1998 International Computer Symposium
Workshop on Computer Networks, Internet, and Multimedia
December 17-19, 1998, N.C.K.U., Tainan, Taiwan, R.O.C.

Architecture and Analysis of Accelerative Pre-Allocation Protocol for WDM Star-Coupled Networks

Chuan-Ching Sue, Wen-Yu Tseng, and Sy-Yen Kuo

Department of Electrical Engineering
National Taiwan University, Taipei, Taiwan
Email:sykuo@cc.ee.ntu.edu.tw

ABSTRACT[†]

In this paper, a new media access control (MAC) protocol, the AP-WDMA (accelerative pre-allocation), is proposed for the WDMA. Through examples and evaluations, the AP-WDMA is demonstrated to be able to reduce the delay through the accelerative mechanism. The degree of delay reduction depends on the traffic load. Higher degree of locality will have higher delay reduction. The AP-WDMA relieves the technology constraints by reducing the tunability to only one end and the table size to only n+2 units of memory space, where n is the number of stations. In addition, it is suitable for wavelength-limited networks. Therefore, the proposed AP-WDMA is indeed a better WDMA protocol.

I. INTRODUCTION

During the past decades, the optical technology has been widely used in the high-speed long distance communications and local area networks (LAN's) [1-3]. For general applications, the wavelength division multiplexing (WDM) [4-11] is the most popular and efficient technology investigated so far to efficiently utilize the very high data rate and the vast bandwidth provided by an optical fiber.

The reservation approach and the pre-allocation approach are two main media access protocols developed for this purpose [12-17]. The pre-allocation WDMA (P-WDMA) is simpler and performs better than the reservation WDMA (R-WDMA) [16,17] under uniform and high traffic load. The P-WDMA has lower implementation and operation complexity; in addition, the R-WDMA has poor performance due to too many collisions and retransmissions when more stations are requesting the same data or control channels. Although several papers were presented to improve the performance of the R-WDMA [11,18] and overcome the above problem, they only illustrate the protocol application on specific WDM architectures with technical constraints like tunability and large tables. The P-WDMA is not always efficient. For example, the time-interleaved pre-scheduling is the most popular and well-developed method for the P-WDMA. With this method, first if the traffic is not uniform across all source-destination pairs, some stations and channels may remain idle all the time while others are waiting for the right time slots and result in great delays. Second, if the source wants to transmit the data just after the pre-scheduled time slot that it can communicate with the destination, it has to wait for the next pre-scheduled time slot for transmission. In both cases, we will have the critical drawbacks of lower channel utilization, lower throughput, and higher delay.

Although several papers have proposed the P-WDMA protocols with optimal performance [21-22] under the consideration of tuning time, processing time or propagation delay, they all assume that the traffic load is always fixed so that they can schedule well according to the fixed load information. In the realistic situations, the traffic load is changeable between uniform and non-uniform or high and low. Thus a low complexity protocol is required to adjust itself to the change of traffic load. In this paper, a novel multi-access phonic protocol, AP-WDMA is proposed for the WDMA. It provides a better media access protocol that is based on the P-WDMA but can eliminate its drawbacks and improve the system performance with the least cost. The protocol enables the source station to possibly transmit the data earlier than the pre-scheduled time slot in the above two cases through the known transmission information and the additional control channel. Even if the earlier transmission can not be achieved, the AP-WDMA still works well as P-WDMA. Thus, it always retains the P-WDMA advantages and improves the performance of P-WDMA. Furthermore, through channel sharing mechanisms, there is no theoretical upper bound on the number of stations that can be added given a fixed number of channels.

The rest of paper is organized as follows. In Section II, we define the architecture considered in this paper and the important terms. The novel media access protocol, AP-WDMA, is proposed in Section III. Section IV depicts how the traffic load affects the reduction of delay. Finally, we conclude our study in Section V.

II. PRELIMINARIES

In this section, we will illustrate the foundations for the AP-WDMA. Subsections A and B give the architecture definition and the terminology, respectively.

[†] **Acknowledgement:** This research was supported by the National Science Council, Taiwan, R. O. C., under the Grant NSC 87-2213-E259-007

A. Architecture

In this paper only the basic architecture is discussed for clarity. The basic architecture is defined as the Fixed Transmission Tunable Reception Passive Star Architecture (FTTR-S) which is a single-hop broadcast-and-select photonic network built over the passive star coupler. A *station* is interconnected either with only one or a cluster of users, where each *user* is a set of corresponding electronic components. If users are connected to a station, they may be interconnected together with any network topology but use the same wavelength for communication at the same time slot with a collision-free protocol.

Meanwhile, let λ_0 represent the *control channel* which is used for control and *ACK* packets and each $\lambda_i, 1 \le i \le g$, represents a *data channel* which is used only for data transmission where λ_0, λ_1, λ_2, ..., λ_g are the available wavelengths in the system. The passive star coupler connects n stations in the system. Let $g \le n$ to accommodate the technology constraints in current devices. Each station is numbered as 1, 2, ..., n. Each λ_i is assigned to a group of stations as a transmitting home channel. Therefore, a group of stations is identified as $S_i = \{k: \lambda(k) = \lambda_i\}$ while λ_i is shared among the stations. For the transmission function, each station in S_i is provided with two fix-tuned transmitters to avoid frequent tuning. One is fixed at λ_0 for controlling accelerative transmission and the other is fixed at its home channel $\lambda(i)$ for data transmission. For the reception function, each station in S_i is equipped with one fix-tuned and one tunable receivers to avoid frequent tuning. The fix-tuned receiver is also fixed at λ_0 to receive the control and *ACK* messages while the tunable receiver can be switched between all data channels from λ_1 to λ_g for the corresponding data reception. Fig. 1 depicts an example architecture of the FTTR-S which contains many users per station. The architecture at each station is shown in Fig. 2. The F-TX represents the fixed transmitter, and the F-RX and the T-RX represent the fixed and the tunable receivers, respectively.

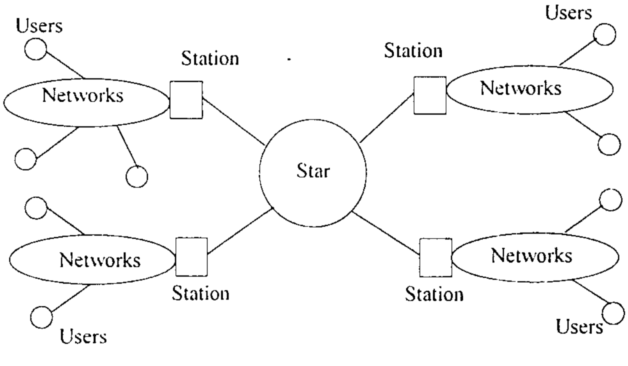


Fig. 1: Example FTTR-S with many users per station.

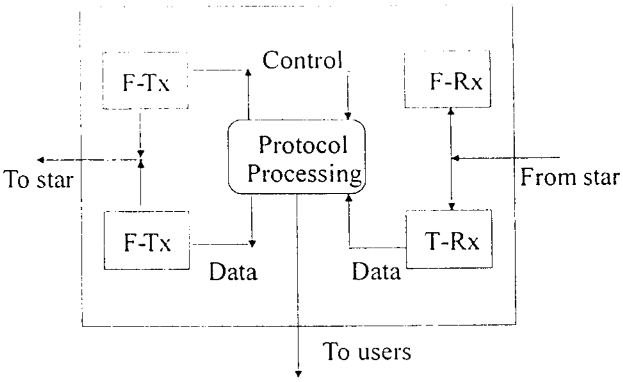


Fig. 2: The architecture at each station.

B. Terminology

In order to make the concept of the AP-WDMA clearer and to make the paper more readable, some definitions and assumptions are described as follows.

- *data packet*: The packet used on λ_i , $1 \le i \le g$, to transmit data as shown in Fig. 3. Only one packet per data slot through the corresponding λ_i is transmitted.
- *control packet*: The packet used on λ_0 to transmit the control message as shown in Fig. 3. This message is used to request a possible early transmission for acceleration or normal pre-allocation transmission.
- *ACK packet*: The packet used on λ_0 but is to transmit the *ACK* message as shown in Fig. 3. It is used to acknowledge the request by the control packet.
- *data slot*, S_d: The time unit on λ_i, is defined as the processing time of transmitting a maximum-size data packet from the source station to the destination station as shown in Fig. 3. It includes the operation time for the destination j to switch its tunable receiver to the corresponding λ_i and propagation delay. Besides, each data packet is assumed to cost the same data slot.
- *control* (*ACK*) *slot*, S_c: The time unit on λ_0, is defined as the processing time of transmitting the control (*ACK*) packet from the source (destination) station to the destination (source) station, including propagation delay as shown in Fig. 3.
- *data* (*control*, *ACK*) *cycle*: From the beginning, every n data (control, *ACK*) slots are organized as a *data* (*control*, *ACK*) *cycle*.
- *transmission pair*, $T(i_x,j)$: At a data slot, the data transmission from $x \in S_i$ to j through λ_i is performed, where $1 \le i \le g$ and $1 \le j \le n$. $T(i_x,j)$ is always initialized by the source station x and determined by j.
- *pre-scheduled pair*, $P(i_x,j)$: At a data slot, the $T(i,j)$ is pre-scheduled. Each node in S_i has opportunity to transmit its data. If $\exists i \in Si$ has data for j, j arbitrates them, selects one (x) from them, acknowledges its selection to x, then x transmits; else each $i \in S_i$ remains idle on λ_i.
- *early pair*, $E(i_x,j)$: On λ_i, the $T(i_x,j)$ occurs at a data slot earlier than the pre-scheduled slot of $P(i_x,j)$ in the pre-scheduled information.

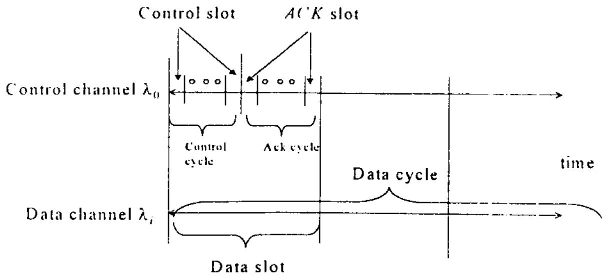


Fig. 3: Control slot (cycle), *ack* slot (cycle), and data slot (cycle), in the definition.

III. MAC PROTOCOL

In this section, a new media access protocol for the AP-WDMA is proposed. Fig. 4 illustrates the protocol algorithm which will be detailed in the following three subsections. Subsection A discusses the operations on each station *i*, including the time-interleaved pre-scheduling. Subsection B illustrates the control packet transmission. Then the *ACK* packet transmission and the $E(\lambda(i)_i,y)$ confirmation is depicted in Subsection C. An example is provided in Subsection D. Finally, Subsection E discusses the conditions to utilize the AP-WDMA appropriately.

```
procedure PROTOCOL;
 begin
   Initial_condition( );
   for every data slot do
   begin
    {simultaneously operates on each station $i$,
    whose corresponding home channel is $\lambda(i)$}
   if $E(\lambda(i)_i,y)$ exists then perform $E(\lambda(i)_i,y)$
   else if $P(\lambda(i)_i,j)$ exists then perform $P(\lambda(i)_i,j)$
   else remain idle;
   {each source $i$ always has a transmitter fixed
    on $\lambda_0$}
   Control_cycle( );
   {each destination $j$ always has a receiver
    fixed on $\lambda_0$}
   if no control packet in Control_cycle( )
   then terminate PROTOCOL
   {no more $P(i,j)$ will perform in the next $n-1$ data slots}
   else ACK_cycle( );
   {confirm possible $E(\lambda(i)i,y)$, normal arbitrated $P(\lambda(i)i,j)$
   or idle for each station i at the next data slot}
   end
end
```

Fig. 4: The media access protocol.

A. Operations on Each Station

In the beginning, the protocol performs the procedure Initial_condition( ) for each station *i* to restart the channel utilization from the beginning of the pre-scheduled table. The configuration of the network, e.g. group information, is set at this time. Then for every data slot, simultaneously on each station *i*, one of the following cases will occur. 1) If $E(\lambda(i)_i,y)$ exists, then perform $E(\lambda(i)_i,y)$ because no other pre-scheduled $P(\lambda(i)_i,j)$, $1 \le i \le g$, $1 \le j \le n$, can perform earlier than $P(\lambda(i)_i,y)$ by the $E(\lambda(i)_i,y)$ construction rules in Subsections B and C. 2) If $E(\lambda(i)_i,y)$ does not exist, then perform $P(\lambda(i)_i,j)$ when station *i* is arbitrated to have the data for station *j*, or will remain idle otherwise.

In Table 1, this pre-scheduled information is established based on the time-channel-station relationship. It illustrates that, for every data cycle, each λ_i is used as the receiving wavelength one by one by each station. And the usage of the table is that, for every data cycle, only the destination *j* can receive the data from the source S_i on λ_i at $T = h$ with $j = (i+h-1) \bmod n$. In the equation, $j = 0$ means $j = n$.

Table 1: The pre-scheduled information based on the time-channel-station relationship.

Channel/T	1	2	...	n-1	N
λ_1	1	2	...	n-1	N
λ_2	2	3	...	n	1
λ_3	3	4	...	1	2
...	...	...	...	...	...
λ_I	I	i+1	...	$(n-2+i)_n$	$(n-1+i)_n$
...	...	...	...	...	...
λ_g	$(g)_n$	$(g+1)_n$	...	$(n-1+g)_n$	$(n+g)_n$
	←the id# of the reception station →				

$(X)_n = X \bmod n$

For the AP-WDMA, each data packet doesn't have to indicate its source address bits and the destination address because the coordination is done in the control channel. The address is indicated by the authorized station, which can transmit and receive on this channel at each data slot in every data cycle. In addition, a station does not have to know all the information in the entire table to correctly execute the protocol. Instead, only the following two points have to be known by each station. 1) For data transmission, each source node in S_i must know the corresponding destination id# at each data slot (on λ_i actually). S_i will store the formula in the memory space which is used to derive id# identical to the information appeared in the corresponding row of λ_i in Table 1. Therefore, in every data cycle, the id# of the destination *j* at $T = h$ is derived from the formula that should be the same as the element in the *h*-th member of the corresponding row. 2) For data reception, each destination *j* must know the corresponding home channel id# of the source at each data slot. The destination *j* has to know these information in the Ar(*j*) array which is shown in Table 2 based on the time-Ar(*j*)-group relationship. Therefore, in every data cycle, the home channel id# of source S_i at $T = h$ that is appeared in the *h*-th member of Ar(*j*) is derived with the formula of $i = (j-$

$h+1$) mod n. If $i = 0, g+1, g+2, \ldots, n$, the corresponding element is kept empty and the station knows that no pre-scheduling is in this data slot.

Table 2: The reception information array Ar(j) in each station.

Ar(j)/T	1	2	3		n-1	n
Ar(1)	1	Empty	...		3	2
Ar(2)	2	1	Empty		4	3
...		...	...		...	...
Ar(i)	i	i-1	i-2		$(i+2)_n$	$(i+1)_n$
...		...	...		...	...
Ar(n)		...	...		2	1

the id# of the home channel for the source station

$(X)_n = X \bmod n$

B. Control Packet Transmission

While data are being transmitted, simultaneously on λ_0 at every data slot, the procedure Control_cycle( ) is performed by each node for the control packet transmission. It is based on the following two policies. 1) The time-interleaved pre-scheduling is used such that, in every control cycle, the i-th control slot is only for station i to transmit the control packet. 2) At the i-th control slot, whether station i transmits the control packet or not is decided by the check-and-send policy. Here, the first rule is to prevent collisions. This is because there is only one λ_0 for all stations to transmit the control packets. Therefore, each data slot is defined to contain one control and one *ACK* cycles where every control slot is dedicated to only one corresponding station with the time-interleaved scheduling. Meanwhile, the check-and-send policy decides the behavior of control packet transmission at the i-th control slot with one of the following three cases: 1) If station $i \in S_i$ has no data for any other station after this data slot, i.e., no other $P(i,j)$ for $1 \le j \le n$ will be performed during the next n-1data slots, then station i remains idle. 2) If the next possible $T(i,j)$ for station i is $P(i,j)$ at the next data slot, then station i transmits the control packet containing the id# of station j as the address bits. 3) Otherwise, an early transmission for acceleration may be possible at the next data slot. Station i will transmit the control packet whose address bits indicate the destination id# of the next possible $T(i,j)$. But, each control packet is without the source address because the i-th control slot already indicates station i as the source station. In addition, no other control message except the destination address is needed in each control packet.

C. ACK Packet Transmission

After the Control_cycle( ), the protocol performs the following steps to construct the $T(\lambda(i)_i,j)$ which is either $E(\lambda(i)_i,j)$ or $P(\lambda(i)_i,j)$. 1) Each destination j always has a receiver fixed at λ_0. If station j finds no control packet during this control cycle (actually all stations should have the same information), it illustrates that there will be no $T(i,j)$ in the next n-1 data slots. No *ACK* packet is transmitted by station j (actually no *ACK* packet is transmitted by any station). The protocol can be terminated at the next data slot and will be restarted under another system request. 2) Else, the procedure ACK_cycle( ) is performed by each station j. In the first part of ACK_cycle( ), the source id# of the possible $T(\lambda(i)_i,j)$ is decided by the check-and-discard policy and n corresponding memory spaces. A random number generator is needed for each receiver in each node to arbitrate among more than one certified node. All the received control packets are checked one by one as they are transmitted in Control_cycle( ). The processing steps are illustrated as follows. a) For each control packet, the source id# (represented as f) of the first control packet destined to each station is stored in the corresponding memory space. Therefore, n memory spaces are prepared. b) The source id# (represented as x) of each following control packet destined to the same station is compared with f. If x and f have the same home channel, randomly select x or f. Else if $P(\lambda(x)_x,j)$ is earlier than $P(\lambda(f)_f,j)$ in the next n-1 data slots (Ar(j) can be referenced), then x replaces f in the memory space. Otherwise, the new incoming control packet is discarded. c) Therefore, after one control cycle, each memory space has a source id# or nothing. Each one forms a transmission pair and has a corresponding pre-scheduled time slot. Sequentially compare the pre-scheduled time slots which have the same home channel id# as that of the source id# in the memory space j and decide the final unique transmission pair. $T(\lambda(i)_i,j)$ can be either $P(\lambda(i)_i,j)$ or $E(\lambda(i)_i,j)$.

After the above decision has been made, the second part of ACK_cycle( ) is for *ACK* packet transmission decided by the following policies. a) The time-interleaved pre-scheduling is used such that the j-th *ACK* slot of each *ACK* cycle is pre-scheduled for station j to transmit the *ACK* packet. This is also to prevent collisions with a reason similar to that in Control_cycle( ). b) At the j-th *ACK* slot, whether station j transmits the *ACK* packet or not is decided by the check-and-send policy with the following steps. i) If station j has no *ACK* for any other station during this *ACK* cycle, station j remains idle. ii) Otherwise, station j transmits the unique *ACK* packet with i as the destination address bits to confirm $T(\lambda(i)_i,j)$. Hence each source station which has transmitted the control packet destined to station j in Control_cycle( ) need only look into the j-th *ACK* slot in the *ACK* cycle. And the selected source i will need to perform $T(\lambda(i)_i,j)$ at the next data slot. In this manner, the processing complexity is reduced. No more *ACK* message including the source address is needed in each *ACK* packet because the j-th *ACK* slot already indicates station j as the source station. In addition, while transmitting the *ACK* packet, station j can already tune its tunable receiver to the corresponding channel $\lambda(i)$ to save the tuning latency. The data packet of $T(\lambda(i)_i,j)$ also needs no address id#. Although in fact only *g ack* slots are used since only *g* available channels can make concurrent transmissions, *n ack* slots are used for low operation complexity.

D. Example

In order to illustrate the details of the AP-WDMA, an example is provided in this subsection. This example is based on an FTTR-S that contains 7 stations, 1 control and 3 data channels. As illustrated in Table 3, the transmission behaviors in one data cycle are considered. Table 3 has four parts: T1, T2, T3 and T4. Each part has its corresponding meaning. T1 shows the channel sharing information: {1,2,3},{4,5} and {6,7} are three groups in the system and their corresponding home channels are λ_1, λ_2 and λ_3, respectively. T2 is the data cycle whose length is determined by the number of nodes in the system. T3 is the seven control and *ACK* cycles, whose duty is to decide the next data slot's behavior in T4. T4 is the original pre-allocation table by the AP-WDMA protocol. How acceleration is achieved is explained next using time slot 2 in data channel λ_1 and time slot 1 in control channel. Although this slot is originally pre-allocated to P(1_x,2), where $x \in \{1,2,3\}$, the real transmission pair is decided by the previous slot in the control channel. There are four possible cases: Normal Pre-allocated (NP), Early Fail (EF), Early Conflict (EC) and Early Successful (ES).

Table 3: The components in the AP-WDMA.

	Time	1	2	3	4	5	6	7	*T2*
	Control	▨							*T3*
1,2,3	λ1	1	2	3	4	5	6	7	
4,5	λ2	2	3	4	5	6	7	1	*T4*
6,7	λ3	3	4	5	6	7	1	2	
T1									

In order to present the example transmission behavior clearly, Table 4 is used to capture all possible situations in the AP-WDMA. The first case in the NP is used to illustrate the arbitration between station 1 and station 3. A random number generator randomly selects station 1. This transmission is deemed as P(1,2). The second case in the NP is used to show how station 2 and station 3 achieve the consistent behavior. Although station 1 wants to make early transmission and station 3 wants to make pre-allocated transmission, station 2 and station 3 use Ar(2) and Ar(3) calculated by the formula derived in the previous Section to know (1,2) is earlier than (1,3) and come to the same conclusion. In EF case, station 1 wants to make early transmission to station 3 but station 4 wants to make the pre-allocated transmission to station 3. Station 3 use Ar(3) to choose station 4 as its destination, therefore after *ACK* cycle station 1 knows that it gets a unsuccessful transmission. In EC case, two early transmissions are arbitrated by station 5. Station 5 uses Ar(5) to choose station 4 as its destination, thus station 1 gets a conflict transmission. In fact, whether EF or EC occurs, the source station doesn't distinguish them but only knows that it doesn't make a successful transmission. Finally in ES, the first case is used to explain the maximum number of slots advanced while the second case is used to show the minimum number of slots advanced.

Table 4: Example transmission behavior of the AP-WDMA.

Cases	Control packets in slot 1	Data packet in slot 2
NP	(1_1,2), (1_3,2)	(1_1,2)
	(1_1,3), (1_3,2)	(1_3,2)
EF	(1_1,3),(2_4,3)	(2_4,3)
EC	(1_1,5),(2_4,5)	(2_4,5)
ES	(1_2,1)	(1_2,1)
	(1_1,3)	(1_1,3)

E. Why AP-WDMA

Theoretically, S_d is dominated by the following factors: d_p, d_o, d_i, d_{offset}, and d_t. Here, d_p is the transmitter processing time, d_o the propagation time in the optical fiber, d_i the corresponding idle time for a data packet smaller than the maximum size, d_{offset} the individual offset time due to the specific transmission distance between the source and destination stations, and d_t the time needed for the corresponding receiver tuned to the data channel. In our protocol, $d_p+d_o+d_i+d_{offset}$ and d_t can be started at the same time. Meanwhile, S_C is only dominated by three factors: c_o, c_{offset}, and c_p, because all packet sizes are the same (without c_i) and the corresponding receiver always senses on λ_0 (without c_t). Here, c_o is the propagation time of the control (or *ACK*) packet in the optical fiber, c_{offset} the individual offset time due to the specific transmission distance between the source and destination stations, and c_p the processing delay of the control packet. If the data rate per channel is assumed to be about 1Gbps, then c_p ≅(control packet size)/1Gbps=8*bits/1Gbps=0.008 *us*, for $n \cong 100$. If the distance between any two stations in the LAN is nearly 1km, then $c_o = 1*10^3/2*10^8 = 5$ us. With $c_{offset} << 5$ *us*, then $S_C \cong 5$ *us*. However, technically, d_t is dominated by the tuning speed of the tunable wavelength filter which can only be tuned at a limited speed and in a limited wavelength range [3]. Generally speaking, the tuning speed will be slower if the tuning range is wider. For example, the number of supported channels and the tuning speed are estimated respectively to be 128 channels and a few *ms* for a Mach-Zehnder filter, 100 channels and 10*us* for an Acousto-optic TE/TM filter, 10 channels and a few *ns* for an Electro-optic TE/TM filter, and 2-3 channels and 1*ns* for the DFB filter. Therefore, $d_t = 10$ *us* is an optimistic assumption for a moderate system. Since Max($d_p+d_o+d_i+d_{offset}, d_t$) is equal to d_t for packets with sizes near 1000bits, S_d is about 10*us*. In this case, 1 control cycle + 1 *ACK* cycle = $2*S_C \cong 10us \cong S_d$.

From the above observation, there is enough time for λ_0 to be used to decide the behavior on each λ_i at the next S_d when the behavior on each λ_i at an S_d is progressing. Therefore, with the behavior of each P(*i,j*) which can be known one data slot before the pre-scheduling, the AP-WDMA is a much better protocol to improve the performance of the P-WDMA and reduce the complexity of R-WDMA. In addition, as the WDM technologies and components continue to be improved, the required switching time of the receiver may be further reduced. However, the AP-WDMA can still perform very well if the scope of the star coupler is selected to satisfy the condition, $2*S_c \leq S_d$. If this can not be achieved, partitioning the star coupler is a possible solution.

IV. EVALUATIONS

This section evaluates the performance of the AP-WDMA. Subsection A consists of a table of examples. Subsection B uses the probability analysis to evaluate the impact of $E(\lambda(i)_{i,j})$ on delay reduction.

A. Examples

In order to illustrate the details of the AP-WDMA, one example is provided in this subsection. The example is based on an FTTR-S which contains 8 stations, 4 data channels and 1 control channel. As illustrated in Tables 5 and 6, the transmission behaviors in 2 data cycles are considered. Table 5 depicts the original transmission behavior by the P-WDMA protocol in Table 1, where "/" means "idle" and *j* in the λ_i row means the P(*i,j*) occurs as pre-scheduled. The actual transmission behavior of AP-WDMA is exploited in Table 6, where "..." means more E(*i,j*) may be achieved if more P(*i,j*) from the 3rd data cycle are considered. Besides, Table 6 indicates that some E(*i,j*) are achieved and some delays are alleviated. For example, in the 1st data cycle, T(1,4) and T(1,5) occur 2 data slot earlier, and T(1,6), T(1,7), and T(1,8) occurs 1 data slots earlier. Furthermore, T(2,1) of the 1st data cycle occurs 6 data slot earlier and that of the 2nd data cycle even occurs 13 data slots earlier. T(3,6), T(3,7), T(3,8), T(3,1) and T(3,2) in the 1st data cycle show that the AP-WDMA generates no E(*i,j*) and only maintains the same pre-scheduled transmission behavior.

Table 5: The example transmission behavior of the P-WDMA.

	the 1st data cycle								the 2nd data cycle							
C/T	1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8
λ_1	1	/	/	4	5	6	7	8	/	/	/	/	/	/	/	/
λ_2	2	/	/	/	/	/	/	1	/	/	/	/	/	/	/	1
λ_3	3	/	5	6	7	8	1	2	3	4	5	6	7	8	1	2
λ_4	4	/	6	7	8	1	2	3	4	5	6	7	8	1	2	3

Table 6: The example transmission behavior of the AP-WDMA.

	the 1st data cycle								the 2nd data cycle							
C/T	1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8
λ_1	1	4	5	/	6	7	8									
λ_2	2	1	1													
λ_3	3	5	/	6	7	8	1	2	3	4	5	6	7	8	1	2
λ_4	4	6	7	8	1	2	3	4	5	6	7	8	1	2	3	4

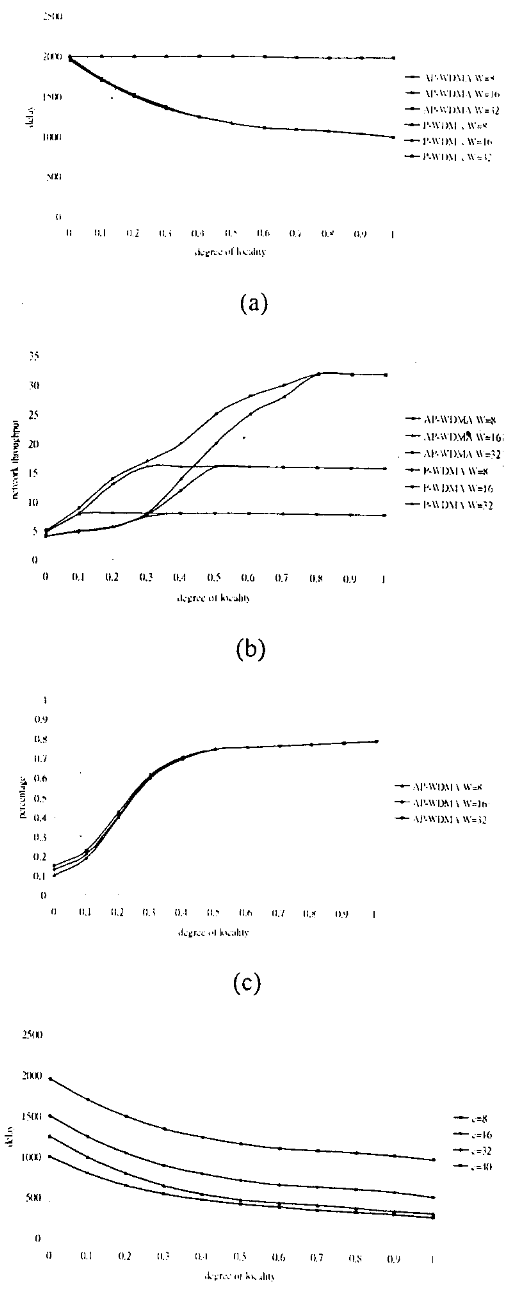
B. Simulation Analysis

To evaluate the effectiveness of the proposed protocol, the AP-WDMA is implemented with C programming language on a Sun SPARC 20 workstation. The P-WDMA was also implemented for the purpose of comparison. The message arrival rate of each channel *i* is a Poisson distribution with a mean σ_i, and the message length is an exponential distribution with a mean of *L* packets. The channel load for channel *i* can be defined as $CL_i = \sigma_i \times L$. The network load can be defined as $W = \sum_{i=1}^{g} CL_i$. Since the phenomena of traffic localities usually happen in present networks, including the metropolitan area networks, a more general traffic source-destination distribution is considered here. Assume *Pi,j* is the probability that stations with home channel λ_i is transmitting to station *j* and it is uniformly distributed among those source stations. The traffic source-destination distribution is derived from the following equality [27]:

$$Pij = \left\{ \begin{array}{ll} 0, & j \leq i \\ \dfrac{(1-p)^{j-i-1}\, p}{1-(1-p)^{n-i-1}}, & j > i \end{array} \right\}.$$

This equation represents a normalized geometric distribution where p ($0 \leq p \leq 1$) determines the level of traffic locality. Although this equation is originally developed for *DQDB*, its locality characteristic is enough to explain the difference between AP-WDMA and P-WDMA.

For each simulation run (800,000 slot times), the data is collected from the 400,000th to the 800,000th slot times. The average message length is 50 packets (L=50) and all channels have the same channel load CL ($CL_i = W/g$, i=1, 2, ..., g). Three kinds of network parameters are considered: The network load (Ω), the degree of locality level (p) and the number of available channels (c). The locality parameter p is considered from 0.0 increases to 1.0 by a step of 0.1. The network average delays obtained by the AP-WDMA and the P-WDMA are measured and compared. The other assumptions are listed as follows: 1) n=40 stations in the network. 2) c={8, 16, 32, 40} data channels on the network. 3) W={8, 16, 32}. 4) p={0.0 (uniform distributed), 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0 (fully locality)}.



(a)

(b)

(c)

(d)

Fig. 5: (a), (b): comparison of network throughput and average delay obtained by the AP-WDMA and P-WDMA under different network loads. (c): early percentage obtained by AP-WDMA under different network loads. (d): delay obtained by AP-WDMA under different number of channels.

Fig. 5(a) shows the average delay obtained by the AP-WDMA and P-WDMA under different locality levels when n=40, c=8, and Ω={8, 16, 32}. It illustrates the average delay (in terms of slots) of each message which is measured as the average time interval since a message arrives at the front of the queue and the transmission of its last packets. We can find that the average delay of the AP-WDMA is much better than that of the P-WDMA. As the source-destination distribution is more uniform (i.e. p=0), a more number of early collisions will happen and a less reduction of delay will achieve. In the contrast, for a higher degree of locality, more early transmission will be occurred and the delay reduction is more obvious. Fig. 5(b) shows that the AP-WDMA outperforms the P-WDMA in the lower degree of locality and has the same throughput with the P-WDMA in the higher degree of locality. This is because the AP-WDMA efficiently utilizes the data slots to obtain high network throughput. When the degree of locality is higher, the number of available transmission pair is lower such that the throughput enhancement by AP-WDMA is limited. Fig. 5(c) shows that the percentages of early transmission in AP-WDMA under different network loads. We can find that, as the network becomes more loaded, the percentage of early transmission becomes higher. No matter what the network load is, the more degree of locality can increase more early transmission. Fig. 5(d) illustrates that more data channels can make more delay reduction. But the delay reduction is not proportion to the number of data channels. In other words, we can say that it is a trade-off between the number of channels and delay requirement when installing an inexpensive and high performance lightwave network.

V. CONCLUSIONS

We have proposed a new multiaccess photonic model, AP-WDMA/early transmission. AP-WDMA is based on the P-WDMA without its drawbacks and improves the system performance with the least cost. The technique is to allow the source station to possibly transmit the data earlier than the pre-scheduled time slot in the P-WDMA through the known-ahead transmission information and the additional control channel. Through the examples and simulations, it was confirmed that the AP-WDMA did reduce delays and improve the system performance significantly in unbalanced traffic load. Even if earlier transmission can not be achieved, the AP-WDMA will still work just like a P-WDMA and retain its advantages. Furthermore, the channel sharing and time interleaving mechanism in the AP-WDMA can easily support more nodes than available channels under the condition provided in Section II-C. In addition, the protocol works well under various traffic loads because it can automatically reduce delay by the accelerative transmission. Therefore, the proposed AP-WDMA is a very powerful and practical media access control protocol.

REFERENCES

[1] C. A. Brackett, A. S. Acampora, J. Sweitzer, et al., "A Scalable Multiwavelength Multihop Optical Network: A Proposal for Research on All-Optical Networks," *IEEE J. Lightwave Tech.*, vol. 11, no. 5/6, pp. 736-753, May/June, 1993.

[2] P. E. Green, L. A. Coldren, K. M. Johnson, et al., "All-Optical Packet-Switched Metropolitan-Area Network Proposal," *IEEE J. Lightwave Tech.*, vol. 11, no. 5/6, pp. 754-763, May/June, 1993.

[3] C. A. Brackett, "Dense Wavelength Division Multiplexing Networks: Principles and

Applications," *IEEE J. Select. Areas Commu.*, vol. 8, no. 6, pp. 948-964, August 1990.

[4] A. Ganz and Y. Gao, "Time-Wavelength Assignment Algorithms for High Performance WDM Star Based Systems," *IEEE Trans. on Commu.*, vol. 42, no. 2/3/4, pp. 1827-1836, Feb./Mar./Apr. 1994.

[5] S. T. Tan and D. Du, "Embedded Unidirectional Incomplete Hypercubes for Optical Networks," *IEEE Trans. on Commu.*, vol. 41, no. 9, pp. 1284-1289, Sept. 1993.

[6] K. A. Williams, T. Q. Dam, and D. Du, "A Media-Access Protocol for Time and Wavelength Division Multiplexed Passive Star Networks," *IEEE J. Select. Areas Commu.*, vol. 11, no. 4, pp. 560-567, May 1993.

[7] J. Sasaki and N. Takato, "An Optical Multiaccessor Designed for a Fiber-Optic Star-Ring LAN," *IEEE J. Lightwave Tech.*, vol. 10, no. 2, pp. 250-254, Feb. 1992.

[8] M. G. Hiuchyj and M. J. Karol, "ShuffleNet: An Application of Generalized Perfect Shuffles to Multihop Lightwave Networks," *IEEE J. Lightwave Tech.*, vol. 9, no. 10, pp. 1386-1397, Oct. 1991.

[9] G. N. Sudhakar, N. D. Georganas, and M. Kavehrad, "Slotted Aloha and Reservation Aloha Protocols for Very High-Speed Optical Fiber Local Area Networks Using Passive Star Topology," *IEEE J. Lightwave Tech.*, vol. 9, no. 10, pp. 1411-1422, Oct. 1991.

[10] M. S. Goodman, H. Kobrinski, M. P. Vecchi, et al., "The LAMBDANET Multiwavelength Network: Architecture, Applications, and Demonstrations," *IEEE J. Select. Areas Commu.*, vol. 8, no. 6, pp. 995-1004, August 1990.

[11] M. S. Chen, N. R. Dono, and R. Ramaswami, "A Media-Access Protocol for Packet-Switched Wavelength Division Multiaccess Metropolitan Area Networks," *IEEE J. Select. Areas Commu.*, vol. 8, no. 6, pp. 1048-1057, August 1990.

[12] P. W. Dowd, K. Bogineni, K. A. Aly, J. A. Perreault, et al., "Hierarchical scalable photonic architectures for high-performance processor interconnection," *IEEE Trans. on Computers*, vol. 42, no. 9, pp. 1105-1120, Sept. 1993.

[13] K. Bogineni, K. M. Sivalingam, and P. W. Dowd, "Low Complexity Multiple Access Protocols for Wavelength-Division Multiplexed Photonic Networks," *IEEE J. Select. Areas Commu.*, vol. 11, no. 4., pp. 590-604, May 1993.

[14] K. Bogineni and P. W. Dowd, "A collisionless multiple access protocol for a wavelength division multiplexed star-coupled configuration: architecture and performance analysis," *IEEE J. Lightwave Tech.*, vol. 10, no. 11, pp. 1688-1699, Nov. 1992.

[15] P. W. Dowd, "Wavelength division multiple access channel hypercube processor interconnection," *IEEE Trans. on Computers*, vol. 41, no. 10, pp. 1223-1241, Oct. 1992.

[16] P. W. Dowd, "Random access protocols for high speed interprocessor communication based on a passive star topology," *IEEE J. Lightwave Tech.*, vol. 9, no. 6, pp. 799-808, June 1991.

[17] Chen-Ken Ko and Sy-Yen Kuo, "Multiaccess Processor Interconnection Using Subcarrier and Wavelength Division Multiplexing," *IEEE J. Lightwave Tech.*, v[illegible] 1, pp. 228-241, Feb. 1997.

[18] C. J. Hou, B. Wang, and C. C. Han, "Design and Analysis of a WDMA Protocol for Passive Star-Coupled Lightwave Networks," *Infocom* 96, pp. 1225-1233, Feb. 1996.

[19] J. P. Weber, B. Stoltz, H. Sano, et. al., "An Integratable Polarization-Independent Tunable Filter for WDM Systems: The Multigrating Filter," *IEEE J. Lightwave Tech.*, vol. 14, no. 12, pp. 2719-2756, Dec. 1996.

[20] E. L. Wooten, R. L. Stone, E. W. Miles, and E. M. Bradley, "Rapidly Tunable Narrowband Wavelength Filter Using LiNbO3 Unbalanced Mach-Zehnder Interferometer," *IEEE J. Lightwave Tech.*, vol. 14, no. 11, pp. 2530-2536, Nov. 1996.

[21] G. N. Rouskas and M. H. Ammar, "Analysis and Optimization of Transmission Schedules for Single-Hop WDM Networks," *IEEE/ACM Tran. on Networking*, vol. 3, no. 2, pp. 211-221, Apr. 1995.

[22] F. Jia, B. Mukherjee, and J. Iness, "Scheduling Variable-Length Messages in a Single-Hop Multichannel Local Lightwave Network," *IEEE/ACM Tran. on Networking*, vol. 3, no. 4, pp. 477-488, Aug. 1995.

[23] P. A. Humblet, R. Ramaswami, and K. N. Sivarajan, "An Efficient Communication Protocol for High-Speed Packet-Switched Multichannel Networks," *IEEE J. Select. Areas Commun.*, vol. 11, no. 4, pp. 568-578, May 1993.

[24] D. Guo, Y. Yemini and Z. Zhang, "Scalable High-Speed Protocols for WDM Optical Star Networks," *IEEE Globecom* '94, pp. 1544-1551, Dec. 1994.

[25] R. Chipalkatti et al, "Protocols for Optical Star-Coupler Network Using WDM: Performance and Complexity Study," *IEEE JSAC* vol. 11, no. 4, pp. 579-589, May 1993.

[26] B. Kannan, S. Fotedar and M. Gerla, "A Protocol for WDM Star Coupler Networks," *IEEE Infocom* '94, pp. 1536-1543, June 1994.

[27] M. A. Rodrigues, "Eraure nodes: Performance improvements for the IEEE 802.6 MAN," in *Proc. IEEE INFOCOM'90*, San Francisco, CA, 1990, pp. 636-643.