Content Aware Burst Assembly – Supporting Telesurgery and Telemedicine in Optical Burst Switching Networks*

Henry Orosco, Lei Wang and Yuhua Chen
Department of Electrical and Computer Engineering, University of Houston
Houston, TX, 77204-4005, USA

Abstract - The emerging Telemedicine and Telesurgery technologies allow patients to share medical experts remotely through communication networks. However, network bandwidth, network latency and jitter (variation of latency), are the obstacles to the widespread use of this technology remotely. Optical Burst Switching (OBS) networks greatly expand network bandwidth in existing network infrastructure by utilizing multiple DWDM channels within a single fiber, enabling high bandwidth applications. However, the burst assembly process in OBS networks introduces latency and jitter, making it unsuitable for high bandwidth, latency sensitive applications such as telesurgery and telemedicine. In this paper, we propose a content aware burst assembly scheme which dynamically adjusts the burst assembly parameters based on the content being assembled. The proposed content aware burst assembly minimizes the latency and jitter within a video frame, as well as across the left-view and right-view frames for 3D vision generation. Simulation results have shown that the proposed scheme can effectively reduce the latency and jitter experienced by video streams, making OBS a promising candidate for supporting telesurgery and telemedicine applications.

Keywords: Optical Burst Switching, OBS, Burst Assembly, Telesurgery, Jitter

1. INTRODUCTION

The emerging Telemedicine and Telesurgery technologies allow patients to share medical experts remotely through communication networks. For example, telesurgery allows a surgeon to perform surgery on a remote patient with the help of robots. In fact, surgical robots are currently being used by surgeons worldwide (but just a few feet away from the patient) to conduct minimally invasive surgeries, which replace traditional open surgeries with 1-2 cm incisions. Theoretically, robotic surgery systems such as da Vinci Surgical Systems [1] can be operated over long distances. However, network bandwidth, latency and jitter (variation of latency) are the obstacles to the widespread use of this technology remotely.

Dense Wavelength Division Multiplexing (DWDM) allows multiple wavelengths, essentially different colors of light, to carry data over a single optical fiber. Each DWDM channel can carry data at 10 Gb/s and beyond, greatly expanding the network capacity over the existing network infrastructure. Optical switching technologies can efficiently support DWDM signals by allowing data to be transported over router nodes without optical-electrical-optical (O/E/O) conversion.

Optical switching technologies can be characterized into optical circuit switching, optical packet switching, and optical burst switching. Optical circuit switching (OCS), also known as wavelength routing, switches data at the wavelength level. In OCS, data passes through routers in pre-established lightpaths. Although the technology has been available for many years, its widespread deployment has been lacking due to the coarse granularity. Optical packet switching (OPS) switches data on a packet by packet basis optically. Despite the fine granularity it promises, optical packet switching is not practical in the foreseeable future due to the synchronization issues associated with the packet header and payload, and the lack of random access optical buffers. Optical burst switching (OBS) allows variable size data bursts to pass core router nodes optically by setting up an optical path on the fly. OBS has a granularity between optical circuit switching and optical packet switching. In OBS, a control header is sent ahead of its corresponding data burst on a separate wavelength to configure the optical switching fabric. The decoupling of the control header and the data burst allows OBS to bypass the synchronization problem which OPS experiences. Currently, OBS is considered the most promising optical switching technology.

* This work is supported in part by the National Science Foundation (NSF) under Grant CNS-0708613, and the Texas Advanced Research Program (ARP) under Grant G096059.
Although OBS can support high bandwidth applications, they are not suitable for applications that require both high bandwidth and low latency. One of the reasons is that in OBS networks, packets are assembled at the ingress router into bursts, and are disassembled back to packets at the egress edge router. The burst assembly process introduces additional latency and jitter which threaten the safety of telesurgery. In this paper, we propose a content aware burst assembly scheme which dynamically adjusts the burst assembly parameters based on the content being assembled. To support multi-view high definition (HD) video used in telesurgery for 3D vision generation, the proposed content aware burst assembly minimizes the latency and jitter within a video frame, as well as across the left-view and right-view frames for 3D vision generation. Simulation results show that the proposed scheme can effectively reduce the latency and jitter experienced by video streams, making OBS a promising candidate for supporting bandwidth demanding and latency sensitive applications such as telesurgery and telemedicine.

The rest of the paper is organized as follows. Section 2 provides the background of OBS networks and burst assembly process. The proposed content aware burst assembly scheme is described in Section 3. In Section 4, we evaluate the performance of the proposed scheme using software simulation. We conclude our work in Section 5.

2. OPTICAL BURST SWITCHING (OBS) NETWORK BACKGROUND

Optical Burst Switching (OBS) Network Overview

Optical Burst Switching (OBS) networks consist of a set of edge routers and core routers. Traffic from upper layers gathers at edge nodes [4], which are responsible for assembling packets into bursts based on destination edge router addresses and burst assembly algorithms. After a burst is formed, the burst header is created and sent ahead of the data burst by an offset time. The burst header and the data burst are loosely coupled physically, as they do not share the same wavelength. The offset between the burst header and the data burst is determined by how long it takes for the burst header to set up an optical path for the data burst in the OBS core network. The offset should be large enough such that data burst never catches the burst header before reaching the egress edge node. If the data burst reaches a core router before the burst header is processed, it will be subsequently dropped.

The core routers use information carried in the burst header to determine where to send the burst and when to reserve a wavelength for the corresponding data burst. There are several wavelength resource management schemes: explicit setup, estimated setup, explicit release, and estimated release [2] [4]. With explicit setup, a connection for the data burst is made immediately upon receiving the burst header at the core router node. In estimated setup, the core router takes the offset information from the burst header and calculates when it needs to reserve the connection in coordination with the arrival of the data burst. With explicit release, the edge node sends another burst control signaling the end of the transmission of the data burst. Consequently, resources are released after receiving such connection termination request. Estimated release, similar to the estimated setup, will calculate when the resources can be released based on the data burst length parameter carried in the burst header. Just-in-Time (JIT) and Just-Enough-Time (JET) are the two protocols based on the respective configurations of the set up and release of resources described above. More specifically, JIT utilizes the explicit setup/explicit release and explicit setup [2][4][9]. JET utilizes the estimated setup/estimated release configuration [2][3][4]. OBS networks also take advantage of the advancement in DWDM which allows multiple wavelengths to carry different data simultaneously in the same optical fiber [10].

Burst Assembly in OBS Networks

In OBS core networks, the basic processing unit is a burst, which is a collection of packets. Packets arrived at the ingress edge router are assembled into bursts based on their destinations. Once a burst is formed, it is transported as an entity in the OBS core network until it reached the egress edge router, where it is disassembled into packets. The criteria of forming a burst in the burst assembly process include [11]: (1) time based assembly, (2) burst length based assembly, (3) mixed time/burst length based assembly, and (4) dynamic assembly. Time-based assembly has a time threshold that, when passed, stops assembling the burst and prepares it for transmission across the network. This algorithm is useful when there is a light traffic load at the assembler. Burst length based assembly has a maximum burst length threshold which triggers the formation of the burst when the threshold is crossed. As a result, burst will form faster under heavy traffic. In the mixed time/burst length-based assembly, a burst is formed when it reaches the maximum burst size, or the timeout threshold is crossed. The mixed burst assembly works under both light and heavy traffic cases,
and is the most popular burst assembly scheme. Dynamic burst assembly can reduce burst loss probability by adjusting the parameters based on the feedback from the measured burst loss in the core network.

Burst assembly algorithms have also been extended to support Quality-of-Service (QoS). For example, the offset based priority scheme adjusts the offset between the burst header and data burst. A larger offset value is given to higher priority burst, allowing it to be scheduled ahead of the lower priority bursts and having better chance of being scheduled successfully [2][12][13]. While such a scheme reduces the burst loss for high priority bursts, it also increases the latency experienced by high priority bursts, and is not suitable for supporting latency sensitive applications such as telesurgery and telemedicine.

In this next section, we propose a content aware burst assembly scheme where burst assembly parameters are adjusted according to the content being assembled, minimizing latency and jitter experienced by latency and jitter sensitive applications.

3. SUPPORTING TELESURGERY AND TELEMEDICINE OVER OBS NETWORKS

Telesurgery Basics

Telesurgery allows surgeon to perform surgery on remote patient with the help of a surgical robot. Many advances in telesurgery have been made in past years. The world’s first telesurgery was performed by researchers at NASA’s Jet Propulsion Laboratory (JPL) and the Telerobotics Laboratory of the Politecnico di Milano on July 7th, 1993 [6]. A robot in Italy was controlled from the USA to perform a biopsy on a model. The setup utilized two geosynchronous satellites and optical fibers to complete the network across continents. The use of satellites and optical fibers to perform surgery has been demonstrated, although it is still far from a practical use of telesurgery. Other possibilities such as using various technologies have been investigated [7][8].

However, all above mentioned efforts are still far from making telesurgery a reality. Nevertheless, some lessons have been learned. Some recognized the need to segregate data and use different protocols accordingly [8], while others looked into the requirements of the network [5][7]. Putting all pieces together, a practical solution to telesurgery will require the use of optical networks, the segregation of data, and the network support of QoS to reduce latency and jitter and to ensure the transmission bandwidth of data.

A telesurgery system is an integrated system which consists of the surgeon side and the patient side connected by a network. We need to be concerned with the amount of data produced by the system, as well as round-trip delay which is crucial to the safety of telesurgery.

At the remote patient side, multi-view high definition (HD) video camera is used to produce the left and right vision for 3D vision reconstruction at the surgeon side. In addition, the surgical robot is controlled remotely by the surgeon. Other personnel and equipment support will include vital sign monitors, microphones/speakers and nurses to help with the surgical procedures. HD video, and usually 4xHD video (four times the resolution of HD video) are recommended for medical use. Multi-view video helps to reconstruct 3D vision at the surgeon site, providing the doctor with the depth perception necessary to carry out an accurate operation.

At the surgeon site, the 3D viewer allows the doctor interpret the stereo vision. In addition, a surgical robotic control interface, vitals display, and a microphone/speaker are necessary. The HD video and vital signs will be transmitted from the patient site to the surgeon site, while the robotic control data is sent from the surgeon site to the remote patient site. Voice communication is bidirectional.

While all above mentioned types of data are of great importance in the success of telesurgery, the round-trip latency from the issuing of robotic control signal to the resulting video displayed at the surgeon site determines the safety of telesurgery. If the robotic control signal gets delayed, it will result in a delayed action of the surgical robot. The result of delaying a stop command can be catastrophic. Video transmission from the patient site to the surgeon site is of equal importance. Without being able to see the result of a robotic control command on patient in time will hinder the doctor’s ability to make decision on the next movement.

Transmitting multi-view HD video presents a challenge. Multi-view video consists of the left view and the right view, each of which consists of a series of video frames. Each frame contains millions of pixels. In the case of HD video, each video frame is 1920 by 1080 pixels. When stereo vision is used, since both the left and the right frames are correlated by time, it is important to maintain this correlation to keep the integrity of the video
displayed. If this correlation is lost, right and left images would be from different times and would incorrectly represent the observed object, causing blurred vision. To transmit the video over the network, the video frames are broken into smaller packets. The video data can also be encoded or compressed to reduce the amount of data transmitted over the network.

**Problem in Transporting Telesurgery Data in OBS Networks**

As we have mentioned before, an optical network is probably the best candidate for supporting telesurgery. Although OBS network can provide the necessary bandwidth for telesurgery, latency and jitter are the two major obstacles in such endeavor. Traditional burst assembly processes introduce latency and jitter to telesurgery. Since jitter is the secondary effect of latency, it is even harder to control than latency itself, which is already a hard problem to solve.

For example, the burst assembly process can introduce jitter within a video frame as a result of different delays experienced by different pixels. Fig. 1 shows a time-based burst assembly process. In the figure, only a single video stream is illustrated. When a video frame is being assembled into a burst, the time threshold may be crossed before the entire frame is included into the burst. Once a burst is formed, triggered by a timeout, the remaining pixels in the same frame have to be assembled into the next burst, resulting in a large discrepancy in latency experienced by different pixels in the same frame. At the destination node, when the first burst is received, since it only contains a partial frame, the receiving video buffer has to wait for the remaining portion of the frame to be received before forwarding the frame to display. The resulting delay in video frame can cause stalled video, threatening the safety of telesurgery.

Similarly, the same problem exists in a fixed burst size based burst assembly shown in Fig. 1. If the burst length threshold is reached before the last few pixels are assembled into the burst, the remaining part has to wait till the next burst is assembled. This will result in video jitter at the destination. Although the problem might be eased by increasing the maximum burst size, such approach does not work with compressed video stream, and can increase burst assembly time unnecessarily.

The problem will be worse if multi-view video is used. Fig. 2 shows a case of assembling the stereo vision into bursts. Using traditional burst assembly schemes, both the left and the right view video frames will be assembled into the same burst, as they are of the same destination as well as the same priority. This will cause an interleaving of the left and the right video streams. The interleaving will increase the probability that either or both of the frames will be incomplete when a burst is formed according to either the time based or burst size based threshold. Consequently, the video buffer at the destination node would have to wait for one or more bursts before the multi-view video frames can be displayed. Another consequence of interleaving is the left and the right frames may be out of synchronization and are no longer correlated by time when displayed. This could produce a current image for the left video but a stale image for the right video or vice versa, which can cause incorrect representation of the object and blurred vision. This is again another result caused by the jitter introduced in the network.
Proposed Content Aware Burst Assembly for Telesurgery

In this paper, we propose a content aware burst assembly scheme to reduce the jitter caused by the burst assembly process. The proposed content aware burst assembly dynamically adjusts the burst assembly parameters such as the timeout value or the maximum burst length based on the content being assembled. The maximum allowable burst length can be dynamically adjusted to take into account the completeness of the frames being assembled. For example, the actual burst length can be stretched to include the last few pixels within the same video frame, reducing the time that the pixels spent at the receiving buffers. The content aware burst assembly scheme also takes input from the video encoder to accommodate different video encoding schemes.

In addition, the proposed content aware burst assembly allows two or more bursts to the same destination to be assembled simultaneously. For example, the left and the right view frames can be assembled into two separate bursts, and transmitted concurrently on the DWDM channels, minimizing the jitter experienced by the pixels across the two views. As a result, the proposed content aware burst assembly can effectively minimize the latency and jitter within a video frame, as well as across the left-view and right-view frames for 3D vision generation.

Fig. 3 shows an example of assembling the left and the right view video frames into two separate bursts, even though they are of the same destination. Once the bursts are formed, they will be transmitted on two different wavelengths simultaneously, greatly reducing the cross view jitter. It also reduces the complexity at the receiving end to separate the two views.

The proposed content aware burst assembly dynamically adjusts the number of bursts being assembled for each destination, as well as how to trigger the formation of the bursts being assembled based on the content of the data. To gain knowledge of the content being assembled, the proposed architecture utilizes content monitoring agents for individual data type received by the edge router. The monitoring feature can be dynamically turned on and off for each type of data. The content monitoring agent will generate proper tags for the data type it monitors which will in turn affect the burst assembler’s decision in the burst assembly process.

4. PERFORMANCE ANALYSIS

In this section, we evaluate the performance of the proposed content aware burst assembly using a modified version of ns2-OBS simulator. We gather the jitter information of the proposed scheme, and compare it with the traditional burst assembly process.

Simulation Setup

We simulate both the traditional burst assembly and the proposed content aware burst assemble using a 14-node, 21-link NSFNET topology as shown in Fig. 4. Each link carries 4 DWDM channels at 1 Gb/s in each direction. Two video streams composed of HDTV video compressed using MPEG2-TS formats are used to compare the proposed scheme with the traditional scheme.

The first stream tested is 1080p25 (e.g. resolution 1920x1080 at a 25Hz frame rate), 18Mbps average, 30Mbps peak and the second stream is 1080p29.97 (e.g. resolution 1920x1080 at a 29.97Hz frame rate), 25Mbps average, 37Mbps peak. Video packets arrive at the burst
assembler following a Poisson process and the average frame size is calculated as a function of the average stream rate divided by the frame rate; the resulting average frame sizes are 90,000 bytes and 104,166 bytes for the 1080p25 and 1080p29.97 streams, respectively. To account for the variation in the frame size due to compression, it is modeled as exponentially distributed. According to the MPEG2-TS format, each packet has a constant size of 188 bytes.

For the current burst assembly methods, a burst size threshold, 160,000 bytes, and a time threshold, 0.1s, are used. Since we focus on the burst size threshold in this particular experiment, the data aware assembler will utilize a dynamic burst size threshold, allowing the pixels at the end of a frame to be included in the burst even if the burst size threshold is crossed. To avoid supersize bursts, we place a limit on the dynamic threshold: The remainder of the frame must not be longer than one-third of the original burst size threshold to be sent with the current burst. This is to keep latencies for data at the beginning of the burst from growing too large. The offset time between the burst header and the data burst is 1μs. Since the jitter is only caused by the ingress edge router and is independent of other traffic in the network, we set node 2 to be the video initiating node and node 13 be the receiving node.

Simulation Results
We have plotted the jitter performance of the traditional approach and the proposed approach in Fig. 5 and Fig. 6 respectively. From the probability distribution function (pdf) and cumulative distribution function (cdf) plots shown in Fig. 7 and Fig. 8, a noticeable difference in jitter is seen between the fixed and dynamic burst size threshold assembly algorithms. The pdf axis is on the left coordinates with the bar graph along the x-axis (jitter). It is seen what percentage of total jitter happens at specific values of jitter. The line graph coordinates with the y-axis on the right and the values of jitter along the x-axis. From this graph it is easy to see that 60% of jitter is less than 0.03s and 80% of jitter is less than 0.09s for the fixed burst size threshold algorithm. The proposed dynamic burst size threshold algorithm shows great improvement over traditional schemes. As we can see, more than 80% of jitter is close to zero.

5. CONCLUSION

To effectively support telesurgery in OBS networks, we have proposed a content aware burst assembly scheme.
which dynamically adjusts the burst assembly parameters based on the content being assembled. The proposed content aware burst assembly minimizes the jitter within a video frame, as well as across the left-view and right-view frames for 3D vision generation. Simulation results have shown that the proposed scheme can effectively reduce the latency and jitter experienced by video streams, making OBS a promising candidate for supporting telesurgery and telemedicine applications.

REFERENCES