Routing for Concurrent Video Sessions in *Ad Hoc* Networks

Shiwen Mao, *Member, IEEE*, Sastry Kompella, *Student Member, IEEE*, Y. Thomas Hou, *Senior Member, IEEE*, Hanif D. Sherali, and Scott F. Midkiff, *Senior Member, IEEE*

***Abstract*—In this paper, we consider the problem of how to optimally support multiple concurrent video sessions in an *ad hoc* network. Specifically, we present a framework on modeling the end-to-end video distortion as a function of routing layer behavior. Our formulation seamlessly integrates the interactions of competing video flows and network layer performance metrics, thus allowing optimal utilization of network resource. We describe a detailed solution procedure based on genetic algorithms (GAs), which allows joint determination of near-optimal routes and rates for the video sessions. We also present a greedy heuristic algorithm that can produce near-optimal solutions and speed up the GA computation. Through simulation results, we demonstrate the superior performance of the GA-based approach.**

***Index Terms*—*Ad hoc* networks, cross-layer optimization, genetic algorithms (GAs), routing, video communications.**

I. Introduction

With advances in digital video technology and wireless *ad hoc* networking, there is a compelling need to support real-time multimedia communications in *ad hoc* networks. Unlike wired networks (e.g., the Internet) or infrastructure based wireless networks, *ad hoc* networks do not require an infrastructure and can be deployed under harsh or denied areas. As a result, such networks may have frequent node and link failures and pose unique difficulties for multimedia communications.

There has been considerable research on enabling video service in wireless *ad hoc* networks [1]–[8]. Video applications have also been successfully demonstrated with *ad hoc* network testbeds [9], [10]. However, these prior efforts mainly focus on end system-based techniques, rather than optimal routing for video sessions. Many efficient protocols have been proposed for quality of service (QoS) routing in *ad hoc* networks, e.g., [11]–[14]. These efforts mainly focus on the optimization of one or more network layer metrics, such as throughput, delay, loss, or path correlation, and are therefore termed *network centric routing* throughout this paper. Although these protocols are shown to be quite effective for data communications, they may not provide good video quality due to the fact that video distortion is a highly complex function of *multiple* network layer metrics [15]. Optimizing network layer metrics does not necessarily guarantee optimal video quality.

It is important to consider the routing for multiple video sessions. This is because video flows compete for limited network resources. Such interactions make the performance of an individual flow coupled with that of other flows. By jointly consider the routing of concurrent sessions, we can optimize the network resource allocation among all flows and maximize a common performance objective.

In this paper, we consider the problem of supporting multiple concurrent video sessions in *ad hoc* networks. The first contribution of this paper is the formulation of an optimization problem from a cross-layer perspective by considering the application layer performance metric (i.e., average video distortion) as a function of network layer behavior (routing of each session). This formulation unifies video distortion with packet loss (due to node/link failures) and delay (due to congestion), while jointly considering the routing for concurrent sessions. The problem formulation exhibits a highly complex objective function and constraints, which renders this problem substantially more difficult than traditional QoS routing problems. We believe that the problem is NP complete, although its proof is not given in this paper.

The second contribution of this paper is the development of a highly competitive solution method based on metaheuristics, namely, genetic algorithms (GAs) [16]. GA-based algorithms have an intrinsic capability to handle a population of solutions, which perfectly suits the nature of our cross-layer concurrent routing problem. GA-based approaches have the unique strength of identifying promising search regions and have less of a tendency to be trapped at a local optimum, as compared with other single solution-based trajectory methods [16]. We find that the complex network wide optimization problem provides the perfect setting for a GA-based method. The complexity due to the interaction among concurrent sessions can be handled rather naturally by GAs because they are intrinsically parallel. Multiple session routing and flow interactions increase computational effort only linearly as compared with single session routing. We demonstrate the superior performance of the GA-based approach over other approaches through simulations, and present a distributed implementation.

The remainder of this paper is organized as follows. In Section II, we formulate the optimal routing problem with a cross-layer approach. We propose a solution procedure based on GAs in Section III and describe a fast greedy heuristic algorithm

Manuscript received April 4, 2005; revised May 29, 2005. This work was supported in part by the National Science Foundation under Grants ANI-0312655, CNS-0347390, DMI-0094462, and DGE-9987586, and Office of Naval Research under Grants N00014-03-1-0521 and N00014-05-1-0179. The review of this paper was coordinated by Prof. D. O. Wu.

S. Mao, S. Kompella, Y. T. Hou, and S. F. Midkiff are with the Bradley Department of Electrical and Computer Engineering, Virginia Polytechnic Institute and State University, Blacksburg, VA 24061 USA (e-mail: smao@vt.edu; sastryk@vt.edu; thou@vt.edu; midkiff@vt.edu).

H. D. Sherali is with the Grado Department of Industrial and Systems Engineering, Virginia Polytechnic Institute and State University, Blacksburg, VA 24061 USA (e-mail: hanifs@vt.edu).

Digital Object Identifier 10.1109/TVT.2005.861208

0018-9545/$20.00 © 2006 IEEE

in Section IV. Simulation results are presented in Section V. We discuss distributed implementation in Section VI and related work in Section VII. Section VIII concludes this paper.

II. Problem Formulation

In this section, we formulate the routing problem for multiple concurrent video sessions in a wireless *ad hoc* network. We assume a wireless link exists between nodes i and j if nodes i and j can communicate with (i.e., within the radio transmission range of) each other. Consequently, the *ad hoc* network 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 wireless links in the network.

In the graph, we characterize each wireless (directed) link $\{i, j\} \in \mathcal{L}$ by the following two parameters: 1) c_{ij} is the available bandwidth of link $\{i, j\}$; and 2) p_{ij} is the mean packet loss probability of link $\{i, j\}$, due to transmission errors or link failures. We assume lower layer dynamics (e.g., interference and fading) could be translated into the network layer metrics.

Consider a set of concurrent video sessions, denoted as $\mathcal{E}$. Each session $\sigma \in \mathcal{E}$ has a source node s_σ and a destination node d_σ. The rate of a video stream, R_σ, is bounded by $\underline{R}_\sigma \leq R_\sigma \leq \bar{R}_\sigma, \sigma \in \mathcal{E}$. The lower and upper bounds of R_σ are determined by the specific video coder and the video sequence encoded at the source node s_σ or the user requirement on received video quality. The decoding deadline for session σ packets is Δ_σ. The optimal routing problem aims to find a set of paths for the video sessions, such that the total distortion of all video sessions is minimized.

In Section II-A, we derive the end-to-end delay and packet loss rate for a session. In Section II-B, we link video distortion, which is an application-level performance metric, to network level delay and loss parameters. In Section II-C, we formulate the cross-layer optimal routing problem. Table I summarizes the notation used in the paper.

TABLE I
NOTATION

Symbol	Definition
$\mathcal{G}\{\mathcal{N}, \mathcal{L}\}$:	Graph representation of an ad hoc network.
$\mathcal{N}$:	Set of vertices.
$\mathcal{L}$:	Set of edges.
$\{i, j\}$:	A wireless link from node i to node j.
c_{ij} :	Available bandwidth of link $\{i, j\}$.
p_{ij} :	Packet loss probability of link $\{i, j\}$.
λ_{ij} :	Average aggregate traffic load on link $\{i, j\}$.
ρ_{ij} :	Utilization of link $\{i, j\}$.
$\mathcal{E}$:	Set of video sessions.
$\{s_\sigma, d_\sigma\}$:	Source/destination nodes of session σ.
x_{ij}^σ :	Index variable defined in (12).
$\mathcal{P}_\sigma$:	Path of session σ, from s_σ to d_σ.
p_σ :	End-to-end loss rate of session σ.
R_σ :	Rate of video session σ.
$\{\underline{R}_\sigma, \bar{R}_\sigma\}$:	The min/max rates of video session σ.
Δ_σ :	Decoding deadline of session σ.
t_{ij} :	Delay on link $\{i, j\}$.
$f_{ij}(y)$:	Probability density function of t_{ij}.
$M_{ij}(s)$:	Moment generating function of t_{ij}.
T_σ :	End-to-end delay of session σ.
$M_\sigma(s)$:	Moment generating function of T_σ.
D_σ^e :	End-to-end distortion of session σ.
D_σ^{enc} :	Encoding distortion of session σ.
D_σ^{cg} :	Session σ distortion due to congestion.
D_σ^{loss} :	Session σ distortion due to packet loss.
θ :	Crossover rate in the GA-based approach.
μ :	Mutation rate in the GA-based approach.

A. Network Layer Performance Metrics

1) Load on a Link: Let $\bar{\mathcal{P}}_\sigma^{ij}$ denote the upstream *partial* path of $\mathcal{P}_\sigma$ from the source node s_σ to the link $\{i, j\}$, exclusive. Note that $\bar{\mathcal{P}}_\sigma^{ij} = \emptyset$ if link $\{i, j\} \notin \mathcal{P}_\sigma$. Then, the average aggregate traffic load on any link $\{i, j\} \in \mathcal{L}$ is

$$\lambda_{ij} = \sum_{\sigma \in \mathcal{E}} R_\sigma \cdot \prod_{\{m,n\} \in \bar{\mathcal{P}}_\sigma^{ij}} (1 - p_{mn}). \tag{1}$$

That is, the average traffic load of link $\{i, j\}$ is the sum of the average rates of the video sessions that pass through this link, decreased by the losses incurred in their upstream links before reaching link $\{i, j\}$. The average capacity utilization of link $\{i, j\}$ is $\rho_{ij} = \lambda_{ij}/c_{ij}, \{i, j\} \in \mathcal{L}$. For stability, a feasible set of routes $\{\mathcal{P}_\sigma\}_{\sigma \in \mathcal{E}}$ should satisfy $\rho_{ij} < 1, \{i, j\} \in \mathcal{L}$.

2) Delay on a Link: We model each link $\{i, j\}$ as a general queuing system with an average input rate λ_{ij} [defined in (1)] and a service capacity c_{ij}. Let the queueing delay on link $\{i, j\}$ be t_{ij} and its probability density function be $f_{ij}(y)$. We assume all moments of t_{ij} are finite, which is true for most queueing systems. For example, when the video traffic is a constant bit rate (CBR) that exhibits short-range dependent (SRD) characteristics, we could model the queueing delay via an exponential distribution, that is

$$f_{ij}(y) = \alpha_{ij} \cdot e^{-\alpha_{ij} y}, \quad \text{for } y \geq 0 \tag{2}$$

where $\alpha_{ij} \stackrel{\text{def}}{=} (c_{ij} - \lambda_{ij})$ is the link's residual capacity. However, for a variable bit rate (VBR) video that exhibits long-range dependent (LRD) characteristics, we could model the link as a fractional Brownian motion (fBm) queueing system, where t_{ij} has a heavy-tailed Weibull distribution [17].

3) End-to-End Delay: The end-to-end delay of session σ, denoted by $T_\sigma, \sigma \in \mathcal{E}$, is the sum of the queueing delay on each link along path $\mathcal{P}_\sigma$, that is

$$T_\sigma = \sum_{\{i,j\} \in \mathcal{P}_\sigma} t_{ij}, \quad \forall \sigma \in \mathcal{E}. \tag{3}$$

We apply the large deviation approximation to obtain an accurate estimate of the overdue probabilities. For the sum of a small number of independent random variables, an estimate based on the *Chernoff bound* [18] is known to be accurate and computationally efficient [19]. In the following, we illustrate such an approximation when link delays are exponentially distributed.

We first derive the moment generating function of t_{ij} as

$$M_{ij}(s) = \mathrm{E}[e^{st_{ij}}] = \frac{\alpha_{ij}}{\alpha_{ij} - s}, \quad \text{for } s < \alpha_{ij}. \tag{4}$$

Assuming delays on the links are independent, the moment generating function of T_σ is

$$M_\sigma(s) = \prod_{\{i,j\} \in \mathcal{P}_\sigma} M_{ij}(s), \quad \text{for } s < \min_{\{i,j\} \in \mathcal{P}_\sigma} \{\alpha_{ij}\}. \tag{5}$$

Define a function $F_\sigma(s)$ as

$$F_\sigma(s) = s\Delta_\sigma - \sum_{\{i,j\}\in\mathcal{P}_\sigma} \log M_{ij}(s). \tag{6}$$

Because $F''_\sigma(s) < 0$, for $s < \min_{\{i,j\}\in\mathcal{P}_\sigma}\{\alpha_{ij}\}$, $F_\sigma(s)$ is a strictly concave function with a unique maximum at s^*_σ. If $\Delta_\sigma > \mathrm{E}(T_\sigma)$ (i.e., the decoding deadline is larger than the average end-to-end delay on the path), we can determine s^*_σ by solving

$$F'_\sigma(s) = \Delta_\sigma - \sum_{\{i,j\}\in\mathcal{P}_\sigma} \frac{1}{\alpha_{ij} - s} = 0. \tag{7}$$

Because $F'_\sigma(\min_{\{i,j\}\in\mathcal{P}_\sigma}\{\alpha_{ij}\}) = -\infty < 0$ and $F'_\sigma(0) = \Delta_\sigma - \mathrm{E}(T_\sigma) > 0$, we have that $0 < s^*_\sigma < \min_{\{i,j\}\in\mathcal{P}_\sigma}\{\alpha_{ij}\}$. From the Chernoff bound [19], the tail distribution of T_σ can be approximated as

$$\Pr\{T_\sigma \geq \Delta_\sigma\} \approx \frac{\exp\{-F_\sigma(s^*_\sigma)\}}{s^*_\sigma \delta(s^*_\sigma)\sqrt{2\pi}}, \tag{8}$$

where $\delta^2(s) = (\partial^2 \log M_\sigma(s)/\partial s^2)$.

Note that the moment generating function of a heavy-tailed Weibull random variable does not exist (although all its moments are well defined). Therefore, the above Chernoff bound approach cannot be applied to delays having such distributions. However, the overdue probability can be computed by taking advantage of the *sub exponential* property. For example, for an independent and inentically distributed (i.i.d.) sequence of heavy-tailed Weibull random variables $\{X_1, \ldots, X_n\}$, we have that $\Pr[\sum_{k=1}^n X_k > x] \approx \Pr[\max_{\{1\leq k\leq n\}}\{X_k\} > x] \approx n \cdot \Pr[X_1 > x]$ [20].

4) End-to-End Loss Rate: Assuming the packet loss processes on the links are independent, the end-to-end loss probability of session σ can be computed as

$$p_\sigma = 1 - \prod_{\{i,j\}\in\mathcal{P}_\sigma} (1 - p_{ij}), \quad \forall\sigma \in \mathcal{E}. \tag{9}$$

B. End-to-End Video Rate-Distortion Model

In [15], Stuhlmuller *et al.* developed an empirical rate-distortion model for a hybrid motion compensated video encoder. For a video sequence encoded at a target coding rate R_σ, the average end-to-end distortion D^e_σ consists of the encoding distortion caused by the lossy video coder D^{enc}_σ and the distortion due to transmission errors, including the distortion caused by overdue packets (i.e., congestion) D^{cg}_σ, and the distortion caused by lost packets (i.e., caused by link failures or transmission errors) D^{loss}_σ. That is

$$D^e_\sigma = D^{\mathrm{enc}}_\sigma + D^{\mathrm{cg}}_\sigma + D^{\mathrm{loss}}_\sigma. \tag{10}$$

From [15] and the results derived in Section II-A, we have

$$D^e_\sigma = \underbrace{D_0 + \frac{\omega}{R_\sigma - R_0}}_{D^{\mathrm{enc}}_\sigma} + \underbrace{\kappa\cdot(1-p_\sigma)\cdot Pr(T_\sigma > \Delta_\sigma)}_{D^{\mathrm{cg}}_\sigma} + \underbrace{\kappa\cdot p_\sigma}_{D^{\mathrm{loss}}_\sigma} \tag{11}$$

where D_0, ω, R_0, and κ are constants for a specific video codec (with fixed encoding parameters) and video sequence, which can be determined by training and curve matching. Because the model in (10) takes into account the effects of INTRA coding and spatial loop filtering, it matches simulation results closely [15].

C. Global Optimal Routing Problem

For delineating an end-to-end path $\mathcal{P}_\sigma$ from s_σ to $d_\sigma, \sigma \in \mathcal{E}$, we define the following index variables:

$$x^\sigma_{ij} = \begin{cases} 1, & \text{if } \{i,j\} \in \mathcal{P}_\sigma, \quad \{i,j\} \in \mathcal{L} \\ 0, & \text{otherwise}, \quad \{i,j\} \in \mathcal{L}. \end{cases} \tag{12}$$

We can now mathematically formulate the problem of application-centric optimal routing for multiple concurrent video sessions.

OPT-CLR

$$\text{Minimize:} \quad D = \sum_{\sigma\in\mathcal{E}} D^e_\sigma \tag{13}$$

subject to:

$$\underline{R}_\sigma \leq R_\sigma \leq \bar{R}_\sigma, \quad \text{for } \sigma \in \mathcal{E} \tag{14}$$

$$\rho_{ij} \leq 1 - \epsilon, \quad \{i,j\} \in \mathcal{L} \quad \text{for some stability tolerance } \epsilon \tag{15}$$

$$\sum_{j:\{i,j\}\in\mathcal{L}} x^\sigma_{ij} - \sum_{k:\{k,i\}\in\mathcal{L}} x^\sigma_{ki} = \begin{cases} 1, & \text{if } i = s_\sigma \\ -1, & \text{if } i = d_\sigma \\ 0, & \text{otherwise} \end{cases} \quad i \in \mathcal{N}, \sigma \in \mathcal{E} \tag{16}$$

$$\sum_{j:\{i,j\}\in\mathcal{L}} x^\sigma_{ij} \begin{cases} \leq 1, & \text{if } i \neq d_\sigma \\ = 0, & \text{if } i = d_\sigma \end{cases}, \quad i \in \mathcal{N}, \sigma \in \mathcal{E} \tag{17}$$

$$x^\sigma_{ij} \in \{0,1\}, \quad \{i,j\} \in \mathcal{L}, \sigma \in \mathcal{E}. \tag{18}$$

In Problem OPT-CLR, the objective function (13) is the sum of the average distortion of all concurrent video sessions. Minimizing (13) achieves a best utilization of network resources, as well as the best overall quality for the video sessions. When optimizing the performance of multiple users, efficiency and fairness are usually orthogonal objectives (i.e., maximizing one may lead to significant decrease in the other). In this work, we choose efficiency as our optimization objective to better use the scarce network resources in *ad hoc* networks. It is worth noting that choosing a different objective function, such as $\min\max\{D^e_\sigma\}$ or an objective function in the form of a utility function $\sum_\sigma f(D^e_\sigma)$, does not change the solution procedure, which is presented in the next section.

There are two sets of optimization variables that form the space of feasible solutions: 1) the set of routing vectors $\{\mathbf{X}_\sigma\}_{\sigma\in\mathcal{E}}$ and 2) the set of rates of video sessions $\{R_\sigma\}_{\sigma\in\mathcal{E}}$. The set of inequalities in (14) gives the range of feasible rates for each video session. In the case of streaming stored video, we have that $\underline{R}_\sigma = R_\sigma = \bar{R}_\sigma$ because the rate is fixed. Inequality (15) is the stability constraint, which ensures that the link delays are

Bounded. The remaining constraints [i.e., (16)–(18)] guarantee that each path $\mathcal{P}_\sigma$ is loop free.[1]

The objective function (13) is a highly complex ratio of high-order polynomials of the x-variables. The objective evaluation of a set of feasible paths involves identifying the joint and disjoint links of the paths (to compute the traffic load on each link), which is only possible when all paths are completely determined. Wang and Crowcroft [21] proved that QoS routing problems having multiple additive and/or multiplicative metrics are NP complete. Our problem has an additive delay metric and a multiplicative loss metric. In addition, our problem has much more complex relationships pertaining to the contribution of any link to the objective function, as well as coupled session delays (rather than constant link delay metrics, as in [21]). As a result, we conjecture that problem OPT-CLR is NP complete, although its proof is not given here. In the rest of this paper, we present an effective heuristic algorithm to address this problem.

III. Genetic Algorithm-Based Solution Procedure

A. Preliminaries

GA is a *population-based* metaheuristic that is inspired by the *survival-of-the-fittest* principle, as derived from its natural evolution context. It has the intrinsic strength of dealing with a set of solutions (i.e., a population) at each step, rather than working with a single, current solution. In each iteration, a number of genetic operators are applied to the individuals of the current population to generate individuals for the next generation. In particular, GA uses genetic operators known as *crossover* to recombine two or more individuals to produce new individuals, and *mutation* to achieve a randomized self-adaptation of individuals. The driving force in GA is the *selection* of individuals based on their fitness (in the form of an objective function). Individuals with a higher degree of fitness will be more likely to be chosen as members of the population for the next generation. The basic assumption within this paradigm is that good solutions often share parts with optimal solutions. The survival-of-the-fittest principle ensures the overall quality of the population increases as the algorithm progresses from one generation to the next.

The potential of GA is best explained by comparing GA with alternative approaches, such as *simulated annealing* (SA) [22] and *tabu search* (TS) [23]. Although these methods have specific strengths in solving complex optimization problems, their performance, when applied to our problem, is sensitive to the neighborhood structure definition. In addition, such approaches have a stronger tendency of being trapped at a local optimum and have slow convergence, which could be attributed to the fact that only a single solution is handled at each iteration. We illustrate this point in Section V.

B. GA-Based Multiple-Session Routing

Fig. 1 depicts the flowchart for our GA-based approach to Problem OPT-CLR. Note that both crossover and mutation are

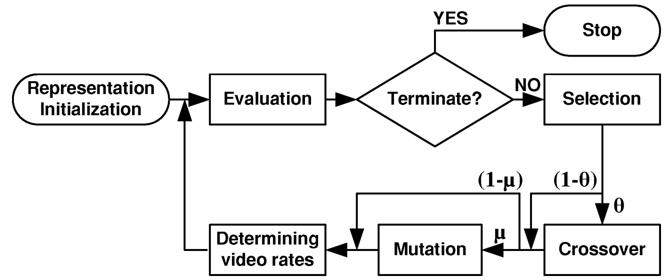


Fig. 1. Flowchart for GA-based routing procedure.

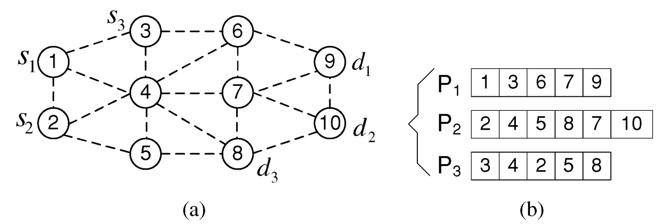


Fig. 2. (a) Example *ad hoc* network with dashed lines representing wireless links. (b) Feasible solution.

performed with certain probabilities (θ and μ, respectively) on the individual solutions. The termination condition in Fig. 1 could be based on the total number of iterations (generations), the maximum computing time, or a threshold of desired video distortion. In what follows, we use the example *ad hoc* network in Fig. 2(a) to illustrate the steps in the GA approach. There are three video sessions in the network, with source-destination pairs $\{1, 9\}, \{2, 10\}, \{3, 8\}$, respectively.

1) Representation and Initialization: To *encode* a feasible solution in the genetic format, we need to define a *gene* first and then map a solution to a sequence of genes (*chromosome*). Naturally, we define a node as a gene and an end-to-end path can be represented as a sequence of genes. Then, for the concurrent routing problem, each feasible solution (or *individual*) consists of a number of paths and, thus, a set of chromosomes [e.g., see Fig. 2(b)].

Then, we need to generate a set of initial solutions (or a *population*). A simple approach would be to randomly append feasible elements (i.e., nodes with connectivity) to a partial solution. Under this approach, a construction process would start with the source node s_σ. It would then randomly choose a link incident to the current end node of the partial path and append this link with its corresponding head node to augment the path, until the destination node d_σ is reached. It is important to ensure the intermediate partial path is loop free during the process. After generating a certain set of paths for each $\{s_\sigma, d_\sigma\}$ pair independently, a population of individuals for our problem can be constructed by randomly selecting paths from the set and verifying for stability conditions. Our numerical results show that a properly designed GA has a good exploratory power and is not very sensitive to the quality of the individuals in the initial population.

2) Evaluation: The fitness function $h(\bar{x})$ of an individual (i.e., $\bar{x} = [\mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3]$) is closely related to its objective function value (i.e., the total distortion D). Because the objective is to

[1] A feasible solution to these constraints could admit circuits whose edges are disconnected from the produced loop-free paths. However, the objective function would automatically prohibit this occurrence.

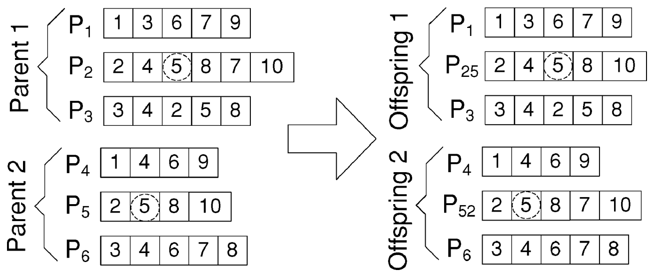


Fig. 3. Crossover operation.

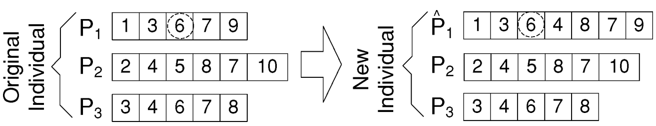


Fig. 4. Mutation operation.

minimize the total distortion [see (13)], we have adopted a fitness function that is defined as the inverse of the distortion value [i.e., $h(\bar{x}) = 1/D(\bar{x}]$.

3) Selection: During this operation, we select individuals that have a better chance or potential to produce "good" offspring in terms of their fitness values. We use the popular *Tournament* selection scheme [16], which randomly chooses m individuals from the population each time, and then selects the best of these m individuals in terms of their fitness values. By repeating this procedure multiple times, a new population can be selected.

4) Crossover: Crossover mimics the genetic mechanism of reproduction in the natural world, in which genes from parents are recombined and passed to offspring. The crossover operation may create new individuals, exposing the search process to a new area of the fitness landscape. Fig. 3 illustrates one possible crossover implementation. For two parent individuals $x_1 = [\mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3]$ and $x_2 = [\mathcal{P}_4, \mathcal{P}_5, \mathcal{P}_6]$, we could randomly pick a session, say session 2 ($\mathcal{P}_2$ in x_1 and $\mathcal{P}_5$ in x_2). If one or more common nodes exist in these two chosen paths, we could select the first such common node that exists in $\mathcal{P}_2$, say $g_r, g_r \notin \{s_2, d_2\}$ (node 5 in Fig. 3). We can then concatenate the nodes $\{s_2, \ldots, g_r\}$ from $\mathcal{P}_2$ with the nodes $\{g_{r+1}, \ldots, d_2\}$ in $\mathcal{P}_5$ (where g_{r+1} denotes the next downstream node of g_r in $\mathcal{P}_5$) to produce a new path $\mathcal{P}_{25}$. Likewise, using the first such node $g_{r'}$ in $\mathcal{P}_5$ that repeats in $\mathcal{P}_2$ (which may be different from g_r), we can concatenate the nodes $\{s_2, \ldots, g_{r'}\}$ from $\mathcal{P}_5$ with the nodes $\{g_{r'+1}, \ldots, d_2\}$ in $\mathcal{P}_2$ to produce a new path $\mathcal{P}_{52}$. The two new individuals generated in this manner are $[\mathcal{P}_1, \mathcal{P}_{25}, \mathcal{P}_3]$ and $[\mathcal{P}_4, \mathcal{P}_{52}, \mathcal{P}_6]$, as illustrated in Fig. 3. If $\mathcal{P}_2$ and $\mathcal{P}_5$ are disjoint, we could swap the entire path $\mathcal{P}_2$ with $\mathcal{P}_5$ instead.

5) Mutation: The objective of the mutation operation is to *diversify* the genes of the current population, which helps prevent the solution from being trapped at a local optimum. Fig. 4 illustrates the mutation of an individual $\bar{x} = [\mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3]$. First, we choose a path $\mathcal{P}_\sigma, \sigma \in \{1, 2, 3\}$, from $\bar{x}$ using equal probabilities of selection. Then, we randomly select an integer value r in the interval $[2, |\mathcal{P}_\sigma| - 1]$, where $|\mathcal{P}_\sigma|$ denotes the cardinality of $\mathcal{P}_\sigma$, and let the partial path $\{s_\sigma, \ldots, g_r\}$ be $\mathcal{P}_\sigma^u$, where g_r is the rth node along $\mathcal{P}_\sigma$. Finally, we could use any constructive approach to build a partial path from g_r to d_σ, denoted as $\mathcal{P}_\sigma^d$, that does not repeat any node in $\mathcal{P}_\sigma^u$ (other than g_r). If no such alternative segment exists between g_r and the destination node d_σ, we keep the path intact. Otherwise, a new path can now be created by concatenating the two partial paths as $\mathcal{P}_\sigma^u \cup \mathcal{P}_\sigma^d$. For the example in Fig. 4, $\mathcal{P}_1$ is chosen for mutation, and node 6 is chosen to be the mutation point, yielding a perturbed path $\hat{\mathcal{P}}_1$ that replaces $\mathcal{P}_1$. The new individual thus created is $\hat{x} = [\hat{\mathcal{P}}_1, \mathcal{P}_2, \mathcal{P}_3]$.

C. Determining Video Rates

As discussed in Section II, the search space of Problem OPT-CLR is the Cartesian product of the set of feasible paths and the set of feasible video rates. The optimal values of these parameters jointly produce the lowest total distortion. In the GA-based approach, the optimal session rates are determined when evaluating the individuals. More specifically, we first use the procedure described in Fig. 1 to evolve a population, assuming each session uses its minimum rate $\underline{R}_\sigma, \sigma \in \mathcal{E}$. Then, during each iteration, we determine the corresponding optimal rates for each individual and use this rate to compute its fitness value.

Because an individual is a solution with a *given* set of feasible paths $\{P_\sigma\}_{\sigma \in \mathcal{E}}$, the problem of finding optimal video rates for a set of given paths (OPT-Rate) can be expressed as follows.

OPT-Rate

$$\text{Minimize:} \quad D(x_k) = \sum_{\sigma \in \mathcal{E}} D_\sigma^e \tag{19}$$

$$\text{subject to:} \quad \underline{R}_\sigma \le R_\sigma \le \bar{R}_\sigma, \quad \forall \sigma \in \mathcal{E} \tag{20}$$

$$\rho_{ij} \le 1 - \epsilon, \quad \forall \{i, j\} \in \mathcal{L}. \tag{21}$$

Note that we do not need to solve OPT-Rate for streaming stored video because the video rates are fixed for such applications. OPT-Rate is a nonlinear optimization problem with nonlinear constraints. It can be efficiently solved using an iterative procedure based on the *sequential quadratic programming (SQP) method*, which is considered one of the most effective methods for solving nonlinear programming problems due to its superlinear convergence performance [24].

IV. Greedy Algorithm for Initial Solutions

In this section, we present an efficient greedy algorithm for Problem OPT-CLR. The algorithm is based on the observation of the key characteristics of the video distortion model. It computes low loss and low congestion paths for the video sessions using an empirical compound routing metric.

Before describing the greedy algorithm, we first examine the total end-to-end distortion D_σ^e of a session $\sigma \in \mathcal{S}$ [see (10) and (11)]. The distortion due to encoder, D_σ^{enc}, is a monotonically *decreasing* function of video rate R_σ. The distortion due to congestion, D_σ^{cg}, however, is a monotonically *increasing* function of R_σ, as well as the rates of all other sessions R_i, that share one or more links with session σ. Both terms are constrained by the stability constraint (15) and are thus determined by the available bandwidths of the path links. The third term D_σ^{loss}, the distortion

Algorithm GREEDY

1. Set the cost of each link $\{i,j\}$ to $c_{ij}(1-p_{ij})$, $\forall\{i,j\}\in\mathcal{L}$;
2. For every video session, $\sigma\in\mathcal{S}$;
3. Use the algorithm in [25] and use the costs defined in 1 to find the path having the maximum end-to-end cost. Let this path be $\mathcal{P}_\sigma$;
4. Decrease the bandwidth of every link on $\mathcal{P}_\sigma$ by $\underline{R}_\sigma$, i.e., setting the link costs as $(c_{ij}-\underline{R}_\sigma)\cdot(1-p_{ij})$, $\forall\{i,j\}\in\mathcal{P}_\sigma$;
5. Go to: 2.
6. After the paths for all sessions are found, solve problem OPT-Rate to determine the rates for the sessions.

Fig. 5. Greedy algorithm for computing initial solutions.

caused by lost packets, is an increasing function of the link loss probabilities. To minimize the video distortion for session σ, we need to find paths having the highest end-to-end bandwidth, the minimal congestion, and the lowest end-to-end loss rate.

We also observe that the D_σ^{enc} curve is concave: when R_σ increases beyond a certain threshold, further increasing R_σ will only cause marginal reduction in D_σ^{enc}. For example, for an H.263 coder with typical settings (e.g., intra rate $1/15$ and frame rate 30 ft/s) using the quarter common intermediate format (QCIF) formatted "Foreman" sequence, there is a decrease of about 100 in D_σ^{enc} when R_σ increases from 40 to 150 kb/s. When R_σ further increases from 150 kb/s to ∞, the corresponding total reduction in D_σ^{enc} is only about 20. A high rate will cause congestion in the bottleneck link, resulting in a much larger increase in D_σ^{cg}. For practical R_σ values, reducing congestion conditions in the network would be more effective than increasing video rates in improving the overall video quality.

In Fig. 5, we describe a greedy heuristic (called GREEDY) for Problem OPT-CLR. In GREEDY, an empirical compound link cost $c_{ij}(1-p_{ij})$, which we call the *effective available bandwidth*, is used. For a given path, its end-to-end effective available bandwidth is the minimum among those of its links. By computing the path with the maximum effective available bandwidth, GREEDY finds the current "widest" path for a session, which has the potential of supporting higher video rates and having low loss rates. Because both link capacity and loss probability are considered in the compound link cost, GREEDY may produce near-optimal solutions to Problem OPT-CLR, as is shown in Section V. For each session, the maximum effective-available-bandwidth path could be computed using the algorithm presented in [25], with a time complexity of $O(|\mathcal{L}|\cdot\log^*|\mathcal{N}|)$, where $\log^* n$ is the *iterated logarithm function*. The overall time complexity of GREEDY is $O(|\mathcal{S}|\cdot|\mathcal{L}|\cdot\log^*|\mathcal{N}|)$.

For the set of computed paths $x_k=\{P_\sigma\}_{\sigma\in\mathcal{S}}$ (which potentially has the minimal congestion and path loss), we solve the nonlinear optimization problem OPT-Rate, which further reduces the overall video distortion by finding the near-optimal video rates for the sessions. It can be easily verified that the path set found by GREEDY is realizable (i.e., it satisfies all the constraints of Problem OPT-CLR). Therefore, the resulting distortion is an upper bound of the optimal distortion. The computational complexity of GREEDY is much lower than GA. It could be used to compute a good initial solution for GA and thus speed up the GA convergence. In practice, GREEDY can be used to quickly compute a set of near-optimal paths for the video sessions. Then, the GREEDY solution could be included into the GA initial population to be further improved, if possible. When GA terminates, the video sessions could switch to the refined routes for better performance.

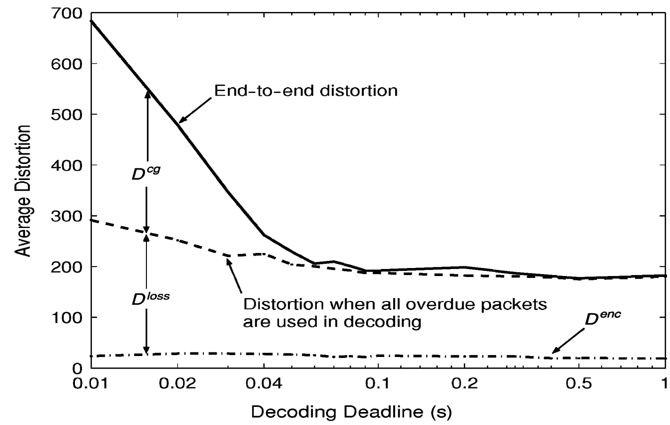


Fig. 6. Distortion versus decoding deadline.

V. Simulation Results

In this section, we present simulation results. In each experiment, an *ad hoc* network is generated at random within a rectangular region. Each video session has a rate between 100 and 400 kb/s. We use an H.263+ codec and the 400-frame, QCIF "Foreman" video sequence. The video is encoded with an intra rate of $1/15$ and a frame rate of 30 frames/s. The corresponding rate-distortion parameters are obtained from [15]. Failure probabilities of the wireless links are uniformly distributed between [1%, 10%]; the bandwidth of a link is uniformly distributed between [100 kb/s, 400 kb/s]. For all results reported in this section, the exponential model (2) is used.

As discussed, there are three key parameters for the proposed GA approach: the *population size*, the *crossover rate* θ, and the *mutation rate* μ. Through simulation studies, we find that the performance of the GA-based routing is quite stable for a wide range of parameter settings. For the results reported in this section, the population size is seven, $\theta=0.4$, and $\mu=0.2$. The greedy algorithm presented in Section IV is used to generate an initial solution, where as the remaining initial solutions are generated using the random constructive method discussed in Section III-B.

A. Dissecting End-to-End Distortion

In Fig. 6, we plot the average distortion versus decoding deadline for a 15-node network with four concurrent video sessions. We set the same decoding deadline value for all sessions. It can be observed that the average distortion is a decreasing function of decoding deadline. For small decoding deadline values, most of the video packets are overdue, resulting in high distortion. As decoding deadline increases, the average distortion quickly decreases because more and more packets are now received in time, contributing to an improved video quality. As the decoding deadline further increases, the real-time application essentially reduces to an elastic data application, where any received packet

TABLE II
AVERAGE DISTORTION VALUES FOUND BY THE ALGORITHMS

Network Size	Topo. 1 9	Topo. 2 9	Topo. 3 11	Topo. 4 11	Topo. 5 11
GREEDY	230.20	81.77	88.29	115.32	77.55
ES	213.94	81.77	67.01	115.32	67.20
GA	214.31	81.77	67.76	115.32	67.63
Norm. Diff.	0.17%	0	1.11%	0	0.62%
Std. Dev.	0.55	0	6.93	0	3.08

is useful for improving video quality. The same trend exists for all cases we simulated, although the specific "knee-point" depends on the network and video parameters.

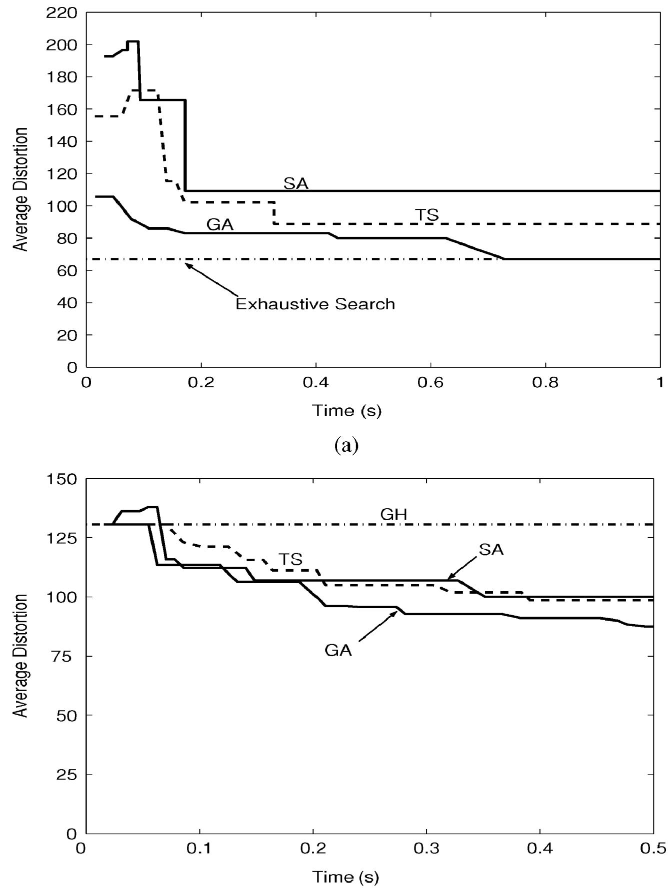
As discussed, the end-to-end distortion of a video session consists of three components: encoding distortion $D^{\rm enc}$, distortion due to packet losses $D^{\rm loss}$, and distortion due to congestion $D^{\rm cg}$. In Fig. 6, the dash dotted (lowest) curve is for $D^{\rm enc}$. From the empirical distortion model (11), $D^{\rm enc}$ is a function of the target encoding bit rate, which is determined by the end-to-end bandwidth and the congestion condition of the path (i.e., the rate is computed by solving Problem OPT-Rate). It is relatively constant for various decoding deadlines in this experiment.

In Fig. 6, the dashed curve is the average distortion computed using both packets that are received in time and packets that are overdue. The difference between the two lower curves corresponds to $D^{\rm loss}$, which is determined by the end-to-end loss rate. Note that $D^{\rm loss}$ decreases gradually as decoding deadline increases. This is because for very tight delay constraints, the GA-based routing may give more preference to delay over loss to meet the delay constraint. That is, GA-based routing may choose a path having a lower end-to-end delay, even though the path may have a higher end-to-end loss. As the delay constraint gets relaxed, GA is allowed to choose paths having a lower end-to-end loss rate while still satisfying the delay constraint. The difference between end-to-end distortion and the dashed curve is $D^{\rm cg}$. When congestion occurs, packets experience large delays and many of them arrive beyond the decoding deadline, causing high distortions. As decoding deadline increases, the end-to end distortion curve converges to the dashed curve, which indicates that congestion has little impact on video distortion when the delay constraint is relaxed.

B. Performance Bounds

One interesting question regarding the GA-based routing is how close its solutions are to the optimal solution. For small networks, it is possible to find the optimal solution in a reasonable amount of time by using an exhaustive search (ES), which, however, is infeasible when network size becomes large. In Table II, we compare the GA solutions to the global optimal solutions computed by exhaustive search, for small networks with three concurrent video sessions. The decoding deadline Δ_σ is set to 0.1 s for all video sessions. GA runs for about 50 iterations in each simulation, and each GA distortion value is the average of 30 runs. In Table II, the normalized difference is computed as $|$GA-ES$|/$ES.

We find that GA performs consistently well in comparison to the optimal solution. In every case, GA either finds the exact global optimal or finds a near-optimal solution with a negligible normalized difference. Moreover, for the distortion values obtained by 30 executions using the same network, the standard deviation is very small for all cases examined, indicating that the GA performance is very stable. Finally, the GA computation time (a few hundred milliseconds, on a Pentium4 2.4-GHz computer with 512 MB memory) is only a tiny fraction of the time required to perform the exhaustive search (2.5–9.1 h).



(a)

(b)

Fig. 7. GA versus trajectory methods. (a) 11-node network, four video sessions (b) 50-node network, ten video sessions.

We also present the GREEDY results in Table II. We find that the GREEDY solutions are also quite competitive. In topologies 2 and 4, GREEDY finds the exact global optima; in the remaining four cases, the GREEDY distortions are close to the the global optima.

C. Comparison With Trajectory Methods

For comparisons, we implement two representative trajectory metaheuristic methods, namely, SA [22] and TS [23]. For best performance, we set the initial *temperature* T in SA to 1 and the temperature decaying ratio γ to 0.5. The *tabu tenure* for the TS implementation is chosen to be five units.

In Fig. 7, we plot the evolutions of the total distortions obtained by GA, SA, and TS. Fig. 7(a) is obtained using an 11-node network with three concurrent video sessions (for which the global optimum could be found by ES) where as Fig. 7(b) is obtained for a 50-node network with ten concurrent sessions. Simulation time for all three algorithms is 1 s. The decoding deadline is set to 0.5 s for the 50-node network simulations, and 0.1 s for the 11-node network simulations.

From Fig. 7, we observe that GA converges to the global optimal very quickly, where as both SA and TS are trapped at local optima (i.e., no improvement after a large number of iterations). This is due to the fact that GA could explore the fitness landscape in parallel by evolving a population of solutions, where as trajectory methods only maintain a single solution and have a higher tendency of being trapped at a local minimum (despite the fact that they all incorporate explicit strategies to avoid such events).

D. Comparison With Network Centric Routing

We compare GA-based routing with traditional network centric routing in solving Problem OPT-CLR. We implement a shortest path (SP) routing algorithm using hop count as routing metric, and a disjoint shortest path (DSP) routing algorithm using loss rate as routing metric [26], both based on Dijkstra's algorithm. In DSP, the link cost is set to $\log(1/(1-p_{ij})), \{i,j\} \in \mathcal{L}$. As a result, the minimum cost path has the highest end-to-end success probability. In both network centric algorithms, we first find the optimal set of paths using the minimum video rates $\{\underline{R}_\sigma\}_{\sigma \in \mathcal{S}}$ and then solve Problem OPT-Rate over the path set to compute the corresponding optimal video rates. To meet the link stability condition in SP, each time when a path is found, we subtract the minimum rate of the corresponding video session from the capacity of each link along this path, where as the next path is found in the "reduced" graph. The computation time of GA is between 500 to 900 ms, whereas the computation time for SP or DSP ranges from tens of milliseconds to about 200 ms.

1) Overall Performance: Fig. 8(a) plots the average distortions found by the three algorithms (i.e., GA, SP, and DSP) for various decoding deadlines. The network consists of 50 nodes with ten video sessions. We find that for very small decoding deadlines, the delay requirement is so stringent that all three schemes yield high distortion. However, for very large decoding deadlines, the delay requirement is so loose that all three schemes can achieve a low total distortion, as long as the stability condition is satisfied. The more interesting region, however, lies between these two extremes, where a well-designed routing scheme can achieve a better performance by finding optimal routes for the video sessions. Within this region, GA outperforms SP and DSP by a significant margin. In Fig. 8(a), the GA average distortion quickly decreases as decoding deadline increases, where as the SP and DSP average distortions are persistently high for small and medium decoding deadlines (indicating that most of the video packets are overdue in these cases). For example, when $\Delta = 0.2$ s, the difference between the average distortions achieved by GA and DSP is 683.8, which translates to a 9.03 dB reduction in PSNR. Similarly, GA achieves a 449.6 reduction in total distortion over SP, which is a 7.49 dB improvement in average PSNR. These improvements are significant, because generally a half decible difference in PSNR is noticeable [27].

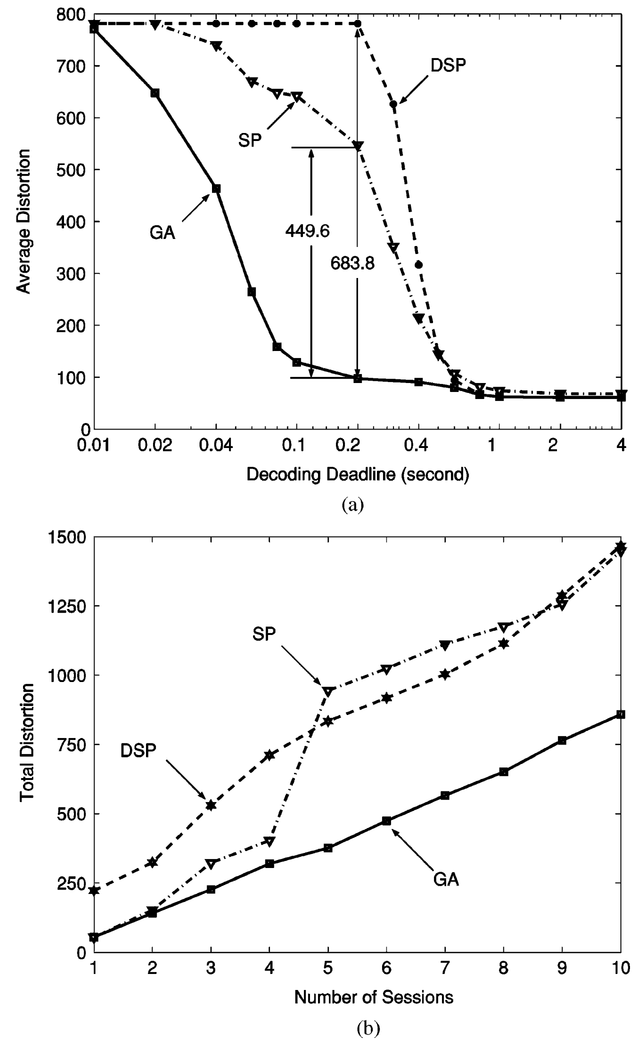


Fig. 8. GA versus SP and DSP approaches. (a) Average end-to-end distortion versus decoding deadline. (b) Total distortion versus number of video sessions.

In Fig. 8(b), we examine the impact of video traffic load on the routing performance. We compare the total distortions found by GA, SP, and DSP, while increasing the number of video sessions in the 50-node network. The decoding deadline is 0.5 s for all video sessions. As expected, both SP and DSP produce higher total distortions than GA, due to the fact that they only use network layer metrics in routing. More specifically, SP does not consider the interaction of the video sessions. Although it computes the shortest path for each session, different sessions may share bottleneck links, resulting in congestion and high packet overdue rates. However, DSP goes to the other extreme by not allowing the sharing of any links, even when a link has abundant bandwidth and a low loss rate. As a result, some "bad" links (i.e., low capacity or high failure probability links) or paths having a large number of hops will be used to satisfy the disjointedness requirement, resulting in an increased total distortion. Another interesting observation from Fig. 8(b) is that the total distortion obtained by GA increases linearly with the number of sessions, which implies that the average distortion for each session is relatively constant, even though the video traffic load has increased nearly tenfold.

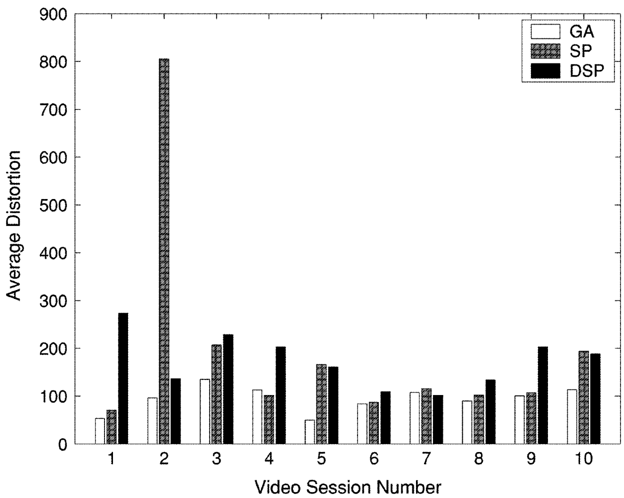


Fig. 9. Average distortion values for each video session in a ten-session, 50-node network obtained by different algorithms.

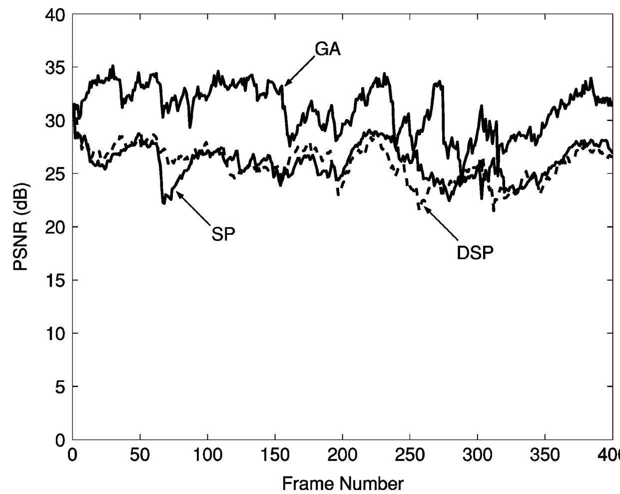


Fig. 10. PSNRs of decoded frames for fifth video session.

2) Performance of Individual Sessions and Frames: So far, we have investigated the impact of optimal routing on total video distortion. Now, we examine the quality of individual video sessions. Specifically, we transmit encoded video on those paths found by GA, SP, and DSP, respectively, and compute PSNRs for decoded video frames that are possibly corrupted due to transmission errors and congestion.

The distortion values for the individual sessions obtained by the three algorithms are plotted in Fig. 9 for a 50-node network with ten video sessions. We find that for most of the sessions (except for sessions 4 and 7), GA achieves a much lower distortion than the two networkcentric algorithms. Although for sessions 4 and 7 the GA distortion is higher than that of SP or DSP, the difference is negligible in both cases. The session distortions of SP and DSP are highly diverse, whereas the GA sessions have relatively even distortions, despite the fact that only the total distortion is minimized. The total distortion achieved by GA is 943.2, which is much lower than those achieved by SP (1956.0) and DSP (1738.4).

The PSNRs of decoded frames for session 5 are plotted in Fig. 10. The frames sent on the GA paths have much higher PSNR values than those sent on the SP or DSP paths. The average session 5 PSNR (over the 400 frames) achieved by the GA-based routing is 31.16 dB, where as the average PSNRs obtained by SP and DSP are 25.93 and 26.06 dB, respectively. Such significant gains (over 5 dB in both cases) are due to the fact that the application layer video quality (rather than network layer metrics) is explicitly optimized and the routing for multiple sessions is jointly optimized.

VI. Distributed Implementation

Existing *ad hoc* protocols can be roughly categorized into *proactive*, whereby a consistent and up-to-date view of the network is always maintained, and *reactive*, whereby route discovery is performed on-demand. We find that the proposed GA-based routing is highly suitable for the proactive routing paradigm. This choice is also motivated by the fact that a routing decision must be made quickly for a new request to reduce response time.

The core of a distributed implementation is to build and maintain network topology and link statistic databases at each node. To this end, we find that several components in the class of link state-based *ad hoc* routing protocols, such as the *optimized link state routing* protocol (OLSR) [28], are suitable for this purpose.

Fig. 11 depicts an implementation architecture of the routing protocol at an *ad hoc* node. With this architecture, each node should detect its one-hop neighbor nodes (e.g., by periodic HELLO messages). Furthermore, each node measures the quality of its links, such as bandwidth, loss rate, and delay. Several effective algorithms (e.g., those proposed in [14]) could be used for this purpose. As in other link state routing protocols, LSAs are periodically broadcast to distribute network topology information and link statistics. A new type of LSA could be used to distribute video session information (e.g., source and destination nodes, and other session specific parameters). To minimize the flooding of control messages, we could use the *multipoint relay* (MPR) technique (RFC 3626) [28], which has been proved to be quite effective for this purpose [28]. Link state and topology information learned from received LSAs are pooled in a *link state database*.

Finally, the GA-based concurrent routing module is built on top of the link state database to compute near-optimal routes for video sessions. Note that the GA-based routing uses several video/codec specific parameters for computing distortion [see (11)]. Usually, the codec-related parameters are readily available, video specific parameters could be measured in advance, or some average values could be used based on the type of the video to be transmitted (video conferencing, sports, etc.).

VII. Related Work

As discussed, the problem addressed in this paper differs from network-layer QoS routing for *ad hoc* networks [11]–[14]. Most of these prior efforts did not explicitly formulate the

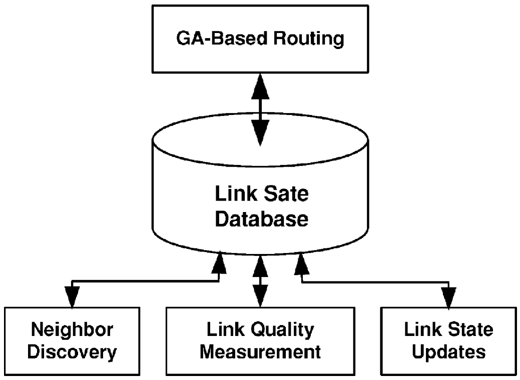


Fig. 11. Distributed implementation architecture at an *ad hoc* node.

objective function with an application layer metric via a cross-layer approach.

Several path/server selection schemes have been developed for video streaming on the Internet. Specifically, in [29], the authors present three heuristics on selecting multiple description (MD) video servers in a content delivery network (CDN). Although shown to be quite effective for CDNs, these algorithms only select servers based on some performance metrics. It is not clear how to determine the paths to the servers for minimizing video distortion. In a recent work [30], Begen *et al.* study the problem of path selection for MD video streaming in service overlay networks. The optimal routing problem is solved via ES, which could have an exponential complexity. Subsequently, the authors proposed an improvement algorithm in [31] by taking advantage of the special structure of the underlying network. This improvement algorithm may not be feasible for *ad hoc* networks that are infrastructureless and have dynamic topologies. Finally, it is worth noting that the interaction of concurrent video flows is not considered in these prior works.

There exist several prior efforts on applying GA to address network layer problems (e.g., shortest path routing [32], [33] and multicast QoS routing [34]). The research presented in this paper builds on these earlier efforts and explores GA's potential to address the more complex cross-layer, video centric optimization problem. The problem investigated in this paper is substantially more difficult because it exploit the design and optimization space across the layers.

VIII. Conclusion

In this paper, we studied the problem of how to optimally support multiple concurrent video communication sessions in an *ad hoc* network. We formulated a networkwide optimal routing problem that minimizes the total distortion of all video sessions. We modeled the end-to-end video distortion as a function of routing layer behavior. Our formulation seamlessly integrates the interaction of competing video flows and network layer link metrics, which allows for computing of optimal routes and for determining the optimal rates for the video sessions. We developed a highly effective solution procedure based on the GA framework for computing near optimal solutions. Through extensive numerical results, we demonstrated that the GA-based approach has clear advantages over other approaches.

References

[1] N. Gogate, D. Chung, S. S. Panwar, and Y. Wang, "Supporting image/video applications in a multihop radio environment using route diversity and multiple description coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 9, pp. 777–792, Sep. 2002.

[2] 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. Sel. Areas Commun.*, vol. 21, no. 10, pp. 1721–1737, Dec. 2003.

[3] S. Mao, S. Lin, Y. Wang, S. S. Panwar, and Y. Li, "Multipath video transport over wireless ad hoc networks," *IEEE Wireless Commun. Mag.*, vol. 12, no. 4, pp. 2–9, Aug. 2005.

[4] S. Mao, X. Cheng, Y. T. Hou, and H. D. Sherali, "Multiple description video multicast in wireless ad hoc networks," *Mobile Netw. Appl. J. (MONET)*, vol. 11, no. 1, Jan. 2006.

[5] S. Mao, D. Bushmitch, S. Narayanan, and S. S. Panwar, "MRTP: A multiflow real-time transport protocol for ad hoc networks," in *IEEE Trans. Multimedia*, to appear.

[6] H. Gharavi and K. Ban, "Dynamic adjustment packet control for video communications over ad-hoc networks," in *Proc. IEEE ICC 2004*, Paris, France, 2004, pp. 3086–3090.

[7] E. Setton, X. Zhu, and B. Girod, "Congestion based multipath routing of multimedia data over ad hoc networks," in *Proc. ICME 2004*, Taipei, Taiwan, R.O.C., 2004, pp. 1619–1622.

[8] T. Yoo, E. Setton, X. Zhu, A. Goldsmith, and B. Girod, "Cross-layer for video streaming over wireless *ad hoc* networks," in *Proc. MMSP 2004*, Siena, Italy, Oct. 2004, pp. 179–182.

[9] Y.-C. Hu and D. B. Johnson, "Design and demonstration of live audio and video over multihop wireless ad hoc networks," in *Proc. IEEE MILCOM 2002*, Anaheim, CA, Oct. 2002, pp. 7–10.

[10] S. Mao, S. Lin, S. S. Panwar, and Y. Wang, "An ad hoc multipath video transport testbed," in *Proc. IEEE VTC 2003-Fall*, Orlando, FL, pp. 2961–2965.

[11] S. Chen and K. Nahrstedt, "Distributed quality-of-service routing in ad-hoc networks," *IEEE J. Select. Areas Commun.*, vol. 17, no. 8, pp. 1488–1505, Aug. 1999.

[12] C. R. Lin and J.-S. Liu, "QoS routing in ad hoc wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 17, no. 8, pp. 1426–1438, Aug. 1999.

[13] C. E. Perkins, E. M. Royer, and S. R. Das, "Quality of Service in *ad hoc* on-demand distance vector routing," IETF Internet Draft, Jul. 2000.

[14] H. Badis and K. Al Agha, "QOLSR, QoS routing for ad hoc wireless networks using OLSR," *Eur. Trans. Telecom.*, vol. 15, no. 4, 2005.

[15] K. Stulmuller, N. Farberand, M. Link, and B. Girod, "Analysis of video transmission over lossy channels," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 6, pp. 1012–1032, Jun. 2000.

[16] *Handbook of Evolutionary Computation*, T. Back, D. Fogel, and Z. Michalewicz, Eds., New York: Oxford Univ. Press, 1997.

[17] I. Norros, "On the use of fractional brownian motion in the theory of connectionless networks," *IEEE J. Sel. Areas Commun.*, vol. 13, no. 6, pp. 953–962, Aug. 1995.

[18] H. Chernoff, "A measure of asymptotic efficiency for tests of a hypothesis based on the sum of observations," *Ann. Math. Statist.*, vol. 23, pp. 493–507, 1952.

[19] A. Elwalid, D. Heyman, T. V. Lakshman, D. Mitra, and A. Weiss, "Fundamental bounds and approximations for atm multiplexers with applications to video teleconferencing," *IEEE J. Sel. Areas Commun.*, vol. 13, no. 6, pp. 953–962, Aug. 1995.

[20] C. Goldie and C. Klüppelberg, *Subexponential Distributions*. Basel, Switzerland: Birkhäuser, 1998, pp. 435–459.

[21] Z. Wang and J. Crowcroft, "Quality-of-service routing for supporting multimedia applications," *IEEE J. Sel. Areas Commun.*, vol. 17, no. 8, pp. 1488–1505, Aug. 1999.

[22] E. Aarts and J. Korst, *Simulated Annealing and Boltzman Machines*. New York: Wiley, 1989.

[23] F. Glover and M. Laguna, *Tabu Search*. Boston, MA: Kluwer-Academic, 1997.

[24] E. R. Panier and A. L. Tits, "A superlinearly convergent feasible method for the solution of inequality constrained optimization problems," *SIAM J. Control Optim.*, vol. 25, no. 4, pp. 934–950, Jul. 1987.

[25] N. Malpani and J. Chen, "A note on practical construction of maximum bandwidth paths," *Inf. Process. Lett.*, vol. 83, pp. 175–180, Aug. 2002.
[26] T. H. Cormen, C. E. Leiserson, and R. L. Rivest, *Introduction to Algorithms*. Cambridge, MA: MIT Press, 1990.
[27] T. Wiegand, N. Frber, K. Stuhlmller, and B. Girod, "Error-resilient video transmission using long-term memory motion-compensated prediction," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 6, pp. 1050–1062, Jun. 2000.
[28] T. Clausen and P. Jacquet, "Optimized Link State Routing Protocol," IETF RFC 3626, Oct. 2003.
[29] J. G. Apostolopoulos, T. Wong, W. Tan, and S. Wee, "On multiple description streaming in content delivery networks," in *Proc. IEEE INFOCOM*, New York, Jun. 2002, pp. 1736–1745.
[30] A. C. Begen, Y. Altunbasak, and O. Ergun, "Multi-path selection for multiple description encoded video streaming," *EURASIP Signal Process. Image Commun.*, vol. 20, no. 1, pp. 39–60, Jan. 2005.
[31] A. C. Begen, Y. Altunbasak, and O. Ergun, "Fast heuristics for multi-path selection for multiple description encoded video streaming," in *Proc. IEEE ICME*, Baltimore, MD, Jul. 2003, pp. 517–520.
[32] C. W. Ahn and R. S. Ramakrishna, "A genetic algorithm for shortest path routing problem and the sizing of populations," *IEEE Trans. Evol. Comput.*, vol. 6, no. 6, pp. 566–579, Dec. 2002.
[33] M. Gen and R. Cheng, *Genetic Algorithms & Engineering Optimization*. New York: Wiley, 2000.
[34] N. Banerjee and S. K. Das, "Fast determination of QoS-based multicast routes in wireless networks using genetic algorithm," in *Proc. IEEE ICC*, Helsinki, Finland, Jun. 2001, pp. 2588–2596.

Shiwen Mao (S'99–M'04) received the B.S. and M.S. degrees from Tsinghua University, Beijing, China, in 1994 and 1997, respectively, both in electrical engineering. He received the M.S. degree in system engineering and the Ph.D. degree in electrical and computer engineering from Polytechnic University, Brooklyn, NY, in 2000 and 2004, respectively.

From 1997 to 1998, he was a Research Member at the IBM China Research Lab, Beijing. During the summer of 2001, he was a Research Intern at Avaya Labs-Research, Holmdel, NJ. Currently, he is a Research Scientist at the Bradley Department of Electrical and Computer Engineering, Virginia Polytechnic Institute and State University, Blackburg. He is the coauthor of a textbook entitled *TCP/IP Essentials: A Lab-Based Approach* (Cambridge University Press, 2004). His research interests multimedita and wireless networking.

Dr. Mao is a corecipient of the 2004 IEEE Communications Society Leonard G. Abraham Prize in the Field of Communcations Systems.

Sastry Kompella (S'04) received the B.E. degree in electronics and communcation engineering from Andhra University. Visakhapatnam, Andhra Pradesh, India, in 1996 and the M.S. degree in electrical engineering from Texas Tech University, Lubbock, in 1998. He is currently a Ph.D. student in the Bradley Department of Electrical and Computer Engineering, Virginia Polytechnic Institute and State University, Blacksburg.

He has worked as a Project Applications Engineer from 1998 to 2003 in Austin, TX. His research focuses on multimedia delivery over wireless networks.

Mr. Kompella received the TTU Competitive Scholarship in 1996 and 1997 and the Outstanding Bachelors Student Award (Gold Medal) in 1996.

Y. Thomas Hou (S'91–M'98–SM'04) received the B.E. degree from the City College of New York, in 1991, the M.S. degree from Columbia University, New York, in 1993, and the Ph.D. degree from Polytechnic University, Brooklyn, NY, in 1998 all in electrical engineering.

From 1997 to 2002, he was a Research Scientist and Project Leader at Fujitsu Laboratories of America, IP Networking Research Department, Sunnyvale, CA. Since the fall of 2002, he has been an Assistant Professor with the Bradley Department of Electrical and Computer Engineering, Virginia Polytechnic Institute and State University, Blacksburg. His research interests are in algorithmic design and optimization for network systems. His current research focuses on wireless sensor networks and multimedia over wireless *ad hoc* networks. In the 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 extensively in these areas.

Dr. Hou is a corecipient of the 2004 IEEE Communications Society Multimedia Communications Best Paper Award, the 2002 IEEE International Conference on Network Protocols (ICNP) Best Paper Award, and the 2001 IEEE Transactions on Circuits and Systemfor Video Technology (CSVT) Best Paper Award.

Hanif D. Sherali is a W. Thomas Rice Endowed Chaired Professor of Engineering in the Industrial and Systems Engineering Department, Virginia Polytechnic Institute and State University, Blacksburg. His area of research interest is in discrete and continuous optimization, with applications to location, transportation, and engineering design problems. He has published almost 200 papers in operations research journals, coauthored four books in this area, and serves on the editorial board of eight journals.

Mr. Sherali is a member of the National Academy of Engineering.

Scott F. Midkiff (S'82–M'85–SM'92) received the B.S.E. and Ph.D. degrees from Duke University, Durham, NC, and the M.S. degree from Standford University, Standford, CA, all in electrical engineering.

He worked at Bell Laboratories and held a visiting position at Carnegie Mellon University, Pittsburgh, PA. In 1986, he joined the Bradley Department of Electrical and Computer Engineering, Virginia Polytechnic Institute and State University, Blacksburg, where he is now a Professor. His research interests include system issues in wireless and *ad hoc* networks, network services for pervasive computing, and performance modeling of mobile *ad hoc* networks.