Multiple Description Video Multicast in Wireless Ad Hoc Networks

Shiwen Mao[†], Xiaolin Cheng[†], Y. Thomas Hou[†], Hanif D. Sherali[‡]
[†]The Bradley Department of Electrical and Computer Engineering
[‡]The Grado Department of Industrial and Systems Engineering
Virginia Tech, Blacksburg, VA 24061
{smao, xiaolin, thou, hanifs}@vt.edu

Abstract

We consider the problem of multicasting multiple description (MD) video in wireless ad hoc networks. We follow an application-centric, cross-layer approach with the objective of minimizing video distortion. The contribution of this paper is twofold. First, we propose a practical MD video multicast scheme that uses multiple trees to achieve an improved error resilience performance. The proposed scheme also takes into account highly diverse wireless link bandwidths by using scalable coding for each description, thus further improving the overall video quality. Second, we formulate the optimized multicast routing as a combinatorial optimization problem and propose an efficient Genetic Algorithm (GA)-based metaheuristic solution procedure. Performance comparison with existing approaches show significant gains for a wide range of network operating conditions.

1 Introduction

As progress in wireless ad hoc network research continues, there is a compelling need to support real-time multimedia services in such networks. In this paper, we study the important problem of multicasting *multiple description* (MD) video in wireless ad hoc networks. MD coding is a technique that generates multiple equally important streams, each giving a low, but acceptable video quality [10]. This decoding independence allows to reconstruct video from *any* subset of received descriptions, achieving a quality commensurate with the number of received descriptions. In addition, MD coding has the forward error correction (FEC) capability: it is possible to recover lost information in one stream using information carried in other received streams [3, 22], but without using any feedback mechanism. These features make MD video an excellent match for multimedia multicast in ad hoc networks, where links are unstable, reliable paths are hard to maintain, and feedback should be suppressed at best.

Multicast is typically implemented by creating a multicast tree rooted at the sender using multicast routing protocols [20, 21]. Although well studied for the Internet, research on multicast routing in wireless ad hoc networks is still in its early stage [14]. In particular, issues such as interference, mobility, frequent link failures, and topology changes have all made this research much more challenging. Since trees are minimally connected, any in-tree link or node failure will partition the tree into two disconnected subtrees. In addition, wireless links in ad hoc networks are much more diverse in terms of quality (e.g., available bandwidth, loss, and delay) than links in wireline networks: any link in an ad hoc network could be highly fragile with dynamic state conditions. Consequently, it is critical to investigate new methodologies for multicast routing in wireless ad hoc networks.

Multicasting MD video was first discussed in CoopNet [17] in the context of *application-level multicast* as a means to prevent web servers from being overwhelmed by a large "flash crowd." In CoopNet, clients form one or more distribution trees rooted at the server for live media streaming. Video is coded into multiple descriptions, each sent on a different tree, in order to reduce the disruption caused by node departures. CoopNet is quite effective in large-scale media multicasting, since it complements the client-server architecture (thus achieving the efficiency of centralized schemes) and exploits the unique strength in scalability of peer-to-peer networks. However, the CoopNet approach is not suitable for MD video multicast in wireless ad hoc networks for the following reasons. First, the main design objective of CoopNet is making servers robust to "flash crowds," with video quality as a secondary consideration. As a result, routing is performed by packing the clients into short and largely balanced trees, in which each tree edge is actually a (possibly large) number of network links. Such a logical link level routing approach cannot be easily translated into a physical-level link routing, which is the primary interest in this research. Second, CoopNet routing does not quantitatively address the routing problem and

Proceedings of the First International Conference on Broadband Networks (BROADNETS'04)
0-7695-2221-1/04 $ 20.00 IEEE

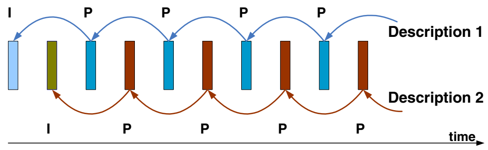

Figure 1. The MD coding scheme.

is not optimized in terms of video quality. Such an approach cannot be optimal in wireless ad hoc networks. Finally, the MD coding scheme used in CoopNet generates balanced descriptions. It does not consider the possibility that different trees may have different bandwidths. Since link qualities in ad hoc networks are highly diverse, such an approach may make the overall performance dependent on the quality of the worst tree. Moreover, in CoopNet, each description is not scalable (i.e., with a fixed rate). Thus, the performance of a tree is dependent on the quality of the worst link.

In this paper, we present a practical MD video multicast scheme using multiple trees. As in CoopNet, video is coded into multiple descriptions, each sent on a different tree. Due to highly diverse wireless links, the end-to-end bandwidths from the sender to the receivers (called *path bandwidth* throughout this paper) are also highly diverse. It would be beneficial to code each description into a number of layers, such that a receiver with a high path bandwidth can receive more layers and achieve an improved video quality, while a receiver with a low path bandwidth can at least receive the base layer for an acceptable video quality. Many sophisticated schemes for multiple description coding have been investigated over the years, e.g., [3,13,15,17,22,23]. For an excellent survey, see [10]. A particularly efficient and practical scheme is based on the time-domain partitioning coding, where multiple descriptions are generated by separating the video frames and coding them separately. A double-description coding scheme using this technique is illustrated in Figure 1, in which the arrows indicate coding dependency of the frames. This simple time-domain partitioning method has been widely used (e.g., in [3, 5, 7, 12, 13, 22]) and will be also employed in this paper.

We take an application-centric, cross-layer approach to formulate MD video multicast routing as a combinatorial optimization problem. In contrast to previous work [18, 20, 21, 25], our objective is to optimize the application layer performance metric, i.e., video distortion. The problem formulated for the cross-layer multicast routing is highly complex and is expected to be NP-complete. Therefore, efficient heuristic algorithms would be most useful in practice. We find that *Genetic Algorithms* (GA) [4] are eminently suitable for addressing problems of this type, that have complex objective functions and large, unstructured combinatorial solution spaces. We construct a GA-based solution procedure, and demonstrate its efficacy through extensive performance studies.

The remainder of this paper is organized as follows. In Section 2, we formulate the problem of finding a pair of optimal multicast trees. In Section 3, we present a GA-based metaheuristic solution procedure. Sections 4 and 5 present performance studies. Related work is discussed in Section 6, and Section 7 concludes the paper.

2 Problem Formulation

Generally, a wireless ad hoc network having N nodes and L links can be modeled as a time-varying directed graph $\mathcal{G}(\mathcal{N}, \mathcal{L})$, where $\mathcal{N}$ is the set of vertices, representing mobile nodes, and $\mathcal{L}$ is the set of edges, representing wireless links. Accurate and computationally efficient characterization of an end-to-end path (or even a single-hop wireless link) in a wireless ad hoc network, which takes into account mobility, interference, and propagation, still remains an open problem. As an initial step, we focus on the network layer characteristics in this paper, assuming that the physical and MAC layer dynamics of wireless links are translated into network layer parameters. For example, we could characterize a link $\{i, j\} \in \mathcal{L}$ by:

- c_{ij}: available capacity of link $\{i, j\}$.
- p_{ij}: probability with which link $\{i, j\}$ fails.
- τ_{ij}: fixed, or the minimum delay of link $\{i, j\}$.
- t_{ij}: mean delay of link $\{i, j\}$.
- δ_{ij}^2: jitter of link $\{i, j\}$.

In practice, these parameters can be measured at every node and distributed throughout the network using Link State Advertisements (LSAs) [9].

2.1 Rate-Distortion Model for MD Video

For video coding and communications, a rate distortion model describes the relationship between the bit rate and the achieved distortion. We consider a sender generating two descriptions, with each description being encoded into two layers. For a general receiver, let d_h be the achieved distortion when only Description h is received, $h = 1, 2$, and d_0 the distortion when both descriptions are received. Clearly, d_0 and d_h, $h = 1, 2$ are functions of the rates of the descriptions. Let R_{base}^h be the base layer rate and R_{tot}^h the total rate (i.e., the aggregate rate of the base and enhancement layer) of Description h, $h = 1, 2$. Moreover, let P_{00}^b (P_{00}^e) denote the probability of receiving both base layers (enhancement layers), P_{01}^b (P_{01}^e) the probability of receiving Description 1's base layer (enhancement layer) only, P_{10}^b (P_{10}^e) the probability of receiving Description 2's base layer (enhancement layer) only, and P_{11}^b (P_{11}^e) the probability of receiving none of these base layers (enhancement layers). The average distortion of the receiver can be computed as follows:

$$D_r = P^b_{00}\{P^e_{00}d_0(R^1_{tot}, R^2_{tot}) + P^e_{01}d_0(R^1_{tot}, R^2_{base}) + P^e_{10}d_0(R^1_{base}, R^2_{tot}) + P^e_{11}d_0(R^1_{base}, R^2_{base})\} + P^b_{01}\{[P^e_{00} + P^e_{01}]d_1(R^1_{tot}) + [P^e_{10} + P^e_{11}]d_0(R^1_{base})\} + P^b_{10}\{[P^e_{00} + P^e_{10}]d_2(R^2_{tot}) + [P^e_{01} + P^e_{11}]d_2(R^2_{base})\} + P^b_{11} \cdot \sigma^2. \quad (1)$$

The rate-distortion region for an *i.i.d.* Gaussian source with the square error distortion measure was first introduced in [16]. For computational efficiency, in [1], Alasti *et al.* employed the following rate-distortion region, which is also used in the present paper.

$$\begin{cases} d_0(R_1, R_2) = \frac{2^{-2(R_1+R_2)}}{2^{-2R_1}+2^{-2R_2}-2^{-2(R_1+R_2)}} \cdot \sigma^2 \\ d_1(R_1) = 2^{-2R_1} \cdot \sigma^2 \\ d_2(R_2) = 2^{-2R_2} \cdot \sigma^2, \end{cases} \quad (2)$$

where σ^2 is the variance of the source. The region defined in (2) bounds the MD regions for any continuous-valued memoryless source with squared error distortion [10]. Note that other empirical rate-distortion models, e.g., the training-based model in [2], can be incorporated into the formulation as well.

2.2 Computing End-to-End Statistics

Consider a multicast session with sender s and a set of receivers $r \in \mathcal{M}$. The session uses two trees $\{\mathcal{T}_1, \mathcal{T}_2\}$, each rooted at sender s. Before formulating the optimal multicast routing problem, we need to compute the average distortion D_r of a receiver $r \in \mathcal{M}$ as a function of link statistics for a *given* pair of trees. Note that we do not mandate disjoint trees, which will unnecessarily shrink the solution space for optimization.

End-to-End Delay: For a tagged receiver $r \in \mathcal{M}$, let the path from the source s to r in tree $\mathcal{T}_h$ be $\mathcal{P}^h_r$, $h = 1, 2$. Although we do not assume any particular probability distribution for the link delays, we do follow the approach in [8] for the end-to-end delay. For member r, the end-to-end delay on its path P^h_r, denoted as t^h_r, could be modeled as a "shifted" Gamma distribution [8]:

$$y(t^h_r) = \frac{\alpha^h_r}{\Gamma(n^h_r)}[\alpha^h_r \cdot (t^h_r - \tau^h_r)]^{n^h_r - 1} e^{-\alpha^h_r \cdot (t^h_r - \tau^h_r)}, \quad (3)$$

for $t^h_r \geq \tau^h_r$, $h = 1, 2$. The end-to-end delay from sender s to receiver r can be interpreted as the total delay of going through n^h_r nodes, each with a processing delay of τ^h_r / n^h_r and an exponentially distributed queueing delay (with mean α^h_r). The parameters of the shifted Gamma distribution can be estimated from the statistics of the links along the path:

$$\begin{cases} \tau^h_r = \sum_{\{i,j\} \in \mathcal{P}^h_r} \tau_{ij}, \ h = 1, 2, \forall r \in \mathcal{M} \\ \alpha^h_r = \frac{\sum_{\{i,j\} \in \mathcal{P}^h_r} t_{ij} - \sum_{\{i,j\} \in \mathcal{P}^h_r} \tau_{ij}}{\sum_{\{i,j\} \in \mathcal{P}^h_r} \delta^2_{ij}}, \ h = 1, 2, \forall r \in \mathcal{M} \\ n^h_r = \frac{(\sum_{\{i,j\} \in \mathcal{P}^h_r} t_{ij} - \sum_{\{i,j\} \in \mathcal{P}^h_r} \tau_{ij})^2}{\sum_{\{i,j\} \in \mathcal{P}^h_r} \delta^2_{ij}}, \ h = 1, 2, \forall r \in \mathcal{M}. \end{cases}$$

Success Probabilities: As indicated in (1) and (2), the video distortion is the highest when both descriptions are lost, since σ^2 is generally much larger than d_0 and d_h, $h = 1, 2$. In order to reduce the possibility of simultaneously losing both the descriptions, the correlation of the loss processes of the two descriptions should be minimized at best [2]. It has been shown in previous work, e.g., [19], that packet interleaving can effectively reduce such a correlation and achieve a significantly improved video quality at the cost of an additional fixed interleaving delay. Consequently, we assume that video packets are *interleaved* with an appropriate interval (i.e., larger than the time-scale of link dynamics) before transmission, such that packet losses within the same frame are relatively independent[1].

Then, the probability of receiving a video packet before its decoding deadline Δ^h_r by receiver r from tree h is:

$$q^h_r = \left[\prod_{\{i,j\} \in \mathcal{P}^h_r} (1 - p_{ij})\right] \cdot \Pr(t^h_r \leq \Delta^h_r), h = 1, 2.$$

The joint probabilities of receiving the layers can be computed as[2]:

$$\begin{cases} P^b_{00} = P^e_{00} = q^1_r \cdot q^2_r \\ P^b_{01} = P^e_{01} = q^1_r \cdot (1 - q^2_r) \\ P^b_{10} = P^e_{10} = (1 - q^1_r) \cdot q^2_r \\ P^b_{11} = P^e_{11} = (1 - q^1_r) \cdot (1 - q^2_r). \end{cases} \quad (4)$$

Optimal Video Rates: Consider a receiver r and its two associated root paths $\{\mathcal{P}^1_r, \mathcal{P}^2_r\}$. We can classify the links in the two paths as the set of joint links, denoted as $\mathcal{J}(\mathcal{P}^1_r, \mathcal{P}^2_r)$, and the sets of disjoint links, denoted respectively as $\bar{\mathcal{J}}(\mathcal{P}^h_r)$, $h = 1, 2$. The minimum bandwidth of $\mathcal{J}(\mathcal{P}^1_r, \mathcal{P}^2_r)$ is defined to be:

$$B^{jnt}_r = \begin{cases} B(\mathcal{J}(\mathcal{P}^1_r, \mathcal{P}^2_r)), \text{if } \mathcal{J}(\mathcal{P}^1_r, \mathcal{P}^2_r) \neq \emptyset \\ \infty, \text{ otherwise,} \end{cases}$$

where $B(\mathcal{P}) = \min_{\{i,j\} \in \mathcal{P}}\{c_{ij}\}$. Then the path bandwidths of receiver r are:

$$\begin{cases} B^h_r = B(\mathcal{P}^h_r), \text{if } \sum_{h=1}^2 B(\mathcal{P}^h_r) \leq B^{jnt}_r, h = 1, 2 \\ B^1_r + B^2_r \leq B^{jnt}_r, \text{ otherwise.} \end{cases} \quad (5)$$

[1] It has been shown in [5] that the loss correlation of two descriptions is quite low once the two paths split after the first set of shared links.

[2] If the path bandwidths of a receiver allow receiving an enhancement layer from Tree 1, but not from Tree 2, then $P^e_{00} = 0$ and $P^e_{10} = 0$; if the receiver can receive an enhancement layer from Tree 2, but not from Tree 1, then $P^e_{00} = 0$, $P^e_{01} = 0$; if the receiver can only receive the base layers, then $P^e_{00} = 0$, $P^e_{01} = 0$, $P^e_{10} = 0$, and $P^e_{11} = 1$.

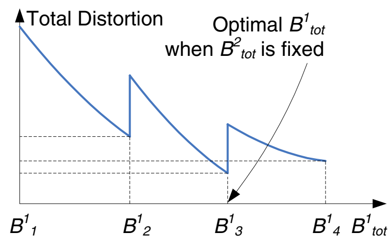

Figure 2. Optimal B^1_{tot} for a fixed B^2_{tot}.

The first line of (5) is for the case when the joint links are not the bottleneck of the paths, while the second line of (5) is for the case where one of the joint links is the bottleneck of both paths. In the latter case, we split the bandwidth of the shared bottleneck link in proportion to the mean success probabilities of the two root paths.

Once the path bandwidths are found, we need to determine the optimal bandwidths of the layers for each description. Clearly, all of the receivers should be able to receive the base layers in order to effectively decode the descriptions. Thus, we set the base layer bandwidth of description h to:

$$B^h_{base} = \min_{r \in \mathcal{M}} \{B^h_r\}, h = 1, 2. \quad (6)$$

The total bandwidth of description h, B^h_{tot}, should be within the range $[B^h_{base}, \max_{r \in \mathcal{M}}\{B^h_r\}]$. For a chosen B^h_{tot}, receivers that satisfy $B^h_r \geq B^h_{tot}$ can receive both layers of description h; other receivers with $B^h_r < B^h_{tot}$ can only receive the base layer of description h.

It can be shown that the average distortion of a receiver D_r is a non-increasing function of the rate B^h_{tot}, $h = 1, 2$ [12]. Therefore, for a fixed B^h_{tot}, the total distortion of all receivers is a piece-wise non-increasing function of B^{3-h}_{tot} with discontinuous jumps at B^{3-h}_r, $r \in \mathcal{M}$. An example with four receivers is illustrated in Figure 2, where the total distortion is plotted as a function of B^1_{tot} for a fixed B^2_{tot}, assuming that $B^h_1 < B^h_2 < B^h_3 < B^h_4$, $h = 1, 2$. In this example, B^1_{base} is set to B^1_1, as given in (6). The total distortion is the highest when $B^1_{tot} = B^1_1$, since there is no enhancement layer for Description 1. If we fix B^2_{tot} and increase B^1_{tot}, the total distortion keeps on decreasing, due to the monotonicity properties of (1) [12]. When B^1_{tot} reaches B^1_2, there is a sudden increase in the total distortion, since Receiver 2 cannot receive the enhancement layer anymore, and so forth. We find that for a fixed B^2_{tot}, we only need to evaluate the total distortion at three points, i.e., $B^1_{tot} = B^1_r$, $r = 2, 3, 4$, in order to find the optimal B^1_{tot}.

Figure 3 plots the total distortion for all feasible combinations of B^1_{tot} and B^2_{tot}. We find that the same monotonicity property holds in this case. Due to the monotonicity properties of (1), we only need to examine the total distortion at points $\{B^1_i, B^2_j\}$, $i, j \in \{2, 3, 4\}$, in order to find the optimal total rates that minimize the total distortion. In general, if there are K_h different path bandwidths in tree $\mathcal{T}_h$, $h = 1, 2$, we only need to evaluate the total distortion at $(K_1 - 1) \cdot (K_2 - 2)$ bandwidth combinations in order to find the optimal bandwidth for both descriptions. Note that the associated computational burden is low, since many wireless links operate at a small number of fixed bandwidths (e.g., $K_1 = K_2 = 4$ for a wireless LAN link).

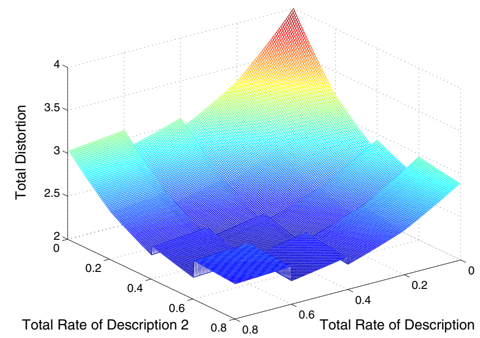

Figure 3. Total distortion for all combinations of B^1_{tot} and B^2_{tot}.

After B^h_{tot} and B^h_{base}, $h = 1, 2$, are computed, the rates of the descriptions, in bits per pixel, can be determined as:

$$\begin{cases} R^h_{tot} = \rho \cdot B^h_{tot}, \ h = 1, 2 \\ R^h_{base} = \rho \cdot B^h_{base}, \ h = 1, 2, \end{cases} \quad (7)$$

where $\rho = \gamma \cdot W \cdot H \cdot R_f$ for a video with frame rate R_f and frame size $W \times H$; γ is a constant determined by the chroma subsampling format (e.g., $\gamma = 1.5$ for the quarter common intermediate format (QCIF)).

2.3 The MD Video Multicast Routing Problem

Before proceeding to the problem formulation, we first define the following two sets of variables for describing the choice of trees. For every link $\{i, j\} \in \mathcal{L}$, define

$$x^h_{ij} \overset{def}{=} \begin{cases} 1, & \text{if } \{i, j\} \in \mathcal{T}_h, h = 1, 2 \\ 0, & \text{otherwise}, h = 1, 2. \end{cases} \quad (8)$$

For every node $i \in \mathcal{N}$, define

$$u^h_i \overset{def}{=} \begin{cases} \text{number of hops from } s \text{ to } i, \ i \in \mathcal{T}_h, h = 1, 2 \\ 0, \ i \notin \mathcal{T}_h, h = 1, 2. \end{cases} \quad (9)$$

Then, we formulate the optimal MD video multicast routing problem (OPT-MM) as follows.

OPT-MM Given a wireless ad hoc network $\mathcal{G}\{\mathcal{N}, \mathcal{L}\}$ and a multicast session $\{s, \mathcal{M}\}$, find a pair of trees $\{\mathcal{T}_1, \mathcal{T}_2\}$, such that the total video distortion of all of the receivers in $\mathcal{M}$ is minimized. That is:

$$\textbf{Minimize:} \quad D = \textstyle\sum_{r \in \mathcal{M}} D_r \quad (10)$$

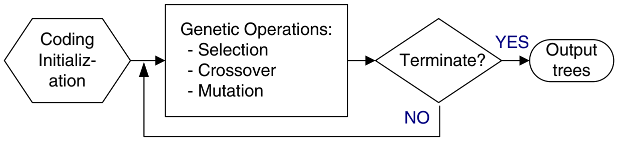

Figure 4. The GA-based multicast routing.

subject to:

$$x_{ij}^h \le \sum_{k:\{k,i\}\in\mathcal{L}} x_{ki}^h,$$
$$\forall i \ne s, \forall j \notin \{i,s\} : \{i,j\} \in \mathcal{L}, h = 1,2 \quad (11)$$
$$\sum_{j:\{j,i\}\in\mathcal{L}} x_{ji}^h = 1,\ i \in \mathcal{M}, h = 1,2 \quad (12)$$
$$\sum_{j:\{j,i\}\in\mathcal{L}} x_{ji}^h \le 1,\ \forall i \in \mathcal{N}\backslash\mathcal{M}, h = 1,2 \quad (13)$$
$$u_i^h - u_j^h + N \cdot x_{ij}^h \le N - 1,\ \forall i,j \in \mathcal{N}, h = 1,2 \quad (14)$$
$$x_{ij}^1 \cdot R_{tot}^1 + x_{ij}^2 \cdot R_{tot}^2 = (1-\epsilon)\cdot\rho\cdot c_{ij},$$
$$\text{for some } \epsilon \in [0,1], \forall\{i,j\} \in \mathcal{L} \quad (15)$$
$$x_{ij}^h \in \{0,1\},\ \forall\{i,j\} \in \mathcal{L}, h = 1,2 \quad (16)$$
$$u_i^h \in \{0,1,\cdots,N-1\},\ \forall i \ne s, h = 1,2. \quad (17)$$

In Problem OPT-MM, constraints (11)–(14) represent the choice of a pair of trees. More specifically, constraint (11) regulates the input-output relation of an arbitrary node, i.e., a node can forward a video stream only if it receives a video stream from its parent node; constraint (12) guarantees that all member nodes are connected in the tree (with a single parent node); constraint (13) and (14) ensures the loop-free property of trees. Constraint (15) guarantees that the links are stable.

In its simplest form, i.e., when there is only one receiver in the group, Problem OPT-MM reduces to a QoS routing problem with two additive (delay and jitter), one multiplicative (loss), and one concave (bandwidth) metrics, which have been shown to be NP-complete [24]. Therefore, we expect that Problem OPT-MM is also NP-complete[3]. It would be futile to pursue exact solutions.

3 GA-Based Solution

We suggest that an attractive strategy to address Problem OPT-MM is to view it as a "black-box" optimization problem and to explore an effective *metaheuristics* approach [6]. In particular, we find *Genetic Algorithms* (GA) [4] are eminently suitable for addressing this type of complex problems. The basic framework for our GA-based multicast routing solution procedure is illustrated in Figure 4. We discuss each component in the sequel.

[3]However, we leave the rigorous proof for this claim for future research.

Coding and Initialization: In GAs, it is important to properly represent a solution, which will facilitate the genetic operations. In our approach, a solution for Problem OPT-MM is a pair of trees. In our implementation, we use an adjacency matrix A_h to describe the connectivity in tree h, $h = 1,2$. That is, if $a_{ij}^h = 1$, link $\{i,j\}$ is in tree $\mathcal{T}_h$, $h = 1,2$; if $a_{ij}^h = 0$, link $\{i,j\}$ is not in tree $\mathcal{T}_h$, $h = 1,2$. Thus, we characterize a solution for our problem as a pair of such adjacency matrices.

With this encoding of solutions, we next generate an initial population. In order to make the individuals evenly distributed across the entire search space, we take a random construction approach. Starting with sender s, we randomly pick links emanating from s and include them (along with the "to-nodes" at which these links are incident) into the partial tree. Note that we only choose new links at each step for which exactly one end node is in the current partial tree in order to avoid loops, until all member nodes are included in the tree. After creating a number of trees in this manner, an individual can be created by randomly pairing the trees. An unbiased initial population is thus generated.

Genetic Operations: Genetic operations operate on the individuals according to their fitness. The fitness of an individual $f(\bar{x}), \bar{x} \equiv \{\mathcal{T}_1, \mathcal{T}_2\}$, is closely related to its objective value. Since the objective is to minimize the total distortion, we define fitness as the inverse of the distortion value: $f(\bar{x}) = 1/D(\bar{x})$. This simple definition appears to work very well computationally, although we intend to explore other fitness definitions in our future effort.

By the *selection* operation, we select the individuals that have more potential to produce better offspring in terms of the fitness value. In our implementation, we use the *Tournament* selection scheme [4]. That is, each time, we randomly pick $k = 2$ individuals in the population and choose the one having a higher fitness value. Repeating this procedure multiple times, we get a set of individuals that have better fitness values (and thus better genes or building blocks). Next, we perform the crossover and mutation operations on these selected individuals.

Crossover mimics the genetic mechanism of reproduction in the natural world, where genes from parents are recombined and passed to offsprings. For a pair of parent individuals, we first randomly choose a tree from each of them. Then, we find a common link in these two selected trees and exchange the corresponding subtrees connected by this link. After swapping the subtrees, we also check the two graphs obtained and make sure that they are feasible, i.e., they are loop free and include all the member nodes. If no such common link exists, we simply swap the two selected trees directly between the two parents. For two parents, crossover is performed with a probability θ, called the crossover rate.

Mutation is the key ingredient of genetic algorithms. It is used to diversify the population's gene pool, and thus keep the computation from being trapped in a local optimum. By randomly changing (mutating) one or more genes in an individual, the mutation produces a new individual with a random "jump" into a new area in the solution space. This operation therefore enables a wider exploration. In our algorithm, we randomly choose a link in a multicast tree (e.g., a link having a low bandwidth or a high failure probability). Removing this link results in two subtrees. Then, we add back a link or a branch that will connect the two subtrees with no loops. For an individual, the probability of being mutated is called the mutation rate μ.

Termination and Output Trees: As discussed, GA evolves a population of solutions toward the optimum. Generally, the more generations, the closer the GA solutions are to the optimum. The termination condition in Figure 4 could be based on the total number of iterations, the maximum computational time, a threshold of desired video distortion, or a combination of these conditions.

Upon termination, the best individual (i.e., the one having the highest fitness value f^*) in the final population is taken as the solution to Problem OPT-MM. Note that in the final population, there may be more than one solution having this best-found fitness value f^*. It is also highly likely that there are other solutions having fitness values close to f^*. These solutions can be kept as back-ups for the multicast session, and can be used when the quality of the selected pair of trees deteriorates, thus reducing need for executing the routing process for every tree interruption [20].

4 Comparison with Layered Coding-Based Approaches

Multicasting layered video has been well studied for the Internet. For comparison, we also formulated the problem of multicast routing for layered video using a pair of trees. For a video with two layers, we investigated the following two approaches:

- LC-I: send the base layer on the tree having a higher minimum path bandwidth and send the enhancement layer on the other tree.
- LC-II: send the base layer on the tree having a higher success delivery ratio and send the enhancement layer on the other tree.

If the base layer is transmitted on $\mathcal{T}_1$ and the enhancement layer on $\mathcal{T}_2$, the rates of the base layer R_b and the enhancement layer R_e can be computed as:

$$\begin{cases} R_b = \min_{r \in \mathcal{M}} \{\rho \cdot B_r^1\} \\ R_e = \min_{r \in \mathcal{M}} \{\rho \cdot B_r^2\}. \end{cases} \quad (18)$$

The average distortion of the video received by r is [19]:

$$D_r^{lc} = \Big[q_r^1 \cdot q_r^2 \cdot 2^{-2(R_b+R_e)} + q_r^1 \cdot (1-q_r^2) \cdot 2^{-2R_b} + (1-q_r^1) \Big] \cdot \sigma^2. \quad (19)$$

The first term on the right-hand-side (RHS) of (19) represents the contribution to D_r^{lc} when both layers are received; the second term corresponds to the case when only the base layer is received, and the third term corresponds to the case when the base layer is lost.

We performed extensive simulations to compare the performance of the proposed scheme. For each experiment, we generated a wireless ad hoc network by placing a number of nodes at random locations in a square region. Connectivity was determined by the distance coverage of each node's transmitter. The source node s and the receivers $r \in \mathcal{M}$ are randomly chosen from the nodes. For every link, the failure probability was randomly chosen from [0, 1] with a truncated exponential distribution (the mean is 0.01); the available link bandwidth was randomly chosen from [100 Kb/s, 800 Kb/s], evenly spaced at 100 Kb/s intervals. The fixed delay τ_{ij} and mean delay t_{ij} of a link is set to 5 ms and 30 ms, respectively; the jitter δ_{ij} is randomly chosen from [7 ms, 17 ms], $\forall \{i,j\} \in \mathcal{L}$. The GA-based routing scheme was implemented using MATLAB 6.5[4]. We set σ^2 to 1, since it only affects the absolute value of distortion, but does not affect path selection decisions. We found that the optimal GA parameter values are quite robust for different network configurations and multicast sessions. In the results presented here and for the subsequent sections, we set the population size to 7, the crossover rate to 0.7, and the mutation rate to 0.3.

In Figure 5, we plot the achieved minimum average distortion among all receivers for various decoding deadlines, obtained for a 10-receiver group in a 30-node network. Distortion deadline is an essential characteristic of real-time multimedia applications that distinguishes them from elastic data applications. Figure 5 plots the average distortion curves obtained by four schemes, namely, LC-I, LC-II, MD video multicast using a single tree (i.e., both descriptions are sent on the same tree), and the proposed MD video multicast scheme using two trees. Each distortion value in the figure is the average of 10 runs.

It can be observed that all the four curves exhibit a similar behavior when the decoding deadline Δ increases. For small Δ, the underlying network cannot satisfy such tight delay requirements, leading to high distortions for all the four schemes. On the other hand, very large decoding deadlines imply that all the packets that are correctly received are useful in improving the video quality. In this case, the achieved distortion is mainly determined by the available bandwidths and loss characteristics of the paths. Between

[4]Note that faster computing can be achieved by porting the code to C.

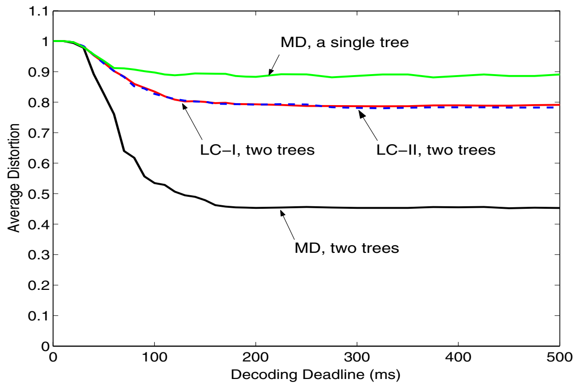

Figure 5. Effect of the decoding deadline: comparison of four video multicast schemes.

these two extremes, the average distortion decreases with Δ. With such curves, we can trade-off delay for better video performance.

In Figure 5, the proposed scheme outperforms all the other three schemes. Sending both descriptions on the same tree has the poorest performance. This is because sharing the bottlenecks greatly reduces the rates of the descriptions (see (5)). In addition, for completely overlapping trees, the correlation between the losses of two descriptions becomes much stronger, making a large interleaving delay necessary. In LC-I and LC-II, sending the layers on different trees effectively increases the video rate and thus improves the video quality. However, the performance of these schemes is limited by the decoding dependency between the layers, i.e., the base layer is needed for an effective decoding of the received video. For good performance, the base layer tree should be reliable for most of the session period. In wireless ad hoc networks, however, such reliable trees are hard to find. Although feedback and retransmissions can significantly improve the performance of layered videos [13], these should be avoided as far as possible in group communications. Another reason for the poor performance is that these schemes do not consider the diverse path bandwidths (see (18)). As a result, receivers having high path bandwidths are not allowed to improve their video quality and the overall performance is determined by the receiver(s) having the minimum path bandwidth in each tree.

5 Comparison with Network-Centric Approaches

In this section, we compare the performance of the GA-based MD-video multicast scheme with two representative network-centric approaches based on Dijkstra's shortest path algorithm. The Bounded Shortest Multicast Algorithm (BSMA) algorithm in [18] is designed to construct minimum-cost multicast trees with delay constraints, and is shown in [21] to achieve the lowest cost among several existing algorithms. We extend BSMA to compute two trees, by running the algorithm twice, using a link cost metric $-\log(1-p_{ij})$ for the first run and a link cost metric $1/c_{ij}$ for the second run. The same decoding deadline Δ is used as a delay bound in BSMA. In another work [20], Sajama and Haas presented the Independent-Tree Ad Hoc Multicast Routing (ITAMAR) procedure, which is a framework for efficient multicast routing in ad hoc networks. ITAMAR continuously maintains a set of multicast trees: one or more for the session to use and the rest as backups. Among the several algorithms proposed in [20], we implemented the Shortest Path Heuristic (SPTH) algorithm for comparison, using the link cost metric $-\log(1-p_{ij})$. After a pair of trees are computed by these two algorithms, we apply the techniques in Section 2.2 to find the optimal rates for the descriptions and compute the achieved distortion using (1).

The achieved average distortions by the schemes are listed in Table 1, each being the average of 10 runs. We find that the GA-based approach significantly outperforms the two network centric approaches in all of the cases studied. This is mainly due to the fact that both BSMA and ITAMAR-SPTH only optimize the network layer performance metrics, which does not necessarily achieve the optimal application layer performance. For completeness, we also present the results for LC-I and LC-II. The GA-based approach again outperforms these layered coding based schemes by a significant amount.

Among the four schemes listed in Table 1, ITAMAR-SPTH has a performance closest to the GA-based routing in terms of distortion values. Therefore, we ran ITAMAR-SPTH and the GA-based routing for a five-member group in a 15-node network, and compared the PSNRs of the reconstructed video frames, in order to demonstrate the efficiency of the proposed scheme. We used an H.263+ codec (originally from the University of British Columbia (UBC)) and the 400-frame "Foreman" trace in the QCIF format. The video sequence was encoded with a frame rate of 30 fps and an intra MB refresh rate of $1/10$. When necessary, the SNR scalable coding was used to code each description into two layers. We implemented the off-line rate control for the enhancement layer, which was missing from the original UBC distribution. Each group of blocks (GOB) was transmitted in a packet to make them independently decodable.

The qualities of the trees found by both schemes are presented in Table 2. Generally, GA is comparable to ITAMAR-SPTH in terms of the success delivery ratio. This is due to the fact that ITAMAR-SPTH uses link loss rates as the routing metric when determining the trees. However, ITAMAR-SPTH does not consider bandwidths and delays in the algorithmic design, making the delay and the bandwidth of the resulting trees unpredictable. For example, a receiver may have an extremely high delay such that al-

Table 1. Average Distortion Achieved by GA and Existing Approaches

	15-node (Δ=100 ms)		30-node Δ=150 ms)		50-node (Δ=250 ms)	
	Dense (12)	Sparse (3)	Dense (25)	Sparse (5)	Dense (40)	Sparse (10)
BSMA [18]	0.6640	0.7039	0.8921	0.7798	0.7077	0.7884
ITAMAR-SPTH [20]	0.6545	0.6372	0.7205	0.6087	0.7023	0.8242
LC-I	0.7502	0.6228	0.8191	0.7384	0.8011	0.8108
LC-II	0.7253	0.6012	0.8015	0.7103	0.7892	0.7780
GA-based Routing	0.4121	0.3600	0.5152	0.4846	0.5333	0.5280

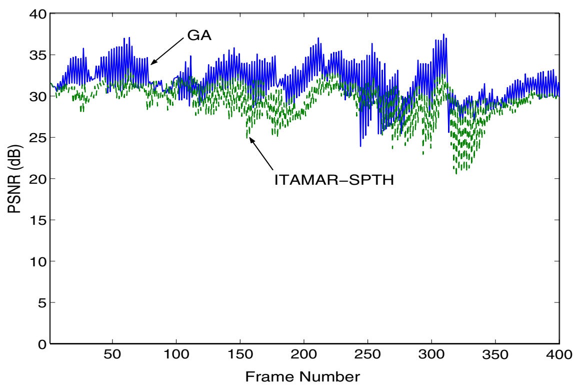

Figure 6. PSNRs of received frames by Receiver 1.

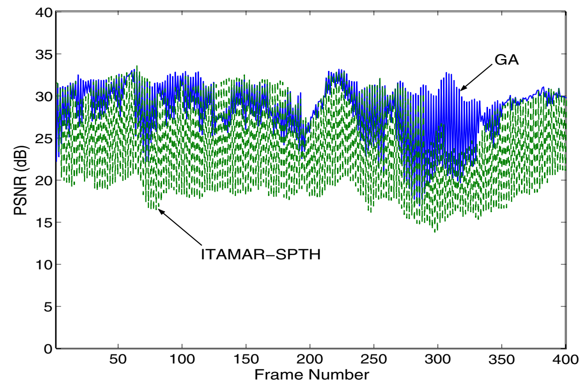

Figure 7. PSNRs of received frames by Receiver 2.

most all of the video packets are overdue (e.g., Receiver 3, Description 2). On the other hand, the GA-based routing optimizes video distortion directly, which is a compound function of the link statistics. Consequently, the GA-based approach achieves much higher video rates than does ITAMAR-SPTH. Such higher video rates greatly reduce the distortion caused by the lossy video coder, and on average, achieved a 3.72 dB improvement in PSNR over the ITAMAR-SPTH algorithm in this experiment.

We also plot the PSNR curves for three representative receivers in Figures 6, 7, and 8. The GA PSNR curves are well above the ITAMAR-SPTH curves for most of the frames. It can be seen that the GA-based routing attempts to achieve a balanced quality for the two descriptions, yielding a better subjective video quality. This is due to the symmetry of the description rates and loss probabilities in the objective function (see (1)). Minimizing such an objective function will drive GA to find balanced trees. It is possible to further reduce the quality difference between the two descriptions by using advanced MD coders. In order to illustrate the decoded video quality, we present the decoded Frame 226 obtained by receivers 1, 2, and 3 in Figure 9. For all the receivers, the pictures obtained by the GA-based routing are much clearer than those obtained by ITAMAR-SPTH. Specifically, the pictures obtained by ITAMAR-SPTH for receivers 2 and 3 are barely recognizable.

One advantage of the network-centric algorithms, such as BSMA and ITAMAR, is that they have lower compu-

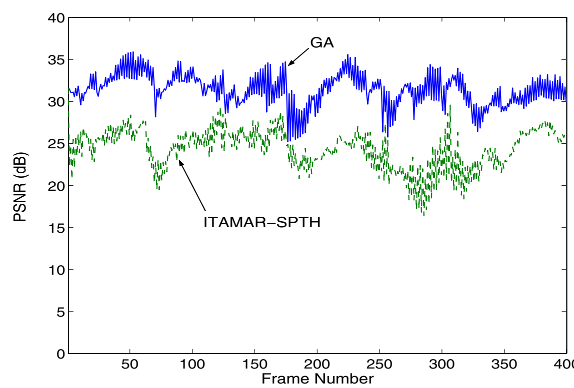

Figure 8. PSNRs of received frames by Receiver 3.

tational complexity than GA-based approaches. However, the efficiency of the GA algorithm can be improved since it is well suited for parallel computation. In addition, our numerical results show that with GA, the greatest improvement in fitness value is achieved after a few initial number of generations, and subsequent improvements are much smaller after these early generations. Therefore, for a delay-sensitive real-time application, GA can compute a set of "good" trees for the application to use after a very small delay. As GA continues to evolve, the trees can be dynamically updated with newly computed (better) trees for enhanced performance.

Table 2. GA-based Routing versus ITAMAR-SPTH

	ITAMAR-SPTH (Δ=100 ms)					GA (Δ=100 ms)				
Member	1	2	3	4	5	1	2	3	4	5
Desc. 1 success delivery ratio	95.6%	47.1%	86.2%	84.2%	99.5%	98.8%	56.6%	98.3%	98.9%	99.6%
Desc. 2 success delivery ratio	99.3%	98.5%	0.2%	98.6%	99.0%	98.1%	99.6%	99.3%	97.9%	98.8%
Desc. 1 BL bandwidth (Kb/s)	100	100	100	100	100	200	200	200	200	200
Desc. 1 EL bandwidth (Kb/s)	0	0	0	0	300	0	300	0	300	300
Desc. 2 BL bandwidth (Kb/s)	100	100	100	100	100	100	100	100	100	100
Desc. 2 EL bandwidth (Kb/s)	0	0	0	0	0	300	0	300	300	300
Average PSNR (dB)	29.53	24.16	24.38	27.55	30.95	31.64	28.63	31.43	31.75	31.70


(a) Receiver 1, GA-based routing.


(b) Receiver 2, GA-based routing.


(c) Receiver 3, GA-based routing.


(d) Receiver 1, ITAMAR-SPTH.


(e) Receiver 2, ITAMAR-SPTH.


(f) Receiver 3, ITAMAR-SPTH.

Figure 9. Reconstructed Frame 226 at the receivers.

6 Related Work

One relevant work, [17], has been discussed in detail in Section 1. In this section, we discuss other related work that provide the background for this investigation.

MD video streaming has been an active research area due to MD video's unique error resilience and open-loop operation capabilities [2, 3, 5, 7, 12, 13]. An empirical MD rate-distortion model has been presented in [2] for computing average video distortions from loss probabilities of path links. The scheme in [5] also shows how to compute the average video distortion from link statistics in the context of overlay networks for unicast MD video streaming. These models can be easily incorporated into the framework presented in this paper. Some other works focus on end-system based schemes for supporting MD video unicast streaming for a set of given paths [7, 8, 13]. The important problem of finding multiple paths is not addressed.

QoS multicast routing has been an active research area for many years. Most of these problems belong to the class of minimum or constrained minimum Steiner tree problems, which are well-known to be NP-complete. Various efficient heuristic algorithms have been proposed (e.g., [18,20]). See [21] for a comparison of the algorithms and see [14] for a survey of multicast routing protocols in wireless ad hoc networks. Most of the algorithms proposed aim to find a single tree using network layer performance metrics. As our numerical results in the previous section show, such network-centric approaches do not necessarily deliver good performance at the application layer.

In [20], Sajama and Haas propose ITAMAR, a class of efficient algorithms that construct a set of alternative trees having low costs. The best tree is then used until it fails, at which time it is replaced by another tree in the set, so that the time between a tree failure and rerouting is minimized. It has been shown in [20] that significant improvement in the mean time between interruptions can be achieved with a small increase in the tree cost and routing overhead. As discussed, our GA-based routing scheme has a similar advantage as ITAMAR in improving the mean time between tree interruptions. But as our simulation results show, ITAMAR is not suitable for our problem since it only optimizes a single metric at a time. While, generally, the performance of such algorithms can be improved by defining a compound routing metric, i.e., a function of capacity, loss, and delay, such a compound metric-approach does not apply to Problem OPT-MM. Since ITAMAR-SPTH is based on Dijkstra algorithm, it requires the compound metric to be additive. Problem OPT-MM has much more complex relationships (nonadditive) pertaining to the contribution of any link metric to the objective function, which excludes the use of any Dijkstra-based algorithm.

In [25], Zhang and Leung propose an orthogonal genetic algorithm for multimedia multicast routing, which is essentially a delay constrained Steiner tree problem. An interesting experimental design method, called orthogonal design, is incorporated into the crossover operation and is shown to greatly improve the convergence speed of the GA. In our early work [12], a GA-based multicast routing scheme for unicast MD video streaming is presented. In the present paper, we study a much more difficult problem of finding a pair of trees, while optimizing the application layer performance (i.e., MD video distortion). Our efforts provide an important methodology for addressing complex cross-layer optimization problems, particularly those involving application and network layers.

7 Conclusions

In this paper, we have proposed a practical MD video multicast scheme for wireless ad hoc networks, which is both error resilient and scalable. We have adopted an application-centric, cross-layer approach and have formulated the multicast routing task as a combinatorial optimization problem. The formulated problem is NP-hard and has a complex structure that precludes obtaining an exact solution with reasonable effort. On the other hand, as demonstrated by our work, Genetic Algorithms are highly suitable for such problems having an extremely complex objective function and a large, unstructured solution space. We have constructed a GA-based solution procedure for the optimized double-tree multicast routing problem. Extensive simulations illustrate significant gains in video quality achieved over existing approaches for a wide range of network operational conditions.

Acknowledgments

The authors would like to thank Prof. Yao Wang at Polytechnic University, Brooklyn, NY, and Dr. Shunan Lin at Harmonic Inc. for helpful discussions and for providing the MD codec. This research has been supported in part by the National Science Foundation under Grants ANI-0312655, CNS-0347390, and DMI-0094462, and by the Office of Naval Research under Grant No. N00014-03-1-0521.

References

[1] M. Alasti, K. Sayrafian-Pour, A. Ephremides, and N. Farvardin. Multiple description coding in networks with congestion problem. *IEEE Trans. Inform. Theory*, 47(3):891–902, Mar. 2001.

[2] J. Apostolopoulos, W. Tan, S. Wee, and G. Wornell. Modeling path diversity for multiple description video communication. In *Proc. IEEE ICASSP*, pages 13–17, May 2002.

[3] J. G. Apostolopoulos. Reliable video communication over lossy packet networks using multiple state encoding and path diversity. In *Proc. SPIE VCIP*, Jan. 2001.

[4] T. Back, D. Fogel, and Z. Michalewicz, editors. *Handbook of Evolutionary Computation*. Oxford University Press, New York, NY, 1997.

[5] A. C. Begen, Y. Altunbasak, and O. Ergun. Multi-path selection for multiple description encoded video streaming. In *Proc. IEEE ICC*, pages 1583–1589, May 2003.

[6] C. Blum and A. Roli. Metaheuristics in combinatorial optimization: overview and conceptual comparison. *ACM Computing Surveys*, 35(3):268–308, Sept. 2003.

[7] J. Chakareski, S. Han, and B. Girod. Layered coding vs. multiple descriptions for video streaming over multiple paths. In *Proc. of ACM Multimedia*, Nov. 2003.

[8] P. A. Chou and Z. Miao. Rate-distortion optimized streaming of packetized media. submitted paper, 2001.

[9] T. Clausen and P. Jacquet. Optimized link state routing protocol, Oct. 2003. IETF RFC 3626.

[10] V. Goyal. Multiple description coding: compression meets the network. *IEEE Trans. Circuits and Syst. for Video Techno.*, 18:74–93, Sept. 2001.

[11] D. B. Johnson, D. A. Maltz, and Y.-C. Hu. The dynamic source routing protocol for mobile ad hoc networks (dsr), Apr. 2003. IETF Internet Draft draft-ietf-manet-dsr-09.txt.

[12] S. Mao, Y. Hou, X. Cheng, H. D. Sherali, and S. F. Midkiff. Multi-path routing for multiple description video over wireless ad hoc networks. Technical report, the Bradley Dept. of ECE, Virginia Tech, April 2004.

[13] S. Mao, S. Lin, S. S. Panwar, Y. Wang, and E. Celebi. Video transport over ad hoc networks: Multistream coding with multipath transport. *IEEE J. Select. Areas Commun.*, 21(10):1721–1737, Dec. 2003.

[14] P. Mohapatra, C. Gui, and J. Li. Group communications in mobile ad hoc networks. *IEEE Computer*, 37(2):70–77, Feb. 2004.

[15] A. E. Mohr, E. A. Riskin, and R. E. ladner. Unequal loss protection: Graceful degradation of image quality over packet erasure channels through forward error correction. *IEEE J. Select. Areas Commun.*, 18(7):819–828, June 2000.

[16] L. Ozarow. On a source coding problem with two channels and three receivers. *Bell Syst. Tech. J.*, 59(10):84–91, Dec. 1980.

[17] V. N. Padmanabhan, H. J. Wang, P. A. Chou, and K. Sripanidkulchai. Distributing streaming media content using cooperative networking. In *Proc. ACM NOSSDAV*, pages 177–186, May 2002.

[18] M. Parsa, Q. Zhu, and J. J. Garcia-Luna-Aceves. An interative algorithm for delay-constrained minimum-cost multicasting. *IEEE Trans. on Networking*, 6(4):461–474, Aug. 1998.

[19] A. R. Reibman, H. Jafarkhani, M. T. Orchard, and Y. Wang. Performance of multiple description coders on a real channel. In *Proc. IEEE ICASSP*, pages 2415–2418, Mar. 1999.

[20] Sajama and Z. J. Haas. Independent–tree ad hoc multicast routing (itamar). *Mobile Networks and Applications*, 8(5):551–566, Oct. 2003.

[21] H. F. Salama, D. S. Reeves, and Y. Viniotis. Evaluation of multicast routing algorithms for real-time communication on high-speed networks. *IEEE J. Select. Areas Commun.*, 15(3):332–345, Apr. 1997.

[22] Y. Wang and S. Lin. Error-resilient video coding using multiple description motion compensation. *IEEE Trans. Circuits and Syst. for Video Techno.*, 12(6):438–452, June 2002.

[23] Y. Wang, M. T. Orchard, V. Vaishampayan, and A. R. Reibman. Multiple description coding using pairwise correlating transforms. *IEEE Trans. on Image Processing*, 10(3):351–366, Mar. 2001.

[24] Z. Wang and J. Crowcroft. Quality-of-service routing for supporting multimedia applications. *IEEE J. Select. Areas Commun.*, 17(8):1488–1505, Aug. 1999.

[25] Q. Zhang and Y. Leung. An orthogonal genetic algorithm for multimedia multicast routing. *IEEE Trans. on Evolutionary Computation*, 3(1):53–62, Apr. 1999.