Extending the Fasnet protocol to WDM rings for short-term fairness control

A. Bianco, D. Cuda, J. M. Finochietto, F. Neri, and C. Piglione
Dipartimento di Elettronica, Politecnico di Torino, 10129 Torino, Italy,
Email: {andrea.bianco, davide.cuda, jorge.finochietto, fabio.neri, chiara.piglione} @polito.it

Abstract—Single-hop wavelength division multiplexing (WDM) optical ring networks operating in packet mode are a promising architecture for the design of innovative Metropolitan Area Networks (MANs). They allow a cost-effective design, with a good combination of optical and electronic technologies, while supporting features like restoration and reconfiguration that are essential in any metropolitan scenario. In this article, we address the fairness problem in a WDM optical network. We introduce the Multi-MetaRing and the Multi-Fasnet fairness protocols, aiming at achieving high aggregate network throughput, throughput fairness and bounded and fair access delays.

I. INTRODUCTION

MANs are considered by many researchers the possible arena for moving the telecom market out of the last-years’ crisis. Nowadays, the metropolitan market is being driven by the demand for new application services (Video-Streaming, Video-on-Demand, Videoconferencing) and the introduction of high speed access. Since MANs are responsible for transporting traffic between different Access Networks (AN) and Wide Area Networks (WAN), they are characterized by high dynamism of traffic patterns, relatively high aggregate bandwidths, and relatively short covered distances. Moreover, since they also offer critical high-value services, they must be reliable and fault tolerant.

Today’s deployed MANs are mainly circuit-based Synchronous Optical Network/Synchronous Digital Hierarchy (SONET/SDH) rings, where optical technology is used exclusively to support point-to-point connections between nodes, which means that each node performs optical-to-electrical conversion, and must electronically process the entire traffic for routing/switching. Besides, the static bandwidth allocation and network monitoring requirements increase the total cost of a SONET/SDH network without being able to accommodate the increasing amount of data traffic in MANs. In fact, although these solutions are highly reliable, and guarantee network management and fault recovery mechanisms, they do not scale well and are cost inefficient due to the complexity of implementing switching devices completely in the electronic domain and with several input/output ports operating at 10 Gbit/s and above. Moreover, being circuit based architectures, they are optimized for voice and not for data traffic.

Optical Packet Switching architectures are excellent candidates to meet the requirements of more dynamic and demanding future MANs. In particular, optical packet switched ring networks can support the increasing traffic demand while warranting the reliability typical of a ring topology.

The choice of an optical packet switched ring network leads to the problem of designing an efficient MAC protocol in order to arbitrate the access of the nodes to the channel resources. An efficient MAC protocol must be able to optimize network throughput, controlling under which conditions packets can be transmitted from the queues of network interfaces. The challenge is to obtain high network utilization while minimizing delay and blocking probability and, simultaneously, providing acceptable efficiency/fairness trade-offs.

The paper is organized as follows. In Section II, we describe the network architecture under study, while in Section III we introduce the fairness problem arising in this architecture and motivate the need of a fair access protocol. Next, in Section IV we describe the MetaRing protocol and its adaptation to our network architecture, while in Section V we do the same for the Fasnet protocol. In Section VI we present simulation results of the proposed access protocols and discuss their performance and limitations. Finally, we draw some conclusions in Section VII.

II. SYSTEM MODEL

We consider a specific WDM optical packet network, whose architecture was proposed, studied and prototyped in the framework of the Italian national project called WONDER [1]. The architecture of the WONDER [2] network is depicted in Fig. 1, while the structure of a node is illustrated in Fig. 2. The WONDER architecture comprises $N$ nodes connected to two counter-rotating WDM fiber rings. Each ring conveys $W$ wavelengths, with $N > W$; each ring is used in a specific way: one ring is used for transmission only, while the second ring is used for reception only. Transmission wavelengths are switched to the reception path, at a folding point between the two rings, as shown in Fig. 1. During the first ring traversal, transmitted packets cross the transmission ring until the folding point, where they are switched to the reception path and then received during the second ring traversal. This architecture presents the interesting fault-recovery characteristics typical of the ring architecture: if each node can become the folding point (i.e., each node has a switching capability), the network preserves the usual ring restoration properties, as described in [2], [3]. Moreover, although the folded bus topology causes the loss of the spatial reuse capability, it reduces the transmission impairment typical of all-optical ring networks, and, at the same time, assures that all the traffic accepted prior to a fault...
event can also be supported after the restoration (note that this may not be the case in ring networks with spatial reuse).

The network is synchronous and time-slotted. The slot duration is determined by technological constraints, such as tuning time and dispersion, by user packet sizes, and by the efficiency of the packet segmentation process. During a time slot, at most one packet can be transmitted by a node in one of the \( W \) available slots (one slot for each wavelength channel). Each node is equipped with a fixed receiver. Receivers are allocated to WDM channels in a way that equalize the traffic across WDM channels, as described in [4]. To provide full connectivity between nodes, each node is equipped with a \textit{fastly} tunable transmitter (implemented as an array of fixed lasers, as shown in Fig.2) and exploits WDM to partition the traffic directed to disjoint subsets of destination nodes; each subset is made by the nodes whose receivers are currently allocated on the same wavelength. Nodes tune their transmitters to the receiver destination wavelength, establishing a single hop connection lasting one time slot. The channel resource sharing is therefore achieved according to a Time Division Multiple Access (TDMA) scheme. Access decisions are based on channel inspection capability (similar to the carrier sense functionality in Ethernet), called \( \lambda \)-monitor. In this way, each node knows which wavelengths have not been used by upstream nodes in the current time slot. Therefore, by default, priority is given to in-transit traffic, i.e., a \textit{multichannel empty-slot} protocol is used.

III. MAC PROTOCOL AND FAIRNESS

From a design perspective, a suitable MAC protocol for the WONDER network must be adaptable to a multi-channel network, avoiding packet collisions and assuring some level of fairness together with acceptable network throughput and delays bounds. A collision may arise when a node tries to insert a packet on a time slot and wavelength which have already been used. This is avoided by giving priority to upstream nodes, i.e., to in-transit traffic, thanks to the \( \lambda \)-monitor capability (see Fig. 2).

A first level of fairness is achieved implementing an efficient a-posteriori [5] packet selection strategy and exploiting the Virtual Output Queue (VOQ) structure (see [6]). While single-channel network protocols use a single FIFO (First In First Out) electrical queue, in a multi-channel scenario the FIFO queuing might lead to performance loss due to the Head of the Line (HoL) problem: a packet at the head of the queue might block other packets which could be transmitted on other channels. The HoL problem has been carefully studied, and it has been demonstrated that it can be solved using one of the VOQ scheme described in [6], which is able to achieve 100\% throughput under uniform and unicast traffic. The basic VOQ idea consist in using separate queues, each one corresponding to a different destination, or to a different set of destinations (e.g. all the nodes receiving on the same wavelength), and to appropriately select the queue which gains access to the channel for each time slot. In the case of the WONDER network, where usually there is more than one node receiving on the same wavelength, there is no difference between adopting a queue for each destination (\( V \) queues) or a queue for each channel (\( W \) queues); for simplicity and without loss of generality we adopt the second solution.

A problem common to ring and bus topologies is the different access priority given to network nodes depending on their position along the ring/bus. Referring again to Fig. 1, it is easy to see that an upstream node can “flood” a given wavelength, as shown in [7], reducing (or even blocking) the transmission opportunities of downstream nodes competing for access to that channel, leading to significant fairness problems.

Due to the application to MANs, the network end-to-end propagation delay is fairly larger than the average packet transmission time. For example, a 50km span leads to 250\( \mu \text{s} \) propagation delay, while a 1000-bit packet at 1Gbit/s lasts 1 \( \mu \text{s} \).

In the following sections we present and adapt to the described network architecture two well known access protocols: the MetaRing and the Fasnet protocol. We outline their major limitations and we conclude proposing a new version of the Fasnet protocol which is able to achieve both high levels of
bandwidth utilization, even under heavy loads, and a high degree of fairness, significantly reducing the impact of the node position along the bus.

IV. MULTI-META RING PROTOCOL

MetaRing [8] (MR) is a well known fairness protocol proposed to overcome the problems arising with protocols that operate with large network cycles [9]–[12], where nodes can access into a new cycle only once the previous cycle is completed. Extensions of MetaRing were previously proposed by our research group for a multi-channel network, but always for a ring topology [13]. In this section we briefly describe the MetaRing mechanism in a folded bus topology, and its extension to a WDM multi-channel architecture, like WONDER.

MetaRing is based on the circulation of a control signal called SAT, standing for SATisfied, which usually rotates in the opposite direction with respect to the data traffic flow. Nodes are assigned a transmission quota, denoted by \( Q \), i.e., a maximum number of packets to be transmitted between two successive SAT visits. On SAT reception, each node normally forwards the SAT with no delay; however, nodes that are not satisfied, i.e., which were not able to transmit \( Q \) packets between two SAT arrivals, delay the SAT until they achieve satisfaction, i.e., either until they transmit their current quota \( Q \), or the node queue becomes empty.

Since WONDER is based on a folded bus, we need to adapt the Meta-Ring protocol accounting for the different network topology we have to deal with. We use one separate SAT for each channel, that is, each SAT regulates the traffic toward the destinations receiving on the corresponding wavelength. We call this protocol the Multi-MetaRing protocol (MMR). Each node transmits up to \( Q \) packets between two consecutive SAT arrivals on the channel controlled by the SAT. Once a node receives the SAT, it renews its quota and, once satisfied, it forwards the SAT to the upstream node; thus, SATs still logically rotate in the opposite direction of data traffic flow. SATs can be transmitted in-band using specific control packets; however, when the network is close to overload, this solution might lead to large SAT access delays when a node with lower priority has to forward a SAT, thus decreasing network performance. We refer to this solution as the in-band Multi-MetaRing protocol. To overcome this problem, we study the MMR performance when SATs are transmitted on a dedicated channel. However, dedicating a wavelength to SAT transmission requires an increase in the available network resources and results in a significant additional complexity in the node structure. We call this solution the out-of-band Multi-MetaRing protocol.

Since WONDER is a multi-channel network and since we choose to adopt a solution with a number of SATs equal to \( W \), it might happen that a node receives a SAT while it is already holding another SAT; we call this effect SAT collision. When a SAT collision occur, two different policies can be adopted: the last arrived SAT is either released or held. We refer to these policies as Released-SAT policy and Held-SAT policy, respectively. In the former case, the last arrived SAT is immediately forwarded to the previous node, and the quota \( Q \) is added to the current quota value \( Q_c \). Note that in the original MR description, the quota is always renewed, i.e., set to \( Q \) when the SAT is released. In Released-SAT MMR, when SAT cannot be used because of SAT collisions, the node quota is increased by \( Q \). To avoid to accumulate quota on underloaded channels, we bound the current quota \( Q_c \) in the following way:

- \( Q_c \) cannot be larger than the corresponding channel queue current length;
- \( Q_c \) cannot be larger than the maximum achievable quota, \( Q_M = k \times Q \), where \( K \) is a parameter set to 5 in the simulation results.

When the Held-SAT policy is adopted, the node can delay all received SATs, independently from the number of SATs it is already holding. The node forwards a SAT only once it achieves satisfaction on the corresponding channel. This policy might lead to remarkable performance losses; indeed, when a node is holding more than one SAT, since each node is equipped with a single transmitter, and therefore it can transmit only one packet at each time slot, some available slots may remain unused.

As highlighted in [13], the queue length and the value of \( Q \) affect protocol performance; these parameters must be carefully set to avoid significant performance loss.

A. Queue Length Effect

It is important to notice that the capacity of the \( W \) queues at node interfaces must be larger or equal than \( Q \). In fact, since a node is forced to release its SATs when the corresponding queue is empty, if the channel queue is significant smaller than the quota, the last nodes (nodes at the end of the bus) might be forced to release the SAT before completing the transmission of the \( Q \) packets. Note that last nodes have lower access priority; thus, all packets must be buffered in the node queue until a new SAT arrives.

B. Quota Effect

In overload conditions, the performance are ruled by the fact that all nodes are able to transmit up to \( Q \) packets between two SAT arrivals, and that all nodes are throttled when the SAT is released by the last node; throughput mainly depends on the ratio between the value of \( Q \) and the SAT propagation delay, which in turn depends on the network architecture and size. Thus, given \( Q \), the smaller the SAT propagation delay, the higher the throughput; on the contrary if the network dimension increases (the propagation delay increases too), a larger value of \( Q \) is needed.

Moreover, if we want to assure a throughput close to the maximum capacity of the network also when there is only one node transmitting on a given channel, we must ensure a quota greater, or at least equal, to the SAT propagation delay. If a smaller value of quota is adopted, a transmitting node would refrain unnecessarily from transmission simply to comply with the Multi-MetaRing rules.

Due to the WONDER architecture, the SAT propagation delay is related to the round trip time (RTT), the time required
to traverse the topology (either transmission or reception ring). The SAT propagation delay can be approximated by $N \times \text{RTT}$. Therefore, large values of quota are needed to achieve good efficiency; however, this means that a fair behavior can be reached only on a large time scale.

Besides, the quota value determines the maximum achievable throughput in overloaded conditions. In deep overload, most nodes are not satisfied, and hold the SAT to get satisfaction. Thus the network repeats cycles of single-node transmissions followed by a SAT propagation the the next node, equal, in our network, to one RTT. If we consider that SATs can be released without any delay (as in the out-of-band MMR case), the maximum throughput is affected by the cost of waiting for SATs to renew the quota and transmit all possible packets from the queue. Thus, $TH_{\text{max}}$ can be approximated as:

$$TH_{\text{max}} = \frac{Q \times N}{Q \times N + N \times \text{RTT}}$$

Note that if we introduce delays when releasing SATs (as in the in-band MMR case), the $TH_{\text{max}}$ value becomes an upper bound, since the actual throughput would be lower because nodes must wait longer for the SATs.

V. **Multi-Fasnet Protocol**

Fasnet [14] is an access protocol originally designed to guarantee fairness on a slotted dual bus topology. In the following subsections, we first analyze the protocol in a folded bus topology with a single channel; next, we adapt the protocol for a multichannel network like WONDER, and we conclude proposing some new strategies to overcome its main limitations.

A. **Single channel Fasnet protocol**

Fasnet is an implicit token passing protocol developed to efficiently use the channel capacity, providing a high level of fairness in resource sharing. To implement Fasnet, each node must be able to listen to the transmission channel except for the first node in the bus, dubbed the master node, which has to listen on the reception one. This can be easily implemented in the node architecture by simply giving to each node the possibility to switch its own $\lambda$-monitor between the transmission bus and the reception one. The master node, being the first node on the transmission bus, does not experiment any packet collisions on the transmission bus.

Fasnet provides fairness operating cyclically; each cycle is associated with a chained transmission of data called train. A train is composed by a locomotive, transmitted by the master node, and by all packets transmitted by network nodes after the locomotive. The master node starts a new cycle, transmitting a new locomotive packet, every time it detects the end of the in-transit train (i.e., an empty slot on the reception channel). We define the train length (TL) as the number of used slots which compose the train, i.e., the number of transmitted packets in the current cycle. After a new locomotive transmission, each node has the chance to access the ring at most for transmitting $Q$ (where $Q$ indicates the quota) packets. Indeed, when a node senses an empty slot on the transmission bus, it seizes the channel for a number of packets equal to the minimum between the quota and the number of packets in the queue. Once a node releases the channel (either for exhausted quota or for empty queue), it has to wait for the next train before attempting to access the channel again.

Each node is said to be in one of the following states:

- **idle**: it has no packets to transmit;
- **wait**: it is waiting for the transit of a new train;
- **defer**: it is deferring the packet transmission because of busy slots. The master node never enters in this state since it has no upstream nodes;
- **Begin Of Train (BOT)**: only the master node enters this state. When it senses the first empty slot on the reception bus after a train transit, it transmits a new locomotive, thus, starting a new cycle.
- **access**: when a node senses, on the transmission channel, the first empty slot on the current in-transit train, it renews its quota, and starts transmitting packets. Each time a node transmits a packet the current quota is decreased by one.

In other words, a node is **idle** as long as its channel queue is empty. If a train passes through the node while it is idle, this train is ignored. Upon a packet arrival the node enters in the **wait** state, where it waits for a new train transit. When it senses a train transit on the transmission bus, the node enters in the **defer** state and it stays there until it senses the first empty slot on the current train; this event causes a transition into the **access** state and the renewal of its quota.

While the node is in the **access** state, it can access the channel and transmit its packets. The node leaves this state according to one of the following events:

- it exhausts its quota;
- it empties its queue;

In the first case it returns to the **wait** state, otherwise it comes back to the **idle** state.

It must be noticed that Fasnet is not able to reach 100% throughput, due to the idle time between two successive cycles. This means that the maximum achievable throughput, when the network is overloaded (note that the overload point depends on the value of $Q$), is mainly affected by the ratio between the maximum train length, which is equal to $N \times Q$ and the cycle duration, which is equal to $N \times Q$ plus the time needed by the master node to detect the end of the current train. In the WONDER architecture, this idle time is equal to $2 \times \text{RTT}$ time slots. This implies that the maximum achievable throughput is given by:

$$TH_{\text{max}} = \frac{N \times Q}{N \times Q + 2 \times \text{RTT}}$$

As a result, the larger the value of $Q$, the larger the maximum achievable throughput. If we assume that the network is not overloaded, which means that a node empties its queue on the current train without exhausting its quota, we can easily
estimate the worst case delay. This happens when a packet arrives as soon as the node has released the channel; the node has to wait for the next train to transmit this packet. Therefore the node worst case delay can be the evaluated as:

$$D_{WC} \approx N \times Q_u + 2 \times RTT$$  \hspace{1cm} (3)$$

where $Q_u$ is the effective average quota used by a node, which can be evaluated considering that, under lightly loaded conditions, the throughput $TH$ is equal to input load $\rho$. Therefore from Eq. 2 we obtain:

$$Q_u = \frac{\rho}{1 - \rho} \times \frac{2 \times RTT}{N}$$  \hspace{1cm} (4)$$

We can immediately observe that $Q_u$ does not depends on the value of $Q$, but is a function of the input load and the network dimension, which means that the trains adapt their length to the network load.

The performance of the Fasnet protocol is limited both in throughput and in delay by the channel idle time needed by the master node to detect the end of the current cycle. We are in front of a trade-off: on the one hand, we want a large value of quota to achieve high throughput but, on the other hand, if we want to ensure low access delays, a low quota value is needed.

### B. Multichannel Fasnet protocol

In a multichannel network the Fasnet protocol is replicated over the different wavelengths, which means that there are $W$ trains, one for each channel, traveling across the network. Therefore, nodes can be in one of the states previously described for each channel independently. If, at the same time slot, a node can access more than one channel then a train collision happens. Since nodes are equipped with a single fastly tunable transmitter (see Fig. 2), i.e., they can transmit at most one packet per time slot. Thus, when a train collision occurs, nodes select the channel to which the longest queue is associated. To take into account the train collision effect, we need to add some state transitions; in particular:

- from the **defer** state to the **wait** state, if a train arrives while the node is already transmitting on a channel with a longer queue associated;
- from the **BOT** state to the **wait** state, if the master node starts a train while it is already transmitting on a channel with a longer queue;
- from the **access** state to the **wait** state if while a node is transmitting on a certain channel, a train on another longer queue channel arrives.

To estimate the throughput in a multichannel network, we need to take into account the traffic matrix; Eq. 2 becomes:

$$TH_{max} = \frac{1}{W} \times \sum_{k=1}^{W} \left( \sum_{i=1}^{N} y_{ik} \times Q \right) / \left( \sum_{i=1}^{N} y_{ik} \times Q + 2 \times RTT \right)$$  \hspace{1cm} (5)$$

where $y_{ij}$ is a control variable defined as:

$$y_{ik} = \begin{cases} 1 & \text{if } p_{ik} > 0 \\ 0 & \text{otherwise} \end{cases}$$

where $p_{ik}$ is the probability that node $i$ transmits on wavelength $k$. The worst case access delay on wavelength $k$ becomes:

$$D_{WC_k} \approx N \sum_{i=1}^{N} y_{ik} \times Q_u + 2 \times RTT$$  \hspace{1cm} (6)$$

Therefore, the Fasnet performance is also limited by the channel idle time in a multichannel network. To improve Fasnet performance, the main idea is to reduce the fixed penalty of having a channel idle for $2 \times RTT$ slots between two cycles, trying to retransmit a train without waiting the end of the current train to be detected by the master node. In the following sections, we propose two possible train retransmission strategies, which achieve larger network utilization and lower access delay.

### C. Fixed-Length Train Strategy

The first train retransmission strategy is called the Fixed-Length Train (FLT) strategy. On a given channel $k$, $k = 1, \ldots, W$, the master node has the possibility to retransmit a train every $C_k$ time slots, where $C_k$ is a counter initialized to $N \times Q$ each time a new train is transmitted on channel $k$. Since $C_k$ is decreased by one at each time slot, using the FLT strategy a new train starts if one of the following events happen:

- if $C_k = 0$;
- if the master node senses the first empty slot after the train reception, and there are no other trains propagating along the channel. This condition may happen only if $C_k$ is initialized to a value larger than $2 \times RTT$.

When using the FLT strategy, nodes access channels cyclically, like in a Time Division Multiplexing (TDM) scheme. The advantage of the FLT technique consists in the fact that, when the network is overloaded, no slots are left empty unless when train collision occurs. Problems might arise if the train length does not match the traffic scenario (e.g., under non uniform traffic patterns); in this case the trains are left partially empty and throughput losses are experienced.

### D. Dynamic-Length Train Strategy

In this section we present the Dynamic-Length Train (DLT) strategy. The DLT strategy tries to guess the “optimal” future train length by looking at the current train utilization, i.e., the percentage of used slots of the currently received train. Indeed, if trains are partially empty, then the train length can be decreased, augmenting the retransmission frequency. On the contrary, if trains are completely full, their length can be increased, lowering the train retransmission frequency.

The DLT mechanism allows the master node to retransmit a new train every $C_k$ slots; but, unlike the FLT strategy, the value of $C_k$ is variable. The value of $C_k$ is updated every
time the master node detects the end of a train considering the train utilization, i.e., the number of busy slots on the train. In particular, $C_k$ is updated in the following way:

$$C_k = \begin{cases} C'_k + k_1 \times C'_l, & \text{iff all train slots are busy} \\ C'_k - k_2 \times C'_l, & \text{otherwise} \end{cases}$$

where $C'_k$ refers to the value of $C_k$ in the previous cycle, and $k_1$ and $k_2$ are two parameters, set respectively to 0.3 and 0.1 in simulation experiments. Besides, it must be noticed that a new state transition must be added. In fact, all nodes except the master may exit from the access state because of a new train arrival; i.e., when nodes sense a busy slot while in the access state. In this case, the node returns from the access state to the defer state. We refer to this effect as Train Overlapping.

Unlike in the FLT strategy where no quota accumulation can occur, in the DLT strategy this becomes an important issue. In fact, nodes might lose some cycles because of the train overlapping effect. However, the current trains are completely full, and, according to the DLT mechanism, the future trains are retransmitted less frequently, so that their length increases; thus, starved nodes can seize the longer trains with their accumulated quota.

To limit quota accumulation, we bound the current quota $Q_c$ in the following way:

- $Q_c$ cannot be larger than the corresponding channel current queue length;
- $Q_c$ cannot be larger than the maximum achievable quota, $Q_M = k \times Q$, where $k$ is a parameter set to 5 in simulation results.

VI. PERFORMANCE EVALUATION

In this section we present performance results obtained by simulation considering a network with $W = 4$ wavelengths and a total of $N = 16$ nodes, for a total ring length of about 25km; more precisely, we considered a ring RTT equal to $121\mu s$. Slots last $1\mu s$, corresponding to a packet size of about 1250 bytes at 10 Gbit/s. The RTT is therefore equal to 121 slots. Each node keeps $W$ separate FIFO queues, one for each channel, with a queue size of about 120000, fixed size, packets.

Two different traffic matrices $T$ are considered: uniform traffic and unbalanced traffic. The element $t_{ij}$ of $T$, $1 \leq i, j \leq N$, represents the mean generation rate of packets from node $i$ to node $j$. In the uniform traffic pattern, the whole capacity of the network is equally shared by all nodes. Thus, $t_{ij}$ for the uniform traffic pattern is given by:

$$t_{ij} = \frac{W}{N} \frac{1}{N - 1}$$

where $\rho$ represents the normalized input load. In the unbalanced traffic pattern, named “1-server”, nodes are partitioned into two separated subsets: server $S$ and clients $C$. The server subset contains only the first node on the bus, named server; this position represents a worst case scenario. The server transmits at a high rate, equal to the capacity of one wavelength, with equal probability to the other $N - 1$ nodes belonging to $C$. The remaining network capacity is shared by client nodes; each client transmits $\frac{1}{N}$ of its traffic toward the servers and the remaining traffic to the other $N - 2$ clients with equal probability. In other words, $t_{ij}, 1 \leq i, j \leq N$, is given by:

$$t_{ij} = \rho \begin{cases} 0, & \text{iff } i = j \\ \frac{1}{N - 1}, & \text{iff } i \in S \land j \in C \\ \frac{W - 1}{N - 1}, & \text{iff } i \in C \land j \in S \\ \frac{W - 1}{N - 1} \frac{1}{N - 1}, & \text{iff } i \in C \land j \in C \land i \neq j \end{cases}$$

Fairness has been evaluated as the difference between the performance achieved by nodes in first and last position on the bus.

A. Multi-MetaRing Protocol

![Multi-MetaRing normalized throughput under uniform traffic scenario](image)

We first analyze the network behavior when the Multi-MetaRing protocol is adopted. We consider three different value of quota, as a function of the minimum quota $Q_m$: $Q = \{0.1 \times Q_m; Q_m; 2 \times Q_m\}$, where in our simulation...
close to or larger than the minimum quota. Multi–MetaRing protocol. On the contrary, when the quota is very low values of quota are not suitable for the all the nodes are starved because of the SA T propagation delay. Indeed, then the normalized throughput falls significantly. Indeed, /BF/BK/BJ

\text{and} /BP/BD/BL/BF/BI

\text{for input load ranging}

\text{the network is lightly loaded, we perform our simulations only}

\text{for input load ranging} \ 0.7 \leq \rho \leq 1.5.

\text{In Fig. 3(a), the normalized network throughput for the in-}

\text{band MMR case is shown when considering quota} Q = 387, \ Q = 1936 \text{ and} Q = 3872. \text{The larger the value of} Q, \text{the larger the}

\text{achieved throughput. The performance behavior is quite}

\text{different according to the value of quota. If the quota is too low}

\text{compared with the minimum quota} Q_m, \text{as in the case of} Q = 387, \text{then the normalized throughput falls significantly. Indeed, all the nodes are starved because of the SAT propagation delay. As expected, very low values of quota are not suitable for the Multi–MetaRing protocol. On the contrary, when the quota is close to or larger than the minimum quota} Q_m, \text{and the input load is not much larger than 1, the network throughput falls}

\text{because the SATs experience a large access delay, due to the}

\text{fact that SATs do not travel on a dedicated channel.}

\text{When the network is close to overload, the last node, which}

\text{usually is retaining a SAT because it is not satisfied, experiences a great number of SAT collisions. However, it cannot}

\text{immediately forward the unneeded SATs because of the busy}

\text{slots used by upstream nodes which released their SATs for empty queue. Therefore, the SAT propagation delay increases}

\text{and then network performance decays. On the contrary, when}

\text{the network is heavily overloaded, i.e., all the nodes are not}

\text{satisfied, no SAT collision occurs since all the nodes release}

\text{their SATs for exhausted quota.}

\text{Fig. 3(b) shows the normalized throughput in the case of the out-of-band MMR; the problems related to the SAT access delay can be overcome adopting a dedicated channel to transmit the SATs. As in the case of the in-band MMR protocol, with larger values of quota the network reaches larger normalized throughput, and the delay saturates for larger input load values. The normalized throughput reaches a maximum, then decreases a little, and finally saturates. This behavior depends on the number of starved nodes. When the network is overloaded (i.e., when all the nodes are not satisfied) the maximum theoretical achievable throughput} T_{H_{\text{max}}}

\text{can be evaluated by Eq.(1).}

\text{Fig. 4 shows the level of fairness provided by the out-of-}

\text{band MMR protocol when} Q = 1936. \text{Although the MMR}

\text{protocol is able to achieve acceptable level of throughput}

\text{fairness (in a large time scale), when the network is lightly}

\text{loaded, it does not provide any control on delays: the first}

\text{nodes always have useful quota to transmit their packets with no access delay, while the last nodes have to wait for empty slots, just as without any fairness protocol. More in}

\text{details, when no wavelength is overloaded, the MMR protocol}

\text{does not influence the network behavior, and thus does not}

\text{improve fairness in the packet delays observed by different}

\text{network nodes. This is due to the fact that if wavelengths are}

\text{not overloaded, it is very unlikely that sources delay SAT}

\text{transmissions because they need more transmission resources.}

\text{When the network is overloaded the saturation delay is about}

\text{\( \frac{Q}{Q} \times N \times (Q + RTT) \), where} Q_t \text{is the channel queue length.}

\text{We conclude by analyzing the impact of the SAT policy on}

\text{network performance. The comparison between the Released-}

\text{SAT policy and the Held-SAT policy is plotted in Fig. 5(a) when the quota is} Q = 3872. \text{Despite the fairness is absolute (see Fig. 5(b)), the network might experience a large normalized throughput fall; in fact, once SATs are synchronized, they travel together, since they are all forwarded to the same node (the upstream one in the transmission ring direction). As expected, due to the presence of W SATs in the network, the achieved throughput when the network is overloaded is about}\ \frac{\mu}{\gamma} = 0.25.

\text{Simulation results have highlighted that the MMR protocol}

\text{is not particularly well suited for the WONDER network; indeed, it presents some problems, which can be summarized as:}

\cdot \text{when the SATs are transmitted in band, the SAT prop-}
aggregation delay might increase because of a large access delay, since the SATs have to wait for an empty slot. We need a dedicated signal channel to solve this problem:

- the MMR protocol needs large Q, comparable with the $N \times RTT$ SAT propagation delay, to reach acceptable performance. This implies large time scale to achieve fairness and large queues;
- the MMR protocol does not affect the performance behavior when the network is under loaded, leading to large access delay unfairness;
- the MMR protocol performance are affected by the SAT collision effect, and a policy to handle the SAT collision is needed;
- throughput losses might be experienced.

B. Multi-Fasnet Protocol

In this section we present the results achieved using the Fasnet protocol, the FLT strategy and the DLT strategy. We perform our simulations considering two different values of quota: $Q = 10$ and $Q = 100$.

As usual, we start analyzing the results under a uniform traffic pattern. Fig. 6(a) shows how the Fasnet normalized throughput is affected by the value of the quota; as expected, larger value of quota achieve larger network utilization, since the idle time between two consecutive cycles has a lower impact if the train length increases. Moreover, it can be noticed, that the throughput behavior is quite different from the one observed using the MMR protocol; indeed, Fasnet exhibits the good property that, as the input load increases, the channel utilization increases up to a maximum after which it remains constant. In Fig. 6(b) the mean access delay is plotted. When the network is lightly loaded, the access delay is independent from the quota value; the train length depends on the input traffic and the network dimension. In overloaded conditions the train length saturates to $N \times Q$ and the access delay saturates to the queue length.

Fig. 7(a) compares the normalized throughput for the Fasnet protocol, the FLT strategy and the DLT strategy under uniform traffic scenario when $Q = 100$; both the FLT strategy and the
DLT strategy improve the network utilization.

Under a uniform traffic scenario, where the network resources are equally shared by all the network nodes, the FLT train length is matched to the traffic; thus, a throughput equal to 1 is achieved when the network is overloaded. When the input load is close to 1, train collisions occur and some bandwidth is wasted. More in details, when the network is close to the overload, it might still happen that a node releases the channel because of empty queue, thus introducing, desynchronization between trains and collisions. On the contrary, the DLT strategy presents some throughput losses with respect to the FLT strategy; the trains are left partially empty since their average length is larger than the optimal one (equal to $N \times Q$ under uniform traffic pattern).

The access delays of the three considered protocols are plotted in Fig. 7(b); the DLT strategy is able to reduce the access delay of about one order of magnitude when the network is lightly loaded. More in details, when the network is lightly loaded, the trains are left partially empty. According to the DLT strategy rules, the master node boosts up the train retransmission frequency, reducing the train length; all the nodes can access the channels more frequently, decreasing their mean access delay.

Fairness performance are shown in Figs. 8 and 9, for Fasnet and the DLT strategy respectively. As expected, Fasnet provides an absolute level of fairness both for throughput and access delay; indeed, all nodes have the same access probability in the current cycle. The DLT strategy maintains the very good fairness level of throughput achieved by the Fasnet protocol, and reduces the access delay significantly at low loads. However, it presents some delay unfairness when the network in lightly loaded. This is mainly due to the train overlapping effect, i.e., the last nodes might lose some cycles when the trains are retransmitted too frequently. More in details, thanks to the quota accumulation, the last nodes are able to seize the longer train when they lose some cycles, thus achieving large throughput; however, when trains overlap, node have to wait for more than one cycle to empty their queues, and access delay increases.

Let us compare the Fasnet, the FLT strategy and the DLT strategy under the 1-server traffic scenario. Fig. 10 shows the normalized throughput and the mean access delay. Although the FLT strategy is very efficient in a uniform traffic scenario, its performance are quite limited under unbalanced traffic conditions. The FLT strategy allocates statically the resources, i.e., it gives to all nodes the same probability to access the channel during a cycle, independently of their input load. This allocation is completely matched under uniform traffic scenario, but when the traffic is unbalanced the trains are left partially empty, so throughput losses are experienced.

On the contrary, the DLT strategy is able to match the train length to input traffic pattern, then, achieving larger value of throughput. Performance increase is more visible if we compare the server and the client throughput. Fig. 11 shows the server and the last node throughput for the Fasnet protocol and the DLT strategy; using the DLT strategy the
server is able to achieve larger value of throughput (about 0.75) before being starved. Clearly, when the network is overloaded, the throughput of all the nodes converges to the same value according to the max-min fairness paradigm.

The simulation results show how the Fasnet performance are mainly limited by the channel idle time between two consecutive cycles; thus, Fasnet needs large value of quota to reach large normalized throughput, while the FLT strategy highlights the need to match the train length to the traffic scenario. Both the problems can be overcome adopting the DLT strategy, which is able to achieve large value of throughput assuring, at the same time, low access delay; while keeping the good fairness properties shown by the Fasnet protocol.

VII. CONCLUSIONS AND FUTURE WORK

The work in this paper was motivated by the idea that single-hop optical packet switched networks can be devised as a cost-effective solution to fulfill future traffic demands of the metropolitan area. Among the challenges that these networks introduce, the definition of access protocols that can guarantee both high throughput and good fairness is crucial to demonstrate the efficiency of these networks in handling data at the packet level.

In this paper, we presented well-known fairness protocols and discussed its possible adaptations to a specific network architecture. Different scenarios were simulated and results were analyzed. In particular, we have concentrated our analysis in the limitations that these protocols offer in terms of throughput and latency. Moreover, we have introduced a new protocol based on Fasnet that has shown high throughput and good fairness properties.

As future work, we intend to investigate the use of these protocols for handling variable-size packets. If variable-size packets are considered, the actual network throughput could
be improved since bandwidth waste can be minimized. Note that when considering fixed-size packets, packets smaller than one time slot waste part of the slot bandwidth. Variable-size packets can be efficiently transmitted as sequences of consecutive time slots [15].

Another issue for future work is the analysis of implementing a network similar to the one considered in this paper, but with asynchronous access, in the sense that slotted access is enforced and no time reference must be shared among nodes. Fasnet appears to be particularly suited to this scenario.

**ACKNOWLEDGMENT**

This work was performed in the framework of the WONDER [1] project partly funded by the Italian Ministry of University, Education, and Research (MIUR).

**REFERENCES**


