Optimized Mesh Routing with Intermediate Recovery for Error Resilient Delivery of MD Coded Image/Video Content

Chetna Singhal[a], Swades De[b,*], Moravapalle Uma Parthavi[c]

[a] *Department of Electronics and Electrical Communication Engineering, IIT Kharagpur, Kharagpur, India*
[b] *Department of Electrical Engineering and Bharti School of Telecom, IIT Delhi, New Delhi, India*
[c] *Department of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA, USA*

Abstract

Multiple Description (MD) source coding is a technique that breaks a media stream into equally important sub-streams which can be sent over different paths for protection against wireless channel errors. In this paper, we explore the possibility of sending these descriptions through different paths that merge at some specific intermediate nodes, where the corrupted descriptions are recovered from the uncorrupted ones, thereby increasing the quality of received image/video at the destination. To quantify the gain with intermediate recovery, we first devise an analytic model with simplifying assumptions on the network system for quantifying end-to-end distortion of MD coded data as a function of path parameters, and demonstrate that on a Lena image transmitted over long multipath routes, one intermediate recovery stage offers up to 9.2% reduction in distortion compared to the traditional multipath transport. Next, accounting the random network topology we formulate mesh route construction as a cross-layer optimization problem to balance between end-to-end packet delivery delay and distortion. Since this problem is highly complex, we propose two alternative delay/distortion minimization heuristics. Further, a jointly delay and distortion optimizing genetic algorithm based meta-heuristic route construction technique is suggested for networks with highly varying link quality. NS2-based simulations of a realistic network scenario demonstrate that, in terms of peak signal-to-noise ratio the intermediate recovery

*Corresponding author. Tel.: +91.11.2659.1042; fax: +91.11.2658.1606.
Email addresses: chetna@ece.iitkgp.ernet.in (Chetna Singhal), swadesd@ee.iitd.ac.in (Swades De), par3thavi@gmail.com (Moravapalle Uma Parthavi)

approach results in substantial improvement, close to 15 dB, in quality of video delivery.
Keywords: Multiple description coding, multipath routing, intermediate recovery, multihop wireless networks, genetic algorithm, cross layer optimization

1. Introduction and Motivation

Due to the ever-increasing popularity of real-time multimedia applications, e.g., Internet TV, video conferencing, and HD (high definition) streaming, there has been a growing demand to enable the use of such applications over wireless networks. However, error prone nature of wireless channels makes the transmission of such content rich media very difficult. Another complexity involved, specially with videos, is the progressive nature of compression schemes, due to which errors in one frame may propagate over several frames, thereby resulting in a poor video reception quality. Hence, there is an urgent need to develop new methods to achieve error resilience for image/video content delivery over wireless mesh networks.

Multiple description coding (MDC) is a progressive scalable technique that offers graceful degradation of video quality with channel errors. MDC does not rely on link layer ARQ (automatic repeat request) or prioritized delivery of some content akin to layered coding. In MDC, source media stream is broken into several substreams, called descriptions. Depending on the MDC type, the substreams may or may not be equally important – leading to symmetric or asymmetric MDC, respectively. The descriptions are constructed such that there is some correlation among them, to help recovery from partial losses. Correlation can be naturally derived using spatial resolution in images/video, temporal resolution in video, frequency content, signal-to-noise ratio (SNR), etc., or it is artificially introduced by adding redundancy, as in transform coding and quantization. The descriptions are transported over different channels so that the chance of losing descriptions corresponding to the same section of image/video is less. Besides improved error resilience, such use of path diversity also warrants several advantages [1, 2, 3], namely, improved aggregate bandwidth, traffic load balancing, and reduced latency. If an end user is unable to process all the descriptions (e.g., due to memory constraints) or the transmitter is unable to send them all (say, due to bandwidth constraints), the descriptions can be dropped.

We observe that, by virtue of underlying link layer flow and error control, on each hop adaptive MCS (modulation and coding scheme) and ARQ ensure low packet error rate. We call them *intra-packet recovery* approaches. However, it is possible that, during some transient phases, e.g., due to mobility, link adaptation may not respond fast, thereby causing high packet losses. To this end, MDC works as a complementary approach to MCS and ARQ, wherein lost packets are estimated from the correctly-received ones, which we call *inter-packet recovery*. Thus, MDC can work effectively with limited MCS and ARQ in delay-constrained video streaming applications.

1.1. MDC with intermediate recovery

In multipath transport (MPT) of MD coded packets, traditionally at the destination the corrupted descriptions are reconstructed using the uncorrupted ones. Instead of recovering at the destination (as in Figure 1(a)), if one can do it intermediately, there is a high chance of improved performance. If we are to construct multipath routes for transmitting the

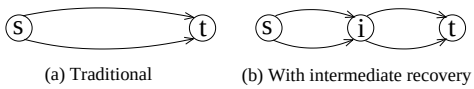

Figure 1: Examples of MD coded transmission over multiple paths.

descriptions, to aid intermediate recovery they should be edge-disjoint but merge at specific nodes (as in Figure 1(b)). This paper explores the concept of *intermediate recovery over optimized multipath routes*.

Definition 1. *Intermediate recovery is a process of recovering from channel errors in a description at an intermediate point, i.e., at a node before the destination, using the correlation information present in the descriptions.*

Consider the traditional MDC-MPT system (Figure 1(a)). Let the number of links in each path between source s and destination t be two, and the probability of losing a description over a link be ϕ. The probability of not receiving any description at t is $4\phi^2$. In MDC-MPT with intermediate recovery in (Figure 1(b)), at node i recovery of a lost description takes

place. If the number of links in path between s and t is two (with one link between s and i and the other between i and t), the probability of not receiving any description at t is $2\phi^2$. Clearly, this preliminary probabilistic analysis tells us that intermediate recovery achieves significant performance gain.

1.2. Related work

Usefulness of multi-stream coding with MPT to overcome the transmission errors in ad hoc networks was demonstrated in [4]. Different MDC techniques for image/video delivery over wireless networks were surveyed in [5]. Optimum redundancy in MDC under various channel fading conditions was investigated in [6]. However, MPT was not in the purview of the work. Several variants of MDC-MPT were proposed in wireless networks [7, 8, 9], where the effects of path diversity along with source diversity of image/video transmission were studied. End-to-end distortion of MDC-MPT system was modeled in [10] as a function of channel bandwidth, error rate, delay between two nodes, and delay jitter. For video transmission over multi-channel multihop wireless networks, distributed packet scheduling was considered in [11] to minimize the distortion and achieve user-level fairness. Another study in [12] considered MD coded video transmission over 802.11 wireless mesh networks, where maximally disjoint multipath was suggested for traffic load balancing purpose. These works however considered packet overlap and recovery only at the destination; the virtues of mesh network were not exploited.

On the multipath route search, sequential k-shortest path route construction was proposed in [13]. An algorithm using k Dijkstra's like computations was developed in [14] to obtain k disjoint paths from a source s to a destination t in a non negatively weighted directed graph that have the minimum total cost (sum of costs of all links in the paths). The above algorithm was modified in [15] to calculate two disjoint paths with a minimum total length from a node to all other nodes using a single Dijkstra like computation in $O\left(m \log_{1+\frac{m}{n}} n\right)$ time, where m is the number of edges and n is the number of nodes in the network. Since Dijkstra's algorithm can be implemented using the technique in [15] in $O\left(\min\left\{m + n\log n, m\log\log C, m + (n\log C)^{0.5}\right\}\right)$, where C is the largest edge cost in the

graph, the disjoint paths from a single node to all destinations with total minimum cost can be constructed in the same time. A parallel version of the algorithm proposed in [16] implemented it in $n^3/\log n$ processors and $O\left(\log^2 n\right)$ time. A distributed algorithm was presented in [17] to compute shortest disjoint path pairs from s to t with an implementation on a network having unit edge costs, achieving communication and time complexities $O(m+\delta n)$ and $O(\delta_2)$, respectively, where δ is the network diameter, and δ_2 is the maximum of the total number of links in the shortest disjoint pairs from i to t, for all i.

The work in [18] discussed cross-layer framework that enhances the network capacity by increasing the number of video sources while preserving QoS constraint of each source in wireless multimedia sensor network. The authors in [19] proposed an approach for layered MDC video delivery in relay-assisted LTE-A (long term evolution-advanced) network. An experimental demonstration of multipath SVC video streaming scheme was reported in [20]. Energy-efficiency gain with width-controllable mesh multipath routing in sensor network was studied in [21]. From a more physical layer perspective, [22] studied cooperative relaying in multihop networks and proposed a cross-layer framework that optimizes multipath routing, scheduling, rates, transmit powers, and selection of cooperative nodes to attain higher throughputs. Mesh multipath routing has also been exploited [23] in avoiding compromized nodes in secure routing.

In a multiuser multihop network, asymmetric MD coded video transmission was studied in [24] with an objective of optimal rate allocation to different users to minimize packet losses over unreliable links. In another application of MDC, H.264 coded video content delivery to multiple users with varying link quality was considered over multihop routes [25], where MDC rate control was practiced to adapt to channel errors and end-to-end delay constraints. Network coding in conjunction with MDC was proposed in [9] for instant decoding of received packets, resulting in improved quality of video streaming over wireless ad hoc networks. The study in [26] showed that MDC improves quality of real-time multimedia communications due to its error recovery potential under various network conditions. The authors in [27] discussed a mechanism based on MDC and multiple gateways in wireless mesh networks to

improve streaming video quality. In [28], adaptive delivery of layered scalable video coding (SVC) or MDC video frames over wireless links was used to improve video quality. Network adaptive MDC scheme was studied in [29] to improve delivered video quality over wireless. In [30], a single description video content at the source was proposed to be transcoded to MDC streams at an intermediate node to improve video transmission quality in multi-radio, multipath environment. In contrast, in our proposed scheme transcoding MPEG4 video to MDC streams is done at the source prior to transmission over the wireless mesh network.

In ad hoc networks, routing with multiple constraints is computationally intractable and a complicated combinatorial optimization problem [31]. Genetic algorithm (GA) is a heuristic approach that has been used to determine network topology in order to minimize average delay [31] and for path discovery [32]. Real-time multimedia applications use GA to find feasible paths in connectionless networks with multiple-constraints [33]. GA has also been used for obtaining minimum cost tree in multicast routing problem [34]. In multicast routing, GA based solutions minimize cost with constrained delay for multimedia applications [35] and optimize network resources (bandwidth and delay) in uncertain network environments [36].

The intermediate recovery concept was explored in [37, 38] by using a spatial interpolation technique on lost descriptions. However, the routing performance optimization over random network topologies was not conducted.

Overall, to our knowledge a detailed analytic study and optimization of MD coded transmission over mesh networks for image/video content delivery is still missing in the literature.

1.3. Key contributions

In this paper, we explore optimal mesh routing over random wireless mesh networks for efficient delivery of MD coded image/video content. In contrast to the approaches in [10, 7], we model the MDC-MPT system with intermediate recovery using various physical layer wireless link parameters. Our key contributions are as follows: (i) An analytical model is developed to quantify the quality of image/video content delivery in a fading channel environment in terms of mean square error (MSE) distortion in a MDC-MPT system with

intermediate recovery. (ii) A cross-layer optimization problem is formulated for multipath route construction to aid intermediate recovery. Also an intermediate recovery aware route construction heuristic is proposed for a two description (2D) MDC system. (iii) For jointly minimizing delay and distortion in a network with high link performance variability, a genetic algorithm (GA) technique is customized to the mesh route construction problem. (iv) NS2 simulations on MDC-MPT are conducted with practical network system parameters for transmission of standard video sequences. We show that the performance gain with intermediate recovery is higher when the error correction capability of the coding scheme is increased. The analytical results with simplifying assumptions on the network settings indicate that, having one recovery stage on a Lena image frame transmitted over long multipath routes leads to about 9% decrease in the end-to-end distortion. Proposed mesh routing in a realistic network environment verifies the analytic claim, where it is also shown that the intermediate recovery performance gain increases with node density. Our simulation results show close to 15 dB peak signal-to-noise ratio (PSNR) gain in video reception quality.

It may be highlighted that, intermediate recovery of MD coded packets along multipath routes serves a different goal in contrast with the joint network coding and routing [39] where reduction of network bandwidth overhead is targeted at the cost of some additional nodal computations.

1.4. Paper organization

The rest of the paper is organized as follows: In Section 2, analysis of end-to-end distortion of MD coded video over multihop wireless mesh routes with intermediate recovery is presented. Section 3 presents the delay/distortion optimization problem formulations and the associated heuristics. GA based mesh route construction strategies are outlined and evaluated in Section 4. Numerical results and NS2 simulation based MDC-MPT system performance are presented in Section 5. The concluding remarks are drawn in Section 6.

2. Distortion Analysis

2.1. Distortion measure

We consider a two description (2D) MDC for image/video transmission (cf. Figure 2), because most of the popularly used techniques are 2D systems. The consideration is also

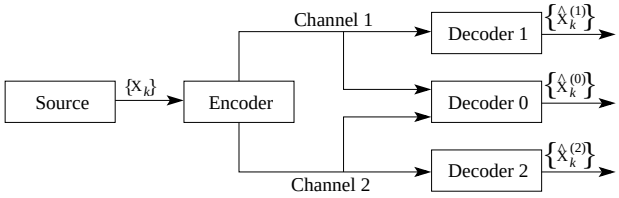

Figure 2: Multiple description coding with two channels and three receivers.

motivated by the fact demonstrated in [40] that the increase in performance gain slows down significantly as the number of descriptions is increased beyond two. The video reception quality at the destination is described using standard MSE distortion, or sometimes by PSNR. If R is the range of values a pixel can take, the relation between PSNR and MSE is:

$$PSNR = 10 \log_{10} \frac{R^2}{MSE} \tag{1}$$

Assume, the source data (e.g., video) $\{X_k\}$ is fed into a MDC encoder to construct two descriptions $\left\{X_k^{(1)}\right\}$ and $\left\{X_k^{(2)}\right\}$, and are transmitted over two channels. Let P_{00}, P_{01}, P_{10}, and P_{11} be respectively the probabilities that both descriptions are available for decoding (i.e., $\left\{\hat{X}_k^{(0)}\right\}$ received), only the first description is available ($\left\{\hat{X}_k^{(1)}\right\}$ received), only the second description is available ($\left\{\hat{X}_k^{(2)}\right\}$ received), and none of the descriptions are available. The corresponding distortions are defined as: $\eth_0 = D\left(\left\{\hat{X}_k^{(0)}\right\}, \{X_k\}\right)$, $\eth_1 = D\left(\left\{\hat{X}_k^{(1)}\right\}, \{X_k\}\right)$, $\eth_2 = D\left(\left\{\hat{X}_k^{(2)}\right\}, \{X_k\}\right)$, and $\eth_3 = D\left(\{\Upsilon_k\}, \{X_k\}\right)$, respectively, where $\{\Upsilon_k\}$ denotes both descriptions are corrupted. Then, the average distortion $\mathcal{D}$ of the video can be expressed as:

$$\mathcal{D} = P_{00}\eth_0 + P_{01}\eth_1 + P_{10}\eth_2 + P_{11}\eth_3 \tag{2}$$

In symmetric MDC, $\eth_1 = \eth_2$. The values of $\eth_i$, $i = 0, 1, 2, 3$, depend on the type of MDC used. For all types of MDC, if $D(\cdot, \cdot)$ increases with the degradation of quality,

$\eth_3 \gg \eth_1, \eth_2 \gg \eth_0$. For analysis of the 2D MDC-MPT performance, we consider a technique for video, called multiple description transform coding (MDTC) [41]. Intermediate recovery system would achieve a lower distortion as compared to the traditional one, irrespective of the coding technique used, as long as the condition $\eth_0 \ll \eth_1, \eth_2 \ll \eth_3$ holds. In any MDC scheme, the distortion introduced when no descriptions are lost is much less than the distortion when one description is lost. But, the method of recovery of a lost description and the associated amount of distortion introduced depend on the particular scheme in hand.

2.2. Network model

A mobile wireless mesh network is modeled as a stochastic directed graph $G(V, E)$, where V is the set of nodes and E is the set of edges. A directed edge from node i to j is denoted by (i, j). Let the cost (in terms of delay) associated with (i, j) be c_{ij}, and the respective paths taken by descriptions 1 and 2 be $\mathcal{P}_1$ and $\mathcal{P}_2$. For the analytic exposition, the number of nodes encountered by the two descriptions (i.e., the number of hops) along the source-to-destination path are assumed equal. If there are N intermediate recovery nodes in a source-destination path pair, each path is divided into $N + 1$ segments. A segment is a path fraction between a source (respectively, an intermediate recovery node) to the subsequent intermediate recovery node (respectively, the destination). The kth segment in the lth path is denoted by H_k^l, for $k = 1$ to $N + 1$, $l = 1, 2$.

2.3. Wireless channel representation

Small-scale Rayleigh fading is accounted in every link (i, j), where the signal experiences rapid fluctuations in a small period of time or space. The fading is assumed to be slow with respect to the data transmission rate. In this case, success/failure of consecutive data bits are not independent. The transmission errors in such a fading channel can be approximately represented by a two state Markov model [42]. The channel can be seen as a two state Gilbert Channel, with Good/Bad states (cf. Fig. 3). Over a link (i, j), if a_{ij} is the transition probability from Good to Bad state and that from Bad to Good state is b_{ij}, with fading margin F, Doppler frequency f_d, and sampling interval $\mathcal{T}$, the steady state link error

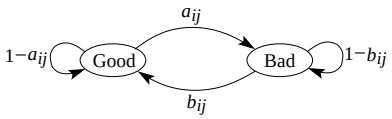

Figure 3: Gilbert channel model.

probability is [42]:

$$\phi_{ij} = 1 - \frac{b_{ij}}{a_{ij} + b_{ij}} = 1 - e^{-\frac{1}{F}}, \tag{3}$$

$$\text{with} \quad b_{ij} = \frac{Q(\theta, \rho\theta) - Q(\rho\theta, \theta)}{e^{1/F} - 1} \tag{4}$$

Here Q is the Marcum-Q function, $\theta = \sqrt{\frac{2/F}{1-\rho^2}}$, and $\rho = J_0(2\pi f_d \mathcal{T})$. f_d depends on the relative velocity v between the nodes i and j and the signal wavelength λ as: $f_d = \frac{v}{\lambda}\cos\varphi$, where φ is the signal incidence angle with respect to relative movement direction of the receiver.

2.4. Aggregated wireless link model

In a 2D MDC-MPT system, the two transmission paths are designated by the superscript l $(= 1, 2)$. A description packet is corrupted when more than τ bits in it are in error. This threshold τ depends on the error correction capability of the coding scheme. Referring to Figure 3, over a link (i, j), the fraction of time spent in Bad state is ϕ_{ij}. To reduce end-to-end delay, we consider bit-level decoding only at the end node or at the intermediate recovery nodes. Since the transmission across the segment H_k^l, $k = 1$ to $N + 1$, $l = 1, 2$, is successful only if all the links $\{(i, j) \in H_k^l\}$ are in Good state, the $2^{|H_k^l|}$ channel states (two corresponding to each link) can be lumped in one Good and one Bad state. The aggregated Bad state corresponds to all the cases where there is at least one link in Bad state. Let the transition probability from the aggregated Good state to aggregated Bad state, aggregated Bad state to aggregated Good state, and the fraction of time spent by the segment H_k^l in the aggregated Bad state be a_k^l, b_k^l, and Φ_k^l, respectively.

Theorem 1. *The state transition probabilities of the aggregated end-to-end link are:*

$$a_k^l = 1 - \prod_{(i,j)\in H_k^l} (1 - a_{ij}), \tag{5a}$$

$$\Phi_k^l = 1 - \prod_{(i,j)\in H_k^l} (1 - \phi_{ij}), \tag{5b}$$

$$b_k^l = \frac{(1 - \Phi_k^l)a_k^l}{\Phi_k^l} \tag{5c}$$

Proof. The aggregated link is in Good state if each of the individual channels are Good. So, $1-a_k^l = \prod_{(i,j)\in H_k^l} (1-a_{ij})$. ϕ_k^l is the fraction of time the aggregated link is Bad, i.e., $\Phi_k^l = \frac{a_k^l}{a_k^l+b_k^l}$. The fraction of time the aggregated link is Good can be viewed as the product of fractions of time the individual channels are Good, i.e., $1 - \Phi_k^l = \prod_{(i,j)\in H_k^l} (1 - \phi_{ij})$. Hence (5c). □

2.5. Corruption probability of a description

Say, an error correction code can resolve up to τ errors in a description. Let $\Psi(n_c, L)$ be the probability of n_c out of total L bits in a description packet being correctly received along the H_k^l hop path in segment k of route l. With τ bits error correction capability, the probability is:

$$P_k^l \stackrel{\Delta}{=} Pr(\text{errors} > \tau) = \sum_{n_f=\tau+1}^{L} \Psi(n_c, L) \tag{6}$$

where $n_f = L - n_c$ is the number of bits flipped over the path segment.

To obtain $\Psi(n_c, L)$, we denote the aggregated system state (Good/Bad) at the first bit transmission as $\zeta(0) = g_0$ (respectively, $\zeta(0) = b_0$) and at the Lth bit transmission (out of which n_c bits are successful) as $\zeta(L) = g_{n_c}$ (respectively, $\zeta(L) = b_{n_c}$). Thus, for example, the transition probability from g_0 to g_{n_c} is: $\Omega_{g_0 g_{n_c}}(L) = Pr\{\zeta(L) = g_{n_c} | \zeta(0) = g_0\}$. Via

z-transform domain analysis for this general Markov process [43] and simplifying we have,

$$\Omega_{g_0 g_{n_c}}(L) = \begin{cases} \sum_{n=1}^{n_c} \binom{n_c}{n}\binom{L-n_c-1}{n-1} \left(a_k^l\right)^n \left(1-a_k^l\right)^{n_c-n} \left(b_k^l\right)^n \left(1-b_k^l\right)^{L-n_c-n}, & 0 < n_c < L \\ 0, & n_c = 0 \\ \left(a_k^l\right)^L, & n_c = L \end{cases} \tag{7a}$$

$$\Omega_{g_0 b_{n_c}}(L) = \begin{cases} \sum_{n=0}^{n_c} \binom{n_c}{n}\binom{L-n_c-1}{n} \left(a_k^l\right)^{n+1} \left(1-a_k^l\right)^{n_c-n} \left(b_k^l\right)^n \left(1-b_k^l\right)^{L-n_c-n-1}, & 0 \le n_c < L \\ 0, & n_c = L \end{cases} \tag{7b}$$

$$\Omega_{b_0 g_{n_c}}(L) = \begin{cases} \sum_{n=0}^{n_c-1} \binom{n_c-1}{n}\binom{L-n_c}{n} \left(a_k^l\right)^n \left(1-a_k^l\right)^{n_c-n-1} \left(b_k^l\right)^{n+1} \left(1-b_k^l\right)^{L-n_c-n}, & 0 < n_c \le L \\ 0, & n_c = 0 \end{cases} \tag{7c}$$

$$\Omega_{b_0 b_{n_c}}(L) = \begin{cases} \sum_{n=0}^{n_c-1} \binom{n_c-1}{n}\binom{L-n_c}{n+1} \left(a_k^l\right)^{n+1} \left(1-a_k^l\right)^{n_c-n-1} \left(b_k^l\right)^{n+1} \left(1-b_k^l\right)^{L-n_c-n-1}, & 0 < n_c < L \\ \left(b_k^l\right)^L, & n_c = 0 \\ 0, & n_c = L \end{cases} \tag{7d}$$

Hence, $\Psi(n_c, L)$ can be obtained as:

$$\Psi(n_c, L) = \left(1-\Phi_k^l\right)\left[\Omega_{g_0 g_{n_c}}(L) + \Omega_{g_0 b_{n_c}}(L)\right] + \Phi_k^l\left[\Omega_{b_0 g_{n_c}}(L) + \Omega_{b_0 b_{n_c}}(L)\right] \tag{8}$$

2.6. Probabilistic representation of intermediate recovery

The error state model of the intermediate recovery system is represented by the state transition diagram in Figure 4. The possible states at an intermediate recovery node

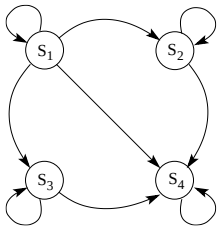

Figure 4: Intermediate recovery system state transitions. Transition probabilities at the kth segment of the path are found in (9).

can be: S_1 (both descriptions are uncorrupted/recoverable), S_2 (description 1 is uncorrupted/recoverable and description 2 is corrupted/unrecoverable), S_3 (description 2 is uncorrupted/recoverable and description 1 is corrupted/unrecoverable), and S_4 (both the descriptions are corrupted/unrecoverable). The system begins at state S_1 at the source. If along the path between source and the next intermediate recovery node one of the descriptions gets corrupted beyond recovery, the system moves into S_2 or S_3 (depending on which description is lost). If both the descriptions are lost simultaneously the system moves to S_4. Once the system moves to either S_2 or S_3, the system can either stay in itself or go to S_4. If the system reaches state S_2 or S_3, the lost description is estimated and transmitted. This estimated description is in a way the best estimate of the lost original description given the other, and is a duplicate of the original uncorrupted description. The system cannot go back from S_2/S_3 to S_1 because there is loss of information which can be only partially recovered by the estimate. Once the system reaches S_4, it remains there for all subsequent state transition opportunities, in which case the mean of the data is transmitted in place of the data itself here. If the probability of loss of a description in a segment H_k^l is P_k^l, for $k = 1$ to $N+1$, $l = 1, 2$, the transition probability matrix U_k is given by:

$$U_k = \begin{pmatrix} (1-P_k^1)(1-P_k^2) & P_k^2(1-P_k^1) & P_k^1(1-P_k^2) & P_k^1 P_k^2 \\ 0 & 1-P_k^1 P_k^2 & 0 & P_k^1 P_k^2 \\ 0 & 0 & 1-P_k^1 P_k^2 & P_k^1 P_k^2 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{9}$$

P_k^l is obtained from (5), (6), (7), and (8). Transition probability matrix U for the overall path is given by

$$U = \prod_{k=1}^{N+1} U_k \tag{10}$$

In a 2D MDC system, (2) gives the end-to-end distortion for transmission over a given pair of paths. Since the system starts in state S_1 at the source, P_{00}, P_{01}, P_{10}, and P_{11} are given by the elements of first row of U, taken in order of the column number.

With the above formulation of end-to-end distortion, some theoretical observations are made in this Section, where for simplicity channel with homogeneous error rate on all links is assumed and no forward error correction capability (i.e., $\tau = 0$) is considered at the packet level.

Theorem 2. *Without intermediate recovery, end-to-end distortion $\mathcal{D}$ increases with the number of hops H in a path.*

Proof. This is proved by showing that the derivative of $\mathcal{D}$ with respect to H is always positive. In symmetric MDC, over a single segment the probability of corruption of the descriptions is the same, i.e., $P_k^1 = P_k^2 = P$ (say). From (2), the end-to-end distortion $\mathcal{D}$ can be written as:

$$\mathcal{D} = (1-P)^2 \eth_0 + 2P(1-P)\eth_1 + P^2 \eth_3 \tag{11}$$

Differentiating $\mathcal{D}$ with respect to P and upon rearranging the terms we have:

$$\frac{d\mathcal{D}}{dP} = 2(1-P)(\eth_1 - \eth_0) + 2P(\eth_3 - \eth_1)$$

Since $\eth_3 \gg \eth_1 \gg \eth_0$, $\eth_1 - \eth_0$ and $\eth_3 - \eth_1$ are positive quantities. Also, $(1-P)$ is always positive. Hence, $\frac{d\mathcal{D}}{dP} > 0$.

With homogeneous error rate along all links, $a_{ij} = a;\ b_{ij} = b\ \forall (i,j) \in E$. With $\tau = 0$ we have from (3), $\phi = \frac{a}{a+b}$. So, the corruption probability of an L bit description packet over H-hop path segment, represented by (6), is reduced to:

$$P = 1 - (1-\phi)^H (1-a)^{(L-1)H} \tag{12}$$

Differentiating P with respect to H, and noting that ϕ and a are less than 1,

$$\frac{dP}{dH} = -(1-\phi)^H (1-a)^{(L-1)H} (\ln(1-\phi) + (L-1)\ln(1-a)) > 0 \tag{13}$$

Therefore, $\frac{d\mathcal{D}}{dH} = \frac{d\mathcal{D}}{dP}\frac{dP}{dH} > 0$. When $L > 1$ and $\tau > 0$, the analytical expression for distortion becomes quite complex.

For $\tau > 0$, simulations on MATLAB are done to verify Theorem 2. Figure 5 shows the distortion performances of 2-D MDC-MPT and repetition coding [44], respectively, over

symmetric 2-path routes. Doppler spread and symbol duration product is $f_d T_s = 0.0004128$. The end-to-end distance is varied from 1 to 50 hops. (In this paper, without loss of generality we have considered $T_s = \mathcal{T}$, the channel sampling interval.) It is noticed that MDC-MPT re-

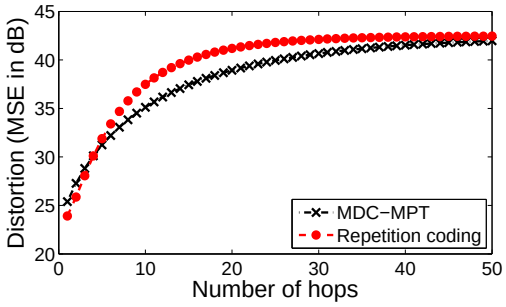

Figure 5: Variation of distortion versus end-to-end distance (hop count). $F = 25$ dB, $f_d T_s = 0.0004128$, $L = 1000$ bits, $\tau = 25$ bits.

sults in lower distortion than repetition coding beyond short (one-hop) source-to-destination distance. This is because, beyond one-hop distance the benefit of intermediate recovery in MDC-MPT can be achieved – which supersedes the gain due to repetition coding. Thus, the proposed MDC-MPT scheme outperforms the trivial repetition coding (in terms of lower distortion) by using half the network resource (bandwidth). □

Theorem 3. *For minimal distortion, the intermediate recovery node should be placed equidistant from the source and destination.*

Proof. For the sections of path between source to intermediate recovery node and intermediate recovery node to destination, let the probabilities of corruption of a description be $\wp_1$ and $\wp_2$. It is assumed that the paths for both the descriptions are symmetric, i.e., the number of hops in both the paths is the same (say κ). Let the intermediate recovery node be placed H hops away from the source. End-to-end distortion $\mathcal{D}$ as a function of H, in terms of $\wp_1$ and $\wp_2$ is given by:

$$\begin{aligned}\mathcal{D} =&(1-\wp_1)^2(1-\wp_2)^2\eth_0 + \left[\wp_1^2+\wp_2^2-\wp_1^2\wp_2^2\right]\eth_3 \\ &+ 2\left[(1-\wp_1)(1-\wp_2)(\wp_1+\wp_2)\right]\eth_1\end{aligned}$$

Differentiating $\mathcal{D}$ with respect to H, $\frac{d\mathcal{D}}{dH} = 2\xi_1 \frac{d\wp_1}{dH} + 2\xi_2 \frac{d\wp_2}{dH}$, where,

$$\begin{aligned}
\xi_1 =& -(1-\wp_1)(1-\wp_2)^2\eth_0 + \wp_1\eth_3 - \wp_1\wp_2^2\eth_3 \\
&+ (1-\wp_2)(1-2\wp_1-\wp_2)\eth_1 \\
\xi_2 =& -(1-\wp_2)(1-\wp_1)^2\eth_0 + \wp_2\eth_3 - \wp_2\wp_1^2\eth_3 \\
&+ (1-\wp_1)(1-2\wp_2-\wp_1)\eth_1
\end{aligned}$$

$$\frac{d^2\mathcal{D}}{dH^2} = \frac{d\xi_1}{dH}\frac{d\wp_1}{dH} + \xi_1\frac{d^2\wp_1}{dH^2} + \frac{d\xi_2}{dH}\frac{d\wp_2}{dH} + \xi_2\frac{d^2\wp_2}{dH^2} \tag{14}$$

With homogeneous error rate along all links, $a_{ij} = a$; $b_{ij} = b$ $\forall (i,j) \in E$. For $\tau = 0$, from (3), $\phi = \frac{a}{a+b}$. Then, considering again description of size L bits, the corruption probabilities of the descriptions $\wp_1$ and $\wp_2$ are given by $\wp_1 = 1-(1-\phi)^H(1-a)^{(L-1)H}$, $\wp_2 = 1-(1-\phi)^{\kappa-H}(1-a)^{(L-1)(\kappa-H)}$.

We have, $\frac{d\wp_1}{dH} = -\frac{d\wp_2}{dH}$, and $\wp_1 = \wp_2$ for $\kappa = 2H$. It easy to show that the first derivative of $\mathcal{D}$ is zero at $H = \frac{\kappa}{2}$ and the second derivative at $H = \frac{\kappa}{2}$ is positive under the assumption that $\eth_3 \gg \eth_1 \gg \eth_0$. □

The numerically computed plot in Figure 6 clearly shows, the end-to-end distortion is

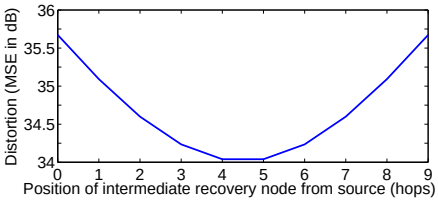

Figure 6: Effect of intermediate recovery position. $\kappa = 11$, $F = 25$ dB, $f_dT_s = 4.128 \times 10^{-4}$, $L = 1000$ bits, $\tau = 25$ bits.

minimized when the recovery node is approximately equi-spaced between the source and destination.

Corollary 1. *The intermediate nodes should be equi-spaced between source and destination to obtain minimum distortion.*

Proof. It is intuitive from Theorem 3 that, with more than one intermediate recovery nodes along the multipath routes, for minimal distortion the intermediate recovery node between two consecutive disjoint multipath segments should be equi-spaced from the segment-end nodes. This, by extension, leads to conclude that the intermediate recovery nodes should be equi-spaced between the source and the destination. □

3. MPT Route Setup for MDC Transmission with Intermediate Recovery

The distortion analysis in Section 2 gave an important quantitative insight to the benefit of mesh multipath routing with intermediate recovery, where for analytical tractability the formulation assumed regular network topology with equal number of hops and the possibility of any number of optimally-located intermediate recovery points over the multipath route. However, in a realistic network setting the availability of such idealized routes with equal number of hops along multiple paths and optimally positioned intermediate recovery points are quite difficult.

In a more practical scenario, the route length of the multiple paths are expected to be different, which, besides impacting the distortion performance, also introduces additional delay due to intermediate recovery. The above analysis therefore does not capture and hence does not optimize the end-to-end distortion and delay tradeoff.

In this section, we reconsider the intermediate recovery from a more realistic network setting perspectives, where the end-to-end delay is additionally accounted along with the distortion. A major task here is the optimal route construction to extract the benefit of intermediate recovery.

A good route for MDC with intermediate recovery should have low delay, low distortion, disjointed links, mesh, low computation. Accordingly, multipath construction for intermediate recovery is modeled as a cross-layer optimization problem in which the objective is a complex function of wireless channel parameters and application content. This application-aware optimization is ideal owing to the changing dynamics of wireless ad hoc mesh networks.

Let $\underline{D} = [\eth_0\ \eth_1\ \eth_2\ \eth_3]$ and $\underline{P} = [P_{00}\ P_{01}\ P_{10}\ P_{11}]$. In order to define paths, two flow

variables x_{ij}^l, $l = 1, 2$ for every $(i, j) \in E$ are introduced:

$$x_{ij}^l = \begin{cases} 1 & \text{if } (i,j) \in \mathcal{P}_l, \\ 0 & \text{otherwise} \end{cases}$$

where $\mathcal{P}_l$ is the path taken by the corresponding description. Also a set S which contains all the intermediate recovery nodes and the source is defined as follows:

$$S = \left\{ i : \sum_{l;j:(i,j)\in E} x_{ij}^l = 2 \right\}$$

Any path in the network can be characterized using the introduced variables. In this section we assume that the processing delay at every node is negligible. In practice, the time taken by both the descriptions to reach the recovery node is different. Also, there may be difference in path lengths of the two end-to-end routes. This means that the description that has arrived earlier has to wait for the other description to arrive so that the process of recovery can begin. Therefore, the total source s to destination t end-to-end delay is:

$$\text{Delay} = \sum_{k=1}^{N+1} \max \left\{ \sum_{(i,j)\in H_k^1} c_{ij}, \sum_{(i,j)\in H_k^2} c_{ij} \right\}$$

The path construction optimization problem OPT-DIST which minimizes distortion subject

to a delay constraint is formulated as follows:

$$\text{Minimize: } \mathcal{D} = \underline{DP'}$$

$$\text{s.t. } \sum_{j:(i,j)\in E} x_{ij}^l - \sum_{j:(i,j)\in E} x_{ji}^l = \begin{cases} 1, & i = s, i \in V, l = 1,2 \\ -1, & i = t, i \in V, l = 1,2 \\ 0, & \text{otherwise} \end{cases}$$

$$\sum_{j:(i,j)\in E} x_{ij}^l = \begin{cases} \leq 1, & \text{if } i \neq t, i \in V, l = 1,2 \\ 0, & \text{if } i = t, i \in V, l = 1,2 \end{cases}$$

$$x_{ij}^1 x_{ij}^2 = 0, \forall (i,j) \in E$$

$$|S| = N + 1$$

$$x_{ij}^l \in \{0,1\}, \forall (i,j) \in E, l = 1,2$$

$$\text{Delay } < T$$

where $\underline{P'}$ is the transpose of $\underline{P}$. The first constraint defines the flow for each particular description at source, destination, and other nodes over each of the paths. The difference between outgoing and incoming flow for each description is negative at the destination and positive at the source. At all other nodes, whether they are recovery nodes or not, this difference is zero. The second constraint enforces that, at the destination there is no outgoing flow. It also makes sure that at all nodes other than the destination, the outgoing flow for each description is at most one. These two constraints are required to obtain simple paths. The third constraint ensures that the paths obtained are link-disjoint (since a link can be used by only one of the two descriptions at a time, the product of the flow variables over link is zero). Almost all multimedia applications related to video are delay sensitive. Hence the last constraint, which also accounts for the additional delay due to intermediate recovery. A corresponding optimization problem OPT-DELAY to minimize delay subject to a distortion constraint can be similarly constructed.

The objective function in the above optimization problem is highly complex function of the path parameters. In particular, the objective function in OPT-DIST is a ratio of higher

order exponentials in path variables with non-linear constraints. Also, a simpler version of OPT-DELAY without any constraint on distortion was proven to be NP-Complete [45].

We observe that, since one description has to wait for another to arrive before the intermediate recovery, path pairs of two different segments need not be node-disjoint – so that the delay and delay difference (i.e., waiting time) is reduced. Enforcing node- and link-disjointedness constraints otherwise add more delay and complexity to the optimization problem. Also, from Theorem 2, distortion increases with the number of hops. So, considering a homogeneous error rate along each link across the network, a path pair taken by the descriptions that minimizes the total number of hops would give a low distortion. To address the above concerns of constrained distortion/delay optimization in route construction, two heuristics are developed to minimize delay and distortion, respectively, without any constraints on the other parameter, with the expectation that minimizing one would tend to minimize the other.

3.1. Route construction heuristics: MIN-DIST and MIN-DELAY

Algorithm 1 captures the developed heuristic routing strategy MIN-DELAY. It first calculates the shortest pair of disjoint paths (in terms of edge cost) between all node pairs in the network. This information is passed to all the other nodes. A node thereafter iterates over all possible N-tuples (for N intermediate recovery nodes) to find the tuple that minimizes delay. Clearly, if the number of intermediate recovery nodes is increased, the iteration step becomes more time consuming. A similar algorithm MIN-DIST can be developed to minimize distortion with a small modification by making all the edge weights 1 (where path delay cost is now the hop count).

Theorem 4. *The MIN-DELAY algorithm takes $O(n^{N+2}) + nS(n, m)$ time to compute paths between all node pairs in a network, where m is the number of edges, n is the total number of nodes, N is the number of intermediate recovery nodes along a source-to-destination path, and $S(\cdot, \cdot)$ represents the complexity of a single run of the Dijkstra's algorithm.*

Proof. If a single run of Dijkstra's algorithm takes $S(n, m)$ time, then by [15] the first step of the algorithm takes $O(nS(n, m))$ time. However this step has to be done once for all

Algorithm 1 Routing Strategy: MIN-DELAY

H ▷ It is the array containing the shortest pair of disjoint paths from every node to every other node in G

procedure FINDPATH(G, s, t, N)

$minDelay \leftarrow -1$

$path \leftarrow NULL$

for all $(v_1, .., v_N) \in V^N, v_i \neq v_j, \forall i \neq j, v_i \neq s, t$ **do**

$currPath \leftarrow \{H[s][v_1], H[v_1][v_2], \cdots, H[v_N][t]\}$

$currDelay \leftarrow Delay(currPath)$

if $minDelay == -1 || currDelay \leq minDelay$ **then**

$path \leftarrow currPath$

$minDelay \leftarrow currDelay$

end if

end for

return $path$

end procedure

nodes periodically. The iteration over all intermediate recovery nodes takes $O(n^N)$ time for a particular source destination pair. For all possible pairs, the iteration takes $O(n^{N+2})$ time. □

It may be noted that, the above routing technique is not scalable, as it iterates over all possible combinations of intermediate recovery nodes. With a prior knowledge of the ideal locations of these nodes, the complexity could be reduced significantly. From Theorem 3 and Corollary 1, the ideal positions of recovery nodes are known. The first term of the complexity can be reduced by searching around those positions. Then, only the second term dominates.

3.2. Performance of routing heuristics

The MIN-DIST heuristic was implemented in C++ and tested on a network with different number of nodes that were uniformly randomly distributed in a 100×100 square units area. Nodal coverage range was taken 15 units. The routing was tested for all nodes that are at

least 30 units apart. The network was assumed to have homogeneous link error rate, with the same Gilbert channel parameters $f_d T_s = 0.0004128$ and $F = 25$ dB. The description error threshold was taken to be 1%. The link weights (delay) were uniformly randomly distributed between 1 and 5. Fig. 7(a) shows an example of constructed traditional disjoint MPT between a source-destination pair. For the same node pair, the constructed multipaths with one intermediate recovery stage is shown in Fig. 7(b).

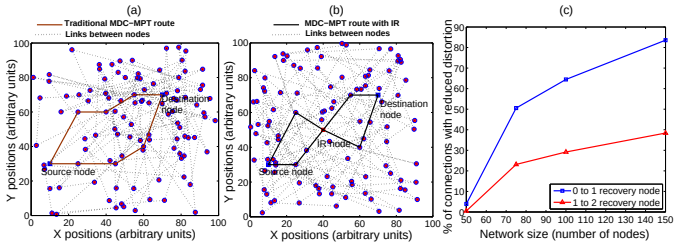

Figure 7: Example MDC-MPT scenarios and node density dependence of intermediate recovery performance.

Figure 7(c) shows the percentage of connections achieving a reduced distortion with intermediate recovery. Note that, the percentage of path pairs achieving a reduced distortion with more intermediate recovery nodes is low. In fact, we observed that when the intermediate recovery nodes is increased from 2 to 3, the additional reduction of distortion was not appreciable at all. This is because, with the increased number of intermediate recovery nodes in a randomly deployed network, the edge-disjoint paths between them gets longer. With more wireless edges along the path, the gain of intermediate recovery is nulled by higher transmission errors. Also, at low node density the performance is poor, because the routing strategy fails to find suitable edge disjoint paths to reduce distortion. Testing for globally optimum paths to check the efficiency of our routing scheme is not feasible due to its computational complexity.

Table 1: Performance gain with one intermediate recovery stage.

Number of nodes	50	75	100
PSNR gain	0.79	0.80	0.72

Table 1 shows the PSNR gain with a single intermediate recovery stage relative to a

system having no intermediate recovery. At any recovery stage, if the number of bit errors in a description is greater than τ, it is considered corrupted and discarded. It may be noted that, while the PSNR gain is apparently low, 0.5 dB gain in PSNR is discernible in multimedia applications [46]. By application of better routing strategies that can find paths of nearly equal length as that of a traditional MDC-MPT system, the performance is expected to increase further. More on multipath route construction will be discussed in Sections 4 and 5.

3.3. Effect of randomness on the heuristic algorithms

Intuitively, minimizing the distortion (equivalently, minimizing total path length) should also minimize the delay. So, it is expected that the path obtained from MIN-DIST heuristic will have a reasonably small delay. To test this hypothesis, a *similarity index* is defined.

Definition 2. *Similarity index in a random network is defined as the percentage of various source-destination connections having delay less than or equal to a given delay threshold.*

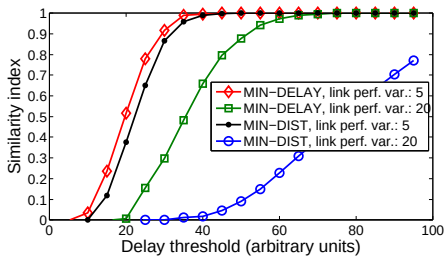

Figure 8: Similarity index comparison under different network conditions.

Figure 8 shows the similarity index with different variances of link weights of a 150 node network deployed in 100×100 square units area, where variance in link weight is an indicator of the difference in link quality due to different inter-nodal distance and/or channel fading related inhomogeneous link conditions. When the variance is low, minimizing distortion essentially minimizes delay. However, when the variance is high, there is a huge gap in the similarity indices. Hence, in networks that are more random in link performance characteristics, minimizing either of distortion/delay does not guarantee that the other variable is

bounded. As a result, in a random network with widely varying link qualities, if, for example, MIN-DIST heuristic is used to construct multiple routes, the end node would receive video frames of good resolution, but the overall video reception quality may not be acceptable due to high packet delivery delay and delay jitter. This observation calls for alternative improved solutions for joint guarantee of delay and distortion bounds. GA based search techniques are explored in the next section in response to this problem.

4. Genetic Algorithms (GA) based Suboptimal Mesh Routing Strategy

GA are meta-heuristic techniques that mimic the process of natural selection to solve discrete optimization problems. GA is particularly interesting in our current problem because they work well when the search domain's size and/or complexity is too high for an analytic solution.

Formulation of solution space and defining the nature of genetic operators is the most challenging part in GA. In this section we explain how the representation is realized in constructing meshed multipath, with one intermediate recovery node only, for low distortion and delay.

Genome: For 2D MDC-MPT route construction with one intermediate recovery node, each genome contains genetic information in the form of four lists, one corresponding to each of the four path segments: subpaths between source and intermediate recovery of the disjoint path pair ($Path_{11}$ and $Path_{21}$, respectively), subpaths between intermediate recovery node and destination in the disjoint path pair ($Path_{12}$ and $Path_{22}$, respectively).

Mutation: Mutation of a genome is performed with probability of p_m which is usually kept low. It (cf. Algorithm 2) involves six steps: (i) one of the four subpaths of the genome is randomly selected; (ii) one of the links in the subpath, e, is randomly selected and deleted from the graph; (iii) all links in the other disjoint subpath of the current path are deleted from the graph (e.g., if $Path_{11}$ was the chosen subpath, all the links from $Path_{12}$ are deleted); (iv) shortest path between the first node of e to the last node of the selected subpath is computed using Dijkstra's algorithm; (v) this computed path is appended in place of the

portion between e and last node in the original chosen subpath; (vi) all the deleted links are replaced in the original graph.

Crossover: This operation (cf. Algorithm 3) is performed with probability p_c and in three steps: (i) one subpath from each of the parent genes is chosen at random; (ii) these subpaths are traversed to find two common nodes, if any; (iii) the path portion between these two common nodes is interchanged to obtain the two children.

Objective functions: To effect OPT-DIST, fitness of a genome can be measured as the distortion corresponding to a path pair when the delay constraint is met, and a high penalty if the delay constraint is not met. A similar fitness function for OPT-DELAY can be defined. However, in cases where there is no path satisfying the requirement, the result is random. To avoid such a situation, a composite of delay and distortion of the corresponding genome, say, $A(Delay) + B(Distortion)$, or $Delay^a Distortion^b$, etc., can be used, where A, B, a, and b are some constants. In the current study, we consider a product-form objective function: $Delay.Distortion$.

4.1. Genetic algorithm

The GA adopted in the problem in hand uses overlapping populations, i.e., a pre-specified portion of the population is replaced at every generation as opposed to the traditional Goldberg algorithm. It also is elitist in approach wherein the best individual of a generation is carried over to the next generation. In each generation the algorithm creates a temporary population of individuals, adds these to the previous population, then removes the worst individuals in order to return the population to its original size. So the new offspring may or may not make it into the population, depending on whether they are better than the worst in the population.

4.2. Complexity analysis

Theorem 5. *Complexity of mutation operator is $S(n, m)$, the complexity of a single Dijkstra's computation.*

Proof. The dominating operator in the Algorithm 2 is computation of shortest paths in a reduced graph G with say m' edges. Since $m < m'$, $S(n, m') \subset S(n, m)$. □

Theorem 6. *Complexity of crossover operator is $O(N_1 + N_2)$, where N_1 and N_2 are the number of edges in subpaths chosen for cross over.*

Proof. The dominating operator in crossover is finding two common nodes in the subpaths. Using hash tables, this can be achieved in $O(N_1 + N_2)$ time. □

Theorem 7. *Finding a path pair using the GA structure explained above takes approximately $O\left(n^2\mathcal{M}\right)$, where $\mathcal{M}$ is the maximum of number of nodes in subpaths involved in crossovers.*

Proof. Let, the size of initial population be $\mathcal{N}$, the number of generations be $\mathcal{G}$, and the replacement ratio be p_r. So, in a generation, the number of mutations $= p_m\mathcal{N}$; the number of crossovers $= p_c p_r \binom{\mathcal{N}}{2}$. Then, the time complexity of GA $= \mathcal{G}\left[p_m\mathcal{N}S(n, m) + p_c p_r \binom{\mathcal{N}}{2} O(N_1 + N_2)\right]$. Since $\mathcal{G}, p_c, p_r$ are constants for steady state GA, and p_m is a small number that can be ignored,

$$\text{Time complexity of GA} = \binom{\mathcal{N}}{2} O(N_1 + N_2)$$

Since, $N_1, N_2 < n$, $O(N_1 + N_2) \subset O(\mathcal{M})$. Assuming the intermediate recovery nodes are equally represented in the initial population, $\mathcal{N} = O(n)$. Hence, the total complexity is $O\left(n^2\mathcal{M}\right)$. □

It may be noted that, the complexity of finding path pair between all node pairs is $O\left(n^4\mathcal{M}\right)$, which is much higher than the estimated complexity in GA-based approach. However, the complexity estimate in Theorem 7 is an overestimate because p_c and p_r are small quantities. This complexity, is significantly higher than the complexity of the heuristics.

4.3. GA implementation results

A GA-based MDC-MPT route construction for intermediate recovery was implemented in C++ using *GALib* (http://lancet.mit.edu/ga/) – a library for genetic algorithms. This routing strategy was tested on networks with similar parameters as in Section 3.2. The number of generations considered was 75, where it was observed that in almost all cases

the algorithm converged in 10-20 iterations. This is also evident from Fig. 9. The link delay variance was fixed at 10 units. Delay and distortion were computed from one node to all other nodes of the network and then averaged. The results clearly demonstrate that, iteration-wise (generation-wise) GA converges quite sharply.

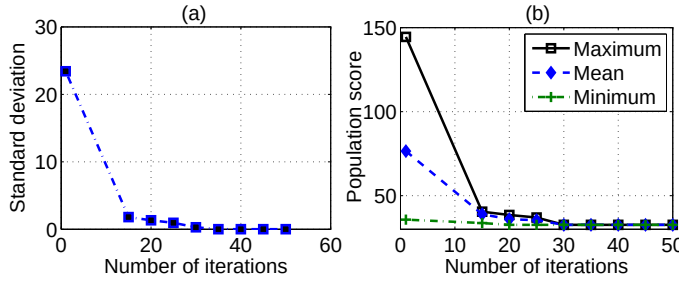

Figure 9: The impact of increasing number of GA iterations (generations) on the values of: (a) Standard deviation within a population; (b) Maximum, minimum, and mean value of population score

Table 2 compares different routing strategies on a network of size 150 nodes in a 100×100 square units area. Clearly, MIN-DIST gives paths of high delay and low distortion, whereas

Table 2: Performance of different routing strategies.

Routing strategy	**Delay**	**MSE distortion**
MIN-DIST	34.8112	32.1909
MIN-DELAY	18.119	34.535
GA	21.9597	32.7261

MIN-DELAY heuristic offers the other extreme, i.e., low delay but high distortion. Using either one of them creates problems for the end user. However, GA provides a path that is an intermediary between that of MIN-DIST and MIN-DELAY both in terms of delay and distortion.

Figure 10 shows delay and distortion of different routing strategies implemented on networks of sizes 50, 75, 100, and 150 nodes. The routing strategies were tested for communications from a single specific node located at the center of the network area, i.e., at (50, 50) to all other randomly located nodes, and the delay and distortion were averaged. It is clearly observed that, GA-based joint distortion and delay optimized routing constructs

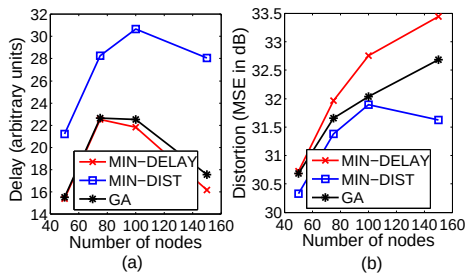

Figure 10: Variation of delay and distortion with node density in different routing strategies: (a) delay; (b) distortion.

meshed path that offers intermediate distortion and delay performance compared to that in MIN-DIST and MIN-DELAY. Furthermore, for small node densities GA provides significant improvement in delay with a marginal distortion tradeoff, which demonstrates utility of the heuristic approach. An additional observation is concavity of delay and the associated distortion measure, which can be explained in context of the overheads in mesh networks. Due to frequent changes in network topology and increased communication overhead, there may be delay or even drop in data packets. Additionally, transmission and queueing delays affect the overall end-to-end delay [47]. When the network is less dense, the route construction complexity is less, and so is queueing delay due to less interfering (highly dispersed) neighbors. As the node density increases, the end-to-end delay increases. However, beyond a point the alternatives for better routes (lower transmission and queueing delay) arise, thereby reducing the effective delay in the network due to close proximity of nodes and multiplicity of possible paths between the end nodes.

5. System Performance Results

In this section we study the performance of MDC-MPT system with intermediate recovery. It is assumed that the paths taken by the descriptions are constructed beforehand.

5.1. Numerical results

In the numerical results, we assume that the path taken by the descriptions has 33 intermediate nodes and homogeneous link errors. In the default system with carrier frequency

= 1850 MHZ, each wireless link is considered to have a Doppler spread and symbol time product $f_d T_s = 0.0004128$. This corresponds to a system with data rate = 256 Kbps and mobile velocity = 10 kmph. The MDTC transform matrix used is $T = [\alpha, 1/2\alpha; -\alpha, 1/2\alpha]$, with $\alpha = 1.2$, corresponding to 10% MDTC redundancy. The ratio of variances of the descriptions is 453, which is computed from the Lena image with the chosen transform matrix. The impact of MDTC transform coefficient, α, on PSNR in MDC-MPT system with and without intermediate recovery is shown in Fig. 11.

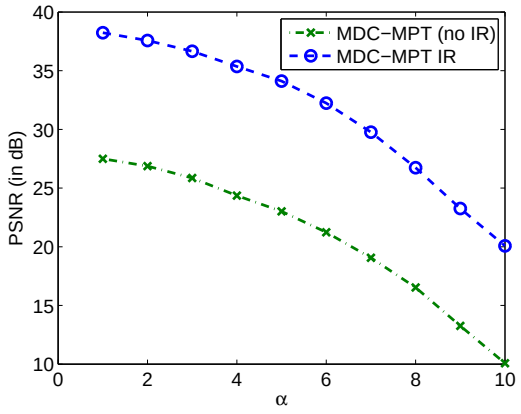

Figure 11: Impact of MDTC coefficient α on PSNR for MDC-MPT system with and without intermediate recovery (IR).

Figure 12 shows the average MSE distortion in (2) (plotted as $10\log_{10}(\text{MSE})$) versus number of intermediate recovery nodes with increasing MDTC redundancy [6] in two different wireless systems. System 1 is the default one, and System 2 is the one with a data rate = 54 Mbps, carrier frequency = 2.4 GHz, mobility speed = 10 kmph. The error threshold τ was taken to be 5% (50 bits in a total of 1000 bits). The intermediate recovery nodes were placed equidistant from each other. The case of zero intermediate recovery nodes corresponds to the traditional MDC-MPT. Compared to the traditional MDC-MPT, one intermediate recovery stage offers a gain about 9.2%. The gain however reduces as the number of recovery stages increases. For example, with two intermediate recovery stages, the additional gain is about 6%. System 2 has a higher distortion due to higher description losses at a higher data rate. Additionally, the systems' distortion reduces as MDTC redundancy increases redundancy.

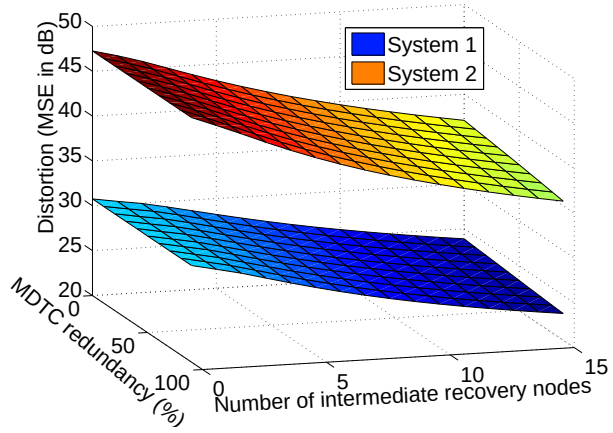

Figure 12: Variation of average MSE distortion with the number of intermediate recovery nodes and MDTC redundancy.

Figure 13(a) shows the average distortion performance versus fading margin. The error threshold is 2.5% (25 bits in a total of 1000 bits). Above fading margin 15 dB, MDC-MPT with intermediate recovery shows appreciably better performance over traditional MDC-MPT. At a small fading margin the channel remains in Bad state longer, hence the probability of losing both descriptions is high, thereby giving limited gain with intermediate recovery. Figure 13(b) shows that, for the same error threshold (2.5%), the effect of mobility on the

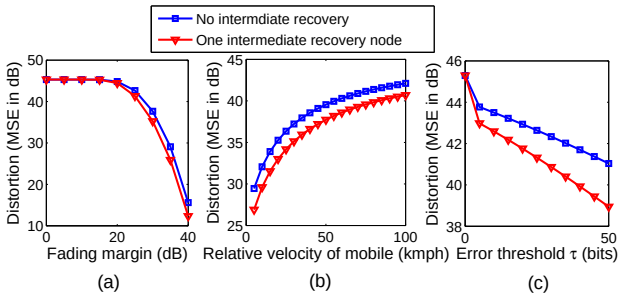

Figure 13: Comparison of MDC-MPT with intermediate recovery with traditional MDC-MPT: (a) at different fading margins; (b) at different relative mobility; (c) at different error thresholds τ (number of error correction bits).

intermediate recovery system is lower than that in traditional MDC-MPT. At higher mobility, Doppler spread increases, thereby increasing the chance of error. Hence the distortion at higher velocities is higher. Figure 13(c) on average distortion versus error threshold shows,

MDC-MPT with intermediate recovery performs better than the traditional one. Increasing error tolerance of the system decreases the distortion, because at a lower threshold the description is discarded sooner for a fewer number of bit errors.

5.2. Simulation settings and results

To test intermediate recovery performance on video delivery, we have implemented 2D MDC-MPT without/with intermediate recovery in NS2. A raw video (.yuv) file is temporally sub-sampled into two descriptions. The descriptions are correlated because of the natural temporal correlation in a video. The raw descriptions are then encoded to standard MPEG4 using FFMPEG framework (http://ffmpeg.org/). Using a multimedia packager MP4BOX (http://gpac.wp.mines-telecom.fr/mp4box/), these files are converted to standard compliant ISO media files. A hint track which contains information on packetizing the data is also added at this stage. Hinting helps the servers to stream without actually understanding the underlying architecture of the media files. Hinting sets the Maximum Transmission Unit (MTU) for transmission. Using Evalvid framework – a video quality evaluation tool-set (http://www.tkn.tu-berlin.de/menue/research/evalvid/) – a traffic trace file corresponding to actual data transmission, containing ID, video frame type (I, P, or B), packet size, and designated sending time is generated. Motivated by the fact that there may not be end-to-end bandwidth guarantee over the multihop networks, e.g., in MANETs, for fast delivery of video streaming content we consider UDP as the transport later protocol. By the in-build link layer ARQ and MCS, intra-packet recovery is attempted. Despite this recovery attempt, during the transient mobility phases, packets can be in error at the transport layer. MDC is used to help in recovery of these lost packets at the MDC decoding stage. UDP traffic similar to the video traffic trace is generated and sent to the destination over the wireless links in NS2. An 802.11 MAC (802.11p [48] for high mobility nodes) and wireless physical layer is set as the transmission medium [49]. To simulate channel fading in NS2, CMU's Ricean model is used.

We use NOAH (no ad hoc routing agent) protocol (http://icapeople.epfl.ch/widmer/uwb/ns-2/noah/) for routing purposes, and the routing is manually configured. This allows us to

study our considered routing strategies (MIN-DIST and MIN-DELAY) for MDC-MPT without/with intermediate recovery, instead of working with the default routing strategies in NS2.

Using a receiver trace that contains information on packet losses, a corrupted .mp4 file is generated and decoded to .yuv format using FFMPEG decoder. Error concealment is done as in standard MPEG decoding (if a frame is heavily corrupted, it is replaced by the correlated frame of other description) and the resultant descriptions are merged, as discussed in, [50]. The intermediate recovery is similar as the decoding process, with the difference that the descriptions after error concealment are transmitted further.

The intermediate recovery performance on video delivery is evaluated in terms of PSNR in dB. The studies in [51] showed that subjective video quality can be derived from PSNR. Further, it was shown in [52] that when a video is encoded at full frame rate and codec is fixed, then PSNR follows a monotonic relationship with subjective quality. In our study, the test video sequences are encoded at highest frame rate (30 fps), and the codec used is MPEG4. Hence, the PSNR values for the video delivery in our simulation study reflect the subjective video quality.

The default system parameters for NS2 simulation of 2D MDC video transmission are as follows. Akiyo, Foreman, and Soccer video sequences (with respective rates 4.01 Mbps, 2.95 Mbps, and 2.38 Mbps), each with resolution 352×288 (CIF); number of frame transmitted: 300; frame rate 30 fps; MTU size 1024 Bytes; MAC type 802.11 (802.11p [48] for high mobility scenarios); carrier frequency 5 GHz; channel bandwidth 10 MHz; maximum velocity: varied, 5 to 120 kmph; nodal transmit power: varied, 0.05 to 1.0 W; average internodal distance 150 m.

In a network with 75 nodes deployed in uniform random topology, maximum node velocity 50 kmph, and 0.2818 W transmit power (with 250 m communication range), the PSNR of Akiyo video with traditional MPT has been 18.26 dB, whereas the PSNR with one intermediate recovery stage is 33.18 dB. With the same network settings, the PSNR of Foreman video with traditional MPT is 17.56 dB, whereas with one intermediate recovery it is 32.38 dB. In Soccer video, the respective values are respectively 20.71 dB 38.47 dB. Two represen-

tative received frames for the three video sequences are shown in 14(a) and 14(b). Thus, a

Figure 14: MDC-MPT performance example: Two sample frames of Akiyo, Foreman, and Soccer sequences

significant improvement in video delivery quality is observed (14.92 dB, 14.78 dB, and 17.76 dB respectively, for the three representative video sequences) with intermediate recovery.

Figure 15(a) shows the reception quality (PSNR) of the Akiyo, Foreman, and Soccer sequences versus transmit power. Despite the fact that in both cases (traditional and intermediate recovery MDC-MPT) the number of errors reduces and PSNR improves with increased transmit power, the PSNR performance is significantly better with intermediate recovery at all transmit power, with an average PSNR gain of 15.58 dB in comparison to traditional MDC-MPT. This gain can be attributed to the fact that lost video descriptions are replaced with corresponding description from a neighboring frame (advantageous due to temporal correlation between frames), which leads to a higher probability of receiving the video descriptions at the destination.

We remark that, the intermediate recovery is an additional error concealment attempt besides a similar step taken at the destination. On the contrary, the traditional MDC-MPT scheme merely delivers the video packets over link-disjoint paths – without any such

description recovery anywhere along the route, and thus it attempts the error concealment only at the destination.

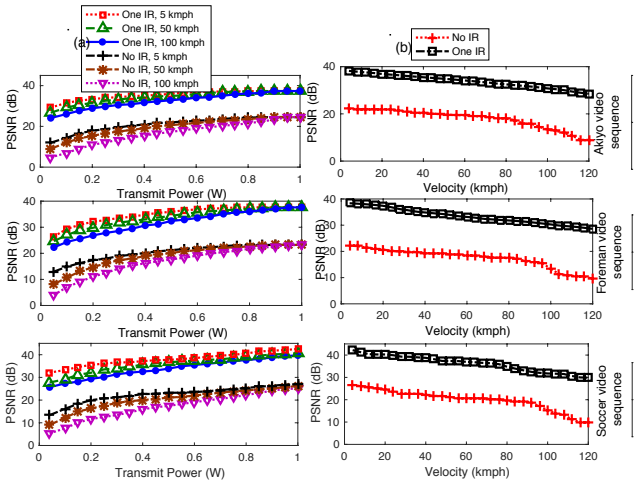

Figure 15: PSNR of Akiyo , Foreman, and Soccer sequences plotted against (a) transmit power; (b) maximum node velocity, at transmit power 0.2818 W. Random network with 75 nodes. IR: intermediate recovery

Figure 15(b) shows PSNR versus maximum velocity, representing the fast fading effect. Intuitively, the PSNR decreases with increased velocity. Even though at low velocities fading does not significantly affect the transmission, MDC-MPT with intermediate recovery outperforms the traditional MDC-MPT. At higher velocities, when there are many transmission errors, performance with intermediate recovery degrades at a slower rate compared to the traditional MDC-MPT.

PSNR of the video frames is shown for Akiyo video sequence in Fig. 16(a), Foreman sequence in Fig. 16(b), and Soccer sequence in Fig. 16(c). It can be observed that MDC-

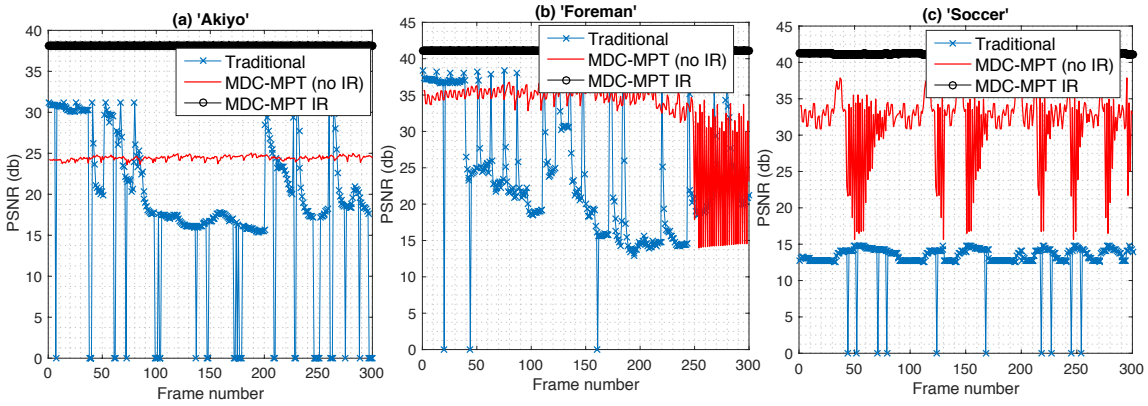

Figure 16: PSNR of video frames in (a) Akiyo, (b) Foreman, and (c) Soccer sequences for a traditional, MDC-MPT (no intermediate recovery (IR)), and MDC-MPT with IR system in a network of randomly distributed 75 nodes

MPT with intermediate recovery ensures an overall improvement of PSNR on average by 46.72% as compared to MDC-MPT without intermediate recovery and 59.94% as compared to a traditional (no MPT and no intermediate recovery) system. Hence, this result clearly reflects the merit of intermediate recovery in terms of improving the video streaming quality in MANETs.

Table 3 shows the average PSNR values and mean opinion score (MOS) (from 5 simulation iterations) for transmission from a node at the center (50, 50) in a random deployment scenario. The average increase in decodable frames count and the average decrease in packet loss at the destination node with intermediate recovery as compared to the traditional MDC-MPT schemes are also given in the table.

Table 3: Average PSNR and MOS values of MDC-MPT without and with intermediate recovery (IR), and average increase in decodable frames and decrease in packet loss due to IR in random network

Node density	MDC-MPT, no IR (PSNR in dB)			MDC-MPT, no IR (MOS)			Increased decodable frames with IR (%)	Decreased packet loss with IR (%)	MDC-MPT with IR (PSNR in dB)			MDC-MPT with IR (MOS)		
	Akiyo	Foreman	Soccer	Akiyo	Foreman	Soccer			Akiyo	Foreman	Soccer	Akiyo	Foreman	Soccer
75	18.06	17.09	19.57	2.60	2.51	2.74	67.32	79.27	33.39	32.82	37.68	3.91	3.52	4.03
100	18.78	18.58	20.43	2.74	2.61	2.83	75.79	82.22	34.71	34.47	39.61	3.98	3.79	4.14
150	19.17	19.04	21.18	2.78	2.72	2.87	81.54	91.19	35.76	35.01	41.05	4.11	3.91	4.28

Increased node density results in a higher PSNR gain and improved subjective quality (higher MOS), which is due to the possibility of finding a better route with more number of nodes deployed in the network. A significant gain in PSNR is apparent with intermediate recovery. It is also noted from Table 3 that, compared to traditional MDC-MPT, on average intermediate recovery results in 74.85% more decodable frames, and an average decrease in packet loss by 84.23%. Since standard MPEG coding uses much complex and efficient error correcting scheme (e.g., it exploits spatial/temporal correlations) as compared to a simple model in our numerical results, the benefit of intermediate recovery can be better exploited in practical systems.

6. Conclusion

In this work, MDC-MPT system performance for image/video content delivery was investigated by optimal meshed multipath route construction and intermediate recovery in wireless mesh networks. Via analytic modeling it was showed that intermediate recovery achieves good performance gains in comparison with the traditional multipath system. A

cross-layer optimization problem was formulated to construct meshed multipath routes with optimized end-to-end distortion and delay. To reduce complexity of constrained optimization, a simple heuristic strategy was proposed that constructs good quality (distortion or delay minimizing) paths for intermediate recovery. Although the heuristic strategy works well in networks with relatively homogeneous link performance, the two minimization criteria (distortion, delay) are not jointly achievable if the link costs vary widely. For this joint objective, GA-based multipath construction strategy was employed, which showed a good delay-distortion tradeoff performance on content delivery. Further, NS2 based MDC-MPT system implementation results demonstrated that intermediate recovery technique indeed achieves a considerable improvement in video reception quality.

The proposed MDC-MPT with intermediate recovery can be applied to disaster recovery situation or can be considered with different technologies along different paths, which will be considered as a future extension.

Acknowledgment

This was supported by the DST grant nos. SB/S3/EECE/0248/2014 and DST/INSPIRE/04/2015/0007 and the IIT Kharagpur ISIRD grant no. IIT/SRIC/ECE/USW/2016-17.

References

[1] S. De, C. Qiao, H. Wu, Meshed multipath routing with selective forwarding: An efficient strategy in wireless sensor networks, Elsevier Computer Networks 43 (4) (2003) 481–497.

[2] J. G. Apostolopoulos, M. D. Trott, Path diversity for enhanced media streaming, IEEE Commun. Mag. 42 (8). doi:10.1109/MCOM.2004.1321395.

[3] S. De, C. Qiao, A hybrid mesh multipath forwarding scheme in wireless ad hoc networks, Elsevier Computer Communications 30 (17) (2007) 3346–3357.

[4] S. Mao, S. Lin, S. Panwar, Y. Wang, E. Celebi, Video transport over ad hoc networks:

multistream coding with multipath transport, IEEE J. Sel. Areas Commun. 21 (10) (2003) 1721–1737. doi:10.1109/JSAC.2003.815965.

[5] M. Kazemi, S. Shirmohammadi, K. Sadeghi, A review of multiple description coding techniques for error-resilient video delivery, Springer Multimedia Systems (2013) 1–27doi:10.1007/s00530-013-0319-z.

[6] A. Sharma, S. De, H. M. Gupta, R. Gangopadhyay, Multiple description transform coded transmission over OFDM broadcast channels, Elsevier Physical Commun. 12 (2014) 79–92.

[7] S. Mao, Y. Hou, X. Cheng, H. Sherali, S. Midkiff, Y.-Q. Zhang, On routing for multiple description video over wireless ad hoc networks, IEEE Trans. Multimedia 8 (5) (2006) 1063–1074. doi:10.1109/TMM.2006.879845.

[8] S. Kompella, S. Mao, Y. T. Hou, H. D. Sherali, Cross-layer optimized multipath routing for video communications in wireless networks, IEEE J. Sel. Areas Commun. 25 (4) (2007) 831–840.

[9] I. Nemoianu, C. Greco, M. Cagnazzo, B. Pesquet-Popescu, A framework for joint multiple description coding and network coding over wireless ad-hoc networks, in: Proc. IEEE ICASSP, Kyoto, Japan, 2012, pp. 2309–2312.

[10] A. Begen, Y. Altunbasak, O. Ergun, Multi-path selection for multiple description encoded video streaming, in: Proc. IEEE ICC, Anchorage, AK, USA, 2003. doi:10.1109/ICC.2003.1203869.

[11] L. Zhou, X. Wang, W. Tu, G.-M. Muntean, B. Geller, Distributed scheduling scheme for videostreaming over multi-channel multi-radio multi-hop wireless networks, IEEE J. Sel. Areas Commun. 28 (3) (2010) 409–419.

[12] B. Rong, Y. Qian, K. Lu, R. Q. Hu, M. Kadoch, Multipath routing over wireless mesh networks for multiple description video transmission, IEEE J. Sel. Areas Commun. 28 (3) (2010) 321–331.

[13] J. Yi, E. Cizeron, S. Hamma, B. Parrein, P. Lesage, Implementation of multipath and multiple description coding in OLSR, in: Proc. 4th OLSR Introp./Wksp., Ottawa, Canada, 2009.

[14] J. W. Suurballe, Disjoint paths in a network, Networks 4 (2) (1974) 125–145.

[15] J. W. Suurballe, R. E. Tarjan, A quick method for finding shortest pairs of disjoint paths, Networks 14 (2) (1984) 325–336.

[16] S. Banerjee, R. K. Ghosh, A. P. K. Reddy, Parallel algorithm for shortest pairs of edge-disjoint paths, J. Parallel Distrib. Comput. 33 (1996) 165–171.

[17] R. Ogier, N. Shacham, A distributed algorithm for finding shortest pairs of disjoint paths, in: Proc. IEEE INFOCOM, Ottawa, Canada, 1989. doi:10.1109/INFCOM.1989.101450.

[18] G. A. Shah, W. Liang, O. B. Akan, Cross-layer framework for qos support in wireless multimedia sensor networks, IEEE Trans. Multimedia 14 (5) (2012) 1442–1455.

[19] S. Nazir, V. Stankovic, H. Attar, L. Stankovic, S. Cheng, Relay-assisted rateless layered multiple description video delivery, IEEE J. Sel. Areas Commun. 31 (8) (2013) 1629–1637.

[20] Z. Bai, S. Li, Y. Wu, W. Zhou, Z. Zhu, Experimental demonstration of SVC video streaming using QoS-aware multi-path routing over integrated services routers, in: Proc. IEEE ICC, Budapest, Hungary, 2013.

[21] X. Sun, H. Chen, X. Xu, X. Yin, W. Song, Opportunistic communications based on distributed width-controllable braided multipath routing in wireless sensor networks, Elsevier Ad Hoc Networks 36 (1) (2016) 349–367.

[22] S. Shabdanov, P. Mitran, C. Rosenberg, Achieving optimal throughput in cooperative wireless multihop networks with rate adaptation and continuous power control, IEEE Trans. Wireless Commun. 13 (7) (2014) 3880–3891.

[23] A. Newell, H. Yao, A. Ryker, T. Ho, C. Nita-Rotaru, Node-capture resilient key establishment in sensor networks: Design space and new protocols, ACM Computing Surveys 47 (2) (2015) 24:1–24:34.

[24] L. Zhou, B. Geller, B. Zhang, A. Wei, J. Cui, System scheduling for multi-description videostreaming over wireless multi-hop networks, IEEE Trans. Broadcasting 55 (4) (2009) 731–741.

[25] X.-L. Zhu, G.-Q. Zhang, X.-J. Song, Y. Hu, X.-Y. Xiang, Research on mdc transmission over wireless mesh network, in: Proc. Intl. Conf. Future Computer and Commun. (ICFCC), Wuhan, China, 2010.

[26] G. Dan, V. Fodor, G. Karlsson, Are multiple descriptions better than one?, in: Prof. IFIP Networking, Waterloo, Ontario, Canada, 2005, pp. 684–696.

[27] H. Cui, D. Qian, X. Zhang, Y. Liu, Video streaming over wireless mesh networks with multi-gateway support, in: Proc. IEEE/IFIP EUC, Hong Kong, 2010, pp. 268–272.

[28] M. Qin, R. Zimmermann, An adaptive strategy for mobile ad hoc media streaming, IEEE Trans. Multimedia 12 (4) (2010) 317–329.

[29] P. Correia, P. Assuncao, V. Silva, Enhanced H.264/AVC video streaming using network-adaptive multiple description coding, in: Proc. IEEE EUROCON, Lisbon, Portugal, 2011, pp. 1–4.

[30] A. El Essaili, S. Khan, W. Kellerer, E. Steinbach, Multiple description video transcoding, in: Proc. IEEE ICIP, Vol. 6, San Antonio, Texas, 2007, pp. 77–80.

[31] R. Elbaum, M. Sidi, Topological design of local-area networks using genetic algorithms, IEEE/ACM Trans. Networking 4 (5) (1996) 766–778.

[32] E. Gelenbe, P. Liu, J. Laine, Genetic algorithms for route discovery, IEEE Trans. Systems, Man, and Cybernetics 36 (6) (2006) 1247–1254.

[33] R. Leela, S. Selvakumar, Genetic algorithm approach to dynamic multi constraint multi path QoS routing algorithm for IP networks (GA-DMCMPRA), in: Proc. COMSNETS, Bangalore, India, 2009, pp. 1–6.

[34] Y. Leung, G. Li, Z.-B. Xu, A genetic algorithm for the multiple destination routing problems, IEEE Trans. Evolutionary Computation 2 (4) (1998) 150–161.

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

[36] B. Sun, L. Li, A QoS multicast routing optimization algorithm based on genetic algorithm, J. Commun. and Networks 8 (1) (2006) 116–122.

[37] M. U. Parthavi, S. De, Mesh routing for error resilient delivery of multiple-description coded image/video content, in: Proc. IEEE ICCCN, Munich, Germany, 2012.

[38] P. De, N. Banerjee, S. De, Exploiting multiple description coding for intermediate recovery in wireless mesh networks, Elsevier J. Network and Computer Applications 37 (2014) 2–11.

[39] R. Ahlswede, N. Cai, S.-Y. R. Li, R. W. Yeung, Network information flow, IEEE Trans. Inform. Theory 46 (4) (2000) 1204–1216.

[40] E. Setton, Y. Liang, B. Girod, Adaptive multiple description video streaming over multiple channels with active probing, in: Proc. Intl. Conf. Multimedia and Expo, Baltimore, MD, USA, 2003. doi:10.1109/ICME.2003.1220966.

[41] V. Goyal, J. Kovacevic, R. Arean, M. Vetterli, Multiple description transform coding of images, in: Proc. Intl. Conf. on Image Process., Chicago, IL, USA, 1998. doi:10.1109/ICIP.1998.723588.

[42] M. Zorzi, R. Rao, L. Milstein, ARQ error control for fading mobile radio channels, IEEE Trans. Veh. Technol. 46 (2) (1997) 445–455. doi:10.1109/25.580783.

[43] R. A. Howard, Dynamic Probabilistic Systems, Wiley, NY, 1971.

[44] Y. Lin, J.-H. Song, V. W. Wong, Cooperative protocols design for wireless ad-hoc networks with multi-hop routing, Mob. Netw. Appl. 14 (2) (2009) 143–153.

[45] C.-L. Li, S. McCormick, D. Simchi-Levi, Finding disjoint paths with different path-costs: Complexity and algorithms, Networks 22 (7) (1992) 653–667.

[46] A. Dua, C. W. Chan, N. Bambos, J. Apostolopoulos, Channel, deadline, and distortion (CD^2) aware scheduling for video streams over wireless, IEEE Trans. Wireless Commun. 9 (3) (2010) 1001–1011.

[47] H. Li, Y. Cheng, C. Zhou, W. Zhuang, Minimizing end-to-end delay: A novel routing metric for multi-radio wireless mesh networks, in: Proc. IEEE INFOCOM, Rio de Janeiro, Brazil, 2009.

[48] IEEE Standards Association, 802.11p, part 11: Wireless LAN medium access control (mac) and physical layer (phy) specifications amendment 6: Wireless access in vehicular environments., IEEE standard for information technology-local and metropolitan area networks-specific requirements (2010) 1–12.

[49] K. Jin, H. Weerasinghe, H. Fu, Enhancement of ieee 802.11 modules in ns-2 and performance evaluation with error rate, in: Proc. Spring Simulation Multiconference, San Diego, CA, USA, 2010, pp. 1–7.

[50] S. Aign, K. Fazel, Error concealment in MPEG video streams over ATM networks, IEEE J. Sel. Areas Commun. 18 (6) (2000) 1129–1144.

[51] J. Korhonen, J. You, Improving objective video quality assessment with content analysis, in: Proc. Int. Wksp. VPQM, Scottsdale, AZ, USA, 2010.

[52] Q. Huynh-Thu, M. Ghanbari, The accuracy of PSNR in predicting video quality for different video scenes and frame rates, Telecommun. Sys. 49 (1) (2012) 35–48.

Algorithm 2 Mutation

procedure MUTATE(Genome g)

$path \leftarrow Random(g.Path_{11}, g.Path_{12}, g.Path_{21}, g.Path_{22})$

$(deli, delj) \leftarrow Random(pathlength(path))$

$Delete(G, (deli, delj))$

for all $(i, j) \in Disj(path)$ **do**

$Delete(G, (i, j)$

end for

$apppath \leftarrow Dijkstra(deli, t)$

$path \leftarrow Trim(path, (deli, delj)) \cup apppath$

$Add(G, (deli, delj))$

for all $(i, j) \in Disj(path)$ **do**

$Add(G, (i, j))$

end for

end procedure

procedure RANDOM($Path_1, Path_2, Path_3, Path_4$)

Returns one of the arguments randomly

end procedure

procedure DISJ($path$)

Returns the disjoint subpath of the given description

end procedure

procedure DELETE($G, (i, j)$)

Deletes the (i, j) from the graph G

end procedure

procedure DIJKSTRA(i, t)

Returns the shortest path from i to t

end procedure

procedure TRIM($path, (deli, delj)$)

Deletes all links after $(deli, delj)$ from $path$

end procedure

procedure ADD($G, (i, j)$)

Adds (i, j) to the graph G

end procedure

Algorithm 3 Crossover

procedure CROSSOVER(Genome g_1, Genome g_2)

$path_1 \leftarrow Random(g_1.Path_{11}, g_1.Path_{12}, g_1.Path_{21}, g_1.Path_{22})$

$path_2 \leftarrow Random(g_2.Path_{11}, g_1.Path_{12}, g_1.Path_{21}, g_1.Path_{22})$

$[s_1, s_2] \leftarrow FindCommon(path_1, path_2)$

if $[sub_1, sub_2] \neq [NULL, NULL]$ **then**

$path_{1new} \leftarrow Subpath(path_1, path_1.src, s1) \cup Subpath(path_2, s_1, s_2) \cup Subpath(path_1, s_2, path_1.dst)$

$path_{2new} \leftarrow Subpath(path_2, path_2.src, s_2) \cup Subpath(path_1, sub_1, s_2) \cup Subpath(path_2, s_2, path_2.dst)$

$path_1 \leftarrow path_{1new}$

$path_2 \leftarrow path_{2new}$

end if

end procedure

procedure FINDCOMMON($path_1, path_2$)

Returns indices two common nodes in $path_1$ and $path_2$

end procedure

procedure SUBPATH($path, n_1, n_2$)

Returns the subpath of $path$ between n_1 and n_2

end procedure