# Exploiting Packet Semantics in Real-time Multimedia Streaming

Sungwoo Hong  Youjip Won
Division of Electronics and Computer Engineering
Hanyang University Seoul, Korea zip: 133-791
{toggiya | yjwon}@ece.hanyang.ac.kr

***Abstract*—In this paper, we propose packet selection and significance based interval allocation algorithm for real-time streaming service. In real-time streaming of inter-frame (and layer) coded video, minimizing packet loss does not imply maximizing QoS. It is true that packet loss adversely affects the QoS but one single packet can have more impact than several other packets. We exploit the fact that the significance of each packet loss is different from the frame type it belongs to and its position within GoP. Using packet dependency and PSNR degradation value imposed on the video from the corresponding packet loss, we find each packet's significance value. Based on the packet significance, the proposed algorithm determines which packets to send and when to send them. The proposed algorithm is tested using publicly available MPEG-4 video traces. Our scheduling algorithm brings significant improvement on user perceivable QoS. We foresee that the proposed algorithm manifests itself in last mile connection of the network where intervals between successive packets from the source and to the destination are well preserved.**

**Keywords**: real-time multimedia streaming, scalable encoding, greedy approach, packet significance, traffic smoothing

## I. Introduction

### A. Motivation and Related Works

Rapid advancement in network and video compression technology have made it possible to enjoy bi-directional interactive multimedia service in ubiquitous fashion. Advancement of network technology has brought us not only the abundance in bandwidth but also more importantly the "variety" of bandwidth choices, e.g. from fast moving speed with low bandwidth of 3.5G or 4G communication technology to Gbyte/sec bandwidth in residential unit (Fiber to the home, FTTH) as shown in Fig. 1.

Real-time video streaming bears unique performance requirement which distinguishes itself from text based best effort data service: bandwidth guarantee and rate variability. Since real-time video streaming requires that each compressed information needs to arrive at destination before its predefined deadline, a certain fraction of bandwidth needs to be guaranteed for that connection either deterministically or stochastically. The size of each video frame can differ by order of magnitude. The variability on video frame size is

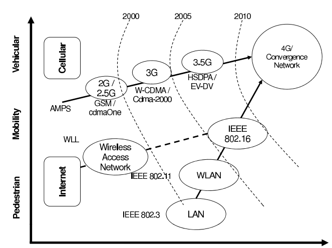



Fig. 1. Evolution of the Network Technology

realized as "bursty" network traffic. This set of efforts is called traffic smoothing (or traffic shaping). To maximize user perceivable QoS, the sender needs to make right choice for two fundamental issues: "what to send?" and "when to send?". The first issue is to select the subset of compressed information to adapt to the available bandwidth. The second issue is about removing the burstiness in the underlying traffic after the selection process.

To decide subset of the compressed for the transmission, priority based packet scheduling algorithm has been the subject of intense research for many years. The key ingredient is to assess the "right" priority to individual packet so that the user perceivable QoS can be maximized. In Politis et al.[7], packet priority is determined based upon the distortion of the displayed video if the packet is lost. If a frame is an anchor frame, e.g. I or P type, it has higher priority than the frames which does not any dependent. Frossard et al.[1] advanced this idea. Frossard et al.[1] suggest to consider the number of dependent frames as well as the total size of dependent frames. However, these works considered only simple GoP structure and burstiness of the traffic was not considered. Numerous efforts have been proposed to reduce the burstiness. M. Hassan et al. suggested layered video streaming algorithm for the QoS by adapting rate variation[3]. D. Jurca et al. suggest packet selection and Scheduling for Multipath Streaming[2]. However, these works do not consider the importance of each frame packet. Algorithm to maximize QoS, burstiness of the traffic, priority based packet scheduling for the available bandwidth should be considered in a single context.

In this work, we propose an elaborate model called packet significance which effectively represents the QoS importance

[0]This work was supported by HY-SDR Research Center at Hanyang University, Seoul, Korea, Under the ITRC program of Ministry of Knowledge Economy, Korea

of a packet and develop a greedy algorithm for packet scheduling based upon the notion of packet significance. The contribution of our work is three folds. First, we develop a notion of packet significance which captures the QoS importance of a packet. Our scheduling framework elaborately harbors the inter-frame dependency as well as inter-layer dependency of a frame. Second, we successfully develop a unified framework for determining "what to send" and "when to send". Traffic smoothing algorithm and priority based packet scheduling have been dealt with in a separate context. To properly exploit the underlying network resource and maximize user perceivable QoS, it is mandatory that these two issues are properly addressed in a single unified framework. Third, our scheduling framework incorporates not only the network's aspect of a packet but also the operating system's aspect of a packet. From network's point of view, bandwidth process is a prime concern. From operating system's point of view, however, packet count process (packets/sec) is more important, since network queue is represented by the array of packet pointers where the size of individual packet does not matter.

The remainder of the paper is organized as follows. Section II introduces the packet significance and traffic smoothing. Then, two fundamental questions is solved in section III. Section IV carries the result of the performance evaluation. We conclude our work in section V.

## II. Packet Significance and Traffic Smoothing

In our context, the notion of "packet scheduling" consists of two ingredients: (*i*) what to transmit and (*ii*) when to transmit. Packet scheduler is required to select a certain fraction of compressed information so that it does not overflow the underlying subnet. In selecting, it is important to properly select the subset of layers so that we can maximize user perceivable QoS. To properly select the subset of layers, it is necessary to gauge the importance of each packet.

### A. Packet Significance

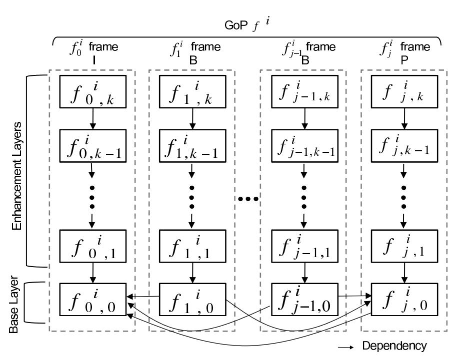



Fig. 2. Dependency of MPEG-4 FGS video

We define the notion of *packet significance* to represent the importance of a frame or layer in a packetThere has been many efforts [1], [7] in this area which tried to represent the importance of packet and scheduled using the packet importance. To decide importance of each packet, they considered

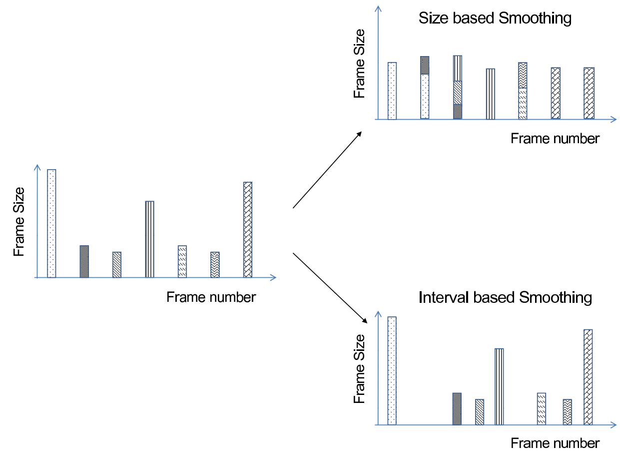



Fig. 3. Size/Interval based Smoothing

size of packet, frame type such as I,P or B frame, number of referencing frames, GoP structure, MSE distortion information in the decoded video and so on. However, these things should be considered in a single framework. In this work, we not only consider the terms mentioned above but also controls the burstiness of the traffic for the higher user perceivable QoS.

Without loss of generality, we assume that video trace file is layer encoded. $f^i_{j,k}$ denotes $k_{th}$ layer information for $j_{th}$ frame of $i_{th}$ GoP. We define a set of *parent* packets and *child* packets for $f^i_{j,k}$. A set of parent packets, $\mathcal{P}(f^i_{j,k})$, denotes a set of packets which are required to decode packet $f^i_{j,k}$. A set of *child* packets of $f^i_{j,k}$ is a set of packets which has $f^i_{j,k}$ as its parent, i.e. $\mathcal{C}(f^i_{j,k}) = \{f^m_{n,l} \mid f^i_{j,k} \in P(f^m_{n,l})\}$. Loss of $f^i_{j,k}$ causes the inappropriate decoding of not only $f^i_{j,k}$ itself but also all packets in its child packet, $\mathcal{C}(f^i_{j,k})$.

Fig. 2 schematically illustrates the dependency among frames and layers. We develop a model to represent quality of an image. An image consists of a set of pixels, which are arranged in a 2 x 2 matrix. The number of pixels in a screen is called resolution, e.g. HD" 1024*768, VGA:640*480 and QCIF:320*240. Each pixel is usually represented by 24bit. Let $f^i_{j,k}(x,y)$ is a pixel value (RGB) at $(x,y)$ position of an image when $f^i_{j,k}$ is properly decoded and $\widehat{f^i_{j,k}}(x,y)$ is a pixel value when $f^i_{j,k}$ is not properly decoded. We define *contribution* $\mathcal{D}(f^i_{j,k})$ as in Eq. 1.

$$10 \log \frac{W \times H \times 255^2}{\sum_{x=0}^{W-1} \sum_{y=0}^{H-1} |\hat{f}^i_{j,k}(x,y) - f^i_{j,k}(x,y)|^2} \qquad (1)$$

where H and W indicate the screen height and width, respectively. $\mathcal{D}(f^i_{j,k})$ gives a quality metric of $f^i_{j,k}$ loss. Significance of a packet $f^i_{j,k}$ is sum of all subsequent PSNR degradation which can occur due to the loss of $f^i_{j,k}$. Significance of $f^i_{j,k}$ is defined as $\mathcal{Q}(f^i_{j,k}) = \sum_{f^l_{n,m} \in \mathcal{C}(f^i_{j,k})} \mathcal{D}(f^l_{n,m})$. It is worth noting that packet significance $\mathcal{Q}(f^i_{j,k})$ effectively captures the information dependency among frames or between layers.

### B. Traffic Smoothing

One of the key issues in packet scheduling is to reduce the burstiness of the network traffic. It is called traffic smoothing. The purpose of deciding when to transmit each packet is to remove the traffic burstiness so that it can make contributions on QoS via minimizing the packet losses. There are two main approaches in realizing traffic smoothing: (*i*) sized based and (*ii*) interval based smoothing. In size based smoothing, the packet scheduler controls the amount of information carried by a single packet so that size of each packet is similar. In interval based smoothing, the interval between the packets is determined based on the size of packet. Larger packet is allocated longer interval. Size based smoothing mandates that single packet can carry more than single frame. MPEG standard does not put any restriction on whether single packet contains more than one frame, however, most of the video streaming system does not allow that because loss of single packet may result in a loss of multiple frames. In addition, when single packet carries more than one frames, decoder needs to locate the boundary of individual frame for decoding. Locating the boundary of each frame can cause severe CPU overhead especially in mobile hand held devices which have a low-end CPU. Fig. 3 illustrates size and interval based smoothing. In this work, we focus on interval based smoothing approach.

There are two different aspects of the underlying network traffic analysis: byte count (byte/sec) and packet count (packets/sec). Most of the existing works on traffic smoothing deal with bandwidth process. In operating systems, kernel maintains fixed length queue of packet pointers for UDP datagram. Packets reside in kernel address space and pointed by these pointers. From network queue's point of view, incoming traffic can become burstier as a result of interval based smoothing which is illustrated in Fig. 3. Subsequently, packet loss can increase due to traffic smoothing process [8]. However, reduction in packet loss does not necessarily imply the improvement on QoS nor improvement on PSNR. On the same token, increase in packet loss does not necessarily imply the QoS degradation. The impact of packet loss over user perceivable QoS varies dependent upon many factors such as frame type of lost packet, its position within GoP, its size and the number of frames it is referring. Packet scheduler needs to determine the transmission timing (or equivalently interval) so that a more important packet becomes less vulnerable to packet loss. Fig. 4 illustrates the result of the physical experiment [8]. While "Lost packet" denotes the ratio between the number of lost packets over total number of packets, "Lost data" is the ratio between the amount of lost data over total amount of data. Packet loss increased when interval based smoothing is applied. However, the total amount of lost data has decreased as a result of smoothing and user perceivable QoS has improved significantly. User perceivable QoS improved significantly not because packet loss decreased but because loss of "important" packets decreased [9]. Interval based traffic smoothing algorithms do not consider the QoS importance of a packet. However, interval based traffic smoothing algorithms successfully distinguish the packets based upon their respective QoS importance. It is found that this phenomenon is due to the inadvertent result of two technical characteristics. The first one is the way video frames are marshalled into packets. As mentioned before, single packet does not contain more than one frame. Since traffic smoothing aims at minimizing rate variability of the bandwidth process, the transmission interval between B type frame packets become smaller as a result of traffic smoothing while the interval between I type packet and its successor (or predecessor) becomes longer. The second technical feature is the way operating system handles queue of packets. When a packet arrives, it is copied into main memory and operating system inserts the packet pointer into the queue of pointers. The way video frame is marshalled and the way operating system handles incoming packets yield very interesting result when they are combined together. Interval based traffic smoothing algorithm controls the interval between outgoing packets to make the data rate smoother; hence, small size packets are more closely populated. From the operating systems's perspective in the receiving end, incoming traffic actually becomes burstier, and gets exposed to more packet loss. Since larger packet has relatively longer interval from the departure of the preceding packet, it is less likely that larger packet finds the queue full. Due to harmonious effort between packetization method and the kernel data structure of packet, traffic smoothing algorithm happens to incorporate packet importance.

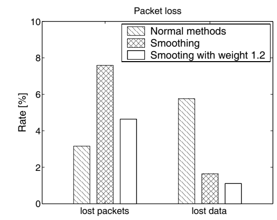



Fig. 4. The Packet Loss and QoS [8]

## III. Significance Aware Packet Scheduling

### A. Packet Scheduling: What to transmit

The question of "what to send" is equivalent to selecting packets among whole video frame. *Packet selection* is a process of determining the subset of packets for transmission satisfying resource constraints. Let $\mathcal{F}(f^i_{j,k})$ be the transmission interval of $f^i_{j,k}$, which is the interval from its immediately preceding packet. Let $\mathcal{S}(f^i_{j,k})$ be the size of $f^i_{j,k}$. Current bandwidth availability information is assumed to be informed to the streaming server or content delivery network (CDN) by the system so that optimal transmission rate can be allocated accordingly [1]. $f^i$ is a set of packets in $i_{th}$ GoP. We define total QoS of selected packets in $f^i$ as in Eq. 2.

$$\xi(f^i) = \underbrace{\sum_{\mathcal{F}(f^i_{j,k})<\infty} \mathcal{D}(f^i_{j,k})}_{A} - \underbrace{\sum_{f^i_{j,k}\ is\ lost} \mathcal{Q}(f^i_{j,k})}_{B} \qquad (2)$$

Condition $\mathcal{F}(f^i_{j,k}) < \infty$ denotes the set of *"selected"* packets. Term A corresponds to PSNR values resulting from transmitting selected packets. The term B denotes QoS degradation caused by a packet loss. Our objective is to maximize $\xi(f^i)$ via properly selecting subset of packets and via properly determining transmission schedule. Our process consists of two phases: packet selection and packet transmission. In packet selection phase, we choose the subset of packets not to exceed a given bandwidth envelope. The packet selection problem is equivalent to knapsack problem where the size and significance of $f^i_{j,k}$ corresponds to the weight and value of an item in knapsack problem, respectively. The capacity constraint of a knapsack problem is determined by the bandwidth envelope as $\mathcal{U} = \int_{t_0}^{t_0+\omega} \rho(t)dt$, where $t_0$, $\omega$ and $\rho(t)$ denote start time of the window, one GoP time length and the available bandwidth at $t$, respectively. To solve the knapsack problem, we take the greedy approach because each choice should be made within time constraint such as delay and bandwidth constraints for the higher QoS.

The selection criteria, $\epsilon(f^i_{j,k})$ is the ratio between QoS significance and its size, i.e. $\epsilon(f^i_{j,k}) = \mathcal{Q}(f^i_{j,k})/\mathcal{S}(f^i_{j,k})$. The algorithm sorts all packets in each GoP with respect to the decreasing order of $\epsilon(f^i_{j,k})$ and selects one by one until the sum of the selected information exceeds the capacity constraint $\mathcal{U}$. It is worth noting that through the bandwidth adaptation and packet selection process, SAPS scheme does not require higher bandwidth but tries to make higher QoS within bandwidth constraint. Hence, it won't impact other packets of different media or other application in terms of bandwidth consumption.

### B. Packet Scheduling: When to transmit

Once we determine the set of packets to transmit, we need to determine packet transmission schedule of selected packets. Here, determining a transmission schedule is equivalent to determining an interval between packet departure to avoid packet loss. We incorporate the packet significance in determining its interval. The key idea is to assign larger interval to more important packet. Let $\delta(f^i_{j,k})$ denote the time interval between $f^i_{j,k}$ and its immediate predecessor and it is defined as in Eq. 3.

$$\delta(f^i_{j,k}) = \frac{\omega \times \mathcal{S}(f^i_{j,k}) \times \mathcal{Q}(f^i_{j,k})}{\sum_{F(f^i_{j,k}) \in f^{(i)}} \mathcal{S}(f^i_{j,k}) \times \mathcal{Q}(f^i_{j,k})} \tag{3}$$

The size of $f^i_{j,k}$ can be greater than maximum transfer unit size and it can span multiple packets. When $f^i_{j,k}$ consists of multiple packets, we evenly distribute these packets on allocated interval. The interval among the packets in $f^i_{j,k}$ is computed as $\delta(f^i_{j,k})/\left\lceil \frac{\mathcal{S}(f^i_{j,k})}{MTU} \right\rceil$.

## IV. Performance Evaluation

### A. Experiment Setup

We examine the effectiveness of Semantics-Aware Packet Scheduling algorithm (SAPS). We compare SAPS with two other packet scheduling algorithms. The first one is size based packet scheduling, which we refer as "Size" for simplicity's sake, where transmission interval is linearly proportional to the size of a frame. The second one, which we refer as "Bit-rate", transmits packets based on the predefined bit-rate by the system after the packet selection process. "Size" and "Bit-rate" does not consider the semantics of a packet in determining the transmission schedule. We simulate in NS-2 [4] over the network topology depicted in Fig. 5. We use three publicly available and widely used video clips for the experiment [6]. Three video clips are compressed by MPEG-4 encoder [5]. All compressed video clips have 300 kbits/sec and 30 frame/sec, with 176*144. GoP structure of compressed video is $I(BBP)^{10}$ with size of 30.

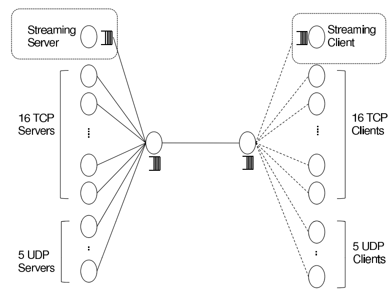
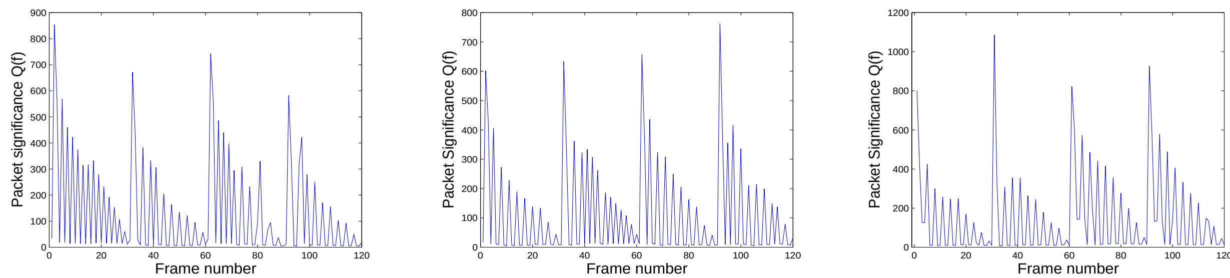



Fig. 5. Topology of the experiment setup

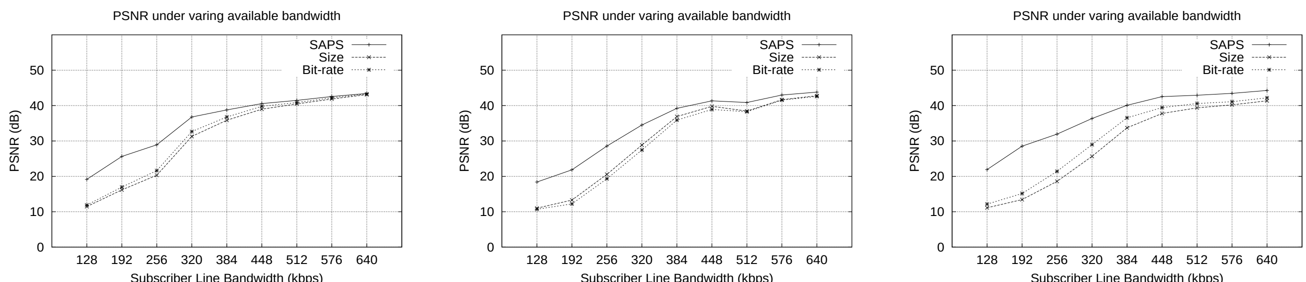
There exist 16 TCP and 5 UDP node pairs sharing the link. File transfer protocol (FTP) application is running over TCP. The maximum bandwidth from each TCP and UDP node pair corresponds to 1Mbyte/sec and 128 Kbyte/sec, respectively. Client starts displaying video 2 seconds after the transmission has begun. If packets arrive out of order sequence, then the respective packet stays at the queue until all of the required packets arrive before the play-out deadline or discarded. In addition, in case that at least single packet is dropped or corrupted during the transmission, and is not able to recover with recovery scheme such as FEC, then the entire packets consisting one frame will be discarded. If a frame is lost during the transmission or arrived later than the play-out deadline, the previous frame concealment scheme is used at the decoder. The average time taken for the calculation of each packet significance is approximately 0.5 seconds. For example, the total time taken to calculate 30 frames' packet significance in 30 FPS movie is $\frac{\frac{FPS}{FPS}+\frac{FPS-1}{FPS}+\cdots\frac{1}{FPS}}{2} = \frac{31}{2}$. Hence the average time taken per frame is $\frac{31}{2*30} \approx 0.5$. We assume that this information has been calculated at encoding time and transmitted to the streaming server or CDN with movie file. Packet significance can be computed off-line and therefore does not interfere with the real-time video streaming session.

Fig. 6 illustrates the packet significance distribution. As can be seen, packet significance varies subject to its frame type and the position within GoP. I frame, at every $30_{th}$, shows very high significance value. P frames, immediately after I frame, are likely to have higher value of packet significance than those frames far from the I frame whether the frame size is big or not. B frames which are located between I (or P) and next I(or P) frame show very low significance value.

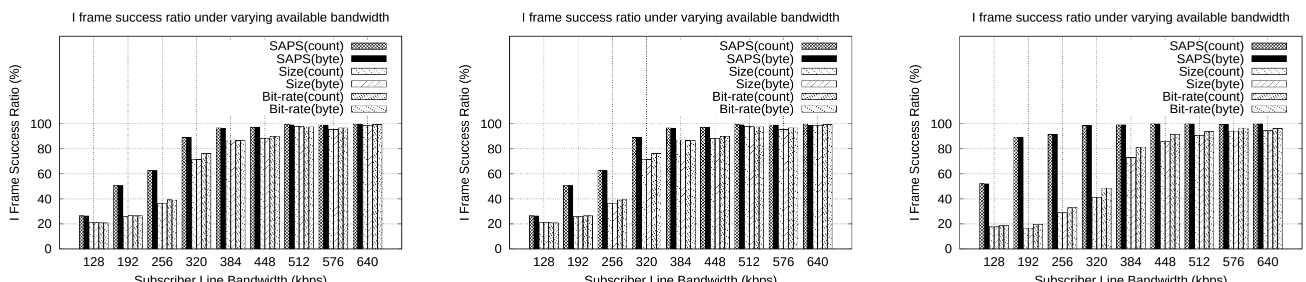



(a) Packet Significance (Mother and daughter) (b) Packet Significance (Salesman) (c) Packet Significance (Miss Am.)

Fig. 6. Packet Significance Value Distribution




(a) PSNR vs. Bandwidth (Mother and daughter) (b) PSNR vs. Bandwidth (Salesman) (c) PSNR vs. Bandwidth (Miss Am.)




(d) PSNR vs. Bandwidth (Mother and daughter) (e) PSNR vs. Bandwidth (Salesman) (f) PSNR vs. Bandwidth (Miss Am.)

Fig. 7. Performance under Varying Client Bandwidth

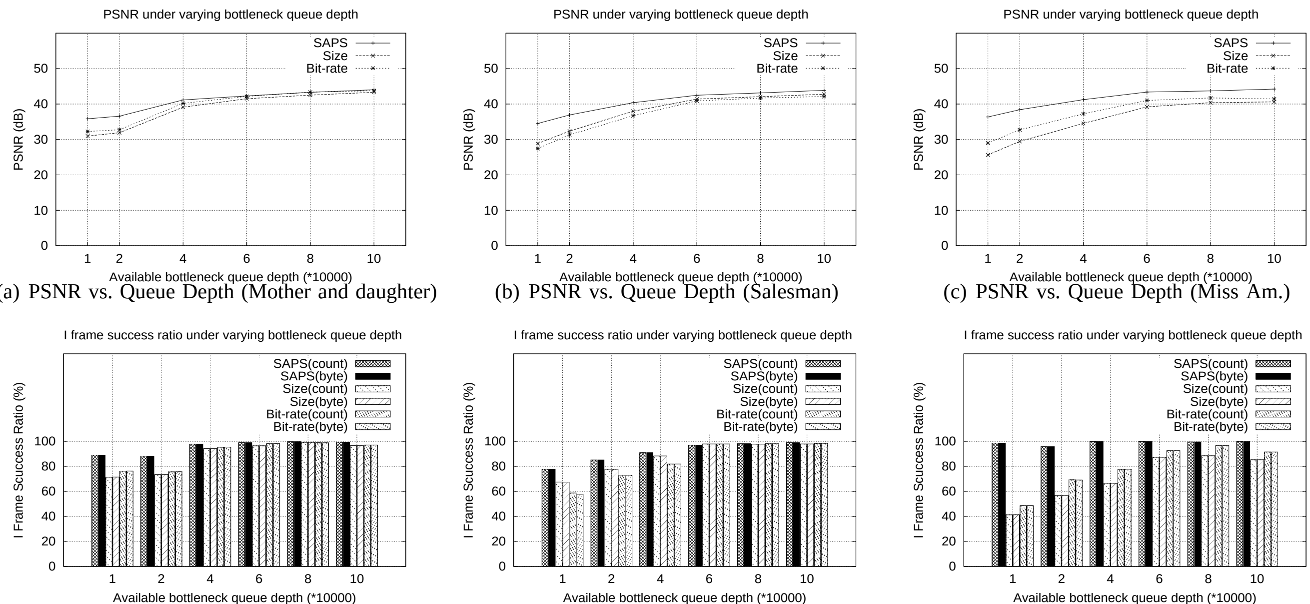
## *B. Effect of Network Bandwidth Availability*

We vary each subscriber line bandwidth at client from 128 kbps to 640 kbps with fixed bottleneck queue depth of 10000 and examine the performance of each scheme. Fig. 7(a), 7(b) and 7(c) illustrate the PSNR of SAPS, Size and Bit-rate scheme as a function of the available scriber line bandwidth at the client. In all figures, PSNR increases with the capacity of bandwidth availability. When scriber line bandwidth link capacity reaches 640 kbps, SAPS, Size and Bit-rate scheme achieve almost same PSNR. However, when link capacity becomes smaller, SAPS manifests its capability of handling significance. For example, when the available bottleneck bandwidth is 256 kbps, the PSNR gain of SAPS over Size and Bit-rate scheme is about ($8_{dB}$, $7_{dB}$), ($8_{dB}$, $9_{dB}$) and ($13_{dB}$, $10_{dB}$) for Mother and daughter, Salesman and Miss Am., respectively.

Fig. 7(d), 7(e) and 7(f) illustrate the packet and byte success rate of I frame with three schemes under different subscriber line bandwidth at client. In all experiment, the total packet and byte success rate was almost the same. However, the success ratio among I, P and B frame was different. For SAPS, it makes high packet significance value of packet such as I frame less vulnerable. In other words, it exposes less important packets such as B frame to a more vulnerable state. Hence, although the total success rate is almost same, the success ratio of each frame is quite different. In all figures, packet and byte success rate of I frame increases as higher bandwidth is allocated. However, SAPS scheme shows higher packet and byte success rate of I frame all the time and which correspondingly explains higher PSNR value under the same available subscriber line bandwidth.

## *C. Effect of Bottleneck Queue Depth*

In this section, we compare three algorithms under varying bottleneck queue depth with fixed subscriber line bandwidth of 320 kbps at client. Fig. 8(a), 8(b) and 8(c) illustrate the performance of SAPS, Size and Bit-rate scheme under different bottleneck queue depth. In all algorithms, PSNR increases




(a) PSNR vs. Queue Depth (Mother and daughter) (b) PSNR vs. Queue Depth (Salesman) (c) PSNR vs. Queue Depth (Miss Am.)

(d) PSNR vs. Packet Loss (Mother and daughter) (e) PSNR vs. Packet Loss (Salesman) (f) PSNR vs. Packet Loss (Miss Am.)

Fig. 8. Performance under Varying Client Queue Depth

with the increase in bottleneck queue depth. When bottleneck queue depth is small, we observe significant difference in PSNR values among SAPS, Size and Bit-rate algorithms. For example, when 20000 is allocated for the bottleneck queue of Salesman video trace, PSNR values are around $32_{dB}$ and $31_{dB}$ in Size and Bit-ratio algorithm, respectively. Under the same bottleneck queue depth, PSNR value is $36_{dB}$ in SAPS algorithm. PSNR values show $4_{dB}$ to $5_{dB}$ higher when we apply SAPS algorithm in scheduling packets. When bottleneck queue is sufficiently large, there is less packet loss due to queue overflow, so difference of PSNR values of the three algorithms becomes small.

In all cases of performance evaluation under different bottleneck queue depth of three schemes, the total packet and byte success rate show no big difference. However, each success ratio for different frames shows big difference in their value for three different algorithms. Fig. 8(d), 8(e) and 8(f) illustrate packet and byte success rate of I frame with three schemes under varying bottleneck queue depth. As shown in the figures, packet and byte success rate increases with increase in bottleneck queue depth. In all cases, SAPS shows higher packet and byte success rate. This is due to the fact that SAPS successfully adapts to bottleneck queue availability so that more important packets become less vulnerable to packet loss with queue overflow.

## V. Conclusion

In this work, we develop a novel packet scheduling framework which properly incorporates the significance of a packet for the real-time streaming. We analyze the data structure of the network packets in operating system and make packet scheduling frameworks effectively exploit the data structure. To achieve this objective, we develop elaborate metric to represent the importance of a packet from user QoS's point of view: Packet Significance. We develop video streaming framework, Significance Aware Packet Scheduling (SAPS), which consists of packet selection and packet scheduling phases taking account of packet significance. Through simulation based experiment, we find that via properly incorporating the packet significance, we can increase PSNR value more than $4_{dB}$ especially when the resource, such as bottleneck queue depth or available bandwidth, is limited. SAPS manifests itself under resource stringent environment, e.g. real-time video streaming in mobile wireless network.

## References

[1] P. Chou and Z. Miao. Rate-distortion optimized streaming of packetized media. *Multimedia, IEEE Transactions*, 8(2):390–404, Apr. 2006.

[2] D D. Jurca and P. Frossard. Video packet selection and scheduling for multipath streaming. *Multimedia, IEEE Transactions on*, 9(3):629–641, 2007.

[3] M. Hassan and M. Krunz. Video streaming over wireless packet networks: An occupancy-based rate adaptation perspective. *Circuits and Systems for Video Technology, IEEE Transactions on*, 17(9):1017–1027, Aug. 2007.

[4] Information Sciences Institute. The Network Simulator - ns-2. [Online]. Available: http://nsnam.isi.edu/nsnam/index.php/User_Information

[5] MEncoder. Program for encoding video+audio. [Online]. Available: http://www.mplayerhq.hu/

[6] D. Park and S. Hong. Video trace file. [online]. Available: http://www.dmclab.hanyang.ac.kr/data/mpeg2data/Video_traces.htm.

[7] I. Politis, M. Tsagkaropoulos, T. Pliakas, and T. Dagiuklas. Distortion optimized packet scheduling and prioritization of multiple video streams over 802.11e networks. *Advances in Multimedia*, 2007-2(6), April 2007.

[8] S. Song, Y. Won, and I. Song. Empirical study of user perception behavior for mobile streaming. In *Proceedings of the tenth ACM international conference on Multimedia*, pages 327–330, Juan-les-Pins, France, 2002. ACM New York, NY, USA.

[9] Y. Won and B. Shim. Empirical study of VBR traffic smoothing in wireless environment. In *Proceedings of the second international workshop on Innovative Internet Computing Systems*, volume 2346(193–204). Springer, June 2002.