Mobile Networks and Applications 11, 63–73, 2006

DOI: 10.1007/s11036-005-4461-5

Multiple Description Video Multicast in Wireless Ad Hoc Networks

SHIWEN MAO, XIAOLIN CHENG and Y. THOMAS HOU
The Bradley Department of Electrical and Computer Engineering, Virginia Tech, Blacksburg, VA 24061, USA

HANIF D. SHERALI
The Grado Department of Industrial and Systems Engineering, Virginia Tech, Blacksburg, VA 24061, USA

Published online: 9 December 2005

Abstract. Multiple description (MD) coding is a new video coding technique that is uniquely suitable for video transport over wireless ad hoc networks. In this paper, we consider the problem of how to support video multicast with MD video in ad hoc networks. We follow an application-centric, cross-layer routing approach with the objective of minimizing the overall video distortion. We propose an MD video multicast scheme where multiple source trees are used. Furthermore, each video description is coded into multiple layers in order to cope with diversity in wireless link bandwidths. Based on this multicast model, we formulate the cross-layer 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.

Keywords: cross-layer design, genetic algorithms, multiple description coding, routing, video multicast, wireless ad hoc networks

1. Introduction

As progress in wireless ad hoc network research continues, there is a compelling need to support multimedia applications (e.g., video) in such networks. Recently, a new video coding technique, called multiple description (MD) coding, has been found to be uniquely suitable to support video applications in dynamic ad hoc networks [15]. Under MD coding, a video session is encoded into multiple equally important streams, each can be transported on a particular path. At the receiver side, each received stream can be independently decoded into a low, but acceptable video quality and the overall video quality commensurates with the number of received descriptions [10]. This feature makes MD video an excellent match for video communication in ad hoc networks.

In this paper, we study the important and difficult problem of *multicasting* MD video in wireless ad hoc networks. Although multicast routing for ad hoc networks has been explored in prior research (see, e.g., [21,22], they are not quite suitable for MD video applications for several reasons. First, many existing algorithms use a single network-layer metric in routing. However, video quality is usually determined by more than one metric, such as loss, delay, jitter, and available bandwidth. Optimizing one metric could lead to significant degradation in the other [13]. As a result, single network-layer metric based approaches do not necessarily offer optimal video quality. Second, many existing protocols are based on the Dijkstra's algorithm, which requires additive routing metrics. For multimedia-centric routing, there exists a highly complex relationship pertaining to the contribution of any link to the video quality, which depends in general on the other links that are included in a fashion that has no particular structural property such as additivity and convexity. Consequently, there remain important open problems in multicast routing for MD video in wireless ad hoc networks.

We present an efficient MD video multicast scheme aiming to cope with the above issues. With this scheme, a number of multicast trees are used, with each multicast tree supporting one video description. Further, we propose each description be coded into a base layer and a number of enhancement layers. Packets belonging to the same description from both the base layer and enhancement layers are transmitted on the same tree. We show that this MD video multicast approach can effectively deal with frequent link failures and diverse link qualities in wireless ad hoc networks. First, since a receiver is a member of multiple trees, it is connected to the source node via multiple paths. When a path is broken due to a failed link, the other path(s) may still be in a good condition, assuming link failures in the network are independent. As a result, such *path diversity* provides enhanced error resilience to an MD video session [2,15]. Second, 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 is desirable to code each description into a number of layers, so that a receiver with a high path bandwidth can subscribe the maximum number of layers of the corresponding description that are allowed by its path bandwidth for best video quality. Figure 1 illustrates the proposed multicast scheme for MD video. Since the video session is coded into two descriptions, two multicast trees are needed. Note that each video description is further coded into a base layer and an enhancement layer. In the example, the path from the sender to Receiver 2 in Tree 1 has enough path bandwidth for both layers of Description 1. But in Tree 2, the

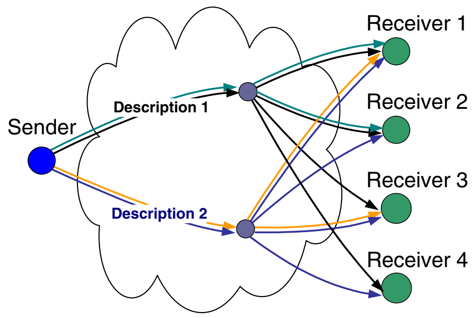

Figure 1. MD video multicast using two trees.

path bandwidth between the sender and Receiver 2 only has enough bandwidth for the base layer of Description 2.

Within the proposed multi-tree multi-layer scheme for MD multicast, the most difficult problem lies in how to find multiple trees such that the overall video quality is optimized. To address this problem, we take an application-centric, cross-layer approach to formulate MD video multicast routing as a combinatorial optimization problem. In contrast to previous work [19,21,22,25], our objective is to optimize the application layer performance metric, i.e., video distortion. It turns out that the optimization problem is highly complex and is expected to be NP-complete. As a result, it is desirable to develop efficient heuristic algorithms in practice. We find that *Genetic Algorithms* (GA) [4] are eminently suitable for addressing problems of this type, which 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. Section 4 presents our performance studies of the proposed approach. Related work is discussed in Section 5, and Section 6 concludes the paper.

2. Problem formulation

We model an ad hoc network with N nodes and L links as a time-varying directed graph $\mathcal{G}(\mathcal{N}, \mathcal{L})$. Accurate and computationally efficient characterization of an end-to-end path (or even a single-hop wireless link [12]) in an 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 layer and MAC layer dynamics are translated into network layer parameters. For example, we could characterize a link $\{i, j\} \in \mathcal{L}$ by the available capacity of link $\{i, j\}$, c_{ij}; the probability with which link $\{i, j\}$ fails, p_{ij}; the fixed, or the minimum delay of link $\{i, j\}$, τ_{ij}; the mean delay of link $\{i, j\}$, t_{ij}; and the jitter of link $\{i, j\}$, δ_{ij}^2. We focus on bandwidth, failure probability, and delay because these are the key characteristics of wireless links, as well as key factors that determine video distortion (see Section 2.1). Table 1 lists the notation used in the paper.

Table 1
Notation

$\mathcal{G}\{\mathcal{N}, \mathcal{L}\}$:	graph representation of the network
$\mathcal{N}$	the set of vertices in the network
$\mathcal{L}$	the set of edges
s	source node
$\mathcal{M}$	the set of receivers
r	a tagged receiver
$\{i, j\}$	a wireless link from node i to j
c_{ij}	available capacity of link $\{i, j\} \in \mathcal{L}$
p_{ij}	probability with which link $\{i, j\}$ fails
τ_{ij}	fixed, or minimum delay of link $\{i, j\}$
t_{ij}	mean delay of link $\{i, j\}$
δ_{ij}	jitter of link $\{i, j\}$
d_0	distortion when both descriptions are received
d_h	distortion when only Description h is received
σ^2	variance of the video source
$\mathcal{T}_h$	the h-th multicast tree
$\mathcal{P}_r^h$	the path from s to r in Tree h
B_{base}^h	bandwidth of the base layer of Description h
B_{tot}^h	total bandwidth of Description h
R_{base}^h	bit rate of the base layer of Description h
R_{tot}^h	total bit rate of Description h
P_{00}^b	probability of receiving both base layers
P_{00}^e	probability of receiving both enhancement layers
P_{01}^b	probability of receiving Desc. 1's base layer only
P_{01}^e	probability of receiving Desc. 1's enhancement layer only
P_{10}^b	probability of receiving Desc. 2's base layer only
P_{10}^e	probability of receiving Desc. 2's enhancement layer only
P_{11}^b	probability that both base layers are lost
P_{11}^e	probability that both enhancement layers are lost
D_r	the average distortion of Receiver r
x_{ij}^h	routing index variable associated with link $\{i, j\}$ in Tree h
u_i^h	routing index variable associated with node i in Tree h
Δ_r	the decoding deadline of Receiver r

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, each being encoded into two layers. For a receiver, denote d_h the achieved distortion when only Description h is received, $h = 1, 2$, and d_0 the distortion when both descriptions are received. Note that when both descriptions are lost, the distortion is σ^2. Clearly, d_0, d_1, and d_2 are functions of the description rates. 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 layers) of Description h, $h = 1, 2$. For the two base layers (each from a description), denote

P_{00}^b the probability of receiving both base layers; P_{01}^b the probability of receiving Description 1's base layer only; P_{10}^b the probability of receiving Description 2's base layer only; and P_{11}^b the probability of receiving neither of these base layers. For the two enhancement layers (each from a description), we define the probabilities P_{00}^e, P_{01}^e, P_{10}^e, and P_{11}^e in the same fashion.

As will be shown in Section 2.2, we choose the base layer rates for the two descriptions such that any receiver can have enough path bandwidths for both base layers. However, a receiver having a low path bandwidth from a tree may not be able to receive the enhancement layer from that tree (e.g., the enhancement layer is dropped at an upstream bottleneck link in the tree). According to a receiver's two path bandwidths and the rates of the layers, the layers it can receive can be classified into the following four cases (see the example in figure 1):

- Case I: it can receive base layers and enhancement layers from both trees (e.g., Receiver 1 in figure 1);
- Case II: it can receive base and enhancement layers from Description 1 but only base layer from Description 2 (e.g., Receiver 2 in figure 1);
- Case III: it can receive base and enhancement layers from Description 2 but only base layer from Description 1 (e.g., Receiver 3 in figure 1);
- Case IV: it can only receive the base layers from the two descriptions (e.g., Receiver 4 in figure 1).

Then, for Case I, the average distortion can be computed as:

$$\begin{aligned} D_r = {} & P_{00}^b\{P_{00}^e d_0(R_{\text{tot}}^1, R_{\text{tot}}^2) + P_{01}^e d_0(R_{\text{tot}}^1, R_{\text{base}}^2) \\ & + P_{10}^e d_0(R_{\text{base}}^1, R_{\text{tot}}^2) + P_{11}^e d_0(R_{\text{base}}^1, R_{\text{base}}^2)\} \\ & + P_{01}^b\{[P_{00}^e + P_{01}^e] d_1(R_{\text{tot}}^1) + [P_{10}^e + P_{11}^e] d_1(R_{\text{base}}^1)\} \\ & + P_{10}^b\{[P_{00}^e + P_{10}^e] d_2(R_{\text{tot}}^2) + [P_{01}^e + P_{11}^e] d_2(R_{\text{base}}^2)\} \\ & + P_{11}^b \sigma^2, \end{aligned} \tag{1}$$

For Case II, the average distortion can be computed as:

$$\begin{aligned} D_r = {} & P_{00}^b\left\{P_{01}^e d_0(R_{\text{tot}}^1, R_{\text{base}}^2) + P_{11}^e d_0(R_{\text{base}}^1, R_{\text{base}}^2)\right\} \\ & + P_{01}^b\left\{P_{01}^e d_1(R_{\text{tot}}^1) + P_{11}^e d_1(R_{\text{base}}^1)\right\} \\ & + P_{10}^b d_2(R_{\text{base}}^2) + P_{11}^b \sigma^2. \end{aligned} \tag{2}$$

The average distortion for Case III can be computed similarly as (2) due to symmetry. For Case IV, the average distortion can be computed as:

$$\begin{aligned} D_r = {} & P_{00}^b d_0(R_{\text{base}}^1, R_{\text{base}}^2) + P_{01}^b d_1(R_{\text{base}}^1) \\ & + P_{10}^b d_2(R_{\text{base}}^2) + P_{11}^b \sigma^2. \end{aligned} \tag{3}$$

The rate-distortion region for an *i.i.d.* memoryless Gaussian source with the square error distortion measure was first introduced in [17]. For computational efficiency, Alasti et al. [1] used the following rate-distortion region, which we use in this paper.

$$\begin{cases} d_0(R_1, R_2) = \dfrac{2^{-2(R_1+R_2)}}{2^{-2R_1} + 2^{-2R_2} - 2^{-2(R_1+R_2)}} \cdot \sigma^2 \\ d_1(R_1) = 2^{-2R_1} \cdot \sigma^2 \\ d_2(R_2) = 2^{-2R_2} \cdot \sigma^2. \end{cases} \tag{4}$$

2.2. *Computing end-to-end statistics*

Consider a multicast session with sender s and a set of receivers $\mathcal{M}$. 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 $\mathcal{P}_r^h$, $h = 1, 2$. To model end-to-end delay, we follow the approach in [9]. For receiver r, the end-to-end delay on its path $\mathcal{P}_r^h$, denoted as t_r^h, could be modeled as a "shifted" Gamma distribution:

$$y(t_r^h) = \frac{\alpha_r^h}{\Gamma(n_r^h)} \left[\alpha_r^h \cdot (t_r^h - \tau_r^h)\right]^{n_r^h - 1} e^{-\alpha_r^h \cdot (t_r^h - \tau_r^h)}, \tag{5}$$

for $t_r^h \geq \tau_r^h$, $h = 1, 2$. The end-to-end delay from sender s to receiver r can be interpreted as the total delay of going through n_r^h nodes, each with a processing delay of τ_r^h / n_r^h and an exponentially distributed queueing delay (with mean α_r^h). The parameters of the shifted Gamma distribution can be estimated from the link statistics as:

$$\begin{cases} \tau_r^h = \sum_{\{i,j\} \in \mathcal{P}_r^h} \tau_{ij} & h = 1, 2, \forall r \in \mathcal{M}, \\ \alpha_r^h = \dfrac{\sum_{\{i,j\} \in \mathcal{P}_r^h} t_{ij} - \sum_{\{i,j\} \in \mathcal{P}_r^h} \tau_{ij}}{\sum_{\{i,j\} \in \mathcal{P}_r^h} \delta_{ij}^2} & h = 1, 2, \forall r \in \mathcal{M}, \\ n_r^h = \dfrac{\left(\sum_{\{i,j\} \in \mathcal{P}_r^h} t_{ij} - \sum_{\{i,j\} \in \mathcal{P}_r^h} \tau_{ij}\right)^2}{\sum_{\{i,j\} \in \mathcal{P}_r^h} \delta_{ij}^2} & h = 1, 2, \forall r \in \mathcal{M}. \end{cases}$$

Success Probabilities: As indicated in (1) and (4), the video distortion is the highest when both descriptions are lost, since σ^2 is generally much larger than d_0, d_1, and d_2. In order to reduce the possibility of simultaneously losing both descriptions, the correlation of the loss processes of the two descriptions should be minimized at best [3]. It has been shown in previous work, e.g., [20], 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_r by receiver r (excluding the constant interleaving delay) from tree h is:

$$q_r^h = \left[\prod_{\{i,j\} \in \mathcal{P}_r^h} \left(1 - p_{ij}\right) \right] \cdot \Pr\left(t_r^h \le \Delta_r\right), h = 1, 2. \quad (6)$$

For Case I where the receiver can receive base and enhancement layers for both descriptions, the joint probabilities of receiving the layers can be computed as:

$$\begin{cases} P_{00}^b = q_r^1 \cdot q_r^2 \\ P_{01}^b = q_r^1 \cdot \left(1 - q_r^2\right) \\ P_{10}^b = \left(1 - q_r^1\right) \cdot q_r^2 \\ P_{11}^b = \left(1 - q_r^1\right) \cdot \left(1 - q_r^2\right) \end{cases} \text{and} \begin{cases} P_{00}^e = q_r^1 \cdot q_r^2 \\ P_{01}^e = q_r^1 \cdot \left(1 - q_r^2\right) \\ P_{10}^e = \left(1 - q_r^1\right) \cdot q_r^2 \\ P_{11}^e = \left(1 - q_r^1\right) \cdot \left(1 - q_r^2\right). \end{cases} \quad (7)$$

For the remaining three cases, the probabilities of receiving the base layers are the same as in Case I. For the probabilities of receiving the enhancement layers, we have the following according to which enhancement layer can be received: (i) Case II: $P_{00}^e = 0$, $P_{01}^e = q_r^1$, $P_{10}^e = 0$, and $P_{11} = (1 - q_r^1)$; (ii) Case III: $P_{00}^e = 0$, $P_{01}^e = 0$, $P_{10}^e = q_r^2$, and $P_{11} = (1 - q_r^2)$; (iii) Case IV: $P_{00}^e = 0$, $P_{01}^e = 0$, $P_{10}^e = 0$, and $P_{11}^e = 1$.

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

$$B_r^{jnt} = \begin{cases} B\left(\mathcal{J}\left(\mathcal{P}_r^1, \mathcal{P}_r^2\right)\right) & \text{if } \mathcal{J}(\mathcal{P}_r^1, \mathcal{P}_r^2) \neq \emptyset; \\ \infty & \text{otherwise,} \end{cases} \quad (8)$$

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

$$\begin{cases} B_r^h = B\left(\mathcal{P}_r^h\right) & \text{if } \sum_{h=1}^{2} B\left(\mathcal{P}_r^h\right) \le B_r^{\text{jnt}},\ h = 1, 2; \\ B_r^1 + B_r^2 \le B_r^{\text{jnt}} & \text{otherwise.} \end{cases} \quad (9)$$

The first case of (9) is for the situation when the joint links are not the bottleneck of the paths, while the second case of (9) is for the situation 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,

[1] It has been shown in [6] that the loss correlation of two descriptions is quite low once the two paths split after the first set of shared links.

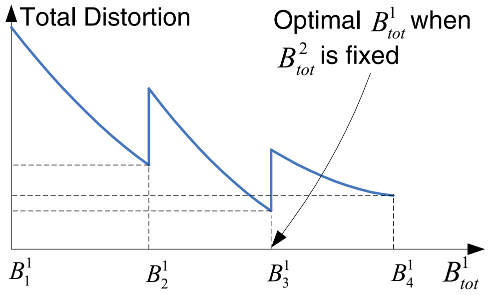

Figure 2. Optimal B_{tot}^1 for a fixed B_{tot}^2.

we set the base layer bandwidth of Description h to:

$$B_{\text{base}}^h = \min_{r \in \mathcal{M}} \left\{B_r^h\right\}, \quad h = 1, 2. \quad (10)$$

The total bandwidth of Description h, B_{tot}^h, should be within the range $[B_{\text{base}}^h, \max_{r \in \mathcal{M}}\{B_r^h\}]$. For a chosen B_{tot}^h, receivers that satisfy $B_r^h \ge B_{\text{tot}}^h$ can receive both layers of Description h; other receivers with $B_r^h < B_{\text{tot}}^h$ 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_{tot}^h, $h = 1, 2$ [14]. Therefore, for a fixed B_{tot}^h, the total distortion of all receivers is a piece-wise non-increasing function of B_{tot}^{3-h} with discontinuous jumps at B_r^{3-h}, $r \in \mathcal{M}$. An example with four receivers is illustrated in figure 2, where the total distortion is plotted as a function of B_{tot}^1 for a fixed B_{tot}^2, assuming that $B_1^h < B_2^h < B_3^h < B_4^h$, $h = 1, 2$. In this example, B_{base}^1 is set to B_1^1, as given in (10). The total distortion is the highest when $B_{\text{tot}}^1 = B_1^1$, since no enhancement layer for Description 1 can be received by any of the receivers. If we fix B_{tot}^2 and increase B_{tot}^1, the total distortion keeps on decreasing, due to the monotonicity property of (1) [14]. When B_{tot}^1 reaches B_2^1, 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_{tot}^2, we only need to evaluate the total distortion at three points, i.e., $B_{\text{tot}}^1 = B_r^1$, $r = 2, 3, 4$, in order to find the optimal B_{tot}^1, which is $B_{\text{tot}}^1 = B_3^1$ in this example.

Figure 3 plots the total distortion for all feasible combinations of B_{tot}^1 and B_{tot}^2. We find that the same monotonicity property holds true in this case. Due to the monotonicity property of (1), we only need to examine the total distortion at points $\{B_i^1, B_j^2\}$, $i, j \in \{2, 3, 4\}$, 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 $(K_1 - 1) \cdot (K_2 - 1)$ 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).

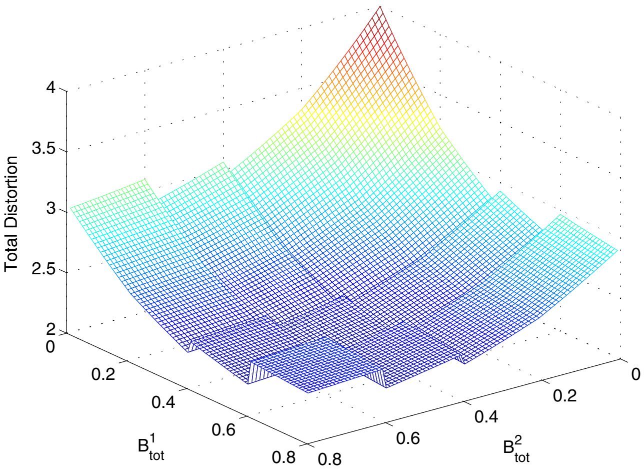

Figure 3. Total distortion for all combinations of B^1_{tot} and B^2_{tot}.

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:

$$\begin{cases} R^h_{\text{tot}} = \rho \cdot B^h_{\text{tot}}, & h = 1, 2, \\ R^h_{\text{base}} = \rho \cdot B^h_{\text{base}}, & h = 1, 2, \end{cases} \tag{11}$$

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 as determined by the chroma subsampling format (e.g., $\gamma = 1.5$ for the quarter common intermediate format (QCIF)).

2.3. The multicast routing problem

For constructing the problem formulation, we define the following two sets of variables in order to describe the choice of trees. For every link $\{i, j\} \in \mathcal{L}$, define

$$x^h_{ij} \overset{\text{def}}{=} \begin{cases} 1, & \text{if } \{i, j\} \in \mathcal{T}_h; \\ 0, & \text{otherwise}, \quad h = 1, 2. \end{cases} \tag{12}$$

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

$$u^h_i \overset{def}{=} \begin{cases} \text{number of hops from } s \text{ to } i, & i \in \mathcal{T}_h \\ 0, \quad i \notin \mathcal{T}_h, & h = 1, 2. \end{cases} \tag{13}$$

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 the receivers in $\mathcal{M}$ is minimized. That is:

$$\text{Minimize:} D = \sum_{r \in \mathcal{M}} D_r \tag{14}$$

$$\text{subject to:} x^h_{ij} \leq \sum_{k:\{k,i\} \in \mathcal{L}} x^h_{ki}, \forall i \neq s, \quad \forall j \notin \{i, s\} : \{i, j\} \in \mathcal{L}, h = 1, 2 \tag{15}$$

$$\sum_{j:\{j,i\} \in \mathcal{L}} x^h_{ji} = 1, \; i \in \mathcal{M}, h = 1, 2 \tag{16}$$

$$\sum_{j:\{j,i\} \in \mathcal{L}} x^h_{ji} \leq 1, \; \forall i \in \mathcal{N} \backslash \mathcal{M}, \; h = 1, 2 \tag{17}$$

$$u^h_i - u^h_j + N \cdot x^h_{ij} \leq N - 1, \; \forall i, j \in \mathcal{N}, h = 1, 2 \tag{18}$$

$$x^1_{ij} \cdot R^1_{\text{tot}} + x^2_{ij} \cdot R^2_{\text{tot}} = (1 - \epsilon) \cdot \rho \cdot c_{ij}, \quad \text{for some } \epsilon \in [0, 1], \forall \{i, j\} \in \mathcal{L} \tag{19}$$

$$x^h_{ij} \in \{0, 1\}, \forall \{i, j\} \in \mathcal{L}, h = 1, 2 \tag{20}$$

$$u^h_i \in \{0, 1, \cdots, N - 1\}, \forall i \neq s, h = 1, 2. \tag{21}$$

In Problem OPT-MM, constraints (15)–(18) represent the choice of a pair of trees. More specifically, Constraint (15) 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 (16) guarantees that all member nodes are connected in the tree (with a single parent node); constraints (17) and (18) ensure the loop-free property of trees. Constraint (19) 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. Such a problem has been shown to be NP-complete in [24]. It is also worth noting that our problem is a far more complex variant of the traditional Steiner tree design problem [11]. Therefore, we expect that Problem OPT-MM is also NP-complete.[2] Furthermore, we note that for enhancing the model from the viewpoint of solvability, there exist several ways of tightening the representation of the constraints (15)–(21) with respect to the convex hull of feasible solutions as expounded by Sherali and Driscoll [23], for example. While this could be useful in a mathematical programming approach for solving Problem OPT-MM, or some relaxation thereof, we will be pursuing a metaheuristic procedure in the sequel.

3. A metaheuristic solution

An effective strategy to address Problem OPT-MM is to view it as a "black-box" optimization problem and to explore an effective *metaheuristic* approach [7]. In particular, we find that *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.

Coding and Initialization: Under GAs, it is critical to represent a solution in a proper form. In our approach, a solution for Problem OPT-MM is a pair of trees. In our implementation,

[2]However, we leave a formal proof of this claim in future research.

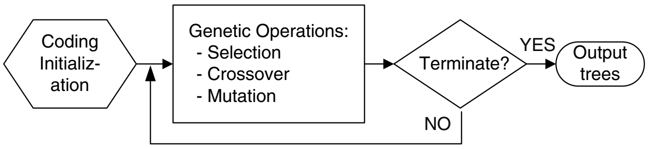

Figure 4. The GA-based multicast routing.

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.[3]

With such 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 values. 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 experimented with both the *Tournament* scheme, where we randomly pick $k = 2$ individuals from the population and choose the one having a higher fitness value, and the *Roulette* scheme, where each individual is chosen with a probability in proportion to its fitness value. We found that the differences among the individual fitness values were generally small. With Roulette selection, the selection probabilities tend to be close to a uniform distribution, which decreases the *selection pressure* and slows down the convergence. However, using Tournament selection we are able to keep sufficient selection pressure as well as population diversity.

Crossover mimics the genetic mechanism of reproduction in the natural world, where genes from parents are recombined and passed to the offspring. 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 receiver 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 gene pool of the population, thus keeping the computation from being trapped at 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 will be 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 the need for executing the routing process for every tree interruption [21].

4. Performance evaluation

In this section, we compare the performance of the GA-based multicast scheme with two representative network-centric algorithms, which are based on Dijkstra's shortest path routing algorithm. The first algorithm, the *Bounded Shortest Multicast Algorithm* (BSMA) algorithm presented in [19], is designed to construct minimum-cost multicast trees with delay constraints, and has been shown to achieve the lowest cost among several existing algorithms [22]. 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, in order to find a tree with the optimal loss characteristics and another tree with the largest bandwidths. The same decoding deadline Δ was used as a delay bound in BSMA.

[3] The adjacency matrix data structure has an $O(N^2)$ storage requirement. The tree-list data structure as described in [5] can be used to reduce the storage to $O(3N)$.

In another work [21], Sajama and Haas presented the *Independent-Tree Ad Hoc Multicast Routing* (ITAMAR) procedure 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, so that the time between a tree failure and rerouting is minimized. Among the several algorithms proposed in [21], we implement 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 we compute the achieved distortion using (1).

4.1. Performance comparison

For each experiment, we generate a wireless ad hoc network by placing a number of nodes at random locations in a square region. Connectivity is determined by the distance coverage of each node's transmitter. The source node s and the receivers $r \in \mathcal{M}$ are randomly chosen. For every link, the failure probability is randomly chosen from [0,1] with a truncated exponential distribution (having a mean of 0.01); and the available link bandwidth is 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 are set to 5 ms and 30 ms, respectively; the jitter δ_{ij} is randomly chosen from [7ms, 17ms] , $\forall\{i, j\} \in \mathcal{L}$. We set σ^2 to 1, since it only affects the absolute value of distortion, but does not affect path selection decisions.

The achieved average distortions by the various schemes are listed in Table 2, each being the average of 10 runs, for different network sizes and multicast groups. We find that the GA-based approach significantly outperformed 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 an optimal application layer performance. An interesting observation from Table 2 is that the improvement achieved by the GA-based multicast routing over other schemes is higher for sparse multicast sessions. This is because for sparse sessions, fewer receivers are involved and there is a greater freedom for the GA-based routing to select links, while the two network-centric algorithms do not explore this freedom very well since they are constrained by their shortest path routing objective.

4.2. GA versus ITAMAR-SPTH

Among the three schemes listed in Table 2, ITAMAR-SPTH has a performance closest to the GA-based routing in terms of distortion values.Therefore, we run ITAMAR-SPTH and the GA-based routing for a five-member group in a 15-node network, and compare the PSNRs of the reconstructed video frames, in order to demonstrate the efficiency of the proposed scheme. We use an H.263+ codec (originally from the University of British Columbia (UBC)) and the 400-frame "Foreman" trace in the QCIF format. We employ a particularly efficient and practical scheme that is based on the time-domain partitioning coding, where multiple descriptions are generated by separating the video frames and coding them separately [2,6,8,14,15]. A double-description coding scheme using this technique is illustrated in Figure 5, in which the arrows indicate the coding dependency of the frames. The video sequence is encoded with a frame rate of 30 fps and an intra MB refresh rate of 1/10.When necessary, the SNR scalable coding is used to code each description into two layers. We implement the off-line rate control for the enhancement layer, which is not available in the original UBC distribution. Each group of blocks (GOB) is transmitted in a packet to make them independently decodable.

The qualities of the trees found by both schemes are presented in Table 3. In general, 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 almost 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. On average, the GA-based multicast routing achieves a 3.72 dB improvement in PSNR over the ITAMAR-SPTH algorithm in this experiment.

Table 2
Average distortion achieved by GA and the two existing approaches.

Ad hoc network	15-node ($\Delta = 100$ ms)		30-node ($\Delta = 150$ ms)		50-node ($\Delta = 250$ ms)	
Multicast group	Dense (12)	Sparse (3)	Dense (25)	Sparse (5)	Dense (40)	Sparse (10)
BSMA	0.664	0.704	0.892	0.780	0.708	0.788
ITAMAR-SPTH	0.655	0.637	0.721	0.609	0.702	0.824
GA-based Routing	0.412	0.360	0.515	0.485	0.533	0.528

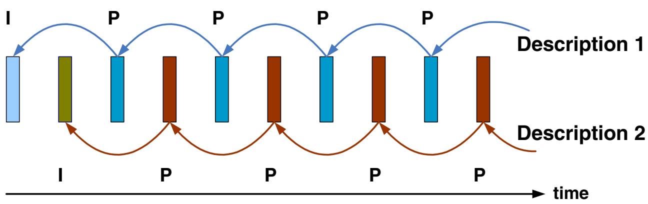

Figure 5. The MD coding scheme.

Table 3
GA-based routing versus ITAMAR-SPTH.

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

We also plot the PSNR curves for three representative receivers in figures 6, 7 and 8, respectively.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 Frame 226 obtained by receivers 1, 2, and 3 in figure 9. For all the receivers, the perceived quality obtained by the GA-based routing is much better 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 computational 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

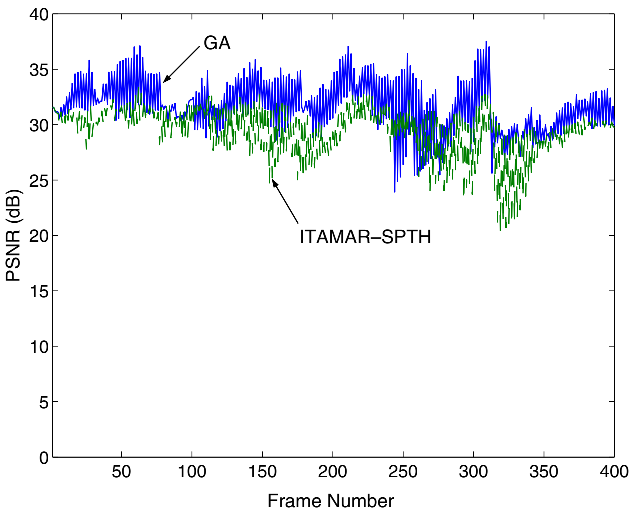

Figure 6. PSNRs of received frames by Receiver 1.

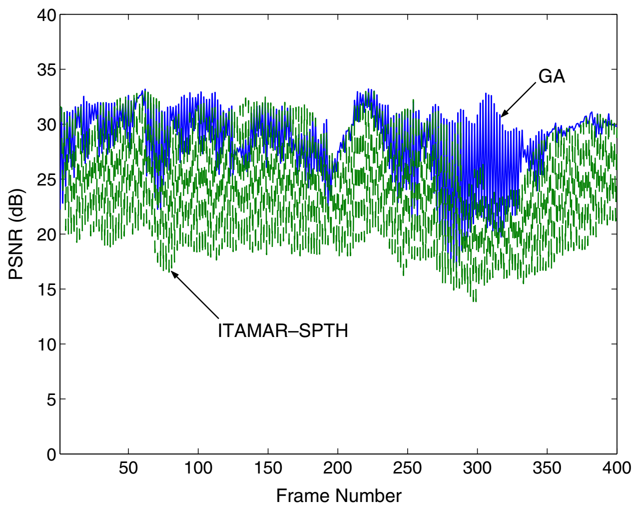

Figure 7. PSNRs of received frames by Receiver 2.

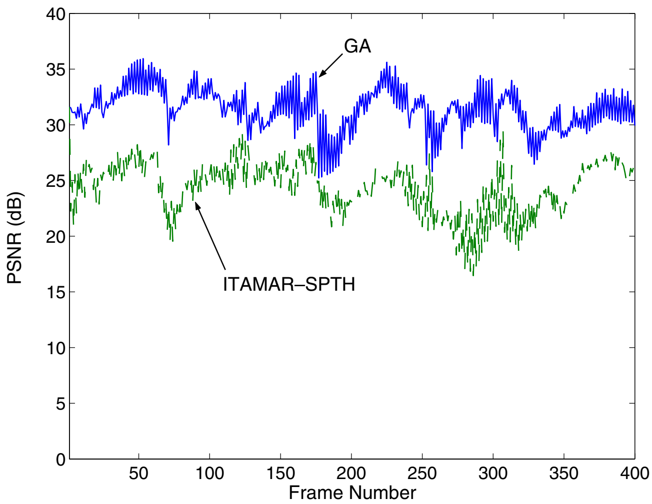

Figure 8. PSNRs of received frames by Receiver 3.

(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.

to evolve, the trees can be dynamically updated with newly computed (better) trees for enhanced performance.

5. Related work

The most relevant work to this research is ITAMAR by Sajama and Haas [21], which has been discussed in detail in Section 4. In this section, we discuss some other related work that contribute to further background knowledge of this investigation.

Multicasting MD video was first discussed in CoopNet [18] 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 for 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 ad hoc networks for the following reasons. First, the main design objective of CoopNet is to make 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 is not optimized in terms of video quality. As a result, such an approach cannot be optimal in 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 an ad hoc network 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 its worst link.

MD video streaming has been an active research area due to MD video's unique error resilience and open-loop operation capabilities [2,3,8,6,14,15]. An empirical MD rate-distortion model has been presented in [3] for computing average video distortions from loss probabilities of path links. The scheme in [6] 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 [8,9,15]. 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 [11], which are well-known to be NP-complete. Various efficient heuristic algorithms have been proposed (e.g., BSMA [19, 21]). We refer readers to [22] for a comparison study and [16] for a survey of multicast routing protocols in ad hoc networks. Most of the proposed algorithms 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 [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 earlier work [14], 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 the application and network layers.

6. Conclusions

In this paper, we proposed a multicast scheme for MD video over ad hoc networks, within which multiple source trees are used. Furthermore, each video description is coded into multiple layers in order to cope with diversity in wireless link bandwidths. We followed an application-centric, cross-layer approach and formulated the MD multicast routing into a combinatorial optimization problem. Due to the complex nature of problem structure, we pursued to develop efficient metaheuristic algorithm instead of exact analytic solution. We found that Genetic Algorithms are highly suitable for such problems and consequently developed a GA-based solution procedure for the MD multicast routing problem. Extensive simulations demonstrate significant gains in video quality achieved over existing approaches for a wide range of network operational conditions.

7. Acknowledgments

The authors wish to thank Prof. Yao Wang of Polytechnic University, Brooklyn, NY, and Dr. Shunan Lin of Harmonic Inc. for helpful discussions and for providing the MD codec. This research has been supported in part by the National Science Foundation (NSF) under Grants ANI-0312655, CNS-0347390, and DMI-0094462, and by the Office of Naval Research (ONR) 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) (2001) 891–902.

[2] J. Apostolopoulos, Reliable video communication over lossy packet networks using multiple state encoding and path diversity. in: *Proc. SPIE VCIP* (2001) pp. 392–409.

[3] J. Apostolopoulos, W. Tan, S. Wee and G. Wornell, Modeling path diversity for multiple description video communication. in: *Proc. IEEE ICASSP*, (2002) pp. 13–17 .

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

[5] M. Bazaraa, J. Jarvis and H. Sherali, Linear Programming and Network Flows, second edition (John Wiley & Sons, Inc., New York, NY, 1990).

[6] A. Begen, Y. Altunbasak and O. Ergun, Multi-path selection for multiple description encoded video streaming. in: *Proc. IEEE ICC* (2003) pp. 1583–1589.

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

[8] J. Chakareski, S. Han and B. Girod, Layered coding vs. multiple descriptions for video streaming over multiple paths, in: *Proc. of ACM Multimedia* (2003) pp. 422–431.

[9] P. Chou and Z. Miao, Rate-distortion optimized streaming of packetized media, Technical report, Microsoft Research, MSR-TR-2001-35 (2001).

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

[11] F. Hwang, D. Richards and P. Winter, *The Steiner Tree Problem* (Annals of Discrete Mathematics, 53, North-Holland, 1992).

[12] A. Kopke, A. Willig and H. Karl, Chaotic maps as parsimonious bit error models of wireless channels, in: *Proc. IEEE INFOCOM* (2003) pp. 513–523.

[13] J. Liu, B. Li, Y. Hou and I. Chlamtac, On optimal layering and bandwidth allocation for multisession video broadcasting, IEEE Trans. Wireless Commun 3(2) (2004) 656–667.

[14] S. Mao, Y. Hou, X. Cheng, H. Sherali and S. Midkiff, Multi-path routing for multiple description video over wireless ad hoc networks in: *Proc. IEEE INFOCOM* (2005) pp. 740–750.

[15] 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)(2003) 1721–1737.

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

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

[18] V. Padmanabhan, H. Wang, P. Chou and K. Sripanidkulchai, Distributing streaming media content using cooperative networking, in: *Proc. ACM NOSSDAV* (2002) pp. 177–186.

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

[20] A. Reibman, H. Jafarkhani, M. Orchard and Y. Wang, Performance of multiple description coders on a real channel, in: *Proc. IEEE ICASSP* (1999) pp. 2415–2418.

[21] Sajama and Z. Haas, Independent–Tree ad hoc multicast routing (ITAMAR), Mobile Networks and Applications 8(5) (2003) 551–566.

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

[23] H. Sherali and P. Driscoll, On tightening the relaxations of Miller-Tucker-Zemlin formulations for asymmetric traveling salesman problems, Operations Research 50(4) (2002) 656–669.

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

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

Shiwen Mao received the B.S. degree in Electrical Engineering and the B.E. degree in Enterprise Management from Tsinghua University, Beijing, P. R. China in 1994. He received the M.S. degree in Electrical Engineering from Tsinghua University in 1997, and the M.S. degree in System Engineering from Polytechnic University, Brooklyn, NY, in 2000. He received the Ph.D. degree in Electrical Engineering from Polytechnic University in 2004.

He was a Research Member at IBM China Research Lab, Bejing from 1997 to 1998. In the summer of 2001, he was a research intern at Avaya Labs-Research, Holmdel, NJ. He was a research fellow at the New York State Center for Advanced Technology in Telecommunications (CATT) at Polytechnic University from 1998 to 2003. Currently, he is a Research Scientist in the Bradley Department of Electrical and Computer Engineering, Virginia Tech, Blacksburg, VA.

Dr. Mao's research interest includes performance analysis and algorithm design for the Internet and wireless networks, with a current focus on realtime multimedia transport over wireless networks.He is the lead recipient of *the 2004 IEEE Communications Society Leonard G. Abraham Prize in the Field of Communications Systems*. He co-authored a recent textbook, *TCP/IP Essentials: A Lab-Based Approach*, published by Cambridge University Press in Oct. 2004. He is a member of Tau Beta Pi and IEEE.
E-mail: smao@vt.edu

Xiaolin Cheng received his B.E. and M.E. degrees in automation from Tsinghua University, Beijing, China in 1997 and 2000 respectively. Currently, he is working toward the PhD degree in the Bradley Department of Electrical and Computer Engineering at Virginia Tech.

From 2000-2003, he was a member of technical staff at Panasonic Beijing Labs, where he developed system software for digital TV set-top-boxes. His current research interests include multipath and multicast routing in wireless ad hoc networks and video transmission over wireless ad hoc networks.Mr. Cheng is a student member of the IEEE.
E-mail: xiaolin@vt.edu

Thomas Hou obtained his B.E. degree (Summa Cum Laude) from the City College of New York in 1991, the M.S. degree from Columbia University in 1993, and the Ph.D. degree from Polytechnic University, Brooklyn, New York, in 1998, all in Electrical Engineering. From 1997 to 2002, Dr. Hou was a principal research scientist and project leader at Fujitsu Laboratories of America, IP Networking Research Department, Sunnyvale, California (Silicon Valley). Since Fall 2002, he has been an Assistant Professor at Virginia Tech, the Bradley Department of Electrical and Computer Engineering, Blacksburg, Virginia.

Dr. Hou's research interests are in the algorithmic design and optimization for network systems. His current research focuses wireless ad hoc networks, sensor networks, and video over ad hoc networks. In recent years, he has worked on scalable architectures, protocols, and implementations for differentiated services Internet; service overlay networking; multimedia streaming over the Internet; and network bandwidth allocation policies and distributed flow control algorithms. He has published over 100 journal and conference papers in the above areas and is a co-recipient of *the 2002 IEEE International Conference on Network Protocols (ICNP) Best Paper Award* and *the 2001 IEEE Transactions on Circuits and Systems for Video Technology (CSVT) Best Paper Award*. He is a member of ACM and a senior member of IEEE.
E-mail: thou@vt.edu

Hanif D. Sherali is the W. Thomas Rice Endowed Chaired Professor of Engineering in the Industrial and Systems Engineering Department at Virginia Polytechnic Institute and State University, Blacksburg, VA. His area of research interest is in discrete and continuous optimization, with applications to location, transportation, and engineering design problems. He has published about 200 papers in Operations Research journals, has co-authored four books in this area, and serves on editorial boards of eight journals. He is a member of the U.S. National Academy of Engineering.
E-mail: hanifs@vt.edu