SECURITY AND COMMUNICATION NETWORKS
Security Comm. Networks 2015; **8**:2798–2812
Published online 4 May 2015 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/sec.1203

RESEARCH ARTICLE

Security–quality aware routing for wireless multimedia sensor networks using secret sharing

Abdelnaser Rashwan[1], Honggang Wang[1]*, Dalei Wu[2] and Xinming Huang[3]

[1] University of Massachusetts, Dartmouth, MA, U.S.A.
[2] University of Tennessee, Chattanooga, TN, U.S.A.
[3] Worcester Polytechnic Institute, Worcester, MA, U.S.A.

ABSTRACT

Security and video quality are progressively significant attributes for wireless multimedia sensor networks. Most of existing research considers security and video quality separately. However, it is crucial to integrate security and video quality together for video transmission because delivering video data across a secure path does not often meet video quality requirements in many traditional approaches. Applying the general concept of secret sharing algorithm on a data packet and delivering it through disjoint multipaths can be considered to deliver the data securely. However, using the general concept of secret sharing is not efficient when large-size video data are routed. To tackle these issues, we propose a novel security and quality aware routing (SQAR) protocol to address these two issues concurrently. We jointly consider security and video quality in wireless multimedia networks by proposing a video distortion model based on a new secret image sharing scheme. In SQAR, a secret image sharing is only applied on the intra-frames of the video codec H.264 and can significantly reduce the transmission overheads. Simulation results show that SQAR scheme can achieve better trade-off between the security and quality over the traditional routing protocols. Copyright © 2015 John Wiley & Sons, Ltd.

KEYWORDS
wireless multimedia sensor networks; security; video transmission

***Correspondence**
Honggang Wang, Electrical and Computer Engineering, University of Massachusetts, Dartmouth, MA, U.S.A.
E-mail: hwang1@umassd.edu

Received 30 November 2013; Revised 22 September 2014; Accepted 3 November 2014

1. INTRODUCTION

The advent of low-cost hardware such as complementary metal oxide semiconductor (CMOS) cameras and microphones has enhanced the growth of wireless multimedia sensor networks (WMSNs) [1]. These wireless interconnected devices have the ability to collect multimedia content such as video and audio streams, still images, and scalar sensor data from the environment. WMSNs are a category of wireless sensor networks (WSN), although they have some requirements that make them different from the traditional sensor networks. These requirements happen as a result of various design constraints such as throughput, delay, jitter, distortion, and loss ratio. Thus, researchers in this field always focus on minimizing latency and overheads at each layer. Additionally, WMSNs are affected sharply by packet losses, which lead to jitter and distortion in the received video. In wireless multimedia networks, packets that come after their playback deadline are useless for video reconstruction at the receiver. Therefore, the main goal of routing protocols for WMSNs is to find those paths that can either satisfy or reduce the end-to-end delay or the end-to-end distortion.

The metrics most commonly used in routing protocols are minimum average-packet-loss, minimum average-packet-delay, expected number of retransmission, and minimum number of hops. Using each individual metric might not optimize the overall network performance efficiently. For instance, packets that choose a path with minimum number of hops may take significant time to reach the destination. This may result in dropping the packets when they exceed the playback deadline and thus increases the total distortion of the received video.

In other routing schemes, other metrics such as the end-to-end path bandwidth are used to find an optimal routing path. Usually, these metrics are subject to one or more constraints such as delay and packet loss [1,2]. However, these metrics can be straightly associated to the received video quality and are not designed to take into consideration the effect of error concealment on the rendered

video quality. A forensics-aware multimedia scheduling scheme is developed by employing a scalable media-aware forensics scheme [3]. It performs a good trade-off between flexibility and overhead. Moreover, this forensics-scheduling scheme operates by taking into consideration multimedia applications delay and authentication constraints. However, sometimes delivering multimedia while only taking into account the requirements of delay does not guarantee good multimedia quality. There have also been many works that use the end-to-end video quality (as it is watched by the end user) as routing metric. More recently, there have been several cross-layer schemes that focus on enhancing the quality of the received video by providing the optimal route of video data over wireless networks. In [4], an optimization problem that takes into consideration the retransmission scheme, along with path selection, and the physical layer transmission scheme is addressed to reduce the expected video distortion at the application layer. In [5] and [6], multipath routing schemes are suggested for video multipath transport by utilizing path diversity. A collection of paths are calculated. One path for each video stream is determined to minimize the received video distortion. In all the previously discussed algorithms, the expected video distortion is employed as a routing metric, which is either pre-estimated [4,7] or produced from video distortion-rate models [6–10], without taking into consideration the effects of video coding and error concealment techniques on routing path determination. Because of pre-estimated video streaming, dynamic optimal routing path determination cannot be achieved to video coding for real-time video applications. To tackle this issue, in [11], a dynamic path routing selection is proposed to integrate with online video coding to achieve the highest-level perceived video quality.

On the other hand, the broadcast nature of WMSNs makes them extremely susceptible to attacks such as eavesdropping, interference, and jamming. For example, attackers can obtain secrets, tamper with the associated sensor hardware, change programming in the sensors, or replace them with malicious sensors under the control of the attacker. If the proposed routing schemes are not designed to prevent such attacks, routing in WMSNs may be difficult. Because of the aforementioned security limitations, security in WMSNs has drawn researchers' attention.

Recently, a number of secure routing schemes have been developed to address the secure routing problem in wireless networks. In [2,12], the concept of secret sharing is used to develop a secure routing protocol. It works by dividing data packets into smaller packets called shares, and these shares are sent through disjoint multipath. An unauthorized user has to intercept at least a threshold number of those shares before the packet can be decrypted. In [13], to secure the data transmission in wireless network, each path frequently transmits a reliability rating that is calculated by the ratio of the successful packet deliveries to unsuccessful packet deliveries over that path. A security scheme is suggested in [14] for image sensors network. This scheme employs the concept of secret image sharing on multiple node-disjoint paths to transmit the image data. In this scheme, the image is divided into overlapped and non-overlapped regions where the non-overlapped region is sent with no encryption, while the overlapped region is divided into small images called shares via secret sharing and delivered according to suitable distribution ratio via multiple paths. In order to secure data integrity of data transmitted over WMSN, an energy-aware wavelet-based watermarking scheme is proposed in [15]. This scheme inserts additional information named watermark into some parts in an image object so that it can be reconstructed to make an assertion about the object. The locations of watermarking are selected by network conditions so that the energy efficiency and security can be accomplished. A security paradigm is presented for WMSNs. The presented protocol obliges each node to obtain a key shared with the central node and pairwise keys shared between the nodes in the cluster. All keys are used for symmetric cryptography to offer various security services [16].

However, these protocols often have to send extra shares to increase the reliability, which could increase the communication overheads. Although these previous security schemes succeed in enhancing the security level significantly, they are difficult to deal with large-size multimedia data because of significant overheads. Furthermore, their routing path selection problem is only focused on choosing the most secure paths and may not guarantee the quality of service (QoS) simultaneously.

The future routing approaches should achieve both security and transmission quality for wireless multimedia transmission based on the following observations: First, different video coding modes [17] lead to various rate-distortion values. If the resulted distortion is used as routing metric, different paths might be selected. In other words, when only one video coding mode is used to select the path, then the selected path might not be the best path to enhance video quality under given network conditions. Second, in many cases, a network might suffer from bad conditions because of a high packet loss rate. Therefore, video coding parameters have to be adapted to lower rate video streaming and find the optimal routing path. In contrast, in good network conditions, high data rate source coding could be used to take full advantage of network resources, resulting in better received video quality [17].

Here, security and quality aware routing (SQAR) employs the secret image sharing algorithm and perceived video quality routing metrics. Because decoding the inter-frames (P-frames) cannot be achieved without successful decoding of intra-frame (I-frame), we suggest securing the video sequence by applying secret image sharing only on the I-frame to produce the shares. The matter of optimal path choice is addressed with taking into consideration the upper layer video coding, along with a shares allocation constraints. Explicitly, path selection and video coding are considered together to be adapted to time-varying characteristics of the network while satisfying the constraints of shares allocation and the end-to-end delay. We followed the same approach in [4], where the expected

end-to-end distortion is used to evaluate the received video quality. As a result, the objective function of the routing problem is designed to minimize the expected end-to-end video distortion under the shares allocation and delay deadline constraints.

This paper is organized as follows. Section 2 describes the framework specification (secret image sharing, shares allocation, etc.) and explains the cross-layer optimization problem formulations. In Section 3, we present our solution that can dynamically choose intermediate nodes that minimize the expected end-to-end video distortion while maintaining the predefined level of security. Section 4 presents our simulation results. Finally, Section 5 concludes the paper.

2. SYSTEM MODEL

Security and quality aware routing takes a similar assumption as the one in [4,18–20], where a controller is assumed to interact with every layer at the source node for the following reasons. First, it obtains all required information (e.g., number of shares over a path and expected end-to-end distortion) to achieve the dynamic routing. Second, it updates the values of the parameters at different layers (e.g., video coding) according to the optimization results and network conditions. To ensure secure data delivery, the SQAR uses the secret image sharing technique in [21]. The network topology and the corresponding channel conditions of each link are assumed to remain unchanged within the service time. Each node maintains a queue containing video packets from various neighbors. The following information such as the global network topology, channel state information, concealment strategy, and node average packet arrival rates is supposed to be accessible to the controller. The SQAR scheme is assumed to be built on top of a proactive routing scheme such as optimized link state routing (OLSR), because each node in OLSR scheme keeps up-to-date routing information that is used in optimal paths determination.

The video clip is divided into a sequence of frames $\{n_1, n_2, \ldots, n_M\}$ to be coded and delivered to a destination D. This video clip consists of a number of collections known as the group of pictures (GOPs). Each frame in the GOP represents an image, and the first frame in GOP is an I-frame, which can be decoded with no reference to other frames. Following the I-frame is a sequence of P-frames, which cannot be decoded without reference to I-frame. Each frame n_z is divided into K slices $\{s_{z,1}, s_{z,2}, \ldots, s_{z,K}\}$. Each slice comprises of a row of macroblocks (MBs). Before transmission, each slice of the I-frame $s_{z,t}$ is split into a number N of shares $\{sh_{z,t,1}, sh_{z,t,2}, \ldots, sh_{z,t,N}\}$ by using a secret image scheme.

The I-frame slice can be retrieved at the destination by a number of shares that is greater than or equal to T as we will explained next. These shares are transmitted to the destination over m disjoint paths. Hence, each node in the network (except the source and the destination) is assumed to be likely compromised. Therefore, these shares must be sent in a way that prevents the slice of those shares from being illegally accessed by an unauthorized user in the intermediate nodes. To achieve this, a shares allocation scheme is applied, it operates by sending $(T-1)$ shares over $(m-1)$ paths; therefore, an unauthorized user has to compromise all the m paths before he or she can decrypt the slice. Moreover, choosing optimal path for each share is achieved by locating the path that offers the minimum expected video distortion. The controller at the source node performs a global optimization to find the optimal coding parameter and the path so that the expected video distortion is minimized. The following points illustrate how our scheme works:

- After the controller obtains all required information such as the global network topology and channel state information, the controller at the source starts sending the first share by performing a global optimization to locate the optimal path that gives the minimum expected distortion with optimal coding parameter. The controller keeps this process until it transmits the $(T-1)$ shares.
- To meet the shares allocation requirements, the controller will exclude all the optimal paths that have been used to send $(T-1)$ shares from sending any more shares and look for a new optimal paths to deliver the rest of the shares $(N-T+1)$. An illegal user has to compromise the entire paths before he or she can obtain an access to the slice of those shares.
- In case of P-frame slice, the secret image sharing algorithm is not applied. Therefore, the slices are transmitted over paths that give the minimum expected video distortion.

Then, the controller chooses the optimal path and coding parameter for each share so that the expected total distortion of the video frame is minimized. To secure the slice of those shares, the shares are distributed between the optimal paths according to shares allocation scheme. The slice of the P-frame $s_{z,t}$ does not go over a secret image sharing as we explained earlier. The controller selects the optimal path and coding parameter for each slice so that the expected total distortion of the video frame is reduced. To playback the frame correctly, all slices of a frame have to reach the destination before a frame deadline. In case of I-frame, each share is packetized into one packet and in case of P-frame each slice is packetized into one packet.

2.1. Secret image sharing

Recently, secret image sharing schemes have been adopted to protect secret images from being compromised. The concept of secret image sharing is that the secret image A is divided into N unreadable image (shares), and the generated shares are individually transmitted to authorized recipients. The secret image A can be retrieved by at least a threshold number of shares T. This is known as a threshold

DOI: 10.1002/sec

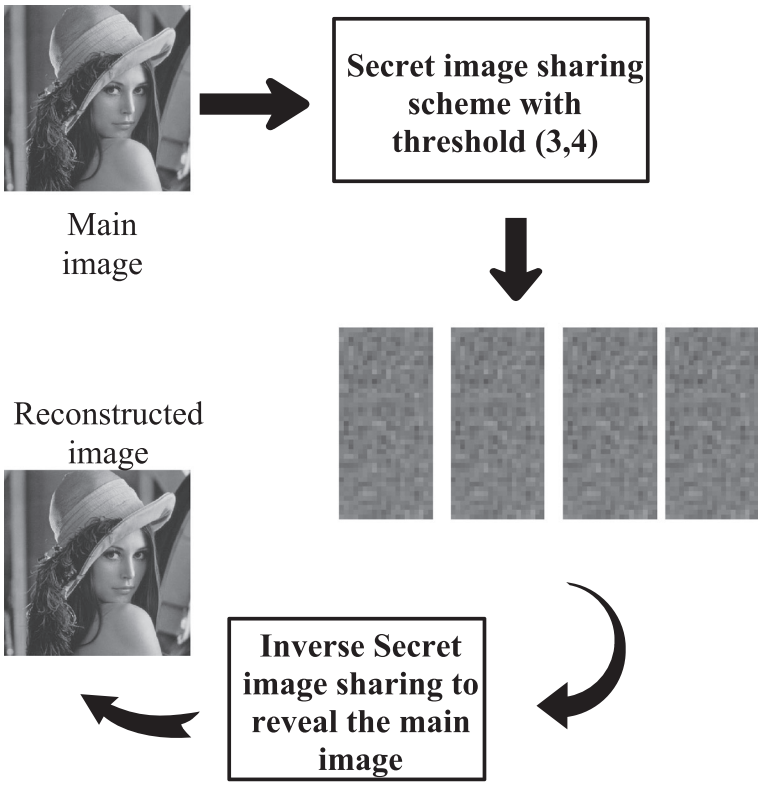

Figure 1. A secret image sharing with a threshold ($T = 3, N = 4$).

scheme (T, N). Figure 1 illustrates a secret image threshold scheme with a threshold ($T = 3, N = 4$). This threshold means that three shares from four shares are required to the secret image. In [21], we showed that losing one pixel in a share results in losing T pixels in the reconstructed image. As a result, the number of redundant shares $(N - T)$ has to be increased to mitigate this loss. This overhead is not efficient in terms of bandwidth and energy for resource-constrained systems such as WSNs. To reduce the secret image sharing overheads, we propose encrypting the I-frame by using secret image sharing. The reason for imposing the secret image sharing in the I-frame (always the first frame in GOP) is that decoding P-frames cannot be achieved without reference to the I-frame. Therefore, encrypting I-frame results in encrypting the P-frame. The resulted shares are distributed between paths according to the shares allocation constraints.

2.2. Shares allocation scheme

In order to meet the maximum security level, the produced shares after applying the secret image sharing in the I-frame slice have to be distributed among disjoint paths. Assume that we have assigned m disjoint paths to deliver N shares to the destination. In [2], the shares allocation scheme indicates that the share used to send $T - 1$ shares must not be used to send the rest of the shares in order to achieve the required security level. In such a case, unauthorized users must compromise all paths in order to decrypt the message. However, the drawback of this scheme is that the disjoint paths have to be determined before the shares are transmitted. Sending shares over already determined paths may not guarantee the video quality. To make sure that shares are sent over disjoint paths in our scheme, after transmitting $T - 1$ shares to the destination, the controller will avoid using these paths and look for new paths when it transmits the rest o the shares.

2.3. Routing metric

When the video frames are sent over a multihop wireless network, those frames will suffer from distortion that is introduced by source coding, channel error, and queuing loss. These errors are typically concealed using an error concealment technique. Error concealment techniques use spatial and temporal correlations in video frames so that lost pixels in lost slices are compensated by the pixels from the received slices of the existing frame or a preceding frame. To aid our discussion, Table I sums up the major notation used in this paper.

Let f_n^i represents the original value of pixel i in frame n, and let $\hat{f}_n^i$ and $\tilde{f}_n^i$ represent its encoded and decoded pixels, respectively. Because of possible packet loss in the channel, $\tilde{f}_n^i$ can be modeled at the encoder side as a random

Table I. Major notation used in this paper.

n	Video frame
M	Number of frames in the video sequence
s	Slice of a video frame
K	Number of slices in the video frame
N	Number of shares sh of slice s
T	Secret sharing threshold number
sh	Share of a slice s
m	Number of disjoint paths
f_n^i	Original value of pixel i in frame n
$\hat{f}_n^i$	Encoded value of pixel i in frame n
$\tilde{f}_n^i$	Decoded value of pixel i in frame n
D_{MB}	Overall expected decoder distortion in one macroblock
d_n^i	Distortion of pixel
P	Packet loss rate
mv	Motion vector
$\hat{e}_n^i$	Quantized residual where inter-coded pixel i is predicted from motion vector $i+mv$ in the previous frame
$Q_{z,t}$	Source coding parameters that can be assigned to a slice t of a frame z.
$Q_{z,t,j}$	Source coding parameters that can be assigned to a share j of a slice t.
$\mathbb{Q}_s$	Set of all admissible values of a slice
$\mathbb{Q}_{sh}$	Set of all admissible values of a share sh
$\varphi_{z,t}, \varphi_{z,t,j}$	Transmission paths of slice and the transmission path of share, respectively
$E[D_{z,t}]$, $E[D_{z,t,j}]$	Expected video distortion of a slice and expected video distortion of a share, respectively
$\Delta_{z,t}, \Delta_{z,t,j}$	Slice delay and share delay, respectively
NO	Set of nodes
LI	Set of links
h	Number of hops in a path
$R_{a-1,a}$	Transmission rate
$e_{a-1,a}$	Packet error probability of the link
$\bar{R}_{a-1,a}$	Effective transmission rate of the link
$P_{a-1,a}^{error}$	Packet error rate due to channel error
P_{a-1}^{delay}	Packet loss rate due to queuing loss
$e_{a-1,a}$	Packet error probability over a wireless link
$[T_{a-1,a}]$	First moment of the service time
$[(T_{a-1,a})^2]$	Second moments of the service time
$E[W_{a-1,a}]$	Waiting time for the packet in the queue
$\theta_{a-1,a}$	Maximum retransmission attempt
L	Packet length
P_{comp}	Probability that the slice reaches the destination with interception
C_{comp}	Probability of compromising at least one node in a path
k_i	Path compromising probability
c_i	Number of shares traverse over path $\varphi_{z,t,j}$

variable. In recursive optimal per-pixel estimate (ROPE) approach, the distortion of MBs D_{MB} is defined as the overall expected decoder distortion in one MB [22].

$$D_{MB} = \sum_{i \in MB} d_n^i \tag{1}$$

$$d_n^i = E\left\{\left(f_n^i - \tilde{f}_n^i\right)^2\right\} \tag{2}$$

$$d_n^i = \left(f_n^i\right)^2 - 2.f_n^i.E\left\{\tilde{f}_n^i\right\} + E\left\{\left(\tilde{f}_n^i\right)^2\right\} \tag{3}$$

It is clear from Equation (3) that the first and the second moments $E\left\{\tilde{f}_n^i\right\}$, $E\left\{\left(\tilde{f}_n^i\right)^2\right\}$ are required to calculate the distortion d_n^i of a pixel. ROPE proposed an optimal recursive algorithm to accurately estimate these two moments for each pixel in a frame. It is assumed that the encoder is aware of the packet loss rate P. An error concealment scheme is assumed to be known at the source and the destination, respectively, and will be employed when a packet is lost. ROPE also supposes that if an MB is lost, the decoder copies reconstructed MB from the previous frame to conceal the loss.

The method of ROPE is explained by the following equations [22]:

(1) Expected distortion of pixel in the intra-MB

$$E\left\{\tilde{f}_n^i\right\} = (1-P).\left(\hat{f}_n^i\right) + P.E\left\{\tilde{f}_{n-1}^i\right\} \tag{4}$$

DOI: 10.1002/sec

$$E\left\{\left(\tilde{f}_n^i\right)^2\right\} = (1-P).\left(\hat{f}_n^i\right)^2 + P.E\left\{\tilde{f}_{n-1}^i\right\} \quad (5)$$

(2) Expected distortion of pixel in the inter-MB

$$E\left\{\tilde{f}_n^i\right\} = (1-P).\left(\hat{e}_n^i + E\left\{\tilde{f}_{n-1}^{i+mv}\right\}\right) + P.E\left\{\tilde{f}_{n-1}^i\right\} \quad (6)$$

$$E\left\{\left(\tilde{f}_n^i\right)^2\right\} = (1-P)\cdot\left(\left(\hat{e}_n^i\right)^2 + 2\,\hat{e}_n^i E\left\{\tilde{f}_{n-1}^{i+mv}\right\}\right) + P.E\left\{\left(\tilde{f}_{n-1}^i\right)^2\right\} \quad (7)$$

The term $\hat{e}_n^i$ represents the quantized residual where inter-coded pixel i is predicted from pixel $i + mv$ in the previous frame where mv is the motion vector.

The routing process in our scheme is based on the calculation of the expected distortion at the source node. ROPE was designed to estimate the expected distortion in the case where the I-frame slices are transmitted through generic wireless networks. In this work, we consider the scenario of sending shares of a video slice for an I-frame over multihop wireless networks. Therefore, we need to modify the ROPE model for I-frames. For P-frames, the calculation will be the same as in [22].

$$D_{sh} = \sum\nolimits_{i\in share} d_n^i \quad (8)$$

$$d_n^i = \left(f_n^i\right)^2 - 2f_{n,sh}^i.E\left\{\tilde{f}_{n,sh}^i\right\} + E\left\{\left(\tilde{f}_{n,sh}^i\right)^2\right\} \quad (9)$$

$$E\left\{\tilde{f}_{n,sh}^i\right\} = (1-P).\hat{f}_{n,sh}^i + P.E\left\{\tilde{f}_{n-1,sh}^i\right\} \quad (10)$$

$$E\left\{\left(\tilde{f}_{n,sh}^i\right)^2\right\} = (1-P)\cdot\left(\hat{f}_{n,sh}^i\right)^2 + P\cdot E\left\{\tilde{f}_{n-1,sh}^i)^2\right\} \quad (11)$$

D_{sh} is the share distortion.When a share is lost, it is concealed by a share of last frame. $f_{n,sh}^i$ represents the original value of pixel i in share sh of frame n. $\tilde{f}_{n-1,sh}^i$ represents the decoded value of pixel i in share sh of previous frame $n-1$. $\hat{f}_{n,sh}^i$ represents the encoded value of pixel i in share sh of frame n.

2.4. Problem formulation

Let K be the number of slices in a frame; M is the number of frames in the video sequence that needs to be encoded and transmitted. $Q_{z,t} \in \mathbb{Q}_s$ is the source coding parameters that can be assigned to a slice t of a frame z. $Q_{z,t,j} \in \mathbb{Q}_{sh}$ is the source coding parameters that can be assigned to a share j of slice t and $\left|\mathbb{Q}_s\right| = \left|\mathbb{Q}_{sh}\right| = g$. A packet represents a share $sh_{z,t,j}$ in the case of I-frame and represents a slice $s_{z,t}$ in the case of P-frame. Let $\varphi_{z,t}, E[D_{z,t}]$, and $\Delta_{z,t}$ denote the transmission path of slice, the expected video distortion of slice, and the slice delay, respectively. Let $\varphi_{z,t,j}$, $E[D_{z,t,j}]$, and $\Delta_{z,t,j}$ be the transmission paths of share, the expected video distortion of share, and the share delay, respectively. c_i denotes number of shares that are transmitted over the path $\varphi_{z,t,j}$.

Both the expected video distortion and the delay of a packet depend on the choices of path and video coding.

The problem is to choose the optimal paths and coding parameters for each slice and shares of the video frame so that the total expected video distortion is minimized, under the packet delay and share allocation constraints in Equation (12).

The problem can be written as follows:

(1) In the case of I-frame

$$min_{\varphi_{z,t,j}\,Q_{z,t,j}} \sum_{z=1}^{M}\sum_{t=1}^{K}\sum_{j=1}^{N} E[D_{z,t,j}] \quad (12)$$

$$st: max\left\{\Delta_{z,1,j=\{1,2,..,N\}}, .., \Delta_{z,K,j=\{1,2,..,N\}}\right\} < \Delta$$

$$\begin{pmatrix} N-(c_1+c_2+...+c_{m-1}) < T \\ c_1+c_2+...+c_m = N \end{pmatrix}$$

(2) In the case of P-frame

$$min_{\varphi_{z,t}\,Q_{z,t}} \sum_{z=1}^{M}\sum_{t=1}^{K} E[D_{z,t}] \quad (13)$$

$$st: max\left\{\Delta_{z,1}, \Delta_{z,2}, .., \Delta_{z,K}\right\} \le \Delta$$

It is important to mention that the optimization is achieved one frame at a time. The first constraint states that all shares of I-slice have to arrive at the destination before a frame delay deadline Δ. To explain the second constraint clearly, assume we have a number of paths m between the source and the destination, and each path $\varphi_{z,t,j}$ can be used to transmit a number of shares c_i. The total number of shares that travels over those paths is N. The constraint says we cannot assign more than $(T-1)$ number of shares to $(m-1)$ number of paths. By doing this, the adversary has to compromise all paths in order to retrieve the slice of shares, which cannot be retrieved by having a number of shares less than T.

Also the inter-slices have to arrive to the destination before the frame delay deadline Δ. We execute global optimization for N number of slice shares to find the optimal path for each share.

3. SOLUTION PROCEDURES

Before transmission, secret sharing scheme with a threshold (T, N) is applied on each slice $s_{z,t}$ of I-frame to produce N shares $\{sh_{z,t,1}, sh_{z,t,2}, \ldots, sh_{z,t,N}\}$. Each share can be coded into possible packets $\{sh_{z,t,i}^1, sh_{z,t,i}^2, ..., sh_{z,t,i}^g\}$ using

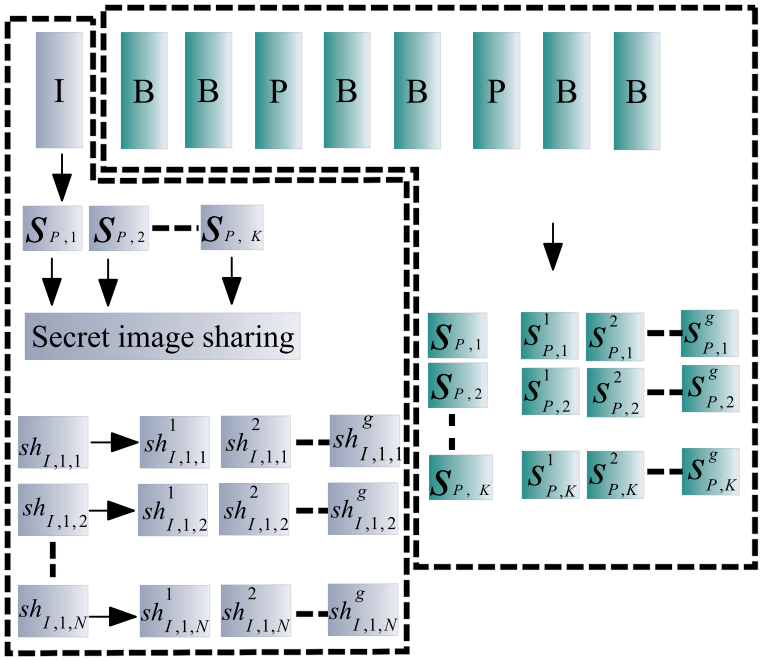

Figure 2. Applying secret image sharing for intra-slice.GOP, group of pictures.

the g coding options. The $sh_{z,t,i}$ is transmitted by sending any of g shares $\{sh^1_{z,t,i}, sh^2_{z,t,i}, ..., sh^g_{z,t,i}\}$. In case of P-frame, the slice $s_{z,t}$ can be compressed into possible version$\{s^1_{z,t}, s^2_{z,t}, ...s^g_{z,t}\}$. We can transmit the $s_{z,t}$ by sending any of $\{s^1_{z,t}, s^2_{z,t}, ...s^g_{z,t}\}$. Figure 2 illustrates applying the secret image sharing on the I-frame slice.

The problem of jointly selecting the pairs of coding parameters and transmission paths for a group of slices can be described as follows: First, the controller determines the optimal paths (an optimal path is the path that gives the minimum expected video distortion) for each possible packets, based on the network topology and link status information. Then, the controller executes a global optimization for a group of packets of a frame to determine the optimal coding parameters and optimal paths for each packet using a dynamic programming [11].

After transmitting $(T-1)$ packets of I-frame (a packet in I-frame represents a share), the controller will exclude all paths that have been used to send $T-1$ shares from sending any more shares so that it meets the shares allocation scheme conditions in second constraints in Equation (12). The controller looks for new optimal paths to transmit the rest of shares $(N - T + 1)$ through new optimal paths. As a result, an illegal user has to compromise all optimal paths that have been used to send the N shares before he or she can reconstruct the slice of those shares. By doing this, we have protected I-frame from any security threat. As we previously mentioned that decoding all other frames (P-frames) in the video sequence depends on decoding the I-frame, thus securing the I-frame makes it impossible for unauthorized users to read the P-frames.

However, with the expected distortion as routing metric, different packets may have different transmission paths and lead to different slice and share distortions. This distortion is caused by signal corruption, packet dropping, and source coding and error concealment. Therefore, the controller needs to evaluate the packet loss probability due to channel impairments and the packet dropping that occurs when packet waits longer than the maximum deadline at intermediate nodes. Calculating this packet loss rate is essential to measure the expected distortion in Equations (1) and (8) and then solving the optimization problem in Equations (12) and (13).

A directed graph G (**NO**, **LI**) denotes the network topology, where **NO** is the set of nodes and **LI** is the set of directed links. The topology is assumed to be static. The path between the video source and the destination consists of a number of links $\{li_{1,2}, li_{2,3}, ..., li_{h-1,h}\}$ where h is the number of hops in the path. The link $li_{a-1,a}$ can be characterized by the following factors: $R_{a-1,a}$ symbolizes the transmission rate of the corresponding modulation and coding scheme. The packet error probability of the link is $e_{a-1,a}$. The effective transmission rate of a link is $\bar{R}_{a-1,a}$.

The delay deadline of the packet is Δ_{a-1}. We suppose that each node has a buffer that keeps coming packets that cannot be sent immediately. Therefore, they will be queued in the buffer. As we just mentioned, packet loss has two primary sources: channel error and queuing loss. In a WSN, the channel error is managed at a medium access control (MAC) layer by retransmitting the packet until it arrives to the destination, or it is discarded (if the number of retransmission surpasses

the retransmission limits). We denote this error as $P^{error}_{a-1,a}$. Queuing loss occurs when queuing delay of a packet goes above the delay limit. Therefore, the packet would be discarded. We symbolize this error as P^{delay}_{a-1}. Without loss of generality, packet loss probability at no_a can be written as

$$P^{total}_{a} = 1 - \left(1 - P^{error}_{a-1,a}\right)\left(1 - P^{delay}_{a-1,a}\right) \tag{14}$$

In the following section, we discuss the calculations of packet error rate due to channel error and packet loss rate due to queuing loss.

3.1. Packet error rate due to channel error

The estimation of packet loss rate and the effective transmission rate for a wireless link depend on the type of the modulation and coding scheme used in the wireless channel. The packet error probability over a wireless link can be estimated with a sigmoid function as in [4].

$$e_{a-1,a} = \frac{1}{1 + e^{\xi(SINR-\delta)}} \tag{15}$$

The effective transmission rate (goodput) for a wireless link between node no_{a-1} and no_a is given by

$$\bar{R}_{a-1,a} = \frac{R_{a-1,a}(w)}{1 + e^{-\xi(SINR-\delta)}} \tag{16}$$

Signal to interference plus noise ratio (SINR) is the signal to interference noise ratio. ζ and δ are constants related to channel modulation and coding schemes. $R_{a-1,a}, w$ are the transmission rate and the bandwidth of the link, respectively.

As wireless network is prone to error, nodes utilize acknowledgment packets to treat this packet loss and error issues. First, a node contends for the MAC to send its packets. Also, the node has to resend the packet until it receives acknowledgement from the destination or it surpass the number of retransmission limit $\theta_{a-1,a}$. The expected packet loss probability due to retransmission error can be written as [23]

$$P^{error}_{a-1,a} = 1 - \sum_{i=1}^{\theta_{a-1,a}} e_a^{i-1}\left(1 - e_a^{i}\right) \tag{17}$$

The maximum retransmission attempt $\theta_{a-1,a}$ can be written in the following equation:

$$\theta_{\mathbf{a}-1,\mathbf{a}} = \left\lfloor \frac{\bar{R}_{a-1,a}(\Delta - \Delta_{a-1})}{L} \right\rfloor - 1 \tag{18}$$

where Δ is budget deadline, L is the packet length, and Δ_{a-1} is the current delay of a packet after it is buffered in a node no_{a-1} queue.

3.2. Packet loss rate due to queuing

Here, we intend to derive the packet loss probability due to queuing delay. First, we need to estimate the packet queuing delay $E[W_{a-1,a}]$. The queue in the relay node can be represented as M/G/1 queues with packets arrival rate π_a. The packet arrival rate can be locally evaluated at a node counting and taking the mean of the total number of incoming packets over a given period of time. Each packet will be resent until it is successfully delivered to the destination or rejected if the number of retransmission attempts surpasses the retries limit. The time that a packet lasts at a relay node depends on the effective transmission rate (goodput) and error probability of the wireless link between the current node and its neighbor node.

To estimate the expected waiting time that the packet lasts in a queue of a node, processing time must be calculated. Processing time occurs due to primary sources. The approximated processing time $T_{a-1,a}$ for a packet over the link $l_{a-1,a}$ can be expressed by [24]

$$[T_{a-1,a}] = \frac{L\left(1-\left(1-P^{error}_{a-1,a}\right)^{\theta_{a-1,a}+1}\right)}{R_{a-1,a}\left(1-P^{error}_{a-1,a}\right)} \tag{19}$$

$$[(T_{a-1,a})^2] = \frac{L^2\left(1-\left(1-P^{error}_{a-1,a}\right)^{\theta_{a-1,a}+1}\right)}{\left(R_{a-1,a}\right)^2\left(1-P^{error}_{a,a-1}\right)^2} \tag{20}$$

The waiting time for a packet in the queue of a node n_{a-1} can be written by

$$E[W_{a-1,a}] = \frac{\pi_a E\left[\left((T_{a-1,a})^2\right)\right]}{2\left(1 - \pi_a\left[T_{a-1,a}\right]\right)} \tag{21}$$

Now, the packet dropping can be calculated easily by [25]

$$\begin{aligned} P^{delay}_{a-1} &= \mathrm{Prob}(E[W_{a-1,a}] + \Delta_{a-1} > (\Delta)) \\ &= \pi_a.\left[T_{a-1,a}\right]\exp\left(-\frac{(\Delta-\Delta_{a-1})\pi_a.\left[T_{a-1,a}\right]}{E[W_{a-1,a}]}\right) \end{aligned} \tag{22}$$

The loss probability of packet traveling over a link can be expressed as

$$P^{total}_{a} = 1 - \left(1 - P^{error}_{a-1,a}\right)\left(1 - P^{delay}_{a-1}\right) \tag{23}$$

After calculating the packet loss probability in Equation (23), the controller can determine the expected distortion in Equations (12) and (13), and then it can decide which path the packet should take to the destination. Here, we plan to see how the number of shares and the number of disjoint paths are affecting the compromised

probability of a slice. A slice can be compromised if at least T of its shares are available to an adversary; before we begin this, we make the following assumptions:

- A path consists of a number of nodes $no_1, no_2, ..., no_h$. A share of slice that travels through this path can be compromised when any node in this path is compromised. In other words, the path is compromised when one or more nodes in this path are compromised.
- According to the shares allocation constraints mentioned previously, the shares are sent to the destination through node-disjoint paths. Let us say a video slice S is split into N shares, and those shares are delivered through m a number of disjoint paths. The shares allocation scheme distributes shares between these disjoint paths so that the number of shares assigned to $(m-1)$ paths become less than T shares; therefore, the slice is only compromised when all the m paths from are compromised.
- The slice is compromised when all the disjoint paths used to send the shares of the slice are compromised. The probability that the slice reaches the destination with interception is

$$P_{comp} = 1 - \prod_{i=2}^{m} (1 - k_i) \tag{24}$$

The term k_i represents a path compromising probability. According to the first assumption, the probability of compromising a path increases as the number of nodes in the path increases.

Assume the number of hops in a path is h_i, and $x_{i,j}$ is the node compromising probability. The source and the destination nodes are assumed to be secure. As mentioned previously, to compromise a slice, all paths m that shares of slice take to traverse to the destination have to be compromised. The probability of compromising at least one node in a path(is also equal to the probability of compromising a path) is given by

$$C_{comp} = k_i = \sum_{i=1}^{h} \binom{h}{i} x_{i,j}^{i} (1 - x_{i,j})^{1-i} \tag{25}$$

Substituting Equation (25) in Equation (24) gives the slice compromising probability P_{comp}:

$$\begin{aligned} P_{comp} = 1 - [(1 - \sum_{y=1}^{h_1} \binom{h_1}{y} x_{1,j}^{y} (1 - x_{1,j})^{h_1 - y}) \times \\ (1 - \sum_{y=1}^{h_2} \binom{h_2}{y} x_{2,j}^{y} (1 - x_{i,j})^{h_2 - y}) \\ \times (1 - \sum_{y=1}^{h_L} \binom{h_L}{y} x_{m,j}^{y} (1 - x_{i,j})^{h_m - y})] \end{aligned} \tag{26}$$

Knowing the number of shares is really important in order to minimize transmission overhead and simultaneously maximize the security level. Sending redundant shares is not efficient in terms of security. By using Equation (27), we can estimate the number of shares that we must send to meet some defined security levels P_{th}.

$$\sum_{v=T}^{N} \binom{N}{v} x^{v} (1 - x)^{N - v} < P_{th} \tag{27}$$

The optimal number of shares that we must send to the destination is T because there are no redundant shares, but sometimes we need to increase the number of shares to combat some random losses. The source node should choose an appropriate value of (T, N) and transmit different numbers of shares through difference paths according to their hop counts and path quality.

3.3. Finding optimal path

The process of choosing optimal paths in our scheme differs from the framework presented in [11]. In our scheme, the controller distributes N slice shares between disjoint paths according to the shares allocation scheme and expected video distortion. To ensure shares are transmitted over a disjoint paths, the controller stops choosing the paths that are utilized to transmit $T-1$ slice shares and uses different optimal paths to send the rest of the slice shares N–T+1. In Figure 3, each arc of the graph represents a link.

The algorithm determines the path that gives the minimum expected distortion to choose as optimal path. The controller calculates the optimal path and after receiving all the feedback information such as average packet arrival rate, SINR, and number of shares (that is sent by each node) from all other nodes.

First, all nodes are labeled with infinity because no path is discovered. As the algorithm carries on and paths are discovered, the labels may change, producing better paths. A label can be either tentative or permanent. Initially, all labels are tentative. When it is found that this label gives the minimum distortion possible from the source to that node, then the node becomes permanent (working node) and never changes again.

The labels on the arcs include the expected distortion, the packet loss probability, and the packet delay incurred by packet traversing over the path [25]. The reason behind keeping the values of packet delay and packet loss is to increase the speed of the calculation of path determination. When a controller checks all the nodes adjacent to the working node (working node is the node that has the shares to transmit), the tentative labels are modified if possible; the entire graph is searched for the tentatively labeled node with the smallest expected distortion. This node is made permanent and becomes the new working node for the next round.

Each arc is labeled with infinite expected distortion, packet loss probability, and packet delay as shown in the undirected graph of Figure 3(a). We intend to locate the

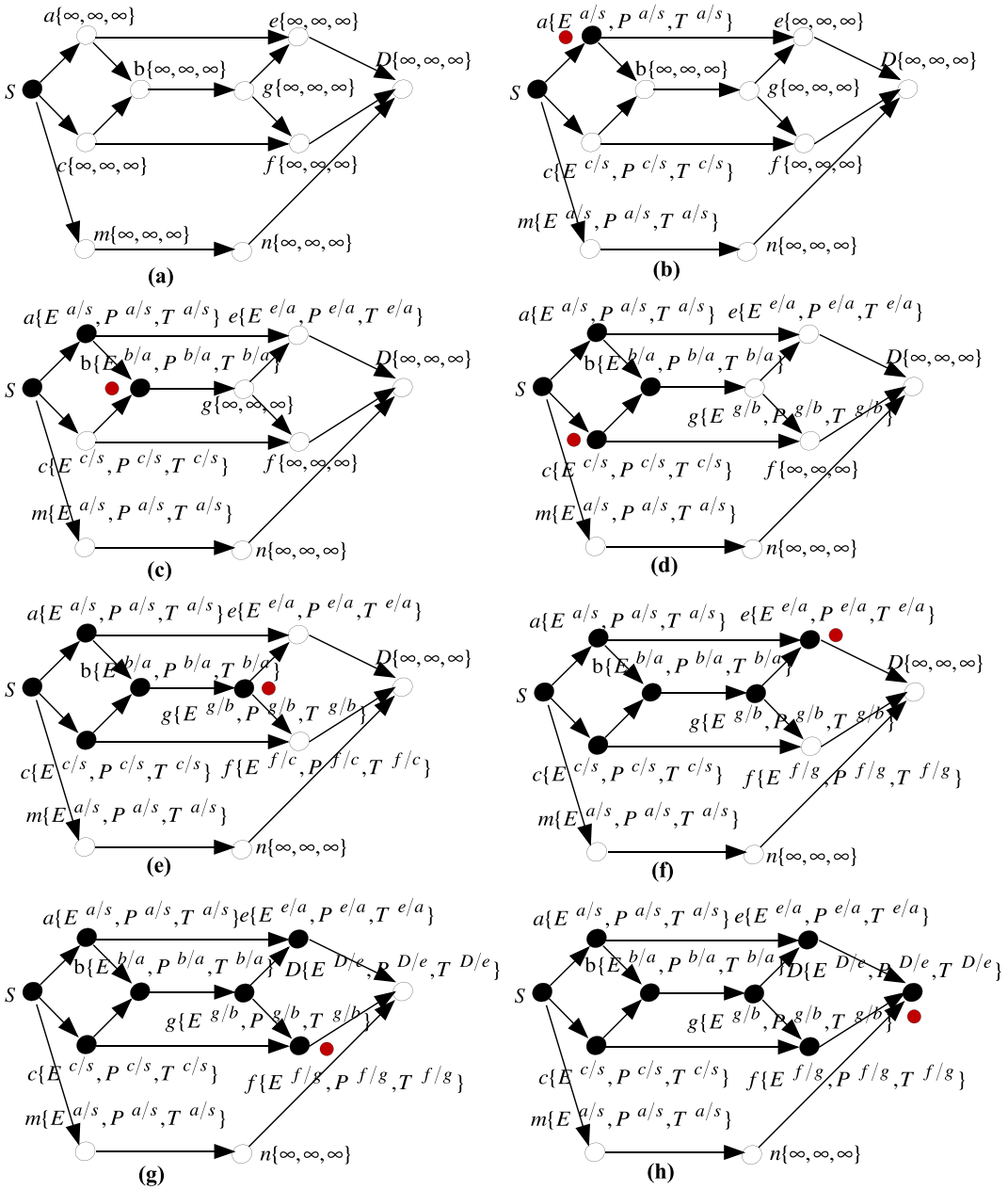

Figure 3. The steps used in computing the optimal path from The source node (S) to destination node (D). (a) is the first step and (f) is the last step. The red filed-in circle indicate the working node.

path with the minimum expected distortion from S (source) to D (destination). At first, we assume that we have four shares $\{sh_{z,t,1}, sh_{z,t,2}, sh_{z,t,3}\}$ with a secret sharing threshold ($T = 2, N = 3$) of the I-frame slice and no path has been discovered between the source and the destination; therefore, all nodes are labeled with an infinite expected distortion, infinite packet loss probability, and infinite packet delay. We begin with the first slice share $sh_{z,t,1}$ by marking node S as permanent (the working node) marked. Then, the controller checks every node adjacent to S and labels each one with the expected distortion, packet loss probability, and packet delay. When a node is remarked, we also label it with the node from which the route was constructed so that we can rebuild the final path. Here, nodes a and c are both labeled with $\left\{E^{a/S}, P^{a/S}, T^{a/S}\right\}$, $\left\{E^{c/S}, P^{c/S}, T^{c/S}\right\}$, and $\left\{E^{m/S}, P^{m/S}, T^{m/S}\right\}$, respectively.

After testing all the tentative nodes in the whole graph a, b, c, e, g, f, m, n, and D, the node that gives the minimum expected distortion for share $sh_{z,t,1}$ becomes the new

permanent node. We assume node a gives the minimum distortion; therefore, the first share $sh_{z,t,1}$ chooses node a to be the new working node and is marked by a red filled-in circle as shown Figure 3(b).

Now, the first share $sh_{z,t,1}$ has arrived to node a, and the controller now starts to calculate the expected distortion at nodes e and b, respectively. To compute the loss probability $P^{e/a}$ that share $sh_{z,t,1}$ would have at node e via the path $S \rightarrow a \rightarrow e$, the controller can directly retrieve the stored value of $P^{a/S}$ in the label of working node a and calculate $P^{e/a} = 1 - \left(1 - P^{a/S}\right)\left(1 - P_e^{total}\right)$ instead of recalculating $P^{e/a}$. The same thing is carried out for the delay $T^{e/a}$. Then, the controller checks all the left tentatively label nodes, which are b, c, e, g, f, m, n, and D. We assume node b has the minimum expected distortion among the left tentative nodes in Figure 3(c); therefore, it is labeled with $\left\{E^{b/a}, P^{b/a}, T^{b/a}\right\}$. With the same procedures as before, the controller would calculate the expected distortion, the packet loss probability, and the packet delay at node g; therefore, it is relabeled with $\left\{E^{g/b}, P^{g/b}, T^{g/b}\right\}$ as presented in Figure 3(d). Then, the controller continues to look for the node that has the minimum expected distortion between all the other nodes (at this step, they are c, e, g, f, m, n, and D). We assume that $E^{c/S}$ has the minimum distortions; therefore, node c is labeled with a red filled-in circle and becomes the new working node. Then, the controller continues to compute the expected distortions that share $sh_{z,t,1}$ would obtain at the two coming nodes b and f of node c.

Let $E^{b/c}$ be the expected distortion that share $sh_{z,t,1}$ would obtain at node b if the shares takes the path $S \rightarrow c \rightarrow b$. We assume that $E^{b/c} > E^{b/a}$, then node b will not be relabeled with $E^{b/c}$. Node f will be relabeled with $\left\{E^{f/c}, P^{f/c}, T^{f/c}\right\}$. We assume that node g has the smallest distortion between the unchecked nodes g, f, e, m, n, and D. Thus, node g becomes the new working node as illustrated in Figure 3(e). The controller then begins to compute the expected distortions that share $sh_{z,t,1}$ would obtain at nodes e and f of node g if the share goes along the paths $S \rightarrow a \rightarrow b \rightarrow g \rightarrow e$ and $S \rightarrow a \rightarrow b \rightarrow g \rightarrow f$, respectively.

Let $E^{e/g}$ be the expected distortion of share $sh_{z,t,1}$ at node e. Assuming that $E^{e/g} > E^{e/a}$, then the label of node e will not be updated. For node f, we assume that $E^{f/g} < E^{f/c}$, then the node will be labeled with $E^{f/g}$ at node f. Then, the controller continues to look for the node with the minimum labeled distortion between the unchecked labeled nodes e, f, m, n, and D. We assume node e has the minimum expected distortion; therefore, node e is labeled as new working node in Figure 3(f).

We assume that node D is relabeled with the expected distortion $E^{D/e}$ that the share would obtain at node D if it travels through the path $S \rightarrow a \rightarrow e \rightarrow D$. Then, the controller starts finding the next working node between the last four nodes f, m, n, and D. We assume that the label of node f has a smaller expected distortion value than the label of node D $E^{D/e} > E^{f/g}$; thus, node f is labeled as the new working node in Figure 3(g). Then, the controller starts calculating the expected distortion $E^{D/f}$ that the share would obtain if it proceeds via the path $S \rightarrow a \rightarrow b \rightarrow g \rightarrow f \rightarrow D$.

We assume that $E^{D/e} < E^{D/f}$; therefore, node D does not need to be relabeled. Then, the controller starts finding the next working node between the last three nodes m, n, and D. Node D has the minimum distortion and is labeled as working node in Figure 3(h). Once node D becomes marked as permanent, the proposed routing algorithm ends looking for optimal path for the share $sh_{z,t,1}$. By recovering back all the stored previous hope nodes in the labels from destination node D to source node S. The optimal end-to-end path is $S \rightarrow a \rightarrow e \rightarrow D$.

Once $(T-1)$(which in our case is $(T-1) = 1$) shares are delivered to the destination, then the controller will avoid using these paths (we indicate them with the blue dot box) that are used to transmit the $(T-1)$ shares, and it chooses new paths to deliver the rest of the shares $(N-T+1)$. We assume the next shares $sh_{z,t,2}$ take path $S \rightarrow c \rightarrow b \rightarrow g \rightarrow f \rightarrow D$ and $S \rightarrow m \rightarrow n \rightarrow D$, respectively. By doing this, we ensure that N shares will be sent over disjoint paths that satisfy the shares allocation condition. Also, it is mentioned that there are redundant shares $(N-T)$. This redundancy is not efficient in terms of security and energy, so that in our design, after we send T shares, the controller always keeps testing the video quality metric and if it is higher than a defined threshold. Then, there will be no need to send any more shares.

4. EXPERIMENTAL RESULTS

4.1. Verification of the proposed distortion model

We firstly investigate the accuracy of our proposed expected distortion model in Section 2. We used Video Distortion Analysis Tool (VDAT) to measure the video distortion; VDAT is a research tool to statistically investigate the video distortion in many wireless channels [26]. We considers the Y component of the first 100 frames of foreman sequence in quarter common intermediate format (176×144). Each frame has nine slices. All frames are coded as P-frame except the first and third frames are coded as I-frame. The reason behind coding the third frame as I-frame and imposing the secret image sharing on this frame is to see how the proposed distortion model behaves when a lost share in the current frame is concealed from a share of previous frame. We cannot do that if the first frame is coded as I-frame because there will be no previous shares to compensate the lost shares in the current frame. A secret image sharing with a threshold $(T = 2, N = 4)$ is applied on each slice of the I-frame to generate four shares, and at least two shares are required to recover the slice. We assume the first frame is received correctly. Our expected distortion model assumes that if a share is lost in the current frame, it is

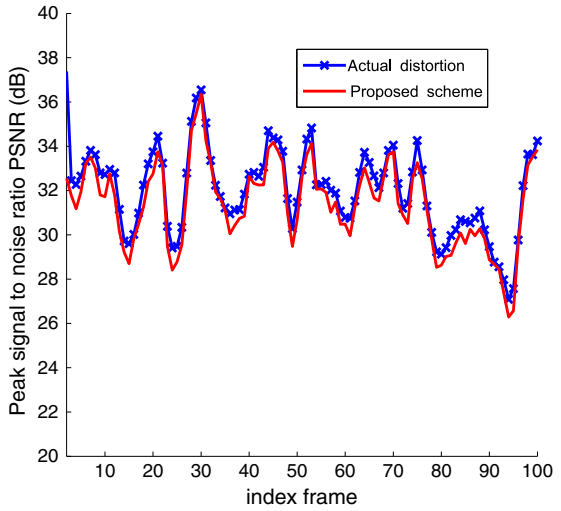

Figure 4. Comparison between actual distortion model with the proposed distortion model.

concealed by a share of previous frame. The slices and shares are transmitted over a lossy channel. The end-to-end distortion is calculated for each video frame sent over a single-hop path. We compare our proposed video distortion model with the actual distortion. Figure 4 shows that the proposed distortion model gives very close results to the actual distortion.

4.2. Routing performance analysis

Our experimental results are based on Matlab. In our simulations, sensor networks with number of nodes 100, 50 are considered, which are randomly deployed in a square field (1000 × 1000). The Y component of the first 100 frames of foreman sequence in quarter common intermediate format (176 × 144) are coded by H.264 (JVT reference software JM 14) codec with 15 frame rate (frame per second) that results in 0.0667 s packet delay deadlines. Only the first and the third frames in the GOP (GOP has 10 frames) are coded as I-frame. Then, the coded video frames (that are resulted from H.264 JM codec) are passed through the network. The radio link between any two sensor nodes is defined by transmission range, which is assumed to be 200 m. A direct acyclic graph, modeled connectivity structure is assumed between the network nodes, is defined by a proactive routing scheme just as OLSR is supposed to be reachable to the controller. The bandwidth of each link is set to 12 MHz according to [22]. In order to incorporate the effect of noise and interference, we choose the *SINR* of each link to be 15 dB. The network topology and the corresponding channel conditions of each link are assumed to remain unchanged within the service time (here, we assume the service time is a duration of one video frame). Each link adapted its modulation and coding scheme based on the received SINR; therefore, links have different goodputs. Each frame has nine slices. Secret sharing scheme with threshold (3,5)is only applied on the I-frame slice individually. Each slice compromises of a single row of MBs. Each slice is packetized as a separate packet case of P-frame, and each share is packetized as a separate packet in the case of I-frame. We use the following quantization step size (QP) $\{4, 8, 12, 18, 22\}$ as tunable source coding parameter. We compare SQAR scheme with the following three well-known schemes for joint optimization of path routing and video coding.

- The first reference scheme is packet loss rate-based routing scheme. The mechanism of choosing the best path in this scheme is achieved by selecting the path that minimizes the average packet loss rate and taking the packet delay deadline into consideration. The source coding (QP) is selected for each frame.
- The second scheme is the packet delay-based routing scheme where the optimal path is selected to minimize the average packet delay. Also, the optimal source coding value is determined for each frame.
- The third scheme is hop count-based routing scheme. This scheme selects the path with the minimum number of hops while satisfying the constraints of packet delay deadline.

Peak signal to noise ratio of the received video is computed for each frame and averaged over all frames using the four different approaches. Figure 5(a) and (b) obviously prove that SQAR offers considerable video quality improvement over the other three reference routing schemes. This is because the SQAR, besides the security enhancement, minimizes the video distortion in finding the optimal path while achieving security enhancement. In contrast, other three routing schemes use different metrics to choose the optimal path without minimizing the received video distortion. We can also observe that increasing the number of network nodes enhances the received video quality, because the dense network has more paths to the destination and that reduce the packet delay.

4.3. Security performance analysis

Here, we perform some simulations in Matlab to evaluate the efficiency of the SQAR scheme in terms of security. We choose the following simulation parameters; the number of nodes in the network is $N = 1000$ nodes are randomly deployed on a square area of 500 × 500 m. The transmission range of each node is increased from 200 to 500 with step size 100. Increasing transmission ranges results in increasing the number of disjoint paths. The compromising probability of a slice increases as compromising probability of a node increase as illustrated

in Figure 6(a). As we mentioned, that path is compromised when any of its nodes is compromised. Therefore, we also can conclude from Figure 6(a) that the compromising probability of a slice decreases as the number of disjoint path increases. The compromising probability of a video slice increases as compromising probability of a node increases as illustrated in Figure 6(b). Also, it can be seen that compromising probability of a video slice increases as the number of redundant shares $(T - N)$ increases as illustrated in Figure 6(b). It is obvious that non-redundant message splitting pattern (where $N = T$) ensures a high-level security.

Figure 7 illustrates the trade-off between the received video quality and the secret sharing threshold. We assumed that a video slice of I-frame is divided into eight shares, so the threshold is $(T = 8)$. Therefore, at least eight shares are needed to reconstruct the video slice at the destination. Redundant shares with addition to the T shares are sent over a lossy network. Three experiments with different number of shares $(N = 9, N = 10, \text{and } N = 11)$ are conducted, and peak signal to noise ratio is measured. We can conclude from Figure 7 that as the number of redundant shares $(N - T)$ increases,

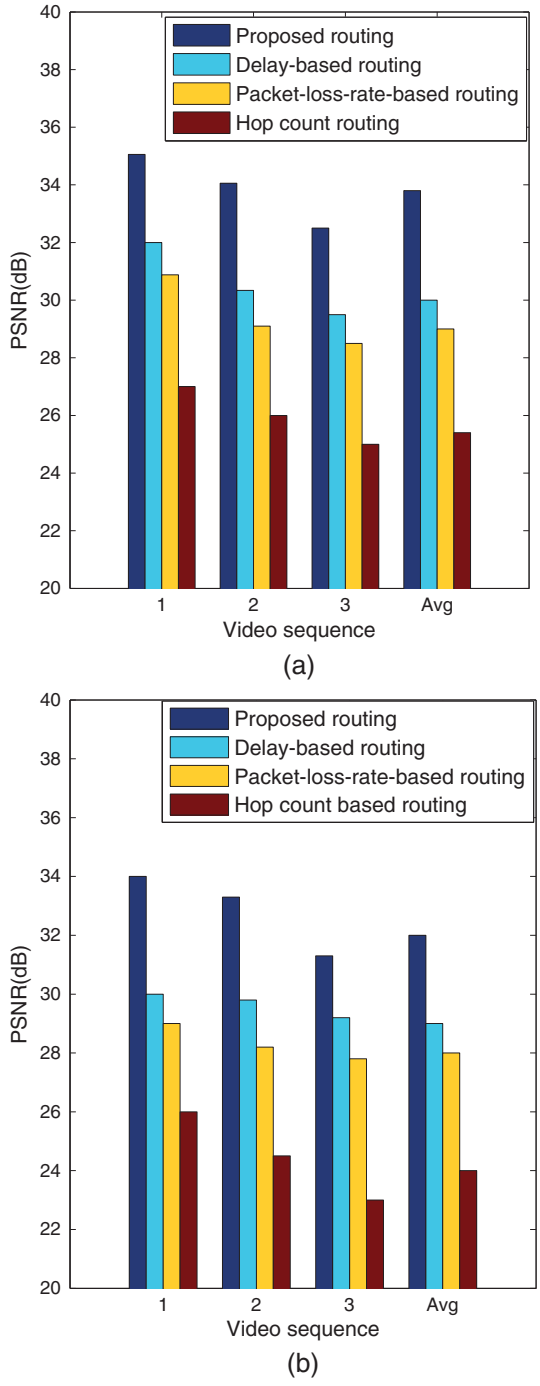

Figure 5. Average received peak signal to noise ratio (PSNR) using different routing schemes with different network size at time budget 0.0667 s: (a) 100 nodes and (b) 50 nodes.

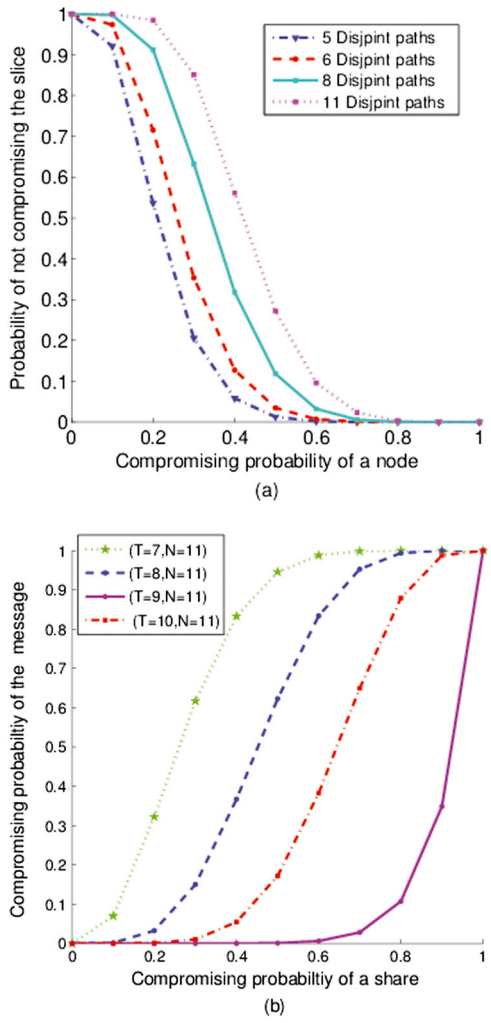

Figure 6. (a) The probability of compromising a slice due to a node compromising probability. (b) The probability of compromising a slice due to slice shares compromising probability.

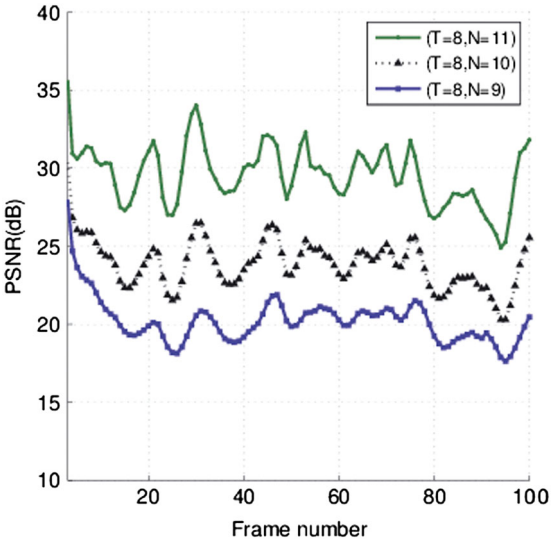

Figure 7. The trade-off between the received video quality and the secret sharing threshold. PSNR, peak signal to noise ratio.

the video quality is improved, but increasing the number of redundant shares is not secure as we explained in Figure 6(b).

5. CONCLUSION

A security and QoS aware routing scheme for WMSNs was proposed by applying the secret image sharing on the I-frames of a video sequence. The SQAR calculates the expected end-to-end distortion to find the optimal paths for every share of the I-frame and every slice of the P-frame. In addition, we developed a model that can evaluate the expected distortion for the I-frame shares sent over a multihop wireless network. QP is used as tunable source coding parameter for both the shares and the slices, and it is adapted to the selected routing path in order to enhance network resources utilization. Simulation results show that the SQAR can improve both the end-to-end video quality and the security by allocating the shares over disjoint paths.

ACKNOWLEDGEMENTS

This work was partially supported by the US National Science Foundation under grant nos. ECCS-1407882, CNS-1451629, CNS-1429120, ECCS-1401121, and IIP-1414250.

REFERENCES

1. Akyildiz IF, Melodia T, Chowdhury K. A survey on wireless multimedia sensor networks. *Computer Networks* 2007; **51**(4): 921–960.
2. Lou W, Liu W, Zhang Y, Fan Y. SPREAD: improving network security by multipath routing in mobile ad hoc networks. *Springer Wireless Networks* 2009; **15** (3): 279–294.
3. Zhou L, Chao H-C, Vasilakos A. Joint forensics-scheduling strategy for delay-sensitive multimedia applications over heterogeneous networks. *IEEE Journal on Selected Areas in Communications* 2011; **29**(7): 1358–1367.
4. Andreopoulos Y, Mastronade I, van der Schaar M. Cross-layer optimized video streaming over wireless multihop mesh networks. *IEEE JSAC* 2006; **24** (11): 2104–2115.
5. Shiwen M, Shunan L, Yao W, Panwar SS, Yihan L. Multipath video transport over ad hoc networks. *IEEE Wireless Communications* 2005; **12**(4): 42–49.
6. Kompella S, Mao S, Hou YT, Sherali HD. Cross-layer optimized multipath routing for video communications in wireless networks. *IEEE Journal on Selected Areas in Communications* 2007; **25**(4): 831–840.
7. Hsien-Po S, van der Schaar M. Multi-user video streaming over multihop wireless networks: a distributed, cross-layer approach based on priority queuing. *IEEE Journal on Selected Areas in Communications* 2007; **25**(4): 770–785.
8. Tong X, Andreopoulos Y, van der Schaar M. Distortion-driven video streaming over multihop wireless networks with path diversity. *IEEE Transactions on Mobile Computing* 2007; **6**(12): 1343–1356.
9. Luo H, Ci S, Wu D. A cross-layer optimized distributed scheduling algorithm for peer-to-peer video streaming over multihop wireless mesh networks, *Proc. IEEE SECON*, Rome, Italy, June 2009; 1–9.
10. Wang H, Tsaftaris S, Katsaggelos A. Joint source-channel coding for wireless object-based video communications utilizing data hiding. *IEEE Transactions on Image Processing* 2006; **15**: 2158–2169.
11. Wu D, Ci S, Luo H, Wang H, Katsaggelos A. Application-centric routing for video streaming over multihop wireless networks. *IEEE Transactions on Circuits and Systems for Video Technology* 2010; **20** (12): 1721–1734.
12. Lou W, Kwon Y. H-spread: a hybrid multipath scheme for secure and reliable data collection in wireless sensor networks. *IEEE Transactions on Vehicular Technology* 2006; **55**(4): 1320–1330.
13. Papadimitratos P, Haas ZJ. Secure data communication in mobile ad hoc networks. *IEEE Journal on Selected Areas in Communications* 2006; **24** (2): 343–356.
14. Wang H, Peng D, Wang W, Sharif H, Chen HH. Image transmission with security enhancement based on region and path diversity in wireless sensor

networks. *IEEE Transactions on Wireless Communications* 2009; **8**(2): 757–765.
15. Wang W, Peng D, Wang H, Sharif H, Chen HH. Energy-constrained distortion reduction optimization for image transmission in Wireless Sensor Networks. *IEEE Transactions on Multimedia* 2008; **10**(6).
16. Kundur D, Okorafor UN, Luh W. HoLiSTiC: heterogeneous lightweight sensornets for trusted visual computing, *Proceedings of International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, Pasadena, CA, USA, December 2006; 267–270.
17. Chen W-K. *4G Wireless Video Communications*. John-Wiley&Sons: West Sussex, UK, 2009.
18. Setton E, Yoo T, Zhu X, Goldsmith A, Girod B. Cross-layer design of ad-hoc networks for real-time video streaming. *IEEE Wireless Communications Magazine* 2005; **12**(4): 59–65.
19. Ci S, Wang H, Wu D. A theoretical framework for quality-aware cross-layer optimized wireless multimedia communications. *Advances in Multimedia* 2008; **2008**: 1–10.
20. Wu D, Luo H, Ci S. Quality-driven optimization for content-aware real-time video streaming in wireless mesh networks, *GOLBECOM 2008*, New Orleans, 2008; 1–5.
21. Rashwan A, Wang H. Partial multimedia secret sharing. *Computer Science and Engineering Journa* 2015.
22. Zhang R, Regunathan SL, Rose K. Video coding with optimal inter/intra mode switching for packet loss resilience. *IEEE Journal on Selected Areas in Communications* 2000; **18**(6): 966–976.
23. Zhai H, Chen X, Fang Y. How well can the IEEE 802.11 wireless LAN support quality of service. *3* 2005; **4**: 3084–3094.
24. Calafate C, Malumbres M, Oliver J, Cano J, Manzoni P. QoS support in MANETS: a modular architecture based on the IEEE 802.11e technology. *IEEE Transactions on Circuits and Systems for Video Technology* 2009; **19**: 678–692.
25. Adlakha S, Zhu X, Girod B, Goldsmith A. Joint capacity, flow and rate allocation for multiuser video streaming over wireless ad-hoc networks, *Proc. IEEE International Conference on Communications (ICC'07)*, Glasgow, Scotland, June 2007; 1747–1753.
26. http://www.mcn.ece.ufl.edu/public/zhifeng/project/VDAT/index.htm [Accessed on 20 July 2014].