

Choices, Features and Issues in Optical Burst Switching

C. Qiao[†] and M. Yoo[‡]

Departments of CSE[†] and EE^{†‡}

University at Buffalo (SUNY), Buffalo, NY 14260

{qiao}@computer.org

July 27, 1999

Abstract

In this article, we first explore design choices in burst-switching and describe a new variation that is especially suitable for optical WDM networks. We then identify main features of optical burst switching (OBS), discuss major differences and similarities between OBS and optical circuit- and packet-switching, and address important QoS related performance issues in OBS.

1 Introduction

Circuit- and packet-switching have been used for many years for voice and data communications. The concept of optical circuit-switching (e.g. wavelength-routing in WDM networks) and optical packet-switching, though still evolving, has also been around for quite a while. Burst-switching[1, 2], on the other hand, is less popular. In particular, *optical burst switching* (or OBS) was introduced only recently for optical (WDM) networks [3, 4], and is thus not as well-understood as optical circuit- and packet-switching. Accordingly, questions such as what are the differences (and similarities) between burst switching and circuit-/packet-switching in general, and specifically, between OBS and optical circuit-/packet-switching have been raised.

A common definition of a burst is a (digitized) talk spurt or a data message. In circuit-switching, a call (or session), which consists of multiple bursts, is treated a basic switching (or transferring) entity in terms of consuming bandwidth as well as setting up switches¹, while in packet-switching, a burst is usually transferred in several packets, each of which being a basic switching entity. As such, a burst may be considered as having an intermediate “granularity” when compared to a call (or session) and a packet.

¹The term “switches” will be used broadly to refer to cross-connects, routers, add-drop-multiplexers etc. as well.

However, there are more to the differences between burst-switching and circuit-/packet-switching than simply the granularity of the switching entity. More specifically, we may express the differences among various switching techniques in terms of whether data will use the *switch cut-through* or *store-and-forward* method. Furthermore, switching techniques may also differ based on how and when the network resources, e.g. bandwidth, is reserved and released, and whether control signals and data are separated by using different channels, and/or by sending one (e.g. data) after another (e.g. a control signal) with a non-zero *offset time*, i.e., a gap or an idle period.

Another issue that is becoming more and more prominent is *which layer* in a protocol stack forwards data at intermediate nodes. For example, IP packets (or datagrams) are traditionally forwarded at layer 3 (or the network layer). With Multiprotocol Label Switching (MPLS) [5], which is similar to virtual-circuit (VC) based packet-switching, they, just as ATM cells, can be forwarded at layer 2 (the data link layer). This issue is of course a key in distinguishing electronic switching from optical (or photonic²) switching, where data is kept in the optical (WDM) layer at intermediate nodes.

Note that, which layer runs directly on top of the optical WDM layer determines the traffic patterns to be supported, and hence is a factor affecting the suitability of a given optical switching technique. For example, wavelength-routing is a natural choice for establishing SONET/SDH connections, while optical packet- and burst switching are more efficient for directly carrying bursty traffic generated by the IP layer. Other important factors have to do with the unique characteristics of optical networks including the lack of optical buffer, limited optical logic and difficulty in achieving synchronization, which explain why optical packet-switching, unlike its electronic counterpart, has not yet become a dominant choice.

This article will distinguish burst-switching from circuit- and packet-switching based primarily on whether *cut-through* or *store-and-forward* is used and how bandwidth is reserved (and released). It will also explore design choices in burst-switching and explain why a new technique, which separates data from control signals, can effectively alleviate the problems such as a large buffer space requirement and a high control complexity, which otherwise exist in burst-switching. Moreover, the article will show how the new burst-switching technique makes OBS a viable choice for building the next generation Optical Internet.

The article is organized as follows. In Sec. 2, we will describe the main characteristics of circuit-switching and discuss the pros and cons of wavelength-routing in WDM networks. In Sec. 3, we will describe *packet-switching* based on either datagrams or virtual circuits (VCs), and discuss main challenges and recent trends in optical packet-switching. In Sec. 4, we will describe and then compare three basic variations of *burst-switching*, namely, *tell-and-go* (TAG), *in-band-terminator* (IBT), and *reserve-a-fixed-duration* (RFD), and discuss why RFD, which has not been studied for electronic networks, is attractive especially when combined with the use of an appropriate offset time. Finally, we will describe a RFD-

²We will hereafter use the term “optical” even if control signals may be processed electronically.

based OBS protocol called *Just-Enough-Time* (JET), and address two important QoS performance issues, namely, burst-dropping and end-to-end latency, in Sec. 5, and summarize the main features of OBS in Sec. 6.

2 Circuit-Switching

Circuit-switching has three distinct phases, circuit set-up, data transmission and circuit tear-down. One of the main features of circuit-switching is its *two-way reservation* process in phase 1, where a source sends a request for setting up a circuit and then receives an acknowledgment back from the corresponding destination.

A circuit is set-up by reserving a fixed bandwidth channel (e.g. a frequency or a time slot) on each link along a path from the source to its corresponding destination. When distributed control is used in phase for “routing” (i.e. finding a path), the *offset time* between a set-up request and data transmissions, T , is *at least* as long as $2P + \Delta$, where P is the one-way propagation delay and Δ is the total processing delay encountered by the set-up request along the path.

Another feature of circuit-switching is that all the intermediate switches will be configured to “latch” the channels (e.g. frequencies or time slots) on the adjacent links in order to form a circuit, and remain so for the duration of the call (or session), L_c . This feature also implies that no buffering (of data) is needed at any intermediate node (except for interchanging time slots in a TDM system).

Note that in a variation of circuit-switching used by the time assignment speech interpolation (TASI) systems for undersea cables, the first phase does not reserve “dedicated” bandwidth. That is, it does not actually set up a circuit but instead, only establishes a route, much like in a VC-based packet-switching to be discussed later. When a burst arrives, a special signal is sent to acquire bandwidth (and set-up a circuit), and after the end of the burst (e.g. silence) is detected, the bandwidth (or circuit) is released. This technique is called *fast circuit-switching* because routing has been done (i.e. a path has been determined) in phase 1, and circuit set-up and tear-down take place at the burst level later. Because of the strict delay-constraint, a digitized talk spurt is sent after the special control signal *without* waiting for the acknowledgment that a circuit has been successfully set up. In other words, circuit set-up is now a *one-way* process. Nevertheless, for data communications, a data message may wait, so a two-way process is still used as in circuit-switching (see [6, 7]³).

2.1 Wavelength Routing

In WDM networks, circuit-switching takes the form of *wavelength routing*, where an all-optical *wavelength path*, consisting of a dedicated wavelength channel on every link, is established between two re-

³[6] studied three protocols. This specifically refers to one of them called *CC* in [6]. The other two will also be referenced later.

remote communicating nodes. In addition to providing high-speed, high-bandwidth pipes that are transparent to bit-rate and coding format, wavelength routing is especially suitable for supporting SONET/SDH communications because (1) SONET/SDH switches communicate with each other at a constant bit-rate that matches the bandwidth of a wavelength (e.g. 2.5Gbps or OC-48), (2) the connection duration, L_c , is long relative to the path set-up time, (3) the number of expensive SONET switches can be reduced with proper traffic grooming and wavelength assignment algorithms and (4) the optical switches (wavelength routers) based on opto-mechanical, acousto-optic or thermo-optic technologies are currently too slow for efficient packet-switching.

However, wavelength-routing will result in a low bandwidth utilization if the traffic to be supported is bursty. Since Internet traffic is *self-similar* (or bursty at all time scales), this means that providing wavelength paths between two remote IP routers may not be efficient. In addition, given a limited number of wavelengths, only a limited number of wavelength paths can be established at the same time, implying that some data may still need to go through O/E/O conversions. If wavelength paths are established (and torn-down) dynamically to alleviate the above problems, the set-up time of a wavelength path based on two-way reservation, which is at least tens of *ms* in a nation-wide backbone network, will be too long for a burst containing a few megabit of data (the size of a small file) because the burst length L_b will be at most a few *ms* given the high bandwidth (e.g. 2.5Gbps or OC-48) of each wavelength.

3 Packet-Switching

In packet-switching, the length of each packet, L_p , can be either fixed, say at S_{fix} , or variable having a minimum S_{min} and a maximum S_{max} . With a fixed packet length, a burst of size L_b will be broken into $\lceil L_b/S_{fix} \rceil$ packets of the same size (by using padding if L_b is not an integer multiple of S_{fix}). With a variable length, the message will be broken into $\lceil L_b/S_{max} \rceil$ packets, and padding is used only if a packet is shorter than S_{min} . For voice communications, fixed-length packets (e.g. digitized voice packets) are used, but for data (computer) communications, variable-length packets (e.g. IP datagrams) are used.

Packet-switching may be based on either *datagrams* or *virtual-circuits* (VCs). A main feature of both variations of packet-switching is *store-and-forward*. That is, a packet needs to be completely assembled (and received) by a source (and each intermediate node) before it can be forwarded. One of the implications of this feature is that a packet will experience a delay which is proportional to L_p at each node. In addition, a buffer of size at least equal to S_{max} is needed at each intermediate node. These implications are some of the reasons why small, fixed-length packets are used in ATM, and why *message-switching*, where an entire message is switched using one header, is not an attractive variation of packet-switching. A disadvantage of using small, fixed-length packets (or cells) is that the percentage of control overhead is higher (due to a relatively large header in each packet/cell) and bandwidth utilization is lower (due to both headers and possible paddings) than using variable-length packets or messages. For example, the

control overhead in ATM (called “cell-tax”) is as high as 10%⁴.

3.1 Datagram versus Virtual Circuit

In datagram-based packet-switching (used by IP), a packet header contains the destination address of the packet, based on which layer 3 forwarding (or routing) is done at every intermediate node. In VC-based packet switching (used by ATM), a VC is set-up first (implying that routing is already done) and hence, each packet’s header carries a label (e.g. a VC identifier), based on which layer 2 forwarding (or switching) is done at every intermediate node.

Note that, in ATM, certain network capacity (bandwidth and buffer space) can be “reserved” for a VC for the purpose of providing quality of service (QoS). The reservation process can be either two-way or one-way. When such a technique is applied to making reservation for a burst of packets (or ATM cells) as in *fast reservation protocol* (FRP) and its variations (see for example, [8, 9]), it becomes quite similar to fast circuit-switching. However, unlike in fast circuit-switching, a talk spurt is segmented into many packets/cells, and each packet/cell (of the burst) over the VC is still stored and forwarded *individually*. That is, the header of each packet/cell will be processed to determine how to configure a switch, a switch will stay in the configured state for a duration of L_p , where $L_p < L_b < L_c$.

3.2 IP Flow and Multiprotocol Label Switching

Normally, layer 2 forwarding is based on finding an exact match between the label carried by a packet and a label created during the VC set-up process and accordingly, it is easier (faster) than layer 3 forwarding⁵. In fact, the simplicity of layer 2 forwarding along with the use of cells is the reason why ATM is once considered as fast packet-switching, and ATM switches can be implemented with a lower cost (in dollars) than IP routers with the same performance (in throughput).

The simplicity of layer 2 forwarding is also a motivation behind Multiprotocol Label Switching (MPLS) (which can also facilitate traffic engineering). The basic idea of MPLS is to establish label switched paths (LSP), which are similar to VCs, so packets (e.g. IP packets or datagrams) may be forwarded at layer 2 instead of layer 3. The establishment of LSPs can be control-driven, i.e., performed by a network according to its topology and connectivity as in Tag-switching but it may also be data-driven as in IP-switching [5]. For example, the first few IP packets of a *flow* are forwarded at layer 3, where a flow could refer to all the IP packets from a source to a destination. (At a finer granularity level, it could refer to all the IP packets from a source host to a destination host, or just those belonging to the same TCP connection).

⁴Recently, the ATM Forum has been working on a specification called Fast (Framed ATM over SONET Transport), which will allow up to 64 Kbytes of data for every header of 4 bytes.

⁵In IP forwarding (or routing), one needs to find the longest substring in the routing table that matches the packet’s destination address.

As soon as the destination recognizes the flow (e.g. when the number of IP packets it received from the source exceeds a given threshold), it triggers the establishment of a LSP so the rest of the IP packets of the flow can be forwarded at layer 2.

3.3 Optical Packet Switching

Optical packet switching is suitable for supporting bursty traffic since it allows statistical sharing of the channel bandwidth among packets belonging to different source and destination pairs. To keep the percentage of the control overhead down, however, it needs fast switches based on Lithium Niobate technologies and semiconductor optical amplifiers (SOAs).

In optical packet switching, the payload (i.e. data) will remain in the optic form, while its header may be processed electronically or optically (though the optical logic is very primitive). There are many challenges in keeping the data in the optical domain [10]. One of the biggest challenges is that there is no optical equivalence of the random access memory (RAM), and accordingly, (1) an optical data signal can only be delayed for a limited amount of time via the use of fiber-optic delay lines (FDLs) before the header processing has to complete, and (2) the length of each packet, in terms of the product of its transmission time and the speed of light, cannot exceed that of the available FDL in order for the optical packet to be “stored”. A related issue is synchronization since each node needs to recognize the header and the end of a packet, and re-align a modified/replaced header with its payload. Consequently, VC-based optical packet switching with small, fixed-length packets (or cells) is more attractive than datagram-based optical packet switching.

In addition to using fixed-length packets almost exclusively (e.g. see [11]), optical packet switching traditionally uses the full-bandwidth of a fiber based on TDM (or soliton) technologies, but has also been adapted to using WDM technologies by transmitting packets at the bandwidth of a wavelength. In particular, in order to facilitate implementation, headers can be transmitted on a separate wavelength or a subcarrier channel (see e.g. [12, 13]). Such use of “out-of-band” control may be regarded as a step to loosen the coupling between control and data, which is tight in the traditional packet-switching. More specifically, using a separate control wavelength or subcarrier channel makes it possible for a node to process the header (and set the local switch) before the payload is fully “stored” (in FDLs). An additional step can also be taken so that the FDLs are used to simply *delay* the payload for a maximum processing time, instead of storing it. In this way, the payload can be of a *variable* length (provided that the header contains the length information or there is a way to recognize the end of a packet), and moreover, the payload can be forwarded to the next node as soon as the local switch is set. These deviations from traditional packet-switching has brought such approaches to optical packet-switching in WDM networks (e.g. see [12]) closer to what we call optical burst switching (OBS), which is to be discussed in Sec. 5.

4 Burst-Switching

In this section, we first describe, and then compare three variations of burst-switching, namely, *tell-and-go* (TAG), *in-band-terminator* (IBT) and, *reserve-a-fixed-duration* (RFD). Of the three, the first two have been studied for the electronic networks but the third has not. In all three variations, bandwidth is reserved at the burst level using a *one-way* process, and more importantly, a burst can *cut-through* switches, instead being stored and then forwarded. This is why, at least theoretically, the size of a burst may be unlimited (just as a call).

4.1 Design Choices

In TAG (based burst-switching), a source first sends a control packet on a separate control channel (similar to a circuit set-up request under distributed control) to reserve bandwidth (and set switches) along a path for the following data, which, unlike in circuit-switching, can be sent on a data channel without having to receive an acknowledgment first. This implies that the offset time T can be (much) less than the circuit set-up time, or even 0 as in packet-switching. After the burst is sent, another control signal (similar to a circuit tear-down signal) is sent to release the bandwidth.

In IBT, each burst has a header just as in packet-switching, as well as a special delimiter (called terminator in [1]) to indicate the end of the burst. Although the difference between IBT and packet-switching (in particular, message-switching), especially in terms of what triggers bandwidth allocation/deallocation, can be very subtle, we note that IBT uses virtual cut-through, instead of store-and-forward. More specifically, in IBT, a source and any intermediate node can transmit the *head* (not necessarily the header) of a burst *even before* the tail of the burst is received. Accordingly, a burst will encounter less delay and in addition, a smaller buffer space is needed at a node, except for the worse case where the entire burst has to be buffered because the bandwidth at the output is not available. Note that, TAG- and IBT-based burst-switching, when preceded by a call set-up phase for voice communications, is more or less the same as fast circuit-switching.

The third variation of burst-switching, RFD, has been studied for optical networks only (see for example, [3, 4, 14]), but not for electronic networks. RFD is similar to TAG in that a control packet is sent first to reserve bandwidth (and set switches), followed by data after an offset time T . What distinguishes RFD from TAG (and other circuit- packet-switching) is that in RFD, the bandwidth is reserved for a duration specified by the control packet which, like a header of a variable-length packet, contains the (expected) burst length. This implies, however, that a burst will have a limited maximum size.

4.2 Comparison of Burst-Switching Variations

Although the three variations use different triggers for bandwidth de-allocation (and allocation), a given burst will consume (almost) the same amount of bandwidth in an ideal situation. Nevertheless, as to be discussed next, RFD is more attractive because it can take advantage of the use of an appropriate offset time while avoiding its potential disadvantages more effectively than the other two variations.

One of the advantages of using an offset time, T , is that since data is buffered (or delayed) at its source, it does not need to wait at intermediate nodes for its corresponding control packet (or header) to be processed. Hence, no buffer is necessary at intermediate nodes, and even if there is buffer space in place at each intermediate node, 100% of the buffer space can be used for conflict resolution. The potential disadvantages of using an offset time include increased end-to-end latency and bandwidth waste. Note that these tradeoffs are reflected in circuit- and packet-switching, but burst-switching and especially RFD can combine the best of the two.

More specifically, in circuit-switching, having a T at least as large as $2P + \Delta$ eliminates the need for buffering data at any intermediate node. In burst-switching, the data can be sent before the last few channels are latched together, (that is, we may have $T < P + \Delta$), and *still* without having to be delayed at any intermediate node. This is because, as long as $T > \Delta$ and the bandwidth reservation by the corresponding control packet (or header) is successful, by the time the data arrives at a switch, the switch should have already been set. Of course, if T is made too small (e.g. $T = 0$ as in IBT), or congestion occurs such that no bandwidth can be reserved prior to the data arrival, the data needs to be delayed. If the data cannot be delayed, because either there is insufficient buffer space for it, or the delay it has encountered has reached the maximum that can be tolerated by the real-time application (e.g. voice communication), the data may be dropped or deflected (i.e. forwarded to an alternate output port).

However, having $T > 0$ may not be desirable for TAG (and IBT) if buffer space at the intermediate node is not a concern at all, as the amount of bandwidth reserved during the offset time will be wasted. Even if bandwidth can be reserved starting at the time of expected arrival of a burst, instead of at the completion of the processing of its control packet (or header), using *just-in-time switching* (as in two-way reservation protocols [6, 7]⁶), the achievable bandwidth utilization will not be as high as using $T = 0$ in TAG and IBT. This is because the (non-zero) offset time creates fragmented bandwidth which cannot be utilized efficiently by other bursts without possibly pre-empting or being pre-empted by the burst.

The case for RFD is different because now, it is possible for a node to make intelligent decisions, based on its knowledge of the duration of each reservation, on how to allocate bandwidth (as well as buffer) so that utilization can be as high as possible and close to that achievable by using $T = 0$. For instance, in RFD, it is possible to determine (1) whether it will be able to satisfy a request for reserving bandwidth without affecting existing reservations, and (2) if a burst needs to be delayed, exactly how long the delay

⁶See the protocol called RIT in [6].

should be and whether there is enough buffer space to provide such a delay. We will discuss several issues related to RFD in more detail along with optical burst switching (OBS) next.

5 Optical Burst Switching (OBS)

In optical burst switching (OBS), a data burst is kept in the optical domain at the intermediate nodes, while its control packet or header can be converted to electronics for processing. Since each burst is transmitted at the full-bandwidth of a wavelength in a WDM network while its control packet or header is transmitted on a separate wavelength, deciding how to schedule bandwidth reservation and set switches should be relatively simpler than in a TDM system (e.g. a telephone network employing pulse code modulation). Accordingly, arguments against fast circuit-switching (or burst-switching) based on its high processing complexity (as well as reasons why burst-switching has not been successful in electronic networks) are no longer valid for OBS. This is true especially with the high processing speed of today's microprocessors and the use of an offset time (as in RFD), which should give each node enough time to process control packets (or headers).

Although OBS may be based any of the three burst-switching variations described above, RFD-based OBS is the most attractive because, as discussed earlier, FDLs are scarce, and RFD is more efficient in utilizing bandwidth and FDLs (especially when using a non-zero offset time). Note that, in order to release reserved bandwidth in IBT-based OBS, the end-of-burst terminator needs to be detected, which could be difficult. Similarly, in TAG-based OBS (see the so-called TAG protocol in [6] terabit burst switching in [15]⁷), loss of a tear-down signal during its transmission will result in bandwidth waste. A possible alternative is to require a source to periodically send out a refresh signal, and only if no refresh signals are received after a time-out period, the bandwidth will be released. But this approach will generate many control signals and with a non-zero probability, result in undesired bandwidth release due to loss of refresh signals.

An important issue related to one-way reservation in general, and OBS in particular (since there is no optical buffer), is how to deal with contention and reduce burst dropping. Another important issue related to OBS using a non-zero offset time is the end-to-end latency encountered by each burst. These two issues are also fundamental when applying OBS to the next generation Optical Internet where differentiated services are to be provided. In the rest of the section, we first describe a RFD-based OBS protocol called Just-Enough-Time (JET) [3, 14], and then address the two issues in more details.

⁷Although in more recent publications such as [16], a RFD-based OBS protocol was proposed for terabit burst switching instead.

5.1 Just-Enough-Time (JET)

The basic idea of JET is shown in Figure 1. More specifically, a source sends out a control packet (similar to a set-up request), which is followed by a burst after an offset time, $T \geq \Delta$ (in Figure 1, it is assumed that the number of hops $H = 3$, and at each hop, the delay encountered by the control packet is δ . Hence, $\Delta = 3 \cdot \delta$). Because the burst is buffered at the source (in the electronic domain), no FDLs are necessary at each intermediate node to delay the burst while the control packet is being processed.

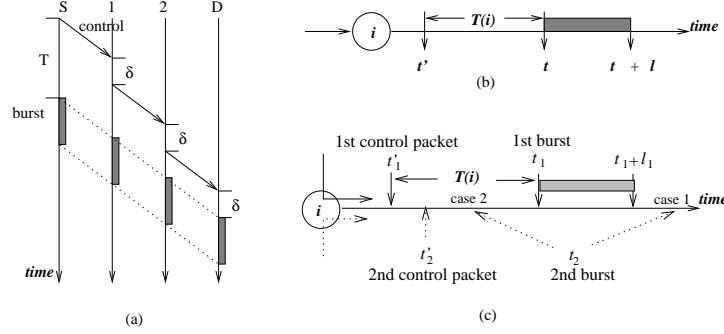


Figure 1: The use of an offset time and delayed reservation in Just-Enough-Time (JET)

A unique feature of JET is the use of delayed reservation (DR) as shown in Figure 1 (b), whereby the bandwidth on the output link at node i (e.g. $i = 1, 2$) is reserved from t , the time at which the burst is expected to arrive, instead of from t' , the time at which the processing of the control packet finishes (and the request for bandwidth reservation is made)⁸. In addition, the bandwidth will be reserved until the burst departure time, $t + l$, where l is the (expected) burst length.

The above discussion implies that in JET, a control packet will include not only the burst length l (as in any RFD protocol), but also the (remaining) value of the offset time T_{offset} (initially, $T_{offset} = T$). To cope with the variable processing delay encountered by a control packet as well as any receiving and transmission delay at each node, the control packet can be *stamped* with its arrival time t_{in} , and *scheduled* for transmission at time t_{out} (as soon as its processing is done), where $t_{in} < t' < t_{out}$. In this way, bandwidth will be reserved from $t = T_{offset} + t_{in}$ (this means that in the figure, $T(i) = T_{offset} - (t' - t_{in})$), and the control packet will carry an updated value of T_{offset} to the next node, which is $T_{offset} - (t_{out} - t_{in})$.

If the requested bandwidth is not available, the burst is said to be blocked, and will be dropped if it cannot be buffered (a dropped burst may then be retransmitted later if necessary). The use of DR can reduce burst dropping (and increase bandwidth utilization) even without using any buffer, as illustrated in Figure 1 (c). More specifically, when the 2nd control packet arrives, it knows that if either $t_2 > t_1 + l_1$ (case 1) or $t_2 + l_2 < t_1$ (case 2), bandwidth for the 2nd burst can be successfully reserved. Note that, if a TAG- or IBT-based OBS protocol is used, there is no way for the 2nd control packet to know that the

⁸Note that $t = t' + T(i)$, where $T(i)$ depends on the remaining value of T as well as the processing time at node i .

bandwidth will be released before the 2nd burst will arrive (case 1), or that the length of the 2nd burst is short enough (case 2).

JET can also take advantage of any FDLs available at an intermediate node by using the FDLs to delay a blocked burst until bandwidth becomes available (even though FDLs are not mandatory in JET). As mentioned earlier, by taking advantage of the information on the duration of each reservation, DR can increase the effectiveness of the available FDLs, just as it can increase bandwidth utilization through scheduling⁹ (for more detailed discussions and quantitative results, see [3, 14]). In addition, if the control delay is relatively large compared to the average burst length, then with the same FDLs, JET will achieve a better performance (e.g. a lower burst dropping probability) than optical packet/cell switching and other OBS protocols that do not use any offset time (i.e. $T = 0$). This is because JET can use 100% of the available FDLs for the purpose of resolving conflicts but these protocols cannot (due to the fact that some FDLs must be used to delay the burst while the header or control packet is being processed).

5.2 Application to the Next Generation Optical Internet

Recently, there have been several initiatives in building an Optical Internet where IP routers are interconnected directly with WDM links (see e.g. [18]). Such an Optical Internet will reduce the control overhead due to the high ATM “cell tax” as well as complex signaling protocols, and eliminate the need for the expensive SONET/SDH switches. It is envisioned that in the *next generation* Optical Internet, IP will run over a WDM layer consisting of WDM switches and WDM links. Having the WDM layer will enable a huge amount of “through” traffic to be switched in the optical domain, and as a result, can reduce the number of expensive terabit routers and high-speed transceivers required at the IP layer (in addition to creating high-speed communication pipes that are transparent to bit-rate and coding format, as mentioned earlier).

OBS can be applied to the next generation Optical Internet as follows. A control packet may be processed by each and every intermediate node running IP to set up a WDM switch and reserve bandwidth on an outgoing wavelength channel, so that the corresponding burst (e.g. several IP packets) will go through only the WDM switches at the intermediate nodes [19]. To further reduce the overhead involved in IP (or layer 3) forwarding, the control packets corresponding to multiple bursts of IP packets belonging to the same IP flow may be *label switched* (i.e. forwarded at layer 2) as in MPLS, by letting the first few control packets set up a label switched path (LSP) for the subsequent ones. Note that, if JET is used, each of the subsequent control packets will still need to carry control information such as the offset time and the duration of the reservation, in addition to a label. Such an approach, which may be called labeled OBS (or LOBS), will be especially useful for multicasting bursty IP traffic at the WDM layer [20].

⁹Scheduling is also useful in RFD-based OBS without an offset time, or in optical packet switching with variable-length packets [17] which, as mentioned earlier, can be quite similar to OBS.

Several related switching techniques for the next generation Optical Internet have also been proposed. For example, a wavelength-routing based technique was proposed in [21], whereby IP packets are first forwarded as usual (i.e. by the IP layer), but as soon as an IP flow is recognized, a wavelength is assigned as a label to all subsequent IP packets of the flow. Such an approach is thus based on wavelength routing as a wavelength path needs to be established.

For the next generation Optical Internet to be ubiquitous, it is important for the WDM layer to be able to provide low burst-dropping probability for high-priority traffic and limited end-to-end latency for delay-sensitive (real-time) traffic. Such a WDM layer capable of supporting basic QoS (e.g. differentiated services) will facilitate as well as complement a QoS-enhanced version of IP (which currently only provides best-effort services). Furthermore, it is necessary not only for carrying some WDM layer traffic such as those for signaling and protection/restoration purposes, which require a higher priority (and a low latency) than other ordinary traffic, but also for supporting certain applications directly (i.e. bypassing IP) or indirectly through other legacy or new protocols incapable of QoS support.

5.3 Prioritized OBS

Lack of buffers in the WDM layer not only makes burst dropping more likely, but also makes existing priority schemes no longer applicable. In this subsection, we will discuss how JET can be extended to support priority and accordingly, reduce the probability of dropping bursts carrying crucial information in the presence of congestion.

Consider a *prioritized OBS* protocol called pJET, where bursts are classified into multiple (e.g. two) classes, and differentiated services are to be provided. For example, class 0 corresponds to best-effort services and can be used for non-real-time applications such as email and FTP, while class 1 corresponds to priority services and can be used for delay sensitive applications such as real-time audio and video communications. Since a dropped class 0 burst may be retransmitted but not a dropped class 1 burst (due to its stringent delay constraint), it is desirable to assign class 1 bursts a higher priority than class 0 bursts when reserving bandwidth to ensure that class 1 bursts incur a lower blocking (dropping) probability.

The main idea of pJET is to assign an *extra* offset time, denoted by t_{offset} , to each class 1 burst (but only a “base” offset time T is used when sending each class 0 burst), while all control requests are still treated equal, i.e. processed in the first-come-first-served (FCFS) order. Intuitively, this extra offset time allows a control packet corresponding to a class 1 burst to make bandwidth reservation in much more advance, thus giving it a greater chance of success, than the control packet for a class 0 burst, which can only “buy tickets at door”.

To illustrate the principle of pJET using terms common to queueing systems, let t_{ai} and t_{si} be the arrival time and the service-start time respectively, of a class i request, denoted by $req(i)$, where $i = 0, 1$. Also, let l_i be the service time (i.e. burst length) requested by $req(i)$. To simplify the following presentation,

let us assume no FDLs at any intermediate node. In addition, we will ignore the effect of the base offset time assigned to both classes of bursts and concentrate on that of the extra offset time assigned only to class 1 bursts.

Since no extra offset time is given to class 0 bursts, a class 0 request, $req(0)$, will try to reserve bandwidth immediate upon its arrival, and will be serviced right away if bandwidth is available (and dropped otherwise). In other words, $t_{a0} = t_{s0}$ when reservation is successful (see Figure 2(b)). However, for a class 1 request, $req(1)$, a delayed reservation is made with an extra offset time, t_{offset} , and hence, it will be serviced at $t_{s1} = t_{a1} + t_{offset}$ when reservation is successful (see Figure 2(a)).

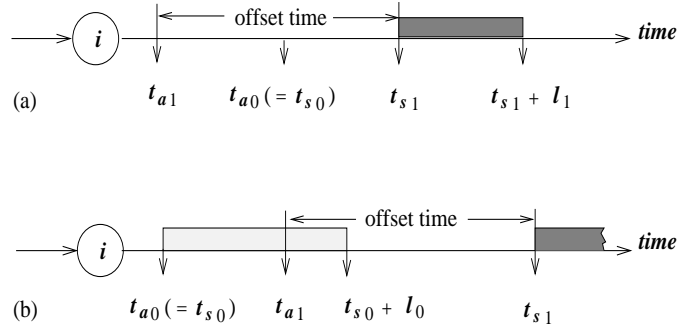


Figure 2: Priority scheme using the offset time combined with DR

Figure 2 illustrates why a class 1 request that is assigned t_{offset} can obtain a higher priority for reservation than a class 0 request that is not. Consider the following two cases where contention between two requests in different classes is possible. In the first case illustrated in Figure 2(a), $req(1)$ arrives first and reserves the bandwidth (using delayed reservation), and $req(0)$ arrives afterwards. Clearly, $req(1)$ will succeed, but $req(0)$ will be blocked if $t_{a0} < t_{s1}$ but $t_{a0} + l_0 > t_{s1}$, or if $t_{s1} < t_{a0} < t_{s1} + l_1$. In the second case illustrated in Figure 2(b), $req(0)$ arrives first, followed by $req(1)$. When $t_{a1} < t_{a0} + l_0$, $req(1)$ would be blocked had t_{offset} not been assigned to $req(1)$. However, such a blocking is avoided as long as $t_{s1} = t_{a1} + t_{offset} > t_{a0} + l_0$.

If $t_{a1} = t_{a0} + \sigma$, where $\sigma > 0$ is very small, t_{offset} needs to be longer than the maximum burst length over all class 0 bursts in order for $req(1)$ to completely avoid being blocked by any $req(0)$. With that much extra offset time, the blocking probability of class 1 bursts becomes independent of the offered load in class 0, that is, class 1 is completely (i.e. 100%) isolated from class 0.

Note that however, a reasonable t_{offset} can be used to achieve a sufficient degree of class isolation. For example, if the length of class 0 bursts are exponentially distributed with an average of L_0 , then $t_{offset} = 3 \cdot L_0$ is sufficient to achieve at least 95% isolation [22]. In addition, with a reasonable degree of class isolation, class 1 bursts will have a blocking probability that is several orders of magnitude lower than class 0 bursts, although the overall blocking probability does not depend on the degree of isolation but rather other factors such as the overall traffic load and number of wavelengths. The results in [22] have

also shown that even with a link load of 0.8 and 40 wavelengths, the blocking probability of class 1 bursts can be as low as 10^{-7} .

5.4 Pre-transmission Delay

For real-time bursts, it is imperative to discuss their end-to-end delay. We first note that the use of a (base) offset time as in JET does not increase the end-to-end delay of a burst as the offset merely substitutes for the total processing delay to be encountered by the corresponding control packet. Compared to circuit-switching, the end-to-end delay in OBS is about $2P$ shorter (where P is the total propagation delay between the burst's source and destination switches, and is typically around tens of milliseconds or ms coast-to-coast). In addition, compared to IP routing which uses store-and-forward, OBS can reduce the end-to-end delay because the data can cut-through the WDM switches as described earlier.

To assess the impact of using an extra offset time as in pJET, let the total processing delay be Δ (typically tens of microseconds or μs) and the length of a burst be l (typically a few μs or less). In addition, assume that in a real-time application, each byte (or bit) generated at the source has to reach its destination in D ms , which is often large compared to Δ or even P . For example, for today's voice and video communications, it may be acceptable for D to be as large as a few hundreds of ms . Nevertheless, if one takes into account the delays introduced by the higher-layer protocols at the source and destination, the available budget for the total delays in the WDM layer, denoted by B , could be significantly smaller.

Let $B = P + \Delta + b$, where b is the available budget for pre-transmission delay and could be as large as several ms ¹⁰. Accordingly, if the extra offset time used in pJET is equal to a few times of L_0 (the average burst length), the increase in the end-to-end delay may not be significant (although the reduction in the blocking probability of higher priority bursts could be significant as discussed earlier).

Note that in RFD-based OBS (such as JET and pJET), it is required that a control packet specify the duration of the following burst, where a burst usually consists of all the IP packets belonging to the same data message. Such a requirement can be easily met when each burst can be assembled in less than a few ms (i.e. the burst assembly time is shorter than the tolerable pre-transmission delay b). Nevertheless, there may be cases where a message is long, the data belonging to the same message arrives (from the IP layer) at a slow rate, and/or each message is too short so that IP packets belonging to different messages (or flows) may need to be assembled into one burst in order to lower the percentage of the overhead introduced by the control packet. In such cases, the time to assemble a burst with all the desirable content, A , may be larger than b .

More specifically, let the time that the burst assembly starts be t_0 . If $A > b$ (which does not necessarily

¹⁰The assumption here is that $b < 2P$ so circuit-switching is out of question. Note that even if b is as large as $2P$, OBS may still be preferred since retransmissions (of a control packet and a burst) may be possible when using OBS, but not when using circuit-switching (after a set-up request fails).

mean that $l > b$ as well since $A \geq l$), an obvious solution is to stop accumulating data prior to (or at) time $(t_0 + b)$ (or in pJET where an extra offset time t_{offset} is used, before $(t_0 + b - t_{offset})$), and immediately send out a control packet specifying the length of the (partial) burst assembled so far, say l' .

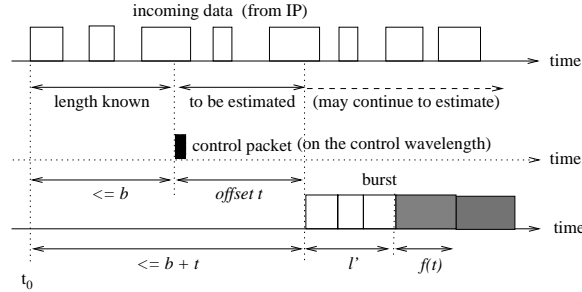


Figure 3: A control packet may be sent with an estimated burst length

If the total offset time t (which may include both the base and extra offset time in pJET) to be used is relatively large (e.g. due to a large Δ), an alternative to the above solution is to continue to accumulate data for another t μ s, but still send a control packet prior to time $(t_0 + b)$ (or $t_0 + b - t_{offset}$ in pJET), which will specify the *expected* length of the (partial) burst to be transmitted. As illustrated in Figure 3, the control packet will specify the length to be $l = l' + f(t)$, where $f(t)$ is the *estimated* average burst assembly rate during the next t μ s (which may be calculated based on the actual rate observed so far). Note that if it is an over-estimation, some extra bandwidth reserved by the control packet will be wasted. On the other hand, if it is an under-estimation, additional data accumulated will have to be transmitted later as a separate (partial) burst. The potential advantage of this alternative is that more data can be sent, thus reducing the percentage of the overhead introduced by the control packets.

Note that we may extend the estimating period t to include the burst transmission time as well. Such a flexibility of OBS in sending out a control packet while a burst is still being assembled, combined with the fact that the line speed of IP routers is reaching OC-48, enables OBS to efficiently support real-time applications such as voice or video over IP (over WDM) that generate data periodically. In addition, in some applications such as file transfers, WWW downloads or video-on-demand, a server can determine the burst length as soon as a request for transferring/downloading a file from its client is processed. Accordingly, the server can send out a control packet specifying the exact burst duration even before the file is retrieved from a storage unit. This essentially overlaps the processing of the control packet at the intermediate nodes with the file retrieval operation, thus reducing the overall end-to-end delay. Finally, as mentioned earlier, in order to reduce delay (and overhead) of layer 3 forwarding of the control packets, labeled optical burst switching (or LOBS) (e.g. similar to [12]) may be adopted for for a long flow (of bursts of IP packets), where the control packets will be forwarded at layer 2.

6 Summary

Optical burst switching (OBS) have the following main features: (1) data is sent in bursts (of packets), implying it can have a variable size, (2) data is transmitted at the bandwidth of a wavelength (not a fiber), (3) data can cut-through optical switches, implying that a burst may also be transmitted before it is completely assembled (and received) at the source (and each intermediate node, respectively), (4) control packets are transmitted on a separate band (wavelength), and possibly with an offset time as well (e.g. in RFD and TAG), (5) bandwidth is consumed only when data is transferred, and (6) the duration of each bandwidth reservation (or the size of the data) may be specified (as in RFD) or unlimited (as in TAG and IBT).

Some of these features (e.g. (3)) are similar to those of optical circuit-switching while some others (e.g. (5)) are similar to those of optical packet-switching. Consequently, OBS will require a limited or even no delaying (or buffering) of the data at intermediate nodes as optical circuit-switching, and achieve an efficient bandwidth utilization when supporting bursty traffic as optical packet-switching. In addition, OBS will result in a smaller (amortized) overhead involved in processing control signals, setting switches and achieving synchronization than optical packet-switching, and a lower pre-transmission (and end-to-end) latency than optical circuit-switching.

Note that, based on the current and near future WDM technologies and expected traffic to be supported (e.g. IP traffic) by the WDM layer, the question of whether there will be an single switching technique (and if so, which one) for use by the optical WDM layer is still open for debate. Nevertheless, as OBS achieves a balance between optical circuit- and packet-switching, it can serve as a base for any future convergence.

References

- [1] E. Haselton, "A PCM frame switching concept leading to burst switching network architecture," *IEEE Communications Magazine*, vol. 21, pp. 13–19, June 1983.
- [2] S. Amstutz, "Burst switching - an introduction," *IEEE Communications Magazine*, vol. 21, pp. 36–42, Nov. 1983.
- [3] M. Yoo and C. Qiao, "Just-enough-time(JET): a high speed protocol for bursty traffic in optical networks," in *Digest of IEEE/LEOS Summer Topical Meetings on Technologies for a Global Information Infrastructure*, pp. 26–27, Aug. 1997.
- [4] C. Qiao, "Optical burst switching - a new paradigm," in *Optical Internet Workshop*, Oct. 1997. (see related links at <http://www.isi.edu/workshop/oi97/>).
- [5] B. Davie, P. Doolan, and Y. Rekhter, *Switching in IP Networks*. Morgan Kaufmann, San Francisco, CA, 1998.

- [6] G. Hudek and D. Muder, "Signaling analysis for a multi-switch all-optical network," in *Proceedings of Int'l Conf. on Communication (ICC)*, pp. 1206–1210, June 1995.
- [7] E. Varvarigos and V. Sharma, "The ERVC protocol for the Thunder and Lightning network: operation, formal description and proof of correctness," Tech. Rep. CIPR 95-05, ECE Dept, UC Santa Barbara, June 1995.
- [8] ITU-T Rec. I.371, "Traffic control and congestion control in B-ISDN." Perth, U.K. Nov. 6-14, 1995.
- [9] P. E. Boyer and D. P. Tranchier, "A reservation principle with applications to the ATM traffic control," *Computer Networks and ISDN Systems*, vol. 24, pp. 321–334, 1992.
- [10] D. Blumenthal, P. Prucnal, and J. Sauer, "Photonic packet switches - architectures and experimental implementations," *Proceedings of the IEEE*, vol. 82, pp. 1650–1667, Nov. 1994.
- [11] D. Hunter et al., "SLOB: a switch with large optical buffer for packet-switching," *IEEE/OSA Journal of Lightwave Technology*, vol. 16, no. 10, pp. 1725–1736, 1998.
- [12] G. Chang, "Optical label switching," in *DARPA/ITO Next Generation Internet PI Meeting*, Oct. 1998. (see <http://www.dynacorp-is.com/darpa/meetings/ngi98oct/agenda.html>).
- [13] D. Blumenthal et. al, "WDM optical IP tag switching with packet-rate wavelength conversion and subcarrier multiplexing addressing," in *Optical Fiber Comm. Conf(OFC)*, pp. 162–164, Feb. 1999. Paper ThM1.
- [14] M. Yoo, M. Jeong, and C. Qiao, "A high-speed protocol for bursty traffic in optical networks," in *SPIE Proceedings, All Optical Communication Systems: Architecture, Control and Network Issues*, vol. 3230, pp. 79–90, Nov. 1997.
- [15] J. S. Turner, "Terabit burst switching," Tech. Rep. WUCS-97-49, Department of Computer Science, Washington University, Dec. 1997.
- [16] J. S. Turner, "Terabit burst switching," *J. High Speed Networks (JHSN)*, vol. 8, no. 1, pp. 3–16, 1999.
- [17] L. Tancevski et al., "A new scheduling algorithm for asynchronous variable-length IP traffic incorporating void filling," in *Proc. Optical Fiber Communication Conference*, pp. 180–182, 1998.
- [18] B. Arnaud, "Architectural and engineering issues for building an optical internet," in *SPIE Proceedings, All optical Communication Systems: Architecture, Control and Network Issues*, vol. 3531, pp. 358–377, Nov. 1998.
- [19] C. Qiao and M. Yoo, "Optical burst switching (OBS) - a new paradigm for an Optical Internet," *J. High Speed Networks (JHSN)*, vol. 8, no. 1, pp. 69–84, 1999.
- [20] C. Qiao et al., "Multicasting in IP over WDM networks," Tech. Rep. 99-05, CSE Dept, University at Buffalo (SUNY), May 1999.

- [21] J. Bannister, J. Touch, A. Willner, and S. Suryaputra, “How many wavelengths do we really need in an optical backbone network,” in *IEEE Gigabit Networking Workshop (GBN)*, 1999. (see related links at <http://www.isi.edu/touch/pubs/gbn99/>).
- [22] M. Yoo and C. Qiao, “A new optical burst switching protocol for supporting quality of service,” in *SPIE Proceedings, All Optical Networking: Architecture, Control and Management Issues*, vol. 3531, pp. 396–405, Nov. 1998.