# 國立交通大學

## 網路工程研究所

## 碩 士 論 文

位元率-失真度最佳化網路影像串流技術
—可調變式編碼多點傳播

Rate-Distortion Optimized Video
Streaming Based Upon SVC Multicasting

研 究 生：林哲民

指導教授：邵家健　教授

中 華 民 國 九 十 七 年 九 月

# 位元率-失真度最佳化網路影像串流技術—可調變式編碼多點傳播
# Rate-Distortion Optimized Video Streaming Based Upon SVC Multicasting

研 究 生：林哲民　　Student：Che-Min Lin

指導教授：邵家健　　Advisor：John Kar-kin Zao

國 立 交 通 大 學
網 路 工 程 研 究 所
碩 士 論 文

A Thesis

Submitted to Institute of Network Engineering

College of Computer Science

National Chiao Tung University

in partial Fulfillment of the Requirements

for the Degree of

Master

in

Computer Science

September 2008

Hsinchu, Taiwan, Republic of China

中華民國九十七年九月

位元率-失真度最佳化網路影像串流技術—可調變式編碼多點傳播

學生：林哲民　　　　　　　　　　　　　　　　　　　　指導教授：邵家健

國立交通大學網路工程研究所 碩士班

## 摘　　要

我們的目的是在於建立一套傳輸技術，用於可調變式編碼的多點傳播，如何做到這般的傳輸是個有趣而複雜的問題，由於這樣的編碼方式會分成較多層次，目的是要使的在不同使用者裝置之間可重複利用的部分增加，使的傳輸上更有效率。並且在目前的多媒體傳輸的網路架構之下，大部分的運用是在無線網路之上，並且有在同個區域網路下裝置同質性較高的特性，本篇論文便是利用編碼的特性和網路的架構來設計傳輸機制。

我們提出了三個傳輸技巧使的網路傳輸更有效率，並且訂定簡化目標，依此建立一套訊息交換機制來協助網路節點做出決定，再提出了四種考慮不同因素的演算法架構來分配多媒體串流，最後建立一個模擬整個架構的程式來跑各種不同網路設定的實驗。

從整個實驗的結果上可以觀察到以下幾項，第一是做到對於使用者裝置上都能獲得較好的播放品質，第二是在整個架構傳輸的網路頻寬使用上是有效率的，第三是使的網路上的串流集中，最後是可以發現到整個網路上的多媒體中繼節點的頻寬的使用上也是有效率的。

Rate-Distortion Optimized Video Streaming Based Upon SVC Multicasting

Student: Che-Min Lin Advisor: John Kar-kin Zao

Institute of Network Engineering

National Chiao Tung University

## ABSTRACT

In this thesis, a video streaming scheme for conducting SVC multicasting is proposed. Scalable Video Coding consists of multiple layers and there is no redundancy between layers. We aim at such characteristics of SVC to design the video streaming scheme.

The proposed scheme employs a combination of three transport techniques. We also define an optimization model and build a message exchange mechanism based on it. We set up four decision ordering algorithms to allocate video streams in the network. Finally, we implement a simulation for running our scheme under different settings.

We can observe four results in our experiments: (1) better playback quality, (2) efficient bandwidth consumption in the network, (3) bitstream aggregation in the network, and (4) well-performed bandwidth usage of media gateways.

# 誌　　謝

首先我最要感謝邵家健老師，這篇論文的題目、方向以及方法，都是經由跟老師不斷的討論、琢磨出來的，期間老師一直給我很多的指導、建議與修正才順利的完成這篇論文，若沒有老師辛勤的指導，我將不能獨力完成此篇論文；另外感謝老師這兩年對我的教導與栽培，無論是學術上研究能力與思考能力的諄諄善誘，或甚至是待人處事上一些教誨與經驗分享，實在使我獲益良多。感謝這段時間以來一起為畢業打拼的夥伴郭芳伯，在與他互相勉勵互相討論論文研究中，獲得了許多寫論文的感想與技巧；另外感謝胡嘉錡學弟幫忙我分攤完成許多實驗的模擬和數據的建立與分析，使我能夠專心的完成我的論文；感謝黃國晉學長在過程中不斷的與我相談一些研究上該注意的細節與不斷給我建議，讓我做一些技術上的諮詢，以及對於一些事物上的經驗分享；感謝楊鈞凱學弟以及大二的劉皓偉、李翰、以及李文平學弟幫我整理一些圖檔，使我不至於花費太多在人工操作上；感謝郭宇軒學長最開始的一些指導以及教導我使用模擬軟體 OMNeT++的技巧；最後感謝工三 620 實驗室曾一起努力過所有成員，包括了張哲維、朱書玄、李明龍、黃凌軒、劉育志學長，曾輔國同學，以及陳星閔、邱博政、王梅瑛、李勝焜、蘇彥霖、楊馥戍學弟妹們。

最後我要感謝我的家人們，辛苦栽培我念書到現在研究所要畢業了，在我最困難的時候一直支持我，在我遇到瓶頸時給我鼓勵與意見，心中有萬千的感激，這篇論文是因為他們而產生的。還要感謝褘涵，謝謝她給予我十分多心靈上的支持與鼓勵，在我盪到谷底的時候始終在我旁邊陪著我度過。

這篇論文需要感謝的人太多了，謝謝你(妳)們。

# Contents

# Figures

# Tables

# Chapter 1 Introduction

## 1.1 Problem Statement

The Scalable Video Coding (SVC) Extension of H.264/AVC [1] specifies a multilayer predictive encoding scheme that enables different video coding layers to be extracted and decoded selectively by different user devices according to their playback capability and transport network throughput. This desirable property causes SVC to be regarded as the ideal technology for providing multimedia multicasting to heterogeneous viewing devices. Nevertheless, a close examination of SVC data format reveals two potential handicaps of its use in heterogeneous multicasting. First, the cumulative bit rates of the SVC layers extracted by individual user devices routinely exceed the bit rates of their corresponding AVC bitstreams due to the loss of coding efficiency in multilayer encoding. When the SVC bitstreams are transported over bandwidth stringent networks such as the cellular telephone networks, this bit rate inflation often implies cost inflation or performance degradation. Second, the inter-layer dependence relations introduced by the multilayer predictive encoding process requires a viewing device to extract and decode all the reference coding layers on which the target layer of the device depends before the device can decode its target layer. Any un-recoverable loss of a reference layer will reduce the decodable layers to those below the lost layer due to inter-layer dependency and thus degrade the playback quality on the viewing devices. In a best-effort delivery network such as the Internet, a congested along a transport rate can thus a degradation of the video playback quality. Thus, a bandwidth efficient and loss resilient transport scheme must be devised in order to ensure SVC to be the preferred encoding format for video streaming.

In this thesis, we attempted to improve the bandwidth efficiency of heterogeneous multi-casting of SVC bitstreams by devising a *distributed bandwidth allocation scheme* that can be implemented by every intermediate node along the transport paths. The objective of our scheme is to maximize the overall video playback quality of the viewing devices enrolled in the heterogeneous multicasting session while minimizing the total bandwidth utilization along the transport paths.

Our bandwidth allocation scheme assumes that the SVC bitstream is transported to different types of viewing devices scattered across multiple stub networks over the Internet via several *multicast trees* each of which is made up of tiers of intermediate nodes (known as the *media gateways*). Every media gateway (MG) receives specific SVC layers from a finite set of upstream nodes along multiple inbound network connections and dispatches these layers to specific downstream nodes through multiple outbound network connections. Each MG decides how to allocate its outbound bandwidth for transporting specific SVC layers to specific downstream nodes based only on the local information it gathers from its upstream and downstream nodes. No information exchange among the media gateways in the same tier is allowed. To simplify the preliminary design of our scheme, we ignore the asynchronous and lossy nature of the transport network with respect to information exchanges regarding bandwidth allocation. Hence, the decisions of individual MGs are unaffected by the order and the potential loss of control packets.

## 1.2 Research Approach

We will first introduce transport techniques for efficient purpose of SVC multicasting.

- **Broadcasting over stub networks** is to compensate the bit rate inflation introduced by SVC multilayer encoding process.
- **En-route Bitstream Aggregation over long haul network** is to minimize upstream bandwidth while maximize serviced device numbers.
- **Information Dispersal and Multipath Diversity** is to cut layers into equal-rated flows to fit into media gateway more easily and to add error protections on layers unequally by their importance shown in the inter-layer dependency.

There are a lot of different video streams in the network and to we will arrange bitstream allocation of them in a bandwidth efficient and loss resilient transport under these three techniques. Then we will formulate our goal in an optimization models about both playback quality and network bandwidth consumption. And we will design algorithms under this model. There are some considerations of SVC we should introduce. Both theoretic and empirical work point to some important factors that are highly relative to the performance of bandwidth allocation of the SVC multicasting as the followings:

- **Inter-layer Dependency**: To avoid wastes of bandwidth, viewing devices must not subscribe to any layer that can't be decoded. On the other hand, they will make subscription based on the extraction path of SVC.
- **Bitstream Characteristics**: SVC layers have their own bandwidth consumption and improvement of playback quality. The Rate Distortion Ratio of a single SVC layer

influences the efficiency of transporting it directly.

- **User Device Types and Distribution over the Internet**: Since we want to arrange the bandwidth allocation among large amount of devices, user device types and their distribution should be considered carefully. The largest amount of viewing devices will dominate the performance of bandwidth allocation.
- **Network Topology**: Since we want to arrange the bandwidth allocation over large scale networks, to make a better bandwidth allocation under a specific sub-network highly depends on its topology.

We will consider these factors and design different decision algorithms to approach our goal.

## 1.3 Thesis Outline

In chapter 2, we will briefly introduce the related work of SVC multicasting. In chapter 3, we will introduce efficient transport techniques. In chapter 4, we describe the optimization model and decision algorithms for bandwidth allocation. In chapter 5, we will explain our message exchange mechanism in detail. In chapter 6, we will show how we implement the scheme and the how we design the experiment models. In chapter 7, we will define the measurements and give the results and its analysis. In chapter 8, we will make a conclusion and tell accomplishments in our works and our future works. And the end of paper, we add a glossary about definitions of symbols we used in this thesis.

# Chapter 2 Related Work

## 2.1 SVC Inter-layer Dependencies and Rate-Distortion Performance

In [2], it is an overview of Scalable Video Coding Extension of H.264/AVC Standard. Scalable Video Coding (SVC) provides scalabilities refers to the removal of parts of the video bitstream in order to adapt it to the various needs or preferences of end users as well as to varying terminal capabilities or network conditions. It has three modes about temporal, spatial, and quality scalability. Spatial scalability and temporal scalability represent the picture size (spatial resolution) or frame rate (temporal resolution). With quality scalability, the bitstream provides the same spatial and temporal resolution, but with a lower fidelity which is often referred to as signal-to-noise ratio (SNR Scalability).

For different scalabilities, viewing devices can easily choose layers based on device capabilities and subscribe layers according to extraction path from base layer to enhancement layers. Hence, they may receive fewer layers if congestion occurs.

Scalable Video Coding encodes a video clips into a base layer and many enhancement layers with different resolutions. To receive layers following different extraction paths will get decodable video clips with the same content but in different resolutions. Every extraction path starts from the same layer which is called the Base Layer. Every viewing device could improve its playback quality by receiving more enhancement layers. Each layer has different improvement of playback quality and different bit rate. The measurement of distortion we used is called Peak Signal to Noise Ratio (PSNR). These characteristics of different layers effect the decisions of serving nodes in the network.

## 2.2 Bandwidth Efficiency of Internet Video Streaming using AVC vs. SVC

In [3], Kim compared three approaches for video multicasting as following:

- The replicated stream approach [4], [5]

  In this approach, source is encoded into multiple independent video streams with different bit-rates because of different compression parameters. And then these streams will be multicast over the network to all viewing devices. AVC Simulcasting is similar to this approach.

- The cumulative layering approach [6], [7]

  In this approach, source in encoded into one base layer and multiple enhancement layers. Base layer can be decoded independently but enhancement layers should be decoded cumulatively. Devices improve their playback quality by receiving more enhancement layers. SVC Multicasting is this kind of approach.

- The non-cumulative layering approach [8], [9] In this approach, source in encoded into multiple independent layers. All layers can be decoded independently and devices improve their playback quality by receiving more layers.

As we referred, the first one is like the AVC Simulcasting and the second one is the SVC Multicasting. He considered and estimated different overheads of layered video such as packetization overhead, protocol overhead, and error control overhead. Then he gave basic Rate Allocation and Stream Assignment algorithms for three different approaches for simulations. He simulated three approaches and analyzed with four measurement average reception rate, average effective reception rate, total bandwidth usage, and efficiency. He got conclusions that the performance of heterogeneous video multicasting schemes depend on the amount of layering overhead and said that if the effective reception rate is the same, replicated stream multicasting is preferred except receivers are clustered in few domains.

However, if we use unicast in the local subnet, SVC layers are redundantly transported and the independency of SVC layers has not been used. Hence, we should broadcast layers in the local subnet to avoid transport of redundant layers. Furthermore, the question of SVC multicasting is not WHETHER SVC is efficient. It is HOW to transport SVC in a efficient way with using its merits and to compensate its shortcomings.

## 2.3 Early Attempt of Peer-to-Peer SVC Streaming

In [10], Baccichet et. al. proposed a combined use of tree-based push transport and en-route progressive rate adaptation of SVC bitstreams to achieve low-latency peer-to-peer video streaming. By enabling the intermediate relaying nodes to extract the SVC layers demanded by their down-stream peers, the proposed scheme reduces the chance of link congestion. Moreover, by forwarding the IP packets carrying an SVC bitstream along several multicast trees, the scheme is expected to amortize the impact of individual packet loss and link failure. Baccichet's scheme can indeed eliminate the need for layer synchronization and thus reduce SVC transport latency by pushing the multi-layer bitstream through several multicast trees.

However, its effectiveness in removing link congestions is somewhat dubious because en-route SVC rate adaptation can only work when same types of user devices are gathered in clusters over the multicast network, which is the case we refer to as *homogeneous clustering*. If different types of devices are dispersed over the network as in the case of *heterogeneous clustering* then the relaying nodes have no choice but to forward almost the entire SVC bitstream to the downstream peers. Besides, the usefulness of IP packet dispersal (without the employment of erasure protection) is also questionable because most SVC NAL units are likely carried in multiple IP packets. The loss of any of these packets will render the entire NAL unit undecodable. As a result, Baccichet's scheme performs well only under low traffic load and mild packet loss.

# Chapter 3 Techniques

The proposed heterogeneous SVC multicasting scheme was based upon the idea of *multipath transport* of *incremental SVC NAL sets* (abbreviated hereafter as SVC-INS or simply INS). The scheme employs a combination of three transport techniques: (1) broadcasting of SVC-INS in local service subnets, (2) aggregation of data flows along their transport paths and (3) multipath transportation of SVC-INS over long-haul networks. This multicasting scheme is particularly suited for delivering SVC bitstreams to different types of user devices scattered over geographically dispersed local subnets. [Figure 1] illustrates a typical deployment of the scheme over multiple wireless networks that span across the Internet.

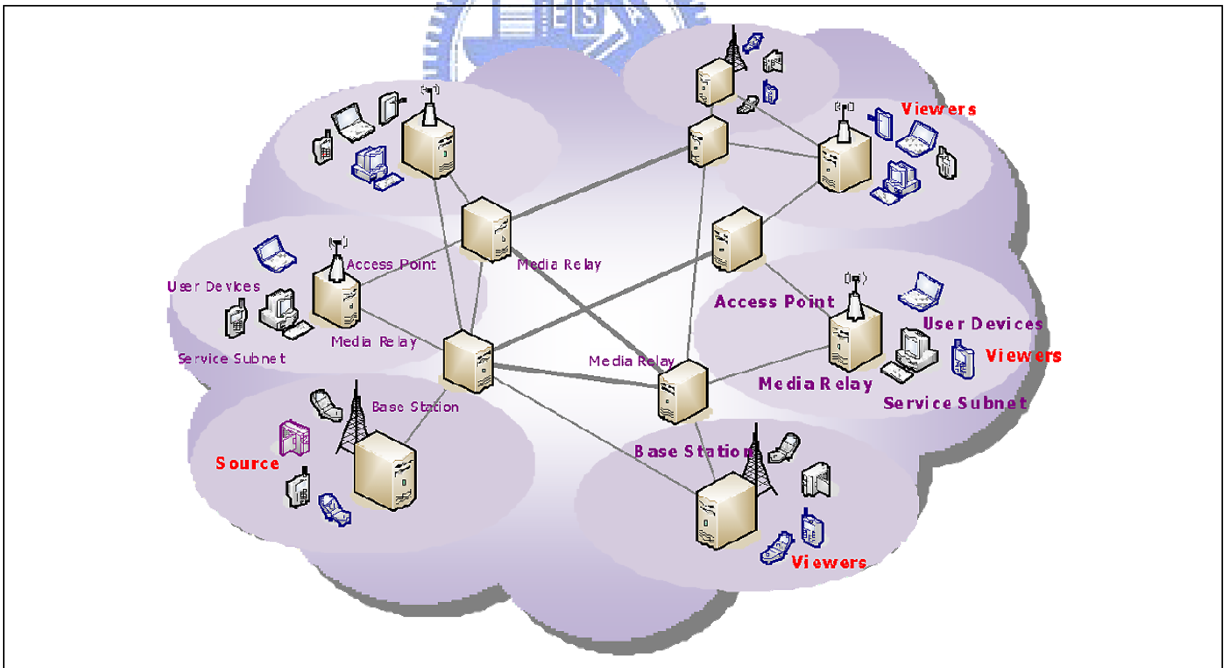



Figure 1: Heterogeneous SVC Multicasting Architecture

Like Baccichet's scheme, our scheme relies on intermediate nodes, known as the *media gateways* (*MGs*), to shape transport traffic. Instead of having the relaying nodes performing bitstream extraction and rate adaptation, our scheme requires its MRs to perform the following operations: (1) broadcast the SVC-INS received from their upstream nodes to the user devices present in their local service subsets; (2) request the SVC-INS needed by their downstream nodes from their upstream nodes while trying to minimize the number of data flows between these nodes; (3) disperse the data flows carrying the same SVC-INS among different MRs in order to balance the traffic load and use unequal erasure protection (UEP) in order to protect data flows carrying different SVC-INS.

Our scheme can be adapted to both content-delivery and peer-to-peer architectures. When it is deployed on top of a content-delivery architecture such as HP MSM-CDM [11], the media relays (MRs) shall be deployed at each service subnets as well as strategic points-of-presence (PoPs) in order to maximize the benefit of flow aggregation and load balancing. When the scheme is used in a peer-to-peer application, some of the more capable peer nodes will select to play the role of MRs and perform the required operations. The peer-to-peer application can use any multiple tree push architecture including Stanford Peer-to-Peer Multicast (SPPM) protocol [1], Split-Stream [12] or Trickle [13] to establish the multicast mesh. In both cases, the MRs shall be deployed at the edge of the Internet in the user service subnets or the ISP stub networks that are connected to the Internet backbone.

## 3.1 Network Topology

For the heterogeneous SVC multicast architecture in [Figure 1], we make an assumption for a network model in this thesis. For a video source and many viewing devices which will subscribe to the video, there are many other intermediate nodes in the network called Media Gateways (MGs) for relaying video to viewing devices. We assume that there are three tiers in the network, that is, tier 0, 1, and 2 from bottom to top and MGs in the same tier may not exchange information with each other. MGs in the tier 0 serve viewing devices in stub network and face to requests from devices directly. MGs in the tier 1 located in upstream aggregation ISPs is as the backbone of the network. MGs in the tier 2 are adjacent to video source and help the video source multicast SVC layers. Our techniques, algorithms, and mechanisms are based on this network model from the bottom to the top.

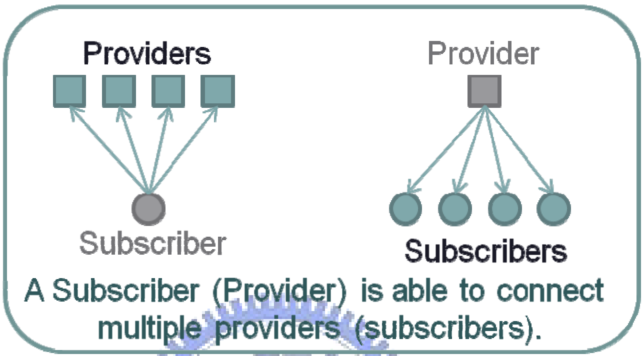



Figure 2: Subscribers and Providers

Since we use a network assumption about tiers, we call them providers and subscribers between two tiers as in [Figure 2]. A subscriber or a provider is able to connect to multiple providers or subscribers. The provider will determine what layer will be served on what link and allocate its bandwidth to the subscription, and the subscriber will make decisions about sending subscription to which provider.

## 3.2 Broadcasting over Stub Networks

The first bandwidth saving technique we employed was the simple broadcasting of the SVC-INS in demand over the local service subnets to the user devices connected to the subnets. Each SVC-INS should be transmitted over separate broadcast channel and protected with

1896

forward error correction (FEC) at the physical layer. This broadcasting mechanism can be used not only over wireless and cellular networks but can also be used in most local access networks including wired Ethernet, cable television networks and ADSL as they are all inherently broadcasting networks.

If an SVC bitstream is subscribed by many user devices in the same subnet then broadcasting of SVC-INS eliminates the need to send multiple copies of the same bitstream over the network and thus incur significant reduction of bandwidth usage. This saving is more than enough to compensate the bit rate inflation introduced by the SVC multilayer encoding process. If an SVC bitstream has few subscribers in a local subnet then it should be converted to H.264 AVC format through SVC-to-AVC bitstream rewriting [14].

## 3.3 En-route Bitstream Aggregation

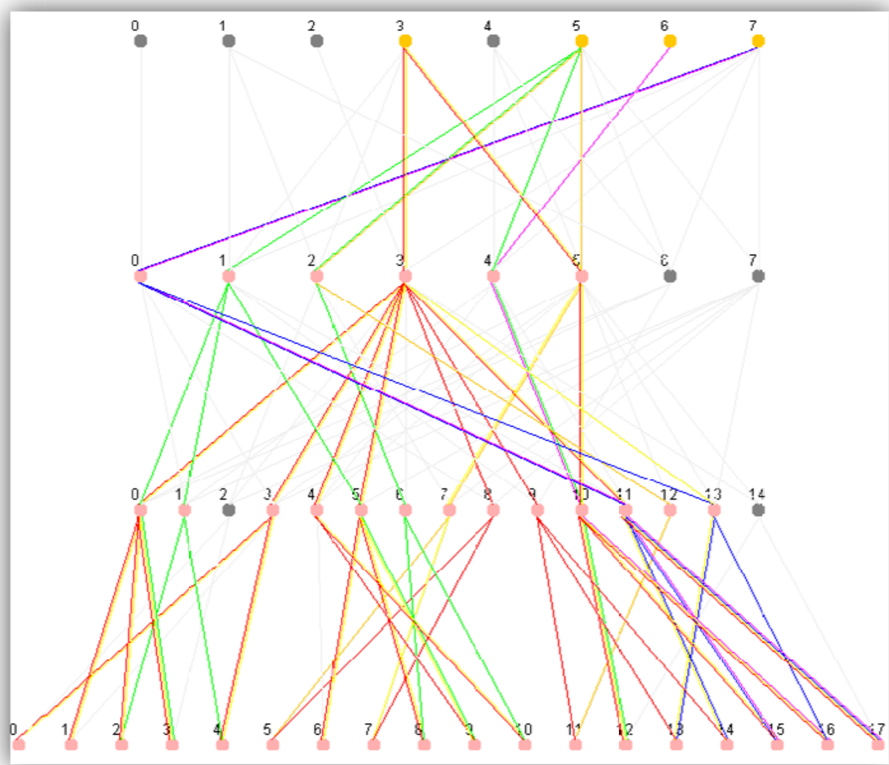


Figure 3: Aggregation of data flows

Another bandwidth saving technique is the aggregation of data flows [Figure 3] along their transport routes. In an SVC multicast, many user devices will require the same set of layer in order to playback the video. The media gateways (MGs) embedded in the multicast trees can function as aggregation points of these data flows.

We designed a *bandwidth reservation protocol* to implement the en-route aggregation of data flows. The protocol adopted a bottom-up approach in conducting bandwidth negotiations among two tiers of MGs like a 4-way handshake protocol. Between the two tiers, the nodes in the lower tier are called *subscribers*, and the upper nodes are called *providers*. Subscribers know the layer it needs and the providers it can access in the upper tier. The purpose of the negotiation is to enable the subscriber to choose a provider for each layer or derived data flow. To do so, the subscriber keeps a matrix of serving probability information of the accessible providers, and makes decisions based on the probability information. Providers should also make decisions on whether to accept or refuse the requests submitted by the subscribers. Every provider publishes a probability list of layers it intends to transport.

## 3.4 Information Dispersal and Multipath Transport

In order to ensure that different SVC-INS can be delivered to the user devices that need them even amidst significant traffic load and packet loss, our scheme employs the communication techniques of multipath diversity (MD) and unequal erasure protection (UEP) to protect the transport of individual INS. Such a protection is particularly important for those INS that are extracted from the lower SVC layers as they are the ones on which the upper SVC layers are depended for motion and residual prediction. Under the protection, each SVC-INS is coded and divided into multiple equal-rated data flows by a spatial-temporal UEP encoder. Each of these data flows is then dispersed into different transport routes in the multicast trees. To decode an INS, a user device or a media relay only need to collect sufficient amount of code words derived from the INS and submit them to a UEP decoder. The design of a spatial-temporal UEP encoder suitable for SVC transport is currently underway. Hence, the investigation of the effectiveness of this technique lies beyond the scope of this paper.

# Chapter 4 Algorithms

## 4.1 Optimization Model

Although falling short of producing a formal proof, we postulate that the problem of bandwidth allocation for a SVC multicasting session that maximizing device playback quality while minimizes total bandwidth consumption is a NP-Hard problem. Hence, a distributed bandwidth allocation algorithm must rely on iterative heuristics to search for the optimal solution. In this chapter, we propose an iterative negotiation between providers and subscribers and four different algorithms to arrange these negotiations that aim at directing the iterative negotiation towards an optimal solution.

We want to maximize the playback quality of viewing devices but we also hope to transport SVC layers in a bandwidth efficient network. Hence, we define the "Rate-Distortion Gain (R-D Gain)" function as (1). The numerator is the PSNR Improvement that is the playback quality gain of a device from receiving a specific layer and the denominator means that the bandwidth consumption of the layer in the network. For every received layer in every viewing device, we divide the PSNR improvement by the cumulative bandwidth consumption of the transport path of such layer. We assume that the transport path of Layer $\mathrm{L}$ in $\mathrm{D}(\jmath)$ from bottom to top is $\{\mathrm{D}(\jmath), \mathrm{MG}(0,\jmath), \mathrm{MG}(1,\jmath), \dots, \mathrm{MG}(\mathrm{T},\jmath)\}$ and $\mathrm{n}_{\mathrm{U}(\mathbb{L})}(\imath,\jmath)$ represents the serving devices counts of device type $\mathrm{U}(\mathbb{L})$ in the transport sub-tree which has the root $\mathrm{MG}(\imath,\jmath)$. Then we calculate the R-D Gain of Layer $\tilde{\mathrm{L}}$ in $\mathrm{D}(\jmath)$ where $\mathrm{d}(\mathrm{D}(\jmath), \mathrm{L})$ means the distortion gain of $\mathrm{D}(\jmath)$ receiving $\mathrm{L}$ and the first term in the denominator means the copies of

L in the local subnet. The value is 1 if unicast, and $\frac{r(L)}{\sum_{\forall U(\mathbb{L}), L \in \mathbb{L}} n_{U(\mathbb{L})}(0,j)}$ if broadcast in the local subnet.

$$\gamma(D(j), L) = \frac{d(D(j), L)}{\frac{r(L)}{\sum_{\forall U(\mathbb{L}), L \in \mathbb{L}} n_{U(\mathbb{L})}(0, j)} + \sum_{i=0}^{T-1} \left( \frac{r(L)}{\sum_{\forall U(\mathbb{L}), L \in \mathbb{L}} n_{U(\mathbb{L})}(i, j)} \right)} \quad (1)$$

## 4.2 Bandwidth Allocation Algorithm

The basic principle of our scheme is that viewing devices will not send subscription requests for layers that can not be decoded. That is, the bottom-up scheme will repeat many rounds and all viewing devices send requests for one or more decodable layers at once based on the dependency of layers. We'll first introduce the negotiation between providers and subscribers. This will show us how every provider and every subscriber negotiates bandwidth allocation of a layer on a network link. Then we design four different sequencing algorithms that differ from considering factors increasingly, as we have referred to, bitstream characteristics, device type population distribution, and network topology. These algorithms will determine whole network or local viewing devices make requests of layers in what kind of order.

### 4.2.1 Provider-Subscriber Negotiation for Bandwidth Allocation

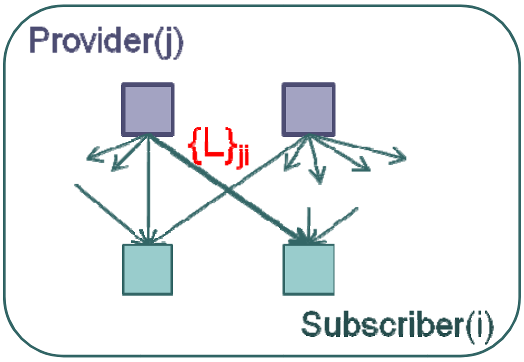



Figure 4: Negotiation for a Layer Set on a link between Subscriber and Provider

This negotiation is to determine a specific layer set of layers transported on a specific link in the network between providers and subscribers as in [Figure 4]. The provider will first compute the expectation value about layers that could be allocated from it for each subscriber as (2). The first term is allowed layer counts for the provider $\mathrm{MG}(i+1,j)$ and the second term means the importance of $\mathrm{MG}(i,j)$ where importance is computed by ratio of expected R-D Gain of a specific subscriber among all subscribers. Then the subscriber will choose the provider which has largest expectation to serve it as (3). At last, providers will cumulate these requests and compare the expectation value of R-D Gain of accepting each request and allocate the bandwidth from the biggest one. The expected R-D Gain is computed as (4) by subscribers and sent to providers. It means that how much R-D Gain any provider could earn if it accepts this request. R-D Gain is total PSNR Improvement of devices divided by bit rate of that layer and then divided by the accessible provider counts $\xi(i,j)$.

$$
\begin{cases}
\mathrm{E_{srv}}\left(i+1, j; i, j; \tilde{\mathrm{L}}\right) = \mathrm{LayerCount_{Allowed}}\left(i+1, j, \tilde{\mathrm{L}}\right) \times \dfrac{\gamma\left(i, j, \tilde{\mathrm{L}}\right)}{\sum_{\forall \mathrm{MG}\,(i,s) \in \mathbb{N}(i+1,j)} \gamma\left(i, s, \tilde{\mathrm{L}}\right)}, \\
\gamma\left(i, j, \tilde{\mathrm{L}}\right) = \dfrac{1}{\xi(i, j)} \times \left( \sum_{\forall \mathrm{U}(\mathbb{L}), \tilde{\mathrm{L}} \in \mathbb{L}} \mathrm{n}_{\mathrm{U}(\mathbb{L})}(i, j) \times \dfrac{\mathrm{d}\left(\mathrm{U}(\mathbb{L}), \tilde{\mathrm{L}}\right)}{\mathrm{r}\left(\tilde{\mathrm{L}}\right)} \right)
\end{cases}
\tag{2}
$$

$$
\mathrm{MG}\ (i+1, \hat{j}) : \hat{j} = j \mid \max_{j} \mathrm{E_{srv}}\left(i+1, j; i, j; \tilde{\mathrm{L}}\right) \tag{3}
$$

$$
\gamma\left(i, j, \tilde{\mathrm{L}}\right) = \frac{1}{\xi(i, j)} \times \left( \sum_{\forall \mathrm{U}(\mathbb{L}), \tilde{\mathrm{L}} \in \mathbb{L}} \mathrm{n}_{\mathrm{U}(\mathbb{L})}(i, j) \times \frac{\mathrm{d}\left(\mathrm{U}(\mathbb{L}), \tilde{\mathrm{L}}\right)}{\mathrm{r}\left(\tilde{\mathrm{L}}\right)} \right) \tag{4}
$$

### 4.2.2 Ordering of Bandwidth Allocation Negotiations

#### 4.2.2.1 Algorithm 1: Ordering without Consideration of Device Distribution

In this ordering of bandwidth allocation negotiations, we allocate the necessary bandwidth to transport SVC layers *one at a time* from the base layer to the highest enhancement layers according to their inter-layer dependency relationship and rate-distortion information. The numbers of rounds will be as same as the amount of layers. The Ordering Algorithm is as [Table 1]. The order is according to the dependency and if there is more than one decodable layer, we will choose a layer based on its rate distortion information. Hence, we will compare their Rate Distortion gain for such layer, which is the sum of PSNR improvement for all device types that can play this layer divided by its bit rate as [Table 2].

The advantage of this way is that the order of the resource reservation could be determined right after encoding, and it performs well when the population distribution makes no difference since it is not sensitive to the population distribution. The performance will be bad if the population distribution is biased.

```
Set Max_RD_Gain to 0
Set Max_Layer to NULL
FOR each layer that is decodable
    CALL Rate_Distortion_Gain with layer RETURNING RD_Gain
    IF RD_Gain > Max_RD_Gain THEN
        Set Max_RD_Gain to RD_Gain
        Set Max_Layer to layer
    END IF
END FOR
FOR each tier from bottom to top
    FOR each node in this tier
        CALL ProviderSubscriberNegotiation with tier, node, and Max_Layer
    END FOR
END FOR
```

Table 1: Pseudo Code of Ordering Algorithm in Algorithm 1 and 2

```
Set Total_RD_Gain to 0
FOR each device_type that can improve its playback quality by receiving LAYER
    Set TMP_RD_Gain to PSNR Improvement for device_type receiving LAYER
    DIVIDE TMP_RD_Gain by Bit Rate of LAYER
    INCREMENT Total_RD_Gain by TMP_RD_Gain
END FOR
RETURN Total_RD_Gain
```

Table 2: Pseudo Code of Function Rate_Distortion_Gain(LAYER) in Algorithm 1

#### 4.2.2.2 Algorithm 2: Ordering with Consideration of Global Device Distribution

In this ordering of bandwidth allocation negotiations, we allocate the necessary bandwidth to transport SVC layers *one at a time* from the base layer to the highest enhancement layers according to their inter-layer dependency relationship and rate-distortion information, and furthermore, the *global device population distribution* is considered in this case. Times of execution rounds will be as same as the number of layers. The Global Ordering Algorithm is as [Table 1]. The order is according to the dependency and if there is more than one decodable layer, we will choose a layer based on its single layer rate-distortion characteristic and the population distribution of viewing devices. Hence, we will compare their R-D gain for types of viewing devices, which is the sum of PSNR improvement for all viewing devices that could play that layer divided by its bit rate as [Table 3]. MGs in the top of the mesh can get the device population information easily by aggregating from the bottom and then propagate the global order to the bottom.

The advantage of this algorithm is that it is sensitive to the population distribution, so different distribution may have orders that fit in such situation. But media gateways and devices may need extra information besides a local network since this is a global view that the population information will be cumulated to the top and then propagate the determined order to the bottom devices.

| |
|---|
| Set Total_RD_Gain to 0<br>FOR each *device_type* that can improve its playback quality by receiving LAYER<br>    Set TMP_RD_Gain to PSNR Improvement for *device_type* receiving LAYER<br>    MULTIPLY TMP_RD_Gain by population of *device type*<br>    DIVIDE TMP_RD_Gain by Bit Rate of LAYER<br>    INCREMENT Total_RD_Gain by TMP_RD_Gain<br>END FOR<br>RETURN Total_RD_Gain |

Table 3: Pseudo Code of Function Rate_Distortion_Gain(LAYER) in Algorithm 2

#### 4.2.2.3 Algorithm 3: Ordering with Local Fair Competition

In this ordering of bandwidth allocation negotiations, we allocate the necessary bandwidth to transport *all decodable SVC layers at a time* from the base layer to the highest enhancement layers according to their inter-layer dependency relationship, rate-distortion information, and the *local device population distribution* as in [Table 4]. So competitions between different layers exist in this ordering in sub-trees of MGs. Differ from algorithm 1 and 2, there will be more than one layer allocated in the network and the competition of requests for all decodable layers may occur. Different local population distributions and local network topology affect local bandwidth allocations.

The advantage of this case is that not only population distribution is considered but also the network topology takes place in the decisions. This local information may make the bandwidth allocation perform better, but the length of extraction path will have large effect in this case.

```
FOR each tier from bottom to top
    FOR each node in this tier
        FOR each layer that is decodable for devices or required by media gateways
            CALL ProviderSubscriberNegotiation with tier, node, and layer
        END FOR
    END FOR
END FOR
```

Table 4: Pseudo Code of Algorithm 3

#### 4.2.2.4 Algorithm 4: Local Ordering with Local Dominated Request

In this ordering of bandwidth allocation negotiations, we allocate the necessary bandwidth to transport *one dominated SVC layer (which brings the largest R-D Gain in the sub-tree) at a time* for one MG from the base layer to the highest enhancement layers according to their inter-layer dependency relationship, rate-distortion information, and the *local device population distribution* as in [Table 5]. There will be many requests for different layers competing in the network but there is a difference between Algorithm 3. That is, all decodable layers will be requested from a subnet but in this Algorithm 4 there will be only one request from a subnet which will bring the largest R-D Gain. The request which represents the largest R-D Gain in a subnet will beat other requests rather than accepting all decodable layers.

The advantage is that a local network view of device distribution will make accurate requests and nodes will not need extra information of nodes in other subnets. The length of extraction path will have smaller effect than Algorithm 3.

```
FOR each tier from bottom to top
    FOR each node in this tier
        FOR each layer that is decodable for devices or required by media gateways
            CALL ProviderSubscriberNegotiation[1] with tier, node, and layer
        END FOR
    END FOR
END FOR
```

Table 5: Pseudo Code of Algorithm 4

#### 4.2.2.5 Examples of four algorithms

We assume that the video source have been encoded into four layers, base layer 0 and enhancement layer 1, 2, and 3. Layer 1 and 2 depend on layer 0 and layer 3 depend on layer 1. Suppose that the device population in whole network is as $\|\{0,3\}\| > \|\{0,1\}\| > \|\{0,2\}\| > \|\{0\}\|$, but the order of Rate-Distortion Gain is as 0>1>2>3. For algorithm 1 and 2, there is only one layer allocated in the network. Allocated order in algorithm 1 is {0, 1, 2, 3} which is the same as the layer R-D characteristics but {0, 1, 3, 2} in algorithm 2 since the population affect the order directly. For algorithm 3, all decodable layers will be allocated in the network, so layer 0 that all devices need will be allocated first. Layer 1 and layer 2 will be the next because they both depend on layer 0 and layer 3 will be the last one since it has the longest extraction path. Algorithm 4 is quite different so we provide another example for it.

Assume that there are two MGs A, B in the same tier and MG A has 1000 devices requesting for layer 1 in its sub-tree and 10 for layer 2 where MG B has 10 devices for layer 1 in its

[1] There is only one difference in it that Providers will provide only one layer which brings largest Rate Distortion Gain for serving it in the sub-tree in the same round. Provider will only serve the dominated layer request in its subnet rather than all decodable layers.

sub-tree and 1000 for layer 2. Suppose that layer 0 has already been allocated, so layer 1 and layer 2 are both decodable now. For algorithm 3 that all decodable layers will be allocated, layer 1 and 2 will be allocated both in MG A and B and layer 3 will be allocated in the next round. But for algorithm 4, only the dominated layer in the sub-tree will be allocated. That is, MG A will accept requests for layer 1 and postpone requests for layer 2 and MG B does the opposite. In the next round, MG A will consider layer 2 and 3 at the same time since layer 3 is decodable after layer 1 being allocated, and MG B can only allocate layer 1 because it is the only decodable layer.

# Chapter 5 Message Exchange

A bandwidth saving technique we employed was the aggregation of data flows along their transport routes. In an SVC multicast, many user devices will require the same set of layers in order to playback the video. The media gateways (MGs) embedded in the multicast trees can function as aggregation points of these data flows.

We designed a *distributed bandwidth reservation message exchange mechanism* to realize en-route aggregation of data flows.

The mechanism is a multiple round bottom-up architecture conducted among two consecutive tiers of network nodes known as the *subscribers* and the *providers*. Each subscriber can negotiate with certain number of providers about the layer it needs and the number of user devices it serves. However, no communication between non-consecutive tiers is permitted nor communication among providers or subscribers in the same tier. The following section will show how we do the message exchange between subscribers and providers.

## 5.1 Message Exchange Mechanism

The bandwidth allocation is initiated by user devices in the lowest tier and propagated towards the providers in the top tier. As in [Figure 5], subscribers register themselves to providers first about the served layer and layer set that refer served layer and the amount of devices that each layer is demanded by sending HELLO messages. The message is for providers to understand how many subscribers connect to it and their requirements and then subscribers submit resource requests REQ messages that maximize probability of successful allocation and maximize aggregation of bitstream. Requests are submitted to chosen providers based on expectation value about layers that could be allocated by providers. They will choose one provider that has highest expectation values for served layer that they demand in each round. If the request failed, they will turn to the provider that has the highest expectation among the rests in next round.

a) The provider receives registrations from subscribers and then derives serving probabilities for every subscriber by computing how many requests are allowed and subscribers' importance for it, and then sends the probability to subscribers who registered to it by sending ACK messages. Then they receive all subscribers' requests in batches and make allocation by comparing the expected R-D Gain of each subscriber if the provider serves its request.

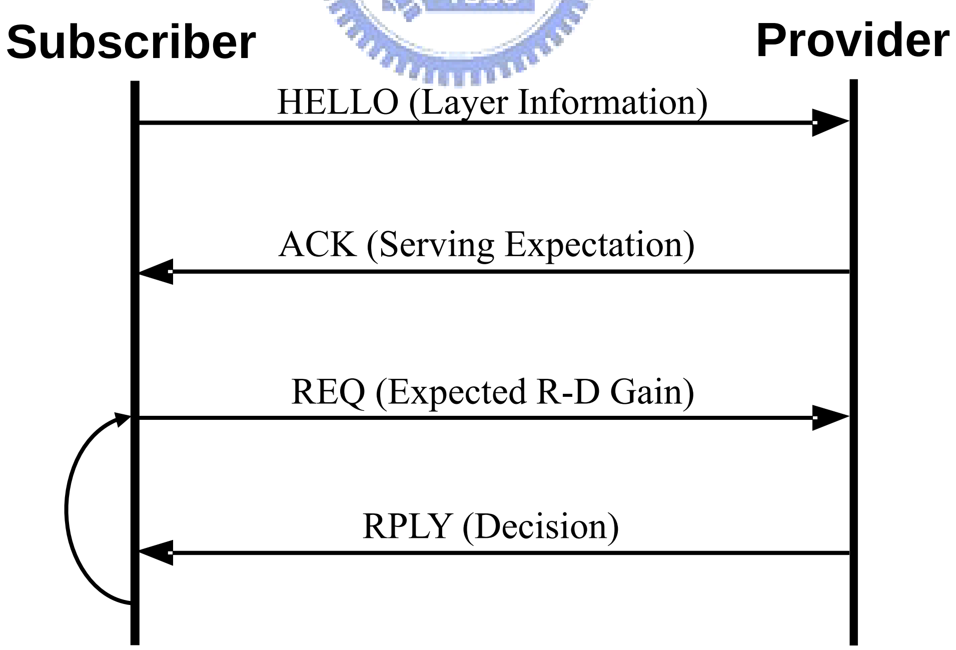



Figure 5: Message Exchange Mechanism

The message exchange in the bandwidth allocation and their payloads are listed below:

- *HELLO* is shown in (5). It is sent by every subscriber $\mathrm{MG}(i,j)$ , which represents the $j^{\mathrm{th}}$ MG in tier $i$. A HELLO message contains serving layer $\tilde{\mathrm{L}}$ , layer information tuples $\{(\mathrm{U}(\mathbb{L}), \mathrm{n}_{\mathrm{U}(\mathbb{L})}(i,j)\}_{\forall \mathrm{U}(\mathbb{L}), \tilde{\mathrm{L}} \in \mathbb{L}}$ which contains device types $(\mathrm{U}(\mathbb{L}))$ and sub-tree population number of a specific device type($\mathrm{n}_{\mathrm{U}(\mathbb{L})}(i,j)$) for all devices type that depend on $\tilde{\mathrm{L}}$, and the accessible provider counts $\xi(i,j)$ .

$$\mathrm{HELLO}(i,j) \vdash \tilde{\mathrm{L}}, \{(\mathrm{U}(\mathbb{L}), \mathrm{n}_{\mathrm{U}(\mathbb{L})}(i,j)\}_{\forall \mathrm{U}(\mathbb{L}), \tilde{\mathrm{L}} \in \mathbb{L}}, \xi(i,j) \tag{5}$$

- *ACK* is shown in (6). It is sent by provider $\mathrm{MG}(i+1,j)$ to every subscriber $\mathrm{MG}(i,j)$ that sent HELLO to it. A ACK message contains serving layer $\tilde{\mathrm{L}}$ , the amount of subscribers that can may send request to it for layer $\tilde{\mathrm{L}}$ as $\mathrm{RepCount}(i+1,j,\tilde{\mathrm{L}})$, and the expectation value $\mathrm{E}_{\mathrm{srv}}(i+1,j;i,j;\tilde{\mathrm{L}})$ about the number of layer $\tilde{\mathrm{L}}$ that can be allocated to $\mathrm{MG}(i,j)$.

$$\mathrm{ACK}(i+1,j;i,j) \vdash \tilde{\mathrm{L}}, \mathrm{RepCount}(i+1,j,\tilde{\mathrm{L}}), \mathrm{E}_{\mathrm{srv}}(i+1,j;i,j;\tilde{\mathrm{L}}) \tag{6}$$

- *REQ* is shown in (7). It is sent by subscriber $\mathrm{MG}(i,j)$ to a chosen provider $\mathrm{MG}(i+1,\hat{j})$ as (8) that it just received since the higher the value of $\mathrm{E}_{\mathrm{srv}}(i+1,j;i,j;\tilde{\mathrm{L}})$ is, the higher probability that it can get serving layer from provider is. A REQ message contains serving layer $\tilde{\mathrm{L}}$ , rest accessible provider counts $\xi(i,j)$ , and the expected R-D gain $\gamma(i,j,\tilde{\mathrm{L}})$.

$$\mathrm{REQ}(i,j;i+1,\hat{j}) \vdash \tilde{\mathrm{L}}, \xi(i,j), \gamma(i,j,\tilde{\mathrm{L}}) \tag{7}$$

$$\mathrm{MG}\,(i+1,\hat{j}) : \hat{j} = j \mid \max_{j} \mathrm{E}_{\mathrm{srv}}(i+1,j;i,j;\tilde{\mathrm{L}}) \tag{8}$$

- *RPLY* is shown in (9). It is sent by provider $\mathrm{MG}(i+1, j)$ to each subscriber $\mathrm{MG}(i, j)$ that sent REQ to it. It sent RPLY in the order of just received R-D gain $\gamma\left(i, j, \tilde{\mathrm{L}}\right)$ of each subscriber. Provider will allow every request until it has no available bandwidth to serve more requests. A RPLY message contains serving layer $\tilde{\mathrm{L}}$ and an answer R which is just a Boolean value about yes or no. Hence, true or false means the provider allowed or rejected.

$$\mathrm{RPLY}(i+1, j; i, j) \vdash \tilde{\mathrm{L}}, \mathrm{R} \tag{9}$$

If there is any subscriber $\mathrm{MG}(i, j)$ that received false value, it will decrease its accessible provider count $\xi(i, j)$ by 1 and then re-calculates its expected R-D Gain and sends request repeatedly to its second choice, third choice, etc.

# Chapter 6 Experiments

## 6.1 Platform

The simulation [Figure 6] is programmed in JAVA with different algorithm implemented in different node classes but in the same randomized network architecture in each execution of the program. The operation system that the simulation executed on is Fedora 8. The simulation will be executed in 100 times in each network congestion case to do some statistical computing while connection settings in the network varies randomly in each round.

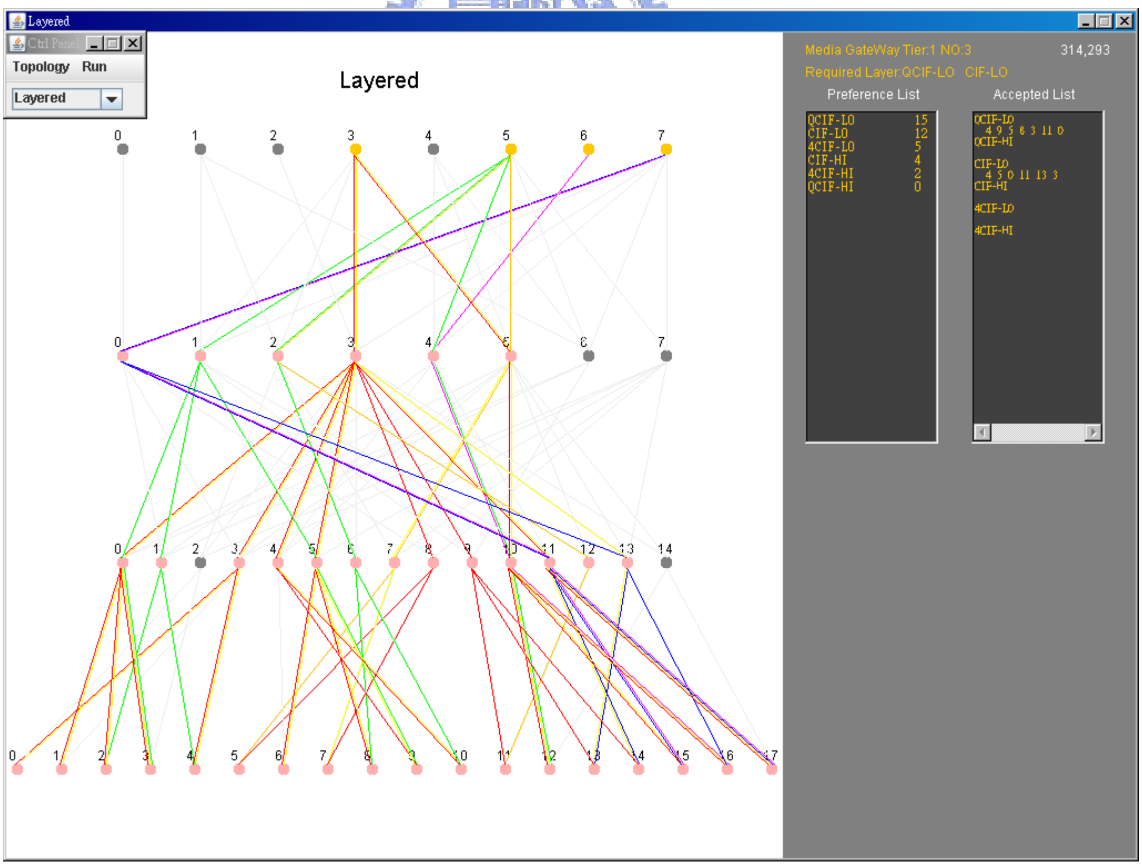



Figure 6: Graphic User Interface of Simulation for Bandwidth Reservation Protocol

## 6.2 Models

### 6.2.1 Network Model

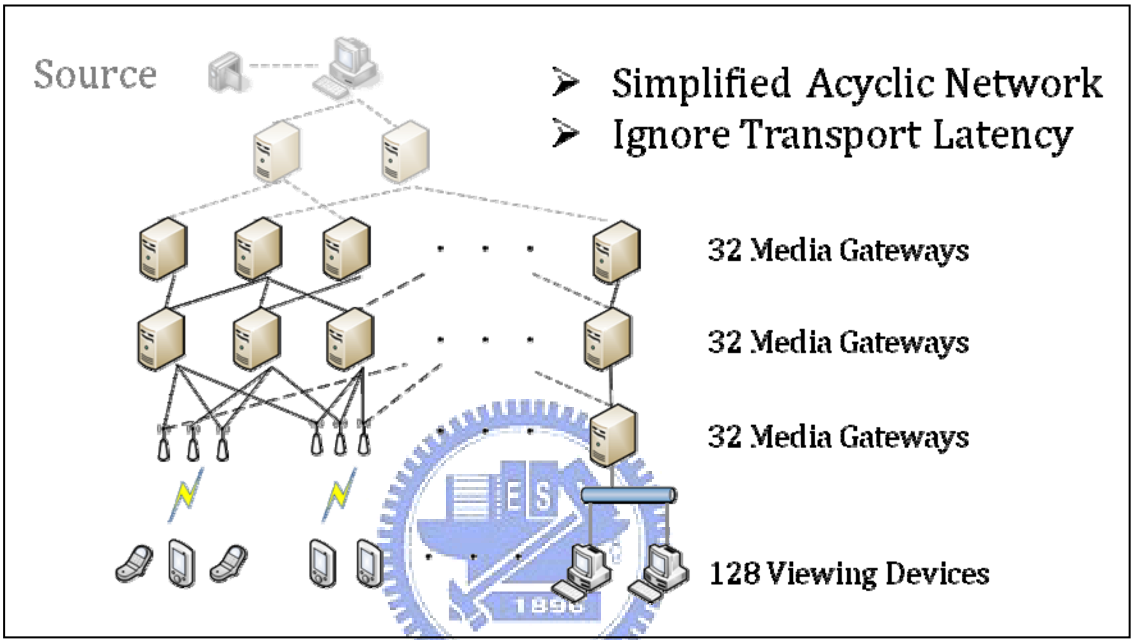



Figure 7: Network Model

In our simulation experiments, we devised a simple multicast mesh with three tiers as in [Figure 7], named as Tier 0, Tier 1, and Tier 2. User devices are placed at the leaf nodes of the mesh and the media gateways (MGs) serve as the intermediate nodes. There are a total of 128 user devices placed in eight Tier-0 local subnets. Hence, the MGs in Tier 1 broadcast their data flows while the MRs in Tier 2 relay data using unicast communication. Each MG has a restricted outbound bandwidth that it can use to serve its subscribers.

We implemented a randomized request mechanism in our simulation. When subscribers are making requests, they see those providers with no difference. So they just choose to send requests randomly with random layers that they require. We run experiments both on the mesh and tree network model.

Devices are subscribing to the clip and running the resource allocation on the network. We tune the Demand/Capacity Ratio as the network congestion factor. Demand is the total bitrates of the devices if they require AVC clips and Capacity are the total outbound bandwidth of the media relays in the local network. We run our experiments to figure out how our mechanisms run under such the network condition of AVC Simulcast.

### 6.2.2 Source Model

The movie clip we use is “crew” and its length is 10 seconds. We have a base layer and 5 enhancement layers, named QCIF-Low (Base Layer), QCIF-High, CIF-Low, CIF-High, 4CIF-Low, 4CIF-High (Enhancement Layers), and the dependency tree are as below. We choose the QP=6 and calculate the bit-rate of each layer by JSVM. The user devices may be 6 types as above, so they just subscribe layers based on its capability. Detail information is shown in [Table 6] and the dependency relationship is shown in [Figure 8].

| **SVC Layers**[2] | | QCIF-L | QCIF-H | CIF-L | CIF-H | 4CIF-L | 4CIF-H |
|---|---|---|---|---|---|---|---|
| **Reference Layers** | | - | QCIF-L | QCIF-L | CIF-L | CIF-L | 4CIF-L |
| **Single Layer Bit Rates**[3] | | 59 | 86 | 85 | 243 | 198 | 540 |
| **Cumulative Bit Rates** | | 59 | 145 | 144 | 287 | 342 | 882 |
| **User Device Counts** | | 6 | 14 | 11 | 19 | 25 | 53 |
| **PSNR**[4] | **QCIF** | 29.96 | 32.80 | - | - | - | - |
| | **CIF** | 28.34 | - | 29.94 | 33.34 | - | - |
| | **4CIF** | 27.61 | - | 28.95 | - | 30.13 | 33.27 |

Table 6: Characteristics of SVC test bitstream

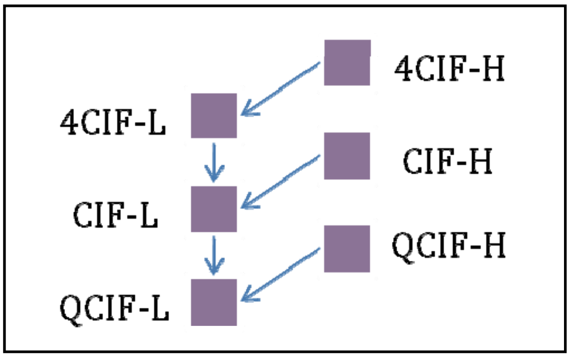



Figure 8: Dependency Relationship

[2] The SVC bitstream used in our simulation experiments was a ten-second clipping of the test video, *crew*.
[3] The unit of bit rate measurements is Kilo-bytes per second.
[4] The SVC bitstream was encoded using JSVM v.9 with a *Qp* value of six (6) between each adjacent layer.

# Chapter 7 Results and Analysis

Conforming to the comparative study conducted by T.H. Kim [3], we show the performance of our SVC multicasting scheme by displaying the values of four main parameters: PSNR reduction, transport efficiency, bandwidth consumption by individual SVC layer, and average media relay bandwidth utilization.

We run the experiment 100 times in fifteen network congestion cases from 60% to 200% which is the X-axis in all our diagrams. The error bars show the ranges of data between $10^{th}$ and $90^{th}$ percentiles in one hundred simulation rounds. In these figures, the red dash lines mark where the transport capacity of each MG equals to the total bit rate (1211 kbps) of the SVC bitstream. The blue dash lines mark where the MR capacity equals to the bit rate of the largest decodable bitstream (882 kbps of 4CIF-H layer).

We set the amounts 6, 14, 11, 19, 25, 53 for six different devices. The population distribution is quite different from the layer characteristic. There are many 4CIF devices but they are the heaviest load in the network, which means the largest bit rates but the least PSNR improvement.

We will show the results of our four different algorithms, a randomized algorithm, and two different AVC algorithms. The first one is the competing AVC algorithm. The behavior of our algorithm 1, 2, and 3 are all the same with AVC. All different layers of AVC compete in the network at the same time. And another one is the dominated AVC which is just like our algorithm 4. MGs pick a dominated AVC layer in the local subnet at once.

The colors for QCIF-L, QCIF-H, CIF-L, CIF-H, 4CIF-L and 4CIF-H are red, yellow, green, cyan, blue, and magenta in order.

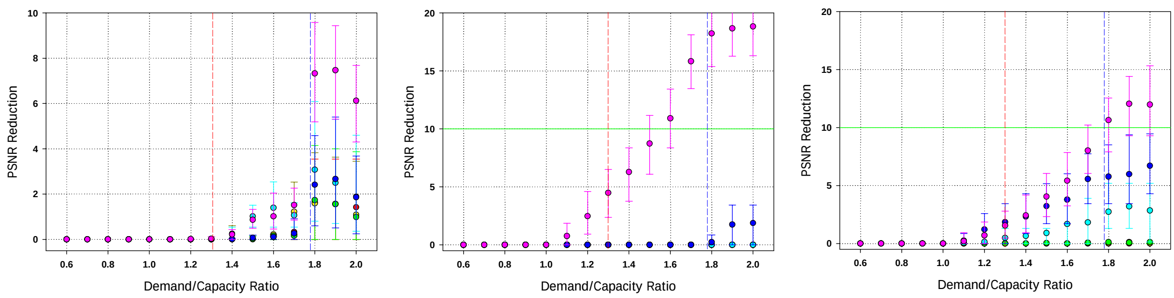
## 7.1 PSNR Reduction

PSNR Reduction of viewing devices is the most intuitive factor that users may concern. This measurement will show us the average distortion for each kind of devices. Compare [Figure 9] to other figures of our algorithms, we will find that what we have proposed perform much better than the natural randomized algorithm, the competing AVC, and the dominated AVC algorithm. For devices running randomized algorithm, all layers have probability to be dropped and the probability is proportion to the population distribution. For devices running AVC algorithms, there are only two results of any device, that is, received and not received. So the PSNR Reduction results shown in AVC algorithms vary from a large range. In competing AVC algorithm, all layers compete at the same time, so the 4CIF-L which has higher bit-rates but not as much PSNR improvement will be the worst one. But in dominated AVC algorithm, only one layer at once in local subnets cause 4CIF-H layer which has the largest population performs not so bad as competing AVC one. It shows that in some subnets other layers will be dropped to serve more 4CIF-H layers.

Under this measurement, flows with UEP approach performs a little bit worse than Layered SVC. It is because we have not prune the extra flows introduced by UEP even if devices subscribe successfully.

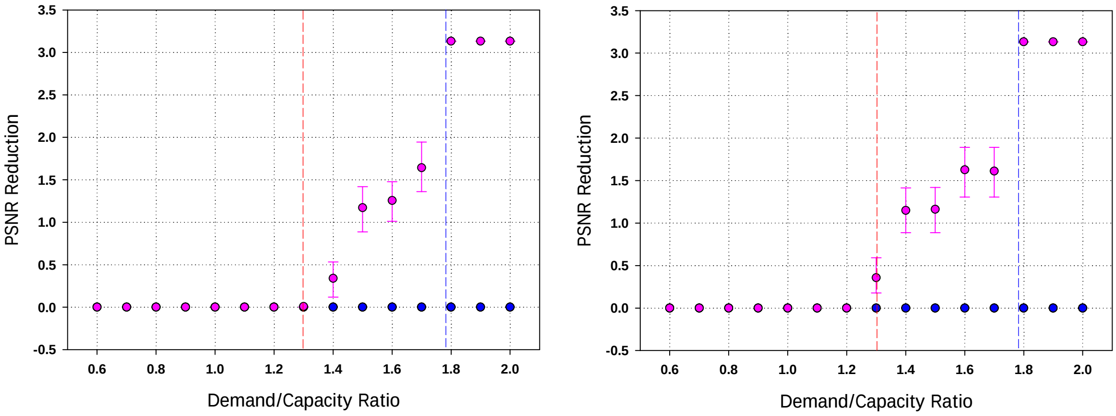
In [Figure 10] [Figure 12], the PSNR reduction is almost the same in either Layered SVC or Flows with UEP cases. It is because the characteristics of layers what we encoded cause that the order of algorithm 1 is same as algorithm 3. In algorithm 1, the order is {QCIF-L, CIF-L, QCIF-H, CIF-H, 4CIF-L, 4CIF-H}. And in algorithm 3, the only difference is that CIF-L and QCIF-L will compete at the same time and CIF-H and 4CIF-L will compete at the same time. Since the bandwidth is quite enough as they are competing, it makes almost no difference in these two algorithms. We start to drop 4CIF-H layers after the red dash line. It means that there are no any MG could serve all six layers at the same time. After blue dash line, we can no longer serve a complete 4CIF-H device. So they can at most receive a 4CIF-L bitstreams.

In [Figure 11], we compare algorithm 2 with algorithm 1. Since algorithm 2 will take population into consideration, so 4CIF-H devices' quality drops less but some CIF-H devices' quality drops. But after the blue dash line, all 4CIF-H layers could no longer be served, CIF-H will be served and 4CIF-H will be dropped. And the same results are shown in [Figure 13].



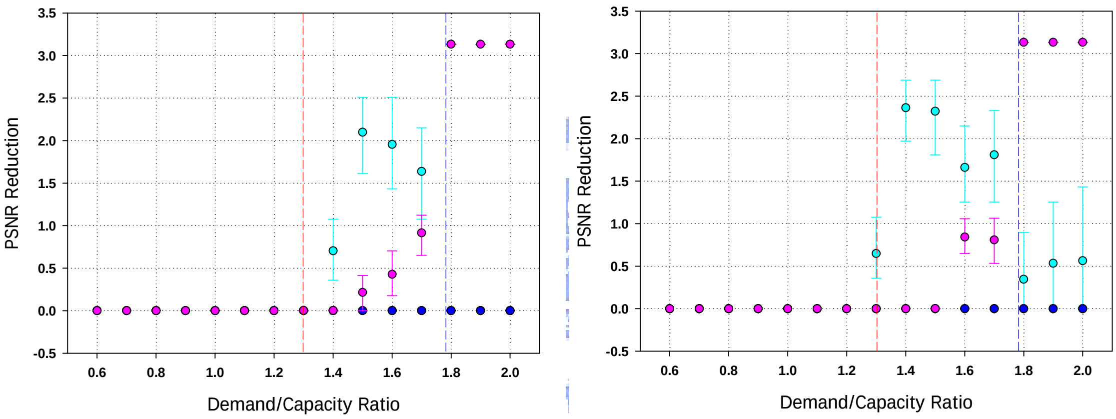
(a) PSNR Loss (Random SVC) (b) PSNR Loss (Layered AVC) (c) PSNR Loss (Layered AVC)

Figure 9: PSNR Reduction



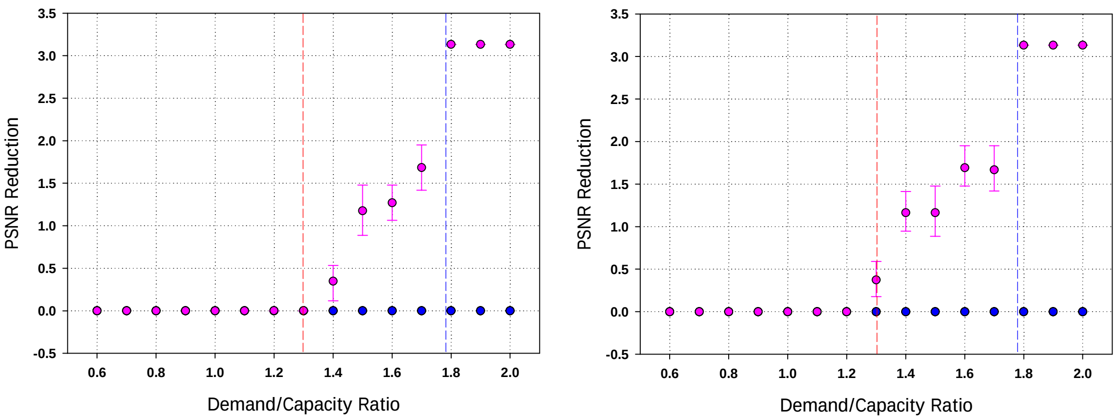

(a) PSNR Loss (Layered SVC) (b) PSNR Loss (Flow with UEP SVC)

Figure 10: PSNR Reduction of Algorithm 1



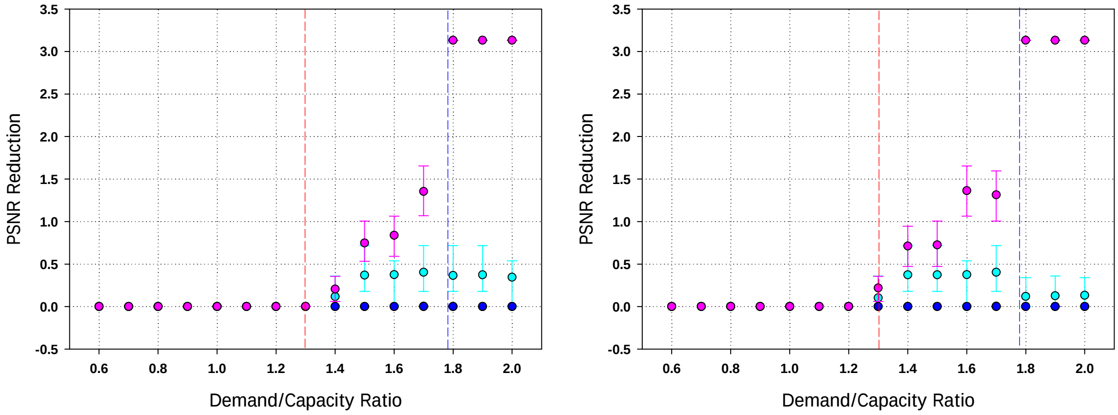

(a) PSNR Loss (Layered SVC) (b) PSNR Loss (Flow with UEP SVC)

Figure 11: PSNR Reduction of Algorithm 2




(a) PSNR Loss (Layered SVC) (b) PSNR Loss (Flow with UEP SVC)

Figure 12: PSNR Reduction of Algorithm 3

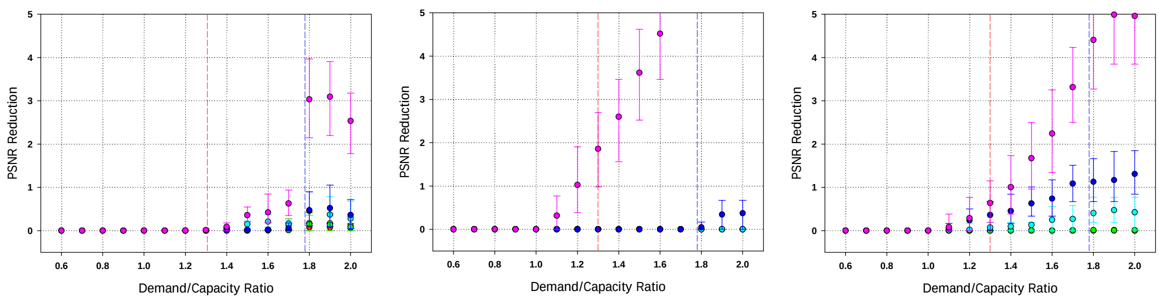



(a) PSNR Loss (Layered SVC) (b) PSNR Loss (Flow with UEP SVC)

Figure 13: PSNR Reduction of Algorithm 4

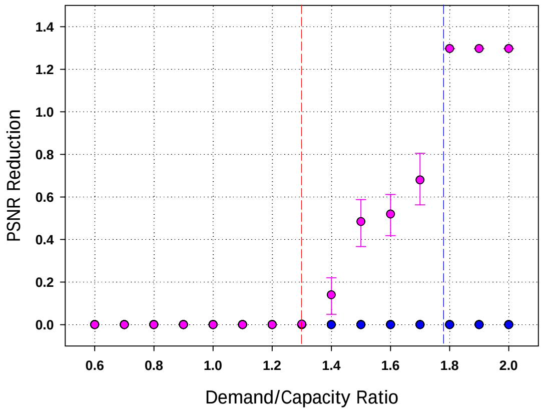
## 7.2 Normalized PSNR Reduction

The first measurement shows the playback quality of devices, but it can not show the influence about the population of viewing devices. PSNR loss means quite different in a large population from a small one. We multiply PSNR reduction by percentage of specific device's population in whole population. In [Figure 14] [Figure 15] [Figure 17] [Figure 18], the results almost show the same as the first measurement but we can see clearly in [Figure 16], normalized PSNR reduction in 4CIF-H is lower than other algorithms but almost the same as CIF-H. It shows how we compromise between layers.

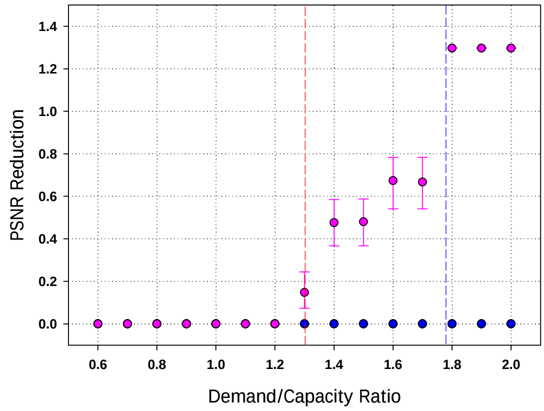



(a) PSNR Loss (Normalized, Random SVC) (b) PSNR Loss (Normalized, Layered AVC) (c) PSNR Loss (Normalized, Layered AVC)

Figure 14: Normalized PSNR Reduction

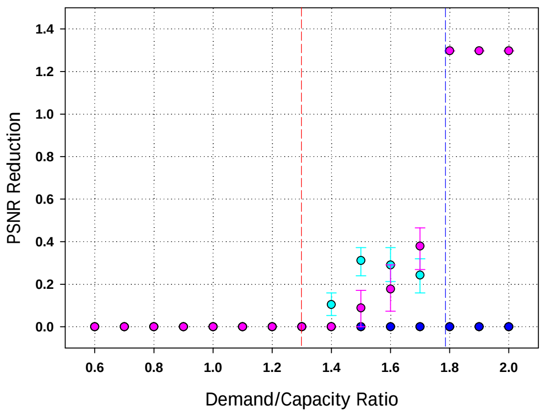



(a) PSNR Loss (Normalized, Layered SVC)

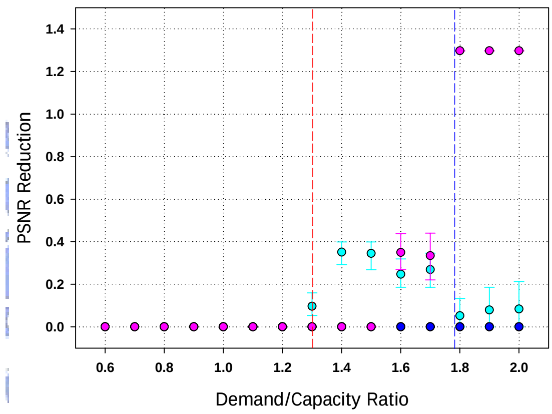



(b) PSNR Loss (Normalized, Flow with UEP SVC)

Figure 15: Normalized PSNR Reduction of Algorithm 1

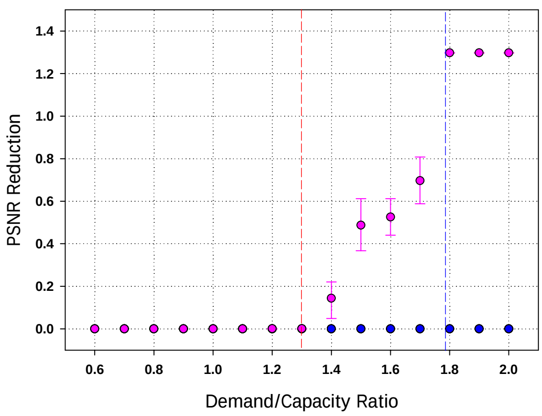



(a) PSNR Loss (Normalized, Layered SVC)

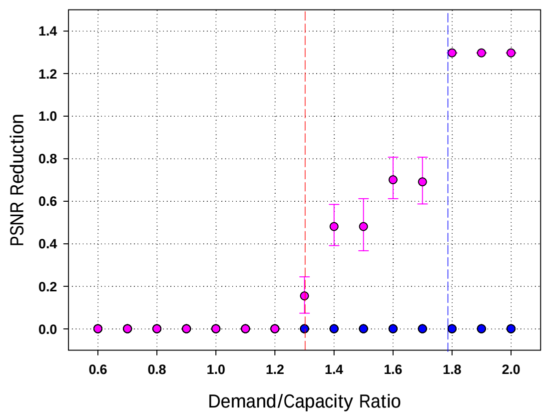



(b) PSNR Loss (Normalized, Flow with UEP SVC)

Figure 16: Normalized PSNR Reduction of Algorithm 2

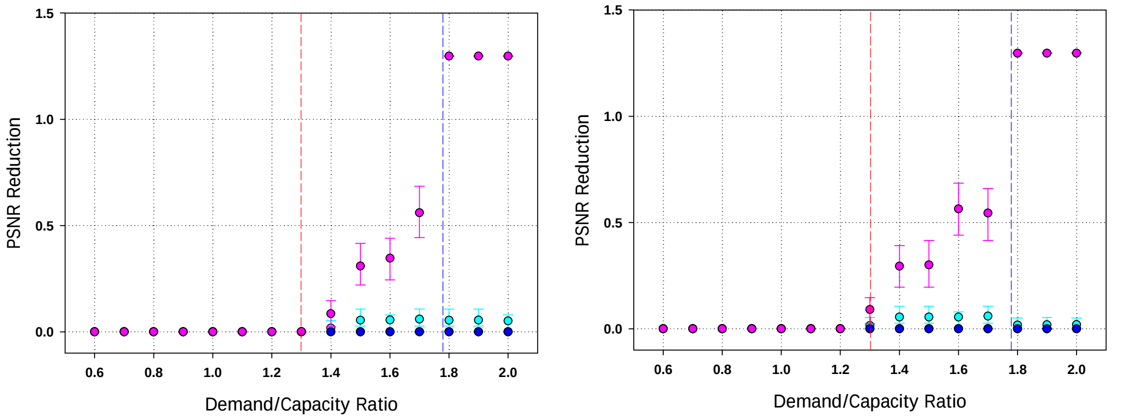



(a) PSNR Loss (Normalized, Layered SVC)




(b) PSNR Loss (Normalized, Flow with UEP SVC)

Figure 17: Normalized PSNR Reduction of Algorithm 3

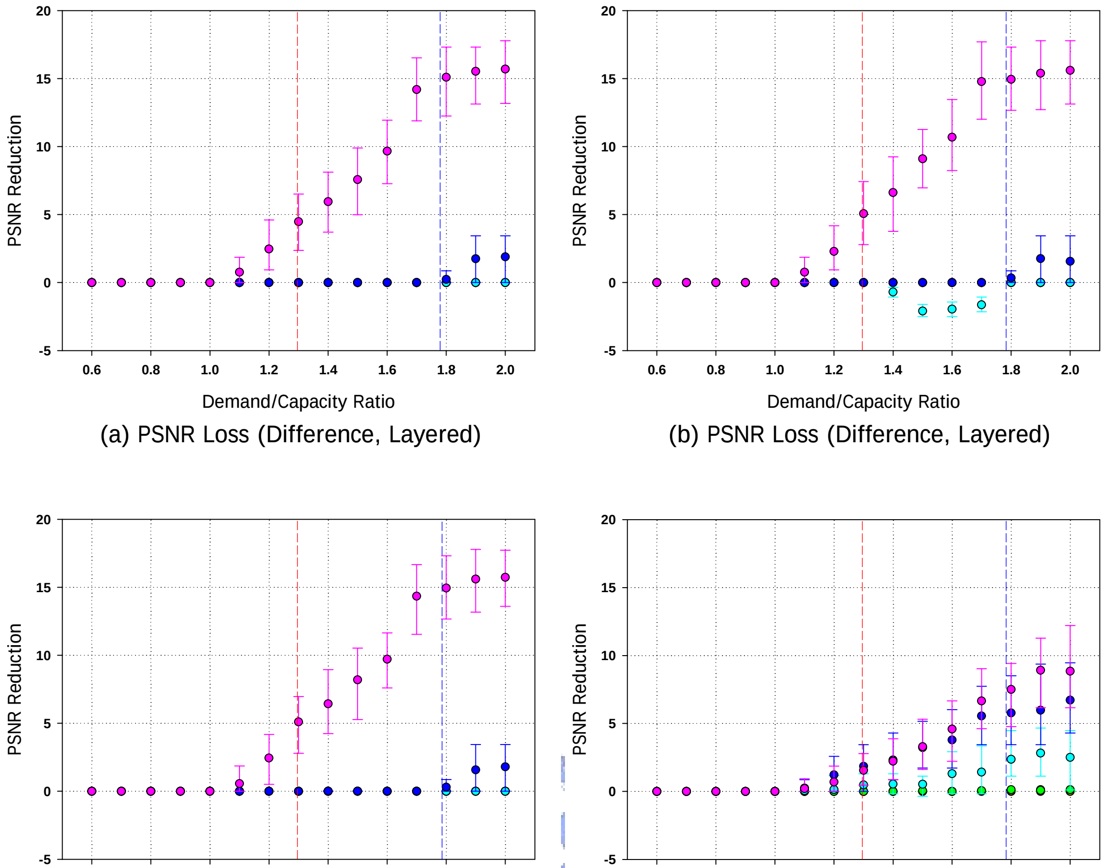



(a) PSNR Loss (Normalized, Layered SVC) (b) PSNR Loss (Normalized, Flow with UEP SVC)

Figure 18: Normalized PSNR Reduction of Algorithm 4

## 7.3 SVC vs. AVC

This measurement is going to show us the benefit we gain from using SVC rather than AVC. We can see clearly in [Figure 19].



(a) PSNR Loss (Difference, Layered)
(b) PSNR Loss (Difference, Layered)
(c) PSNR Loss (Difference, Layered)
(d) PSNR Loss (Difference, Layered)

Figure 19: Difference between SVC and AVC of four algorithms

## 7.4 Efficiency

*Efficiency* is going to show how the mechanism works. It is calculated by the total bitrates that devices receive divided by the total bitrates of traffic flow in every links on the network. Because of the broadcast at the bottom tier, the efficiency may be larger than 1. We measure the efficiency instead of the sum of bitrates on the network because if the user devices get less, the sum of bitrates on the network will be less relatively. To avoid misleading this, we measure the efficiency. Since the same reason, flows with UEP approach performs not as well as Layered SVC approach due to the extra flows.

The Efficiency is almost always at 1 even larger than 1 after the blue dash line. It is because 4CIF-H is the most inefficient. Layered SVC approach performs much better than other approaches in [Figure 20].

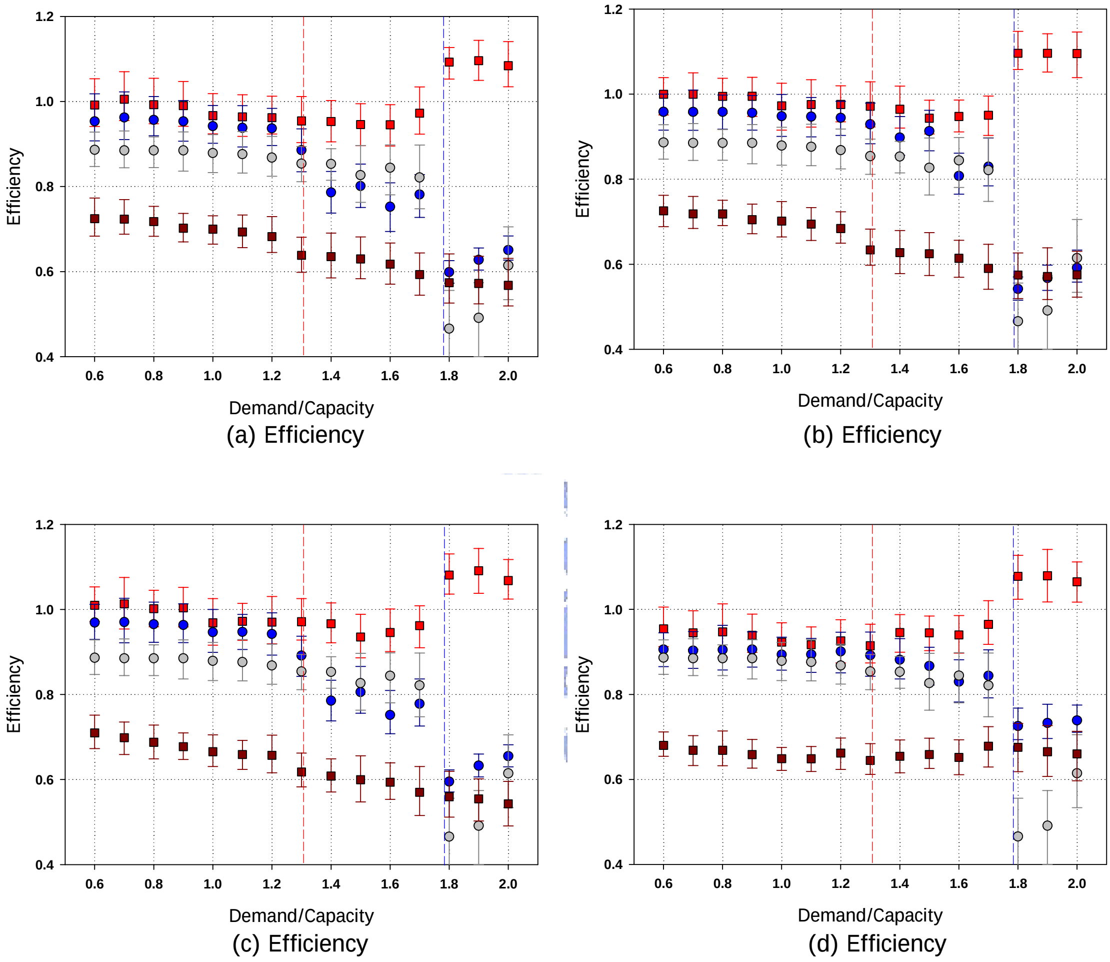



Figure 20: Efficiency for four algorithms

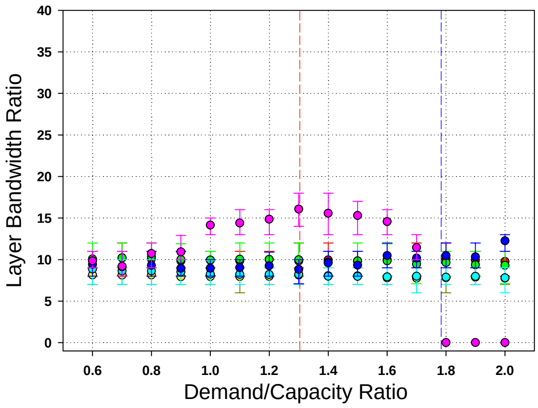
## 7.5 Layer Bandwidth Ratio

*Layer Bandwidth Ratio* is about the amount of each layer in each tier. This may show the aggregation of layers between tiers. The original demand amount of layers and amount of layers broadcasted is shown in [Table 7]. And in [Figure 21] [Figure 22] [Figure 23] [Figure

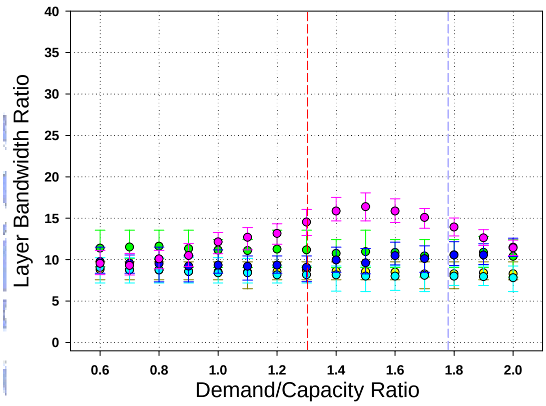
24], we show the amount of layers between the medium tier and the top tier. Layers in the network are reduced from 1/2 to 1/6.

| | QCIF-L | QCIF-H | CIF-L | CIF-H | 4CIF-L | 4CIF-H |
|---|---|---|---|---|---|---|
| Original Device Demand | 128 | 14 | 108 | 19 | 78 | 53 |
| Between Bottom & Medium Tier | 31.42 | 10.22 | 31.26 | 13.34 | 29.34 | 24.93 |

Table 7: Original Layer Counts and Layers Counts in lower tiers

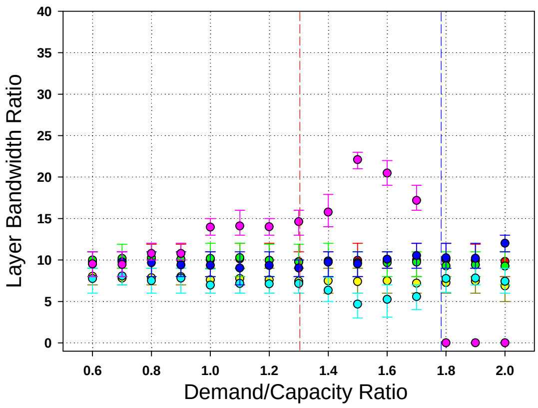


(a) Layer Bandwidth Ratio (Layered SVC)

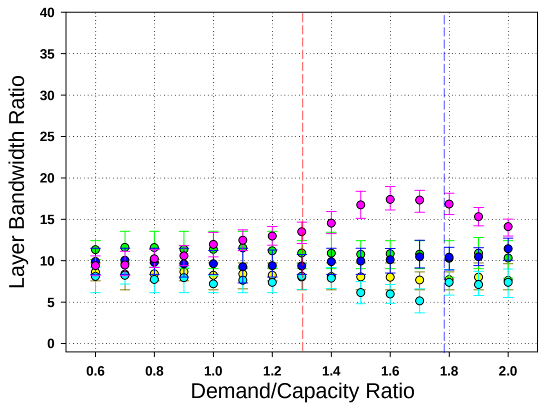


(b) Layer Bandwidth Ratio (Flow with UEP SVC)

Figure 21: Layer Bandwidth Ratio of Algorithm 1

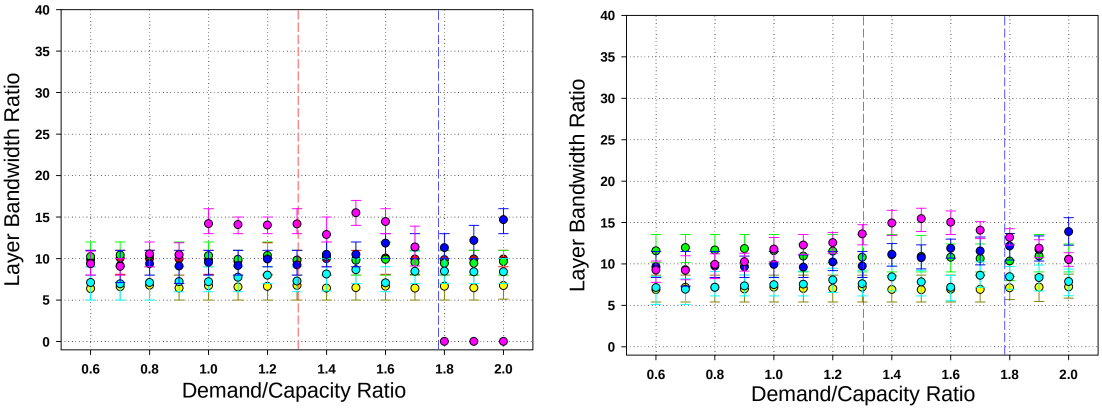


(a) Layer Bandwidth Ratio (Layered SVC)



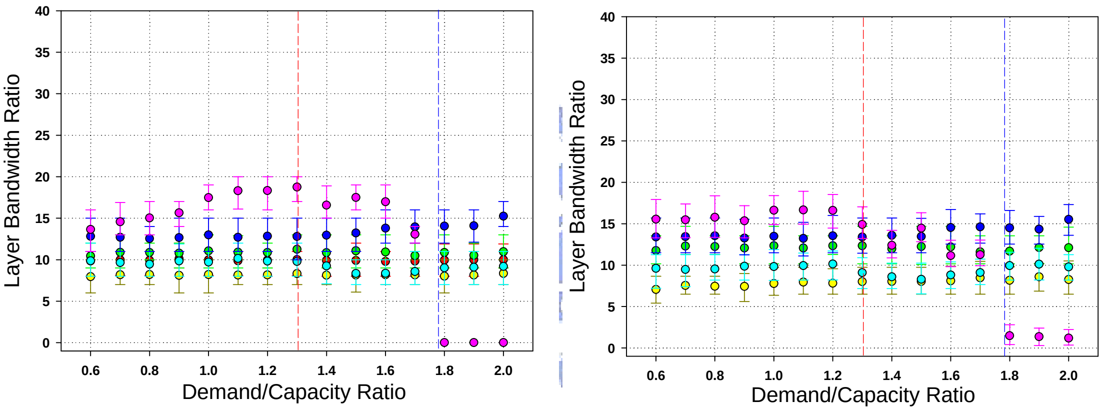
(b) Layer Bandwidth Ratio (Flow with UEP SVC)

Figure 22: Layer Bandwidth Ratio of Algorithm 2




(a) Layer Bandwidth Ratio (Layered SVC) (b) Layer Bandwidth Ratio (Flow with UEP SVC)

Figure 23: Layer Bandwidth Ratio of Algorithm 3




(a) Layer Bandwidth Ratio (Layered SVC) (b) Layer Bandwidth Ratio (Flow with UEP SVC)

Figure 24: Layer Bandwidth Ratio of Algorithm 4

## 7.6 Average Media Gateway Bandwidth Usage

*Average Media Gateway Bandwidth Usage* shows the load of MGs in each tier and we can figure out whether MGs are used well or not. Since some MGs may be unused because of the aggregation of layers, this measurement will tell us how we use the bandwidth in those used MGs (active MGs). Colors that represent top, medium, and bottom tiers are blue, green, and red. In [Figure 25] [Figure 26] [Figure 27] [Figure 28] [Figure 29], we can find that we use

MGs better than randomized algorithm and AVC algorithms. Use most bandwidth of MGs in the medium tier where aggregation happens most and the tier with heaviest load in the network. But we use less in the top tier due to the aggregation. Figures of flows with UEP approach show us how flows can almost be filled in all active MGs' capacity.

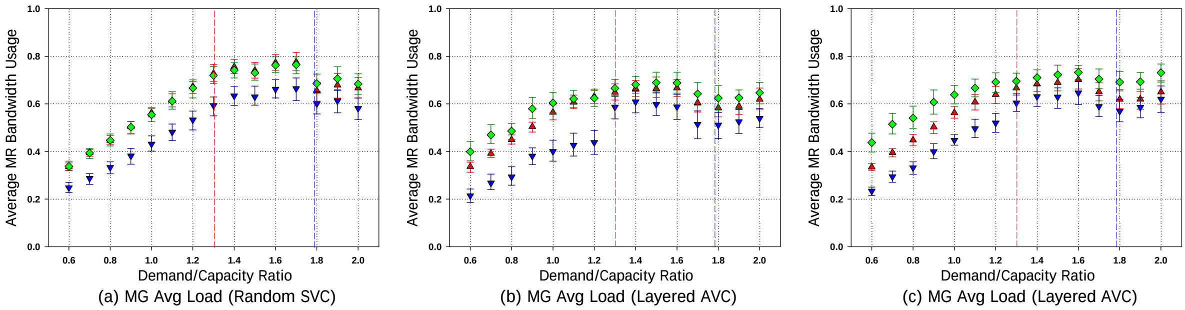



Figure 25: Average Media Gateway Bandwidth Usage of algorithms

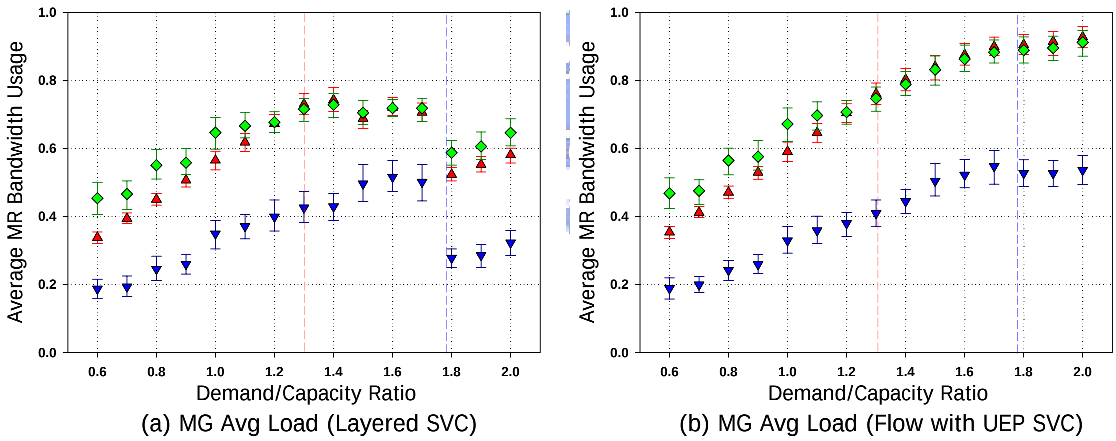
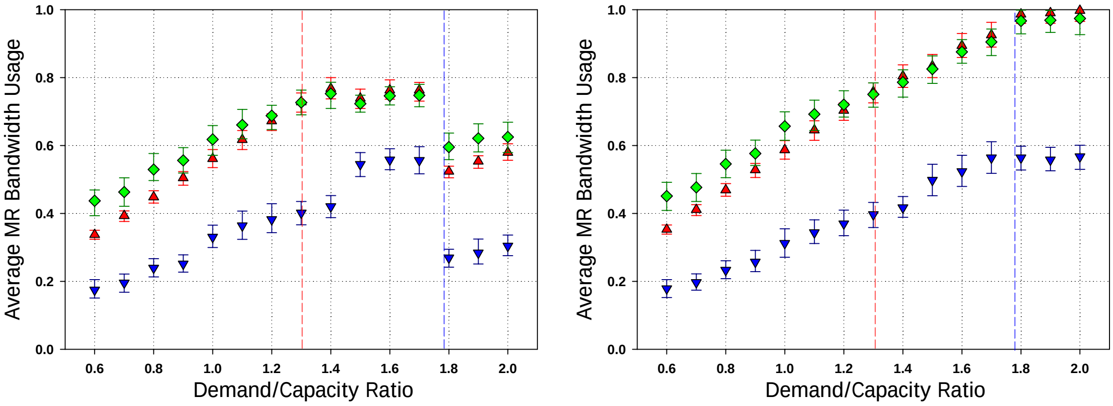
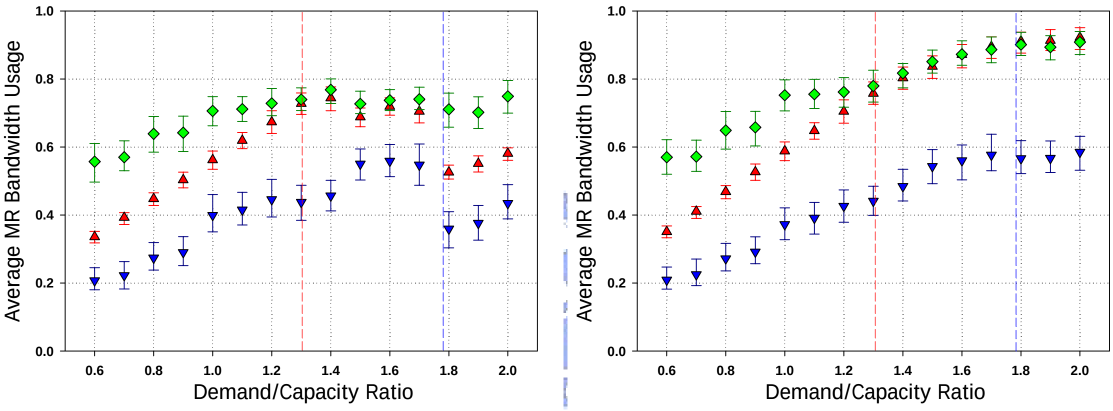



Figure 26: Average Media Gateway Bandwidth Usage of Algorithm 1



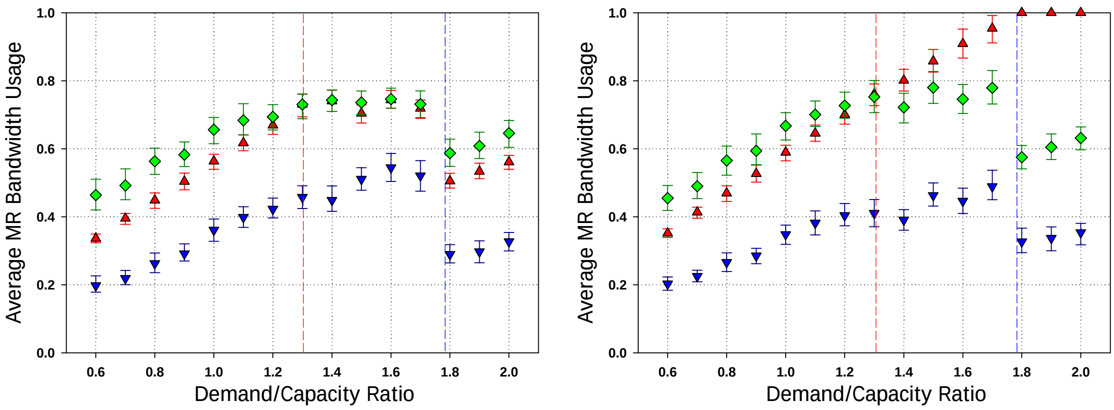

(a) MG Avg Load (Layered SVC) (b) MG Avg Load (Flow with UEP SVC)

Figure 27: Average Media Gateway Bandwidth Usage of Algorithm 2




(a) MG Avg Load (Layered SVC) (b) MG Avg Load (Flow with UEP SVC)

Figure 28: Average Media Gateway Bandwidth Usage of Algorithm 3




(a) MG Avg Load (Layered SVC) (b) MG Avg Load (Flow with UEP SVC)

Figure 29: Average Media Gateway Bandwidth Usage of Algorithm 4

# Chapter 8 Conclusion

## 8.1 Accomplishment

We proposed a new scheme for streaming video based upon SVC Multicasting. We can serve not only with better playback quality but also in an efficient way to transport. And we show that SVC performs better than AVC if we can transfer layers in a correct way.

We proposed an optimization model,

- **Rate-Distortion Gain**: To optimize R-D gain will make the multicasting have better playback quality and decrease the network bandwidth consumption.

We proposed a four way handshake bandwidth allocation protocol and implemented a complete simulation for our scheme and had results as following,

- Well performed in **Playback Quality**.
- Higher **Efficiency** for network bandwidth consumption.
- Do **Aggregation** in media gateways.
- Use media gateways' bandwidth well.
- SVC Multicasting performs much better than AVC Simucasting.

## 8.2 Future Work

We shall attempt to estimate the average performance of Rate-Distortion Gain our distributed bandwidth allocation algorithm using stochastic models and devise modification to the proposed algorithms in order to avoid local optima.

We will consider not only bandwidth but also latency and error in the future and implement it on a network simulator such as OMNeT++.

# Glossary

| Symbols | Definitions |
|---|---|
| L | *Layer ID* as concatenation of *Dependency ID*, *Quality ID* and *Temporal ID* of SVC NAL Units |
| $U(\mathbb{L})$ | Type of User Device with target SVC layers set $\mathbb{L}$. |
| $n_{U(\mathbb{L})}$ | Number of Devices $U(\mathbb{L})$ in the network |
| $n_{U(\mathbb{L})}(i,j)$ | Number of Devices $U(\mathbb{L})$ in the sub-tree whose root is $MG(i,j)$ |
| T | Total Tier Numbers. |
| $MG(i,j)$ | $j^{th}$ Media Gateway in Tier i |
| $D(j)$ | $j^{th}$ User Device (as a leaf of SVC multicast tree) |
| $\xi(i,j)$ | Fan In of $MG(i,j)$ |
| $RepCount(i,j,\tilde{L})$ | Replication Count of Layer $\tilde{L}$ at $MG(i,j)$ |
| $LayerCount_{Allowed}(i,j,\tilde{L})$ | Allowed Layer Count of Layer $\tilde{L}$ at $MG(i,j)$ |
| $\mathbb{N}(i,j)$ | Children Set of MGs that is accessible to $MG(i,j)$ |
| $\tilde{L}$ | *Layer ID* of Serving Layer |
| $\mathcal{D}(\tilde{L})$ | Device Type Set of all device types that need $\tilde{L}$ |
| $\mathcal{L}(D(j))$ | Set of all layers that $D(j)$ needs |
| $d(U(\mathbb{L}),\tilde{L})$ | PSNR Improvement of Device Type $U(\mathbb{L})$ receiving Layer $\tilde{L}$ |
| $d(D(j),\tilde{L})$ | PSNR Improvement of Device $D(j)$ receiving Layer $\tilde{L}$ |
| $r(\tilde{L})$ | Bit rate of Layer $\tilde{L}$ |
| $\gamma(i,j,\tilde{L})$ | Rate-Distortion Gain of $MG(i,j)$ receiving Layer $\tilde{L}$ |
| $\gamma(\tilde{L})$ | Rate-Distortion Gain of allocating Layer $\tilde{L}$ |
| $\gamma(D(j),\tilde{L})$ | Rate-Distortion Gain of $D(j)$ receiving Layer $\tilde{L}$ |
| $\gamma_{Total}$ | Total Rate-Distortion Gain |
| $E_{srv}(i+1,j;i,j;\tilde{L})$ | Expectation Value of number of layer $\tilde{L}$ that subscriber $MG(i,j)$ could get from provider $MG(i+1,j)$. |

# References

[1] T. Wiegand, G. Sllivan, J. Reichel, H. Schwarz, M.Wien. "Joint Draft ITUT Rec.H.264 – ISO/IEC 14496-10/ Amd.3 Scalable Video Coding," *ISO/IEC/JTCI/SC29/ WG-11 and ITU-T SG16 Q.6, JVT-X201*, July 2007.

[2] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the Scalable Video Coding Extension of the H.264/AVC Standard," *IEEE TRANSACTIONS on CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, VOL. 17, NO. 9*, September 2007.

[3] T. H. Kim, "Scalable Video Streaming over Internet". *PhD Thesis*, School of Electrical and Computer En-gineering, Georgia Institute of Technology, January 2005

[4] S. Y. Cheung, M. H. Ammar, and X. Li, "On the use of destination set grouping to improve fairness in multicast video distribution," *IEEE INFOCOM '96*, San Francisco, CA, Mar. 1996.

[5] T. Jiang, M. H. Ammar, and E. W. Zegura, "Inter-receiver fairness: a novel performance measure for multicast ABR sessions," *ACM SIGMETRICS ' 98*, Madison, WI, June 1998.

[6] X. Li, S. Paul, and M. H. Ammar, "Layered video multicast with retransmission (LVMR): Evaluation of hierarchical rate control," *IEEE INFOCOM '98*, San Francisco, CA, Mar. 1998.

[7] S. McCanne, V. Jacobson, and M. Vetterli, "Receiver driven layered multicast," *ACM SIGCOMM '96*, Stanford, CA, Aug. 1996.

[8] J. Byers, M. Luby, and M. Mitzenmacher, "Fine-grained layered multicast," *IEEE INFOCOM 2001*, Anchorge, AK, Apr. 2001.

[9] V. K. Goyal, J. Ko˘cevi′c, R. Arean, and M. Vetterli, "Multiple description transform coding of images," *ICIP '98*, Chicago, IL, Oct. 1998.

[10] P. Baccichet, T. Schierl, T. Wiegand, B. Girod. "Low Delay Peer-to-Peer Streaming using Scalable Video Coding". *Proc. IEEE International Conference on Image Processing (ICIP) 2007, pp. 1-22*, 2007.

[11] S. Wee, J. Apostolopoulos, W. Tan, S. Roy. “Research and Design of a Mobile Streaming Media Content Delivery Network”. *Proc. IEEE ICME* 2003.

[12] M. Castro, P. Druschel, A.-M. Kermarrec, A. Nandi, A. Rowstron, A. Singh, “SplitStream: High-bandwidth content distribution in a cooperative environment,” *Proc. IPTPS’03*, Berkeley, CA, Feburary 2003.

[13] Y. H. Guo, J. K. Zao, W. H. Peng, L. S. Huang, C. M. Lin, F. P. Kuo. “*Trickle*: Resilient Real-Time Video Multicasting for Dynamic Peers with Limited or Asymmetric Network Connectivity”. *Proc. IEEE Int’l Symposium on Multimedia (ISM)*, San Diego, CA, December 2006.

[14] A. Segall. “SVC-to-AVC Bit-Stream Rewriting for Coarse Grain Scalability”. *Joint Video Team, doc. JVT-V035*, Marrakech, Morocco, January 2007.