Routing for Multiple Concurrent Video Sessions in Wireless Ad Hoc Networks

Shiwen Mao† Sastry Kompella† Y. Thomas Hou† Hanif D. Sherali‡ Scott F. Midkiff†
†The Bradley Department of Electrical and Computer Engineering, Virginia Tech, Blacksburg, VA 24061
‡The Grado Department of Industrial and Systems Engineering, Virginia Tech, Blacksburg, VA 24061
{smao, sastryk, thou, hanifs, midkiff}@vt.edu

***Abstract*— Real-time multimedia communication is an important service that should be supported in wireless ad hoc networks. In this paper, we consider the problem of how to optimally support multiple concurrent video communication sessions in an ad hoc network. Our problem formulation follows an *application-centric cross-layer approach* with the objective of minimizing the average distortion for all video sessions via finding optimal paths for each session. Since this network-wide optimization problem is shown to be NP-complete, we pursue to develop competitive heuristic algorithms to address this problem. We find that *Genetic Algorithms* (GA) are eminently efficient in solving such cross-layer problems with complex objective functions and constraints. We describe a detailed solution procedure based on the GA approach and use numerical results to demonstrate its superior performance over other conventional approaches. Our efforts in this work provide an important methodology for addressing cross-layer network-wide optimal routing problems for video applications.**

I. Introduction

With the recent 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, in addition to simple data communications. Unlike wired networks (e.g., the Internet) or infrastructure-based wireless networks, ad hoc networks typically are deployed under harsh or denied areas and exhibit frequent *node or link failures* in addition to network congestion. As a result, such a network environment poses a much more difficult problem for real-time multimedia communications (e.g., video).

In this paper, we consider the important problem of supporting multiple concurrent video sessions in wireless ad hoc networks. This problem is of importance since it captures the scenario that there are typically more than one real-time multimedia communication sessions sustained by an ad hoc network and these sessions may share the same network resources (link bandwidth, buffer) and might interact with each other. *The first contribution of this paper is that we formulate 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).* In particular, our constraints at the network layer address not only packet losses due to frequent node/link failures, which are unique to ad hoc networks, but also traditional network problems such as delays due to congestion. In other words, our application-centric problem formulation seamlessly unifies video distortion with packet loss (due to node/link failures) and delay via routing for each session.

The formulated model for this problem exhibits a highly complex objective function and constraints, which renders this problem substantially more difficult than traditional network-centric (single network layer) QoS routing problems. We find that the problem is NP-complete. *Accordingly, the second contribution of this paper is that we develop a highly competitive solution method based on Genetic Algorithms (GAs)* [1]. 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. Furthermore, 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 [1]. We find that the complex network-wide optimization problem provides the perfect setting for a GA-based method. The complexity in the objective function does not create much difficulty for this type of a procedure other than some algebraic calculations. The complexity due to the interaction among concurrent sessions can also be handled rather naturally by GAs since they are intrinsically parallel, and concurrent multi-session routing and interactions increase computational effort only linearly as compared with the single session routing [2]. We demonstrate the superior performance of the GA-based approach over traditional approaches using extensive numerical results.

The remainder of this paper is organized as follows. In Section II, we formulate the optimal routing problem for multiple concurrent video sessions via a cross-layer approach. In Section III, we propose a solution procedure based on GAs. In Section IV, we use extensive numerical results to demonstrate the efficacy of the GA-based approach and its superior performance over other approaches. We discuss related work in Section V, and Section VI concludes the paper.

II. Problem Formulation

In this section, we formulate the application-centric routing problem for multiple concurrent video sessions in a wireless ad hoc network. To model a wireless ad hoc network as an associated graph, we assume that a wireless link exists between nodes i and j if nodes i and j can communicate with each other. For example, a link may exist if nodes i and j are within reachable distance of their radio transmitter.

Consequently, the wireless 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:

- c_{ij}: The capacity, or bandwidth of link $\{i, j\}$;
- p_{ij}: The mean packet loss probability of link $\{i, j\}$, due to transmission errors or link failures.

Other characterizations of a wireless link in ad hoc networks can be incorporated into our model as well.

In this network, we assume there exists 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 as $\underline{R}_\sigma \leq R_\sigma \leq \overline{R}_\sigma, \forall \sigma \in \mathcal{E}$. The lower and upper bounds of R_σ are determined by the specific video encoder and video sequence used at the source node s_σ. The decoding deadline for session σ packets is Δ_σ. The objective is to find optimal paths for the video sessions such that the overall video distortion of all the sessions is minimized.

A. End-to-End Delay and Loss Rate

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

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

Using such index variables, the choice of path $\mathcal{P}_\sigma$ can be represented by a routing vector $\mathbf{X}_\sigma = \{x_{ij}^{\sigma}\}_{\{i,j\} \in \mathcal{L}}$ having $|\mathcal{L}|$ elements, each of which corresponds to a link and has a binary value.

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 $\{i, j\}$, for each $\{i, j\} \in \mathcal{P}_\sigma$. Then, the average aggregate traffic load on any link $\{i, j\} \in \mathcal{L}$ is:

$$\lambda_{ij} = \sum_{\sigma \in \mathcal{E}} x_{ij}^{\sigma} \prod_{\{m,n\} \in \bar{\mathcal{P}}_\sigma^{ij}} (1 - p_{mn}) R_\sigma \ . \tag{2}$$

In other words, 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 $\rho_{ij} = \lambda_{ij}/c_{ij}$, $\forall \{i, j\} \in \mathcal{L}$. For stability, a feasible set of routes $\{\mathcal{P}_\sigma\}_{\sigma \in \mathcal{E}}$ should satisfy $\rho_{ij} < 1, \forall \{i, j\} \in \mathcal{L}$.

2) Delay on a Link : Since real-time video traffic typically has stringent delay requirements, it is necessary to consider link delays due to congestion. We model each link $\{i, j\}$ as a general queuing system [3] with an average input rate λ_{ij} (defined in (2)) and a service capacity c_{ij}. Let the probability density function of queueing delay t_{ij} on link $\{i, j\}$ be $f_{ij}(y)$. We assume that all of the 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 can model the queueing delay via an exponential distribution [3], i.e.,

$$f_{ij}(y) = \alpha_{ij} e^{-\alpha_{ij} y}, \quad \text{for } y \geq 0, \tag{3}$$

where $\alpha_{ij} \stackrel{def}{=} (c_{ij} - \lambda_{ij})$. On the other hand, for a variable bit rate (VBR) video that exhibits long-range dependent (LRD) characteristics, we can model the link as a fractional Brownian motion (fBm) queueing system, where t_{ij} has a heavy-tailed Weibull distribution [4]. It is important to point out that the problem formulation and, more importantly, the proposed GA-based solution approach do not depend on a specific traffic behavior, queueing model, and video distortion model.

3) End-to-End Delay : Assuming that the delays on the links are independent, the end-to-end delay of session σ, denoted by T_σ, $\forall \sigma \in \mathcal{E}$, is the sum of the queueing delay on each link along path $\mathcal{P}_\sigma$, i.e.,

$$T_\sigma = \sum_{\{i,j\} \in \mathcal{L}} x_{ij}^{\sigma} t_{ij} \ . \tag{4}$$

We can then apply the large deviation approximation to obtain an accurate estimate of the overdue probabilities. In the following, we illustrate such an approximation when link delays are exponentially distributed, using the *Chernoff Bound* [5]. First, the moment generating function of t_{ij} can be derived as: $M_{ij}(s) = \mathrm{E}[e^{s t_{ij}}] = \frac{\alpha_{ij}}{\alpha_{ij} - s}$, for $s < \alpha_{ij}$. Assuming that the link delays are independent, the moment generating function of T_σ is:

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

Define $F_\sigma(s)$ as:

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

Since ${F_\sigma}''(s) < 0$, $\forall s < \min_{\{i,j\} \in \mathcal{P}_\sigma} \{\alpha_{ij}\}$, $F_\sigma(s)$ is a strictly concave function with a unique maximum at s^*. If $\Delta_\sigma > \mathrm{E}(T_\sigma)$ (i.e., the decoding deadline is larger than the average end-to-end delay on the path[1]), we can determine s^* by solving:

$$F_\sigma'(s) = \Delta_\sigma - \sum_{\{i,j\} \in \mathcal{L}} \frac{x_{ij}^{\sigma}}{\alpha_{ij} - x_{ij}^{\sigma} s} = 0. \tag{7}$$

Since $F_\sigma'(\min_{\{i,j\} \in \mathcal{P}_\sigma} \{\alpha_{ij}\}) = -\infty < 0$ and $F_\sigma'(0) = \Delta_\sigma - \mathrm{E}(T_\sigma) > 0$, we have that $0 < s^* < \min_{\{i,j\} \in \mathcal{P}_\sigma} \{\alpha_{ij}\}$. From the Chernoff Bound, the distribution of T_σ can be approximated as [5]:

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

where $\delta(s) = \sqrt{\frac{\partial^2 \log M_\sigma}{\partial s^2}}$.

Note that the moment generating function of a heavy-tailed Weibull random variable does not exist (although all of 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

[1] This assumption is reasonable for practical applications. Otherwise, a large amount of video packets will be overdue, yielding an intolerable received video quality.

example, for an i.i.d. sequence of heavy-tailed Weibull random variables $\{X_1, \cdots, X_n\}$, we have that $\Pr\left[\sum_{k=1}^{n} X_k > x\right] \approx \Pr\left[\max_{\{1\leq k\leq n\}}\{X_k\} > x\right] \approx n\Pr\left[X_1 > x\right]$ [6].

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

$$p_\sigma = 1 - \prod_{\{i,j\}\in\mathcal{L}} (1 - x_{ij}^\sigma p_{ij}), \forall \sigma \in \mathcal{E}. \tag{9}$$

B. Video Rate-Distortion Modeling

For video coding and communications, a rate distortion model describes the relationship between the bit rate and the achieved video distortion. In the following, we introduce an empirical rate-distortion model that links the overdue and loss probabilities derived in the last subsection to video distortion, which is an important application layer video quality measure.

In [7], 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_σ^e consists of the encoding distortion caused by the lossy video coder, D_σ^{enc}, and the distortion due to transmission errors, including the distortion caused by overdue packets (i.e., congestion), D_σ^{cg}, and the distortion caused by lost packets (i.e., link failure), D_σ^{loss}. That is,

$$D_\sigma^e = D_\sigma^{enc} + D_\sigma^{cg} + D_\sigma^{loss}. \tag{10}$$

From [7] and subsection II-A, we have

$$D_\sigma^e = D_0 + \frac{\omega}{R_\sigma - R_0} + \kappa(1-p_\sigma)Pr(T_\sigma > \Delta_\sigma) + \kappa p_\sigma, \tag{11}$$

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. Since the model in (10) takes into account the effects of INTRA coding and spatial loop filtering, it matches simulation results closely [7].

C. The Global Optimal Routing Problem

We can now mathematically formulate the problem of optimal cross-layer routing for multiple concurrent video sessions:

OPT-CLR

$$\textbf{Minimize}: \quad D = \sum_{\sigma\in\mathcal{E}} D_\sigma^e \tag{12}$$

subject to:

$$\underline{R}_\sigma \leq R_\sigma \leq \overline{R}_\sigma, \ \text{ for } \sigma \in \mathcal{E} \tag{13}$$

$$\rho_{ij} \leq 1 - \epsilon, \quad \forall\{i,j\} \in \mathcal{L}, \text{for some } \epsilon > 0 \tag{14}$$

$$\sum_{j:\{i,j\}\in\mathcal{L}} x_{ij}^\sigma - \sum_{k:\{k,i\}\in\mathcal{L}} x_{ki}^\sigma = \begin{cases} 1, & \text{if } i = s_\sigma \\ -1, & \text{if } i = d_\sigma \\ 0, & \text{otherwise} \end{cases}, \ \forall i \in \mathcal{N}, \sigma \in \mathcal{E} \tag{15}$$

$$x_{ij}^\sigma \in \{0,1\}, \quad \forall\{i,j\} \in \mathcal{L}, \sigma \in \mathcal{E}. \tag{16}$$

We now provide an interpretation for the above problem formulation OPT-CLR. The objective function (12) is the sum of the average distortion of all of the concurrent video sessions. Minimizing (12) achieves a better utilization of network resources, as well as the best overall quality for the video sessions. The set of inequalities in (13) gives the range of feasible rates for each video stream, which is determined by the video sequence and encoder parameters. The inequality in (14) is the stability condition, which ensures that the link utilization (i.e., the ratio of the average aggregate traffic load on the link, and the link capacity) is less than 1. The remaining constraints (i.e., (15) and (16)) guarantee that each path $\mathcal{P}_\sigma$ from s_σ to d_σ is a valid path and is loop-free.[2]

In Problem OPT-CLR, there are two sets of tunable variables that form the search (optimization) space of feasible solutions. They are (i) the set of routing vectors: $\{\mathbf{X}_\sigma\}_{\sigma\in\mathcal{E}}$; and (ii) the set of rates of video sessions: $\{R_\sigma\}_{\sigma\in\mathcal{E}}$. The objective function (12) 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 (in order to compute the traffic load on each link), which is only possible when all the paths are completely determined. Wang and Crowcroft [8] 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 time-varying and coupled session delays (rather than constant link delays as in [8]). As a result, we conjecture that problem OPT-CLR is NP-complete and it is futile to pursue exact solutions. Although exact solutions are not obtainable, we find that a GA-based approach is highly suitable to address such a complex optimization problem. We describe this approach in the next section.

III. Genetic Algorithm-Based Routing for Multiple Concurrent Video Sessions

In this section, we present a detailed GA-based routing approach for multiple concurrent video sessions, which appears to produce near-optimal solutions to Problem OPT-CLR (based on our experimental results presented in Section IV).

A. Genetic Algorithms

We suggest that the best strategy to address Problem OPT-CLR is to view the problem as a "black-box" optimization problem and explore an effective *metaheuristic* approach. In particular, we find that GAs [1] are eminently well-suited for addressing this type of complex problems. 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

[2]Note that 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.

with a single, current solution. In each iteration, a number of genetic operators are applied to the individuals of the current population in order 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 that the overall quality of the population increases as the algorithm progresses from one generation to the next.

B. GA-Based Multiple-Session Routing

Figure 1 depicts the flow-chart for our GA-based approach to Problem OPT-CLR, which includes the following components: *data structure, initialization, evaluation, selection, crossover,* and *mutation.* Note that both crossover and mutation are performed with certain probabilities (θ and μ, respectively) on the individuals. 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. Note that similar coding, crossover, and mutation operations have been used in, e.g., [9], [10] for shortest path routing. In the present paper, we extend these schemes to handle multiple concurrent sessions and for the more complex cross-layer problem. In what follows, we use the example wireless ad hoc network (with three video sessions) in Fig. 2(a) to illustrate the aforementioned steps in the GA approach.

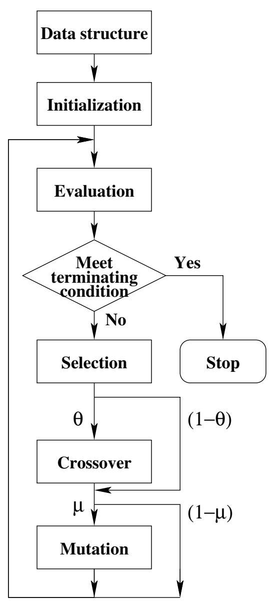

Fig. 1. Flow-chart for GA-based routing.

1) Data Structure : In order 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*). Such encoding should be suitable for fitness computation (which is determined by the objective function) and genetic operations. We define a node as a gene. Naturally, an end-to-end path can be represented as a sequence of nodes, i.e., genes. Then, for the routing problem of multiple concurrent video sessions, each feasible solution (individual) consists of a number of paths and, thus, a set of chromosomes, denoted as, e.g., $[\mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3]$ (see, e.g., Fig. 2(b)).

2) Initialization: A simple approach to generate the initial population would be to randomly append feasible elements

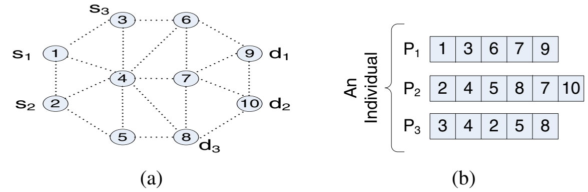

Fig. 2. (a) An example wireless ad hoc network; the dashed lines indicating wireless links. (b) An individual.

(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 that 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 sets 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.

3) Evaluation: The fitness function $h(\bar{x})$ of an individual, $\bar{x} = [\mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3]$, is closely related to the value of the objective function (i.e., the total distortion D) produced by this individual. Since the objective is to minimize the total distortion (see (12)), we have adopted a fitness function that is defined as the inverse of the distortion value, i.e., $h(\bar{x}) = 1/D(\bar{x})$. This simple definition appears to work very well, although we intend to explore other fitness definitions in our future effort.

4) Selection: During this operation, we select individuals for crossover that have a better chance or potential to produce "good" offsprings in terms of their fitness values. We used the popular *Tournament* selection scheme [1], 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.

5) Crossover: Crossover mimics the genetic mechanism of reproduction in the natural world, in which genes from parents are recombined and passed to offsprings. The crossover operation may create new individuals, and expose the search process to a new area of the fitness landscape. Figure 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 ($\mathcal{P}_2$ and $\mathcal{P}_5$). 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 $\{s_2, \cdots, g_r\}$ from $\mathcal{P}_2$ with the nodes $\{g_{r+1}, \cdots, d_2\}$ 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

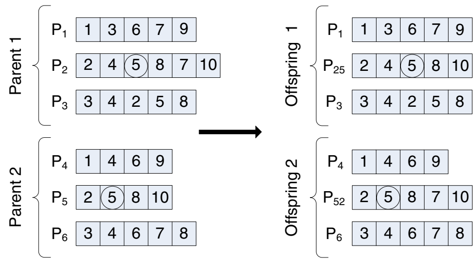

Fig. 3. The crossover operation

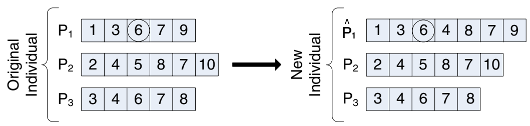

Fig. 4. The mutation operation

g_r), we can concatenate the nodes $\{s_2, \cdots, g_{r'}\}$ from $\mathcal{P}_5$ with the nodes $\{g_{r'+1}, \cdots, d_2\}$ in $\mathcal{P}_2$ to produce a new path $\mathcal{P}_{52}$. The two offsprings 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.

6) 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. Figure 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 $\{s_\sigma, \cdots, g_r\}$ be $\mathcal{P}^u_\sigma$, where g_r is the r-th node along $\mathcal{P}_\sigma$. Finally, we can use any constructive approach to build a partial path from g_r to d_σ, denoted as $\mathcal{P}^d_\sigma$, which does not repeat any node in $\mathcal{P}^u_\sigma$ (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 $\mathcal{P}^u_\sigma \cup \mathcal{P}^d_\sigma$. For the example in Fig. 4, $\mathcal{P}_1$ is chosen for mutation and node 6 is chosen to be the mutation point, yielding the revised path $\hat{\mathcal{P}}_1$ to replace $\mathcal{P}_1$. The new individual thus created is $\hat{x} = [\hat{\mathcal{P}}_1, \mathcal{P}_2, \mathcal{P}_3]$.

7) Tuning the Encoding 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. An optimal solution should yield a combination of a set of the best paths and the set of the optimal video rates, which jointly produce the lowest total distortion. In the GA-based approach, we first use the procedure described in Fig. 1 to evolve a population, assuming that 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. That is, for a given set of feasible paths $\{P_\sigma\}_{\sigma\in\mathcal{E}}$ (i.e., an individual), Problem OPT-CLR reduces to the following embedded nonlinear optimization problem:

OPT-Rate

Minimize: $$D(x_k) = \sum_{\sigma\in\mathcal{E}} D^e_\sigma \qquad (17)$$

subject to:

$$\underline{R}_\sigma \leq R_\sigma \leq \overline{R}_\sigma, \quad \forall\sigma \in \mathcal{E} \qquad (18)$$

$$\rho_{ij} \leq 1 - \epsilon, \quad \forall\{i, j\} \in \mathcal{L}, \text{for some } \epsilon. \qquad (19)$$

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* [11].

IV. Numerical Results

In this section, we present the simulation results obtained by using the GA-based approach for the concurrent video routing problem. In each experiment, a wireless ad hoc network was generated by placing a number of nodes at random locations in a rectangular region. Each video session had a rate bounded by 100Kbps and 400Kbps. We used an H.263+ codec and the 400-frame "Foreman" trace in the quarter common intermediate format (QCIF). The video was encoded with an Intra Rate of $1/15$ and a frame rate of 30 fps. Each group of blocks (GOB) was transmitted in a packet to make them independently decodable. The rate-distortion parameters were found to be $D_0 = 0.38$, $R_0 = 18.3$, $\omega = 2537$, and $\kappa = 750$ [7]. Failure probabilities of the wireless links were randomly chosen from [1%, 10%]; the bandwidth of a link was randomly chosen between [100Kbps, 400Kbps].

A. Optimality of GA Results

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 an optimal solution in a reasonable amount of time by using an exhaustive search (which, however, is infeasible for even moderate-sized networks). Table I compares the performance of the GA-based approach to the optimal results provided by exhaustive search, for small networks with three concurrent video sessions and $\Delta = 0.1$ s. GA runs for 50 iterations in each experiment, and each GA distortion value is obtained by averaging over 10 runs. We find that GA performs consistently well with respect to the optimal solution. Moreover, the computation time (a few hundred milliseconds, using a Pentium4 2.4GHz computer with 512 MB memory) is only a tiny fraction of the time required to perform the exhaustive search (about 2.5 to 9.1 hours). This gives a good idea of the time efficiency of the GA-based approach.

TABLE I
Optimality of solutions found by GA-based routing

Network	Topo. 1 9 Nodes	Topo. 2 9 Nodes	Topo. 3 11 Nodes	Topo. 4 11 Nodes
BF (dB)	24.83	26.58	26.65	27.86
GA(mean) (dB)	24.82	26.46	26.56	27.83
GA(std. dev.)	0.007	0.09	0.097	0.08

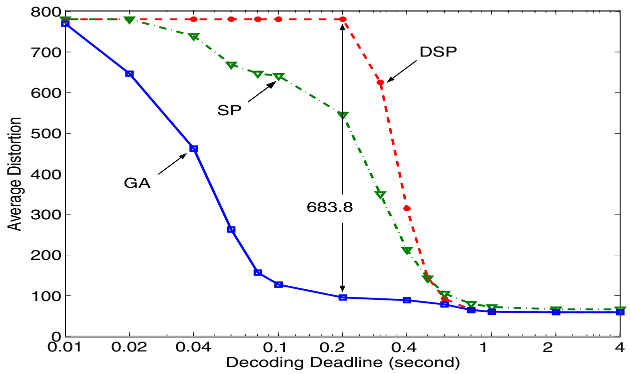

Fig. 5. Average end-to-end distortion versus decoding deadline.

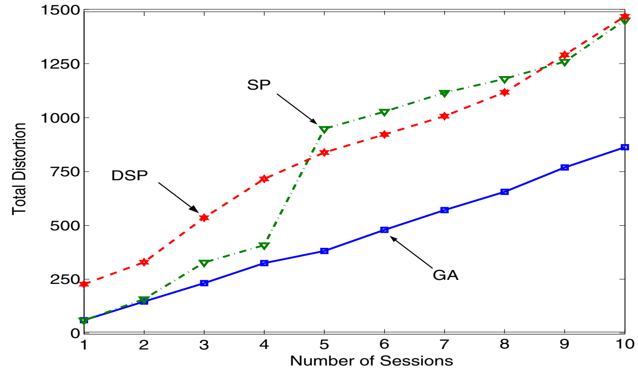

Fig. 6. Total distortion versus number of video sessions.

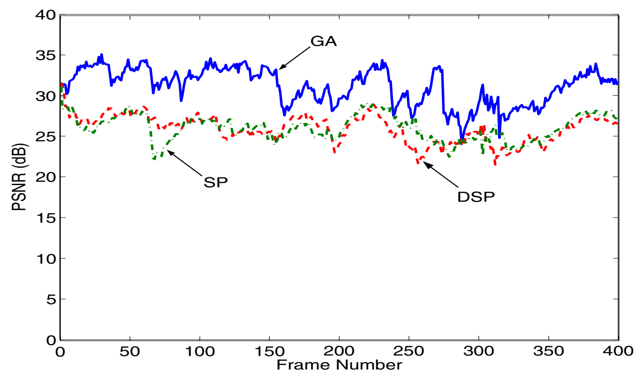

Fig. 7. PSNRs of decoded video frames.

B. GA versus Network-centric Routing

In this subsection, we compare GA-based routing with traditional network-centric routing, in order to further demonstrate the advantages of the GA-based approach. More specifically, we implement a Dijkstra's Algorithm-based shortest path routing algorithm (SP) using hop-count as the routing metric, and a disjoint shortest path routing algorithm (DSP) using loss rate as the routing metric [12]. In order to guarantee the 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, while the next path is found in this "reduced" graph.[3]

Figure 5 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 10 video sessions. We find that for very small decoding deadlines, the delay requirement is so stringent that all the three schemes yield high distortion. On the other hand, for very large decoding deadlines, the delay requirement is so loose that all the three schemes can achieve a low total distortion, as long as the stability condition is satisfied. The most interesting region, however, lies in 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. 5, the GA average distortion quickly decreases as decoding deadline increases, while the SP and DSP average distortions are persistently high for small and medium decoding deadlines (implying that most of the video packets are overdue in these cases). When $\Delta = 0.2$ s, the difference between the average distortions achieved by GA and DSP is 683.8, which translates to a significant 9.03 dB reduction in peak signal to noise ratio (PSNR) (computed as $10 \cdot \log\left(255 \times 255/D_\sigma^e\right)$).

In Fig. 6, we compare the total distortions found by GA, DSP, and SP while increasing the number of sessions in the 50-node network, in order to examine the impact of video traffic load on the routing performance. The decoding deadline is 0.5 s for all of the video sessions. It can be seen that both SP and DSP produce much higher total distortions than GA, due to the fact that they only use the network layer metric in routing. More specifically, SP does not consider the interaction of the video sessions. Although it computes the shortest path for each session, such shortest paths may share bottleneck links, resulting in congestion and high packet overdue rates. On the other hand, DSP goes to another 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 in order to satisfy the disjointedness requirement, resulting in an increased total distortion. Another interesting observation from Fig. 6 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, although the video traffic load increases nearly ten times.

So far we have investigated the impact of optimal routing in average video distortion. In order to illustrate its effect on individual video frames, we transmit encoded video on the computed paths found by GA, SP, and DSP, respectively, and plot the PSNRs of decoded frames of Session 5 in Fig. 7. It can be observed that most of the frames sent on the GA paths have much higher PSNR values than those sent on the SP or

[3]Since the resulting routes by DSP and SP depend on the order of the sessions in routing, we evaluate all possible orders and use the best results for comparison with GA. When the set of shortest paths is found, we also solve OPT-Rate to find the optimal rates for the sessions.

(a) GA. (b) SP. (c) DSP.

Fig. 8. Frame 148 of the reconstructed video.

DSP paths. The average Session 5 PSNR (over all of the 400 frames) achieved by the GA-based routing is 31.16 dB, while the average PSNRs obtained by SP and DSP are 25.93 dB 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 is directly optimized in the GA-based routing, rather than network layer metrics. We also present decoded Frame 148 in Fig. 8, obtained by the GA-based routing, SP and DSP. Clearly, the decoded frame in Fig. 8(a) has a much better visual quality than the two frames in Figs. 8(b) and 8(c).

V. Related Work

It should be clear that the problem addressed in this paper differs from the network-layer QoS-routing problems for ad hoc networks [13]–[17]. In these efforts, the focus has been on addressing network-layer routing problems from various perspectives (e.g., associativity [13] of wireless links, differentiated link state updates [14], end-to-end resource guarantees [15], [16], and selecting node or edge disjoint paths [17]). In contrast to the present paper, most of these prior efforts do not explicitly formulate the objective function with an application layer metric via a cross-layer approach. Consequently, although these approaches could obtain optimal network layer performance, they may not yield optimal performance at the application layer. In Section IV, we illustrate this point with numerical examples.

There exist some prior efforts on applying GA to address network layer problems. For example, GA has been explored in network-centric routing in [9], [10], [18]. These efforts have taken the important step in exploring the potential of GA for optimized QoS routing. The research problem addressed in this paper builds upon these earlier efforts and aims to make a major leap forward by exploring GA's potential to address the more complex cross-layer optimization problem. This problem is more relevant to multimedia communications and more difficult than the network-centric GA problems addressed in prior works, since an enlarged design and optimization space across the layers is exploited.

VI. Conclusions

In this paper, we have studied the important problem of how to optimally support multiple concurrent video communication sessions in an ad hoc network. We made two main contributions in this work. First, we have formulated an application-centric network-wide optimal routing problem with an objective function that minimizes the average distortion for all video sessions. Our problem formulation seamlessly integrates the impact of packet losses due to frequent node/link failures and congestion. Second, we have developed a highly competitive solution procedure based on the GA framework for this cross-layer optimization problem. Through extensive numerical results, we show that the GA-based approach is eminently suitable to address such complex cross-layer routing problems for concurrent multiple video sessions. Our efforts in this work provide an important methodology for addressing the problem of network-wide optimal routing for multiple concurrent video sessions.

Acknowledgments

This research has been supported in part by the National Science Foundation under Grant Numbers 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.

References

[1] T. Back, D. Fogel, and Z. Michalewicz, Eds., *Handbook of Evolutionary Computation*, Oxford University Press, New York, NY, 1997.
[2] S. Mao, Y.T. Hou, X. Cheng, H.D. Sherali, and S.F. Midkiff, "Multi-path routing for multiple description video over wireless ad hoc networks," in *Proc. IEEE INFOCOM 2005*, Miami, FL, Mar. 2005.
[3] L. Kleinrock, *Queueing Systems: Volumn I: Theory*, John Wiley & Sons, Inc., New York, NY, 1975.
[4] I. Norros, "On the use of fractional brownian motion in the theory of connectionless networks," *IEEE Journal on Selected Areas in Communications*, vol. 13, no. 6, pp. 953–962, August 1995.
[5] 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 Journal on Selected Areas in Communications*, vol. 13, no. 6, pp. 953–962, August 1995.
[6] C. Goldie and C. Klüppelberg, *Subexponential Distributions*, pp. 435–459, Birkhäuser Publishing Ltd., 1998.
[7] K. Stulmuller, N. Farberand, M. Link, and B. Girod, "Analysis of video transmission over lossy channels," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 6, pp. 1012–1032, June 2000.
[8] Z. Wang and J. Crowcroft, "Quality-of-Service routing for supporting multimedia applications," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 8, pp. 1488–1505, August 1999.
[9] M. Gen and R. Cheng, *Genetic Algorithms & Engineering Optimization*, John Wiley & Sons, Inc., New York, NY, 2000.
[10] C.W. Ahn and R.S. Ramakrishna, "A genetic algorithm for shortest path routing problem and the sizing of populations," *IEEE Trans. on Evolutionary Computation*, vol. 6, no. 6, pp. 566–579, Dec. 2002.
[11] M.S. Bazaraa, H.D. Sherali, and C.M. Shetty, *Nonlinear Programming: Theory and Algorithms*, John Wiley & Sons, Inc., New York, NY, second edition, 1993.
[12] T.H. Cormen, C.E. Leiserson, and R.L. Rivest, *Introduction to Algorithms*, The MIT Press, Cambridge, Massachusetts London, England, 1990.
[13] C.-K. Toh, *Ad Hoc Mobile Wireless Networks: Protocols and Systems*, Prentice Hall, New York, NY, 2001.
[14] R. Sivakumar, P. Sinha, and V. Bharghavan, "CEDAR: A core-extraction distributed ad hoc routing algorithm," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 8, pp. 1454 – 1465, August 1999.
[15] C.E. Perkins, E.M. Royer, and S.R. Das, "Quality of service in ad hoc on-demand distance vector routing," July 2000.
[16] C.R. Lin and J.-S. Liu, "QoS routing in ad hoc wireless networks," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 8, pp. 1426–1438, August 1999.
[17] P. Papadimitratos, Z.J. Haas, and E.G. Sirer, "Path set selection in mobile ad hoc networks," in *Proc. ACM MobiHoc*, June 2002, pp. 1–11.
[18] N. Banerjee and S.K. Das, "Fast determination of QoS-based multicast routes in wireless networks using genetic algorithm," in *Proc. IEEE ICC*, June 2001, pp. 2588–2596.