO service nodes under overloaded condition.

matter which CO node starts or ends first, the access delays of node 2 and 3 are always 20 and 40 slots. No segments of downstream CO service will be blocked. Also, the other violation situations are recovered from the usage of RC and CD array, as shown in Fig. 13. The throughput of both CO service node is constant and node 2 do not use the free slots of node 3's requests when node 3 becomes active.

Therefore, in order to ensure the proper operation of CO service, especially under overload situation, it is necessary to use RC and CD array. Even in light load situation, it may be helpful to control CO service with RC and CD array.

3.3 CO and QA services

One of the first issues of interest is to make possible the integration of the CO service into the DQDB protocol architecture. Since the CO service makes use of QA slots, it is reasonable to evaluate how the CO service may affect the performance of the QA service and vice versa.

Due to the fact that the CO service operates at a higher priority level than the QA service, it is obvious that the requests upcoming from the downstream QA nodes will not have any effect on the number of QA slots that the CO service may get to use. In other words, even though a QA node may have placed a request by setting the request bit in a slot passing by on the request bus, the CO can still issue its request on the same QA slot by using the high-priority request bits reserved for this purpose in the ACF fields of the slot. Furthermore, the CO node will get to see the data slots before the QA nodes located downstream. On the contrary, the CO nodes may face some problems getting the required bandwidth when competing for bandwidth with one or more QA nodes located upstream. In this case, the QA nodes have somehow to be made aware of the presence of CO nodes. According to the IEEE 802.6 standard, a QA service node requires to take into account all the requests before transmitting a segment and let all outstanding requests be satisfied before transmitting its own QA segment. However, in practice only the requests having been sent and having reached the QA node will be served in the proper sequence. This constraint has a direct impact on the quality of service that a CO node may get.

In order to evaluate the impact on the QoS on the CO service due to the presence of QA nodes at the upstream end of the data bus, we have conducted the following study. Consider a system consisting of two nodes, a QA node is placed at the upstream end of the data bus while the second node implements the CO service. Under this scenario, at the beginning of each simulation the node implementing the QA service occupies the whole channel bandwidth, i.e., the node is ready to transmit at every time slot. After a warm-up period, the CO node is activated with a bandwidth requirement set to a fixed value for the whole duration of the simulation. Under these network conditions, the CO node should be able to gain access to the channel at regular intervals as well as to be able to acquire the needed bandwidth share.

Figure 14 depicts the mean access delay for the node implementing the CO service under this scenario (labeled without modification in the figure) for various bandwidth requirements. The mean access delay reported for most bandwidth requirements was found equal to two times the distance between the CO node and the QA node. However, for CO bandwidth requirements equal or greater than 70% of the channel capacity, the reported mean access delay increased as a function of the CO bandwidth requirements. This increase clearly indicates that the CO node is unable to get the bandwidth required. This results mainly by the fact that a QA node can gain access to the bus without having first to place its request. According to the standard, a QA node having a segment ready to transmit can transfer the contents of the RC to CD and proceed to decrement CD. As soon as CD=0, the QA node can transmit the segment even without having to place the request for transmission. In the case of light to mild CO capacity requirements, the number of slots left by the CO service are enough for the QA service to operate without affecting the operation of the CO service. In this case, the time between the instant when the CD is transferred to RC and CD=0 is long enough to allow the number of slots required by the CO node to pass by. However, as the requirements of CO increases, the QA node has a clear advantage to gain access. As soon as the QA has a segment to transmit the contents of RC is transferred to CD and the countdown process starts. In this case, CD is decreased by one only by counting the number of free slot seen passing by on the data bus. Due to the distance between the QA and CO nodes, the requests being issued by the CO nodes after the QA node has set its RC and CD counters are not taken into account. It is clear that the QA node can gain access at a higher rate than it should. In order for the CO node to be able to get the required network capacity, the QA node needs to be made aware of the requirements of the CO node, i.e., of the spare capacity.

In order to evaluate the spare capacity, we propose the use of one of the following two methods described below. The principle of operation of both methods consists in throttling the QA nodes based on the traffic intensity (bandwidth requirements) of the CO nodes.

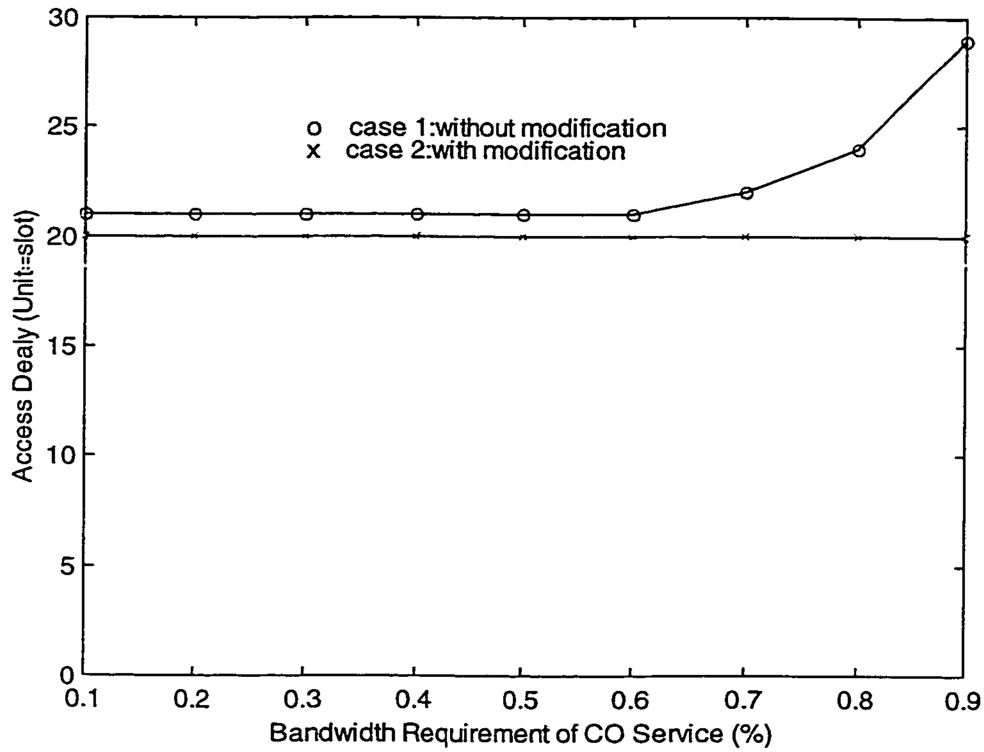


Figure 14. Access Delay of CO service node downstream to a QA service node Vs. required bandwidth.

Method 1: In this first method, we propose a slight modification to the QA service. Since the QA nodes require to be made aware of the needs of the CO nodes, QA

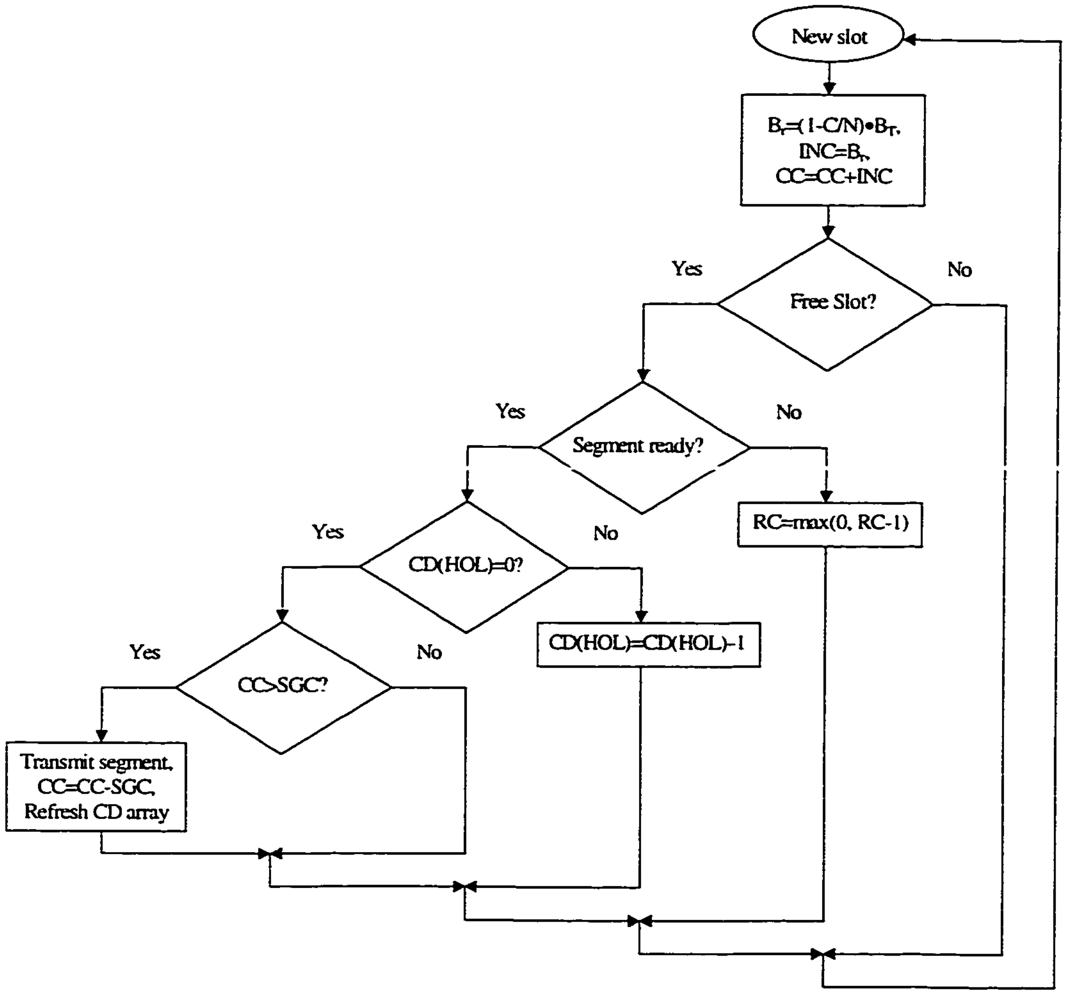


Figure 15. QA service node procedure for data bus – method 2.

nodes need to keep track of the number of CO requests during a certain period. In this way, QA nodes can change their request rates accordingly. Assuming that the length of the observation period is set to N slots and that a QA node has counted C requests of type CO, QA node can determine the amount of available bandwidth to it by:

$$B_r = (1 - C / N) \bullet B_T \tag{4}$$

where B_T is the total channel bandwidth (per bus).

Under this method, a QA node implements a similar procedure to the CO service. However, in this case the SGC of a QA node is fixed to B_T and INC is set to Br, which varies in a slot by slot basis according to (4). In the same way that for the CO service, a QA node makes use of a credit counter, CC. A QA node will be allowed to issue a request provided that the node has a segment ready to transmit, the CD counter has reached zero and it has accumulated enough credits (CC>SGC). Each time the node gets to transmit a segment, it decreases CC by SGC and updates its CD array. Figure 15 depicts the procedure of QA service node with reference to the data bus.

Method 2: Method 1 requires each QA node to keep track of the number of CO data slots transmitted during a time period of length N. In order to achieve this, each QA nodes must be provided with an additional counter. Furthermore, setting up the length of the observation period may depend on the system configuration, e.g., cable length and distance between nodes. Method 2 provides a more effective way for a QA service node to determine the bandwidth requirement of the CO service and requires little change to the specifications of the QA service as specified in the IEEE 802.6 standard.

According to the QA service specifications, once the value of RC moves to CD the QA node will move into the countdown state. From there on and as long as the QA node has not sent its request, the RC counter will be incremented by one for each new request seen in the request bus, as the new coming requests will be. The CD counter is decreased by one for each free slot passing on the data bus. In other words, the QA service specifications have been defined without taking into account the requirements of the CO service. It is therefore required to modify the QA service to take into account the requirements of the CO service. To overcome this problem, the method presented herein is based on the principle that the CO request should be allowed to be served before all outstanding QA requests. This can be easily implemented by modifying the rules governing the CD and RC counters. Since all the required information to keep in sequence the channel access requests is embedded in these two counters. In this case, the QA node does not require to estimate the bandwidth requirement of downstream CO service.

Figure 16 depicts the flow chart of the proposed method. A node implementing the QA service should take into account both the low and high priority requests passing by on the reverse bus. Furthermore, if the QA node sees a high-priority request and it finds itself in the countdown state, it should increment the CD counter by one. By adding one to the CD counter, the node is practically moving back on the distributed queue kept across the MAN.

Both methods can ensure the QoS required by the CO service as shown in Fig. 14 (case 2: with modification). Under the worst scenario, even if an upstream QA node attempts to fully occupy the channel bandwidth, the CO node is able to get its share. Under these conditions, the length of the period required for a CO node to acquire its share does not exceed the round-trip time from the upstream QA node fully using the bandwidth and the CO node. In the case that the QA node only makes use of part of the total bandwidth, the CO service can acquire the required bandwidth in a shorter period of time. More details will be mentioned in following section.

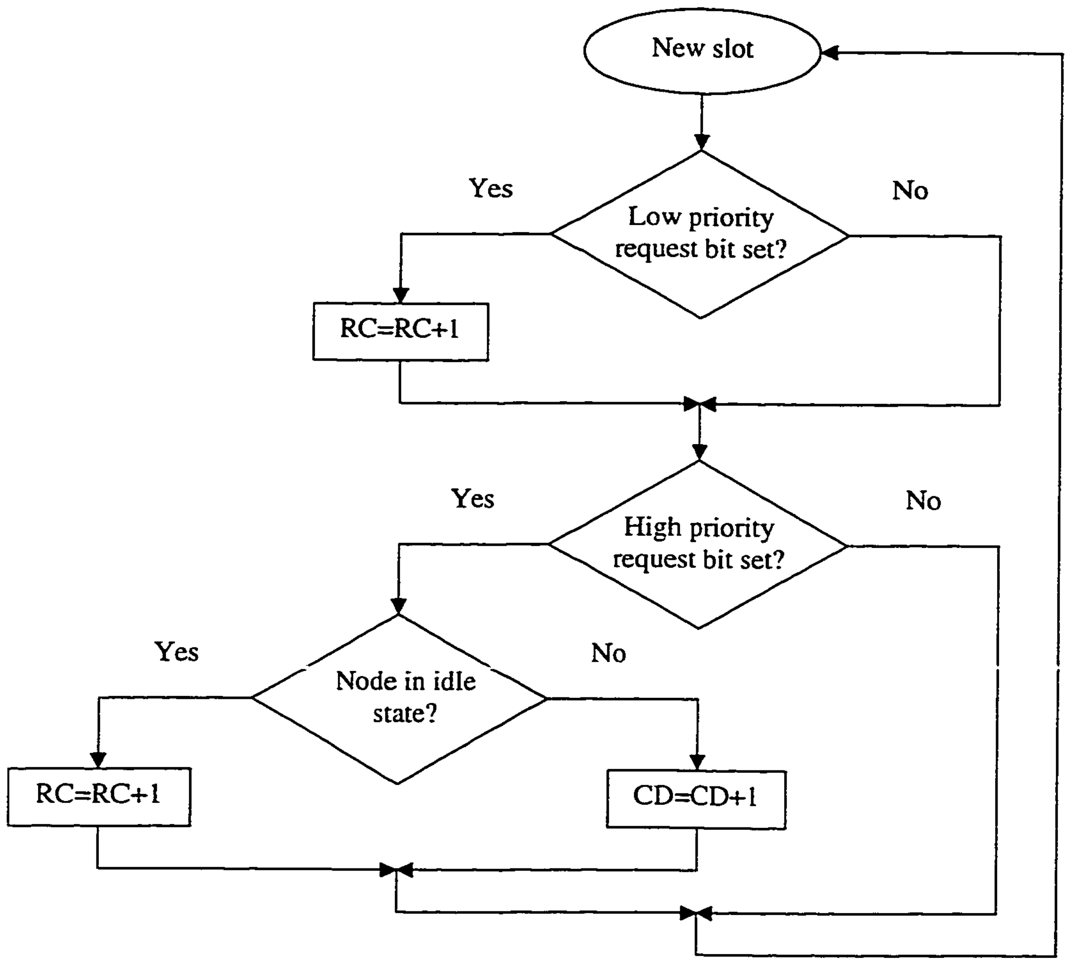


Figure 16. QA service node procedure for data bus – method 2.

From here on, we only consider the use of the second method 2. This is due to the fact that this second method requires only a simple modification to the QA service as specified in the IEEE 802.6 standard as well as to its simplicity of implementation.

3.4 Access mechanism and flow chart of CO service

Upon points mentioned above, we could give the whole procedure of proper CO data service that can guarantee required bandwidth with access delay limited. As CO data service mainly supports traffic as video transmission, real time control and so on, it should operate in high priority to ensure their QoS. In order to ensure that CO data can access media prior to any other data, QA service should be modified to be suit for this priority order then supports CO data service to get access as soon as possible. That is the reason for using different priority request level and trying to let QA service know the bandwidth requirement of CO service.

Figure 17 and 18 depict the flow charts of the CO service procedures to access the data bus and set request on reverse bus respectively. Unlike the QA service, a CO node is required to send a request for each segment to transmit and can send multiple requests according to its mechanism. The QA service can send its segment after CD is set by the value of RC, even though the request of this segment is not sent yet, i.e., before CD decreases to zero. But CO service must send request first then this segment with request sent can access to the network in next free slot after its CD goes to zero. This request is important to upstream CO and QA service nodes to amount the requirement of this CO node. Thus, those upstream QA and CO nodes can do correspondent behavior to record this request and let free slot passing down to serve it. CO service sends its requests only depending on the accumulated credit and if the request bit of the slot on reverse bus is free. So multiple requests will be sent and multiple segments with request sent will wait in the queue for service.

On reverse bus (Figure 17): For each coming slot on the reverse bus (Request Bus), the node increases its credit first (add INC to CC). If accumulated credit exceeds the maximum credit limit (CRM), CC is limited to the value of CRM. The purpose of limiting CC to CRM is to prevent CC from accumulating too much credit during idle period of CO service. Otherwise, the CO node may occupy too much bandwidth for a long time after it becomes active other than it should be. If the high priority request bit of this slot is set by one downstream node, request counter increases by one. If not, the node will check whether CC is greater than SGC. If CC $\geq$ SGC, this means that the node has enough credit to send a request for its foremost segment of those waiting in the queue and without having request sent. After sending a request, CC decreases by the value of SGC, a new element is added to CD array with its value equal to RC and RC is set to 0. And a new segment is added to the list of segments that have already set a request and wait to be transmitted.

On data bus (Figure 18): For the data bus, for each coming slot, if the slot is occupied, the node does not do any things. If it is a free slot, the node does different things depending on which state it is. If the node is idle, RC is decremented by one. Otherwise, if the first element of the CD array (CD(0)) is greater than zero, CD(0) is decreased by one. If CD(0) is equal to zero, the node will use this slot and transmits the segment associated to the request, i.e., the one for which CD(0) is equal to 0. As the segment is sent, the CD array is updated.

From these two flow charts, we find that the node sends its segments just by the way as same as QA service (The procedure on the data bus is similar to the procedure in QA service.). The additional thing that CO service needs to do is the node has to refresh its CD array after sending its segment. Due to the first element of CD array is zero and meaningless, node should remove it from CD array and rearrange the index of CD array. Then the next element of CD array is began to

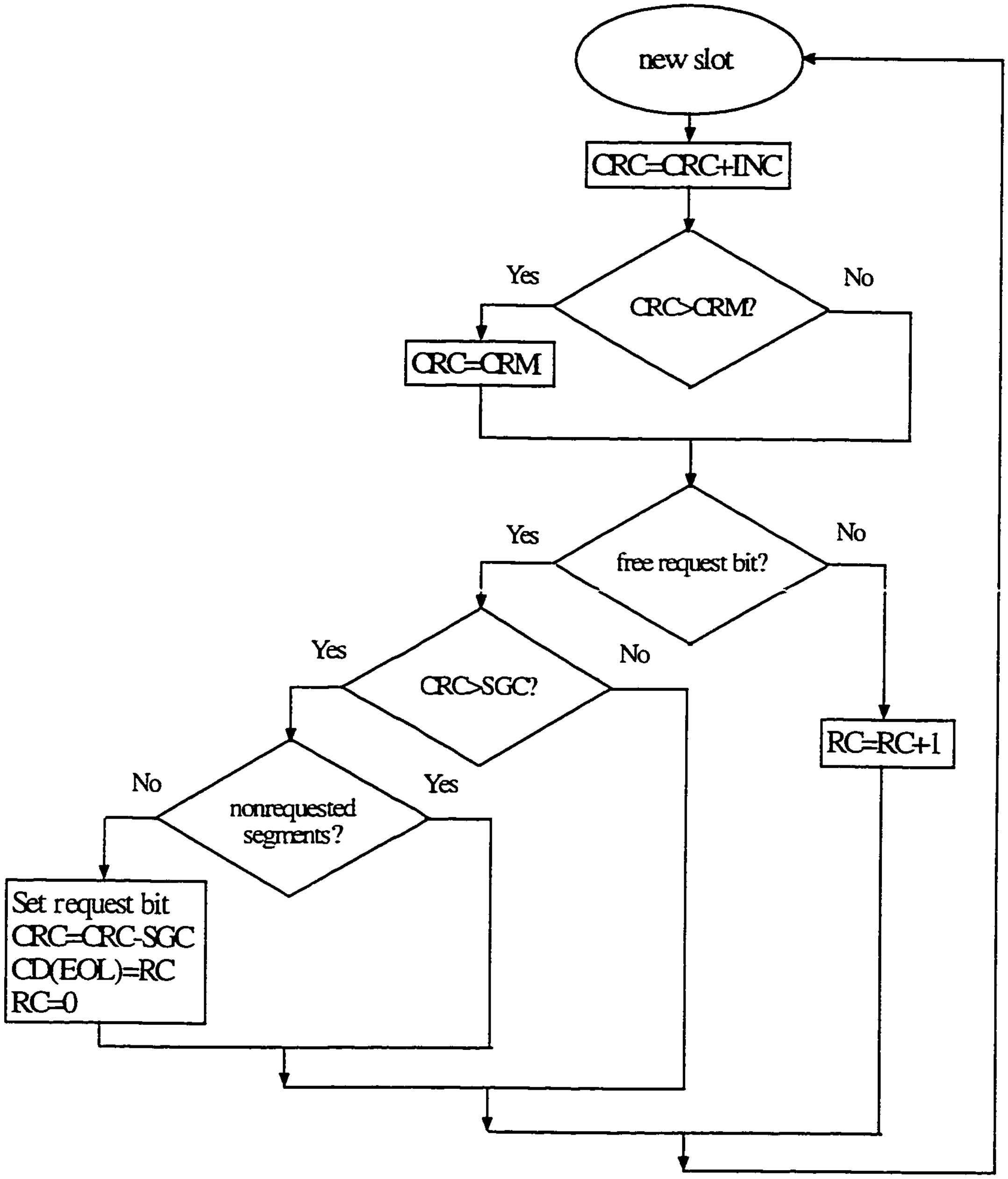


Figure 17. Access mechanism of CO service to the reverse bus (request bus).

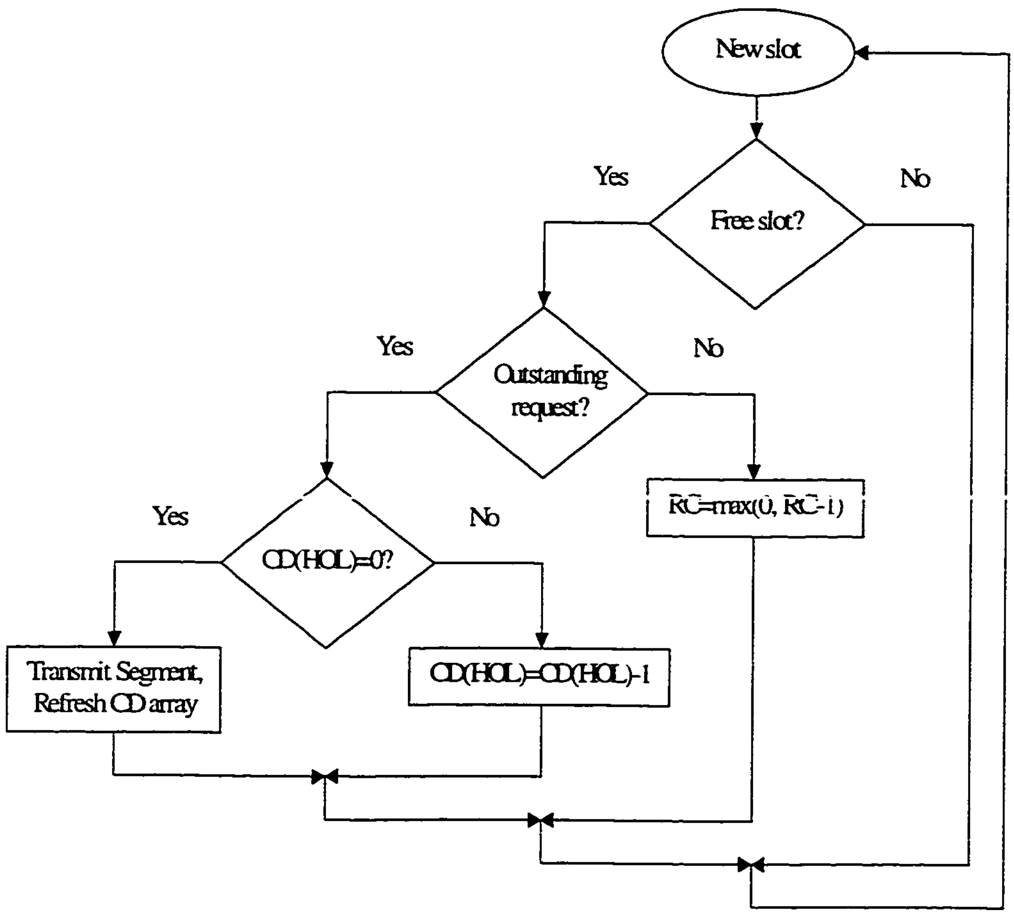


Figure 18. Access mechanism of CO service to data bus.

decrease if there is one. Only those segments for which a request has been sent can actually be transmitted.

3.5 Dynamic setting of CRM

So far, we have analyzed the performance of the GBW protocol used by the CO service and proposed methods to make the QA and CO services coexist. Furthermore, we have also suggested using an array of RC and CD counters to implement more effectively the CO service. In this section, we propose the use of a method to set up the service parameters as required to guarantee the QoS requirements of various applications.

Assume a DQDB network with the most upstream node (node 1) implementing the QA service and two other nodes (node 2 and 3) implementing the CO service and requiring 25% and 40% of the total bandwidth, respectively. The distance between two consecutive nodes is set to 10 slots. Node 2 is upstream to node 3 and becomes active before node 3 does. Based on equation (2), we fix the value of CRM of node 2 and 3 to 156, 186 respectively. According to equation (2), these values of CRM should work out to guarantee the bandwidth requirements of the node. At the beginning of the simulation, the QA service node (node 1) takes the whole channel capacity, then node 2 becomes active after 400 slots time and node 3 becomes active 400 slots time later. As seen from Fig. 19, before node 3 starts its transmission, node 2 is able to get 25% of the total bandwidth. As node 3 becomes active, the bandwidth of node 2 decreases to 20% instead of keeping to 25% as expected. Similar results are obtained even in the absence of QA service node (node 1). And when switching the bandwidth requirement of node 2 and 3, the result shows that after node 3 becomes active the throughput of node 2 decreases to 37.5% from 40%. Several similar situations were detected during our simulations. When the network becomes larger and more CO service nodes send their segments at the same time, this problem tends to happen more often.

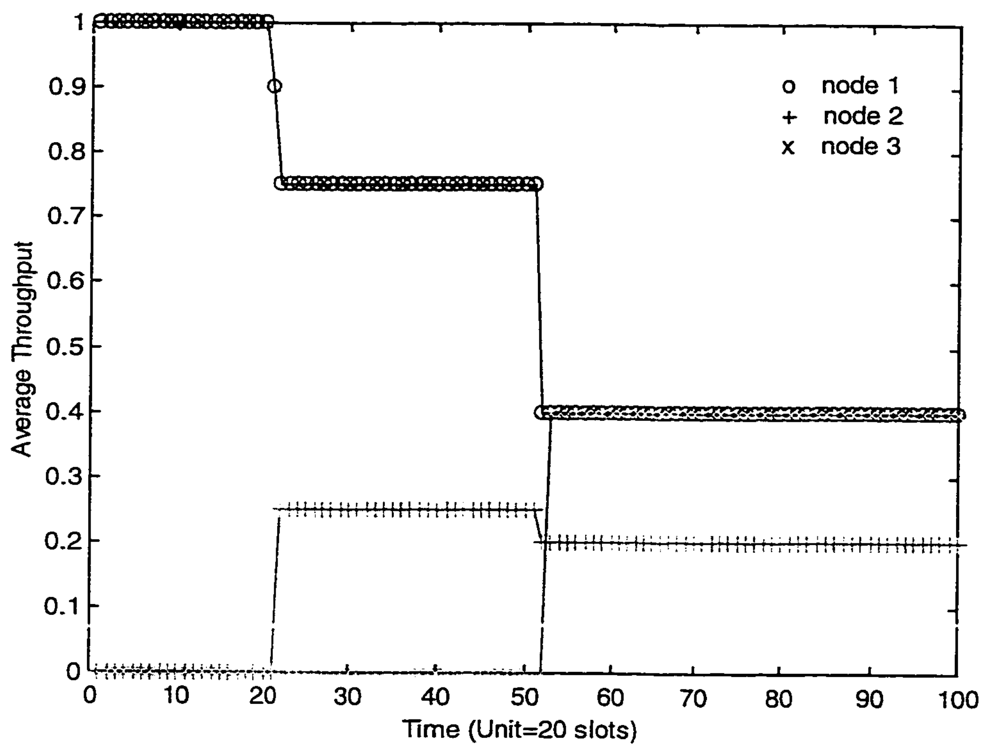


Figure 19. The average throughput (gotten bandwidth) of CO and QA service nodes without using dynamic adjustment (node 2: 25%, node 3: 40%).

The interference among these CO service nodes can be explained as follows. According to [52], a node making use of the CO service requires to place a request for every segment to transmit. Under some traffic conditions, a node may have to delay its request due to the fact that other nodes may have already set the request bit of the passing slot. Consider the following scenario (depicted in Fig. 20), which is based on the above example. Assume a sequence of slots and that node 3 has already set requests in slots 1, 4, 6, 9 and 11 according to CO service for its 40% of total bandwidth (we can get these index numbers through analyzing the CC with INC and SGC). Node 2 plans to set slots 1, 5 and 9 to request 25% of the network bandwidth. However, since node 3 has used slot 1, node 2 has to delay its request by one slot to set request in slot 2. As CRM limits the credit that node can accumulate, node 2 also has to delay its following request by one slot to slot 6, 10, and 14. Again, as slot 6 has been set by node 3, node 2 has to delay one slot to set slot 7, 11 and 15. Node 2 will find again that node 3 has also set slot 11 and its request will be delayed once more. Finally, node 2 can only set slots 2, 7, 12, 17 and so on through this procedure. Under this situation, the actual bandwidth node 2 can get is 20% of the total bandwidth (one slot in every five slots).

As the downstream CO service nodes have advantage to set requests on reverse bus, they shrink the space for upstream CO service node to access the network. In order to overcome this problem, nodes upstream should be allowed to transmit more often. In order to be able to do so, two possible methods can be considered:

1) to increase INC or

2) *to change CRM.*

The purpose to change CRM is to let node accumulate more credit so that it will have more chance to access network without changing its bandwidth requirement. If we fix the CRM according to equation (2), a node can set a request only at particular time slots. If there is a delay due to conflict with downstream node's requests, the node has to delay all the requests to follow. Thus, by increasing CRM dynamically when necessary, the node will not delay all following requests when forced to delay one request.

The purpose of increasing INC is to raise the bandwidth available to the CO service node. So changing CRM seems more feasible since it does not change the total CO service requirements of the network.

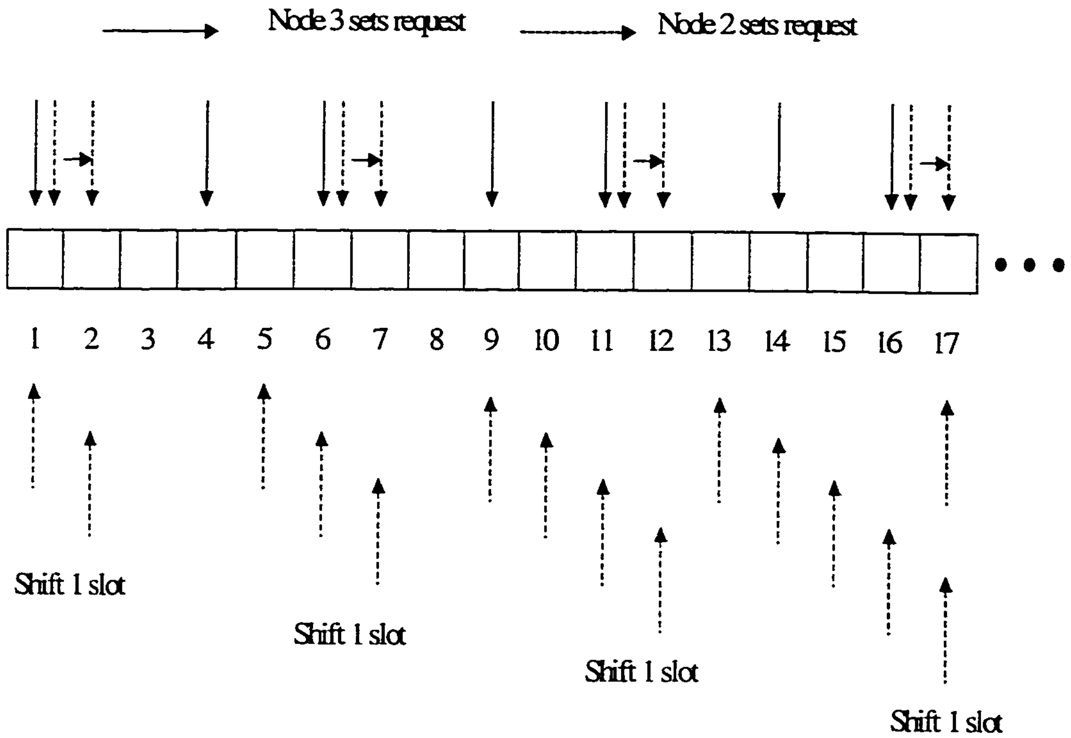


Figure 20. The illustration of bandwidth depression of upstream CO service node.

The value of CRM should be changed dynamically allowing the node to adapt to the loading conditions of the network. In other words, CRM has to be changed to allow the node to compensate for the delays encountered, but without affecting the performance of other users. The main objective is to be able to accommodate the requirements of all active nodes as much as possible. Equation (2) establishes the baseline on how to setup CRM. However in order to be able to compensate for delays encountered, a node should be able to accumulate more credits while waiting to gain access to the network. For the case of two consecutive transmissions, this principle can be simply stated as follows:

$$\text{CRM} - \text{SGC} \geq \text{SGC} - (\lceil \text{SGC/INC} \rceil - i) \bullet \text{INC} \tag{5}$$

with $i = 1, 2, 3, \ldots$ and $i \leq \lceil \text{SGC/INC} \rceil$

which can be simply interpreted as follows. After a node has been successfully in sending a request and having paid SGC units to do so, it should be able to compensate for the delay by being allowed to transmit i slots before than normally expected.

The above equation defines the maximum value of the credit counter, CRM, as follows:

$$CRM \geq 2 \bullet SGC - (\lceil SGC/INC \rceil - i) \bullet INC \qquad (6)$$

with $i = 1, 2, 3, \ldots$ and $i \leq \lceil SGC/INC \rceil$

This principle will be used to illustrate the method to dynamically adjust the value of CRM.

Another important issue is how to decide if the node has got enough bandwidth to satisfy its requirement. Two methods can be considered to perform this task:

- to check the throughput of the node, or
- to check the number of segments waiting in the queue without request sent (or buffer size)

To check the buffer size is ambiguous because the buffer size depends not only on the bandwidth requirement but also the traffic type and other traffic characteristics. It is difficult to decide if the requirement is satisfied at which buffer size. To check the throughput will be more feasible. A margin ε of accuracy can be defined to determine the level of accuracy. For a given bandwidth requirement, if the difference between the actual throughput and the node's requirement is greater than ε, the node is said to have problems to get its requirement fulfilled and its CRM should be increased. Otherwise, the CRM can be decreased or nothing will be changed depending on the value of CRM. The period of measurement depends on the value of

ε. The period should be long enough so that the accuracy of measured throughput is smaller than ε.

We illustrate the proposed solution using a simple example. If a CO node requires 25% of bandwidth, it will set one request in every four slots. However, the node may be required to delay its requests when some other CO nodes downstream have already set the request bits of the corresponding slots. When other CO nodes downstream are also setting, the following way provide a basic idea to set down this problem of delay:

> *As a CO node with a 25% of bandwidth requirement must wait for four slots passing by on reverse bus before it is allowed to send a request, the delay of a request results on the delay of the requests to follow. If we increase CRM of this node to another value, that can let the node send the following request only after three slots passing on the reverse bus, the node will be able to compensate the last delay. Thus, if the new CRM is increased to 2• SGC-3• INC (=5• INC because originally SGC=4• INC in this example), after sending a request the CC will decrease to CRM-SGC=5• INC-4• INC=INC. This idea is based on the following inequality:*
>
> $$\mathrm{CRM}-\mathrm{SGC}+(\mathrm{SGC}/\mathrm{INC}-1)\bullet\mathrm{INC} = \mathrm{CRM}-\mathrm{SGC}+3\bullet\mathrm{INC} \geq \mathrm{SGC} \qquad (7)$$
>
> *After rearrangement, (7) becomes*
>
> $$\mathrm{CRM} \geq 2\bullet\mathrm{SGC}-3\bullet\mathrm{INC} = 5\bullet\mathrm{INC} \qquad (8)$$
>
> *Then the CO node can send next request only after three slots passing by instead of four. The delay incurred in the last request is canceled by this advance.*

Figure 21 depicts the result of increasing CRM using the above method. Under the same situation as figure 19, increasing the CRM to 5•INC can properly solve the problem in allowing to acquiring 25% of the bandwidth. Any value of CRM below this one will not make any improvement. For generality, we should give a procedure taking into account different bandwidth requirements on how to change CRM compatible with the GBW protocol.

For CO service with any bandwidth requirement, INC is set equal to bandwidth requirement in transmission rate if SGC is set to the value of total transmission bandwidth. So in theoretical point-view, one slot in every SGC/INC slots will be used by this CO service. As CRM is firstly set to $\lceil SGC/INC \rceil \bullet INC$ according to [52], we can add transmission chance to CO service on the following (for a detailed analysis, see the appendix A):

If $INC < 2 \bullet SGC/3$,

The new CRM will obey

$$CRM - SGC + (\lceil SGC/INC \rceil - 1) \bullet INC \geq SGC, \tag{9}$$

That is

$$CRM \geq 2 \bullet SGC - (\lceil SGC/INC \rceil - 1) \bullet INC. \tag{10}$$

If $INC \geq 2 \bullet SGC/3$

The (6) is not suit because the old CRM is greater than the new one if we apply equality. And $\lceil SGC/INC \rceil = 2$ in this condition. We can crease CRM to

$$CRM - SGC + (\lceil SGC/INC \rceil - 2) \bullet INC \geq SGC \tag{11}$$

That is

$$CRM \geq 2 \bullet SGC \tag{12}$$

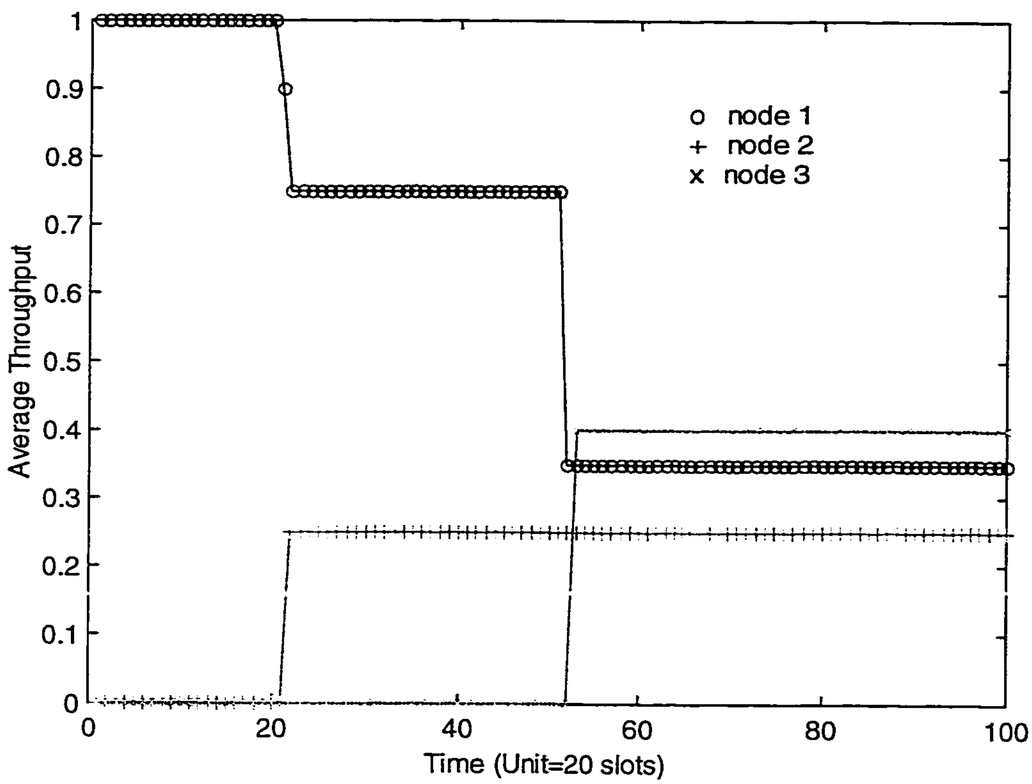


Figure 21. The average throughput (gotten bandwidth) of CO and QA service nodes with using dynamic adjustment (node 2: 25%, node 3: 40%).

Under some conditions, the CO service has to delay its request more than two slots. So it is not enough to compensate just one slot. Figure 22 illustrates this situation. In this instance, there are two CO service nodes, the upstream node requires 80% of total bandwidth and downstream node requires 16.67% of total bandwidth (1/6 of total transmission bandwidth). The CRM, and INC of the upstream CO node are 248 and 124 (SGC is 155 according to Equation (2)). After the downstream node becomes active, the upstream node can only reach 60%-70% of total bandwidth (Fig. 22 (a)). If the CRM of upstream node is increased to 310 following (12), the throughput of the upstream node is around 70%-80% (Fig. 22 (b)). Although the throughput has been improved, there still is difference between the required and acquired bandwidth. That means the upstream CO node has been unable to compensate a delay and CRM should be increased to a higher value to provide more access chances. As $\lceil SGC/INC \rceil$ is equal to 2 and the CRM has reached 2•SGC, we have let the accumulated credit to be greater than 2•SGC after a request is set and 2 slots have passed on the reverse bus. That is

$$CRM - SGC + \lceil SGC/INC \rceil \bullet INC \geq 2SGC \tag{13}$$

As $CRM - SGC + \lceil SGC/INC \rceil \bullet INC \geq 2SGC$ will generate a new CRM that will not be greater than the old CRM. Thus, we apply in a similar way as before:

$$CRM - SGC + (\lceil SGC/INC \rceil - 1) \bullet INC \geq 2SGC \tag{14}$$

i.e.,

$$CRM \geq 3SGC - (\lceil SGC/INC \rceil - 1) \bullet INC \tag{15}$$

In this example we get a new CRM = 341 through (15). The result in this CRM shows the throughput of node 2 has increased to 79% but still below the required bandwidth. Applying the same methodology and again, we find that the next CRM is 372, which is defined by:

$$CRM - SGC + \lceil SGC/INC \rceil \bullet INC \geq 3SGC \tag{16}$$

because this CRM is smaller than the value from

$$CRM - SGC + (\lceil SGC/INC \rceil - 2) \bullet INC \geq 2SGC \tag{17}$$

This time the acquired throughput reaches to 80% as shown in figure 22 (d). In this case, CRM has changed three times before the node can get the required bandwidth.

We are now in the position of defining a general principle to dynamically adjust the value of CRM according to the network conditions and nodes' requirement:

1. Set i = 0 and j = 1;
2. Set the CRM initially to the value specified by Equation (2), i.e. CRM = $\lceil SGC/INC \rceil \bullet INC$;
3. Count the number of segments sent during ls slots;
4. IF | measured throughput – required capacity | < ε THEN goto 3;
5. IF measured throughput < required throughput THEN

    IF i < $\lceil SGC/INC \rceil$ THEN

        i = i + 1;

    ELSE

        i = 0 and j = j + 1;

  ELSE

    IF i >0 THEN

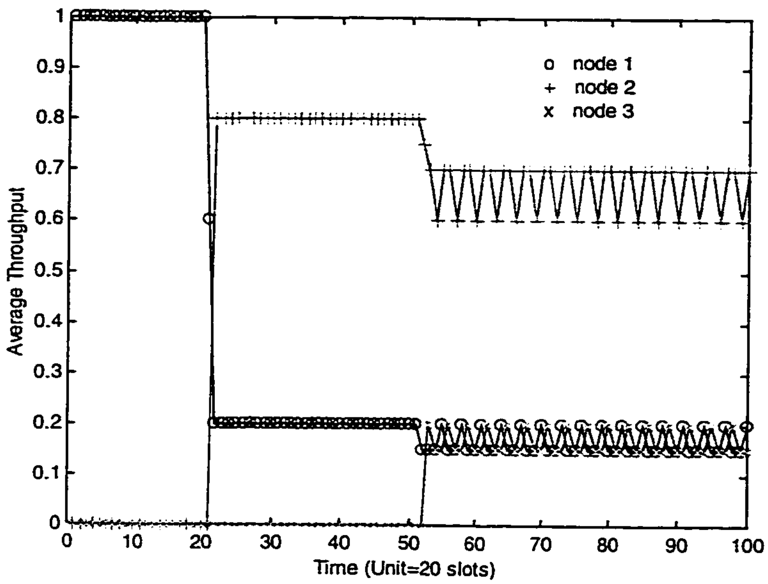
        i = i –1;

    ELSE

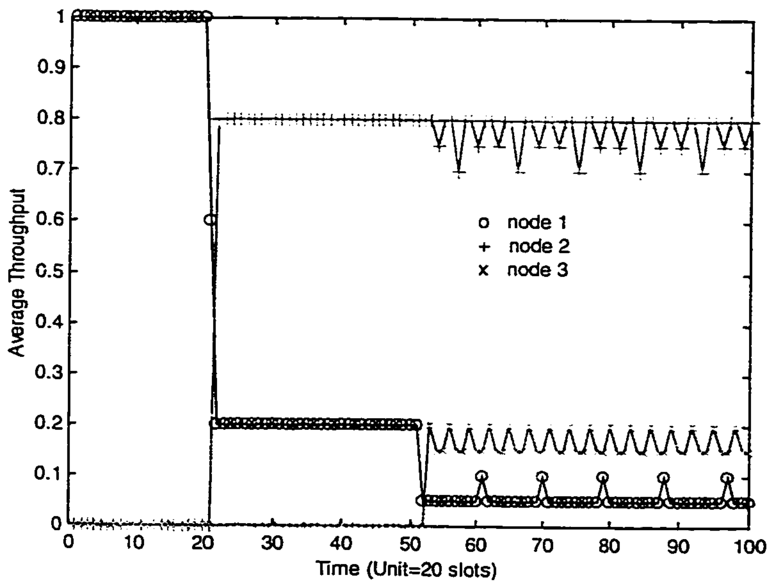
        IF j > 1 THEN

            j = j - 1
6. CRM ≥ $j \bullet SGC - (\lceil SGC/INC \rceil - i) \bullet INC$;
7. Goto 3

Through this procedure, CRM can be set dynamically as a function of the acquired capacity and required bandwidth. Figure 21 shows the throughput of upstream node requiring 25% of total bandwidth when using this procedure with ε = 0.05 and ls = 10 slots and Figure 22(d) gives the throughput of the upstream node requiring 80% of total bandwidth when using this procedure with ε = 0.005 and ls = 40. All these

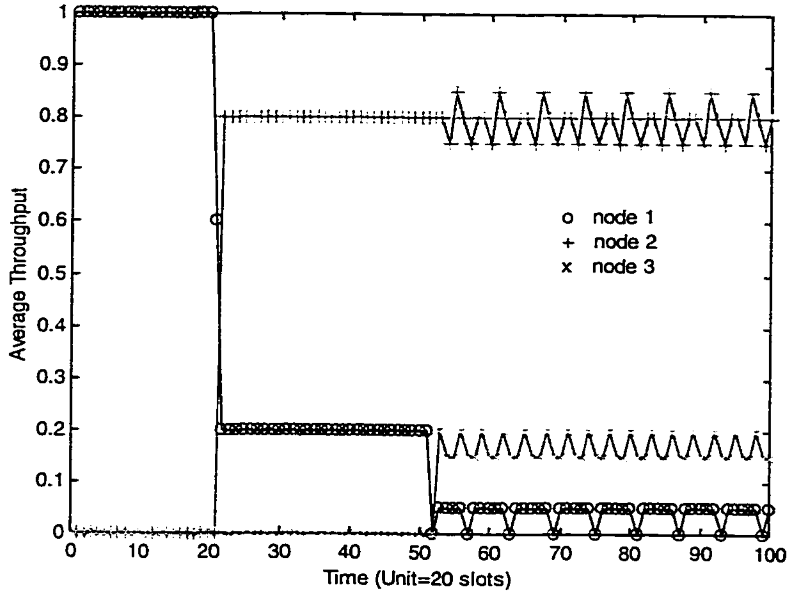


(a) CRM=⌈SGC/INC⌉•INC=248 of node 2

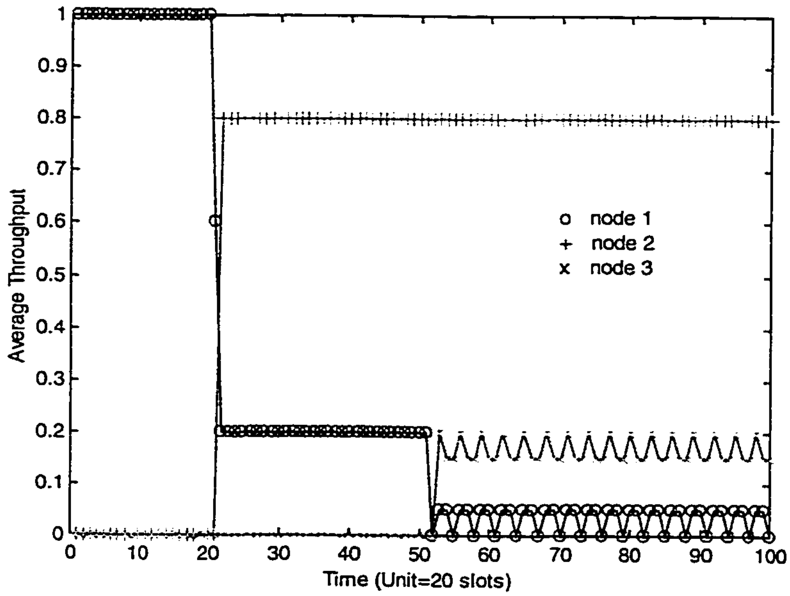


(b) CRM= 2SGC=310 of node 2

Figure 22. The average throughput (gotten bandwidth) of CO and QA service nodes with and without using dynamic adjustment (node 2: 80%, node 3: 16.67%).



(c) CRM=3SGC-INC=314 of node 2



(d) CRM=4SGC-2INC=372 of node 2

Figure 22. The average throughput (gotten bandwidth) of CO and QA service nodes with and without using dynamic adjustment (node 2: 80%, node 3: 16.67%).

results show that CO nodes can readjust their CRM to suit for different network situation and ensure its required bandwidth quickly.

3.6 Access delay analysis

Although CO data service can ensure the bandwidth required by application, another service quality also should be considered: access delay. For some application as video and audio transfer, segments with very long delay can not make any sense to users and will be discarded by the application. Though, we expect to apply CO data service with limited access delay.

According to access mechanism of CO data service, CO service nodes set high priority request bit in reverse bus for its segment waiting in the queue. QA service nodes will record these high priority requests depending on their state. If QA service node is idle, RC will increase by one. If there are segments waiting in the queue of QA service node, CD will increase by one and thus the delay to send its own segment for satisfying the requirement of CO service. For CO service nodes, these high priority request will be recorded by RC and same number of free slots will pass by before CO service nodes send their own segments. Thus, CO service nodes can fetch the bandwidth from QA service nodes when network is overloaded and send segments depending on the order which is indicated by CD and RC of each CO service node based on distributed queueing theory.

Because CO service nodes only record high priority request passing on reverse bus, the worst situation in which the CO service nodes will experience the longest access delay is when there isn't any QA service nodes downstream and all the bandwidth is occupied by other CO service nodes and upstream QA service nodes. The CO service node has to wait until the upstream QA service nodes notice its requests and let free slots pass by to reach it. This access delay depends on the distance between the CO service node and the QA service nodes that will shrink their bandwidth for the requirement of the CO service node. Thus, the access delay of CO service is bounded by the following inequality:

$$T_{access-delay} \leq 2 \bullet T_{1n} \quad (18)$$

where T_{1n} is the time taken by slot traveling from the most upstream node (node 1) to node n (the observed CO service node). The equality exists only in the situation that the most upstream node is the unique QA service node and the network is overloaded. The actual value of access delay of CO service depends on the distance between the CO service node and nearest upstream QA service node and whether this QA service node can supply the required bandwidth. If QA service applies BWB mechanism to solve the unfairness, the access delay of CO service will decrease due to additional free slot left by QA service. But it will make QA service take more time to convergence to steady state. If there is a QA service node downstream to CO service node, the CO service node will occupy the free slots destined for the QA service node because the CO service node can not beware of the existence of downstream QA service node. The access delay of CO service node is reduced in this manner while downstream QA service node will suffer longer access delay.

3.7 Unfairness to QA service node

CO data service provides guaranteed bandwidth for those applications requiring QoS guarantee. Though, there still are QA data services to support general data transfer which only worry about the rightness of transmission. The unfairness happens to these QA service nodes when network is overloaded. Due to different position of each node, upstream nodes tend to occupy more bandwidth than downstream nodes.

When there is CO data service co-existing, the problem of unfairness will be more serious.

Fig. 23 illustrates one situation of this kind of problem. Here, four nodes compete to access network. Node 1 and 3 are QA data service nodes and try to occupy all available bandwidth. Node 2 and 4 are CO data service nodes and are active at time of the 2000th and 4000th slot with requirement of 40% bandwidth. Node 2 stops sending segments around the time of the 7000th slot after node 4 finishes at the time of the 6000th slot. The CO service nodes can get its bandwidth requirement and QoS without any problem. All CO service nodes can access to network and get its required bandwidth with limited access delay. But for QA service nodes, there is a serious unfairness among them. In Fig. 23, node 1 almost fetches all the bandwidth at first and lets node 3 only occupy a small fraction of total bandwidth because node 1 is upstream to node 3. After CO service nodes become active, the throughput of node 3 is shut down to 0 and all rest of bandwidth left by CO service is used by node 1. Until all CO service nodes have ended their transmission, the downstream QA node can retrieve a small part of bandwidth with still existing unfairness.

In the IEEE 802.6 DQDB standard, BWB mechanism has been adopted to solve the unfairness in QA service nodes. Here we apply this technique to check if it still works when CO service exists with QA service. Furthermore, another method called DQDB+/- is introduced to get a better resolution [5].

3.6.1 Bandwidth balancing mechanism

A bandwidth balance mechanism has been introduced in the final draft of the DQDB standard. This mechanism, whose behavior depends on a parameter called the bandwidth balancing modulus (BWBM same as β in (1)), requires each node to leave one usable slot empty, and therefore available to the downstream nodes when it has already used a number of slots equal to the BWBM. In this way a sort of channel of free slots is created on each of the two buses, which permits an effective distributed control on the sharing of the system bandwidth for QA data service nodes.

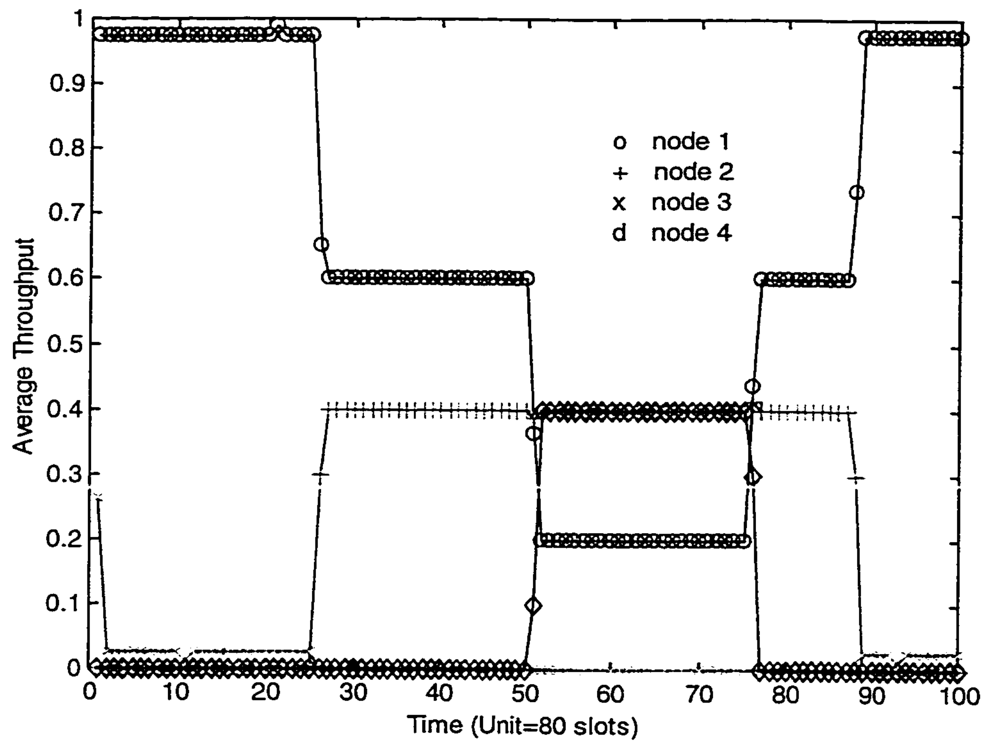


Figure 23. The throughput of CO and QA nodes without applying any method to solve the unfairness of QA nodes (node 2 and 4 are CO service nodes with requirement of 40% bandwidth, node 1 and 3 are QA service nodes trying to use all available bandwidth).

The net effect is that each QA service node is granted at least a minimum amount of bandwidth to be available all the time. This value is called the control rate. When the network is overloaded, the QA service nodes that offer a load under the control rate (the so-called non-limited stations) and CO data service nodes are managed to transmit under all their required traffic rate, whereas the bandwidth that are not used by the non-limited nodes and CO service nodes are divided evenly among the other stations. The cost that must be paid for such an improvement in fairness is a modest loss in the total available bandwidth. It should also be noted that the unfairness observed under low traffic conditions in the distribution of access delay is not solved by the BWB mechanism. This inconvenience is not as serious as the unpredictability of the bandwidth sharing under heavy traffic conditions.

Fig. 24 gives the result of using BWB mechanism to QA service nodes with same configuration as Fig. 23. During all the time, both QA service nodes (node 1 and 3) are sharing the bandwidth left by CO service with same portion. But a portion of bandwidth is wasted due to the BWB mechanism. The wasted bandwidth can be calculated as following [34]:

$$B_w = B_t \bullet \frac{1-s}{1+N\beta} \tag{19}$$

where B_t is total bandwidth, s is the portion in percentage which used by non-limited nodes and CO service nodes, β is the BWBM and N is the number of QA nodes which controlled by BWB mechanism. The BWBM is generally set to 8 because this value can get a good balance between the bandwidth convergence and bandwidth wastage. The smaller the BWBM is, the more bandwidth is wasted and the faster all the QA nodes converge to a steady state in which all QA nodes share same portion of bandwidth, and vice versa.

In the simulation we choose the BWBM as 8 and then the wasted bandwidth is 1/17 of total bandwidth when no CO service exists, 3/85 when one CO service becomes active and 1/85 when both CO service nodes are active. Furthermore, it takes a little bit long time for both QA service nodes to converge to balancing state especially after node 4 starts to send its segment and stops sending. When upstream CO service node (node 2) is active, it fetches the usable slots from both QA nodes. Then these two QA nodes can retain to steady state quickly. After downstream CO node (node 4) becomes active, it mainly fetches usable slots from downstream QA node (node

3). Thus, the balance is destroyed seriously and it takes long time to converge to steady state. Same situation happens after downstream CO node (node 4) stops sending.

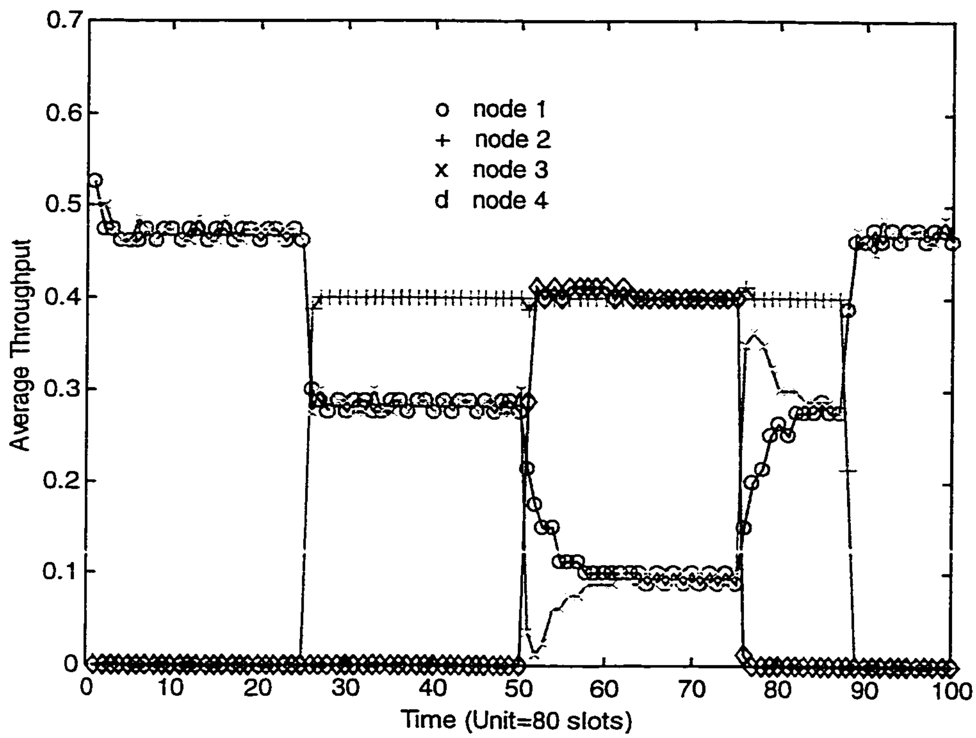


Figure 24. The throughput of CO and QA nodes with applying BWB mechanism to solve the unfairness of QA nodes (node 2 and 4 are CO service nodes with requirement of 40% bandwidth, node 1 and 3 are QA service nodes trying to use all available bandwidth).

So two major problems exist when applying BWB mechanism. The wasted bandwidth is extremely great, especially when there isn't any CO service and non-limited QA service. Also the time to take the QA nodes converging to steady state is too long. This will let downstream QA node suffer longer access delay. Some scheme has been proposed to overcome these two major problems for BWB mechanism. [2] introduces a closed loop bandwidth balancing mechanism to solve the wastage of bandwidth. [3] shows a mechanism to equalize the access delays of the different nodes, irrespective of the node's offered load. [4] proposes a method to adapt the bandwidth allocation according to the offered load.

3.6.2 DQDB +/- mechanism

DQDB+/- is a new mechanism introduced by [5] based on DQDB QA access protocol. This new scheme has several advantages over bandwidth balancing mechanism, such as:

- Better adaptation towards changes of the network load,
- Full utilization of the bus, and
- Better performance at high transmission speeds and/or physically long buses.

In DQDB+/-, QA nodes are partitioned into two sets: the set of overloaded QA nodes and the set of underloaded QA nodes. An underloaded QA node obtains all the bandwidth it needs for transmission. An overloaded QA node cannot satisfy its bandwidth requirements and obtains the same bandwidth as all other overloaded QA nodes like in BWB mechanism.

All the nodes use reverse bus to send reservation requests to upstream nodes. However, only underloaded QA nodes send a reservation request for each segment following the same QA access protocol as before. If an underloaded QA node becomes overload, it stops sending reservation requests and sends a signal (+) on reverse bus to notify the upstream nodes that it is overloaded. Once the signal is set, no more reservations are sent. If an overloaded QA node becomes underloaded, it sends an opposite signal (-) on reverse bus to indicate that it is underloaded. Then the QA node resumes sending reservation requests, one for each segment.

Before a QA node is allowed to transmit a segment it has to consider all reservations from downstream nodes. For each reservation request and for each overloaded QA node downstream the QA node has to leave an empty slot on data bus.

To implementing DQDB+/-, two additional bits are needed in the slot header, referred to as the plus bit and minus bit. Except for busy bit, all other bits are set by a node for sending reservation requests.

An underloaded QA node sends a reservation request by setting a request bit in a slot on reverse bus. One request bit is set for each segment to be transmitted. If an underloaded QA node becomes overloaded, it sets a plus bit in a slot on reverse bus. After setting the plus bit, no more reservation requests are transmitted. If an overloaded QA node becomes underloaded, it sets a minus bit and resumes setting request bits.

Each QA node determines its turn to transmit a segment with four counters, the request counter (RC), the countdown counter (CD), the overload request counter (ORC), and the overload countdown counter (OCD). RC and CD have the same functions as before in DQDB standards. An idle QA node, i.e., a QA node that does not have a segment queued for transmission, increments RC for each passing slot on reverse bus with the request bit set. ORC is incremented for each passing slot on reverse bus with the plus bit set and decremented by one for each slot with the minus bit set. For each empty slot passing by on data bus, the QA node decrements RC by one as long as RC is greater than zero.

If a segment arrives at an idle QA node, the contents of RC and ORC are copied to CD and OCD respectively, and RC is set to zero. The value of ORC remains unchanged. Now, RC is still incremented for each set request read on reverse bus. ORC is still incremented for each set plus bit and decremented for each set minus bit. For each empty slot on data bus, CD is decremented by one. If CD is zero, the QA node decrements OCD by one. If an empty slot is read and both CD and OCD are zero, the empty slot is used for transmission of the segment. If the QA node has more segments waiting for transmission, RC and ORC are copied to CD and OCD, and RC is set to zero.

Each QA node can determine whether it is overloaded or underloaded. Most of the required information is stored in counters RC, CD, ORC, and OCD. Three additional counters are needed. SEG_CTR contains the total number of segments queued for transmission, SLOT_CTR is incremented for each arriving slot on reverse bus, and BSY_CTR is incremented for each busy slot read on data bus. A QA node evaluates its state each time after an interval of basis slots have passed by on reverse bus (SLOT_CTR = basis). Then it calculates

$$QUOTA = \frac{basis - BSY\_CTR - RC - CD}{ORC + 1} \tag{20}$$

and sets counter SLOT_CTR and BSY_CTR to zero. QUOTA provides the local value of a share of the network load, that is, the maximum number of slots a QA node can transmit during a period of basis slots. If SEG_CTR > QUOTA the QA node is overloaded; otherwise, the QA node is underloaded. If a state change has occurred, the QA node takes the appropriate action as described above.

The value for parameter basis can be chosen from a wide range of values without having an effect on the performance of the network. Since the propagation of information in a dual bus architecture is limited by the round-trip delay, we set basis to the sum of the slot lengths of data bus and reverse bus. Then QUOTA denotes the maximum number of segments each QA node can transmit in round-trip delay.

Fig. 25 shows the results of setting up QA nodes by applying DQDB+/- in same network configuration as Fig. 23 and 24. Here QA nodes can equally share the bandwidth left by CO service without any waste of bandwidth. Furthermore, the time it takes to reach steady state is quicker than it with BWB mechanism. In previous simulation, it takes at least 240 slots time (three times of round-trip delay) to reach steady state in which all overloaded QA nodes share the rest bandwidth equally by using BWB mechanism. With applying DQDB+/- and having tried all possible situations that we change the active time of each node, the maximum time to let overloaded QA nodes reach steady state is less than 240 slots time and generally, the convergence time is around 120 slots (two times of round-trip delay). [6] provides some mathematical analysis of basic theorem of DQDB+/-.

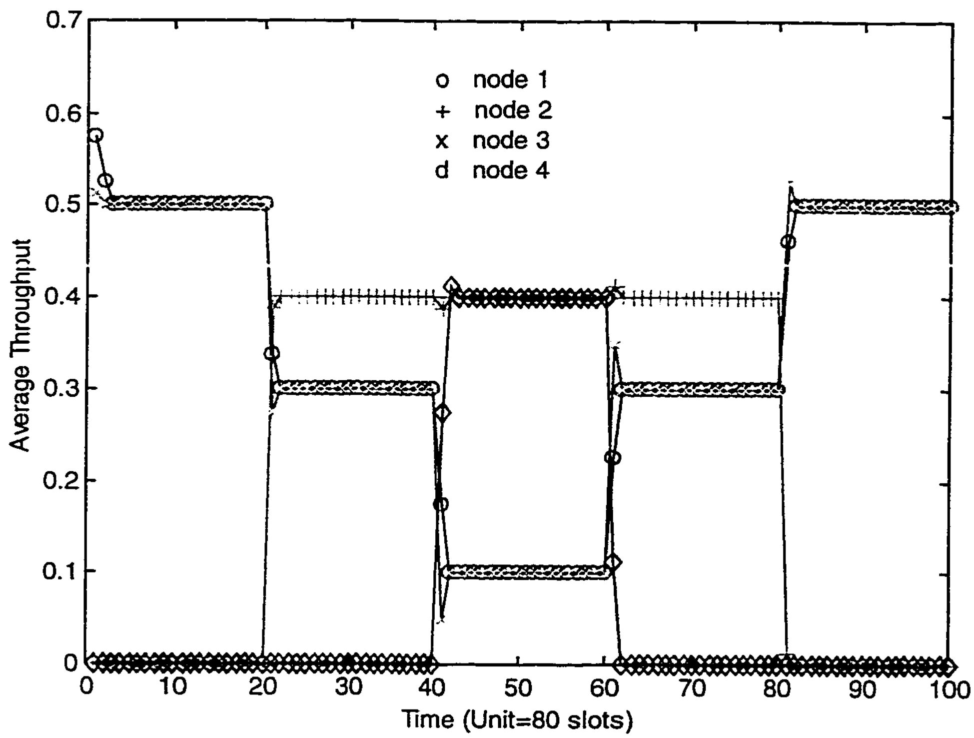


Figure 25. The throughput of CO and QA nodes with applying DQDB+/- mechanism to solve the unfairness of QA nodes (node 2 and 4 are CO service nodes with requirement of 40% bandwidth, node 1 and 3 are QA service nodes trying to use all available bandwidth).

There are also many other ways to solve the unfairness of QA nodes under overloaded situation. [7] provides an access protection solution to limited the access ability under heavy load to solve the unfairness. [8, 9] proposes the slot reuse to improve the efficiency of protocols and then decreases the unfairness under heavy load. In [10, 11, 12, 13], a multiple priority solution is improved to settle the unfairness down.

Chapter 4

Real Traffic Analysis

The purpose of metropolitan area networks is to interconnect local area networks and communication systems dispersed over a city area, providing integrated communication services such as voice, data and video. Nowadays, it is expected that a variety of traffic types will be carried by the DQDB network. Among the various diverse types of traffic, a large number require high bandwidth. Unlike data transmission, the concept of service quality is of vital important in many new services, in other word, different traffic types can have different requirements on the kind of services.

In [14, 15, 16, 17], the concept of ATM is introduced into the DQDB MAN and a pure ATM solution is adopted to support isochronous and non-isochronous services. Although the details in the references are different, data, CBR (constant bit rate) and VBR (variable bit rate) services are supported by the QA access method in a unified manner.

The CO service in QA access method as the most recently proposed service is mainly to provide the convenience to those application such as voice, video and DS1/DS3/E1/E3 circuit emulation services with high quality requirement. These applications need to have higher priority than the other services especially when the network is busy because most of them are time sensitive application.

In this chapter, we undertake the analysis on the stability of the CO service under two traffic models. The concerned issues are throughput of the CO service, access delay and response time of both CO and QA services. For the MPEG-2 traffic, we also consider the time delivery of each video frame. The access delay is defined as the period between the time that a node sends a request and the time at which the segment is sent. The response time includes the time that a segment stays in a node.

4.1 Self-similar Traffic Analysis

In recent years, a new traffic called self-similar traffic has been observed from LAN and WAN [19, 20, 21]. The observed data series have the following common features [22]:

1. Qualitative features of a typical sample path:
   a) There are relatively long periods where the observations tend to stay at a high level, and on the other hand, there are long periods with low levels.
   b) When one looks only at short time periods, then there seem to be cycles or local trends. However, looking at the whole series, there is no apparent persisting trend or cycle. It rather seems that cycles of (almost) all frequencies occur, superimposed and in random sequence.
   c) Overall, the series looks stationary.
2. The variance of the sample mean seems to decay to zero at a slower rate than n^{-1}. In good approximation, the rate is proportional to $n^{-\alpha}$ for some $0 < \alpha < 1$.
3. The sample correlation decay to zero at a rate that is in good approximation proportional to $k^{-\alpha}$ for some $0 < \alpha < 1$.
4. Near the origin, the logarithm of the periodogram plotted against the logarithm of the frequency appears to be randomly scattered around a straight line with negative slope.

The appropriate mathematical results with detailed statistical analyses have been reported in [23, 24, 25, 26]. In the simulation for self-similar traffic, all the self-similar traffic series are adopted from the measured value at Bellcore on August and

October 1989. In the simulated network configured with 10 nodes, the node 1, 4, 6, 9 are QA service nodes while the other four implement the CO service. All ten nodes have self-similar traffic as input stream. The transmission rate of the network is 15 Mbps so that the average bandwidth requirement of ten nodes is a slightly over the network capacity. Thus, the system will experience both overload and underload periods.

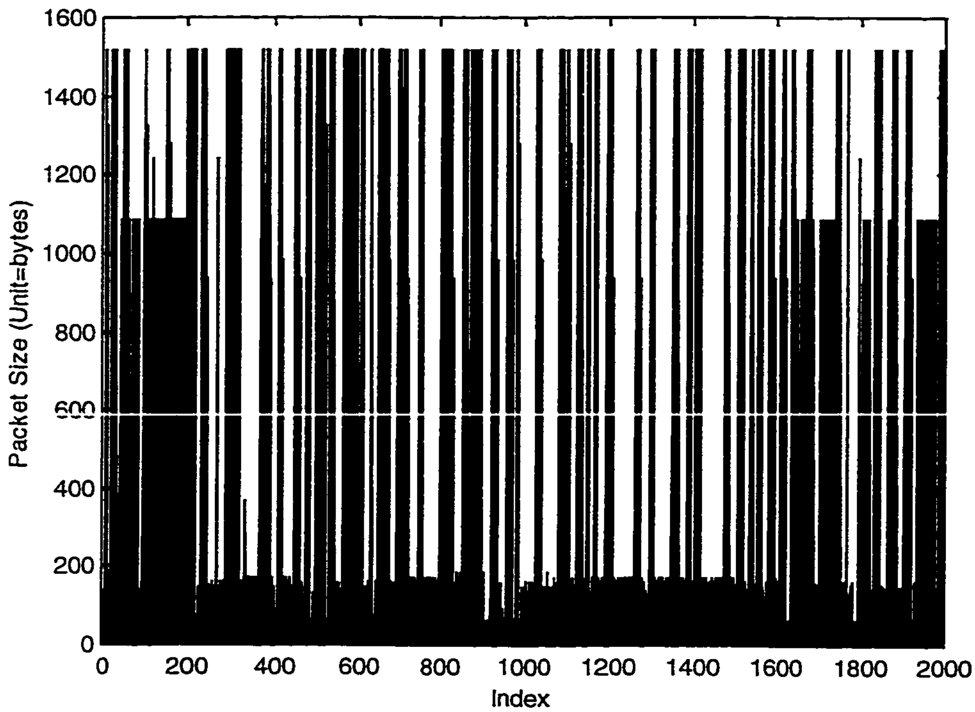


Figure 26. The packet series of self-similar traffic.

As the self-similar traffic is too bursty (as seen in figure 26), it will lead to a longer service time if we do not dynamically change the setting of INC value. So it is not suitable to calculate INC through the mean arrival rate in a certain period. We have to use a dynamic adjustment of INC to control the throughput of the CO service. In the first step, the proposed dynamic adjustment method realizes the change of INC based on the size of the latest arriving packet. Thus, we define two transmission rates: the peak cell rate and the low cell rate. The reason for defining these two rates is that the self-similar traffic has a property similar to a switch (this property can be seen from figure 26). The sizes of most of the packets are the maximum packet size (around 1500 bytes) or the minimum packet size (around 100 bytes) The peak cell rate is defined to the rate that the node gets the packets of maximum size continuously. For the low cell rate, the packets of minimum size are assumed to exist on the node permanently. Then the INC is calculated through these two cell rates. When a packet arrives to the node, its size is checked. If the size of this packet is over a threshold (in the simulation, we let the threshold be the average of maximum and minimum packet size), we set the INC according to the peak cell rate. Otherwise, the INC is set according to the low cell rate.

Table 1. The simulated results of each node with incoming Self-similar traffic under heavy load (two levels).

Node Index	Mean Arrival Rate	Efficience	Access Delay	Response Time	Mean Buffer Size
1	0.0931	88.10%	27.4023	1594.85	32.3
4	0.1189	79.50%	4.4124	6319.59	921.6
6	0.1498	75.20%	22.3592	765.94	16.68
9	0.0407	100%	23.6781	989.69	60.32
2	0.0705	100%	2.96	162.04	5.65
3	0.142	100%	2.92	189.61	23.98
5	0.1109	100%	3.44	225.53	13.39
7	0.1241	100%	5.51	218.62	25.5
8	0.0968	100%	4.58	215.58	26.85
10	0.1109	100%	12.39	225.52	23.35

Table 1 indicates the simulation results for our simulations. As the self-similar traffic is too bursty, the response time in some period is still long, all CO service nodes won't experience the unfairness existing in the QA service nodes if the QA nodes don't apply BWB mechanism or other similar mechanism. The CO service requirement should not exceed the total bandwidth for ensuring its QoS, theoretically. Thus, in some period when more than two CO nodes have burst traffic, the whole CO service requirement may exceed the network limit and it has to experience a longer service time. From the results of access delay and response time, the CO service is proved to suit for self-similar traffic comparing to the QA service, especially in the busy period.

In setting INC with the use of this method, there is still minutia that we haven't paid attention to. The interarrival time between two successive packets is not constant so that just setting the INC based on the packet size is improper in some situations. Sometime the CO service is over-supplied, while other times it is under-supplied. Because of different interarrival time, the arrival rate of each instant is changed dramatically. A better solution should take into consideration both the interarrival time and the packet size. Therefore, more levels of the value of the INC are given. With the packet size and interarrival time, a more accurate arrival rate can be found. Thus, based on the more accurate arrival rate and given more choices of the value of INC, the node can adjust its throughput quickly with the change of the incoming traffic to achieve a better service quality.

Table 2 gives the result that includes the consideration of both packet size and buffer size in INC setting. As the interarrival time has impact on the buffer size in one moment, the simulation partly uses the interarrival time as one element of its verification rule for bandwidth requirement level. In the simulation, there are four levels of INC setting: heavy load and large arriving packet (level 1), heavy load and small arriving packet (level 2), light load and large arriving packet (level 3), light load and small arriving packet (level 4). After a packet arrives to the node, the following procedure goes through:

1. The buffer size is examined;
2. If the buffer size is accumulated over 30 segments (nearly to the number of segments that the maximum packet holds), the heavy load is said to happen, else goto 3. If the arriving packet is over the packet size threshold, INC is set to level 1. If the arriving packet is small, INC is set to level 2
3. The load is light. If the arriving packet is over the packet size threshold, INC is set to level 3. If the arriving packet is small, INC is set to level 4.

The result shown below indicates that the CO service with four INC levels can give guaranteed throughput with smaller response time and buffer size than those shown in table 1. And the CO service becomes more stable.

Table 2. The simulated results of each node with incoming Self-similar traffic under heavy load (four levels).

Node Index	Mean Arrival Rate	Efficiency	Access Delay	Response Time	Mean Buffer Size
1	0.0931	88%	16.43	6217.8	847.5
4	0.1189	72.70%	17.18	4352.2	941.8
6	0.1498	81%	27.47	2831.8	731.9
9	0.0407	81.80%	22.85	6472.8	1072.8
2	0.0705	100%	18.28	196.6	28.3
3	0.142	100%	14.27	225.5	15.2
5	0.1109	100%	14.01	146.3	19.8
7	0.1241	100%	14.6	194.5	22.4
8	0.0968	100%	12.05	198.5	20.5
10	0.1109	100%	17.92	173.6	22.6

If more INC levels are defined and more exact thresholds are decided, the outcomes of simulation will be more accurate. Of course, this will bring more complexity to the system.

4.2 MPEG II Traffic Analysis

In the MPEG coding scheme, three types of frames have been defined. Each type uses a slightly different coding scheme. I-frames use only intra-frame coding based on the discrete cosine transform and entropy coding. P-frames use similar coding algorithm to I-frames, but with addition of motion compensation to the previous I or P-frame. B-frames are similar to P-frames, except that the motion compensation is done with respect to previous I or P-frame, the next I or P-frame, or an interpolation between them (see figure 27).

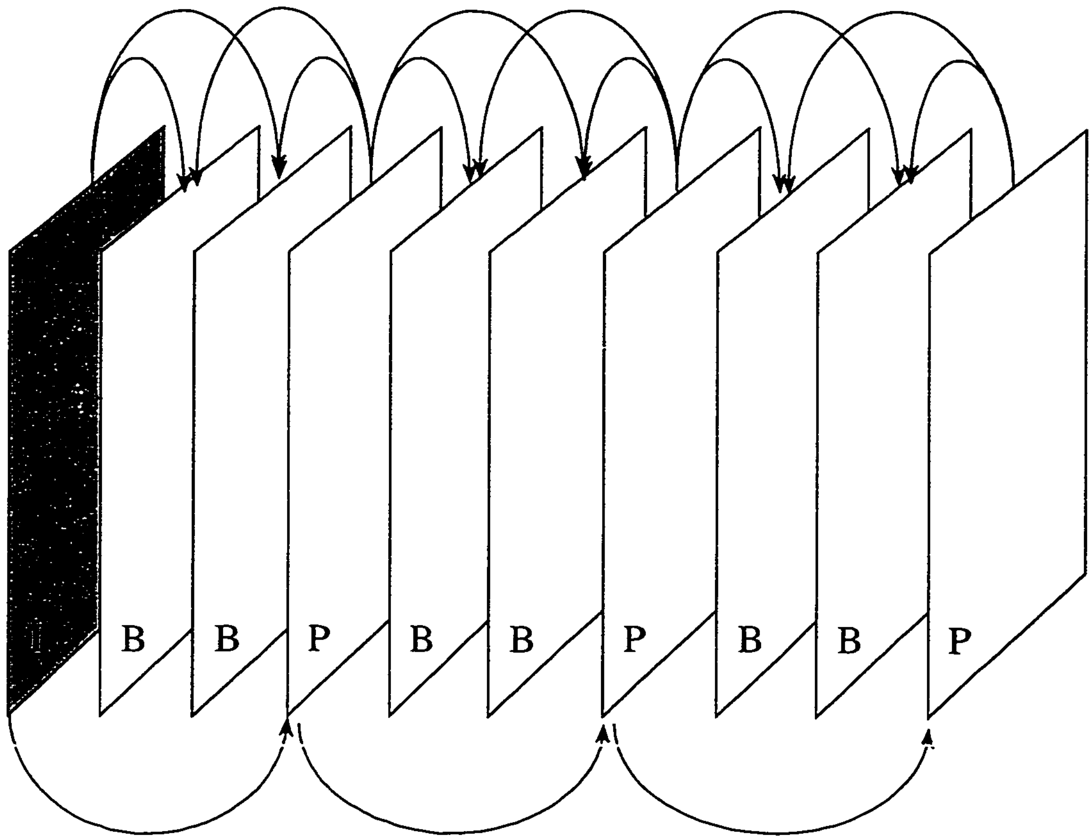


Figure 27. Example of MPEG temporal structure.

Typically, I-frames require more bits than P-frames. B-frames have the lowest bandwidth requirement. The different ways in coding frames result in different traffic characteristics for the different frame types. At the encoder side, the frames are arranged in a deterministic periodic sequence, e.g. “IBBPBBPBBPBBPBBI”, called Group of Pictures (GOP). This pattern is repeated periodically until the whole stream is compressed. The pattern is defined by the numbers and types of P and B-frames between successive I-frames. The transmitted sequence is a permutation of the ordering of the original sequence, e.g. “IPBBPBBPBBPBBIBB”. The GOP plays a major role in the source statistics of an MPEG video stream as it fixes the periodic nature of the MPEG-2 video source [27, 28].

Figure 28 gives the statistics of traffic series of MPEG movie Mr. Bean in unit of bytes. After an I-frame there are several P and B frames following. The size of I-frame is around 60000 bytes while P and B frames ranges from 40000 to 10000 bytes. The average frame size over a long period is around 20000 bytes.

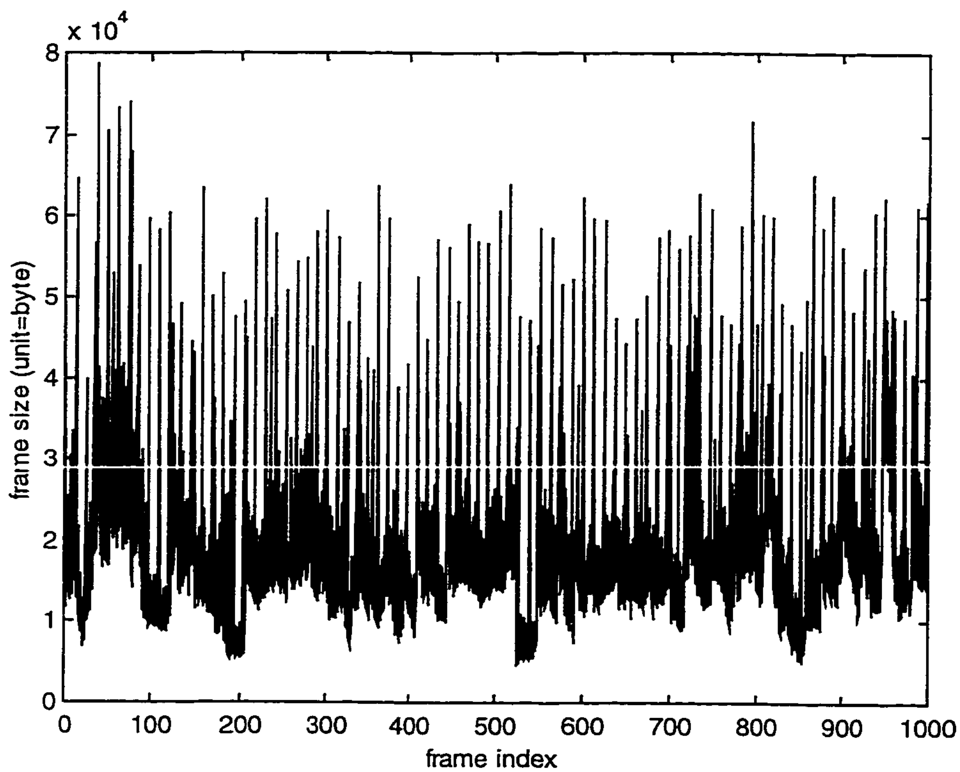


Figure 28. The frame series of movie Mr. Bean.

First, we can take the simple situation in which only one of ten nodes in the network supports MPEG-2 video traffic and employs CO data service to transmit segments. All other nodes traffic is Poisson distributed traffic and are set to QA service or CO service. The transmission rate of the DQDB network is around 80 Mbps so that the peak rate of MPEG-2 traffic will not exceed the total bandwidth of network.

In the first simulation for MPEG-2 traffic, the CO service node calculates its INC for each frame after the frame arrives to the node (called *frame base INC calculation*).

Thus, the node will send all segments of one frame before the next frame comes into the node. As the time interval between two conseutive frames is fixed to 1/30 seconds, the INC is set as:

$$\text{INC}=\text{SGC} \bullet (\text{frame size in bits} \bullet 30 / \text{transmission rate}) \qquad (21)$$

Under this INC calculation method, the node will send the frame uniformly spread over a frame time as shown in figure 29. The transmission time of each frame will be around 1/30 sec.

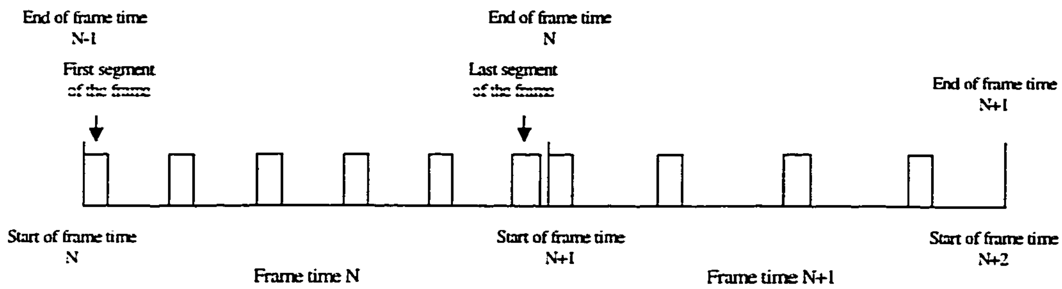


Figure 29. The segments transmission of the frame base INC calculation

Table 3 gives the value of frame size and the transmission time of these frames. This transmission time is the time elapsed from the arrival of the frame to the node to the time that the last segment of the frame leaves the node. Under this INC setting, the transmission time for each frame is kept around the 1/30 sec (6060 slots) regardless of the frame size. That conforms to the above analysis of the transmission time.

Table 3. The transmission time of MPEG-2 video traffic with frame base INC calculation.

	Frame 1	Frame 2	Frame 3	Frame 4
Frame Size (in slots)	637	80	78	221
Transmission Time (in slots)	6012	5966	5969	6004

If we transmit the MPEG-2 frame in the previous way, the delay is as large as 1/30 sec. This is because the node sends its segments periodically within one frame time according to the INC which is fixed in a frame-time basis. In order to guarantee a better QoS, one way is to increase the required bandwidth of CO service for the transmitting of MPEG-2 video traffic, i.e. to increase the value of INC, whenever possible.

Our purposed method is as follows: In the AFC of each QA slot, there is one bit reserved for future use. If we use this bit to indicate the slot usage i.e. to indicate if the slot is used by CO service, combined with the CO service requests from downstream nodes each node can verify the bandwidth occupancy of CO service of the other nodes. (Or we can deduct one request bit for this purpose.) If a CO node sends a segment in one slot, the slot usage bit is also set to indicate it is a CO service slot. Then each node can calculate the bandwidth occupied by CO service in a passing certain period in the following way.

Each node requires two additional counters. One (COSC) is used to record the number of CO requests seen on reverse bus in this certain period. The other (CORC) is to count the number of CO service slots seen on the data bus. Then the node can calculate the percentage of CO service applied by the other nodes using the following formula:

Usage of CO service (%) = (COSC + CORC) / observation period (slot) (22)

Table 4. The transmission time of MPEG-2 video traffic with slot base INC calculation.

	Frame 1	Frame 2	Frame 3	Frame 4
Frame Size (in slots)	637	80	78	221
Transmission Time (in slots)	672	112	109	253

If the bandwidth occupied by the CO service including the target CO node's requirement does not exceed the total bandwidth of network, this CO node can increase its bandwidth requirement by certain value until the bandwidth of CO service reaches the full network transmission ability. We call this implementation *slot based INC calculation* because we calculate the INC and make the decision at each slot time in the simulation. Table 4 shows the new transmission time of each frame under the mentioned method. Here we increase INC by the value that makes the total occupied bandwidth of CO service equal to the bandwidth of network. The transmission time is decreased dramatically from around 6000 slots to the value near the frame size in slots compared to frame based INC calculation. Figure 30 illustrates the scenario of segments transmission that appears in the slot base INC calculation. Unlike the frame base INC calculation, the node will try to send the segments in a block until there isn't any segments left in the queue.

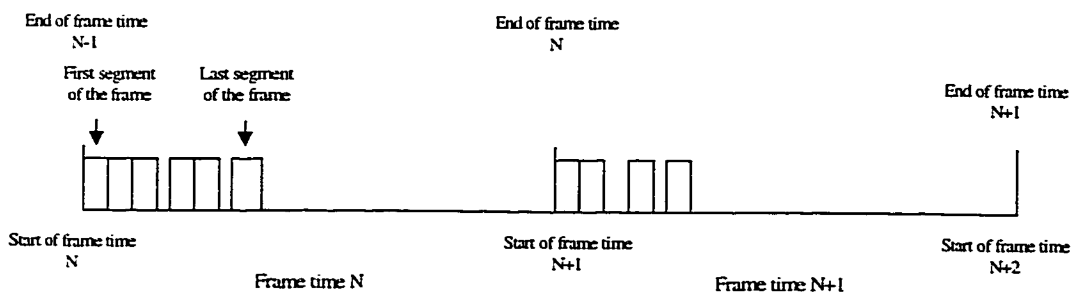


Figure 30. The segments transmission of the slot base INC calculation

Conflicts may arise when more than two CO nodes are sending MPEG-2 video segments in the same time. All nodes attempt to occupy their full bandwidth share. In this case, the overall CO service requirement may exceed the capacity of the network. Thus, proper mechanisms should be proposed to let these nodes to increase their bandwidth evenly or proportionally according to the frame length.

With the same scenario as before (10 nodes in the network), nodes 2, 3, 5, 7, and 8 support MPEG-2 video traffic. We consider both frame base and slot shemes to set the value of INC. The average MPEG-2 traffic input equaling to 60% of total bandwidth (so that the transmission rate is around 40 Mbps), the two methods are compared in terms of the probability of frame loss. If one frame can not be transferred totally in 1.5/30 second (one and half frame time), a deadline missing is said to happen. Table 5 gives the five MPEG video statistics. The mean arrival rate of five MPEG-2 streams is 21.27 Mbps. But the peak arrival rate of one MPEG-2 stream may be over the network capacity. Thus, the deadline missing can not be avoided. If we want to diminish all deadline missing, the best way is to expand the network capacity. Here we just verify if these two methods can limit the number of missing deadlines.

Table 5. The five MPEG-2 video traffic properties.

MPEG Video	mean rate (Mbps)	peak rate (Mbps)
Movie "Terminator"	2.66	19.09
News	3.65	41.53
Movie "Mr. Bean'	3.83	54.98
MTV	4.59	47.87
Soccer Game	6.54	43.77

Under the frame base INC calculation, the available bandwidth will be less than the required bandwidth of CO service when one of MPEG video reaches its peak rate. Under this situation, the node only can occupy the available bandwidth and try to send the remaining within the next frame. Thus, at the beginning of each frame time, the node calculates the INC, the required bandwidth, based on the frame size and the segments left before (i.e., left bandwidth requirement). At the end of this 1/30 sec, the node will count the number of the segments that are sent and count the number of segments left in the queue. These segments can not be transmitted and will be added to the next incoming frame to calculate the new INC for the next 1/30 sec. In the slot base INC calculation, the node just works as mentioned before. Each CO service node just tries to use all available bandwidth left by the other CO service nodes.

Table 6. The number of deadline missing in 10 seconds by applying both slot base and frame base INC calculation with different starting time.

	No. of deadline missing (slot base)	No. of deadline missing (frame base)
Node 2	6	70
Node 3	7	51
Node 5	16	0
Node 7	6	0
Node 8	4	0
total	39	121

During the simulation, we first let each CO node get MPEG-2 frames input at different times in a 1/30 second but they still start in same 1/30 second. Table 6 lists the number of deadline missing when applying both the slot-base and frame-base INC calculation. The slot base INC calculation shows better results than the frame base INC calculation as far as the number of deadline missing is concerned. The frame base INC calculation exposes the unfairness in deadline missing. The upstream CO service nodes tend to have more deadline missing than the downstream CO service nodes. As the CO service need to send request for each segment before it is transmitted and the free high priority request bits can not satisfy the requirement of upstream CO nodes. When the total CO requirement exceeds the network capacity, the upstream nodes have to delay their transmission until the overall CO requirement decreases. The downstream CO nodes have advantage in setting their requests.

Table 7 presents the number of deadline missing when all the CO service nodes start to send at almost the same time. This is one kind of worst situations. The probability that more than two CO service nodes reach peak rate is increased. Although the slot based INC calculation method has smaller number of deadline missing, both INC calculation methods exhibit unfairness towards the upstream CO service nodes.

Table 7. The number of deadline missing in 10 seconds by applying both slot base and frame base INC calculation with same starting time.

	No. of deadline missing (slot base)	No. of deadline missing (frame base)
Node 2	39	102
Node 3	28	47
Node 5	7	0
Node 7	4	0
Node 8	1	0
total	79	149

In order to resolve the unfairness on the distribution of the number of deadline missing for the frame based INC calculation method, a solution is proposed which we hope could solve this problem. Because the downstream CO service nodes can access the network without too many disturbances through the advantage in request setting, to send one frame in just one frame time is not a problem for these nodes. Thus, we can expend the transmission time of one frame of downstream nodes to certain value that is limited by the deadline missing time (That is transferring a frame in certain period between one frame time and the deadline missing time, i.e. one and half frame time, instead of just in one frame time.). Thus, the bandwidth requirement of downstream CO nodes is decreased and upstream CO nodes will have less worries about the short of bandwidth.

In the following simulation, we expend the transmission time of a frame to 1.3 frame time. When a deadline missing happens, the node stops expending the frame transmission time for at least five frames. The result indicated in table 8 shows the unfairness still exists but the number of deadline missing is decreased.

Basically, neither the slot base INC calculation nor the frame base INC calculation method can completely solve the problem of deadline missing. They just provide ways to allocate the bandwidth evenly. CO service is set up under the assumption that total CO service requirement should not exceed the network. While the real reason of the deadline missing is the shortage of available bandwidth during some period of time. By the way, if all MPEG-2 stream do not start simultaneously in 1/30 second, the probability that more than two MPEG-2 streams reach the peak arrival rate will be extremely small. Then the chances of the deadline missing will also be small.

Table 8. The number of deadline missing in 10 seconds by applying frame base INC calculation with same starting time (modified).

	No. of deadline missing (frame base)
Node 2	53
Node 3	39
Node 5	8
Node 7	0
Node 8	0
total	92

The CO service of the DQDB subnetwork for the MPEG-2 video transmission is just like the UBR service in the ATM network, if we choose the frame base INC calculation method. To have a good understanding on how this similarity is, a DQDB network is given with the capacity of 40Mbps, but only four nodes exist in the DQDB network. Node 2 and 4 are CO service nodes for the MPEG-2 transmission. Node 1 and 3 are saturated QA service nodes that want to occupy each slot for its own segments transmission. Figure 31 illustrates the simulation results of the throughput for these four nodes. The QA service nodes adapt the bandwidth balance to evenly share the bandwidth left by the CO service as the ABR service in the ATM network. The throughput figure shows that CO services work as VBR of ATM to adjust its bandwidth according to the frame size and QA services compete for the left bandwidth as the ABR service of the ATM network.

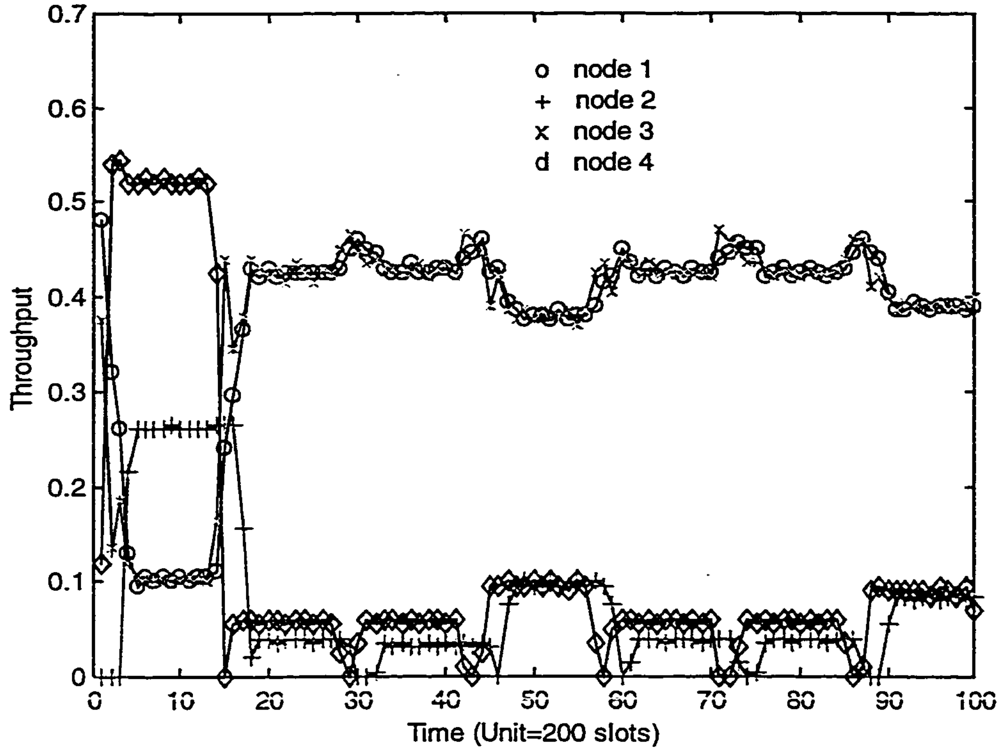


Figure 31. The throughput of each nodes in the DQDB network with node 2 and 4 sending MPEG-2 frames.

Chapter 5

Conclusions and Future Work

Interest on MANs has been spurred by the need of providing broadcasting facilities rarely found in the traditional point-to-point and switched network techniques used in WANs. The purpose of MANs is pivotal in the sense that LANs and WANs can be interconnected within a city for the purpose of providing high-speed and integrated services. After many years of research on MANs, a number of alternatives have been explored and rejected. DQDB has received widespread support from service providers and users, and has been standardized by the IEEE 802 committee as IEEE 802.6. Three services have been defined: the isochronous service, which provides support for user that require a constant interarrival time in effect, i.e., a circuit-switched service; the connection-oriented virtual circuit service, which supports users sharing a virtual channel connection, and the connectionless data service, which provides support for connectionless communication via the LLC protocol. However, the development of more applications with new challenges requires to properly service such applications as video and audio. Performance demands are becoming more critical, especially when the traffic load to MANs includes a large number of video and audio packets, interworking, broad range activities, etc. These are the main reasons the design, analysis and evaluation of CO service are considered as important attributes in providing new type of service in DQDB networks to support high QoS applications.

In chapter 1, the reasons behind the creation of MANs were presented. A detailed definition was also given with a description of the services provided by the MANs. Many possible protocols have been proposed to support MANs and the DQDB protocol was selected by IEEE 802 committee as the standard MAN. Chapter 1 also stated the objectives, problems, and challenges to which this thesis provides some solutions. We identified that the CO service is a good solution to satisfy the required high QoS of some applications. General questions were formulated, for which answers were provided throughout this thesis.

We started introducing DQDB from its history and evolution in chapter 2. The basic features illustrate the attractiveness of DQDB network to those high-speed data services. There are two topologies supported by DQDB networks: open bus and looped bus. For different service requirements, two types of slots (QA and PA slots) have been defined in the standard. The QA access method uses an on-demand access method (packet-switched like). A node implementing this access method carries on the following steps:

- Sends a request for transmission to the upstream nodes,
- Joins the distributed queue by using the countdown counter,
- Maintains the distributed queue with the request counter,
- Allows the passage of enough empty slots to satisfy the request of the nodes ahead of it in the distributed queue,
- Sends its own data when its turn comes.

The CO service has been developed on top of the medium access control protocol of DQDB. Along the way PA access method is also mentioned to explain how to realize isochronous service and connection oriented virtual circuit service. The main pitfall of DQDB network is the unfairness under heavy load network conditions. The BWB mechanism is adopted by IEEE 802.6 committee as the standard to solve this pitfall.

Chapter 3 first described the fundamental theorem for the implementation of CO service. This theorem was proposed by Karvelas [52]. A credit counter and three parameters (SGC, INC, and CRM) are needed to guarantee the bandwidth. Nevertheless, the given definition can not ensure a proper operation of CO service under many situations. Thus, we had to solve the encountered problems step by step and define the CO service thoroughly.

Since the CO service will be used in conjunction with other services, it is important and necessary to let all services co-exist. Both CO and QA services transmit through QA slots, some modifications to the QA service have been needed. Two possible solutions were considered and both proved to get the similar results. The first method employs a mechanism similar to the CO service allowing the node to recognize the bandwidth left by the CO service then to confine the QA service demands. The first method restrains the segments of the QA service sending according to the accumulated credit. Compared to the first techniques, the second method offers a simpler solution in accommodating the QA and CO services.

The second problem encountered in the simulation was to figure out that the absence of RC and CD in the CO service will let downstream CO service experience long access delay. In some of worst situation, some segments were blocked in the queue even when the node has become idle for a long time. Different situations of this kind of violation are analyzed and deciphered with usage of RC and CD array.

Furthermore, another problem was detected when setting the CO service using the Karvelas' method to set up the protocol's parameters [52]. A detailed analysis was built to find out the misbehavior. Due to minimum effect on other CO service nodes, a dynamic setup mechanism for CRM was proposed.

Two flow charts were included to explain the whole procedure to be implemented by the CO node in respect of data bus and request bus. It was shown that the longest access delay encountered by a node under the worst case will not exceed the round trip time from the node to the most upstream node.

In chapter 4, two types of real traffic were taken into account. Self-similar traffic is a recently commonly recommended traffic model. The proposed method for the INC setting is based on both the arriving packet size and the length of queue which holds the segments waiting to be sent.

MPEG coding scheme becomes the most popular compressing scheme for video. We explored the use of the CO service to transmit VBR MPEG-2 traffic. Two possible solutions were suggested to set the value of INC. As we want to send the MPEG frame as soon as possible, the CO service tries to capture all available bandwidth left by other CO services until its CO service queue becomes empty (solution 1). Under the second scheme we calculated the INC on a frame base (solution 2). For each arriving frame, INC is set to the value that can guarantee the frame to be sent out in a fixed period. If the available bandwidth can not fulfill the requirement, the unsent parts of frame will be added to next frame for new INC calculation. The number of frames having missed the deadline is statistically counted.

In recent presented IEEE 802.6 DQDB CO service standard, CD and RC counters are also mentioned. But a different way of setting INC is adopted. Due to the multiple requests supported by CO service, an array of CD or RC counters is needed although it isn't mentioned by the standard. The way to set the INC is very similar to VBR service in ATM network. Two arrival rates were defined to set INC and SGC. Only when both credit counters get enough credit, a request can be sent.

Further work on CO service in DQDB includes the specification and modification of a standard which is documented by the IEEE 802.6 in 1994. We have already studied the GBW protocol for additional scenarios with two priority sources simultaneously active. In these studies the GBW protocol proved to be quite insensitive to load characteristics, distance enlargement and total load at high and/or low priority. However, additional studies are necessary on several points:

- Service requirement of other variable bit rate applications: The GBW protocol may be tuned for specific applications but prospective users have not yet joined the standardization effort.
- Guaranteed delays: In its current form, the CO service can guarantee both the bandwidth and access delay but the access delay can be improved through many efficient ways such as slot reuse and so on.

References

1. IEEE Standard 802.6-1990, *IEEE Standard for Local and Metropolitan Area Networks: Distributed Queue Dual Bus (DQDB) Subnetwork of a Metropolitan Area Network (MAN)*, July 1991.
2. H. I. Sigiuk, B. H. Pardoe, *DQDB network with a closed loop bandwidth balancing,* Computer Communications, 19 (1996), pp. 517-527.
3. G. Cena, R. Sisto, A. Valenzano, *A new strategy for fair access to DQDB metropolitan area networks*, IR DAI/ARC 4-94 (Politecnico di Torino, 1994).
4. G. Cena, L. Durante, R. Sisto, A. Valenzano, *Adaptive bandwidth balancing mechanisms for DQDB networks*, Computer Communications, 20 (1997), pp. 804-819.
5. I. F. Akyildiz, J. Liebeherr, A. N. Tantawi, *DQDB+/-: A fair and waste-free media access protocol for dual bus metropolitan networks*, IEEE Transaction on Communications, Vol. 41, No. 12, December 1993, pp. 1805-1815.
6. J. Liebeherr, I. F. Akyildiz, A. N. Tantawi, *Distributed fair and waste-free media access protocols for dual bus metropolitan area networks*, IBM Research Div., Tech. Rep. RC 17489, Dec. 1991.
7. L. N. Kumar, A. D. Bovopoulos, *Access protection solution for heavy load unfairness in DQDB*, Computer Communications, Vol. 18, No. 12, Dec. 1995, pp. 896-910.
8. O. Sharon, *A proof for lack of starvation in DQDB with and without slot reuse*, IEEE/ACM Transactions on Networking, Vol. 5, No. 3, June 1997, pp. 410-419.
9. O. Sharon, A. Segall, *A simple scheme for slot reuse without latency for a dual bus configuration*, IEEE/ACM Transactions on Networking, Vol. 1, No. 1, February 1993, pp. 96-104.
10. L. N. Kumar, A. D. Bovopoulos, *3-tier structured access protocol to control unfairness in DQDB MANs*, Computer Communications 19 (1996), pp. 389-406.

11. D. G. Jeong, C. H. Choi, W. S. Jeon, *Adaptive bandwidth waste rate: a fair bandwidth sharing scheme for the DQ protocol with multiple priority classes*, Computer Communications, Vol. 17, No. 8, August 1994, pp. 571-580.
12. T. Y. Huang, J. L. C. Wu, *Priority promotion DQDB networks to improve fairness*, Computer Communications, Vol. 17, No. 5, May 1994, pp. 332-338.
13. A. E. Kamal, A. Bissonauth, *Priority mechanism for the DQDB network*, IEEE Proc.-Commun., Vol. 141, No. 2, April 1994, pp. 98-104.
14. S. Casale, V. Catania, A. L. Corte, *Service integration issues on an ATM DQDB MAN*, Computer Communications 17 (6) (1994), pp. 407-418.
15. S. Casale, V. Catania, A. L. Corte, *A service management on an ATM DQDB MAN*, Computer Communications, 16 (3) 1993, pp. 147-154.
16. A. Lombardo, S. Palazzo, *An architecture for a pure ATM metropolitan area network*, GLOBECOM'89, Dallas, Texas, USA, pp. 1828-1834.
17. L. Pu, H. Gang, L. Lemin, T. Sausu, *The design of a broadband integrated service LAN*, IBCN & S, Copenhagen, Denmark, 1993, pp. 19-23.
18. A. L. Garcia, *Probability and random processes for electrical engineering*, Second edition, Addison Wesley 50037, 1994.
19. W. E. Leland, M. S. Taqqu, W. Willinger, D. V. Wilson, *On the self-similar nature of ethernet traffic*, ACM/SIGCOMM Computer Commun. Rev., Vol. 23, 1993, pp. 183-193.
20. S. M. Klivansky, A. Mukherjee, C. Song, *On long-range dependence in NSFNET traffic*, Preprint, 1994.
21. V. Paxon, S. Floyd, *Wide-area traffic: The failure of poisson modeling*, Proc. ACM/SIGCOMM'94, London UK, pp. 257-268.
22. J. Beran, *Statistics for Long-Memory Processes*, CHAPMAN & HALL, 1994.
23. W. Willinger, M. S. Taqqu, R. Sherman, D. V. Wilson, *Self-similarity through high-variability: Statistical analysis of ethernet LAN traffic at the source level*, IEEE/ACM Transactions on Networking, Vol. 5, No. 1, February 1997, pp. 71-86.

24. M. W. Garrett, W. Willinger, *Analysis, modeling, and generation of self-similar VBR video traffic*, Proceeding of the ACM Sigcomm'94, London, UK, 1994, pp. 269-280.

25. C. Huang, M. Devetsikiotis, I. Lambadaris, A. R. Kaye, *Modeling and simulation of self-similar variable bit rate compressed video: A unified approach*, Proceeding of the ACM/SIGCOMM'95, Cambridge, MA, August 1995, pp. 2225-2231.

26. J. Gordon, *Pareto process as a model of self-similar packet traffic*, Proceeding of the ACM/SIGCOMM'95, Cambridge, MA, August 1995, pp. 2232-2236.

27. P. Pancha, M. El Zarki, *MPEG Coding for Variable Bit Rate Video Transmission*, IEEE Communications Magazine, May 1994, pp. 54-66.

28. P. Cuenca, A. Garrido, F. Quiles, L. Orozco-Barbosa, *Performance Evaluation of Cell Discarding Mechanisms for Hierarchical VBR MPEG-2 Video Traffic over ATM Networks*, Proceeding of ICC'98, Atlanta, USA, June 1998.

29. T. Madron, *Local area networks: New technologies, Emerging standards*, New York: Wiley, 1994.

30. J. Martin, K. Chapman, J. Leben, *Local area networks: Architectures and Implementations*, Englewood Cliffs, NJ: Prentice Hall, 1994.

31. W. Bauerfeld, *A tutorial on network gateways and interworking of LANs and WANs*, Computer Network and ISDN Systems, Vol. 13, 1987, pp. 187-193.

32. K. Annamalai, *Emerging high-speed local-area networks and wide-area networks*, Bellingham, Wash., USA : SPIE, 1995.

33. G. Kessler, D. Train, *Metropolitan Area Networks: Concepts, Standards, and Services*, New York: McGraw-Hill, 1992.

34. M. Sadiku, *Metropolitan Area Networks*, Bocan Raton, FL: CRC Press, 1995.

35. W. Stallings, *Local and Metropolitan Area Networks*, fifth edition, Upper Saddle River, NJ: Prentice Hall, 1997.

36. R. E. Dettra, *Metropolitan Area Networks – Data Transport Highways*, AT&T Technology, Vol. 6, No. 1, 1991, pp. 8-11.

37. J. Gantz, Does the world need MANs?, Networking Management, Vol. 8, No. 6, 1990, pp.70-72.

38. F. A. Tobagi, F. Borgonovo, L. Fratta, *Expressnet: a High Performance Intergrated-service Local Area Network*, IEEE Journal on Selected Areas in Communications, Vol. SAC-1, No. 5, November, 1983, pp. 898-913.

39. F. A. Tobagi, M. Fine, *Performance of Unidirectional Broadcast Local Area Networks: Expressnet and Fasnet*, IEEE Journal on Selected Areas in Communications, Vol. SAC-1, No. 5, November 1983, pp. 913-926.

40. F. Borgonovo, *ExpressMAN: Exploiting Traffic Locality in Expressnet*, IEEE Journal on Selected Areas in Communications, Vol. SAC-5, No. 9, December 1987, pp. 1436-1443.

41. J. O. Limb, Fasnet: *Proposal for a High Speed Local Network*, in N. Naffah, Office Information Systems, Amsterdam: North-Holland, 1982, pp. 87-91.

42. J. O. Limb, C. Flores, Description of Fasnet – A Unidirectional Local Area Communication Network, Bell System Technical Journal, Vol. 61, No. 7, September 1982, pp. 1413-1440.

43. S. Mirchandani, R. Khana, *FDDI: Technology and Applications*, New York: John Wiley & Sons, 1993.

44. F. E. Ross, *An Overview of FDDI: The Fiber Distributed Data Interface*, IEEE Journal on Selected Areas in Communications, Vol. SAC-7, No. 7, September 1989, pp. 1043-1051.

45. D. Tsac, *FDDI: Chapter Two*, Data Communications, December 21, 1991, pp. 59-70.

46. G. Watson, D. Cunningham, *FDDI and Beyond: A Network for the 90's*, IEEE Review, April 1990, pp. 131-134.

47. R. M. Newman, Z. L. Budrikis, J. L. Hullett, *The QPSX Man*, IEEE Communications Magazine, Vol. 26, No. 4, April 1988, pp. 20-28.

48. K. Sauer, W. Schodl, *Performance Aspects of the DQDB Protocol*, Computer Networks and ISDN system, Vol. 20, 1990, pp. 253-260.

49. V. P. T. Phung, R. Breault, *On the Unpredictable Behavior of DQDB*, Computer Networks and ISDN System, Vol. 24, 1992, pp. 145-152.

50. C. C. Bisdikian, *A Performance Analysis of the IEEE 802.6 (DQDB) Subnetwork with the Bandwidth Balancing Mechanism*, Computer Networks and ISDN Systems, Vol. 24, 1992, pp. 367-385.

51. W. Stallings, *Networking Standards*, MA: Addison-Wesley, 1993, pp. 398-433.

52. D. Karvelas, *The Guaranteed Bandwidth and Balancing Mechanism for High Speed Metropolitan Area Networks*, ICC'94, New Orlean, pp. 696-700.

53. P. Martini, *Connection Oriented Data Service in DQDB*, Computer networks and ISDN systems, Vol. 26, 1994, pp. 679-694.

54. CCITT Recommendation G.703, *Physical/Electrical characteristics of hierarchical digital interfaces*, Vol. III, Fascicle III.4 of the CCITT Blue Books-General aspects of digital transmission systems; terminal equipments.

55. ANSI T1.102-1987, *American National Standard for telecommunication-Digital Hierarchy-Electrical Interfaces*.

56. ANSI T1.107-1988, *American National standard for telecommunication-Digital Hierarchy-Format Specifications*.

57. CCITT Recommendation G.707, *Synchronous digital hierarchy bit rates*, Vol. III, Fascicle III.4 of the CCITT Blue Books-General aspects of digital transmission systems, terminal equipments.

58. CCITT Recommendation G.708, *Network node interface for the synchronous digital hierarchy*, Vol. III, Fascicle III.4 of the CCITT Blue Books-General aspects of digital transmission systems, terminal equipments.

59. CCITT Recommendation G.709, *Synchronous multiplexing structure*, Vol. III, Fascicle III.4 of the CCITT Blue Books-General aspects of digital transmission system, terminal equipments.

Appendix A

Analysis in changing CRM

To increase the CRM, we must ensure the new CRM should be greater than the old one from Equation (2). From formula (5), the new CRM is

$$CRM \geq 2 \bullet SGC - (\lceil SGC/INC \rceil - 1) \bullet INC \qquad \text{(i)}$$

While the old CRM from formula (2) is

$$CRM \geq \lceil SGC/INC \rceil \bullet INC, \qquad \text{(ii)}$$

The following inequality should be satisfied:

$$2 \bullet SGC - (\lceil SGC/INC \rceil - 1) \bullet INC \geq \lceil SGC/INC \rceil \bullet INC \qquad \text{(iii)}$$

Thus

$$2 \bullet SGC \geq 2 \bullet \lceil SGC/INC \rceil \bullet INC - INC. \qquad \text{(iv)}$$

If we apply equality, we get

$$2 \bullet SGC = 2 \bullet \lceil SGC/INC \rceil \bullet INC - INC \qquad \text{(v)}$$

Because SGC $\geq$ INC, the above equation can be written as

$$2 \bullet SGC = (2 \bullet n-1) \bullet INC, \qquad n = 1, 2, 3, 4, \ldots \qquad \text{(vi)}$$

For n = 1(SGC = INC), the inequality (iii) is always true. For n = 2 (2•INC ≥ SGC > INC), from the equation (vi) we get

$$SGC/INC = 3/2 \qquad \text{(vii)}$$

Thus, if 2•SGC/3 > INC ≥ SGC/2, i.e.,

$$\frac{SGC}{INC} \in \left(\frac{3}{2}, 2\right]$$

the inequality (iii) is true. Otherwise (SGC > INC ≥ 2•SGC/3), the new CRM will not be greater than the old one and we have to increase it further. Based on the same idea, the next CRM follows:

$$CRM - SGC + (\lceil SGC/INC \rceil - 2) \bullet INC \geq SGC \qquad \text{(viii)}$$

Since $\lceil SGC/INC \rceil = 2$ under the condition, the inequality can be written as

$$CRM \geq 2 \bullet SGC \qquad \text{(ix)}$$

The new value of CRM will be 2•SGC.

When n = 3 (3•INC ≥ SGC > 2•INC), the equation (vi) becomes

$$SGC/INC = 5/2 \qquad \text{(x)}$$

Then if 2•SGC/5 > INC ≥ SGC/3, i.e.,

$$\frac{SGC}{INC} \in \left(\frac{5}{2}, 3\right]$$

the inequality (iii) is true. Otherwise, we apply the same strategy. Then the CRM follows

$$CRM - SGC + (\lceil SGC/INC \rceil - 2) \bullet INC \geq SGC \qquad \text{(xi)}$$

After rearrangement, (xi) becomes

$$CRM \geq 2 \bullet SGC - (\lceil SGC/INC \rceil - 2) \bullet INC = 2 \bullet SGC - INC \qquad \text{(xii)}$$

Also, we can get

When $n = 4$ ($4 \bullet INC \geq SGC > 3 \bullet INC$), if $2 \bullet SGC/7 > INC \geq SGC/4$, i.e.,

$$\frac{SGC}{INC} \in \left(\frac{7}{2}, 4\right]$$

the new CRM comes from (iii). Else we apply (viii).

When $n = 5$ ($5 \bullet INC \geq SGC > 4 \bullet INC$), if $2 \bullet SGC/9 > INC \geq SGC/5$, i.e.,

$$\frac{SGC}{INC} \in \left(\frac{9}{2}, 5\right]$$

the new CRM comes from (iii). Else we apply (viii). And so on.

In generality, when we need to increase CRM from the original value from Equation (2), the following rule is obeyed:

IF

$$\frac{SGC}{INC} \in \left(\left(\frac{2i-1}{2}\right), i\right] \qquad i = 2,3,4,...$$

THEN

$$CRM \geq 2 \bullet SGC - (\lceil SGC/INC \rceil - 1) \bullet INC \qquad \text{(xiii)}$$

ELSE i.e.,

$$\frac{SGC}{INC} \in \left(i-1, \left(\frac{2i-1}{2}\right)\right] \qquad i = 2,3,4,...$$

THEN

$$CRM \geq 2 \bullet SGC - (\lceil SGC/INC \rceil - 2) \bullet INC. \qquad \text{(xiv)}$$

In the case that CRM should be increased more than one time, an order is followed. First of all, CRM is changed within the following inequality:

$$CRM \geq j \bullet SGC - (\lceil SGC/INC \rceil - i) \bullet INC \qquad \text{(xv)}$$

where j and i are integers, $j \geq 2$ and i is from 0 to $\lceil SGC/INC \rceil$. The initial value of j is 2 and the initial value of i is 0. At first step of CRM increasing, i equals to 1 or 2 based on the value of SGC/INC belonging to which range. The next step is to increase i by 1 to get a greater CRM. When i reaches to $\lceil SGC/INC \rceil$, we can not increase it any more and have to reset i to 0 and increase j by 1. When decreasing CRM, it goes to an opposite way. First i is decreased. When i equals to 0, it is reset to $\lceil SGC/INC \rceil$ and j is decreased by 1 until CRM reaches to the value of Equation (2).