Reducing Channel Capacity for Scalable Video Coding in a Distributed Network

Hyunpil Kim, Sukhan Lee, Junghee Lee, and Yongsurk Lee

In recent years, the development of multimedia devices has meant that a wider multimedia streaming service can be supported, and there are now many ways in which TV channels can communicate with different terminals. Generally, scalable video streaming is known to provide more efficient channel capacity than simulcast video streaming. Simulcast video streaming requires a large network bandwidth for all resolutions, but scalable video streaming needs only one flow for all resolutions. In previous research, scalable video streaming has been compared with simulcast video streaming for network channel capacity, in two user simulation environments. The simulation results show that the channel capacity of SVC is 16% to 20% smaller than AVC, but scalable video streaming is not efficient because of the limit of the present network framework. In this paper, we propose a new network framework with an SVC extractor. The proposed network framework shows a channel capacity 50% (maximum) lower than that found in previous research studies.

Keywords: Distribution networks, SVC, IPTV, simulcast, video streaming.

I. Introduction

The ubiquitous network infrastructure is going through an evolution to provide all the digital communication services through IP-core networks. In this environment, one of the important services is video streaming service. Traditionally, video contents have been developed with the TV broadcasting system, but the development axis has shifted to video on demand services and wireless display communication systems. On the ubiquitous network infrastructure, there are many kinds of physical layers. The style of the terminals includes, among others, a great variety of bandwidth, the transfer characteristic, and quality of service guarantee, and the type of physical layer environment involves a wide variety of size, performance, and resolution of the terminal. Thus, having different speeds on the access networks is making the process complex. In this environment, video contents have to be adaptively provided for the various physical layers and terminal environments [1].

On ubiquitous network infrastructures, whether wired or wireless communications, the broadcasting service is provided using the same high-level application layer, but various types of low level physical infrastructures such as fiber, cable, LAN, WLAN, WiMAX, WiBRO, and HSDPA are used. Even though the characteristics and the bandwidth of each infrastructure are different, the communication between heterogeneous terminals has to be kept. Consequently, not only the type of the infrastructure but also the terminals, including TVs, set-top boxes, desktop and notebook computers, PDAs, and cell phones using various physical infrastructures, have to support compatibility among the heterogeneous terminal environments.

Terminal environments include the performance of the terminal, the transfer rate, and supporting resolution. As a result,

---

Manuscript received Jan. 14, 2010; revised Sept. 1, 2010; accepted Sept. 14, 2010

This work was supported by the OmniFlow Processor Research Team of ETRI from the Research Program of development of Scalable Micro Flow Processing Technology.

Hyunpil Kim (phone: +82 2 2123 2872, email: hpkim@mpu.yonsei.ac.kr) and Yongsurk Lee (email: yonglee@yonsei.ac.kr) are with the School of Electrical & Electronic Engineering, Yonsei University, Seoul, Rep. of Korea.

Sukhan Lee (email: sh1026.lee@samsung.com) is with IMS Security Solution Team, Samsung Techwin, Seongnam, Rep. of Korea.

Junghee Lee (email: hilee@etri.re.kr) is with the Internet Research Laboratory, ETRI, Daejeon, Rep. of Korea.

doi:10.4218/etrij.10.0110.0033

it is difficult to provide a fixed bandwidth between end-to-end nodes on this kind of IP-based infrastructure, and the transfer characteristic is changed dramatically by the mixed heterogeneous infrastructure. Hence, a stable and seamless network is essential in providing the broadcasting service, and this requires the technology of end-to-end quality guarantee. In particular, a broadcasting service capable of accepting a change of bandwidth while providing the service is needed.

Scalable video coding (SVC) is one solution to satisfy these technical requirements because SVC supports temporal, spatial, and resolution changes caused by the removal of the part of the bitstream. There are several kinds of scalable video encoding standards: MPEG-2 [2], H.263 [3], and MPEG-4 Part 2: Visual [4], [5]. However, these have a lower encoding efficiency and do not support various kinds of scalability. Moreover, a communication network has a fixed throughput, and the packet is only divided into success or failure. Also, the service observing the communication network status with adjusting bitrate is not required, and users watch a TV or PC at a fixed point. These are the reasons why the standards mentioned above did not commercially succeed.

To address the disadvantages of the previous codec, a Joint Video Team (JVT) consisting of ISO/IEC Moving Picture Experts Group (MPEG) and ITU-T Video Coding Experts Group (VCEG) is standardizing SVC [6]-[10] as an amendment of H.264/MPEG-4 AVC [9]. We propose a new network framework with a new router using an extraction decision engine (EDE) and SVC extractor to reduce the required SVC data bandwidth. In addition, we compare the SVC environment in the proposed framework with the previous research on the related subject. The paper is organized as follows. In section II, we present the theory of the assumed models and the probability formula for the current and proposed network framework. In section III, the simulation we use is presented. We analyze the simulation results in section IV. In section V, we discuss a possible future project, and we conclude in section VI.

II. Theory

1. Assumption of the Models

This paper assumes the same IPTV network environment as [11], which has N heterogeneous multimedia terminals, and each user is provided with K channels that have a priority by popularity. Thus, to conclude the priority of the channel, Zipf distribution is used with parameter α ($0<\alpha<1$).

The required bitrate while the simulcast (AVC) is being used is referred to as $R_{avc,l}$. We assume four watchable resolutions (QCIF, CIF, SDTV, and HDTV), and the required bitrates are 0.128 Mbps, 0.384 Mbps, 1.5 Mbps, and 6 Mbps, respectively. The required bitrate with SVC transmission is $R_{svc,l}$. From [12], [13], to have the same video quality, SVC encoded video has a coding penalty, ε. Consequently, SVC video contents require a higher bitrate than AVC encoded video as

$$\begin{aligned} R_{\text{svc},1} &= R_{\text{avc},1}, \\ R_{\text{svc},l} &= R_{\text{avc},1} + \sum_{i=2}^{l} (R_{\text{avc},i} - R_{\text{avc},i-1})(1+\varepsilon). \end{aligned} \tag{1}$$

In (1), ε means the code complexity for SVC to have the same video quality as AVC. According to [14], ε could be increased as much as 10%, but in reality, ε is regarded as 20% to 30%.

2. Capacity Demand

Each user is assumed to be a random variable $n_{k,l}$ (range) with K channels and L resolutions, which means a user can watch k channel with l resolution. When AVC encoded video data is transferred with simulcast, the network demand channel capacity is shown as

$$C_{\text{avc}} = \sum_{k=1}^{K} \sum_{l=1}^{L} R_{\text{avc},l} 1_{(n_{k,l}>0)}. \tag{2}$$

However, when SVC encoded data is transferred, there is no reason to transfer the data of the lower resolution than asked resolution l. As a result, the channel capacity required is calculated as

$$C_{\text{avc}} = \sum_{k=1}^{K} \sum_{l=1}^{L} R_{\text{svc},l} 1_{(n_{k,L}=0,\ n_{k,L-1}=0,\ \dots,\ n_{k,l+1}=0,\ n_{k,l}>0)}. \tag{3}$$

3. User Behavior Model

Two user models are defined by [11]. In user model I, every user watches video contents with the terminal supporting only one specific resolution. However, in user model II, a user can randomly select a specific resolution using the multi-resolution support terminal. In the two user models above, every TV channel's popularity amongst users can be modeled by the Zipf distribution function. Each channel has π_k independent probability to be selected by a user, and k-th popular channel has probability defined as

$$\pi_k = dk^{-\alpha}, \qquad \text{for} \quad k = 1, 2, 3, \dots, K. \tag{4}$$

In (4), α is the Zipf distribution parameter and d is the normalization factor to make the sum of the probability into 1.

Thus, on the two user behavior models, the probability generating function of the random variables (users) $n_{1,1},\dots,n_{K,L}$

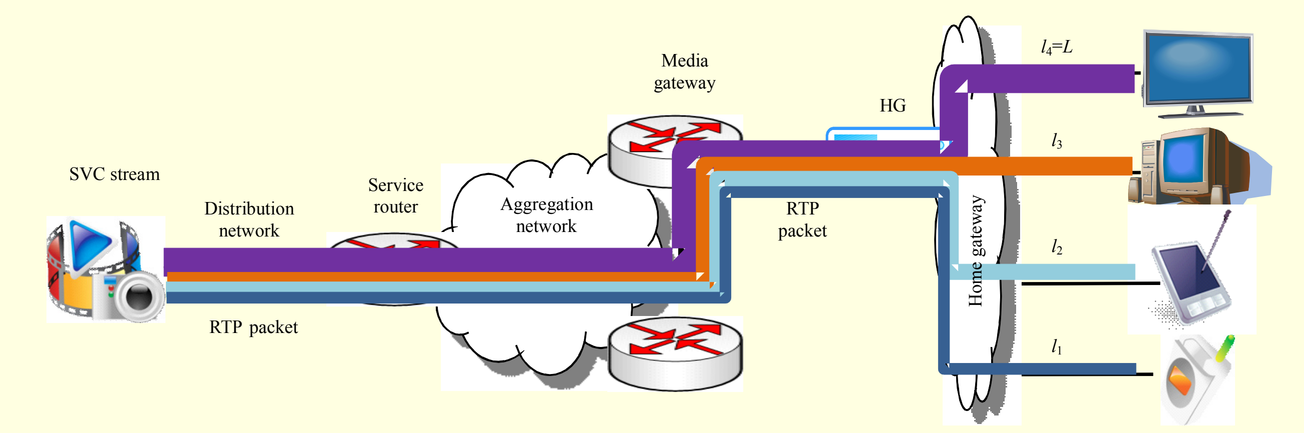


Fig. 1. Multirate multicast-capable IPTV distribution network.

watching l resolution and k channel equals

$$F(z_{k,l};\forall k,l)=E[\prod_{k=1}^{K}\prod_{l=1}^{L}z_{k,l}^{n_{k,l}}]. \tag{5}$$

4. Current Network Frame [15]

At first, in user model I, each group of user N has the terminal that supports one specific resolution l among L fixed groups. Consequently, the user who is the element of the group l can receive l resolution video data, and the sum of N_l is the same as N. In this case, the number of cases is $\prod_{l=1}^{L}\binom{K+N_l}{K}$ when users who can see the different resolution watch K channels independently. This number is substituted into (5) to imply the probability generating function as

$$F(z_{k,l};\forall k,l)=\prod_{l=1}^{L}[1-a_l+a_l\sum_{k=1}^{K}\pi_k z_{k,l}]^{N_l}. \tag{6}$$

Activity grade a_l means the probability that the user is active, and users of the each group l are assumed to have the same activity grade.

Second, user model II is different from user model I in that all users can randomly select the resolution l. Consequently, users are not divided into resolution groups, but the probability b_l to select the resolution is added. Because the different user watches resolution l and channel k, the number of cases is $\binom{N+KL}{KL}$. This number is also substituted into (6) to have the probability generating function as

$$F(z_{k,l};\forall k,l)=[1-a_l\sum_{k=1}^{K}\pi_k\sum_{l=1}^{L}b_l z_{k,l}]^{N}. \tag{7}$$

Figure 1 shows the current network framework. In Fig. 1, when many users who use a different resolution demand transmission to an SVC server, the SVC server will stream the SVC data to users. In this case, a media gateway will deliver all the requested SVC data to the user because the media gateway cannot process layer 7 protocol. However, we can intuitively find that there is no need to transfer SVC data of l_1, l_2, and l_3 because SVC data of L (=l_4) resolution contains SVC data of l_1, l_2, and l_3. In other words, if the media gateway can extract the SVC data of l_1, l_2, and l_3 from SVC data of L (l_4), the media gateway will have enough energy to spare as much as transmitting SVC data of l_1, l_2, and l_3.

5. Proposed Network Framework

As shown in Fig. 1, the current network framework is not efficiently able to apply the scalability of scalable encoded data. For example, we assume a situation where a user wants to watch TV sequentially; resolution l_1, l_2, l_3, and l_4. Although l_1, l_2, and l_3 do not need to be transferred, the current network framework sends all redundant data before l_4 data is requested.

In this paper, we propose a flow-based network framework as shown in Fig. 2. The media gateway has an EDE and SVC extractor which control SVC data. Therefore, when a channel is requested, a node always transfers the maximum resolution L data. If the requested channel has a lower resolution l than L, data for resolution l is extracted by the SVC extractor of the router. Consequently, as shown in Fig. 2, data from l_1, l_2, and l_3 does not need to be transferred in the aggregation network. As a result, the channel capacity can be reduced. The probability of the proposed network framework user model I in which every user watches video contents with the terminal supports only one specific resolution for watching a channel is calculated as

$$\mathrm{Pr}_{\mathrm{prop}_1}=\mathrm{Pr}_{\mathrm{user}_1}\times(1-\mathrm{Pr}[n_{k,1}=0,\ldots,n_{k,L}=0]). \tag{8}$$

On the other hand, in user model II, the watching probability

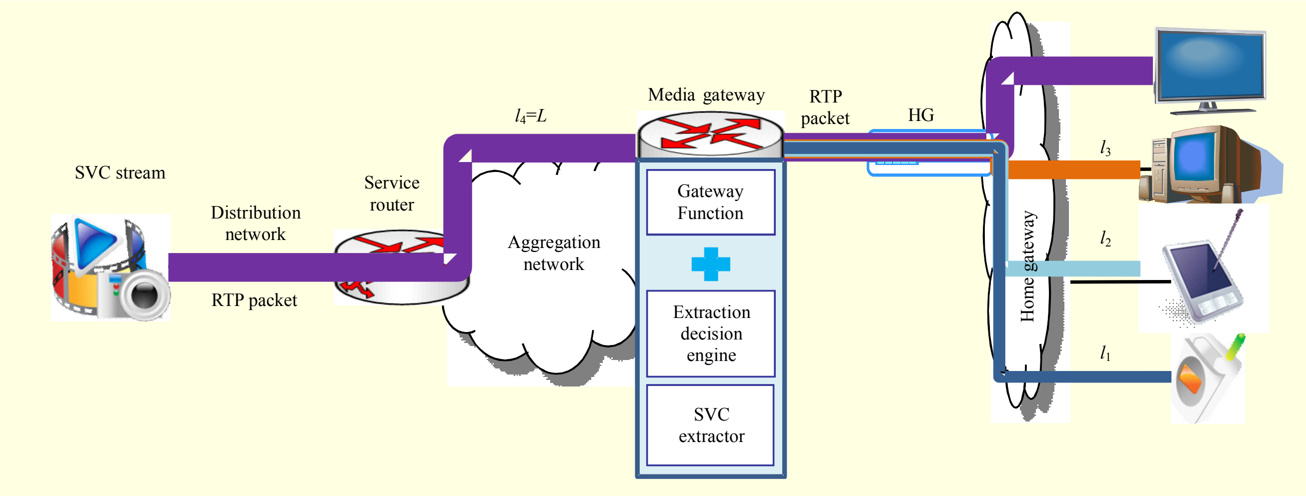


Fig. 2. Proposed distribution network framework.

for each resolution l is defined independently with probability b, so the probability is calculated as

$$\Pr_{\text{prop}_2} = \Pr_{\text{user}_2} \times \Big(1 - \sum_{l=1}^{L} b_l \Pr[n_{k,1} = 0, \ldots, n_{k,L} = 0]\Big). \quad (9)$$

III. Simulation

There is no difference between an exact solution and a Gaussian approximation for calculation of channel capacity by [11]. Thus, in this paper, we also use Gaussian approximation to simulate. We calculate the mean value $E[X]$ and standard deviation value ε with the given probability generating function. Consequently, through a complementary cumulative probability distribution function which refers to the tail distribution function, the error probability for each channel capacity is calculated. After that, the simulator picks the channel capacity which satisfies the minimum error rate and draws to a graph under AVC, SVC on the current network framework, and SVC on the proposed network framework for two user models.

For this process, we apply Gaussian approximation to (2) and (3). In an AVC environment, the mean value and the squared mean value to calculate the standard deviation are calculated as

$$E[C_{\text{avc}}] = \sum_{k=1}^{K} \sum_{l=1}^{L} R_{\text{avc},l} \Pr[n_{k,l} > 0], \quad (10)$$

$$\begin{aligned} E[C_{\text{avc}}^2] = {} & \sum_{k=1}^{K} \sum_{l=1}^{L} R_{\text{avc},l}^2 \Pr[n_{k,l} > 0] \\ & + \sum_{k_1=1}^{K} \sum_{k_2=1}^{K} \sum_{l_1=1}^{L} \sum_{l_2=1}^{L} 1_{(k_1,l_1) \neq (k_2,l_2)} R_{\text{avc},l_1} R_{\text{avc},l_2} \\ & \cdot \Pr[n_{k_1,l_1} > 0, n_{k_2,l_2} > 0]. \end{aligned} \quad (11)$$

In addition, in an SVC environment,

$$E[C_{\text{svc}}] = \sum_{k=1}^{K} \sum_{l=1}^{L} R_{\text{svc},l} A_{k,l}, \quad (12)$$

$$\begin{aligned} E[C_{\text{svc}}^2] = {} & \sum_{k=1}^{K} \sum_{l=1}^{L} R_{\text{svc},l}^2 A_{k,l} \\ & + \sum_{k_1=1}^{K} \sum_{k_2=1}^{K} \sum_{l_1=1}^{L} \sum_{l_2=1}^{L} 1_{(k_1,l_1) \neq (k_2,l_2)} R_{\text{svc},l_1} R_{\text{svc},l_2} B_{k_1,k_2,l_1,l_2}, \end{aligned} \quad (13)$$

where $A_{k,l}$ is calculated as

$$\begin{aligned} A_{k,l} & = \Pr[n_{k,L} = 0, n_{k,L-1} = 0, \ldots, n_{k,l+1} = 0, n_{k,l} > 0] \\ & = \Pr[n_{k,L} = 0, n_{k,L-1} = 0, \ldots, n_{k,l+1} = 0] \\ & \quad - \Pr[n_{k,L} = 0, n_{k,L-1} = 0, \ldots, n_{k,l+1} = 0, n_{k,l} = 0], \end{aligned} \quad (14)$$

and $B_{k,l}$ is calculated as

$$\begin{aligned} B_{k_1,k_2,l_1,l_2} & = \Pr[n_{k_1,L} = 0, \ldots, n_{k_1,l_1+1} = 0, n_{k_1,l_1} > 0, \\ & \qquad n_{k_2,L} = 0, \ldots, n_{k_2,l_2+1} = 0, n_{k_2,l_2} > 0] \\ & = \Pr[n_{k_1,L} = 0, \ldots, n_{k_1,l_1+1} = 0, \\ & \qquad n_{k_2,L} = 0, \ldots, n_{k_2,l_2+1} = 0] \\ & \quad - \Pr[n_{k_1,L} = 0, \ldots, n_{k_1,l_1+1} = 0, n_{k_1,l_1} = 0, \\ & \qquad n_{k_2,L} = 0, \ldots, n_{k_2,l_2+1} = 0] \\ & \quad - \Pr[n_{k_1,L} = 0, \ldots, n_{k_1,l_1+1} = 0, \\ & \qquad n_{k_2,L} = 0, \ldots, n_{k_2,l_2+1} = 0, n_{k_2,l_2} = 0] \\ & \quad + \Pr[n_{k_1,L} = 0, \ldots, n_{k_1,l_1+1} = 0, n_{k_1,l_1} = 0, \\ & \qquad n_{k_2,L} = 0, \ldots, n_{k_2,l_2+1} = 0, n_{k_2,l_2} = 0]. \end{aligned} \quad (15)$$

IV. Analysis

In this section, the influence of the different parameters on the required capacity for both the AVC and SVC streaming mode is explored through case studies. We consider the required downstream capacity in the aggregation and distribution part of an IPTV network. There are four types of subscribers (that is, L=4): mobile subscribers that receive low resolution video (QCIF) because of technology limitations (small dimensions of the devices and power consumption limitations), computer users requesting streamed video in CIF format, and TV set-top box receivers, some requesting SDTV format, and others requesting HDTV format. In all examples to follow, we consider L=4, $R_{\text{avc},l}$={0.128, 0.384, 1.5, 6}Mbps, and a_1=a_2=a_3=a_4=0.8 in user model I. In user model II, α=0.8 and b_l={0.1, 0.3, 0.5, 0.1}. For user behavior in model I, we evenly distribute the total number of users N over the four different resolutions, that is, $N_l = N/4$, for $1 \leq l \leq 4$. We use MATLAB for modeling with Gaussian approximation.

1. Impact of the SVC Coding Penalty

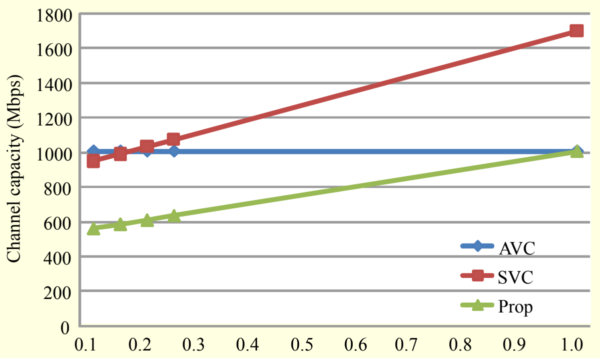
The SVC coding penalty ε determines the range where SVC is more beneficial than AVC with respect to the capacity demand. There exists a limit ε beyond which applying SVC is not beneficial anymore to the saving capacity. However, on the contrary, more capacity is required in the SVC delivery scenario than in the AVC one.

Figure 3 shows the graph of the required channel capacity with the minimum broadcasting error probability; P_{unav}=10^{-4}, the number of channel; K=300, the activity grade; α=0.6, the total number of users; N=1000 for user model II, N_1=N_2=N_3=N_4=250 for user model I, and the channel preference; b_1={0.1, 0.3, 0.5, 0.1}. Figure 3(a) shows the required channel capacity of user model I, and graph (b) is for user model II.

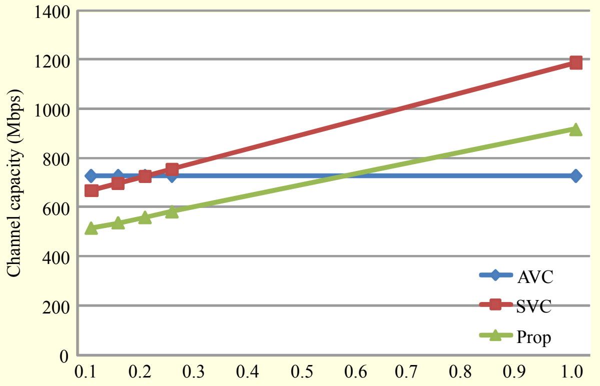
According to [11], each calculated limit point of ε that the channel capacity of SVC and AVC cross together is 0.1667 and 0.2008 for user model I and II, respectively. However, under the proposed network framework, each limit point of ε is 1 and 0.6 for user model I and II, respectively. Consequently, even though a coding penalty can be more than 60%, the channel capacity of the router is similar to the current network framework.

2. Growing Bouquets of TV Channels – Impact of K

In this section, AVC and SVC data transmission on the current network and the proposed network are compared in two models, and the change of the channel capacity is shown in Fig. 4. At the moment after K is 700, the channel capacity of SVC is over the channel capacity of the AVC in user model I, and the same case occurrs when K is 1,000 in user model II.



(a) Coding penalty ε (%) for user model I



(b) Coding penalty ε (%) for user model II

Fig. 3. Determining the limit ε beyond which is there is no capacity gain at P_{unav}=10^{-4} when applying SVC streaming mode instead of simulcast. The scenario parameters are (a) L=4, K=300, α=0.6, N_1=N_2=N_3=N_4=250, a_1=a_2=a_3=a_4=0.8, $R_{\text{avc},l}$={0.128, 0.384, 1.5, 6} Mbps and (b) L=4, K=300, α=0.6, N=1,000, a=0.8, b_l={0.1, 0.3, 0.5, 0.1}, $R_{\text{avc},l}$={0.128, 0.384, 1.5, 6} Mbps.

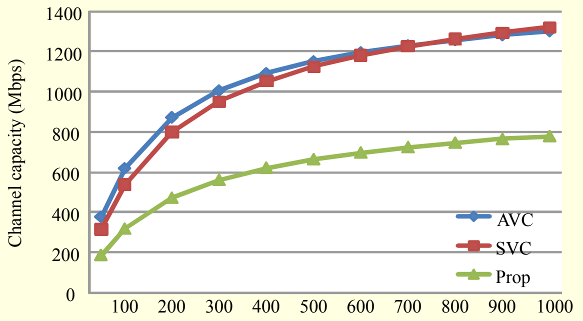
This case can be described with (10) and (12). Equation (10) is for the probability of the AVC transmission, and (12) is for SVC. A_k and B_k stand for (14) and (15), respectively. From these two equations, by increasing the number of TV channels, K, the probability for watching a channel looks the same. That is,

$$\Pr[n_{k,L} > 0] \cong \Pr[n_{k,L} = 0, \ldots, n_{k,l+1} = 0, n_{k,l} > 0].$$

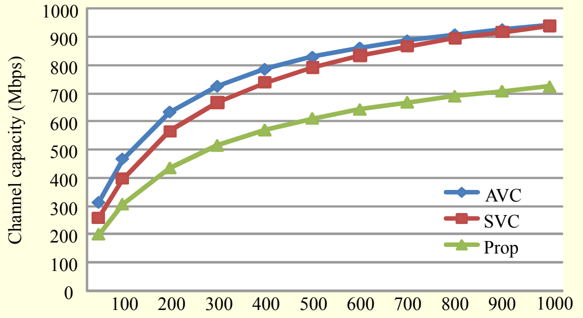
However, because the required bitrate of SVC transmission is about 10% to 30% higher than AVC, the required channel capacity SVC is higher than AVC according to the increase of the provided number of channels, K.

In the proposed network framework, the channel capacity for SVC transmission is dramatically lower than the conventional network framework. After transmitting the maximum resolution, a lower resolution will be extracted on the router. Therefore, the required channel capacity for user model I is converged to

$$R_{\text{svc},L} N_L K d \prod_{k=1}^{K} k^{-\alpha} a_L, \tag{16}$$



(a) Number of channel K for user model I



(b) Number of channel K for user model II

Fig. 4. Capacity demand at $P_{\mathrm{unav}}=10^{-4}$ for varying number of channels, K. The scenario parameters are (a) $L=4$, $\alpha=0.6$, $N_1=N_2=N_3=N_4=250$, $a_1=a_2=a_3=a_4=0.8$, $R_{\mathrm{avc},l}=\{0.128, 0.384, 1.5, 6\}$ Mbps and (b) $L=4$, $\alpha=0.6$, $a=0.8$, $b_l=\{0.1, 0.3, 0.5, 0.1\}$, $N=1{,}000$, $R_{\mathrm{avc},l}=\{0.128, 0.384, 1.5, 6\}$ Mbps.

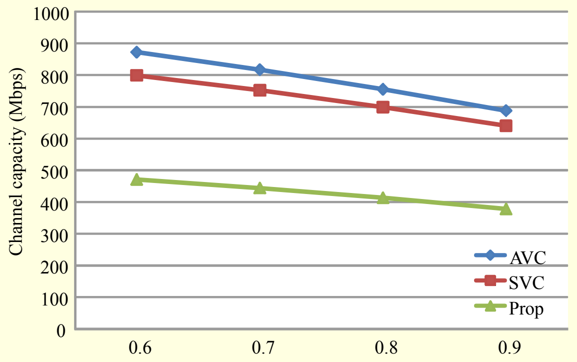
by increasing the number of channels. Also, for user model II, required channel capacity is converged to

$$R_{\mathrm{svc},L} N_L K d \prod_{k=1}^{K} k^{-\alpha} a(1-\Pr[b_1=0,\ldots,b_L=0]). \quad (17)$$

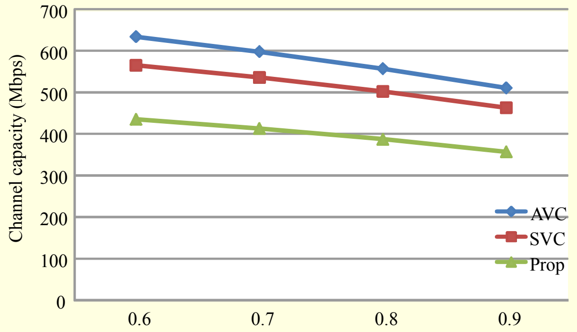
In Fig. 4, the required channel capacity of the proposed architecture is about 30% to 50% lower than the current network framework. That is, almost 1.5 to 2 times more SVC data can be transmitted in the current network framework than in the proposed network framework.

3. Impact of Zipf Distribution's Parameter α

Zipf distribution's parameter α stands for the factor of the style of users. The more α value each Zipf distribution has, the more users will watch the same channel, and the lower the channel capacity is required. In Fig. 5, the proposed model has the same trend as the lower channel capacity. As shown in Fig. 5, the required channel capacity is saturated due to Gaussian approximation. The Gaussian approximation does not account for the fact that there exists a limit capacity and the values obtained beyond it are non-realistic [11].



(a) Zipf distribution's parameter α for user model I

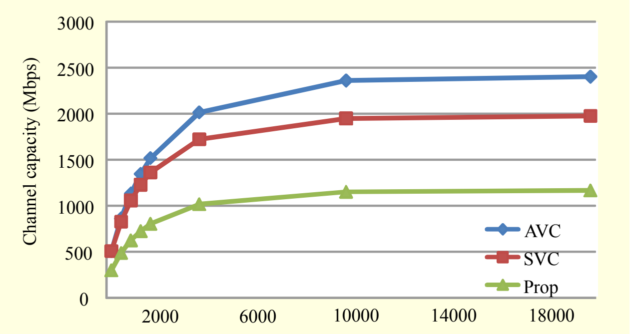


(b) Zipf distribution's parameter α for user model II

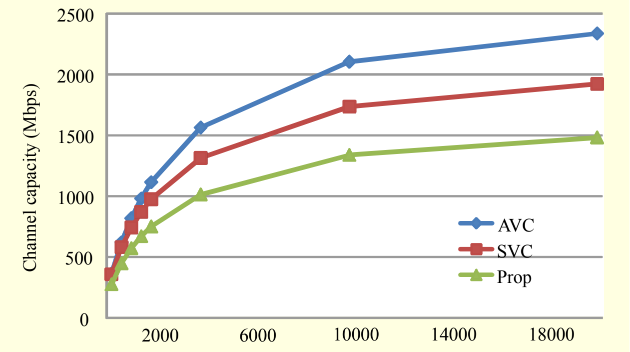
Fig. 5. Capacity demand in function of α for $P_{\mathrm{unav}}=10^{-4}$. The scenario parameters are (a) $L=4$, $K=300$, $\alpha=0.6$, $N_1=N_2=N_3=N_4=250$, $R_{\mathrm{avc},l}=\{0.128, 0.384, 1.5, 6\}$ Mbps, $a_1=a_2=a_3=a_4=0.8$ and (b) $L=4$, $K=300$, $\alpha=0.6$, $N=1{,}000$, $R_{\mathrm{avc},l}=\{0.128, 0.384, 1.5, 6\}$ Mbps, $a=0.8$, $b_l=\{0.1, 0.3, 0.5, 0.1\}$.

4. Impact of the Number of Users N

In this section, we show that by increasing the number of users, N, we can study the channel capacity changing on the current network framework and the proposed network framework. Figure 6 shows the simulation result with $N=20{,}000$ ($N_1=N_2=N_3=N_4=5{,}000$ in user model I). In this example we see the limitation of the Gaussian approximation. Some of the values for the capacity demand for $N\geq 10{,}000$ just exceed the maximum required capacity (namely, 2,360.6 Mbps in the AVC scenario, 1,949.4 Mbps in the SVC scenario, and 1,150.9 Mbps in the SVC scenario of the proposed framework in user model I). The Gaussian approximation does not account for the fact that there exists a limit capacity, which means obtaining values beyond it is not realistic. Therefore, in such situations we set the corresponding capacity-demand value to the maximum possible. We see that the difference in required capacity between user model I and user model II converges to one value. The reason for this is that with increasing N, at some point the probability that all resolutions of all channels are requested is nearly 1.



(a) Number of users *N* for user model I



(b) Number of users *N* for user model II

Fig. 6. Impact of N on the capacity demand when varying N up to 20,000 users for $P_{unav}=10^{-4}$. The scenario parameters are (a) $L=4$, $K=300$, $\alpha=0.6$, $N_1=N_2=N_3=N_4=N/4$, $a_1=a_2=a_3=a_4=0.8$, $R_{avc,l}=\{0.128, 0.384, 1.5, 6\}$ Mbps and (b) $L=4$, $K=300$, $\alpha=0.6$, a=0.8, $b_l=\{0.1, 0.3, 0.5, 0.1\}$, $R_{avc,l}=\{0.128, 0.384, 1.5, 6\}$ Mbps.

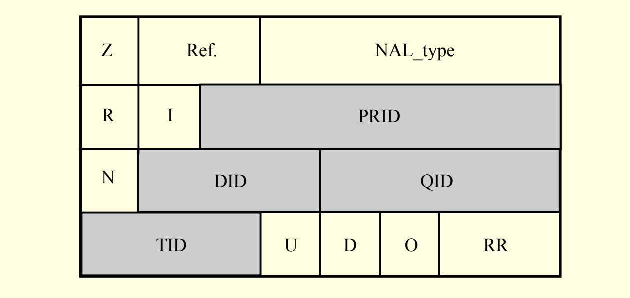


Fig. 7. SVC NAL unit.

V. Future Work

Figure 7 shows the header of a network abstraction layer (NAL) unit of SVC for network transmission. As indicated in Fig. 7, an NAL unit header has much information that defines features and orders of SVC data like priority ID (PRID), dependency ID (DID), quality ID (QID), and temporal ID (TID). If this information is extracted and broadcasted at the media gateway in Fig. 2, we can solve the bandwidth problem which occurs when SVC is applied to the current network. Therefore, we are currently developing a device which extracts and retransfers SVC data in media gateway. We are also developing a network processor which is specialized for flow control (such as an SVC data control) used in the device and studying an algorithm which efficiently extracts and retransfers SVC data. We hope that the channel capacity of SVC data is intelligently managed and controlled using both this device and algorithm.

VI. Conclusion

Due to the efficient usage and management of the terminal's resources, SVC has received the spotlight for the next generation video codec. In spite of it, SVC is not popular because implementing it into hardware with real-time processing is difficult due to its complexity. In addition, since the channel capacity is limited by SVC coding penalty ε, there is not a significant advantage compared with AVC. Though the first problem will be solved by a high-performance processing unit with a newer process technology, the second problem is not regarded as an easy one because there are many elements to consider, such as frameworks of the current network, transfer protocol, and codec. However, we proposed a router with an extractor for SVC extraction in this paper, and to have higher reliability of the simulation, the proposed model is compared with the current model on the same simulation performed reference [11].

Our simulation result showed that with the proposed network framework, transmitting SVC encoded data needs about 40% to 50% less channel capacity in user model I and about 30% to 40% less in user model II than the conventional network framework. This result demonstrates, against previous conclusions, that SVC is an efficient way to transmit video data. In addition, the proposed network framework is easy to apply on the current network system because it adds the EDE and an SVC extractor on the current router.

References

[1] S.-J. Bae, "A Standardization Trend of the Ubiquitous Contents Service," *HN FOCUS*, vol. 12, 2006, pp. 40-45 (in Korean).

[2] ITU-T and ISO/IEC JTC1, "Generic Coding of Moving Pictures and Associated Audio Information–Part 2: Video," ITU-T Recommendation H.262 ISO/IEC 13818-2 (MPEG-2), 1994.

[3] ITU-T, "Video Coding for Low Bit Rate Communication," ITU-T Recommendation H.263, Version 1: Nov. 1995, Version 2: Jan. 1998, Version 3: Nov. 2000.

[4] W. Li, "Overview of Fine Granularity Scalability in MPEG-4 Video Standard," *IEEE Trans. Circuit Syst. Video Technol.*, vol. 11, no. 3, Mar. 2001, pp. 301-317.

[5] ISO/IEC 14496-2, "Information Technology-Coding of AudioVisual

Objects - Part 2: Visual," International Standard, 2nd ed., Dec. 2001.

[6] ITU-T document, "Joint Draft 10 of SVC Amendment," Joint Video Team JVT-W201, JVT 23rd Meeting, San Jose, USA, Apr. 2007.

[7] ITU-T document, "Proposed Modifications for Joint Scalable Video Model," Joint Video Team JVT-W202, JVT 23rd Meeting, San Jose, USA, Apr. 2007.

[8] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the Scalable Extension of the H.264/MPEG-4 AVC Video Coding Standard," Joint Video Team, Doc. JVT-U145, Hangzhou, China, Oct. 2006.

[9] ITU-T and ISO/IEC JTC 1, "Advanced Video Coding for Generic Audiovisual Services," ITU-T Recommendation H.264 and ISO/IEC 14496-10 (MPEG4-AVC), Version 1: May 2003, Version 2: Jan. 2004, Version 3: Sept. 2004, Version 4: July 2005.

[10] IETF, "RTP: A Transport Protocol for Real-Time Applcation," RFC 1889, Jan. 1996.

[11] Z. Avramova et al., "Comparison of Simulcast and Scalable Video Coding in Terms of the Reqired Capacity in an IPTV Network." *Packet Video*, Lausanne, Switzerland, Nov. 2007.

[12] M. Wien, H. Schwarz, and T. Oelbaum, "Performance Analysis of SVC," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 9, 2007, pp. 1194-1203.

[13] I. Amonou et al., "On SVC Profile A," Joint Video Team of ITU-T VCEG and ISO/IEC MPEG, Doc. JVT-V056, Marrakech, Morocco, Jan. 2007.

[14] ISO/IEC JTC1/SC29/WG11 N6880 "Applications and Requirements for Scalable Video Coding," JVT document, Hong Kong, China, Jan. 2005.

[15] Peng Chen et al., "A Network-Adaptive SVC Streaming Architecture," *ICACT*, vol. 2, 2007, pp. 955-960.

Hyunpil Kim received the BS in electrical and electronic engineering in 2005 from Yonsei University, Seoul, Korea, where he is currently a PhD candidate. His research interests include flow-based network processors, multicore processor systems, scalable video coding for MID, process scheduling algorithms for multicore processors, and VLSI design.

Sukhan Lee received the BS and MS in electrical engineering from Yonsei University, Seoul, Korea, in 2007 and 2009, respectively. In 2009, he joined Samsung Techwin, Seongnam, Korea, where he is currently working on the development of systems-on- chip, image signal processors, networks-on-chip, and FPGA systems.

Junghee Lee received the BS and MS degrees in electronic engineering from Kyungpook National University, Daegu, Korea, in 1984 and 1990, respectively. Since 1984, she has been working at ETRI, Daejeon, Korea, where she is currently with the OmniFlow Processor Research Team. Her current research interests include management of network resources, future Internet, and Internet QoS.

Yongsurk Lee received his BS from Yonsei University, Seoul, Korea, in 1973, and his MS and PhD from the University of Michigan at Ann Arbor, in 1977 and 1981, respectively, all in electrical engineering. He worked in the field of microprocessor design in Silicon Valley from 1982 to 1992. Since 1993, he has been a professor in the Department of Electrical and Electronic Engineering, Yonsei University, Seoul, Korea. His current research interests include computer architectures, microprocessors, and network processors.