Wireless Pers Commun (2011) 56:535–557
DOI 10.1007/s11277-010-9988-5

Enhancing the Performance of Video Streaming in Wireless Mesh Networks

Xiaoling Qiu · Haiping Liu · Dipak Ghosal · Biswanath Mukherjee · John Benko · Wei Li · Rashmi Bajaj

Published online: 21 April 2010

Abstract Multihop wireless mesh networks (WMNs) provide ubiquitous wireless access in a large area with less dependence on wired networks. However, some emerging applications with high bandwidth requirement and delay and loss constraints, such as video streaming, suffer poor performance in WMNs, since high compression rates and/or high packet loss rates deteriorate the video quality. In this paper, we propose a novel mechanism composed of (1) a network route selection scheme which provides paths for multiple video streams with the least interference, called Minimum Interference Route Selection (MIROSE) and (2) an optimization algorithm that determines the compression rates depending on the network

X. Qiu (✉) · D. Ghosal · B. Mukherjee
Computer Science Department, University of California, One Shields Avenue, Davis, CA 95616, USA
e-mail: xqiu@ucdavis.edu

D. Ghosal
e-mail: ghosal@cs.ucdavis.edu

B. Mukherjee
e-mail: mukherje@cs.ucdavis.edu

H. Liu
Department of Electrical and Computer Engineering, University of California,
One Shields Avenue, Davis, CA 95616, USA
e-mail: hpliu@ucdavis.edu

W. Li
School of Engineering and Computer Science, Victoria University of Wellington,
PO Box 600, Wellington 6140, New Zealand
e-mail: wei.li@ecs.vuw.ac.nz

J. Benko · R. Bajaj
France Telecom Research Lab, 801 Gateway Blvd, Suite 500, South San Francisco, CA 94080, USA

J. Benko
e-mail: john.benko@orange-ftgroup.com

R. Bajaj
e-mail: rashmi.bajaj@orange-ftgroup.com

condition, called Network State Dependent Video Compression Rate (NSDVCR) algorithm. Simulation results of the proposed mechanisms show the significant improvement of the video quality measured with a popular metric, Peak-Signal-to-Noise Ratio (PSNR), compared with standard routing and default compression rates.

Keywords Wireless mesh networks · Video streaming · Rate adaptation · Route selection · Simulation analysis

1 Introduction

Multihop wireless mesh networks can provide ubiquitous wireless access in a large area with less support from wired networks. However, current multihop mesh networks pose a number of challenging design problems which limit end-to-end network performance, such as throughput, delay, and congestion [1]. First, in a large mesh network, centralized MAC-layer scheduling and synchronization of link transmissions of the entire system is not practical. Most deployed mesh networks are based on Wi-Fi devices, which utilize the random access mechanism of 802.11. Therefore, frequent interference among neighbor nodes or links significantly deteriorates wireless transmissions in mesh networks. Second, a traffic flow over a wireless path of multiple hops suffers "in-flow" interference, i.e., the interference with the previous hop and/or the next hop [2]. So the transmission in a link experiences interference from other transmissions of the same flow in other links in the path. Third, dynamic traffic requests negatively influence the traffic balance among nodes in the mesh network, which could result in local congestion in a certain area.

Even with scarce bandwidth, mesh networks are capable of supporting "light-weight" applications that need low bandwidth and can tolerate high delay, such as emailing, instant messaging or web surfing. However, other "heavy-weight" applications that need high capacity and less delay, such as real-time video, often have poor performance. The "heavy-weight" flow introduces "in-flow" interference and consumes significant network resources, which could severely influence traffic distribution and may block other traffic in the same part of the network. For example, in Fig. 1, two clients send requests for real-time video to remote servers. Since two clients are located close to each other, the two video streams share almost the same path from the gateway to the destinations and severely contend for the limited resources. As a result neither flow may obtain the required bandwidth, and also interfere with other traffic in the area. This scenario is also validated in [3]. The authors conducted extensive testbed experiments of video streaming in multihop wireless mesh networks and concluded that "both inter- and intra- flow interference have a great impact on video quality. This is particularly true in wireless multihop scenarios."

"Heavy-weight" and "light-weight" applications are similar to the classic definition of "elephant" and "mouse" flows in network measurement. However, "heavy-weight" applications contain more constraints. In addition to the the large number of packets in "elephant" flows, "heavy-weight" applications also have delay and loss constraints. For example, if some packets of a real-time video do not arrive at the destination within a certain time, they are useless and can be dropped. Like "mouse" flows in the Internet, the number of "light-weight" flows in mesh networks is normally much larger than that of "heavy-weight" flows. However, "heavy-weight" applications usually badly influence other existing flows, break the load balancing and deteriorate the whole network performance. Therefore, to efficiently allocate the limited resource to different applications, we should concentrate more on "heavy-weight" applications rather than controlling all end-to-end flows or all applications, which is not

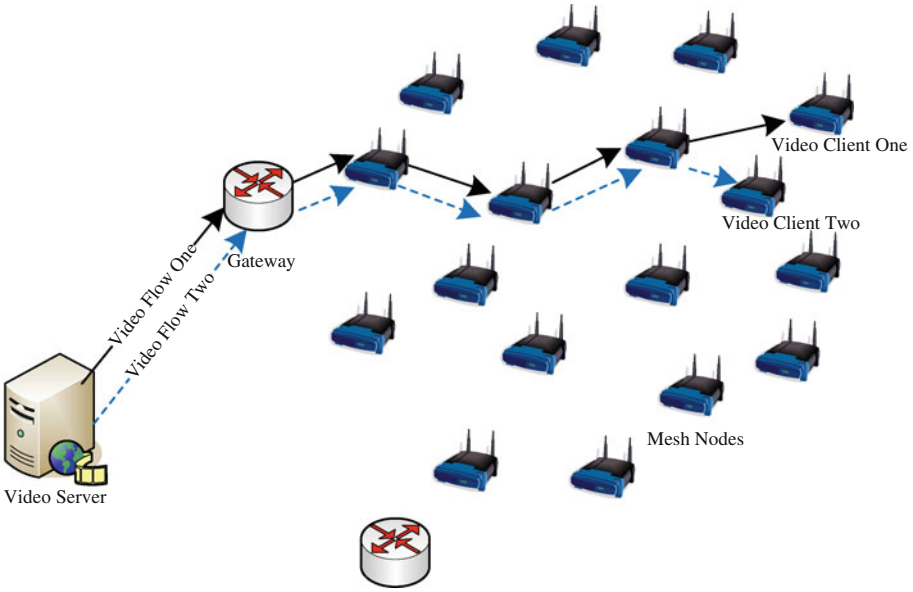

Fig. 1 Multiple video streams contend for limited resource in a WMN

practical and necessary. In our study, we consider "light-weight" applications as the background traffic, which are assumed to uniformly distribute in the whole network, and propose a novel mechanism to efficiently support transmission of "heavy-weight" applications in WMNs.

In this paper, we consider real-time video as the "heavy-weight" application. We develop a method to improve the performance of real-time video delivery taking into consideration all the negative attributes of a wireless mesh network.

The rest of the paper is organized as follows. We introduce the system model in Sect. 2. The minimum interference route selection algorithm and network state dependent video compression rate optimization algorithm [4] will be presented in Sect. 3 and 4, respectively. In Sect. 5, we discuss simulation results of both route selection and rate optimization for multiple video streams in wireless mesh networks. Finally, we present related work in Sect. 6 and present our conclusions in Sect. 7.

2 System Model

2.1 Introduction to Video Streaming

Uncompressed raw video contains spatial and temporal redundancy which make it inefficient when transmitting over networks or storing on media. Spatial redundancy refers to redundancy in a single frame which can be reduced by intraframe compression, similar to image compression. Temporal redundancy can be reduced by interframe compression including motion compensation. With temporal compression, only changes between consecutive frames are encoded so as to decrease bit rate of the compressed video. MPEG [5,6] and H.261 [7] uses temporal compression.

Raw video streams are compressed as a group of frames, called a group of pictures (GOP). A GOP contains one Intra coded frame (I frame), several Predicted frames (P frames), and maybe some Bi-directional predictive frames (B frames). An encoded video contains two GOP parameters n and m, where n is the distance between two I frames, also called the GOP length, and m is the distance between two anchor frames (I frame or P frame). For example, $(n, m) = (12, 3)$ indicates that the GOP consists of $IBBPBBPBBPBB$.

In a GOP, the first frame is the I frame which is encoded independent of other frames. I frames do not rely on temporal compression, and need more bits to transmit. P frames are based on previous I frames and P frames in the same GOP, which compensate I frames to improve video quality and achieve a lower bit rate. B frames are based on past and future I and P frames in the same GOP, but not other B frames.

After transmission, a frame can be successfully decoded at the client if all frames in the GOP have been correctly received and decoded. However, some frames may be lost. Concealment is used in decoding to compensate for lost frames. If the first frame of GOP, I frame, is lost, the codec can use the most recent correctly displayed frame from the previous GOP to compensate for the lost frame. Similarly, if P or B frames are lost, the lost frames and all the following frames in a GOP are replaced by the decoded I frame in this GOP. Note that losing different types of frames have different impact on the video quality. This is further discussed in Sect. 4.

2.2 Factors Influencing Video Quality

Through our experiments, we found that the following are the main factors that impact the video quality in wireless mesh networks:

- Network Condition: Depending on the number of hops and link quality of each hop, the video quality of clients with different geographical locations can vary significantly. The video quality may not be acceptable if the stream has to pass through a long and poor quality path.
- Routing: In mesh networks, the quality of multiple video streams can decrease if they contend for the same network resources as shown in Fig. 1. Furthermore, the network conditions get accordingly degraded due to the severe congestion caused by the video streams.
- Compression Rate: Since most video compression schemes, including MPEG-4, employ lossy compression, some information will be inevitably dropped. Therefore, the compression rate also influences the quality of the video at the receiver.

Among these factors, the network condition is dynamic but can be monitored and determined. Although it is not possible to predict the video requests generated by the clients and the link qualities are not under our control, different routing choices can potentially help avoid congestion in a certain parts of the network. For example, comparing different routing in Figs. 1 and 2, it is obvious that selecting diverse paths for different video streams can significantly improve their qualities and mitigate poor network conditions. In the network layer, our mechanism re-route multiple video flows if they share similar paths. We present our approach in Sect. 3.

In addition to route selection, it is also important to dynamically adjust the compression rate according to network conditions and routing information. Video compression rate is another factor that we can control. If a client is located far from the gateway or if a link quality in the path between a client and the gateways is poor, or if two separate video streams follow almost the same path from the gateway to the client, a high compression rate can

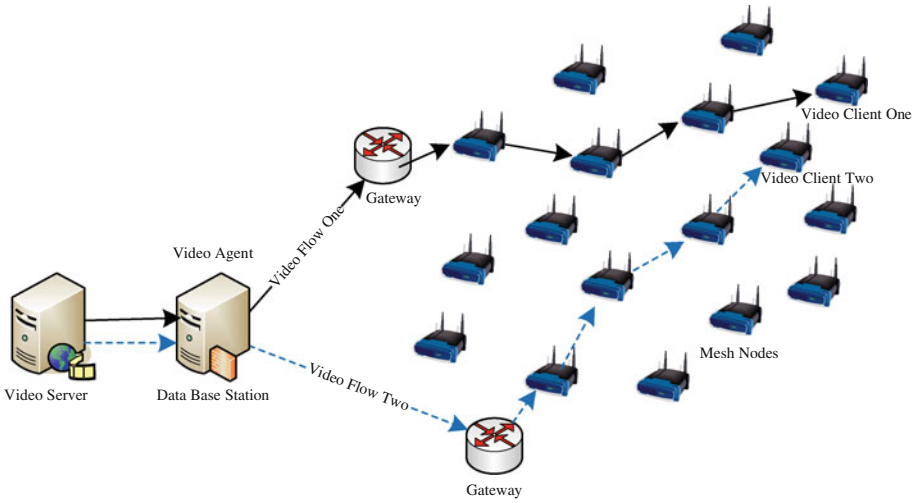

Fig. 2 Video agent in mesh networks

help reduce high packet loss rate. Based on extensive testbed experiments, the study in [3] also concluded that "In a multihop wireless mesh network, the tradeoff between streaming rate and video quality needs to be considered carefully. High rates do not necessarily bring high video quality." Therefore, there exists an optimal video compression rate for a given network conditions. We present our proposed dynamic video compression rate optimization algorithm in Sect. 4.

2.3 Network Proxy and Optimal Video Compression Rate

In order to support our proposed approach, we need some infrastructure support. In particular, we assume that there is a video agent deployed at the edge of the mesh network. Most current mesh networks, both for public or private access, are normally deployed and managed by a service provider which is responsible for the registration and authentication of the users. The mesh networks and the proposed video agent is shown in Fig. 2. The video agent is a logical entity and can be located in the base stations between the gateway and Internet. We assume that the video agent has enough buffer and computing power to implement the the dynamic compression algorithm proposed in this paper.

Since both the upstream and downstream traffic needs to travel through the base station, the video agent can improve the delivery of real-time video measured in terms of the PSNR as follows:

1. The video agent receives video requests from the clients, and chooses proper routes for each video flow which minimizes the overall path contention from the multiple video streams.
2. The video agent temporarily buffers the video content and adjusts the compression rates according to the condition of the path from the gateway to the requesting client.

Selecting a route and choosing the compression rate are correlated and jointly influence the performance objectives. However, it is intractable to search the optimal solutions for both problems simultaneously. We decompose these two problems and solve them independently in the next two sections. In the next section, we design a minimum interference route selec-

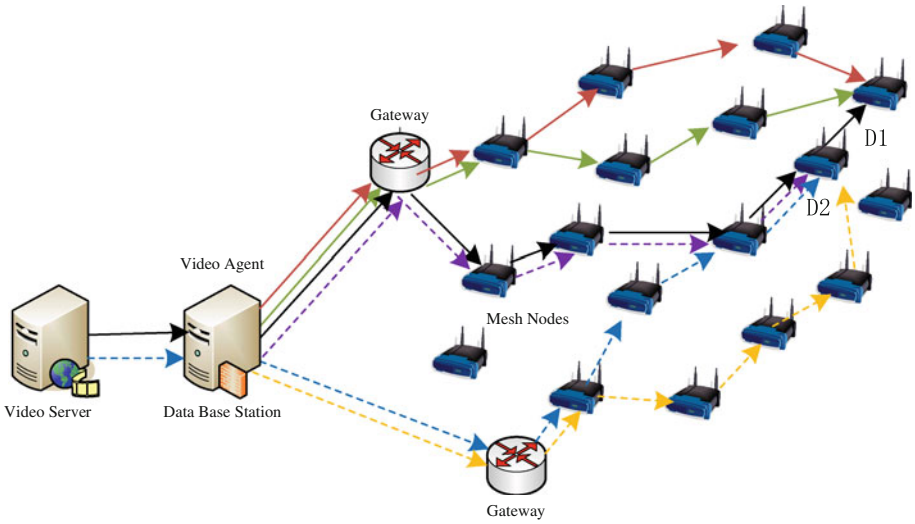

Fig. 3 Potential paths to video clients

tion algorithm (MIROSE) to select a set of least interference paths. Based on the network condition and video properties, we design a video compression rate optimization algorithm referred as NSDVCR and described in Sect. 4.

3 Minimum Interference Route Selection Algorithm (MIROSE)

3.1 Algorithm

A few currently popular wireless routing algorithms, such as DSR [8], usually provide several available paths from the video agent/gateway to the requesting client. Additionally, we can modify some other famous algorithms, such as AODV [9], to maintain multiple paths in addition to the chosen one. These potential paths, as shown in Fig. 3, provides the solution set for the best paths of all clients while reducing the mutual interference and negative influence on the whole network.

In order to quantify the interference relationship of different paths, we introduce the correlation function $C_2(p_1, p_2)$ for two routing paths p_1 and p_2. Suppose there are n hops/links along path p_1 denoted by $h_1, h_2, \ldots, h_n$ and m hops/links along path p_2 denoted by $l_1, l_2, \ldots, l_m$. With the interference index between two hops,

$$d_{k,l} = \begin{cases} 1 & \text{if hop } k \text{ interferences with hop } l \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

the correlation function $C_2(p_1, p_2)$ is defined as

$$C_2(p_1, p_2) = \sum_{k=1}^{n} \sum_{l=1}^{m} d_{k,l}. \tag{2}$$

$C_2(p_1, p_2)$ is the metric of the correlation of two paths p_1 and p_2, measured as the sum of the interference index values of any two hops from p_1 and p_2. Note that $d_{k,l} = d_{l,k}$, so

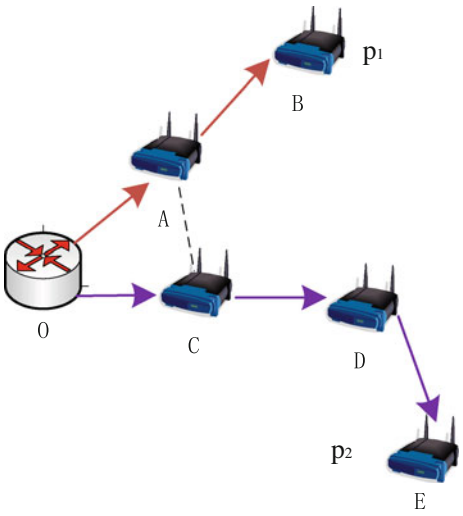

Fig. 4 Correlation function of two paths

$C_2(p_1, p_2) = C_2(p_2, p_1)$. Figure 4 gives a simple example of how to calculate $C_2(p_1, p_2)$. In Fig. 4, Path p_1 traverses from O, A and B; path p_2 traverses from O, C, D and E. The first and second hops of p_1 interfere with the first and second hops of p_2; therefore, $C_2(p_1, p_2) = C_2(p_2, p_1) = d_{p_1^1,p_2^1} + d_{p_1^1,p_2^2} + d_{p_1^1,p_2^3} + d_{p_1^2,p_2^1} + d_{p_1^2,p_2^2} + d_{p_1^2,p_2^3} = 1+1+0+1+1+0 = 4$.Here p_1^1 means the first hop of path p_1. Others have the similar meaning.

For a set of paths, $[p_1, p_2, \ldots, p_k]$, we introduce the interference function $G(p_1, p_2, \ldots, p_k)$ as

$$G(p_1, p_2, \ldots, p_k) = \sum_{i=1}^{k} \sum_{j=1, j\neq i}^{k} C_2(p_i, p_j). \tag{3}$$

Note that $G(p_1, p_2) = C_2(p_1, p_2) * 2$ since both interference of p_1 on p_2 and p_2 on p_1 is counted.

For multiple clients in the networks, D^1 to D^K, suppose there are multiple possible paths $\{D_1^i, \ldots, D_{N_i}^i\}$ for any client i. Then any choice $[D_{p_1}^1, \ldots, D_{p_K}^K]$ could be a possible set of paths for clients D^1 to D^K, where $D_{p_i}^i$ is chosen from $\{D_1^i, \ldots, D_{N_i}^i\}$. The global optimal set of paths $[D_*^1, D_*^2, \ldots, D_*^K]$ should have minimum interference, which means

$$G(D_*^1, \ldots, D_*^K) = min_{D_{p_i}^i \in \{D_1^i, \ldots, D_{N_i}^i\}} G(D_{p_1}^1, \ldots, D_{p_K}^K) \tag{4}$$

However, the optimal set of paths is not always unique. For example, in Fig. 5, the WMN contains 1 gateway O and 14 mesh nodes denoted from A to N. Path p_1 traverses from O to A, B ,C and D; path p_2 traverses from O to E, I ,J and H; path p_2^- traverses from O to E, F, G and H; and path p_3 traverses from O to K, L ,M and N. The dashed line indicates interference between two nodes. We can see that $C_2(p_1, p_2) = 4$; $C_2(p_2, p_3) = 9$; $C_2(p_1, p_3) = 3$; $C_2(p_1, p_2^-) = 7$; $C_2(p_2^-, p_3) = 6$. Therefore, $G(p_1, p_2, p_3) = G(p_1, p_2^-, p_3) = 32$. Then the load balancing becomes the second metric to break the tie. In Fig. 5, the path (p_1, p_2^-, p_3) is potentially better since the traffic distribution is more balanced.

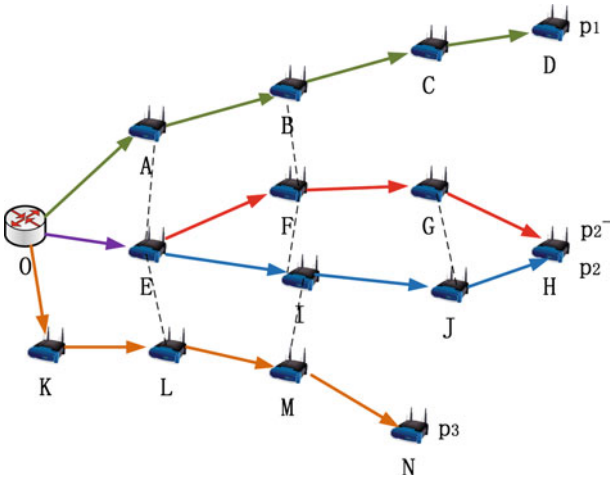

Fig. 5 Choosing among paths if two interference functions are the same

To quantify the load balancing, we introduce the Q function. In a set of paths, $p_1, p_2, \ldots, p_n$, the interference from other paths to the given path p_i is defined as

$$Q(p_i) = \sum_{j=1, j\neq i}^{n} C_2(p_i, p_j). \tag{5}$$

And the mean and variance of Q of this set of paths are defined as $E[Q] = \frac{1}{n}\sum_{i=1}^{n} Q(p_i)$, and $Var[Q] = \frac{1}{n}\sum_{i=1}^{n}(Q(p_i) - E[Q])^2$.

Suppose there are L optimal sets of paths determined by Eq. 4, then the best set s satisfying the load balancing metric is

$$\text{minimize}_{s\in(1,\ldots,L)}\{Var[Q_s]\} \tag{6}$$

In the network, "heavy-weight" applications could emerge and terminate any time. So it is necessary to develop a real-time update algorithm for new demands. Without losing generality, suppose there are N existing clients, with optimal paths $D^1, \ldots, D^N$ by Eq. 4 and Eq. 6, and M new clients arriving. Algorithm 1 updates the optimal paths for all clients. Note that

Algorithm 1 Algorithm for New Clients.

1: With fixed $[D^1, \ldots, D^N]$, find the optimal paths for the new client only, $D^{N+1}, \ldots, D^{N+M}$ by Eq. 4 and Eq. 6. The interference function $G(D^1, \ldots, D^{N+M}) = A$
2: Recalculate paths for all $N + M$ clients, and find the optimal path $\overline{D^1}, \ldots, \overline{D^{N+M}}$ with interference function $G(\overline{D^1}, \ldots, \overline{D^{N+M}}) = B$
3: **if** $\frac{B}{A} < \theta$ **then**
4: Choose the set of path $[\overline{D^1}, \ldots, \overline{D^{N+M}}]$
5: **else**
6: Choose the set of path $[D^1, \ldots, D^{N+M}]$
7: **end if**

$\theta \in [0, 1]$, since $B \leq A$. And θ controls the balance between efficiency and oscillation of the algorithm. When $\theta = 0$, the algorithm will always keep the paths for N existing clients and

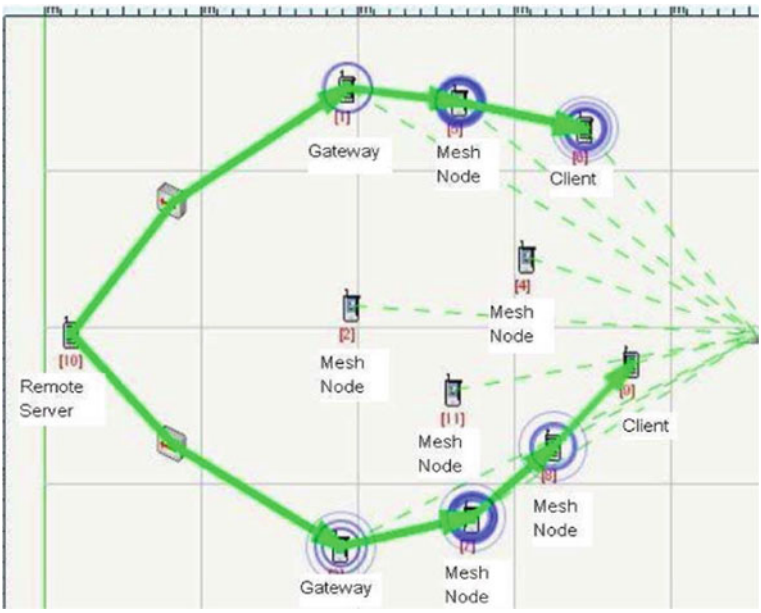

Fig. 6 Routes determined by MIROSE with two user requests in the network

choose paths for M new clients. On the other hand, when $\theta = 1$, the algorithm will utilize new paths for $N + M$ clients, which is a global optimal solution but has more oscillations.

3.2 Simulation Results of MIROSE

In this section, we evaluate the MIROSE algorithm by comparing its performance with popular wireless routing algorithm, AODV. Note that MIROSE algorithm can be combined with other routing algorithms to reduce interference with multiple paths, and thereby can improve system throughput.

The topology of our simulation is shown in Fig. 6 captured from QualNet [10], which consists of seven wireless mesh nodes with two gateways and one remote server. We use constant bit rate (CBR) source in the application layer to evaluate the performance MIROSE. There are two requests from users to the remote server in the network. We compare the performance based on different metrics including throughput and packet loss rate.

From our simulation observation, we find out that AODV selects two paths that share a number of common links which leads to severe interference between the two paths while MIROSE chooses two paths with least interference and also achieves load balancing in the wireless mesh network.

The simulation results of these scenarios have been presented in the following figures. The system throughput (defined as the sum of the individual throughput of each request) is shown in Fig. 7. Figure 7 shows that the throughput of MIROSE is better than AODV. When CBR rate is small, both AODV and MIROSE achieve the same throughput since the packet generation rate is much less than the network capacity. When rate of CBR sources is increased, throughput increases until they reach the path capacity. In this region, MIROSE performs better as it finds two paths with less interference and achieves load balancing.

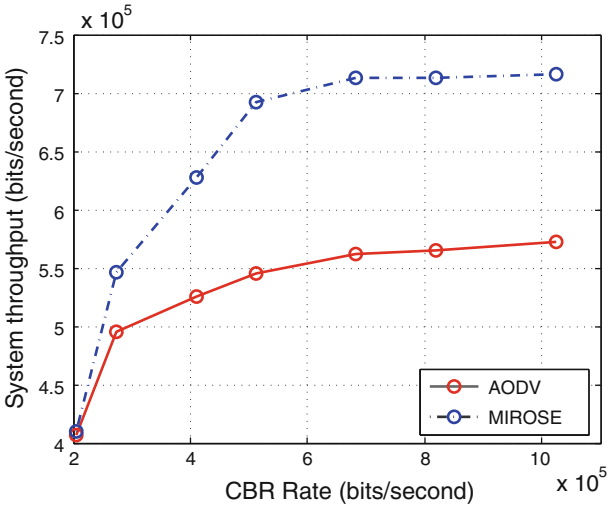

Fig. 7 Throughput comparison between AODV and MIROSE algorithms

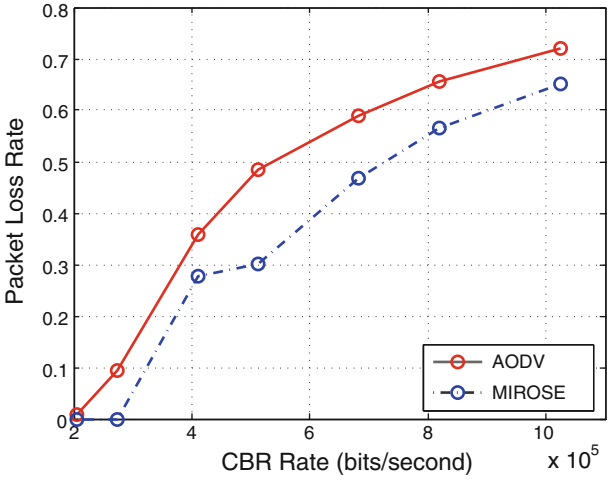

Fig. 8 Avg. packet loss rate comparison between AODV and MIROSE algorithms

The average packet loss rates of two paths is shown in Fig. 8. It is clear that AODV has higher packet loss rate than MIROSE. Similar to throughput, when the overall CBR rate is small compared to the network capacity, both AODV and MIROSE do not have any packet loss. Due to less interference between the two paths, MIROSE can tolerate higher rate of the CBR sources without significant packet losses. Moreover, MIROSE has less packet loss rate than AODV at the same CBR rate. This is because MIROSE select two paths with less interference to reduce congestion.

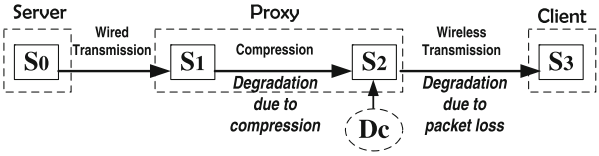

Fig. 9 Dynamic compression rate selection

4 Network State Dependent Video Compression Rate (NSDVCR) Optimization

The generic model in Fig. 9 shows the basic idea. The video proxy receives the video frames, denoted as S_0, from a remote server. It decompresses the video frames to extract the raw video denoted as S_1. Then the proxy compresses the raw video again to S_2 with an optimal compression rate determined by our algorithm NSDVCR, which is introduced in the next section. The video client eventually receives frames S_3 with distortion due to packet loss in the path from the proxy to the client.

As shown in the Fig. 9, the streaming rate after the compression in the video proxy, D_c, is the key parameter in the whole system. Two different aspects determines the optimal value of D_c. First, the ratio $\frac{D_c}{D}$, where D is the raw data transmission rate, is the video compression ratio. The smaller the ratio, the more information loss due to the compression in the proxy. If this ratio is too low, the received video quality at the client is inevitably poor even if there are no losses in the network. Second, since the transmission inside the wireless network cannot be error free, the current network condition, such as any congestion in the path or poor link quality, will influence D_c. Under poor network condition, a smaller value of D_c is preferred. On the other hand, when the network condition is good, the video can be streamed at a higher rate. Therefore, there is a tradeoff in selecting the value of D_c. In this section, we design a video compression rate optimization algorithm referred to as NSDVCR based on the video properties and the network condition.

In a GOP of (n,m), suppose the frame size ratio of three kinds of frames is $I : P : B = a : b : 1$ (typically the ratio is between $3 : 2 : 1$ and $5 : 3 : 1$). The video is compressed as r frames per second (fps). Note that in our algorithm, the values of n, m and r in frame S_2 are kept the same as in frame S_0 shown in Fig. 9. Inside the network, the packet size is z bits. Therefore, the number of packets contained in an I frame is

$$n_I = \lceil \frac{D_c * n/r}{1 + \frac{b}{a}(\frac{n}{m} - 1) + \frac{1}{a}(n - \frac{n}{m})} / z \rceil \tag{7}$$

And we can derive the number of packets contained in an B/P frame is $n_B = \frac{1}{a} n_I$ and $n_P = \frac{b}{a} n_I$, respectively.

Most Wi-Fi devices utilize adaptive rate fallback based on link quality (ARF in [11]) to improve the system capacity. Note that the link quality is measured in terms of the packet loss rate. The devices keep track of the packet loss rate. If the current packet loss rate exceeds a pre-defined threshold , it means the network condition is worse than expected and the sender needs to decrease the transmission rate. Otherwise, the sender increases the transmission rate until it reaches some pre-defined max limit, say 54 Mbits/s. In our system, we obtain the packet loss rate information from network layer, denoted as L_p. Consequently, we can get the loss rates of $I/P/B$ frame which are given by $L_I = 1 - (1 - L_p)^{n_I}$, $L_P = 1 - (1 - L_p)^{n_P}$, and $L_B = 1 - (1 - L_p)^{n_B}$, respectively.

We could calculate all 2^n combination of frame loss pattern to determine the cost[1] of packet loss inside a GOP. However, it is too expensive to calculate all the cost of each loss pattern. Therefore, we simplify the optimization algorithm to balance both efficiency and accuracy shown belows.

The average cost of a frame distortion consists of two parts: (1) the cost due to compression and (2) the cost due to transmission loss as shown in Fig. 9. The compression cost of a frame is a function of video compression rate and is defined as $C_c = g(\frac{D_c}{D})$. This function is related to video contents and can be obtained through experiments as described later. The other cost is introduced by packet loss. Assuming that the frames are independent, we obtain an approximate estimate of the packet loss cost inside a GOP. The total cost inside a GOP is given by

$$C_{\text{total}} = L_I C_I + (1 - L_I)C_c + \sum_{i=1}^{\frac{n}{m}-1} \binom{\frac{n}{m} - 1}{i} L_P^i (1 - L_P)^{\frac{n}{m}-1-i} [iC_P + (\frac{n}{m} - 1 - i)C_c] + \sum_{j=1}^{n-\frac{n}{m}} \binom{n - \frac{n}{m}}{j} L_B^j (1 - L_B)^{n-\frac{n}{m}-j} [jC_B + (n - \frac{n}{m} - j)C_c] \tag{8}$$

where, C_I, C_P, and C_B are average loss cost of I/P/B frames in a GOP, respectively. The first and second terms of Eq. 8 indicates the cost of I frame in case it is lost and in case it is not lost, respectively. The third and fourth terms are the cost of P and B frame, respectively. Note that the distortion of I frame loss will propagate until the next GOP, and C_I includes this effect. Since there are several P frames in a GOP, C_P is the average cost of P frame loss in a GOP.

There are multiple metrics to describe the cost, or video quality degradation. In this paper, we adopt the Peak Signal-to-Noise Ratio (PSNR) [12]. Some experiments of loss cost of I/P/B frames are shown in Fig. 10 using an enhanced framework for video transmission and quality evaluation (EvalVid) [13–16]. In Fig. 10, we test the video Foreman (QCIF, 400 frames, 25 fps, GOP(12,3)) [17]. X-axis is the frame number in a GOP when $n = 12, m = 3$ with sequence IBBPBBPBBPBB, respectively, and y-axis is PSNR value. The PSNR value of video with 600 and 300 kbps compression rate are 40.44461 and 36.37099 dB, respectively. The circle-sign and square-sign lines show the PSNR value with one individual type of frame loss in a GOP in different position. The triangle-sign and diamond-sign lines explicitly show the gap of PSNR values before and after the particular frame loss with compression rate 600 and 300 kbps, respectively. As can be seen from Fig. 10, different frames in different position affect the video quality differently. First, I frame affects the GOP PSNR most since the loss influence of I frame will propagate until the next GOP. Second, the effect of P frame loss decreases as the position of the P frame in the GOP increases. This is because the loss of P frame just affects frames after it until the next GOP. Third, a loss of B frame affect PSNR the least since the loss of B frame will not propagate in the GOP. Finally, different compression rates have different loss cost of I/P/B frames, but the loss cost has the same trend and similar values. In our algorithm, C_I, C_P, and C_B are average loss cost of I/P/B frames in a GOP.

Therefore, the average cost of a frame is given by $C(D_c) = C_{\text{total}}/n$. Note that here C is function of D_c. The optimization function then becomes

$$\min_{\{D_c\}} C(D_c) \tag{9}$$

[1] The cost here refers to the degradation in the video quality.

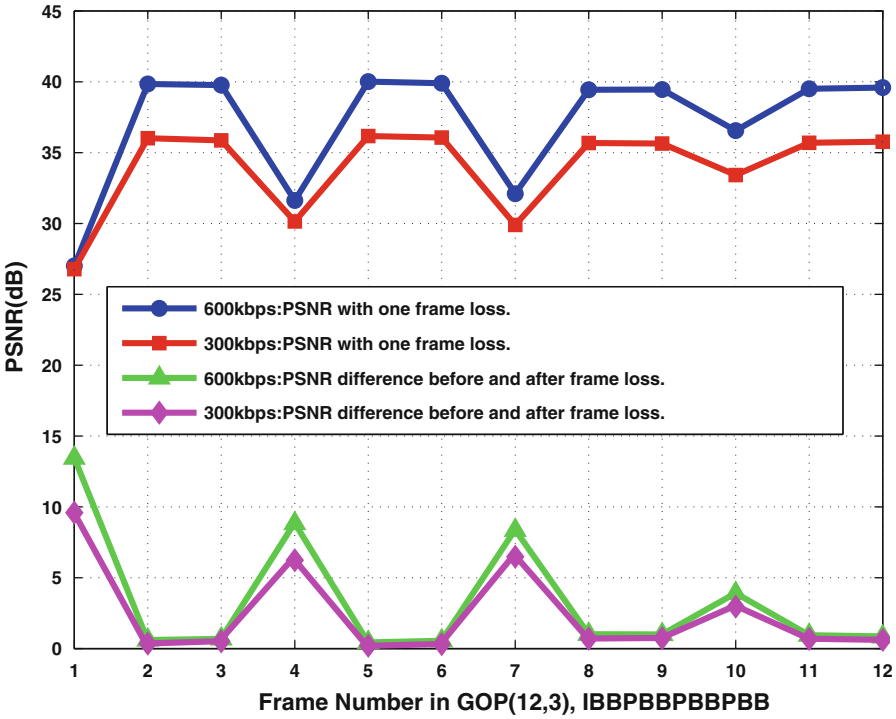

Fig. 10 Impact on PSNR for different types of frame loss in a GOP

where D_c is the decision variable. Note that our optimization framework is general in that we can use different metrics as the objective to improve video quality over wireless access networks. If Mean Squared Error (MSE) is used as a metric, let $C(D_c) = \text{MSE}(D_c)$. MSE is defined as $\text{MSE} = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}\|I(i,j) - K(i,j)\|^2$, where I and K are two frames with size of $m * n$ pixels, where one of the frames is considered a noisy approximation of the other.

Another popular objective metric to evaluate video quality is PSNR. PSNR can also be used in our optimization algorithm. In this paper, we use PSNR as a example to evaluate our mechanism. PSNR is defined as follows.

$$\text{PSNR} = 10 \times \log_{10}\left(\frac{MAX_I^2}{\text{MSE}}\right) \tag{10}$$

where MAX_I is the maximum pixel value of the image. Note that PSNR was originally proposed as the metric to compare a single frame. In this study we utilize the average value of PSNR of multiple frames in a video as our major objective to improve.

To use PSNR as our metric, we let $C = -\text{PSNR}$ since we would like to maximize PSNR. Consequently, higher PSNR value correspond to better quality. We performed some experiments to determine how the PSNR changes depending on different video compression rates. The results for the standard videos CoastGuard, Foreman, and News [17] are shown in Fig. 11. From our experiments, we find that when compression ratio is higher than some threshold, PSNR does not improve. This is because every compression algorithm has its limit. This means that we cannot get more benefits when the video stream rate is higher than a certain

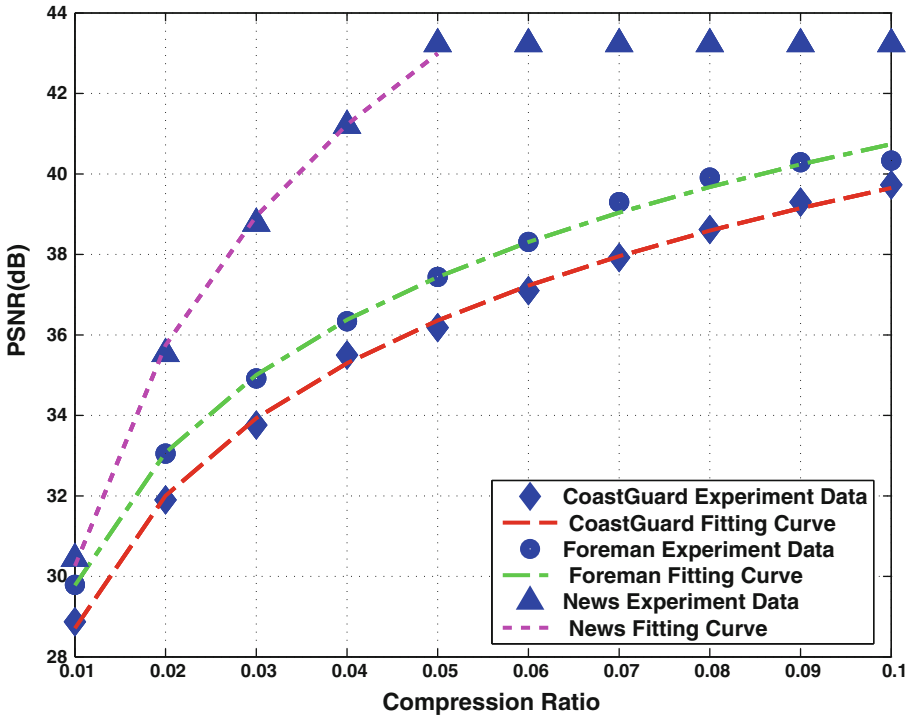

Fig. 11 A curve fitted model of PSNR for different QCIF videos CoastGaurd, foreman, and news

value, and it is a waste of network bandwidth if we compress the video with compression ratio higher than the threshold. We also find that PSNR depends on the video content. Video with less motion (like News) has higher PSNR with the same compression rate, and meets the compression threshold earlier and vice-verse.

Based on the above experiments and curve fitting method, C_c can be fitted using the following function

$$C_c = -(x * \ln \frac{D_c}{D} + y), x, y \in \mathbb{R}, D_c \leq D * H \tag{11}$$

where H is the threshold of compression ratio mentioned in the previous paragraph. In Fig. 11, the data points are experimental values fitted with the logarithmic function in Eq. 11. The parameters, D, x and y, depend on the content of the video and the coding algorithm, whose reference values can be obtained by non-linear regression techniques. In the implementation, those parameters could be either adjusted by the network administrator or by a machine-learning mechanism. If the network administrator has a clear prediction of the video on transmission, they can pre-set those values, and adjust them later if necessary. With the equally divided time periods, the machine-learning mechanism can also provide the estimated values for those parameters in one time period based on the measurements in the previous ones.

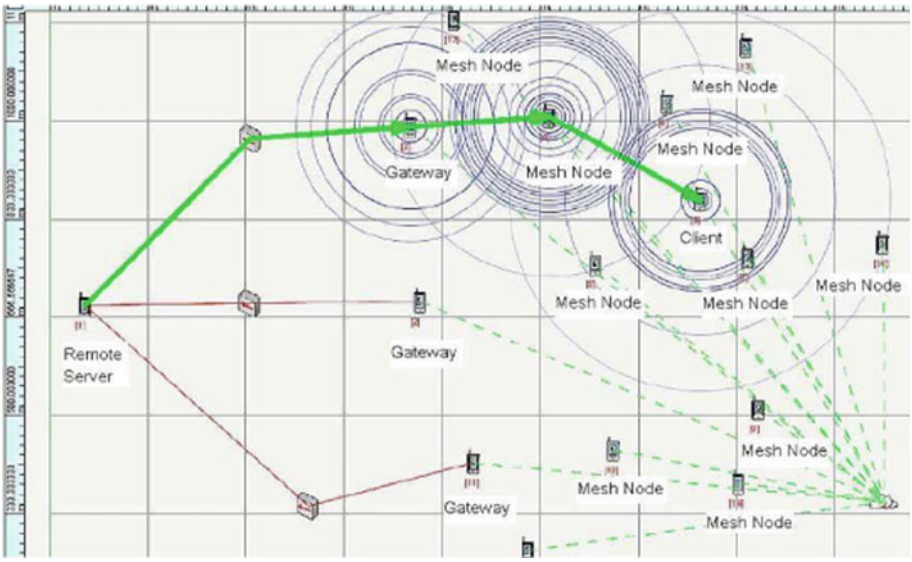

Fig. 12 Random topology simulation

5 Results and Discussions

In this section, we evaluate the performance of the video compression rate optimization scheme using QualNet [10] and a revised EvalVid framework [13–16]. We evaluated our schemes using both a grid topology and a random topology in a WMN. These two topologies are designed to have similar average hops to gateways and the video requests are two hops from the gateways. The grid topology contains 16 mesh nodes, including two gateways. One mesh node two hops away from the gateway has a video request from the remote video server and all other mesh nodes have CBR background traffic with different traffic intensity, which is the traffic generation rate from the remote server. In the random topology shown in Fig. 12, node 1 is a remote sever. The mesh network contains 15 nodes. Three of the nodes, nodes 5, 3 and 11 are gateways. Node 5 serves video streaming requests and all other nodes have CBR background traffic from the remote server.

We used the Foreman [17] video which is in the Quarter Common Intermediate Format (QCIF) with 176×144 Y pixels/frame and 88×72 Cb/Cr pixels/frame. Foreman is a video with normal motion. We use QualNet 4.0 [10] for simulations. At the physical layer, we use 802.11b with data rate 2 Mbps. The MAC protocol is 802.11. We use MIROSE as routing protocol and UDP is used at the transport layer. We have two applications, CBR and video streaming.

In the simulation analysis, different transmission rates of the CBR traffic has been employed to evaluate the performance of our algorithm for different network conditions. Every node has the same CBR rate during one simulation. From these simulations, we find out that our proposed NSDVCR is able to adaptively choose the optimal streaming rate depending on the underlying topology and the interference pattern. Results can be referred to [4]. We do not show the results here due to space limitation.

In order to evaluate the performance of our algorithm, we compare PSNR of optimal video compression rate with the PSNR of some typical video compression rates found in

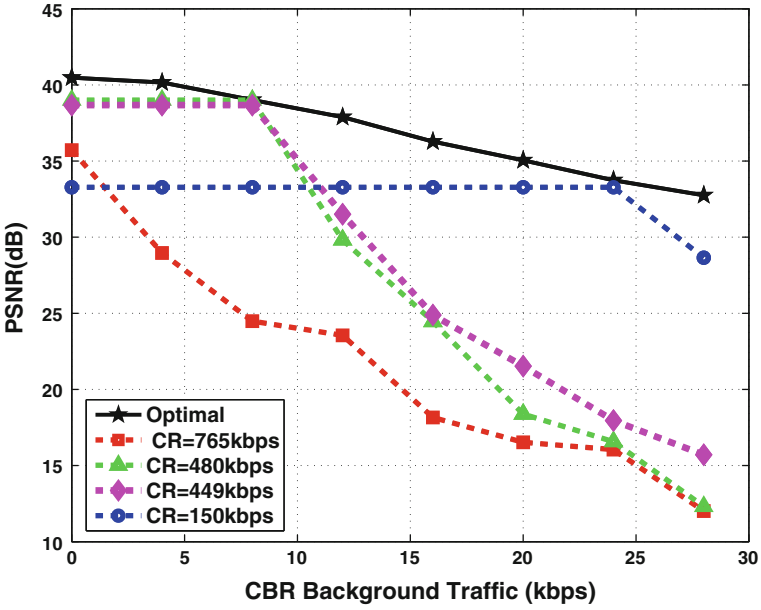

Fig. 13 PSNR vs. different background traffic intensity (grid topology)

video sharing websites. In particular, we considered the following compression rates: 765, 480, 449 and 150 kbps. Simulation results are shown in Fig. 13 for the grid topology and in Fig. 14 for the random topology. Note that "CBR Rate" of X-axis is the packet generation rate of one node.

From the two figures, we can see that the "Optimal" line and all other compression rate lines have almost the same trend and relationship. Here we use Fig. 14 as an example for analysis; the analysis also works for Fig. 13. In Fig. 14, the solid line is the optimal PSNR obtained from our video rate optimization algorithm. We can see that using our scheme, the video quality is the best since it determines the optimal video compression rate to fit the current network condition. Using any of the typical video transmission rates yields lower performance than our proposed algorithm. The "Optimal" line can be considered as the upper bound of video quality since our algorithm is designed to obtain the optimal video compression rate so as to attain best video quality. When the background traffic is light, higher video transmission rate results in higher PSNR since most network resource is given to the video flow and there are very few packet losses in the network. When the background traffic is high, if the video transmission rate is higher than the path capacity, packet losses will be high which will result in lower PSNR. As can be seen from Fig. 14, some video compression rates (e.g., 449 and 480 kbps) have better performance than that of 150 kbps video compression rate when background traffic is below 24 kbps (CBR rate of one node). On the contrary, 150 kbps streaming rate results in a better PSNR value when background traffic is high. However, for all scenarios, the PSNR of the compressed video obtained from our algorithm is the best. As mentioned before, our optimization algorithm is designed to find the optimal video compression rate. If there is no video compression optimization algorithm, the default compression rate cannot be guaranteed to provide the best performance. Another observation that can be made from Fig. 14 is that the video PSNR remains constant before it exceeds

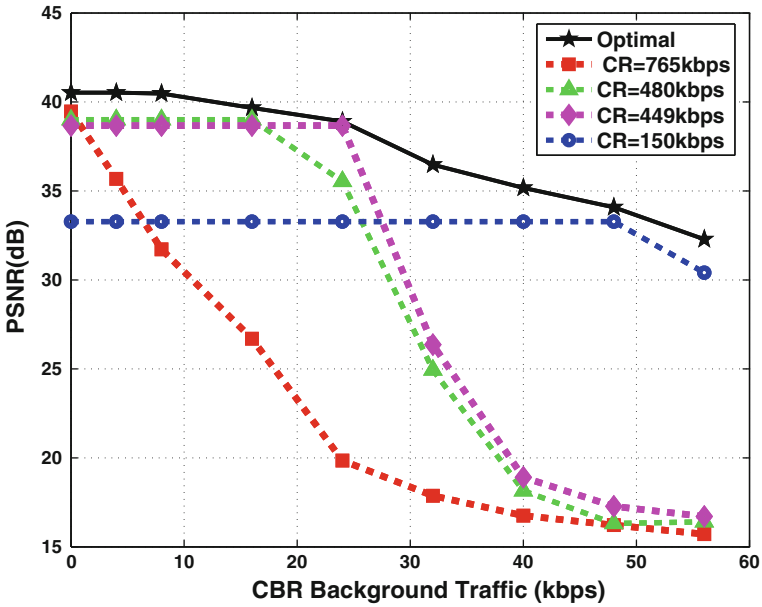

Fig. 14 PSNR vs. different background traffic intensity (random topology)

Table 1 Optimal Compression Rate (CR) for different background traffic intensities with three video requests in a random topology

CBR rate (kbps)	Avg. PSNR (dB)	CR 1	PSNR 1 (dB)	CR 2	PSNR 2 (dB)	CR 3	PSNR 3 (dB)
0	40.53	742	40.53	742	40.53	742	40.53
10	40.53	710	40.52	742	40.53	742	40.53
20	40.51	690	40.49	742	40.53	721	40.52
40	40.50	673	40.48	691	40.49	715	40.52
60	39.59	530	39.25	510	39.01	705	40.52
80	38.37	421	38.03	432	38.17	470	38.91
100	37.04	335	36.42	348	36.46	440	38.25
120	32.60	150	33.28	102	31.24	150	33.28

the network bandwidth. This corresponds to the flat portion of the curves. Beyond that, the PSNR value decreases quickly because of packet loss.

We also evaluated our proposed NSDVCR algorithm with three video requests in the random topology, shown in Fig. 12. Nodes 5, 16 and 9 receive video streams and nodes 4, 6, 8, 10, and 13 receive CBR traffic from the remote server node 1. All parameters keep the same except that the physical model is IEEE 802.11a, physical data rate is 6 Mbps. The optimal compression rates and average PSNR values of node 5, 16, and 9 under different background traffic are shown in Table 1. Since node 9 has the least number of hops and lowest interference, it has the highest optimal compression rate compared to nodes 5 and 16 under the same background traffic intensity. We also compare our results with typical video compression rates for all three video requests shown in Fig. 15. The PSNR values in the

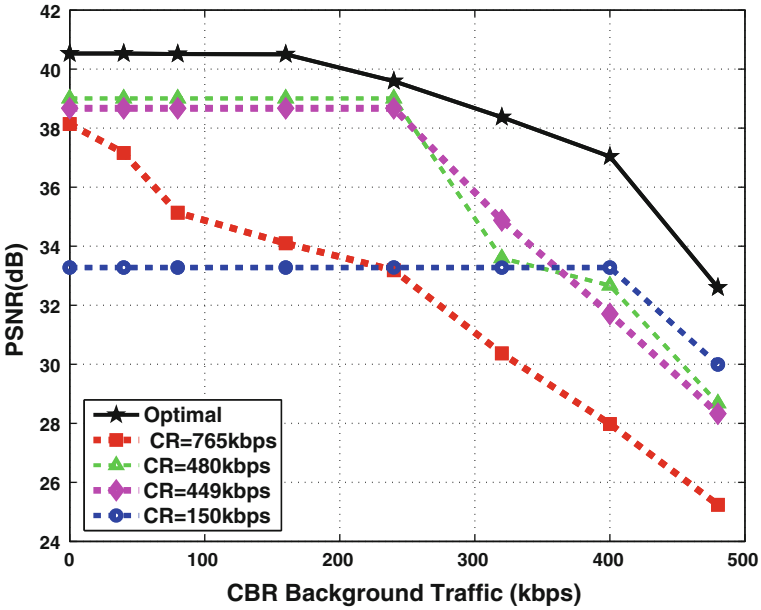

Fig. 15 Average PSNR of three video streams in the random topology

figure are the average PSNR of the three video streams. We can see that the PSNR curves have the same trend as the single video request results discussed earlier.

6 Related Work

In this section, we give a brief overview of recent research in this field. A number of studies [18–22] have proposed methods to improve the performance of video streaming in WMNs and wireless ad hoc networks. From both experimental studies [3] and theoretical models [20,18,19,21] it is shown video quality in WMNs is severely affected by routing algorithm and video compression rate. In [18], the authors just concentrated on the video rate allocation. In [19], the authors proposed an optimal routing scheme to minimize congestion in ad hoc networks. However, note that both routing and video compression rate have impacts to the video quality for users in wireless networks. Concentrating on one aspect and neglecting the other aspect is not enough.

The work in [20] designed an algorithm for routing and rate allocation simultaneously for video streams in ad-hoc networks with the aim to minimizing video distortion and network congestions. It seems that our work is similar to their work since we are making efforts in the same directions. However, our work is different from theirs. First, their algorithm is based on a generic network model suitable for both wired and wireless networks. The network model does not capture the distinct characteristic of wireless networks. The performance of the optimization is based on the preciseness of the model. Our algorithm is designed directly based on wireless channel interference information in and between paths. Second, the video distortion model in [20] just describes the video distortion with different rate. It cannot model the important information that different types of loss frame have different effects to the video

quality while our optimization algorithm more precisely model the characteristics of I/P/B frames.

In [21], the authors present a joint capacity, flow and rate allocation algorithm for video over ad-hoc networks. Our work is different from theirs in that we concentrate on find a set of minimal interference routes to support multiple video over WMNs while [21] is more concentrate on capacity and flow assignments. From testbed results in [3], we know that both inter- and intra- flow interference have a great impact on video quality. Our work directly designed to solve this problem.

Unlike ours, the work in [22] assumes an overlay network infrastructure to convey information of each link so as to optimize different control parameters across the protocol layers, thus to optimize video quality. This paper demonstrate the need for cross-layer optimization in order to provide an efficient solution for video streams in WMNs, so as [23–25]. Our framework is a cross-layer design of network layer (packet loss rate, route selection) and application layer (video compression rate optimization).

Other related papers are [26–29]. In [26], the authors proposed an algorithm for video rate adaptation, which is evaluated by a subjective quality model in [27]. In [28], the authors concentrate on an optimization algorithm for rate allocation for multiple video streaming sessions sharing multiple access networks. In [29], the authors have proposed a framework for rate-distortion optimized streaming. However, these papers are all based on a single-hop wireless network environment. Our optimization algorithm NSDVCR is more versatile, which can be used for both single-hop and multihop wireless networks. Finally, our proposed scheme is more generic in that different kinds of metrics can be used to improve video quality. The proposed method is not restricted to only one or two specified metrics.

7 Conclusions

Heavy weight applications which have high bandwidth and low latency requirements such as VoD and IPTV, impose challenges when implemented over multi-hop wireless mesh networks. In this paper, we analysis the reasons that cause video distortion in wireless mesh networks. We introduce a proxy at the edge of the wireless mesh network to enhance the delivery of video traffic to clients. We proposed a route selection algorithm MIROSE that can choose the minimal interference routes to make better use of network resources. We also proposed an optimization algorithm NSDVCR that determines the optimal video streaming rate and adapts to the network condition. The algorithms are implemented in the proxy that determines the routes for video streaming and determines the optimal video compression rate. The simulation results show significant improvement in the received video quality when the proxy adopts the optimal streaming rate compared to any of the popular pre-defined rates. Note that the general model in NSDVCR is easy to extend to support other compression mechanism or metrics. For example, replacing the *cost* by other metrics than PSNR provides the comparison of video qualities in many other ways.

References

1. Akyildiz, I. F., & Wang, X. (2005). A survey on wireless mesh networks. *IEEE Communications Magazine, 49*(9), s23–s30.
2. Cheng, X., Mohapatra, P., Lee, S.-J., & Banerjee, S. (2008). MARIA: Interference-aware admission control and QoS routing in wireless mesh networks. In *Proceedings of IEEE ICC*, Beijing, China.
3. Cheng, X., Mohapatra, P., Lee, S.-J., & Banerjee, S. (May 2008). Performance evaluation of video streaming in multihop wireless mesh networks. In *Proceedings of NOSSDAV*, Braunschweig, Germany.
4. Qiu, X., Liu, H., Ghosal, D., Mukherjee, B., Benko, J., & Li W., et al. (2009). Adaptive video compression rate optimization in wireless access networks. In *Proceedings of IEEE LCN*, Switzerland.
5. ISO/IEC 11172. *Information technology-coding of moving pictures and associated audio for digital storage media at up to about 1.5 MBit/s.*
6. ISO/IEC, Document 13818-2 (MPEG-2). *Generic coding of moving pictures and associated audio, part 2: Video, recommendation H.262, International Standard*.
7. ITU-T. (1993). Rec. H.261 Vers. 3/1993, video codec for audiovisual services at p*64 kbit/s.
8. RFC 4728. *The Dynamic Source Routing Protocol (DSR) for mobile ad hoc networks for IPv4*.
9. RFC 3561. *Ad hoc on-demand distance vector*.
10. Scalable Network Technologies. http://www.scalable-networks.com.
11. Kamerman, A., & Monteban, L. (1997). WLAN-II: A highper performance wireless LAN for the unlicecsed band. *Bell Labs Technical Journal.*
12. Netravali, A. N., & Haskell, B. G. (1995). *Digital pictures: Representation, compression, and standards, 2nd ed*. New York: Plenum Press.
13. Klaue, J., Rathke, B., & Wolisz, A. (2003). EvalVid: A framework for video transmission and quality evaluation. In *Proceedings of the 13th international conference on modelling techniques and tools for computer performance evaluation*.
14. Lie, A., & Klaue, J. (2007). EvalVid – a framework for video transmission and quality evaluation. In proceedings of the 13th international conference on modelling techniques and tools for computer performance evaluation, pp. 255–272.
15. EvalVid. A video quality evaluation tool-set. http://www.tkn.tu-berlin.de/research/evalvid/.
16. Ke, C.-H., & Chilamkurti, N. (2008). A new framework for MPEG video delivery over heterogeneous networks. *Computer Communications, 31*(11), 2656–2668.
17. YUV QCIF reference videos. http://www.tkn.tu-berlin.de/research/evalvid/qcif.html.
18. Zhu, X., & Girod, B. (Sept 2006). Media-aware multi-user rate allocation over wireless mesh networks. In *Proceedings of IEEE first workshop on operator-assisted (wireless mesh) community networks*, Berlin, Germany.
19. Setton, E., Zhu, X., & Girod, B. (2004). Minimizing distortion for multi-path video streaming over ad hoc networks. In *Proceedings of international conference on image processing (ICIP)*.
20. Zhu, X., Singh, J. P., & Girod, B. (2006). Joint routing and rate allocation for multiple video streams in ad hoc wireless networks. *Journal of Zhejiang University Science A, 7*(5), 727–736.
21. Adlakha, S., Zhu, X., Girod, B., & Goldsmith, A. J. (2007). Joint capacity, flow and rate allocation for multiuser video streaming over wireless ad hoc networks. In *Proceedings of ICC*.
22. Andreopoulos, Y., Mastronarde, N., & Van Der Schaar, M. (2006). Cross-layer optimized video streaming over wireless multihop mesh networks. *IEEE Journal on Selected Areas in Communications*.
23. Setton, E., Yoo, T., Zhu, X., Goldsmith, A., & Girod, B. (2005). Cross-layer design of ad hoc networks for real-time video streaming. *IEEE Wireless Communications Magazine, 12*(4), 59–65.
24. Shankar, N. S., & van der Schaar, M. (2007). Performance analysis of video transmission over IEEE 802.11a/e WLANs. *IEEE Transactions on Vehicular Technology, 56*(4), 2346–2362.
25. Zhu, X., & Girod, B. (Sept 2007). Video streaming over wireless networks. In *Proceedings of European signal processing conference*.
26. Zhu, X., van Beek, P., & Girod, B. (Sept 2007). Distributed channel time allocation and rate adaptation for multi-user video streaming over wireless home networks. In *Proceedings of IEEE international conference on image processing*, San Antonio, USA.
27. Zhu, X., & Girod, B. (Sept 2008). Subjective evaluation of multi-user rate allocation for streaming heterogeneous video contents over wireless networks. In *Proceedings of IEEE conference on image processing (ICIP)*.
28. Zhu, X., Agrawal, P., PalSingh, J., Alpcan, T., & Girod, B. (Sept 2007). Rate allocation for multi-user video streaming over heterogenous access networks. In *Proceedings of the 15th international conference on multimedia*.
29. Chou, P. A., & Miao, Z. (2006). Rate distortion optimized streaming of packetized media. *IEEE Transactions on Multimedia, 8*(2), 390–404.

Author Biographies

Xiaoling Qiu is currently a Ph.D. candidate of the Computer Science Department of University of California, Davis. She received a Master and Bachelor degree from Beijing University of Posts and Telecommunications, China. Her research areas include optimization of video delivery in WLAN and wireless mesh networks. She also did research in the area of congestion control in wireless sensor networks.

Haiping Liu is currently a Ph.D. candidate in the Department of Electrical and Computer Engineering at University of California, Davis. He received a Master degree in Information Engineering Department from the Chinese University of Hong Kong. He also got his Bachelor degree from Beijing University of Posts and Telecommunications. His research areas include the efficient resource allocation, and the novel network architecture design in wireless networks.

Dipak Ghosal is a Professor in the Department of Computer Science at the University of California, Davis. He received his Ph.D. degree at the University of Louisiana in 1988, his Master degree at the Indian Institute of Science, Bangalore, in 1985 and his Bachelor degree at the Indian Institute of Technology, Kanpur, in 1983. Professor Ghosal's primary research interests are in the areas of high-speed and wireless networks with particular emphasis on the impact of new technologies on the network and higher layer protocols and applications. He is also interested in wireless ad hoc and sensor networks and end-system aware optimization of transport protocols. He is a member of IEEE.

Biswanath Mukherjee holds the Child Family Endowed Chair Professorship at University of California, Davis, where he has been since 1987, and served as Chairman of the Department of Computer Science during 1997 to 2000. He received the B.Tech. (Hons.) degree from Indian Institute of Technology, Kharagpur, in 1980, and the Ph.D. degree from University of Washington, Seattle, in 1987. He served as Technical Program Co-Chair of the Optical Fiber Communications (OFC) Conference 2009. He served as the Technical Program Chair of the IEEE INFOCOM '96 conference. He is Editor of Springer's Optical Networks Book Series. He serves or has served on the editorial boards of eight journals, most notably IEEE/ACM Transactions on Networking and IEEE Network. He is Steering Committee Chair of the IEEE Advanced Networks and Telecom Systems (ANTS) Conference, and served as General Co-Chair of ANTS in 2007 and 2008. He is co-winner of the Optical Networking Symposium Best Paper Awards at the IEEE Globecom 2007 and IEEE Globecom 2008 conferences. He won the 2004 UC Davis Distinguished Graduate Mentoring Award. He also won the 2009 UC Davis College of Engineering Outstanding Senior Faculty Award. To date, he has supervised to completion the Ph.D. Dissertations of 40 students, and he is currently supervising approximately 20 Ph.D. students and research scholars. He is author of the textbook "Optical WDM Networks" published by Springer in January 2006. He served a 5-year term as a Founding Member of the Board of Directors of IPLocks, Inc., a Silicon Valley startup company. He has served on the Technical Advisory Board of a number of startup companies in networking, most recently Teknovus, Intelligent Fiber Optic Systems, and LookAhead Decisions Inc. (LDI). He is a Fellow of the IEEE.

John Benko currently works as a system architect for Orange Labs in San Francisco. With over 10 years of experience in the telecommunications industy, his areas of interests include 802.11 standardization and video transmission over wireless, as well as a broad range of mobile topics. John has a Bachelor's and Master's degree in Engineering from MIT.

Wei Li received his Ph.D. degree in electrical engineering from the University of Victoria, Canada in 2004. He is currently a Senior Lecturer with Victoria University of Wellington. His research interests are in information theory and contemporary wireless systems. His current research areas are: compressive sensing, UWB channel modeling and system design, locating and tracking based on wireless sensor networks; wireless network self-optimization and heterogeneous networks.

Rashmi Bajaj works as a Senior Systems Engineer at Orange Labs San Francisco. Her current research areas include wireless networks, including local area networks (WLAN) and metropolitan area networks (WMAN). She is also a technology analyst and contributes to internal strategic teams at Orange. Rashmi holds a Master of Science in Computer Engineering from University of Cincinnati, Ohio (USA) and a Bachelor in Electronics and Communication Engineering from University of Madras, India.