Implementation of medium access control for interconnecting slotted rings to form a WDM metropolitan area network

Charalambos Linardakis, Helen-Catherine Leligou, Alexandros Stavdas, and John D. Angelopoulos

Telecommunications Systems Laboratory, School of Electrical and Computer Engineering, National Technical University of Athens, 9 Heroon Polytechniou Street, Zografou, 15773, Athens, Greece
chlin@telecom.ntua.gr; nelly@telecom.ntua.gr; astavdas@telecom.ntua.gr; jangel@telecom.ntua.gr

RECEIVED 28 JULY 2004; REVISED 24 SEPTEMBER 2004; ACCEPTED 24 SEPTEMBER 2004; PUBLISHED 22 OCTOBER 2004

A system of slotted interconnected rings employing a combination of wavelength-division multiple access (WDMA) with time-division multiple access (TDMA) can serve a metropolitan area without electro-optical conversion and buffering of payload except at system entry points. The multiple rings overcome the power budget limitations of the single ring extending the reach of the system to even the largest metropolitan areas, the WDM dimension provides flexibility and ease of evolution, and the TDMA dimension offers the efficiency of multiplexing gain particularly under bursty traffic. The system control information is transferred on a dedicated wavelength and is processed in the electrical domain at the ring nodes and the hub, which interconnects the rings. The algorithms control the access to each ring and the scheduling of slots among the rings, based on explicit reservations, to adapt efficiently to the fluctuating offered load. We present the design and hardware implementation of the access control algorithms for such a system built in the framework of the Information Society Technologies (IST) project DAVID (data and voice over DWDM). © 2004 Optical Society of America

OCIS codes: 250.0250, 060.4250.

1. Introduction

Wavelength-division multiplexing (WDM) technology has evolved from a way to multiply the transmission capacity to a networking solution integrated in the optical network architecture. It allows for cost-effective circuit provisioning but still lacks low-cost fast reconfiguration to compete with the time-division dimension. The increase of data traffic over voice drives the need for systems featuring multiplexing gain, scalability, and bandwidth efficiency. Two recent research projects, DAVID (data and voice over D-WDM) [1] and HORNET (hybrid optoelectronic ring network) [2], have investigated ways to exploit WDM in ring topologies. They both design and implement flexible and dynamic metro systems devoting one wavelength to control information, on which all packet access (ADD–DROP) decisions are based. This is also the strategy employed in optical burst switching in wide area networks (WANs) [3] but at the expense of high inefficiency and loss. In contrast, in the distances of a metro network it is possible to exchange reservation information quickly, thus achieving both high efficiency and low latency, while still avoiding any buffering in the optical domain as well as any optical-to-electrical conversion of data.

The European Information Society Technologies (IST) DAVID project [1], on which this research is based, has developed a slotted WDM packet-mode metro system consisting
of multiple WDM rings interconnected at a central point, the hub, featuring fully dynamic traffic control for each slot and flexible access mechanisms at a speed of 10 Gbit/s per wavelength. These mechanisms support different node configuration schemes in terms of number of transceivers, allowing for a smoother migration toward advanced optical switching networks in the cost-sensitive metro area, as elaborated in Ref. [4]. Two types of controller are needed: the node controller, to regulate the node access to the ring slots, and the hub controller, to schedule slots from one ring to another interconnecting several rings. The implementation architecture of the access controllers is the focus of this paper. A simplified version of the controllers, still allowing for the validation of the concept, was implemented in hardware and integrated in the DAVID demonstrator [5].

The paper is organized as follows: The architecture of the metro system and the functionality of the controllers are described in Section 2. The implemented architecture for both the node and hub controller is presented in Sections 3 and 4, respectively. Finally, conclusions are drawn in Section 5. (A preliminary version of this research was presented in Ref. [6].)

2. Controlling a Bufferless Metro System

The architecture of the DAVID packet-switched metro network is shown in Fig. 1. It consists of R WDM rings, which can exchange slots at the hub, via a bufferless optical space switch [7]. The ring nodes interface to edge routers or switches at the end of access networks via a variety of interfaces (e.g., Gigabit Ethernet in business areas, passive optical networks in mixed or residential areas, cable head-ends, or any other legacy system). Every node transmits data in one of the W available wavelengths $\lambda_1 - \lambda_w$ on each ring at a rate of 10 Gbit/s in fixed size slots. An additional wavelength $\lambda_0$ is used to transfer the control information, also traveling in fixed size slots of equal duration with the data slots. In the full architecture, rings are connected to a mesh of packet-switched optical ADD–DROP multiplexers (ADMs) in the core, thus creating a complete optical WAN [1]. However the WAN part of the DAVID project is outside the scope of this paper, which focuses on the MAN.

The overall system works as a combined wavelength–time–space distributed multiplexer. A fixed optical slot duration and format is used for all kinds of encapsulated traffic to facilitate an easy burst mode operation and optical switching at the hub. To adjust the arriving payload to the fixed size slot, an aggregation–segmentation unit creates payload with a suitable size, which can have a duration of the order of a few microseconds. The time slots on all wavelengths of the same ring are aligned in order to form a multislot. There is a locked timing relationship between the time slots in the control wavelength and the time slots in the data wavelengths, with a one-to-one correspondence. All addressing resides in the control slot, and a time shift of one slot is used to allow for processing the contents of the control slot ahead of the arrival of the data multislot, as shown in Fig. 2.

Each slot in the control channel carries information for all the W supported wavelengths in the data multislot and consequently consists of W individual wavelength-control regions, which are further divided in subfields, as shown in Fig. 3.

At each ring node, a medium access control (MAC) protocol is employed to arbitrate access to the slots regulating both the time and wavelength dimensions [wavelength-division multiple access (WDMA) with time-division multiple access (TDMA)]. To achieve collision-free operation, transit traffic has priority over local traffic; i.e., every node inserts traffic only to empty slots, destined to the appropriate ring. The $S$ field indicates the status of the slot ($S = 1$, corresponds to occupied slots), DA the destination node address, and SA the source node address. The logic actions executed over the subfields of each individual wavelength control can be divided in the ADD function and the DROP function. The reception of a data slot (DROP function), occurs when a node has detected its own MAC
address (unique for each node) in the destination address field. Then the status bit is reset, setting the slot free, and the source address is forwarded to the data manager. Data transmission (ADD function) is decided only when an empty \((S = 0)\) slot has been detected, and in this case, the MAC addresses of the destination (DA) and the source (SA) nodes are inserted in the control payload while the status bit \((S)\) is set to 1, indicating a data slot that is no longer empty. [More details on the rest of the fields, including the destination ring (DR), are provided below.] The end result is that all the information needed by a node to decide a data packet \(\text{ADD–DROP}\) action resides in the control channel, thus making it sufficient to terminate only the control payload and process it in all transit nodes, until the respective data payload has reached its destination.

In the hub, to achieve a bufferless exchange of traffic from ring to ring, a combination of space switching with wavelength conversion is employed, exploiting the information on the control channel. The hub defines input–output combinations of slots (permutations), which the space switch at the hub will bring out a round trip later. These input–output permutations change every one slot duration time. When integral multislots from ring to ring are switched at the hub, the output rings are stored in a matrix (called hereafter “permutation matrix”), which has \(R\) rows (one per input ring), and every column corresponds to one slot. As shown in Fig. 1, input rings 1, 2, 3, 4 will be switched to output rings 1, 3, 4, 2, respectively, one round trip time after slot \(i\) and to output rings 2, 3, 4, 1 one slot later. Once these permutations have been decided, they are announced in the individual wavelength-control region, namely, the DR field by the hub controller. Per destination ring queuing of traffic is employed at the ring nodes, to avoid head-of-line blocking. Thus, when a data slot destined to ring \(j\) awaits transmission in a node’s queue, this can be inserted into any empty
Fig. 2. Relation between the control and the data channels in the metro ring.

Fig. 3. Control slot format.

The hub controller responds to traffic fluctuations when deciding these permutations, on the basis of information collected from all nodes about the amount of traffic waiting in the queues. One field per destination ring (called explicit reservation field) is implemented in the reservation region of the format shown in Fig. 3. As described in Ref. [8], each node adds to the explicit reservation field it receives, the length of its own relevant queue, expressed in slots. As a result, when the control slots arrive back to the hub, each of the $R$ fields of the reservation region contains the aggregate of all slots waiting at the queues of the traversed ring (say, $n$) and destined to the corresponding ring. At the hub a reservation matrix is maintained as follows: the $R$ fields of the reservation region of ring $n$ are stored in column $n$ of the matrix, implying that an equal number of slots are required for serving the queued slots. This is an $R \times R$ matrix, and each element $R[n,m]$ contains the most recently announced information about the number of slots waiting at the queues of ring $n$ and destined to nodes of ring $m$.

To decide the ring-to-ring permutations, the following simple heuristic scheduler was implemented to allow an evaluation of the architecture. The reservation matrix $R[n,m]$ is inspected, and the output ring $m$ is assigned to input ring $n$ if $R[n,m] = \max \{ R[n,j] \text{ for } j = 1 \text{ to } R \}$ (given that $m$ has not been assigned to another ring for the same time slot). Thus, the ring-to-ring throughput changes dynamically, adapting to traffic demands. For the proper operation of the multiring system the interconnected rings have to be of the same length so that the slots are synchronized when they arrive at the hub. The access delay performance of the system is of the order of just a few round-trip times,
thanks to the good knowledge of the distributed multiplexing situation gathered by the explicit reservations and to the local action of the empty slot access protocol. A detailed performance evaluation can be found in Ref. [8], whereas in this paper the focus is on the implementation.

The control slot may include additional fields such as the payload priority (P) and class reservation (CR) field, allowing for the implementation of algorithms enforcing quality-of-service differentiation, as proposed in Ref. [9]. Leased line emulation is possible marking prearbitrated (PA) slots and the address (PA-Address) of the node that is allowed to use them. Additional fields enabling the enforcement of fairness mechanisms (such as those proposed in Refs. [10, 11]) may also be included in the fairness enforcement and reservation region.

The collision-free data transport scheme cannot be extended beyond the reach of the ring topology, so buffering of the payload is inevitable in the gateway toward the WAN; hence all data are converted from optical to electrical and back [1]. This is however outside the scope of this research, which deals with the metro part only.

The format of the control slot is as follows: 32 bits are used to code each individual wavelength-control region. This is analyzed in 8 bits for each of the destination, source, and prearbitrated address fields (allowing for the support of 256 nodes), 4-bit-wide destination ring field (allowing for \( R = 16 \) metro rings, respectively), thus leaving 1 bit to code each of the status, priority, class reservation, and prearbitrated slot indications. Hence 24 bits are dedicated to address fields in total, 4 for the destination ring and another 4 for control of the slot. For 32 data wavelengths, this leads to 1024 bits of information per slot for the wavelength control region. Even for a slot duration of 1 \( \mu s \) at 2.5 Gbit/s, there is plenty of room for error-correction fields and for the above-mentioned information for fairness mechanism enforcement. The slot time implemented in the demonstrator was 1 \( \mu s \) to stress the electronic processing, but any longer slot duration may be chosen.

The processing requirements imposed on the MAC controller of the ring node include the execution of the empty slot access protocol for all W wavelengths in each slot. On the other hand, during each time slot, the hub controller has to copy all the packet-related information from the arriving control slots of all rings to the appropriate position in the control slot of the destination ring—as necessary to preserve the data slot and control slot correspondence—to control or configure the optical switch matrix. It also has to update the destination ring (DR) field in the outgoing control slots. These constraints dictate their implementation in hardware. The time required for processing the control channel either in the ring nodes or at the hub is of the order of few microseconds. An equal delay is introduced in the data slots so that control and data slots remain synchronized.

3. The MAC Controller at the Ring Nodes

3.A. Architecture

The DAVID ring node includes burst mode transceivers (BMTs), operating at 10 Gbit/s, traffic manager board(s), and a node control board where the MAC controller resides. The minimum node configuration (with only one data wavelength) is given in Fig. 4. It includes two BMTs, one for the control channel and one for the data wavelength, one traffic manager board (TMB), and a node control block (denoted as NCB in the figure). Traffic is injected into or received from the metro through legacy network interfaces, such as Gigabit Ethernet.

The BMTs are responsible for OEO conversions and provide or receive the payload of the control and data wavelength through a parallel (16-bit-wide) interface in each direction operating at 155 MHz. In the TMB, the traffic format adaptation takes place; hence it includes the data manager and the necessary memory modules. The node control board includes the MAC controller and a microprocessor used for its configuration—i.e., the current
node address and the supported wavelengths are defined as well as the contracted service parameters—depending on the employed protocols. The MAC controller communicates with one BMT to receive and transmit the control channel contents and with one or more TMBs, to trigger data transmission or reception. (In the DAVID demonstrator, the TMB and the NCB were integrated in a single board.) Through the data manager interface, the latter informs the MAC controller about new-formed data packets awaiting transmission delivering at the same time the destination node address. In the other direction, the MAC controller provides the data manager with the wavelength and the destination or source address of the data payload to be transmitted or received.

The internal organization of the MAC controller is shown in Fig. 5. To allow control slot reception, processing, and transmission to be performed in parallel, the control slot content memory consists of three different regions (32 × 32 each), each capable of storing one control slot content. The storage of the incoming control slot and transmission of the prepared control slot content toward the BMT is undertaken by the so-called BMT part.

When new control slot content has been stored, the MAC controller entity is triggered and starts processing the individual wavelength control fields one by one, executing the empty slot access protocol. To make an ADD decision, it uses the information received by the data manager and stored in the “add traffic info” memory, following the discipline of
data packet queuing, i.e., per destination ring. On the other hand, a packet drop is possible only when there is a packet with a destination address equal to the node MAC address, in which case the data manager is notified. Assuming a tunable transceiver, the MAC forwards to the data manager the wavelength where the ADD-DROP function will take place and the destination–source address of the packet as well as the priority, if priority distinction is employed.

3.B. Implementation Issues

An important advantage of the presented scheme is that it can support different metro network architectures in terms of node configurations. Ring nodes can be equipped with a set of fixed receivers and transmitters, equal to or less than the supported wavelengths. To cover all the possible combinations, two control registers indicating the transmission and reception wavelengths, respectively, are employed by the MAC controller entity driving it to execute the ADD function part of the logic described in Section 2 on the transmission wavelengths and the DROP function on the reception ones. For example, a node can be equipped with one fixed transceiver (i.e., the transmission control register is identical to reception control register with a single 1 only) in which case only one individual wavelength-control field undergoes processing. If more fixed transceivers exist, the same logic is executed an equal number of times over the defined wavelengths.

Another option is to decouple transmission (characterized as upstream) from reception (called hereafter downstream) wavelengths, which allows for short-term cost-effective implementation as described in Ref. [12]. This has also been adopted for the DAVID demonstrator, where one upstream and one downstream wavelength were implemented. In this case, the MAC controller inspects the upstream wavelength in order to add data while it checks the downstream wavelength so as to receive data.

As regards latency, it takes just a few clock cycles per transceiver for processing. For the reception, the control slot memory is accessed two times (one for read and one for write), while for transmission the “add traffic info” memory is also accessed. These memories are implemented on chip, owing to their limited size; hence processing for either reception or transmission takes three clock cycles. (The internal clock used was 77.76 MHz.) To maintain the synchronization between control slots and data, the introduced latency is in multiples of the slot time. To support up to 25 (32) wavelengths, the introduced latency is one (two) slot time(s). If a more sophisticated MAC protocol is employed, the required processing time may exceed the slot time. In this case, to keep the processing throughput equal to one control slot content per slot time, pipelined techniques may be applied, introducing a need for an extra control slot content memory region (i.e., the memory requirements increase by 1024 bits.) Even for this very short slot and implementation in field programmable gate array (FPGA) technology, the control slot processing time remains of the order of few microseconds, which is almost negligible when the propagation time for a ring of 80 km is 400 $\mu$s. For an ASIC (application-specific integrated circuit) implementation, a clock frequency of 155 MHz will be used for internal MAC control operation, allowing for the support of 32 or even more wavelengths with one slot latency.

The organization and handling of pending traffic information greatly affects both the memory requirements and the performance. Since per destination ring queuing is employed, the MAC controller needs to keep $R$ counters, containing the number of packets that wait to be transmitted toward each of the $R$ interconnected rings. These are increased by one every time a new packet is formed and decreased by one each time a transmission decision is made. But when a packet is added in the metro ring, the relevant destination address must also be inserted into the DA field of the control slot. This address can be announced to the MAC controller either when the payload has just been formed or (later) when the relevant transmission has been decided. The first implies per-packet information
storage inside the MAC controller, increasing the memory requirements, whereas the second increases the introduced latency, since upon empty slot detection, the MAC controller communicates with the data manager to obtain the DA to be written in the control slot. (The second one was adopted in the DAVID demonstrator.) In any case, the maximum amount of information exchanged between the MAC controller and the data manager in each slot is related to one ADD and one DROP action, which can be coded in 29 bits, allowing for the use of serial interfaces.

To this end, an equivalent gate count of 460,000 was reported by the synthesis tool while the memory requirements, stemming only from the control slot content storage, are just 3 Kbits. The results show that the presented MAC controller is a flexible, low-cost component capable of supporting different node configurations.

4. HUB Controller Design

4.A. Architecture

The hub controller is implemented in an FPGA placed on a dedicated board, which interfaces to:

- several BMTs (equal in number to the interconnected rings) to receive or transmit the control channels;
- the gateway, for the exchange of information on packets from or to the WAN; and
- the space-wavelength switch.

On the same board, a microprocessor for the configuration of the controller is located. The same burst mode transceivers are used for the node and the hub. The interface between the hub controller and the gateway board is similar to the one between the data manager and the MAC controller at the ring node, since the functionality is the same: packets ready to enter the metro rings are announced and are transmitted upon hub controller’s permission. The throughput of this interface is low, and a serial interface is used. The interface with the switch carries in every slot \( W \times R \times (\log_2 W + \log_2 R) \) bits to define the output fiber and wavelength for every input fiber and wavelength. For \( W = 32, R = 4 \) and slot duration = 1 \( \mu \)s, the interface throughput is 896 MHz. (For the DAVID demonstrator, a parallel 16-bit bus interface operating at 155 MHz was used.)

The internal organization of the hub controller supporting two metro rings is shown in Fig. 6. One block is responsible for the content of each control channel and is denoted as “ring \times MAC controller” in the figure. At system setup phase, the “init” blocks generate the first control slots, setting all fields to zero except from the DR fields, to allow for traffic insertion. Once these slots arrive back at the hub, these entities become inactive and the hub controller enters its normal operation. At this time, it has three processing tasks:

First, the control information corresponding to each data packet has to be copied (switched) to the control channel of the DR and in the appropriate position (corresponding to wavelength), so that control and data remain in correspondence. For example, if data slot \( i, j \) with \( i \) indicating the fiber and \( j \) the wavelength is switched to data slot \( m, n \) then the packet’s destination and source address from the control channel \( j \) has to be copied to control channel \( m \) at the \( n \)th position corresponding to the \( n \)th wavelength. To do so, once the control information has been stored in the control slot memory, exactly as happens in the ring node, the ring MAC controllers are triggered. These operate in a synchronized way (thanks to “synchro” entity) in order to exchange the data-related information (destination address, source address, priority) through the “i_xchange” blocks.

The second processing step includes the exchange of information between the MAN and the WAN. The gateway is assigned a ring ID. When this ID appears in the permutation
matrix, the system checks whether there are packets at the gateway awaiting transmission, and the “GW MAC” entity is triggered. Information about the traffic waiting at the gateway to be inserted into the metro rings is kept in the “GW queue status block,” in the same way it is kept at the “add traffic info” block at the node. In case that a decision for packet forwarding from the WAN to MAN is made, the relevant information is delivered to the appropriate ring MAC controller via “i_xchange” block, and the gateway board is also notified.

Finally, the ring MAC controllers insert the DR information in the control slots and configure the optical switch, according to the content of the permutation matrix. The permutation matrix can either be programmed by a microprocessor, keeping ring-to-ring capacities fixed, or be dynamically updated by a scheduler, hence varying ring-to-ring capacities to adapt to temporal traffic properties. The scheduler bases its decisions on the \((R \times R)\) reservation matrix where the reservation fields of all the arriving control slots are stored. While the scheduler defines all input–output permutations, each “ring MAC controller” block does not need to know the whole permutation matrix but the row(s) corresponding to it, denoted as \(P[1]\) for ring 1 and \(P[G]\) for the gateway, in Fig. 6. The complexity of the scheduler depends on the employed algorithm.

4.B. Implementation Issues

The memory requirements include the control slot content memory requirements, the reservation matrix, and the permutation matrix storage requirements. As regards the permutation matrix dimensions, different options exist, depending on the system architecture. Assuming that the hub switches multislots from ring to ring, the number of rows is equal to the number of the attached rings \(R\), as shown in Fig. 1. Otherwise (as was the case for the DAVID demonstrator), the number of rows is equal to \(W \times R\).

When the permutation matrix is programmed (and not dynamically calculated), each column corresponds to one slot and columns are cyclically inspected. The number of
columns should be at least equal to the number of rows, in each case, to allow for full connectivity. In this case, the switch throughput is equally shared to all permutations. The minimum throughput that can be assigned to a permutation pair is \((10 \text{ Gbit/s}) / T\), where \(T\) is the number of columns.

The processing throughput of the hub controller is equal to one control slot per ring per slot time. The complexity of the design is very low: the gate equivalent is approximately 420,000 per supported metro ring assuming that the permutation matrix is programmed.

To support a high number of metro rings (eight or more), the termination of the control channels may be performed in more than one board, since one BMT interface per ring is required. The exchange of information between collocated boards is still possible, but delay is introduced. The situation will be dealt with in the same way as for the ring node.

5. Conclusions

Within the distances of metropolitan areas, the limited round-trip delay allows a reservation approach in realizing on-the-fly all-optical switching. This makes feasible a WDM network with high efficiency despite the lack of buffering except at the entry. The architecture of interconnected slotted WDM rings controlled in the electrical domain on the basis of a control wavelength can cover any size of MAN decoupling the control into two algorithms: one controlling the access to ring slots and the other scheduling the exchange of slot permutations among the rings in the hub on the basis of cross-ring reservations. The use of fixed slots facilitates both the access to the rings and the complex control scheduling among the rings, allowing a simple implementation of the two controllers in fast \(H/W\). Their integration in the DAVID demonstrator achieving low latencies and flexibility in system architecture and configuration adds a significant step toward the realization of optical-packet-switched metropolitan networks.

Acknowledgment

The research presented in this paper was funded in part by the EU project IST-1999-11742 “DAVID.”

References and Links


