Dynamic Layering and Bandwidth Allocation for Multi-Session Video Broadcasting with General Utility Functions

Jiangchuan Liu [1] Bo Li [1] Y. Thomas Hou [2] Imrich Chlamtac [3]

csljc@cs.ust.hk bli@cs.ust.hk thou@vt.edu chlamtac@utdallas.edu

[1] Department of Computer Science, The Hong Kong University of Science and Technology

[2] The Bradley Department of Electrical and Computer Engineering, Virginia Tech, Blacksburg, VA, USA

[3] Erik Jonsson School of Engineering and Computer Science, University of Texas at Dallas, USA

Abstract - **For video broadcasting applications in a wireless environment, layered transmission is an effective approach to support heterogeneous receivers with varying bandwidth requirements. There are several important issues that need to be addressed for such layered video broadcasting systems. At the session level, it is not clear how to allocate bandwidth resources among competing video sessions. For a session with a given bandwidth, questions such as how to set up the video layering structure (*i.e.,* number of layers) and how much bandwidth should be allocated to each layer remain to be answered. The solutions to these questions are further complicated by practical issues such as *uneven* popularity among video sessions and video layering overhead. This paper presents a systematic study to address these issues for a layered video broadcasting system in a wireless environment. Our approach is to employ a generic utility function for each receiver under each video session. We cast the joint problem of layering and bandwidth allocation (among sessions and layers) into an optimization problem of total system utility among all the receivers. By using a simple 2-step decomposition of inter-session and intra-session optimization, we derive efficient algorithms to solve the optimal layering and bandwidth allocation problem. Practical issues for deploying the optimal algorithm in wireless networks are also discussed. Simulation results show that the optimal layering and bandwidth allocation improves the total system utility.**

I. Introduction

With the proliferation of web-based services and rapid growth of wireless communication devices, layered video broadcasting is becoming an important multimedia application. An important advantage associated with layered video for such broadcast applications is that diverse user access devices can be easily supported – devices (*e.g.,* cellular phone, PDA, laptop) with varying bandwidth and processing capability have the option to subscribe to an appropriate number of layers of a video program (or session) to meet their unique requirements and physical constraints. Hence, a single video session with multiple layers can simultaneously accommodate a group of users with different capacity requirements. As an example, under the *cumulative layered transmission* [2,3], a raw video is compressed into several layers. The most significant layer, called the *base layer*, contains the data representing the most important features of the video, while additional layers, called *enhancement layers*, contain data that progressively refine the reconstructed video quality. The layers are then distributed to receivers via broadcast channels by a layered transport protocol.

Recent advances in video coding have made it possible to encode video with a very flexible layering structure [12]. In such coders, both the bandwidth of a layer and the number of layers can be dynamically manipulated with a fast response time. In particular, advanced video streaming standards such as the MPEG-4 Delivery Multimedia Integration Framework (DMIF) [1] are capable of performing fast layer stream setup and termination at a very low cost. Such flexibility in video coding has enabled further opportunity to deliver video contents with much improved efficiency and performance.

There are several important issues that remain to be addressed for the delivery of layered video in a broadcast environment. First, we need a bandwidth allocation mechanism to allocate bandwidth among video sessions (or programs). In a wireless environment, the total bandwidth is a constrained resource that is shared among competing video sessions. A straightforward approach is to share the total system bandwidth equally among all the sessions. Such an approach however is not advisable since each session is of different significance and should be treated differently in terms of bandwidth allocation. For example, a popular video session attracting a large number of receivers should be allocated with more bandwidth resources (consequently providing better perceptual quality and more revenue) than a session with few receivers.

Second, for a video session under a given bandwidth budget, it is not clear how the layering structure for this session should be organized. In particular, questions such as how many layers should be generated for this video session and how much bandwidth should be allocated for each layer remain to be answered. There are several practical issues that need to be considered when addressing the above questions. The first is the *layering overhead.* Under a given session bandwidth, increasing the number of layers means smaller bandwidth for each layer, and hence finer adaptation granularity on the

Bo Li's research was supported in part by grants from Research Grant Council (RGC) under contracts AoE/E-01/99 and HKUST6196/02E. J. Liu's work is partially supported by a Microsoft fellowship.

receiver's side. The drawback here is that more layers will bring more overhead (for both coding and transport), which diminishes the benefits from the improved granularity in adding more layers [4,12]. Another issue is that, under a typical wireless broadcast environment, receivers' capacities generally exhibit some kind of *clustered* distribution instead of uniform distribution. This is because receivers usually use some standard access interfaces. Therefore, if the bandwidth allocation among the layers can explore this property, the mismatch of bandwidth between a receiver's capacity and the layers can be reduced, which translates into better performance at the receiver's end.

This paper presents a systematic study to address the layering and bandwidth allocation (among video sessions and layers) for video broadcasting. Our study explores the flexible and dynamic property of advanced video encoders at the source side to meet the diverse requirements from the receivers. We introduce a generic utility function for each receiver, which takes into account the receiver's physical capacity, actual received bandwidth, and number of received layers. The utility function is designed to be general enough to accommodate various performance measures, *e.g.,* throughput, video's perceptual quality, user satisfaction and fairness. We show that the layering and bandwidth allocation problem can be formulated into an optimization problem of maximizing the total system utilization, which is a sum of the utilities among all the receivers in the system. By using a simple 2-step decomposition of inter-session and intra-session allocation, we derive computationally efficient (polynomial time) algorithms for both inter-session and intra-session optimization problems. Furthermore, we address some important issues in practice and demonstrate that the optimal allocation algorithm can be implemented with existing layered video coders, where both the computation overhead and deployment complexity are kept at low levels.

To investigate the performance of our optimization algorithms, we conduct simulations under various settings. Our results offer some valuable insights on several key factors such as layering overheads, perceptual video quality, and receiver capacity distribution, and provide some guidelines for the design of layering structure for video broadcasting.

We organize the rest of the paper as follows. In Section II, we describe the system model and introduce the notion of utility function for our investigation. Section III formulates the problem of optimal layering and bandwidth allocation for video broadcasting. We also derive computationally efficient algorithms to solve the problem. Section IV discusses some implementation and computation issues. In Section V, we present numerical results to demonstrate the performance of our optimization algorithms. Finally, Section VI presents some related work and Section VII concludes this paper.

II. System Modeling and Utility Function

In this section, we describe the system model for our investigation of optimal video layering and bandwidth allocation in a broadcasting environment. We also introduce the notion of a utility function for each receiver, which serves as a primary metric in our overall system optimization.

A. System Model

As suggested in [13], we mainly focus on the adaptation in a wireless local loop or an individual cell in a cellular network. This is because, in current networks, the wireless link bandwidth is much more valuable than the bandwidth of wired links, and thus becomes a dominant factor in overall system optimization. We also present some discussions on multi-cell adaptation in Section IV.

In such a wireless broadcast system, there is a central access point (base station or mobile switching center). A set of video programs (called *sessions*), $\boldsymbol{S}$, are simultaneously distributed to the receivers from this central point, which assigns a total bandwidth to all the video sessions. For each session, the encoded video is further partitioned into multiple layers. As an abstraction, we assume that the video encoders are located inside the central point and all bandwidth allocation and layering operations are performed at this point. We will show later that, with advanced scalable video codecs, generating a compressed video stream and partitioning it into layers can be implemented in two steps: only some simple assembling algorithms need to be implemented in the central point for video layering while the computation-intensive operations for video coding can be located elsewhere. Moreover, the layering operation in this case can even be applied to pre-stored video streams.

The central broadcast point also performs management functions such as user registration and authentication. Moreover, a video program (session) guide is sent to all receivers via a dedicated broadcast channel. A receiver who is interested in a particular video session should first send a request to the central point along with a description of its capability (*i.e.,* access capacity). Upon admission into a video session, the receiver will subscribe to a set of cumulative layers (starting from the base layer) commensurate with its capacity. Note that a receiver cannot subscribe to a fraction of a layer. The adaptation granularity on the receiver's side is thus at the layer level, which could result in a mismatch between a receiver's capacity and the video layers if the number of layers is limited. On the other hand, since each video layer is associated with an encoding and transport overhead, for a total session bandwidth budget, increasing the number of layers will lead to bandwidth inefficiency in encoding. [1]

As mentioned earlier, the capacity of the receivers in a network typically follows some clustered distribution. Thus, we let the central point be adaptive in setting the layering structure and bandwidth allocation so as to exploit the clustering property of the receivers' capacity distribution. Specifically, this scheme dynamically determines the number of layers for each session and allocates the bandwidth among sessions as well as layers within a session to maximize the system's performance. This is a viable approach since the central point has complete knowledge of the capacity constraint for each receiver in each session.

In our system model, we use *channel* as the basic unit for bandwidth allocation. A channel in a wireless system

[1] The overhead depends on the number of layers as well as the bandwidth of each layer [4,12].

represents a fixed unit for data transmission, *e.g.,* a time slot in TDMA systems, a frequency in FDMA systems, or a logical allocation unit such as the logical channel in WCDMA [1]. We further assume that each video layer can occupy only an integral number of channels, and a receiver's capacity is also expressed in the number of channels.

B. *Utility Function*

A challenging issue for multi-session video broadcasting is *heterogeneity*. First, each receiver has a different capacity, which imposes an upper bound of the video bandwidth it can subscribe to. Second, each video session enjoys different popularity and should thus be treated differently in bandwidth allocation. For example, a video session showing a newly released movie attracting a large number of receivers clearly should have preferential treatment in bandwidth allocation than another less popular video session with few receivers, even if both sessions use similar video coding format. To quantify such heterogeneity among receivers and video sessions, we introduce the notion of *utility function* for each receiver. The total system utility is the sum of the utilities among all the receivers for all the sessions in the system.

There are two categories of utility functions used in the literature. One category can be called "absolute" utility — referring to performance metrics being directly used as the utility function. For example, the video bandwidth delivered to the receiver [9], or the video quality perceived by the receiver, which can be measured by the Mean-Opinion-Score (MOS) or the Peak Signal-to-Noise Ratio (PSNR). In general, an absolute utility function for a given video content is a function of the video bandwidth and, for layered coding, depends on the number of layers delivered to the receiver [12]. The other category can be called "relative" utility, which is a "transformed" metric to represent a receiver's satisfactory given its expectation. A relative utility not only depends on the bandwidth and number of layers delivered to the receiver, but also its expected bandwidth, or its own capacity. For example, a typical relative utility function called Inter-Receiver Fairness (IRF) is given by the actual received bandwidth at a receiver normalized with respect to its capacity [10].

How a utility function should be exactly defined remains a debatable issue. The choice can actually depend on a number of factors (*e.g.,* encoding and transmission algorithms) and more important, the design objective of the system. For example, from a network provider's perspective, an absolute utility function, such as throughput, is preferable if the revenue is proportional to the total received bandwidth of all the receivers. But such an absolute utility tends to favor broadband receivers over those narrowband receivers. For the latter, a relative utility function seems more suitable from a receiver's perspective.

Instead of limiting our scope to a specific absolute or relative utility, we introduce a generic utility function which takes into account of several essential parameters for layered video applications. We define the utility for a particular receiver subscribing a video session *j,* denoted as $\mu_j(k,r,l)$, to be a function of the receiver's capacity *k*, its actual received video bandwidth *r*, and the total number of its subscribed layers *l.* There are several important advantages of the above framework for utility functions. First, by taking *k*, *r*, and *l* as parameters, our framework can accommodate both absolute and relative utility functions for layered video. For example, an absolute utility that characterizes the perceptual video quality for a receiver can be denoted as $\mu_j(k,r,l) = Q(r,l)$, where *Q*(*r*,*l*) is a mapping from the layering structure to the perceptual video quality. On the other hand, an extension of IRF, the *Application-aware Fairness Index* (AFI), has the form of $\mu_j(k,r,l) = Q(r,l)/Q(k,1)$ [11], which normalizes the receiver's perceived video quality with respect to its maximum expected quality (a single layer video with the bandwidth being equal to the receiver's capacity). Second, since the rate-quality relation for video compression depends on the video sequence and the video encoder, it is very difficult to characterize such a relation by using a closed form function for complex video coders (particularly for a layered video coder). Our utility function does not require such explicit characterization. It takes only discrete parameters that are observable from network services. Furthermore, our optimal allocation algorithms do not put any continuity or differentiable requirements on the utility function. As a result, only some sampled points of the function need to be calculated by the layered coder, and a simple table-search algorithm for the pre-stored values is sufficient for our optimal allocation algorithms.

We now have a 4-tuple, $(N, \boldsymbol{S}, M_{j,k}, \mu_j)$ for our system model, where *N* is the total number of available channels for our system, $\boldsymbol{S}$ is the set of the sessions (sharing the total bandwidth *N*), $M_{j,k}$ is the number of receivers with a capacity of *k* channels in session $j \in \boldsymbol{S}$. The problem to solve is to find an appropriate layering structure for each session as well as bandwidth allocation among sessions and layers such that the total system utility is maximized.

III. Optimal Layering and Bandwidth Allocation

In this section, we formally describe the utility-based optimization problem for video layering and bandwidth allocation. We also develop efficient polynomial time algorithms to solve this optimization problem.

A. *Mathematical Formulation*

Denote R_j the *layer allocation vector* for video session *j*, $R_j = (r_j^1, r_j^2, ..., r_j^{L_j})$, where L_j is the total number of layers of the allocation vector, and r_j^i the cumulative bandwidth up to layer *i*. Under a given allocation vector for a session, a receiver shall subscribe to as many layers as possible, subject to its access capacity. That is, a receiver in session *j* with a capacity of *k* channels should subscribe to layers 1, 2,…, $l_{j,k}^*$, where $l_{j,k}^* = \max\ \{\ l \mid r_j^l \le k\ \}$. In this case, the cumulative subscription bandwidth for the receiver is $r_{j,k}^* = r_j^{l_{j,k}^*}$, and its utility is thus $\mu_j(k, r_{j,k}^*, l_{j,k}^*)$.

Let system utility be the sum of the utilities among all the receivers in the system. Our objective is to achieve the maximum system utility by properly choosing a layering structure, *i.e.,* the number of layers for each session, and allocating the total bandwidth among the sessions and layers. The notations for this optimal layering and allocation problem are summarized in Fig. 1.

V : total number of available channels for the system;

$\boldsymbol{S}$: the set of video sessions in the system;

$M_{j,k}$: the number of receivers that are in session *j* with a capacity of *k* channels;

R_j : the rate allocation vector for layers in session *j*, $R_j = (r_j^1, r_j^2, ..., r_j^{L_j})$;

L_j : the total number of layers in R_j ;

r_j^i : the cumulative bandwidth up to layer *i* in R_j ;

K_j : the maximum capacity among all receivers in session *j*;

h : the bandwidth overhead for layering (measured by channel per layer);

$\mu_j(k,r,l)$: the utility function for a receiver in session *j*. *k* is the receiver's capacity, *r* is its actual received bandwidth, and *l* is the number of subscribed layers corresponding to *r*;

Q(*r*,*l*) : the mapping from layering structure to perceptual video quality;

$U_j(n_j)$: the utility of session *j* under a given session bandwidth budget of n_j channels;

$\hat{U}_j(n_j)$: the optimal utility of session *j* under a given session bandwidth budget of n_j channels.

Figure 1. Notations.

Assuming the *session bandwidth budget* for session *j* is n_j channels, any possible R_j should therefore satisfy $r_j^{L_j} \le n_j$. In addition, denote K_j the maximum capacity among all the receivers in session *j*. Clearly, it does not help to set the cumulative bandwidth to a video layer to be higher than K_j because no receiver can subscribe to this layer. Finally, if $\mu_j(k, r_j^l, l) \ge \mu_j(k, r_j^{l+1}, l+1)$ for $k \ge r_j^{l+1}$, then the (*l*+1)th layer allocation is not useful since subscription to layer *l*+1 will not further improve the utility to any receiver. Hence, we say that R_j is a *feasible layer allocation vector* if (1) $L_j > 0$; (2) $0 < r_j^1 < r_j^2 < ... < r_j^{L_j} \le \min\{K_j, n_j\}$; and (3) $\mu_j(k, r_j^l, l) < \mu_j(k, r_j^{l+1}, l+1)$ for $k \ge r_j^{l+1}$, *l*=1,2,…, L_j-1.

The input to the optimization algorithm is a 4-tuple $(N, \boldsymbol{S}, M_{j,k}, \mu_j)$, and the output is the maximum system utility U^*, together with the corresponding optimal allocation vectors, $R_j, j \in \boldsymbol{S}$, which gives the bandwidth allocation n_j for each session $j \in \boldsymbol{S}$, the total number of layers (L_j) for session *j*, and the bandwidth allocation ($r_j^i - r_j^{i-1}$) for each layer *i* in session *j*. This optimal layering and bandwidth allocation problem can be formally stated as follows:

$$(\textit{OPT-SYS}) \quad \max \ U = \sum_{j \in \mathbf{S}} \sum_{k=1}^{K_j} M_{j,k} \mu_j(k, r_{j,k}^*, l_{j,k}^*), \qquad (1)$$

$$\text{s.t.} \quad R_j \text{ is a feasible allocation vector}, j \in \mathbf{S},$$

$$\sum_{j \in S} r_j^{L_j} \le N .$$

B. Intra-session and Inter-session Bandwidth Decomposition

Since there are a finite number of channels in the system and the rate allocations among layers and sessions are in unit of a channel, there is a finite number of feasible rate allocation vectors for the sessions. Therefore, there exists an optimal solution for *OPT-SYS*.

To solve the optimization problem, we first introduce the notion of *session utility*, and use a decomposition technique for intra-session and inter-session allocations. The session utility $U_j(n_j)$ for session *j* is the total utility of all the receivers in the session under a feasible layer allocation vector R_j, and the *optimal session utility*, $\hat{U}_j(n_j)$, is the maximum of $U_j(n_j)$ among all possible allocation vectors. The following lemma shows that problem *OPT-SYS* can be solved in two steps. First, we perform *optimal intra-session allocation*, which optimally sets the layering structure (*i.e.,* the number of layers in a session) and allocates channels among the layers under each possible session bandwidth budget n_j. Second, we perform *optimal inter-session allocation*, which optimally allocates the total system bandwidth *N* among sessions $j \in \boldsymbol{S}$ based on the results of the optimal intra-session allocation.

Lemma 1 (Decomposition Lemma): For a total number of *N* channels in the system, the optimal system utility is the maximum of the sum of all the optimal session utilities. That is, $U^* = \max_{\sum_{j \in \mathbf{S}} n_j \le N} \sum_{j \in \mathbf{S}} \hat{U}_j(n_j)$.

Proof: First, by definition of U^*, we have that $U^* \ge \max_{\sum_{j \in \boldsymbol{S}} n_j \le N} \sum_{j \in \boldsymbol{S}} \hat{U}_j(n_j)$. Second, denote the session bandwidth allocation for U^* as ($n_1^*, n_2^*, ..., n_{|\boldsymbol{S}|}^*$), and the corresponding session utilities as $U_1(n_1^*), U_2(n_2^*)$, …, $U_{|\boldsymbol{S}|}(n_{|\boldsymbol{S}|}^*)$. We therefore have $U^* = \sum_{j \in \boldsymbol{S}} U_j(n_j^*) \le \sum_{j \in \boldsymbol{S}} \hat{U}_j(n_j^*) \le \max_{\sum_{j \in \boldsymbol{S}} n_j \le N} \sum_{j \in \boldsymbol{S}} \hat{U}_j(n_j)$. Combining these two inequalities, we have the lemma.

The decomposition lemma enables us to solve problem *OPT-SYS* through separate intra-session and inter-session allocations. In the following subsection, we describe these two sub-problems and present efficient algorithms for each of them.

C. Intra-Session Layering and Rate Allocation

For session *j*, assume the session bandwidth budget is given by n_j channels. The objective of optimal intra-session allocation is to find an appropriate layering structure (*i.e.,* number of layers) and the rate allocation for each layer such that the sum of the utilities among all the receivers in this session is maximized. We formally state the optimal intra-session layering and rate allocation problem as follows:

$$\textit{OPT-INTRA}(j,n_j)\quad \max\ U_j(n_j)=\sum_{k=1}^{K_j} M_{j,k}\mu_j(k,r^*_{j,k},l^*_{j,k}), \qquad (2)$$

$$\text{s.t.}\quad R_j \text{ is a feasible allocation vector.}$$

We use a recursive algorithm to solve this problem. The key idea is as follows. Since session j's bandwidth budget is n_j channels and the maximum capacity among all receivers in this session is K_j, and the fact that rate allocation for each layer is an integral number of channels, the number of layers for session j can only take countable number of values, *i.e.,* 1, 2, …, min{n_j, K_j}. We start with the 1-layer case, *i.e.,* there is only a single layer (base layer) for the session. In this case, the number of channels for this layer can vary from 1 to n_j, and we can easily calculate the utility for each allocation. Then we add one more layer on top of the 1-layer case and calculate the session utility for the 2-layer case. In general, upon the rate allocation for the (l-1)th layer, the l th layer can be laid on top of the (l-1)th layer using some remaining channels. Note that when considering an l-layer structure, only the receivers that can subscribe to layer l-1 in the previous step may be eligible to subscribe to a higher layer l (due to receiver capacity limitation). Therefore, given the optimal session utility for the case of l-1 layers, we only need to add the utility difference of these receivers, while not the receivers subscribing to lower layers (1 to l-2).

We now define an auxiliary function $\pi(m,l)$ as $\max_{L_j=l, r_j^l=m}\sum_{k=1}^{K_j} M_{j,k}\mu_j(k,r^*_{j,k},l^*_{j,k})$ for l=1,2,…,min{n_j, K_j} and m=1,2,…, min{n_j, K_j}, *i.e.,* the optimal session utility when a total number of l layers are generated and the cumulative bandwidth up to layer l is m channels. The solution to the problem *OPT-INTRA* (j,n_j) is clearly given by $\max_{1\le l\le \min\{n_j,K_j\},1\le m\le \min\{n_j,K_j\}}\pi(m,l)$. Based on the above discussions, we give a recurrence relation of $\pi(m,l)$ in Fig. 2.

For $n_j > K_j$, *i.e.,* the session bandwidth budget is higher than the maximum receiver capacity, we let $\hat{U}_j(n_j)=\hat{U}_j(K_j)$. Once the optimal session utility is obtained, the corresponding layer allocation vector can be easily obtained by applying a backtracking method on the recurrence relation for $\pi(m,l)$.

The correctness of the recurrence relation can be proved by induction. For the base case l=1, there is only one layer to be generated with bandwidth m. $\pi(m,1)$ is thus $\sum_{k=1}^{m-1} M_{j,k}\mu_j(k,0,0)$ $+\sum_{k=m}^{K_j} M_{j,k}\mu_j(k,m,1)$. The first term is the total utility of the receivers that cannot subscribe to the layer (k<m), and the second term is the total utility of all other receivers (k≥m). Note that $\mu_j(k,0,0)$ can be set to a very small value or even a negative value to ensure that, under an optimal allocation, all the receivers can subscribe to at least one layer (the base layer).

```
(Base case)
  For $l = 1, m \le \min\{n_j, K_j\}$,
    $\pi(m,1)$
      $= \sum_{k=1}^{m-1} M_{j,k}\mu_j(k,0,0) + \sum_{k=m}^{K_j} M_{j,k}\mu_j(k,m,1)$;
(Recursion)
  For $1 < l \le \min\{n_j, K_j\}, 1 < m \le \min\{n_j, K_j\}$,
    $\pi(m,l)$
      $= \max_{1\le i<m} \left\{ \pi(i,l-1) + \sum_{k=m}^{K_j} M_{j,k}\Delta(k,m,i,l) \right\}$,
    where $\Delta(k,m,i,l) = \mu_j(k,m,l) - \mu_j(k,i,l-1)$;
For all other cases, $\pi(m,l)$ is set to 0.
```

Figure 2. Algorithm for $\pi(m,l)$ calculation.

For the general case $1 < l \le \min\{n_j, K_j\}$, there are l layers to be generated, which can be viewed as adding a new layer to the case with only l-1 layers. Without loss of generality, we assume this new layer is layer l, and suppose i is the cumulative bandwidth up to layer l-1. All the receivers that subscribe to layer l should have capacities greater than i. Therefore, in the (l-1)-layer case, all such receivers should subscribe to layer l-1, the highest layer. The difference of the session utility when layer l is generated on top of the (l-1)-layer case is thus $\sum_{k=m}^{K_j} M_{j,k}\Delta(k,m,i,l)$.

Since $\pi(i,l-1)$ is the optimal session utility for the (l-1)-layer case, $\pi(m,l)$ is given by

$$\max_{1\le i<m}\left\{ \pi(i,l-1)+\sum_{k=m}^{K_j} M_{j,k}\Delta(k,m,i,l) \right\}.$$

D. Inter-session Rate Allocation

The objective of inter-session bandwidth allocation is to optimally allocate the total N channels in the system to different sessions $j \in \boldsymbol{S}$ so that the system utility is maximized. Given that the optimal session utilities, $\hat{U}_j(n_j)$, $j \in \boldsymbol{S}$, n_j=1,2,…, K_j, have been calculated in the optimal intra-session layering and rate allocation, the inter-session allocation problem can be stated as follows:

$$(\textit{OPT-INTER})\qquad \max\ U^* = \sum_{j\in \boldsymbol{S}} \hat{U}_j(n_j), \qquad (3)$$

$$\text{s.t.}\quad n_j > 0, j \in \mathbf{S},\ \text{and}\ \sum_{j\in\mathbf{S}} n_j \le N.$$

This optimization problem can also be solved using a recursive algorithm. We define an auxiliary function $\omega(n,i)$ as

$\max_{n=\sum_{j=1}^{i} n_j} \sum_{j=1}^{i} \hat{U}_j(n_j)$ for n=1,2,…,N, and i=1,2,…,$|\boldsymbol{S}|$, *i.e.,* the maximum total utility of sessions 1, 2, …, i when a total bandwidth of n channels are allocated to these sessions. The solution to problem *OPT-INTER* is thus $\max_{1\le n\le \min\{N,\sum_{j\in S}K_j\}} \omega(n,|\boldsymbol{S}|)$. The algorithm in Fig. 3 can be used to calculate $\omega(n,i)$.

(Base case)

$$\omega(n,i)=\begin{cases}\hat{U}_1(n), & \text{for } i=1,\ 1\le n\le\min\{N,K_1\},\\ \sum_{j=1}^{i}\hat{U}_j(K_j), & \text{for } n\ge\sum_{j=1}^{i}K_j;\end{cases}$$

(Recursion)

For $1<i\le|\boldsymbol{S}|$, $1\le n\le\min\left\{N,\sum_{j\in S}K_j-1\right\}$,

$$\omega(n,i)=\max_{1\le m\le\min\{K_i,n-i+1\}}\{\ \omega(n-m,i-1)+\hat{U}_i(m)\ \}$$

For all other cases, $\omega(n,i)$ is set to 0.

Figure 3. Algorithm for $\omega(n,i)$ calculation.

The correctness of the algorithm for calculating $\omega(n,i)$ can be proved by induction as well. We start from the base case of i=1. That is, only session 1 is considered. For a given total bandwidth n, $1\le n\le\min\{N,K_1\}$, the total utility is simply the session utility of session 1, *i.e.*, $\omega(n,i)=\hat{U}_1(n)$.

The case of $1<i\le|\boldsymbol{S}|$ can be viewed as adding session i to an allocation for i-1 sessions. According to Lemma 1, the session utility for session j depends only on its own session bandwidth and is independent of the status of other sessions. Therefore, assume its session bandwidth is set to m, the maximum total utility for all i sessions should be $\omega(n-m,i-1)+\hat{U}_i(m)$. Therefore, the value of $\omega(n,i)$ can be obtained by checking all possible settings for m, *i.e.,* $\omega(n,i)=\max_{1\le m\le\min\{K_i,n-i+1\}}\{\ \omega(n-m,i-1)+\hat{U}_i(m)\ \}$.

E. Complexity

To perform inter-session bandwidth allocation, we should calculate the session utilities for all possible session bandwidth budgets, *i.e.,* solving problems *OPT-INTRA* (j,n_j) for j=1,2,…,$|\boldsymbol{S}|$ and n_j = 1, 2, ...,K_j . Note that, if $\pi(m,l)$ for $1\le m\ \ \le K_j$ and $1\le l\le K_j$ are available, then all the above problems can be solved. Fortunately, these values of $\pi(m,l)$ can be obtained using the recurrence relation (see Fig. 2) in polynomial time $O[(K_j)^3\cdot E]$, where E is the time for calculating $\sum_{t=m}^{K_j}M_{j,k}\Delta(k,m,i,l)$.

For the optimal inter-session allocation algorithm, when all session utilities are available, its time complexity is $O(|\boldsymbol{S}|\cdot N\cdot K^{\max})$, where $K^{\max}=\max_{j\in\boldsymbol{S}}K_j$.

We can employ several techniques to speed up the optimization algorithms. First, an absolute utility function $\mu_j(k,r,l)$ depends only on the receiver's actual received bandwidth r and the corresponding number of layers l, and is independent of the receiver's own capacity k. Hence, in Fig. 2, we have

$$\begin{aligned}&\sum_{k=m}^{K_j}M_{j,k}\Delta(k,m,i,l)\\&=\sum_{k=m}^{K_j}M_{j,k}[\mu_j(k,m,l)-\mu_j(k,i,l-1)]\\&=\sum_{k=m}^{K_j}M_{j,k}[\mu_j(m,m,l)-\mu_j(m,i,l-1)]\\&=\Delta(m,m,i,l)\cdot\sum_{k=m}^{K_j}M_{j,k}\ .\end{aligned}$$

Since $\sum_{k=m}^{K_j}M_{j,k}$, k=1,2,…, K_j are invariants in the execution of the algorithm, they can be precomputed and stored in $O(K_j)$ space. Therefore, E is $O(1)$ and the time complexity of the optimal intra-session allocation algorithm is simply $O[(K_j)^3]$. For a relative utility, since $\mu_j(k,r,l)$ depends on k, E is $O(K_j)$ in general. However, there are still some common relative utilities functions that are of $O(1)$ complexity. For example, consider the relative utility function *Application-aware Fairness Index* (AFI) defined as $\mu_j(k,r,l)$ $=Q(r,l)/Q(k,1)$ [11], we have

$$\begin{aligned}&\sum_{k=m}^{K_j}M_{j,k}\Delta(k,m,i,l)\\&=\sum_{k=m}^{K_j}M_{j,k}[\frac{Q(m,l)}{Q(k,1)}-\frac{Q(i,l-1)}{Q(k,1)}]\\&=[Q(m,l)-Q(i,l-1)]\cdot\sum_{k=m}^{K_j}\frac{M_{j,k}}{Q(k,1)}\ ,\end{aligned}$$

where $\sum_{k=m}^{K_j}M_{j,k}/Q(k,1)$, k=1,2,…, K_j , can be precomputed and stored as well, and thus E remains $O(1)$.

Second, in the recursion for calculating $\pi(m,l)$ in Fig. 2, if $\sum_{k=m}^{K_j}M_{j,k}\Delta(k,m,i,l)\le 0$ for some $i<m$, then it shows that such bandwidth setting for layers l-1 and l is no longer useful. Furthermore, for layer l with bandwidth m, we do not need to check other settings for layer l-1 that are higher than i, because we have $\mu_j(k,r,l)<\mu_j(k,r',l)$, $r\le r'$ for a practical video encoder and as a result, for i<i'<m, we have

$$\begin{aligned}&\sum_{k=m}^{K_j}M_{j,k}\Delta(k,m,i',l)\\&=\sum_{k=m}^{K_j}M_{j,k}[\mu_j(k,m,l)-\mu_j(k,i',l-1)]\\&\le\sum_{k=m}^{K_j}M_{j,k}[\mu_j(k,m,l)-\mu_j(k,i,l-1)]\\&=\sum_{k=m}^{K_j}M_{j,k}\Delta(k,m,i,l)\\&\le 0\ .\end{aligned}$$

Finally, if $\sum_{k=m}^{K_j} M_{j,k}\Delta(k,m,i,l) \le 0$ for all $1 \le i < m$, then setting the cumulative bandwidth of layer l to m is useless. It can be shown that setting a higher layer (>l^{th}) to m does not improve the session utility in this case. Hence, we can simply assign 0 to $\pi(m,l)$, as well as to $\pi(m,l+1)$, …, $\pi(m,m)$.

To demonstrate the efficiency of our optimization algorithms in practice, we implement the optimization algorithms using C++ on an Intel Pentium III 900MHz PC with 256MB memory. The execution times under different settings with the AFI utility function are listed in Table 1. We find that the solutions can be computed within a reasonably short time period, which shows that it is suitable for real-time applications.

TABLE I. EXECUTION TIME FOR THE OPTIMIZATION ALGORITHMS.

Setting	Execution Time (ms)		
$N,\lvert\mathbf{S}\rvert,K^{\max}$	Intra-Session Allocation*	Inter-Session Allocation	Joint Allocation
(64,8,12)	0.9	1.2	8.4
(128,10,15)	1.4	2.9	16.9
(512,20,30)	2.1	7.0	49.0

* For one session with maximum receiver capacity being equal to K^{max}.

IV. IMPLEMENTATION CONSIDERATIONS

In this section, we discuss some implementation issues in practice, including the choice of layered video codec, the computation overhead in an online implementation, and adaptation for multi-cell networks.

A. *Choice of Video Codec*

In the video coding area, scalable coding typically refers to layered coding. In this paper, we are particularly interested in scalable video coders with a flexible layering structure and fine-granularity in terms of rate control. Recent advances in scalable video coding have demonstrated that this is possible and can be done efficiently. A representative technique is bit-plane coding, which uses embedded representations in compression [12]. For illustration, there are 64(8x8) DCT coefficients for each video block. All the most significant bits from the 64 DCT coefficients form bitplane 0, all the second most significant bits form bitplane 1, and so forth. In the output stream, the bitplanes, not the coefficients, are placed sequentially. Hence, layers can be generated by an assembling and packetization procedure, which can truncate the embedded stream in any position to achieve a specified output rate. This post-encoding method is different from the traditional scalability tools that use a fixed layering structure and perform rate control at the source coding stage. As a result, bitplane-based scalable coding can achieve very flexible layering structure, which makes it possible to produce arbitrary number of layers and to fine tune the rate of each layer with a fast response time. Bit-plane coding has been adopted in the MPEG-4 Fine Granularity Scalability (FGS) [12].

Regarding the layering overhead, there are several factors that need to be considered. For example, a packetization scheme can affect the overhead since different packetization schemes use a different amount of bits for layer identification, synchronization and error concealment, leading to different overheads. In the experiments described in the next section, we use a wide range of settings to take into account of such layering overhead.

B. *Online Implementation*

Under an online implementation of the optimization algorithm, the whole system performs adaptation (to achieve the optimal layering structure and rate allocation among layers/sessions) either periodically or when the system conditions change (*e.g.,* when some receivers join or leave the cell or video sessions, or the total available bandwidth for all the video sessions, *N*, is changed by the central point to achieve a better balance between video traffic and other voice or data traffic in the same network). Such an online adaptation can be done by taking advantage of the flexible layering structure provided by advanced layered coders, as discussed previously.

There is a potential issue of computational overhead associated with online adaptation, but as we have shown earlier, our algorithm runs reasonably fast for real-time adaptation. Furthermore, since our algorithm is based on the bandwidth distribution among all the receivers (instead of the bandwidth of an individual receiver), the adaptation algorithm needs to be executed only when the distribution has changed significantly, which can be easily identified by using standard statistical methods, *e.g.,* the Pearson's χ^2 -test or the Kolmogorov-Smirnov test. Finally, according to the principle of decomposition, the session utility of a particular session is independent of the receiver status of other sessions. Hence, when the session status changes, only the session's own utility needs to be re-calculated and one execution of inter-session bandwidth allocation is needed.

C. *Multi-cell Adaptation*

It is worth noting that our optimal layering and bandwidth allocation algorithm is also applicable to other broadcast- or multicast-capable networks. Another possible extension is to use it in a multi-cell network, where the receivers can move across cells and smooth handoff thus becomes a crucial issue. However, if the receiver distribution is uniform in the whole network, the allocations given by independent cells should be similar from a statistical point of view. Although the receiver distributions may be highly heterogeneous, we found that, with the optimal allocations, the subscription bandwidth for a receiver usually does not change drastically during handoff. In most cases, the bandwidth difference is less than 20 %, which can possibly be masked on the receiver's side by using seamless transition techniques for video streams [4]. Since a global optimization with cell collaborations is usually of high complexity and incurs extra overheads for information exchange among cells, we recommend that each cell performs allocation independently. This practice is also well suited for FGS coding since its layer partitioning as well as rate control is performed as a post-encoding process, and can easily be implemented at each access point without generating replicated streams from the video source.

V. Numerical Investigations

In this section, we conduct experiments to demonstrate the performance of the optimal layering and rate allocation algorithms for video broadcasting. We also compare it to commonly used non-optimal allocation schemes to show performance improvement.

A. Simulation Settings

To show the heterogeneous nature of the receivers, we model the capacity of different receivers in a session with a multi-modal distribution. Specifically, we observe that the access link and video decoding component of a receiver typically follows some specific standards or use customized software/hardware [2]. Thus, we assume there are several clusters each following a Gaussian distribution. In our simulation, we assume the bandwidth of each channel is 28.8 Kbps, with the minimum and maximum receiver capacities being 2 and 25 channels, respectively. This range covers the rate of many available wireless link access technologies and is also the typical dynamic range for existing scalable video coders, such as the MPEG-4 FGS coder. The standard deviation of a cluster is set to 10% of the cluster mean. Therefore, most bandwidth differences are within ± 10%, yet a few reach about ± 40% or more, which reflects the flexibility in device design.

We use an enhanced MPEG-4 Fine-Granularity Scalable (FGS) video encoder to generate layered video streams. A standard video test sequence "Foreman (CIF)" is used in our study. The TM-5 rate control model is adopted to control the bit-rate of the base layer. The number of the enhancement layers as well as their respective bandwidth is allocated by an assembling and packetization module. As in previous studies [4], we define the layering overhead per layer, h, as follows: assume L layers are generated at bandwidth r, and a single-layer stream with the same video quality has bandwidth r_0, h is given by (r- r_0)/(L-1) channels per layer.

B. Intra-session Allocation

In this subsection, we focus on a single session and conduct experiments to show the performance and behavior of the optimal algorithm for intra-session layering and rate allocation.

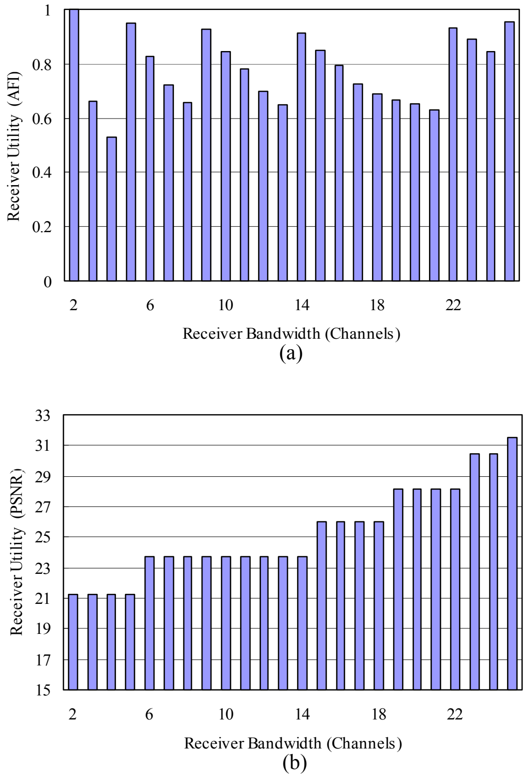
B.1 Effect of Utility Functions

We have used a series of utility functions to study their impact, including typical mappings used in the literature [9,10], as well as mappings for practical layered video encoders. Specifically, we present the results for an absolute utility function $\mu_s(k,r,l) = Q(r,l)$, where $Q(r,l)$ is the objective video quality measured by the Peak Signal-to-Noise Ratio (PSNR) with a unit in dB, and the relative utility function *Application-aware Fairness Index* (AFI) described before.

Table 2 presents the optimal allocation vectors with the above two utility functions for a capacity distribution of 6 clusters. The layering overhead h is 0.5 channel per layer, which is moderate. We find that, with different utility functions, the corresponding optimal bandwidth allocation for each layer is different. For a system employing an absolute utility function, the rate allocation typically favors receivers with high bandwidths. This can be observed in Fig. 4, where the receiver utility under different access capacity is plotted. Under the absolute utility function (PSNR), the utility value is non-decreasing with respect to the receiver access capacity. As a result, the optimal allocation tends to allocate more layers in the high capacity region so that higher utility can be obtained from this region. On the contrary, the relative utility function (AFI) does not favor high capacity receivers because the utility is normalized. Such observations confirm our arguments in Section II. For the rest of the experiments in this section, we will show results with the AFI utility function only.

Table II. Layer Allocation vectors for intra-session allocation with different utility functions

Utility Function	Optimal Layer Allocation Vector
PSNR	(2,6,15,19,23,25)
AFI	(2,5,9,14,22,25)



(a)

(b)

Figure 4. Receiver utility for optimal allocation with different utility functions: (a) Relative utility, Application-Aware Fairness Index (AFI), and (b) Absolute utility, Peak Signal-to-Noise Ratio (PSNR) in dB.

B.2 Impact of Layering Overhead

Figure 5 shows the optimal session utility as a function of the bandwidth allocated to the session. The layering overheads are 0, 0.2, 0.5, 1.0 channel per layer, respectively, which cover both light and heavy overhead cases. In this figure, as well as the remaining figures, the session utility (or system utility) is normalized by the number of receivers in the session (or system). Not surprisingly, all the curves are non-decreasing as the session bandwidth increases. The optimal session utility of h=0 (no layering overhead) achieves the ideal session utility, 1, when the bandwidth budget is at least 25 channels. In this case, each receiver has a layer whose cumulative bandwidth perfectly matches the receiver's capacity. However, if the layering overhead is taken into account, the ideal session utility cannot be achieved because the overhead counteracts the benefits from increasing the number of layers. In all these cases, the session utility converges to a steady value for bandwidth greater than 25 channels, the highest access capacity among all the receivers in the session.

Figure 6 shows the optimal session utility as a function of the number of layers for different layering overheads. It can be seen that, when *h*=0, the session utility is non-decreasing with the number of layers for the given session bandwidth budget. However, if *h*>0, the session utility is no longer non-decreasing and has a maximum value at a certain number of layers (in this example, about 4 to 6 layers, depending on the overheads). Intuitively speaking, below that number of layers, the adaptation granularity on the receiver's end is somewhat coarse; above which, more layering overhead is incurred.

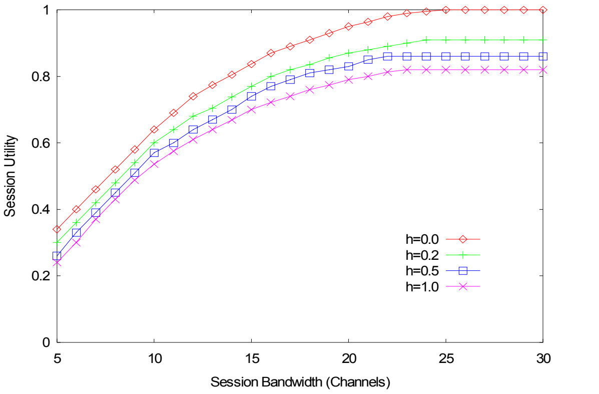


Figure 5. Optimal session utility as a function of session bandwidth for different layering overheads (*h* channel per layer).

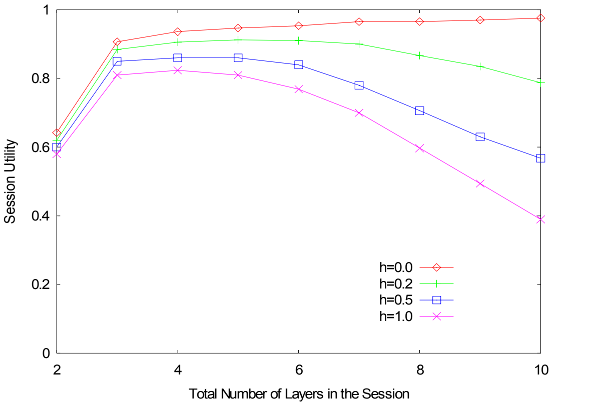


Figure 6. Optimal session utility for a given number of layers with different layering overheads. Session bandwidth is 25 channels.

The above results clearly demonstrate that, if the layering overhead is not considered, the use of "thin" layers, *i.e.*, generating more layers for a given session bandwidth budget, is preferable. On the other hand, if the layering overhead is considered, there exists an optimal number of layers such that the session utility is maximized. This optimal number can be found using our optimal intra-session layering and rate allocation algorithm.

Note that, regardless of whether layering overhead is considered or not, the session utility (as well as the corresponding allocations) under different session bandwidth budget is different, even if the same number of layers are generated. For example, in Fig. 8, for the 15-channel case, the maximum session utility is 0.78. However, in the 25-channel case, it can reach 0.86. In a bandwidth-limited network, it is thus necessary to use an inter-session allocation scheme to optimally allocate the available bandwidth to different sessions.

B.3 Optimal versus Non-Optimal Allocations

In this experiment, we compare the performance of our optimal allocation scheme and a scheme employing a fixed layering structure. Again, we focus our study on a single session. In the literature, a widely recommended fixed allocation scheme is the exponential allocation, in which the cumulative layer rates are exponentially spaced by a constant factor $\alpha > 1$, *i.e.*, $r_j^{i+1} = \alpha\, r_j^i$. This is the scheme adopted in the Receiver-driven Layered Multicast (RLM) protocol [3] and many other experiments [7,11]. Given the session bandwidth budget $n_j (\le K_j)$, the lower bound of the base layer bandwidth n_b, and the number of layers L_j, α can be calculated as $\sqrt[(L_j-1)]{n_j / n_b}$. In this experiment, we assume that L_j is fixed to 5 layers for the exponential allocation. For $n_j > K_j$, we assume the allocation is the same as that for $n_j = K_j$.

Figure 7 shows the session utility as a function of session bandwidth for the optimal allocation and the exponential allocation. Clearly, the session utility under the optimal allocation is greater than that under the exponential allocation. In particular, under the optimal intra-session allocation, the session utility is non-decreasing while under the exponential allocation, the behavior of the session utility is sometimes unpredictable. This is because the exponential allocation is not aware of the receivers' bandwidth distribution for the session. It may allocate the layer bandwidth to be a level with few receivers, and hence aggravates the bandwidth mismatches. We also show the results with different numbers of layers for the two allocation schemes in Fig. 8. Again, there are significant gaps between the two schemes even if the numbers of layers are the same. These results reaffirm that the optimal choice of the number of layers must be used in conjunction with the optimal bandwidth allocation for each layer, and vice versa.

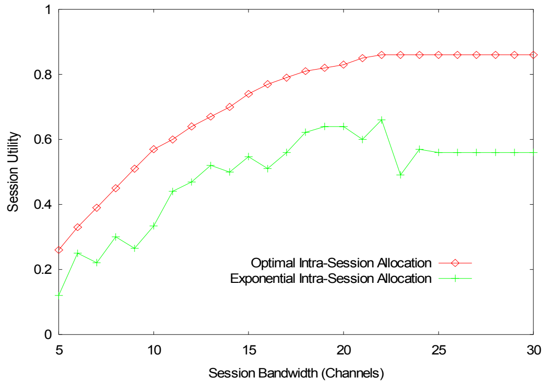


Figure 7. Session utility as a function of session bandwidth for the optimal and exponential allocation schemes. Layering overhead *h* is 0.5 channel/layer.

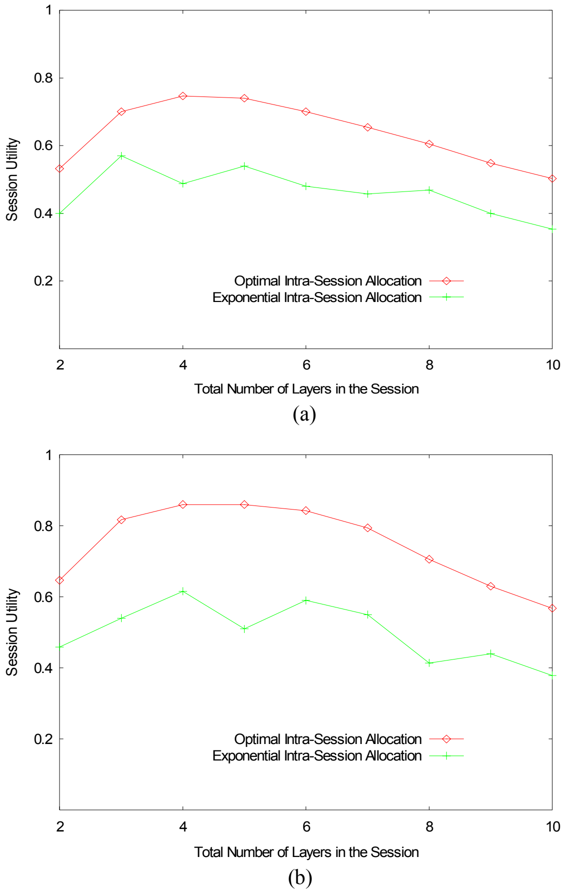


Figure 8. Session utility as a function of the number of layers for the optimal and exponential allocation schemes. (a) Session bandwidth is 15 channels, layering overhead is 0.5 channel/layer, (b) Session bandwidth is 25 channels, layering overhead is 0.5 channel/layer.

C. *Joint Intra- and Inter-session Allocation*

We also study the effect of joint intra-session and inter-session layering and bandwidth allocation, and try to identify the respective contribution of the optimal intra-session and inter-session allocations to the total system utility, specifically for the case wherein the sessions have uneven populations.

We assume that the demand probabilities for different video sessions follow a Zipf distribution [15]. This distribution has been widely used in the literature and captures the difference in terms of popularity for the video sessions. The Zipf distribution is expressed as

$$p_j = \frac{(1/j)^\theta}{\sum_{j=1}^{|S|}(1/j)^\theta}, \qquad j = 1, 2, \ldots, |\boldsymbol{S}|, \tag{4}$$

where θ is a parameter called *skew factor*. For θ=0, the Zipf distribution is reduced to a uniform distribution with $p_j = 1/|\boldsymbol{S}|$. However, the distribution becomes increasingly "skewed" as θ increases, *i.e.,* a few popular video sessions attract many more receivers than the others. In other words, the session popularities are differentiated.

We consider all four possible combinations of the intra-inter session allocation: (1) OptIntra-OptInter, where both intra- and inter session allocations are optimal; (2) OptIntra-UniInter, where only intra-session allocation is optimal and inter-session allocation is a uniform allocation; (3) ExpIntra-OptInter, where only inter-session is optimal and intra-session is exponential allocation; and (4) ExpIntra-UniInter, where both are non-optimal. In the experiments, we assume that there are 500 receivers belonging to 10 sessions. We vary the skew factor θ for session popularity distribution from 0 to 1. The number of clusters for the receiver capacity distribution in a session is uniformly distributed from 2 to 9. We then draw 500 samples from the above model to obtain a receivers' status distribution for the whole system.

Figure 9 shows the system utilities with different skew factors for all the four combinations. It is clear that the optimal intra-inter allocation scheme outperforms all the other schemes. Comparing these curves, specifically the curves of OptIntra-UniInter and ExpIntra-OptInter, we find that the contribution of the optimal inter-session allocation becomes more important as the skew factor increases. Note that a higher skew factor means that some popular video programs attract much more receivers than others. It is thus advisable to allocate more channels to these sessions. Specifically, in Fig. 9, for a total system bandwidth of 128 channels, when θ >0.5, ExpIntra-OptInter outperforms OptIntra-UniInter. This reaffirms our claim that the optimal inter-session allocation should be considered.

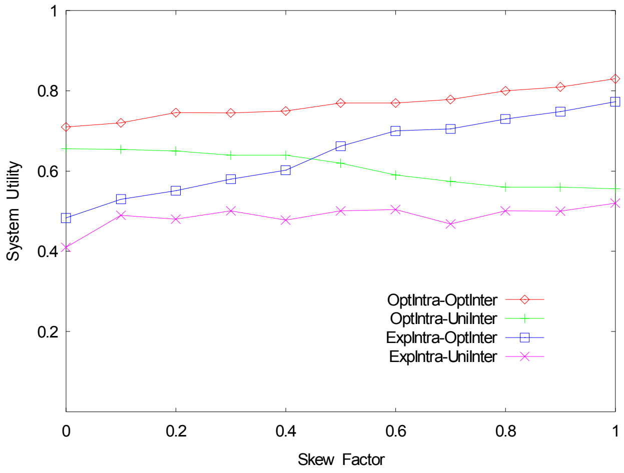


Figure 9. Total system utility for different skew factors. Total system bandwidth *N* is 128 channels.

VI. RELATED WORK

There has been extensive work on layered video transmission for both wired and wireless networks [2]-[11]. McCanne *et al.* [3] proposed the first practical receiver-based adaptation algorithm for layered video multicast over the Internet. This algorithm, known as Receiver-driven Layered Multicast (RLM), sends each video layer over a separate multicast group. A receiver periodically joins a higher layer's group to explore the available bandwidth. Since the adaptation is done only on the receiver's side, the granularity is considerably coarse given that the number of layers is limited and the bandwidth for each layer are predetermined at the source.

To remedy this mismatch between a receiver's capacity and the bandwidth of the video layers, the use of thin layers or dynamic layer bandwidth allocation on the sender's side [5]-[11] has been proposed in the literature. Specifically, Shacham [5] presented an optimal layer bandwidth allocation algorithm that maximizes the total utility for all the receivers. It employs an absolute utility function that depends only on the received bandwidth. Optimal algorithm using relative utility functions are presented in [6,11]. These allocation algorithms use end-to-end adaptation for the Internet environment and focus only on a single session case. Kar *et al.* [8] presented a distributed algorithm that maximizes the total utility for all the receivers belonging to different sessions by employing some intermediaries. In the above optimization schemes, the number of layers is usually assumed to be predetermined. Layering overheads, in particular, the overhead that depends on the number of layers, are not considered. In addition, they are restricted to specific utility functions or have some restrictions on the utility functions that can be used, such as continuous, differentiable, strictly concave or convex.

In mobile wireless networks, the adaptability of layered video is used to trade-off the carried traffic and the bandwidth degradation, *i.e.,* minimizing the overload probability of the system by temporally reducing some receivers' subscription levels, and at same time, ensuring the degree of fairness among receivers [13]. Many utility functions have been considered in existing works. However, their optimization objective is different to our problem. For example, their primary focus is the unicast case rather than broadcast or multicast with heterogeneous receivers as we have addressed in this paper.

Our work is motivated by these previous efforts. We consider layered adaptation in a wireless network, and employ a general utility formulation, which can accommodate different measures such as throughput, video quality, user satisfaction and fairness. It also takes into account of the bandwidth overhead for layered video, as existing experimental results show that such overhead is not negligible in practice [4,12]. We further consider the optimal allocation for multiple video sessions with heterogeneous popularity. Our optimization algorithm is general since it imposes very weak constraints on the utility function.

VII. CONCLUSIONS AND FUTURE WORK

In this paper, we presented a systematic study of dynamic layering and bandwidth allocation (among sessions and layers) for video broadcasting in a wireless environment. We employed a generic utility function for each receiver under each video session. We cast the joint problem of layering and bandwidth allocation into an optimization problem of total system utility among all receivers. By using a 2-step decomposition technique for inter-session and intra-session allocations, we derived efficient algorithms to obtain the optimal layering and bandwidth allocation. Numerical results showed that the optimal layering and bandwidth allocation significantly improves the total system utility. Practical issues for deploying the optimal algorithm in typical wireless networks were also discussed. We demonstrated that our algorithm can be efficiently supported by the recently developed scalable video codecs such as MPEG-4 FGS with low overall system complexity.

We are currently implementing our algorithms in a wireless LAN testbed using available layered video coders and conducting experiments to demonstrate the advantaged of our optimization algorithms in a practical setting. For multi-cell adaptation, handoff between cells remains a non-trivial undertaking given that video contents are highly dependent. Issues like drift reduction [12] and layer synchronization are also interesting research topics. We are also considering multiple access points and investigating the impact of handoff and developing effective handoff algorithms.

REFERENCES

[1] ISO/IEC/SC29/WG11, "Delivery multimedia integration framework, DMIF (ISO/IEC 14496-6)," *International Standards Organization*, February 1999.

[2] D. Wu, Y.T. Hou, and Y.-Q. Zhang, "Scalable video coding and transport over broadband wireless networks," *Proceedings of the IEEE*, vol. 89, no. 1, pp. 6-20, January 2001.

[3] S. McCanne, V. Jacobson, and M. Vetterli, "Receiver-driven layered multicast," in *Proceedings of ACM SIGCOMM'96*, August 1996.

[4] P. de Cuetos, D. Saparilla, and K. W. Ross, "Adaptive streaming of stored video in a TCP-friendly context: multiple versions or multiple layers," in *Proceedings of International Packet Video Workshop*, April 2001.

[5] N. Shacham, "Multipoint communication by hierarchically encoded data," in *Proceedings of IEEE INFOCOM'92*, May 1992.

[6] Y. Yang, M. Kim, and S. Lam, "Optimal partitioning of multicast receivers," in *Proceedings of ICNP'00*, November 2000.

[7] S. Gorinsky and H. Vin, "The utility of feedback in layered multicast congestion control," in *Proceedings of NOSSDAV'01*, June 2001.

[8] K. Kar, S. Sarkar, and L. Tassiulas, "Optimization based rate control for multirate multicast sessions," in *Proceedings of IEEE INFOCOM'01*, April 2001.

[9] B. Vickers, C. Albuquerque, and T. Suda, "Source adaptive multi-layered multicast algorithms for real-time video distribution," *IEEE/ACM Transaction on Networking*, vol. 8, no. 6, pp. 720-733, December 2000.

[10] T. Jiang, E.W. Zegura, and M.H. Ammar, "Inter-receiver fair multicast communication over the Internet," in *Proceedings of NOSSDAV'99*, June 1999.

[11] J. Liu, B. Li, and Y.-Q. Zhang, "An end-to-end adaptation protocol for layered multicast using optimal rate allocation," to appear in *IEEE Transactions on Multimedia*.

[12] W. Li, "Overview of the fine granularity scalability in MPEG-4 video standard", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 11, no. 3, pp. 301-317, March 2001.

[13] A.K. Talukdar, B.R. Badrinath, and A. Acharya, "Rate adaptation schemes in networks with mobile hosts," *ACM/IEEE MOBICOM'98*, October 1998.

[14] X. Sun, F. Wu, S. Li, W. Gao and Y.-Q Zhang, "Seamless switching of scalable video bitstreams for efficient streaming," in *Proceedings of IEEE ISCAS'02*, May 2002.

[15] A. Dan, D. Sitaram, and P. Shahabuddin, "Scheduling policies for an on-demand video server with batching," in *Proceedings of ACM Multimedia"94*, October 1994.