Complexity and Heuristics for Wireless Broadcast with Noncumulative Layered Data

Jiangchuan Liu, *Member, IEEE*, Bin Li, *Senior Member, IEEE*, Bo Li, *Senior Member, IEEE*, and Xi-Ren Cao, *Fellow, IEEE*

***Abstract*—Layer transmission generates multiple layers for a video program, enabling a receiver to selectively subscribe to the layers commensurate with its bandwidth. It is an effective solution to the problem of bandwidth heterogeneity in video broadcasting. However, two important issues remain to be addressed. First, how does a receiver select the subset of layers to achieve the highest bandwidth utilization? Second, how does the sender optimally allocate the layer bandwidth to match the diverse bandwidth requirements from the receivers? We formally investigate these problems in noncumulative layered broadcasting, where *any* subset of the layers can be used to reconstruct the video. We formulate both the optimal layer subscription problem for a receiver and the optimal layer bandwidth allocation problem for the sender. We show that the former has an effective solution, while the latter is computationally intractable. Three efficient heuristic algorithms are then proposed for the allocation problem, and simulation results show that all of them significantly outperform nonadaptive allocation algorithms.**

***Index Terms*—Bandwidth allocation, broadcasting, noncumulative layered video.**

I. Introduction

BROADCASTING, an efficient vehicle for group communications, is inherently supported by wireless networks [1]. A typical application for broadcasting is real-time video distribution to a large population of receivers. It is envisioned that mobile users with a variety of hardware configurations, such as cellular phones and laptops, will be able to easily access various video services through wireless links in the near future. Since, in a broadband wireless network, these receivers may have different network resources, such as the number of channels they can simultaneously access, a single transmission rate is unlikely to satisfy all the receivers. It is thus desirable to use *multirate transmission*, in which the receivers within one broadcast session can receive video streams at different qualities [3], [15].

A widely cited multirate transmission approach is *layered transmission* [3]. In this approach, the video is encoded into a set of layers and then distributed to receivers via separate broadcast channels. For a receiver, the video quality is low if only one layer is decoded yet can be refined by subscribing to more layers if extra bandwidth is available. There are two layering schemes, *cumulative* and *noncumulative*. In the former, layers are dependent, where a layer with the highest importance, called a *base layer*, contains the data representing the most important features of the video; additional layers, called *enhancement layers*, contain data that progressively refine the reconstructed video quality [3], [5]. A receiver should always subscribe to layers consecutively, starting from the base layer. In the noncumulative layering, however, all layers have the same priority and any subset of the layers can be used for video reconstruction [5], [16]. Therefore, it is more flexible than the cumulative layering. Specifically, given L layers, a receiver would have 2^L choices for layer subscription, while not at most L in the cumulative layering case. As such, the diverse bandwidth demands from the receivers can be better satisfied. In addition, the system can be more robust in the presence of errors, especially in the context of wireless transmission [7].

In practice, noncumulative layering can be achieved through multiple description (MD) video coding, which generates multiple layers (called *descriptions*) for the source signal using an MD scalar quantization or an MD transform [7], [8], [20]. Recently, much effort has been devoted to develop and improve MD coders, and their efficiency is now quite close to existing single description coders. However, we are aware that some important issues remain to be addressed for transmitting such video in a heterogeneous broadcast environment.

First, for a receiver, how do we select the subset of layers that best matches its bandwidth requirement? This is trivial for the cumulative layered broadcasting, but not so for the noncumulative case as the selection policy is much more flexible. Second, even if the selection is optimal for a receiver, there could still be some mismatch between its requirement and the possible subscription bandwidths, given that the number of layers is practically limited by an MD encoder; hence, the adaptation unit on the receiver's side is a coarse-grained layer. Through simulation, we find that such mismatch causes nonnegligible degradation on user satisfaction. Nevertheless, we believe it can be minimized using some receiver-aware bandwidth allocation on

Manuscript received March 1, 2003; accepted September 18, 2003. The editor coordinating the review of this paper and approving it for publication is Y.-B. Lin. This work was supported in part by a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China, under Contract CUHK C001/2050312. The work of B. Li was supported in part by grants from Research Grants Council (RGC) under Contracts HKUST 6196/02E and HKUST6204/03E, a grant from NSFC/RGC under Contract N_HKUST605/02, and a grant from Microsoft Research under Contract MCCL02/03.EG01.

J. Liu was with the Department of Computer Science and Engineering, The Chinese University of Hong Kong, Shatin, NT, Hong Kong. He is now with the School of Computing Science, Simon Fraser University, BC, Canada (e-mail: csljc@ieee.org;jcliu@cs.sfu.ca).

B. Li is with the China Motion Telecom Group, Ltd., China (e-mail: bin.li@chinamotion.com).

B. Li is with the Department of Computer Science, The Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong (e-mail: bli@cs.ust.hk).

X.-R. Cao is with the Department of Electrical and Electronic Engineering, The Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong (e-mail: eecao@ust.hk; eecao@ee.ust.hk).

Digital Object Identifier 10.1109/TWC.2004.837394

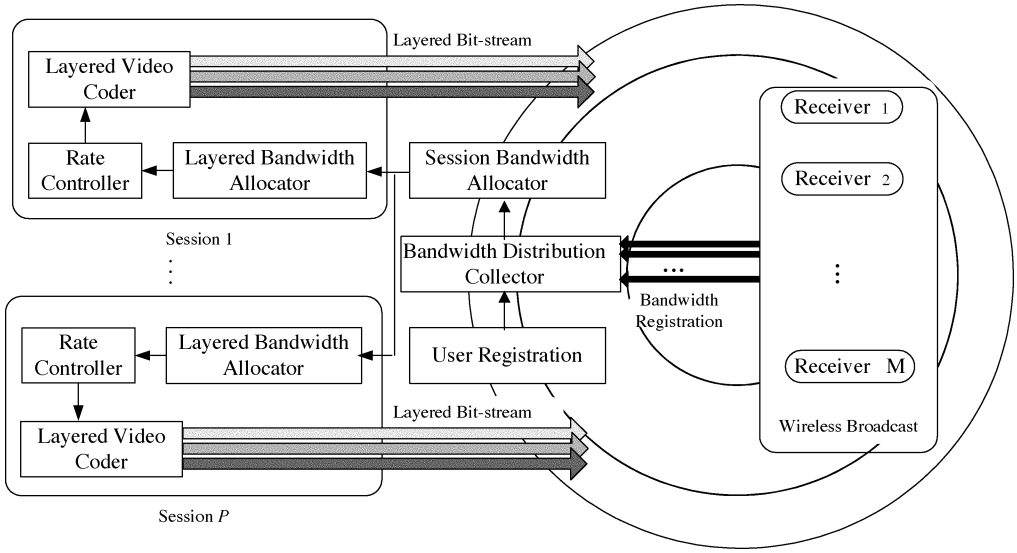


Fig. 1. System diagram.

the sender's side, especially considering that the bandwidth distribution of the receivers is not totally chaotic. Most wireless access enabled receivers use some typical hardware platforms, such as cellular phones, PDAs, and laptops; also the 3G wireless networks standard provides a series of access rates for different receivers, including 8, 64, 144, and 384 Kb/s, etc. [1].

In this paper, we formally investigate the above issues for noncumulative layered broadcasting. More explicitly, we seek answers to the following two important issues: 1) given a total bandwidth budget of the layers, what should be the optimal bandwidth for each layer? and 2) given the layer bandwidth allocation, to which subset of layers a receiver should subscribe?

Our main contributions in this paper can be summarized as follows.

1) We formulate the problem of optimal layer bandwidth allocation for the sender and the problem of optimal layer subscription for each receiver, both with the objective of maximizing a fairness index, a common objective for broadcasting protocols [12], [15].
2) We show an effective solution to the receiver subscription problem. For the sender bandwidth allocation problem, we find that it is computationally intractable and hence propose a set of heuristic algorithms from different perspectives.

We also carry out extensive studies on the performance of the heuristic algorithms under various settings. The results show that all of them offer significant improvements over nonadaptive allocation schemes and achieve nearly optimal performance in terms of the expected fairness index (EFI) for the receivers. Meanwhile, their computation overheads are kept at low levels. The impact of the key factors, including receiver bandwidth distribution, session bandwidth, and layer number, are also examined.

The rest of the paper is organized as follows. In Section II, we define the system model. The optimal adaptation problem is formulated in Section III with discussions on its complexity. Section IV presents a set of heuristics for the sender-based allocation. Their performances are studied and compared in Section V. Some related work is discussed in Section VI. Finally, Section VII concludes the paper.

II. System Model and Definitions

Fig. 1 illustrates our system model, in which a set of video programs is distributed from a central access point, i.e., a base station or mobile switching center. Each program is encoded using a layered video coder to noncumulative layers, which are then delivered over some broadcast channels. The video coders are either located in the access point or connected to it through wired links. As in many previous studies, we assume that the wireless bandwidth is much more valuable than that of wired links and thus becomes a dominant constraint in the overall system optimization.

The central broadcast point also performs management functions such as user registration and authentication. Moreover, a video program guide is sent to all receivers via a dedicated broadcast channel. A receiver who is interested in a particular program should first send a request to the central point, along with a description of its capability (i.e., its bandwidth requirement). Upon admission into a video session, the receiver can subscribe to a set of layers commensurate with its requirement.

A video program and its receivers constitute a *broadcast session* or *session* for short. We assume that an external protocol is used to allocate the total bandwidth among sessions (see [2] for example) and hence focus on the adaptation problem within a single session. For each receiver, the adaptation problem is to find an optimal layer subscription strategy. Note that a receiver cannot subscribe to a fraction of a layer; the adaptation granularity on the receiver's side is thus at a layer level. This could result in a mismatch between a receiver's bandwidth requirement and the total bandwidth of the subscribed video layers. To minimize such a mismatch, we also let the central point be adaptive in allocating the bandwidth of the layers. The problem is thus to find an optimal allocation according to the receivers' requirements.

TABLE I
PARAMETERS OF THE SYSTEM MODEL

Notation	Description
N	Session bandwidth
L	Total number of layers
r_k	Bandwidth of layer k
R	Layer bandwidth allocation, $R=(r_1,r_2,\ldots,r_L)$
M	Total number of receivers in the session
b_i	Bandwidth requirement of receiver i
B	Bandwidth requirements of the receivers, $B=(b_1,b_2,\ldots,b_M)$
S	Set of all layers, $S=\{1,2,\ldots,L\}$
S_i	Subscription of receiver i
$FI_R(S_i)$	Fairness index of receiver i with allocation R; $FI_R(S_i)=b_i^{-1}\sum_{k\in S_i} r_k$

III. PROBLEM FORMULATION

A. Problem Formulation

Our model can be formally characterized by a 5-tuple: (N, L, R, M, and B). Here, N is the session bandwidth, which imposes an upper bound for the total bandwidth of the layers for the video program (session); L is the total number of layers, which is constant for a practical encoder; $R = (r_1, r_2, \ldots, r_L)$, where r_k is the bandwidth of layer k, $\sum_{k=1}^{L} r_k \le N$, and R is referred to as a *layer bandwidth allocation* or simply an *allocation*; M is the total number of receivers in the session; and $B = (b_1, b_2, \ldots, b_M)$, where b_i is the bandwidth requirement of receiver i. The major notations used in our model are summarized Table I.

In our system, a bandwidth can take only discrete values, for two reasons. First, in a wireless network, bandwidth allocation is channelized; second, the output bandwidth of an MD video coder is discrete, given that there are only a finite number of quantizers in compression. For simplicity of exposition, we assume a *channel* is the minimum allocation unit, and a bandwidth is always expressed in the number of channels.

Let S be the set of all layers $S = \{1, 2, \ldots, L\}$. A *subscription* of a receiver is a subset of S, and the *subscription bandwidth* is the total bandwidth of the layers in this subscription, i.e., $\sum_{k\in S} r_k$. Assume $S_i \subseteq S$ is the subscription of receiver i, its utility is defined as $b_i^{-1}\sum_{k\in S_i} r_k$. This utility definition is also referred to as a *fairness function or fairness index* (FI) of the receiver [12], [15], and we therefore use $FI_R(S_i)$ to denote it.[1] In general, the bandwidth requirement of a receiver is determined by its capability or capacity, and it cannot subscribe to layers that exceed this physical constraint. The subscription problem for receiver i thus can be described as

$$(\text{OPT}-\text{RECV})\quad \text{Maximize}\quad FI_R(S_i)=b_i^{-1}\sum_{k\in S_i} r_k$$
$$\text{Subject to}\ S_i\subseteq S \text{ and } \sum_{k\in S_i} r_k \le b_i. \quad (1)$$

[1]It is worth noting that there are many other utility definitions in the literature [12], and the choice actually depends on a number of factors, such as the encoding algorithm and the transmission scheme, and, more importantly, the design objective of the system. Although we focus on fairness index in this paper, our optimization algorithms are generally enough to accommodate many other utility definitions.

Given an allocation $R = (r_1, r_2, \ldots, r_L)$, the solution to the above problem, or the *optimal subscription*, can be denoted as

$$S_i^*(R) = \arg_{s_i}\left\{\max_{s_i\subseteq S}[FI_R(S_i)], \sum_{k\in S_i} r_k \le b_i\right\} \quad (2)$$

and the corresponding fairness index is $FI_R(S_i^*) = b_i^{-1}\sum_{k\in S_i^*} r_k$.

The allocation problem for the sender is, given requirements $B = (b_1, b_2, \ldots, b_M)$ of the receivers, to find the allocation $R = (r_1, r_2, \ldots, r_L)$, $\sum_{k=1}^{L} r_k \le N$ such that the EFI of all the receivers is maximized, that is

$$(\text{OPT}-\text{SEND})$$
$$\text{Maximize}\quad \text{EFI}(R) = \frac{1}{M}\sum_{i=1}^{M} FI_R(S_i^*)$$
$$\text{Subject to}\quad \sum_{k=1}^{L} r_k \le N$$
$$S_i^*(R) = \arg_{s_i}\left\{\max_{s_i\subseteq S}[FI_R(S_i)], \sum_{k\in S_i} r_k \le b_i\right\}. \quad (3)$$

B. Complexity of the Problem

Theorem 1: OPT-RECV, in its general form, is NP-hard.

Proof: For OPT-RECV, maximizing $b_i^{-1}\sum_{k\in S_i} r_k$ is equivalent to maximizing $\sum_{k\in S_i} r_k$ for the given b_i. The latter is essentially an optimization version of the 0/1 knapsack problem, which is known to be NP-hard [9].

An implication in the above proof is that the integer (channelized) representation of a subscription bandwidth is not bounded; hence, each subset of the layers would have a distinct total bandwidth. In practice, however, the total bandwidth of the layers is at most N, which is limited by the system. Hence, though the number of the subsets of S grows exponentially with L, the total number of distinct subscription bandwidths is at most N. For small or moderate N, the algorithm shown in Fig. 2 can thus be used to solve this constrained OPT-RECV problem.

Input:
R, b_i;
Initialization:
1: $V_1 := \{r_1\}$;
Calculating all possible subscription bandwidths
2: **for** k:=2 **to** L **do** /* Try each layer, either subscribe to or not */
3: set $V_k := V_{k-1} \cup \{r_k\}$;
4: **for** each $r \in V_{k-1}$ **do** $V_k := V_k \cup \{r + r_k\}$.
Output:
5: $t := \max\{r : r \in V_L \cup \{0\}, r \le b_i\}$; /* Best-matching subscription bandwidth*/
6: $FI_R := t / b_i$. /* Corresponding fairness index */

Fig. 2. Algorithm for OPT-RECV with bounded integer bandwidth.

The idea of this algorithm is to find set V_L that contains all possible subscription bandwidths using layers 1 through L (lines 1–4). The optimal subscription bandwidth for receiver i is thus $\max\{r : r \in V_L, r \le b_i\}$ (line 5–6).

Since $|V_L| \le \sum_{k=1}^{L} r_k \le N$ for a bounded integer bandwidth, the time complexity for calculating V_L is $O(LN)$, and that for FI_R is $O(N)$ by using a bitmap as the data structure for sets. The complexity of this algorithm is thus bounded by $O(LN + N) = \mathrm{O}(LN)$. Once the bandwidth for the optimal subscription is calculated, the corresponding subscription $S_i^*(R)$ can be easily obtained by back tracking the iteration.

The solution to problem OPT-RECV is only a subroutine for solving problem OPT-SEND. To solve OPT-SEND, we need to check different layer bandwidth allocations to find the optimal one. For each allocation, we need to calculate set V_L only once and then find the optimal subscriptions for the receivers based on V_L. This requires $\mathrm{O}(LN + MN)$ time for an M-receiver session. To improve it, we can precompute the distribution for the bandwidth requirements of the receivers; since there are at most N different subscription bandwidths and the receivers with the same requirement always have the same optimal subscription bandwidth, all the optimal subscription bandwidths can be found in time $O(N)$. The time to calculate EFI of a session for each allocation thus remains $O(LN)$, which is independent of session size.

Nevertheless, an allocation R is essentially a partition of an integer into exactly L positive integer parts; the total number of possible allocations is given by the *partition function* $P(N, L)$, which is known to grow exponentially with L even for a bounded session bandwidth [11]. Therefore, though for each allocation we can check its EFI efficiently, the problem of OPT-SEND remains intractable.

IV. Heuristics for Layer Bandwidth Allocation

A brute-force approach to problem OPT-SEND needs an exponential time to exhaustively search the allocations, which is impractical for real-time adaptation. We therefore resort to approximate solutions. In this section, we propose three heuristic algorithms and discuss their design rationales.

A. Cumulative Layering Based Allocation (CLA)

It is known that for cumulative layered broadcasting, layer bandwidths can be optimally allocated in time $O(LN^2)$, given the same settings as in Section III [10]. The key point in this optimal algorithm is that if we add layers one by one, starting from the base layer, the cumulative layer bandwidth monotonically increases; thus, only the receivers subscribing to the highest layer in the previous step have the potential of subscribing to the new layer. Hence, we can divide the problem into L stages and then apply dynamic programming with each stage being solvable in $O(N^2)$ time. Unfortunately, as a generalization of cumulative layering, noncumulative layering does not have this optimization structure because the layers are not subscribed consecutively—when a layer is added, the bandwidth of a subscription containing this new layer does not necessarily raise the existing possible subscription bandwidths.

Nevertheless, as a special case of noncumulative layering, the optimal allocation for cumulative layering can serve as a heuristic to our problem. That is, assuming $R' = (r'_1, r'_2, \ldots, r'_L)$ is the optimal layer bandwidth allocation for cumulative layering under the same setting of (N, L, M, B), we simply use R' as the allocation to the noncumulative layering case.

In existing multiple description coders (MDC), a simple uniform allocation R^u is often used [7], where each layer has the same bandwidth. We now have the following observation.

> In noncumulative layered broadcasting, we have $\mathrm{EFI}(R') \ge \mathrm{EFI}(R^u)$ when using R' and R^u as the allocation, respectively. That is, using R' as a heuristic to our problem is no worse than using the uniform allocation R^u.

This is simply because, in the uniform allocation, all layers have the same contribution, and thus any subscription policy yields the same EFI as the cumulative subscription policy. Although the uniform allocation has been widely used, the use of R' as an approximation in this case is clearly a better alternative.

B. Merge-Based Allocation (MBA)

Assuming there is no constraint on the number of layers, the use of N layers each with a unit bandwidth (one channel) is obviously an optimal allocation because this yields the finest adaptation granularity for any receiver. A simple merge-based

```
Initialization:
  R*_L := (r_1, r_2, ..., r_L), where r_k = 1, k = 1,2,...,L;
  /* R*_L is a temporal variable representing the allocation vector of the highest EFI so far
achieved */
Merging:
  for T:=L+1 to N do {          /* try all possible total layer bandwidths */
    R_T := (r_1, r_2, ..., r_T), where r_k = 1, k = 1,2,...,T;
    K := T;
    while K > L do {            /* iteratively merge layers */
      K := K - 1;   Δ_EFI := ∞;
      for i := 1 to K do        /* find two candidates for merging */
        for j := i + 1 to K+1 do {
          R' := R_{K+1};
          Merge layers i and j in R';
          if EFI(R_{K+1}) - EFI(R') < Δ_EFI
          then { Δ_EFI = EFI(R_{K+1}) - EFI(R'), i* = i, j* = j; }
        }
      R_K = R_{K+1};
      Merge layers i* and j* in R_K;
    } /* end of while loop */
    if EFI(R_L) > EFI(R*_L) then R*_L := R_L;
  }
Output:
  R := R*_L;        /* Layer bandwidth allocation */
```

Fig. 3. Pseudo-code for the MBA.

heuristic can start from this unconstrained case and iteratively merge layers until the number of layers is reduced to L. In each step, two layers that result in the smallest EFI degradation are selected for merging. Given that the initial number of layers is set to N, the total layer bandwidth of the resultant allocation is always N for this greedy merging process. As we will see in the next section, this is not always the best setting and sometimes leads to noticeable performance degradation as compared to the use of a somewhat lower bandwidth. Therefore, in our MBA algorithm, we apply the above merging process for different initial numbers of layers (from L to N) and choose the one that yields the highest EFI. A pseudo-code for this algorithm is presented in Fig. 3.

In MBA, there are $T - L$ iterations (the *while* loop) for each possible total layer bandwidth (T), where the kth iteration requires $O(K^2LT)$ time to find two locally optimal layers for merging. Since N is generally much larger than L, the total time complexity of this algorithm is bounded by $O(LN^5)$. In practice, however, it can be greatly reduced by an observation that many entries in R_K could be identical; thus, only one of them needs to be considered as a candidate for merge.

C. Multigranular Allocation (MGA)

A brute-force algorithm for our allocation problem suffers high computation complexity. However, if the allocation unit is coarse, the execution time can still be practically acceptable. The MGA heuristic allocates layer bandwidths in a multigranular fashion, starting from a coarse-grained allocation, and then locally adjusting the bandwidths of its layer at a finer granularity to achieve a higher EFI.

A pseudo-code for this MGA algorithm and its subroutines is shown in Fig. 4. First, given a coarse allocation unit of $m(> 1)$ channels, we find an optimal allocation R' using a brute-force algorithm (this can be easily implemented using a depth-first search or dynamic programming, and the subroutine for this purpose is referred to as brute-force-optimization (m)). Clearly, R' is not necessarily optimal with the finer allocation unit of one channel. Thus, in the second step, we adjust the bandwidth of each layer of R' in a limited range (or so-called "locally") with an adjustment unit of one channel, and the resultant allocation that yields the highest EFI is set as the solution. Formally speaking, this step is to solve the problem as follows:

> Given $R' = (r'_1, r'_2, \ldots, r'_L)$, find $R^* = (r'_1 + \Delta_1, r'_2 + \Delta_2, \ldots, r'_L + \Delta_L)$, $\Delta_i \in [-p, p]$, $i = 1, 2, \ldots, L$, such that EFI(R^*) is maximized.

Here, $[-p, p]$ is the range of local adjustment for each layer. In our implementation, we let p be equal to m. Subroutine local-optimization (R', m) is used in for this step, which further invokes local_adjustment (l, R', m) to find R^* using a depth-first search.

Our experience shows that for $L < 10$ and $m = N/10$, the execution time of the MGA algorithm is practically acceptable

Auxiliary Variables:

R', R^*, m

/* m specifies a coarse-grained allocation unit */

Multi-Granular Allocation

Step 1: Coarse-granular optimal allocation:

R' := Brute-Force-Optimization (m);

Step 2: Fine-Granular local optimization:

R^* := Local-Optimization (R', m);

Output:

R := R^*; /* Layer bandwidth allocation */

Fig. 4. Pseudo-code for the MGA.

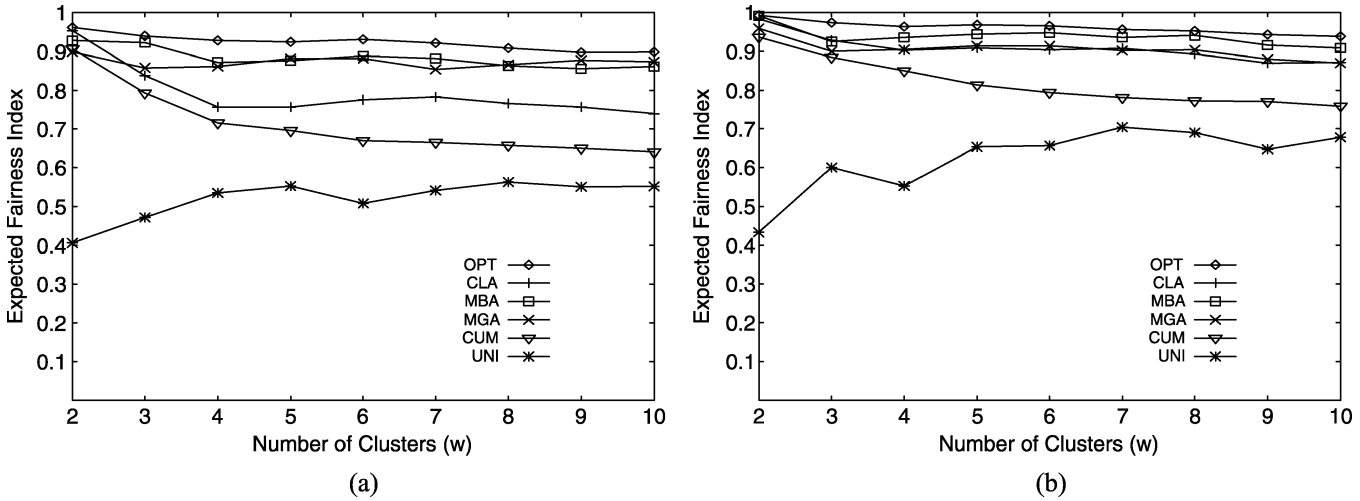


Fig. 5. EFIs for different bandwidth distributions. (a) $L = 3$. (b) $L = 4$. Acronyms: OPT—optimal allocation, UNI—uniform allocation, CLA—cumulative layering based allocation, MBA—merge-based allocation, MGA—multigranular allocation, and CUM—cumulative layering with optimal allocation.

(<30 ms). The performance of the MGA algorithm with this setting is also reasonably good, as demonstrated in the next section.

V. Numerical Results

In this section, we present numerical results for the bandwidth adaptation algorithms for noncumulative layered broadcasting. Our main objective is to investigate the key factors that influence the EFI, including the receiver bandwidth distribution (B), session bandwidth (N), and the number of layers (L), as well as to identify the tradeoffs among the heuristic allocation algorithms.

A. System Configurations

To reflect the heterogeneous nature of wireless devices, we assume that the receivers' bandwidth distribution model consists of w clusters, each following a Gaussian distribution. In our study, the minimum and maximum receiver bandwidths are 2 and 128 channels, respectively. This order of magnitude for access bandwidth covers that of a broad spectrum of network access techniques as well as MDC video coders. The standard deviation of a cluster is set to 10% of the cluster mean. Therefore, most bandwidth differences are within $\pm 10\%$, yet a few reach about $\pm 40\%$ or more, which reflects the flexibility in device design. We assume that a session has 200 receivers and thus draw 200 samples from the model to obtain a distribution of the receivers' bandwidth requirements, or *receiver bandwidth distribution* for short.

For the sake of comparison, we also implement the *uniform allocation* (UNI), which is widely used in theoretical as well practical studies [7], [10]. For small numbers of layers ($L = 3$ and 4), the optimal allocations (OPTs) are calculated using a brute-force algorithm as well. Therefore, both the performance gains (compared to the uniform allocation) and degradations (compared to the optimal allocation) of the heuristics can be observed.

For the MGA, we use an allocation unit (m) of six channels for the coarse-grained allocation. This yields an execution time of no more than 30 ms, a reasonable time for real-time adaptation.

B. Effect of Receiver Bandwidth Distribution

Fig. 5 shows the EFI for the allocation algorithms under different receiver bandwidth distributions. It can be seen that the performances of the adaptive algorithms (OPT, CLA, MBA, and MGA) are reasonably good for all the distributions. In particular, the performances of MBA and MGA are quite close to the optimal EFI, so is CLA with four layers $(L = 4)$. All of

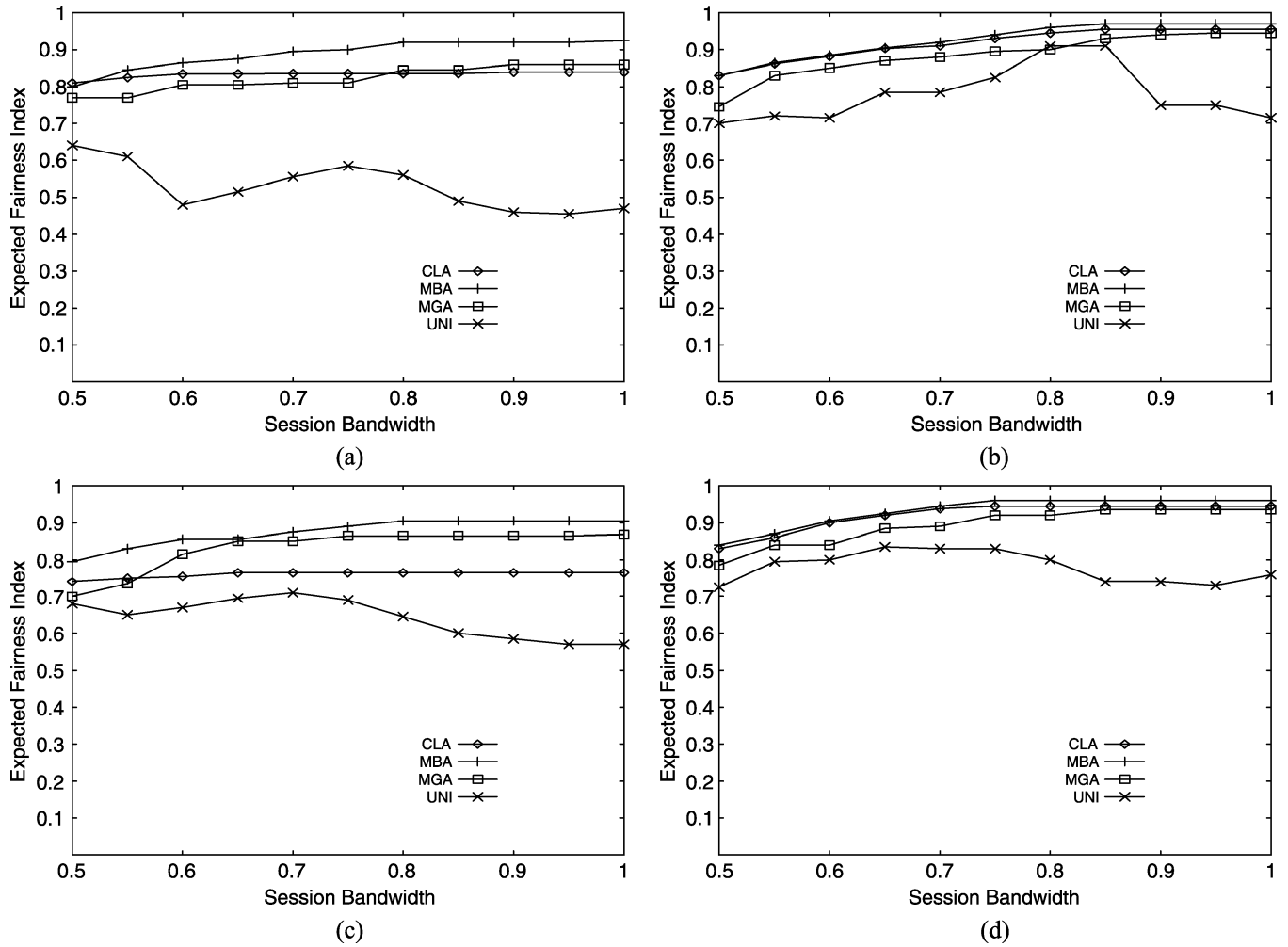


Fig. 6. EFI as a function of the session bandwidth. Session bandwidth is normalized by 192 (1.5 times of the maximum receiver bandwidth). (a) Distribution-1, $L = 3$. (b) Distribution-1, $L = 6$. (c) Distribution-2, $L = 3$. (d) Distribution-2, $L = 6$.

them significantly outperform the nonadaptive algorithm UNI; the gap is constantly over 0.2 and often higher than 0.4.

Fig. 5 also shows the EFI for cumulative layered broadcasting with the optimal allocation under the same conditions (referred to as CUM). Note that CLA uses the same allocation as CUM but has more flexible layer subscription policy; hence, it achieves much better performance. The above results imply that noncumulative layering can effectively improve the adaptation granularity and, yet, an adaptive and receiver-aware allocation is necessary.

When the number of clusters in the bandwidth distribution increases, the EFI for the optimal algorithm decreases, which is also the general trend for the three heuristic algorithms. Intuitively, when the number of clusters is large, the generated distribution resembles a uniform distribution, while the clustering behavior becomes less obvious; it is therefore difficult for the adaptive algorithms to adjust layer bandwidths to match the uniformly distributed requirements. On the other hand, the uniform allocation tends toward improving, though it remains much worse than the adaptive schemes.

In the following parts, we further investigate the impact of session bandwidth and layer number and use two receiver bandwidth distributions as representatives: Distribution-1 ($w = 3$ clusters) and Distribution-2 ($w = 9$ clusters).

C. Effect of Session Bandwidth

The relationship between the EFI and the session bandwidth is shown in Fig. 6. For all the bandwidths, the adaptive algorithms perform better than the uniform allocation algorithm, and, in general, MBA is the best among them. Moreover, in many cases, the EFIs saturate after a certain bandwidth, suggesting that the bandwidth of each layer, hence, their total bandwidth, is not necessarily the higher the better for heterogeneous receivers.

Such a phenomenon of saturation also implies that the total bandwidth of an optimal allocation could be less than a given session bandwidth. This explains why in MBA we should check all possible total layer bandwidths that are less than the given session bandwidth. Since we always choose the total layer bandwidth that yields the highest EFI, the EFI of MBA is nondecreasing with the increase of session bandwidth. CLA and MBA have inherent heuristics to find an appropriate total layer bandwidth, which do not greedily occupy the whole session bandwidth; hence, their EFI curves are nondecreasing as well. On the other hand, UNI simply divides a session bandwidth uniformly; its EFI not only is much lower than that of the adaptive schemes but also can be lower with a higher bandwidth than that with a lower bandwidth (see Fig. 6).

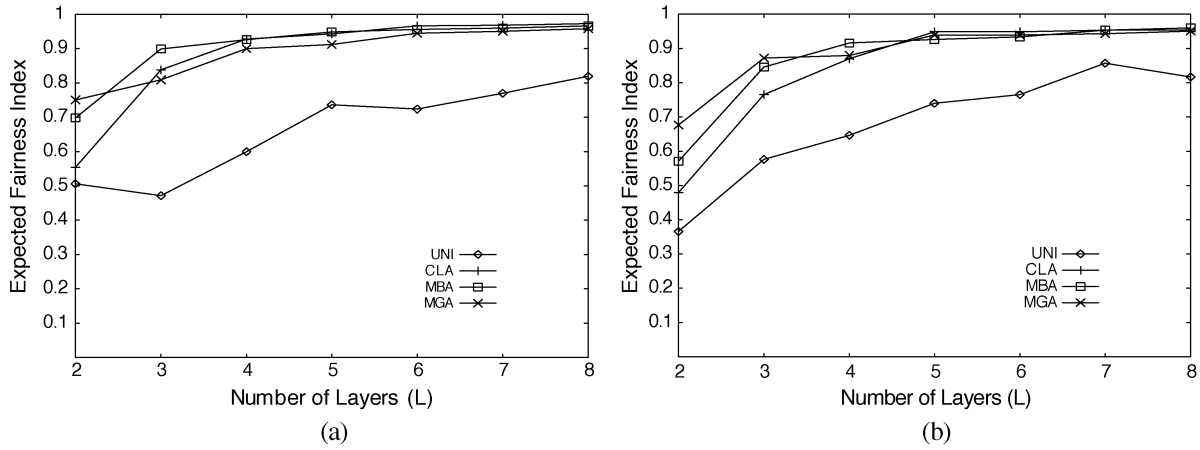


Fig. 7. EFI as a function of the layer number. (a) Distribution-1. (b) Distribution-2.

D. Effect of the Number of Layers

Fig. 7 shows the EFI as a function of the number of layers. The results again validate our observations that MBA is generally better than the other two heuristics, and yet all of them significantly outperform the uniform allocation algorithm. In addition, we find that the EFIs of the adaptive algorithms are generally improved with an increase of the number of layers. In fact, it can be formally proved that the EFI is *nondecreasing* for an optimal allocation algorithm, because the adaptation granularity for a receiver can always be improved by dividing one layer into two. This is also true for MBA as the merging process reduces EFI, and the use of more layers means less merges are needed. For CLA and MGA, though this property is not theoretically valid, it does exhibit in our empirical data. More importantly, for these adaptive algorithms, satisfactory performance can be achieved by using only a limited number of layers, say 4 to 5.

These nice properties, however, do not hold for the nonadaptive algorithm UNI, due to its unawareness of the bandwidth distribution of the receivers. For example, in Fig. 7(a), the EFI of UNI with six layers is evidently lower than that with only five layers; a similar situation happens in Fig. 7(b) when comparing the EFI with seven layers and that with eight layers. In addition, the EFI of UNI is remarkably lower than that of all the heuristic algorithms, and the gap is over 0.3 for some small value of L. Although we can expect that such a gap shrinks with an increase of L, it remains over 0.1 for eight layers, which is far from satisfactory. Since it is difficult for state-of-the-art MDC coders to generate a large number of layers, and the use of more layers also increases the overhead for network management, we believe that our adaptive allocation is an effective tool to improve the fairness for noncumulative layered video broadcasting.

E. Computational Overheads

Finally, we investigate the computational overheads of the allocation algorithms. We implemented the three heuristic algorithms as well as the brute-force optimal algorithm using C++. Fig. 8 shows the computation times for the implemented algorithms running on our PC (Pentium 4, 1.2 GHz, 512-M memory).

From Fig. 8, we can see that the computation overhead of the brute-force (optimal) algorithm significantly increases when increasing the number of layers. This is because the algorithm needs

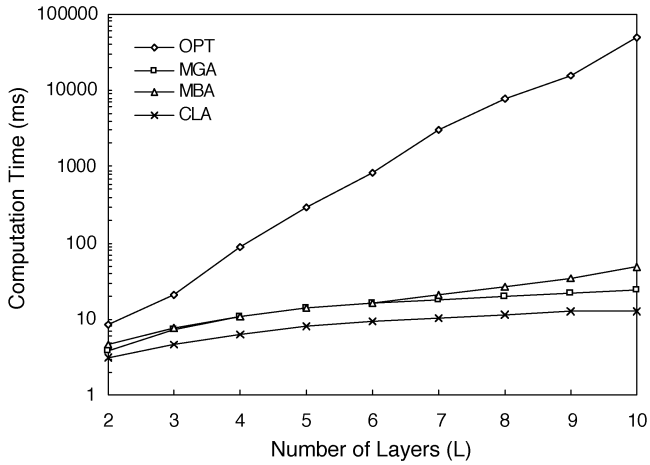


Fig. 8. Computational overheads as a function of number of layers for different allocation algorithms. Session bandwidth= 128 channels. For MGA allocation, the allocation unit for the coarse-granular allocation is 12 channels ($\approx$128/10).

to check all possible allocations, each with $O(LN)$ time, and their total number grows very fast even if some branch-and-bound techniques are used [13]. For eight layers, its computation time is over 10 s, though there is no other background task in our PC. It is worth noting that the central access point would have to support concurrent video sessions as well as perform many other operations, and the allocation algorithm is to be invoked periodically. Hence, the computational overhead of the brute-force algorithm is still too high to be afforded. On the other hand, our heuristics are reasonably fast (generally less than 30 ms) yet achieve nearly optimal EFI (close to one). In view of these, we believe that it is worth using such heuristics, at least at the current stage.

VI. Related Work

Due to its potentials in media delivery over error-prone networks, noncumulative or MD coding has received much attention recently; see [16] for a comprehensive survey. For MD video streaming over the Internet, Apostolopoulos *et al.* [8], [17] studied the use of path diversity to achieve robust transmission for a customized MD coder. Padmanabhan *et al.* [14]

proposed an MD video-based streaming protocol to accommodate the dynamics of overlay networks. Nahrstedt *et al.* [18] designed a three-dimensional subband MD coder and proposed a modified transmission control protocol for streaming the video. The protocol, called dubbed R-TCP, collects network state information for adaptive source and channel rate control. It was later extended to the context of broadcast with the objective of reducing transmission errors while not handling bandwidth heterogeneity [19].

For video multicast or broadcast to heterogeneous receivers, existing work has focused on the use of cumulative layering [3], [4], [10]. In this case, a greedy subscription policy yields the optimal subscription for each receiver; efficient and optimal layer rate allocation algorithms on the sender's side have also been developed, as described in Section IV. The basic idea of the adaptation algorithms for cumulative layering, such as receiver-driven adaptation, can be used in the noncumulative case as well [5], [6]. Since noncumulative layering is more flexible, higher bandwidth utilization can be achieved for receivers. Specifically, Byers *et al.* [6] demonstrated that a Fibonacci-like layer bandwidth allocation for noncumulative layering enables fine-grained receiver adaptation as well as minimizes the layer join or leave operations for receivers in a dynamic environment. However, they did not specifically target video distribution; the constraints of session bandwidth and layer number for practical video coders were not considered. On the other hand, existing studies on MD coding often assume that all layers are equally important and a uniform bandwidth allocation is thus adopted [7]. Although the MD coding with nonuniform bandwidth allocation has been shown to be feasible [8], such flexibility in broadcasting has seldom been advocated.

VII. Conclusion

This paper presented a formal study on the problem of bandwidth adaptation for noncumulative layered video broadcasting. Our objective is to improve user satisfaction in a broadcast session by employing an optimal layer subscription on the receiver's side as well as an optimal layer bandwidth allocation on the sender's side. We formulated these two optimization problems and presented effective algorithms for both of them. Their performances were evaluated and compared under various network configurations. Our main results of the evaluation are as follows.

1) All the heuristic algorithms significantly outperform the UNI, in terms of the EFI for a session.
2) MBA is usually better than MGA and CLA, though the difference is not significant. In general, their performance is quite close to the optimal allocation.

To deploy our adaptation framework in a dynamic environment where both the session bandwidth and the receivers' status change over time, a critical concern is its computation overhead. We found that the execution times for the algorithms are generally less than 30 ms, which is reasonably fast for real-time adaptation. In addition, our algorithms are based on the bandwidth distribution of all the receivers, while not the bandwidth of an individual receiver. Therefore, to minimize the perceptually annoying bandwidth fluctuation, a reallocation is needed only when the distribution has substantially changed.

References

[1] Y.-B. Lin and I. Chlamtac, *Wireless and Mobile Network Architectures*. New York: Wiley, 2001.

[2] J.-Y. Jeng and Y.-B. Lin, "Equal resource sharing scheduling for PCS data services," *ACM/Kluwer J. Wireless Networks*, vol. 5, no. 1, pp. 41–45, Jan. 1999.

[3] S. McCanne, V. Jacobson, and M. Vetterli, "Receiver-driven layered multicast," in *Proc. ACM SIGCOMM* , Aug. 1996, pp. 117–130.

[4] Y. Yang, M. Kim, and S. Lam, "Optimal partitioning of multicast receivers," *Proc. IEEE ICNP*, 2000.

[5] T. Kim and M. H. Ammar, "A comparison of layering and stream replication video multicast schemes," in *Proc. NOSSDAV*, July 2001.

[6] J. Byers, M. Luby, and M. Mitzenmacher, "Fine-grained layered multicast," in *Proc. IEEE INFOCOM*, Apr. 2001.

[7] Y. Wang and Q.-F. Zhu, "Error control and concealment for video communication: A review," *Proc. IEEE*, vol. 86, pp. 974–997, May 1998.

[8] J. G. Apostolopoulos and S. Wee, "Unbalanced multiple description video communication using path diversity," in *Proc. IEEE Int. Conf. Image Processing*, Oct. 2001.

[9] R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*. New York: Freeman, 1979.

[10] J. Liu, B. Li, and Y.-Q. Zhang, "An end-to-end adaptation protocol for layered video multicast using optimal rate allocation," *IEEE Trans. Multimedia*, 2002.

[11] D. S. Mitrinovic, J. Sandor, and B. Crstici, *Handbook of Number Theory*: Kluwer, 1996.

[12] J. Liu, B. Li, and Y.-Q. Zhang, "Adaptive video multicast over the internet," *IEEE Multimedia*, vol. 10, no. 1, pp. 22–31, Jan./Feb. 2003.

[13] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*, 2nd ed. Cambridge, MA: MIT Press, 2001.

[14] V. N. Padmanabha, H. J. Wang, P. A. Chou, and K. Sripanidkulcha, "Distributing streaming media content using cooperative networking," in *Proc. NOSSDA*, May. 2002.

[15] T. Jiang, E. Zegura, and M. Ammar, "Inter-receiver fair multicast communication over the internet," in *Proc. NOSSDAV*, June 1999.

[16] V. K. Goyal, "Multiple description coding: Compression meets the network," *IEEE Signal Processing*, vol. 18, pp. 74–93, Sept. 2001.

[17] J. Apostolopoulos, T. Wong, S. Wee, and D. Tan, "On multiple description streaming with content delivery networks," in *Proc. IEEE INFOCOM*, June 2002.

[18] K. Nahrstedt and S. D. Servetto, "Video streaming over the public internet: Multiple description codes and adaptive transport protocols," in *Proc. IEEE Int. Conf. Image Processing (ICIP)*, Oct. 1999.

[19] S. D. Servetto and K. Nahrstedt, "Broadcast-quality video over IP," *IEEE Trans. Multimedia*, vol. 3, pp. 162–173, Mar. 2001.

[20] Y. Wang and S. Li, "Error-resilient video coding using multiple description motion compensation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, pp. 438–452, June 2002.

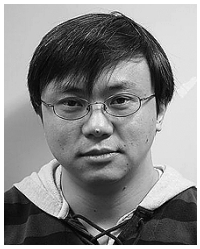
Jiangchuan Liu (S'01-M'03) received the B.Eng degree (*cum laude*) from Tsinghua University, Beijing, China, in 1999, and the Ph.D. degree from The Hong Kong University of Science and Technology in 2003, both in computer science.

He is currently an Assistant Professor in the School of Computing Science, Simon Fraser University, BC, Canada, and was an Aassistant Professor at The Chinese University of Hong Kong from 2003 to 2004. He is a co-inventor of one European patent (granted) and two U.S. patents (pending). His research interests include Internet architecture and protocols, media streaming, wireless *ad hoc* networks, and service overlay networks.

Dr. Liu was a recipient of a Microsoft research fellowship (2000) and a Hong Kong Young Scientist Award (2003). He won first-class honors in several regional and national programming contests. He serves as TPC member for various networking conferences, including the IEEE INFOCOM'04 and '05. He was TPC Co-Chair for The First IEEE International Workshop on Multimedia Systems and Networking (WMSN'05), Information System Co-Chair for IEEE INFOCOM'04, and a Guest Editor for the ACM/Kluwer *Journal of Mobile Networks and Applications* (MONET) special issue on energy constraints and lifetime performance in wireless sensor networks. He is a member of Sigma Xi.

Bin Li (S'96-M'97-SM'03) received the B.Eng degree in automatic control from Huazhong University of Science and Technology, Wuhan, China, in 1991, and the M.Phil. and Ph.D. degrees in electrical and electronic engineering from The Hong Kong University of Science and Technology, Hong Kong, in 1996 and 2003, respectively.

Between 1991 and 1994, he worked in the Technology Center, Guangdong Branch, China Telecom where he was the key member in designing the first online navigation tool for China Telecom. Since July 1997, he has been with China Motion Telecom, one of the main telecom service providers in Hong Kong and China, where he is the Executive Director and Chief Operating Officer. His research interests include traffic engineering in mobile cellular networks and multimedia applications in wireless networks. He has published 20 paper in IEEE journals and conference proceedings.



Bo Li (S'89-M'92-SM'99) received the B.S. (*summa cum laude*) and M.S. degrees in computer science from Tsinghua University, Beijing, China, in 1987 and 1989, respectively, and the Ph.D. degree in computer engineering from the University of Massachusetts, Amherst, in 1993.

Between 1994 and 1996, he worked on high performance routers and ATM switches at the IBM Networking System Division, Research Triangle Park, NC. Since then, he has been with the Computer Science Department, The Hong Kong University of Science and Technology, Hong Kong, where he is now an Associate Professor. He is also an Adjunct Researcher at Microsoft Research Asia (MSRA). His current research interests include wireless mobile networking supporting multimedia, video multicast, and all-optical networks using WDM. He coauthored the first paper on proxy server placement in 1999.

Dr. Li has been on the editorial board for the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, the *ACM Journal of Wireless Networks* (WINET), IEEE JOURNAL OF SELECTED AREAS IN COMMUNICATIONS (JSAC)—Wireless Communications Series, *ACM Mobile Computing and Communications Review* (MC2R), *SPIE/Kluwer Optical Networking Magazine* (ONM), and *KICS/IEEE Journal of Communications and Networks* (JCN). He served as a Guest Editor for the *IEEE Communications Magazine* special issue on active, programmable, and mobile code networking (April 2000), IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS special issue on protocols for next generation optical wdm networks (October 2000), *ACM Performance Evaluation Review* special issue on mobile computing (December 2000), and *SPIE/Kluwer Optical Networks Magazine* special issue on wavelength routed networks: architecture, protocols and experiments (January/February 2002). Currently, he is the Lead Guest Editor for a special issue of IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS on recent advances in service-overlay network, and a Guest Editor for a special issue of IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS on *ad hoc* networks. In addition, he has been involved in organizing over 30 conferences, especially the IEEE Infocom since 1996. He was the Co-TPC Chair for the IEEE Infocom'2004.

Xi-Ren Cao (SM'89-F'96) received the M.S. and Ph.D. degrees from Harvard University, Cambridge, MA, in 1981 and 1984, respectively.

He was a Research Fellow at Harvard University from 1984 to 1986. He then worked as a Principal and Consultant Engineer/Engineering Manager at Digital Equipment Corporation, until October 1993. Since then, he has been a Professor of the Hong Kong University of Science and Technology (HKUST), Hong Kong. He is the Director of the Center for Networking at HKUST. He owns three patents in data- and telecommunications and has published two books: *Realization Probabilities—The Dynamics of Queuing Systems* (New York: Springer Verlag, 1994) and *Perturbation Analysis of Discrete-Event Dynamic Systems* (New York: Kluwer, 1991), coauthored with Y. C. Ho. His current research interests include discrete-event dynamic systems, communication systems, signal processing, stochastic processes, and system optimization.

Dr. Cao received the Outstanding Transactions Paper Award from the IEEE Control System Society in 1987 and the Outstanding Publication Award from the Institution of Management Science in 1990. He is an Associate Editor at Large of IEEE TRANSACTIONS OF AUTOMATIC CONTROL, and he is/was a member of the Board of Governors of IEEE Control Systems Society, Associate Editor of a number of international journals, and Chairman of a few technical committees of international professional societies.