Available online at www.sciencedirect.com
ScienceDirect
ELSEVIER
Computer Communications xxx (2007) xxx–xxx
computer communications
www.elsevier.com/locate/comcom

Directional geographical routing for real-time video communications in wireless sensor networks

Min Chen [a], Victor C.M. Leung [a,*], Shiwen Mao [b], Yong Yuan [c]

[a] Department of Electrical and Computer Engineering, University of British Columbia, Vancouver, BC Canada V6T 1Z4
[b] Department of Electrical and Computer Engineering, Auburn University, AL 36849-5201, USA
[c] Department of Electronics and Information Engineering, Huazhong University of Science and Technology, Wuhan 430074, China

Abstract

In this paper, we address the problem of real-time video streaming over a bandwidth and energy constrained wireless sensor network (WSN) from a small number of dispersed video-sensor nodes (VNs) to a sink by combining forward error correction (FEC) coding with a novel multipath rouing scheme called directional geographical routing (DGR). DGR constructs an application-specific number of multiple disjointed paths for a VN to transmit parallel FEC-protected H.26L real-time video streams over a bandwidth-limited, unreliable networking environment. The multiple paths in DGR facilitate load balancing, bandwidth aggregation, and fast packet delivery. Extensive simulation experiments over randomly generated WSNs show that DGR has the following advantages: (i) lower delay, (ii) substantially longer network lifetime, and (iii) a better received video quality. In particular, DGR improves the average video peak signal-to-noise ratio (PSNR) by up to 3dB, compared to a traditional geographic routing scheme.

Keywords: Forward error correction; H.26L video; Load balancing; Multipath routing; Reliability; Wireless video communication; Wireless sensor network

1. Introduction

With the recent advances in wireless sensor networks (WSNs), it is foreseeable that video sensors will be supported in such networks, for applications such as battlefield intelligence, security monitoring, emergency response, and environmental tracking [1,2]. This paper investigates H.26L real-time video communications in a video sensor networks (VSNs), where video streams are transmitted under a number of resource and performance constraints, such as bandwidth, energy, and delay. Though a high compression ratio makes H.26L [3,4] real-time video applications suitable for low bit-rate channels, the received video quality is susceptible to transmission errors. It remains a challenging problem to deliver H.26L video data with a high quality of service (QoS) in WSNs with bandwidth-limited error-prone wireless channels. Due to the bandwidth limitation of a VSN, we consider only a small number of video-sensor nodes (VNs), which have video capture capability, taking turns to transmit video to a single sink; i.e., only one VN transmits video to the sink at any time.

Since the compressed video bit stream is extremely sensitive to transmission errors due to the frame dependency, error control techniques such as forward error correction (FEC) and automatic repeat request (ARQ) are necessary to obtain the high reliability required by video services [5]. Of these two error control mechanisms, FEC has been commonly suggested for real-time applications due to the strict delay requirements and semi-reliable nature of media streams [6]. However, links in a WSN may not have adequate bandwidth to satisfy the higher bandwidth requirement of FEC coding. Thus, conventional single-path routing schemes typically based on shortest paths [8,29]

[*] Corresponding author. Tel.: +1 604 822 6932; fax: +1 604 822 5949.
E-mail addresses: minchen@ece.ubc.ca (M. Chen), vleung@ece.ubc.ca (V.C.M. Leung), smao@ieee.org (S. Mao), yy_hust@hotmail.com (Y. Yuan).

0140-3664/$ - see front matter © 2007 Elsevier B.V. All rights reserved.
doi:10.1016/j.comcom.2007.01.016

are not very effective to support video transmissions in unreliable and bandwidth-limited WSNs, as they will cause either significant degradation in the perceived quality of the video at the sink nodes if FEC coding is not used, or large queuing delays due to insufficient bandwidth if FEC coding is used. Furthermore, transmitting a video stream using the shortest path will drain the energy of the nodes along this path and shorten the network lifetime. Thus, considering the constraints in bandwidth and energy in WSNs and delay in video delivery, we propose to divide a single video stream into multiple sub-streams, and exploit multiple disjointed paths to transmit these sub-streams in parallel. For efficient multipath routing of these parallel sub-streams from the source to the sink, we propose a novel directional geographic routing (DGR) scheme.

In WSNs, multipath routing is used to establish multiple paths between each source–sink pair. Most applications of multipath routing in WSNs aim to increase the reliability for a single flow [24–26,28]. In contrast, multipath routing is used in the proposed DGR scheme to support the delivery of multiple flows in a VSN, while the responsibility of reliable data delivery in the routing layer is relieved by the use of FEC coding.

Similar to many previous multipath routing schemes, the proposed DGR scheme also encounters the route coupling problem [7], caused by interference between packets transmitted over different paths between the same source–destination pair. If the number of paths is small (e.g., 2 or 3), non-interfering paths may be established. However, if a large number of paths are required by a specific application, non-interfering paths cannot be guaranteed due to the limited spatial size in proximity to the source–sink. In such cases, the best approach is to spatially distributing these paths as evenly as possible.

Given the scenario presented in Section 6, Figs. 1 and 2 show the OPNET simulation results of DGR's path construction and illustrate DGR's adaptability to an application-specific path number (*PathNum*). As an example, with a minimum *PathNum* of 2 in Fig. 1, DGR tries to pick two paths that do not interfere with each other. If we do not assume that the sensor nodes are aware of their geographic positions (i.e., geographic coordinates), which implies that *DistanceToCenterLine* in Fig. 1 cannot be obtained, the length of the paths may be longer than the optimized paths which traverse closely along the center line. Let N_s be the minimum number of neighbors among the source and sink nodes. In Fig. 2, $N_s = 11$; therefore the maximum possible value of *PathNum* is 11 if it is required that no two paths traverse the same set of node(s). This maximum *PathNum* value is only achievable in an ideal WSN where the node density is sufficiently high. Assuming that this is the case, DGR constructs all 11 paths as illustrated by the simulation result in Fig. 2. In practice, for a large *PathNum*, DGR spreads the paths in all directions in the proximity of the source and sink nodes, which implies that packets along some paths are likely to be forwarded to a neighbor farther to the sink than the node itself. Thus, DGR differs from traditional geographic routing scheme [29,30], in which each node forwards packets to a neighbor that is closer to the sink than the node itself until the packets reach the sink.

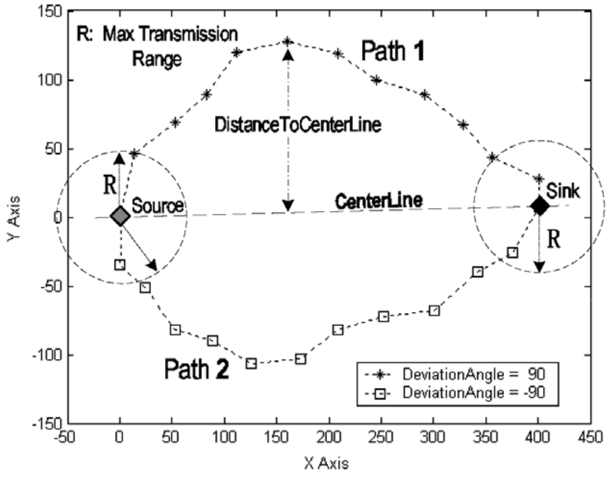


Fig. 1. Minimum number of paths constructed in DGR.

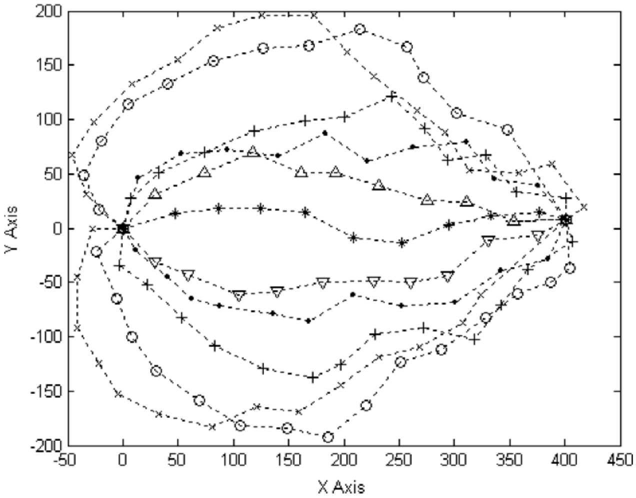


Fig. 2. Maximum number of paths constructed in DGR.

Using concurrent multiple paths in DGR also has an important limitation; i.e., DGR does not work well when a number of VNs send video to the sink simultaneously, as multiple intersecting paths interfere with each other severely. However, due to the limited bandwidth of a VSN, it is reasonable to assume that at any time instance only one VN sends video to the sink. In fact, due to the complexity and higher power consumption of VNs, we expect that among the large number of sensor nodes in a VSN, only a small number of them are VNs, while the rest are less capable low-cost sensor nodes that function as relays, which also have lower power consumption than the VNs. A few (one or more) more advanced sensors are equipped with video camera and coding capability, which serve as video sources monitoring a few sensitive locations in the area, while most of the other sensors with low-end design and low cost are scattered in the field to relay captured video to the sink.

The rest of this paper is organized as follows. Section 2 presents related work. Section 3 describes the architecture of a typical VSN, and issues concerning the design of DGR. The real-time video transmission scheme based on DGR is described in Section 4. In Section 5, we present an analysis to derive the key performance metrics for DGR and the shortest path based scheme. Simulation model and experiment results are presented in Sections 6 and 7, respectively. Section 8 concludes the paper.

2. Related work

Our work is closely related to video transmissions over wireless networks, and aspects of WSNs including reliable data transfer, multipath routing, geographic routing, and QoS provisioning for time-constrained traffic. We will give a brief review of the existing work in these areas.

A survey is presented in [9] on video streaming over wireless local area networks (WLANs). However, to the best of our knowledge, real-time video transmissions over WSNs has not been fully investigated before. The most closely related work is the following. He and Wu [14,15] studied the resource utilization behavior of a wireless video sensor and analyze its performance under resource constraints. They focused on analytical studies. Bucciol et al. [10] proposed a cross-layer ARQ algorithm for H.264 video streaming in 802.11 WLANs, which gives priority to perceptually more important packets at (re)transmissions. In [11], a transmission strategy is examined that provides adaptive QoS to layered video for streaming over 802.11 WLANs. In [12,13] hybrid transmission techniques that combine ARQ and FEC are proposed for improved real-time video transport over WLANs. However, only single-hop network scenarios are investigated in [13]. By comparison, this paper considers real-time video transmission in multi-hop WSN environments. Mao et al. [16] combined multi-stream coding with multipath transport, and showed that, in addition to traditional error control techniques, path diversity provides an effective means to combat transmission errors in ad hoc networks. In [16], the dynamic source routing (DSR) protocol is extended to support multipath routing. With their extension, multiple maximally disjoint routes are selected from all the routes returned by a route query. However, only two paths are constructed in their simulation model and only two sub-streams are considered. By comparison, our scheme is adaptive to an application-specific *PathNum*.

There are increasing research efforts on reliable data transfer in WSNs [19–26]. In these work, hop-by-hop [19,20] and end-to-end [21,22] error recovery, and multipath forwarding [24–26] are the major approaches to achieve the desired reliability. The Pump-Slowly, Fetch-Quickly (PSFQ) approach [19] works by distributing data from source nodes in a relatively slow pace and allowing nodes that have experienced data losses to recover missing segments from immediate neighbors aggressively. PSFQ employs hop-by-hop recovery instead of end-to-end recovery. In [20], the authors proposed RMST, a transport protocol that provides guaranteed delivery. RMST is a selective NACK-based protocol that can be configured for in-network caching and repair. In the event-to-sink reliable transport (ESRT) protocol [21], the sink adaptively achieves the expected event reliability by controlling the reporting frequencies of the source nodes. Several acknowledgement based end-to-end reliable event transfer schemes are proposed to achieve various levels of reliability in [22]. Yuan et al. [23] proposed a virtual MIMO based cross layer design in which the nodes can adaptively form the set of cooperative nodes to transmit data among clusters. A hop-by-hop recovery scheme and multi-hop routing scheme are integrated into the virtual MIMO scheme to jointly provide energy efficiency, reliability and end-to-end QoS guarantee. Compared with the above reliable sensor data delivery schemes, the DGR method proposed in this paper is closer to those that employ multipath routing to increase reliability. In [24], multiple disjoint paths are set up first, then multiple data copies are delivered using these paths. In [25], a protocol called ReInForM is proposed to deliver packets at the desired reliability by sending multiple copies of each packet along multiple paths from the source to the sink. The number of data copies (or, the number of paths used) is dynamically determined depending on the probability of channel error. Instead of using disjoint paths, GRAB [26] uses a path interleaving technique to achieve high reliability. These multipath routing schemes for WSNs aim at increasing the reliability for a single flow [24–26]. In contrast, multipath routing in this paper is mainly used to support the delivery of multiple flows in a WSN, while the required level of reliability is achieved using FEC. Thus, in applying multipath routing, our goal is to maximize the load balancing effect by spreading traffic evenly in the network, and using all possible paths to maximize the end-to-end capacity.

Geographic routing is a routing scheme which forwards packets based on the locations of the network nodes. In most position-based routing approaches, the minimum information a node must have to make useful routing decisions is its position (provided by GPS, Galileo, etc.), the position of its neighbors (through beaconing), and the final destination's location. The most popular forwarding method in this category is greedy forwarding, where forwarding decisions are made locally based on information about each nodes one-hop neighborhood [29,30]. Greedy Perimeter State Routing (GPSR) uses geographical locations of the nodes to make greedy routing decisions. To route around areas where greedy forwarding cannot be used, the protocol tries to find the perimeter of the area. Packets are then routed around the problem area by following the perimeter. Traditional geographic routing schemes [29,30] typically forward each packet to a neighbor closer to the sink than the forwarding node itself until the packet reaches the sink. Instead, DGR spreads the paths in all directions in the proximity of the source and sink nodes, which means that packets are not necessarily forwarded to neighbors that are closer to the sink than the forwarding nodes.

QoS provisioning for time-constrained traffic in WSNs has many applications, such as real-time target tracking in battlefield environments, emergent event triggering in monitoring applications, etc. There are increasing research efforts in this area. SPEED [35] is an adaptive real-time routing protocol that aims to reduce the end-to-end deadline miss ratio in WSNs. Akkaya et al. proposed an energy-aware QoS routing protocol to support both best effort and real-time traffic at the same time [36]. The purpose is to meet the end-to-end delay constraint of the real-time traffic while maximizing the throughput of the best effort traffic at the same time. Akkaya et al. also used a Weighted Fair Queuing (WFQ) based packet scheduling to achieve the end-to-end delay bound in [37]. Yuan et al. [38] proposed an integrated energy and QoS aware transmission scheme for WSNs, in which the QoS requirements in the application layer, and the modulation and transmission schemes in the data link and physical layers are jointly optimized. In [39], EDDD provides service differentiation between best effort (BE) and time-sensitive traffic by deploying the BE filter and the RT (real-time) filter. The BE filter intends to balance the global energy and to prolong network lifetime, while the end-to-end delay is not a primary concern. The RT filter intends to provide better end-to-end delay performance for time-sensitive traffic. In this paper, we use multiple paths to increase the end-to-end capacity and achieve the QoS requirements in terms of end-to-end latency.

3. Directional geographic routing scheme

In this section, we discuss the key design issues of the novel DGR scheme proposed in this paper.

3.1. Architecture of video sensor network

In a VSN, VNs equipped with video capturing and processing capabilities are tasked to capture digital visual information about target events or situations, and deliver the video streams to a sink node [14]. Generally, a VN should be equipped with a battery of higher energy capacity than an ordinary sensor node, since it is already equipped with a relatively expensive camera that would become useless if the VN ran out of energy.

However, it is economically infeasible and sometime unnecessary to equip all the sensor nodes with video capturing and processing capabilities, especially for large scale and/or dense WSNs. This paper considers a VSN architecture, as illustrated in Fig. 3, where a small number of VNs are sparsely deployed among a much larger number of densely deployed low-power sensor nodes. The set of VNs only cover the target regions remotely monitored by the sink. The inexpensive ordinary sensor nodes perform the simple task of forwarding packets that carry sensed video data to the sink. Due to bandwidth limitation of a typical WSN link, we consider that the VNs take turns to send video streams to the sink; i.e., at any instance only one of the VNs is actively sending video data to the sink.

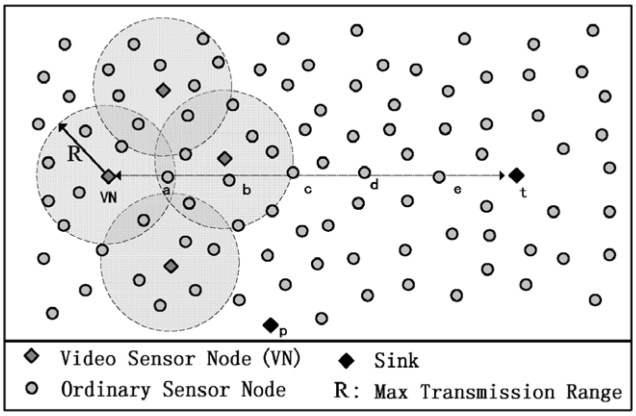


Fig. 3. System architecture of video sensor network.

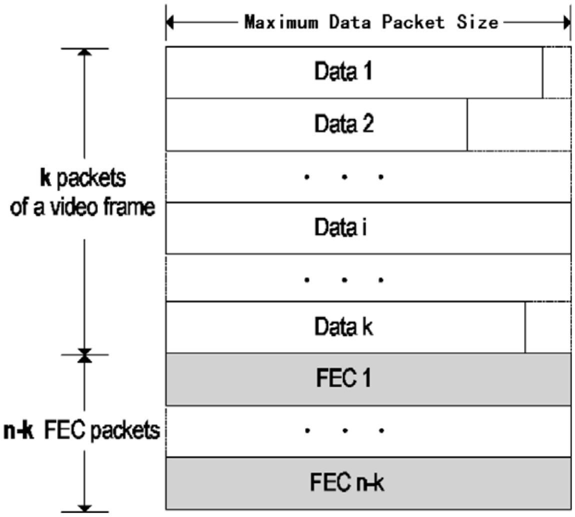


Fig. 4. FEC coding scheme.

To combat unreliable transmissions over the wireless environment and satisfy the strict end-to-end delay requirements, we assume that a FEC coding scheme is employed whereby each VN generates redundant packets to increase error resilience for real-time video transmissions.

We implement the FEC coding scheme proposed in [40], where $n - k$ redundant packets are generated to protect k data packets of a video frame, as shown in Fig. 4. The size of each FEC packet is equal to the maximum size of the data packets. If any k of the n packets in the coding block are received by sink, the corresponding video frame can be successfully restored.

3.2. Obtaining mapping coordinates

In the global coordinate system of Fig. 5, o is the origin; h is the node initiating next hop selection. The absolute

Please cite this article in press as: M. Chen et al., Directional geographical routing for real-time video communications ..., Comput. Commun. (2007), doi:10.1016/j.comcom.2007.01.016

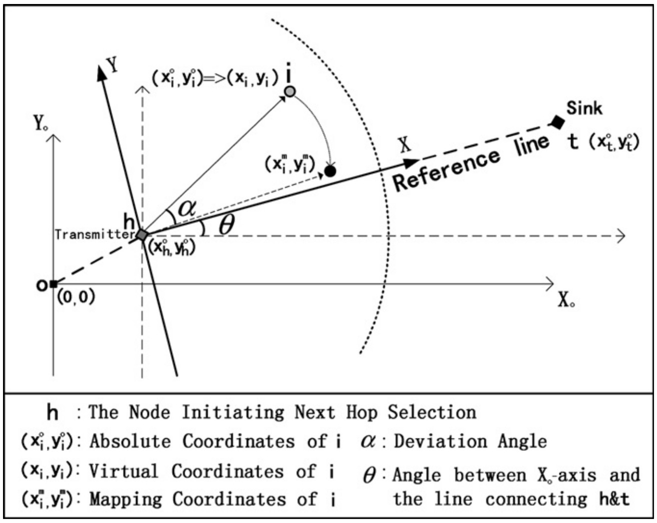

Y
$(x_i^o, y_i^o) \Rightarrow (x_i, y_i)$ i
Sink
Y_o
X
t (x_t^o, y_t^o)
(x_i^m, y_i^m)
Reference line
α
h
θ
Transmitter
(x_h^o, y_h^o)
O
(0,0)
X_o
h : The Node Initiating Next Hop Selection
(x_i^o, y_i^o): Absolute Coordinates of i
(x_i, y_i): Virtual Coordinates of i
(x_i^m, y_i^m): Mapping Coordinates of i
α : Deviation Angle
θ : Angle between X_o-axis and the line connecting h&t

Fig. 5. Obtaining mapping coordinates.

coordinates of node h (x_h^o, y_h^o) is piggybacked in the message broadcast by h. Thus, a neighbor node i knows the position of its upstream node h (x_h^o, y_h^o), its own position (x_i^o, y_i^o), and the sink's location (x_t^o, y_t^o).

In this paper, we employ the *virtual coordinates* proposed in [17]. The *virtual coordinates* of a node (e.g., i in Fig. 5) are defined as the coordinates in the virtual two-dimensional coordinate system where the node's upstream node (e.g., h in Fig. 5) is the origin, and the X-axis is the line between the upstream node (e.g., h in Fig. 5) and the sink. In the example shown in Fig. 5, the *virtual coordinates* of i is denoted by (x_i, y_i), which can be calculated by (1).

$$\begin{cases} x_i = \cos(\theta)\cdot(x_i^o - x_h^o) + \sin(\theta)\cdot(y_i^o - y_h^o), \\ y_i = \cos(\theta)\cdot(y_i^o - y_h^o) - \sin(\theta)\cdot(x_i^o - x_h^o), \\ \theta = \arctan(\frac{y_t^o - y_h^o}{x_t^o - x_h^o}). \end{cases} \tag{1}$$

In Fig. 5, *ReferenceLine* is defined as the straight line between the origin of the virtual coordinate system (e.g., h) to the sink; *DeviationAngle* (α) is defined as the angle that specifies how much a path is expected to deviate from the *ReferenceLine* at the origin point. If we rotate *virtual coordinates* round the origin by α, the rotated coordinates are defined as *mapping coordinates*. If $\alpha > 0$, the rotation is clockwise, and if $\alpha < 0$, the rotation is anticlockwise. As an extreme example, $\alpha = 0$ means that a path will be set up along the direction from h to the sink; i.e., the shortest path. The *mapping coordinates* are used to evaluate a node's eligibility of becoming the next hop node during the operation of constructing a path with deviation angle α. In the example shown in Fig. 5, the *mapping coordinates* of i is denoted by (x_i^m, y_i^m), which can be calculated by (2).

$$\begin{cases} x_i^m = \cos(\alpha)\cdot x_i + \sin(\alpha)\cdot y_i, \\ y_i^m = \cos(\alpha)\cdot y_i - \sin(\alpha)\cdot x_i. \end{cases} \tag{2}$$

3.3. Next hop node selection strategy

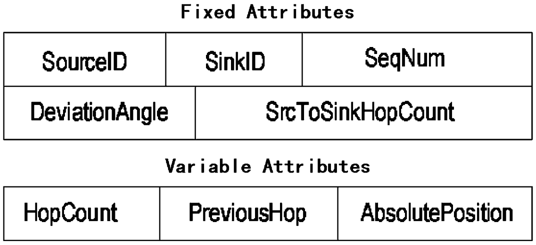
To establish a direction-aware path, a probe (PROB) message is broadcast initially by the source for route discovery. A selected next hop will continue to broadcast PROB message to find its next hop, and so forth.

The information contained in a PROB is shown in Fig. 6. The PROB message is identified by the *SourceID*, *SinkID* and *SeqNum*. *DeviationAngle* (denoted by α) specifies the unique direction where a path will traverse during path establishment. If α is a negative value, a path will be established below the *ReferenceLine*; otherwise, above the *ReferenceLine*. *SrcToSinkHopCount* (denoted by H_s) is the minimum hop count from the source to the sink. Let R be the maximum transmission range of a sensor node. Let D_{src}^t be the calculated distance between the source and the sink. H_s is equal to $\lceil \frac{D_{\text{src}}^t}{R} \rceil$. The fixed attributes in a PROB are set by the source and not changed while the PROB is propagated across the network. On the other hand, when an intermediate node broadcasts a PROB, it will change the variable attributes in the message. *HopCount* is the hop count from the source to the current node. *PreviousHop* is the identifier of the current node. *AbsolutePosition* denotes the absolute coordinates of the current node.

A node receiving a PROB will calculate its *virtual coordinates* based on its upstream neighbor's position indicated in *AbsolutePosition* field of the PROB. Then, *mapping coordinates* is calculated based on the *virtual coordinates* and *DeviationAngle* according to (2).

In Fig. 7, the point $(R, 0)$ is called the *StrategicMappingLocation*, which is located on the *ReferenceLine* at a distance R from the upstream node h. In practice, a next hop neighbor of h may not be found whose *mapping coordinates* is located at the *StrategicMappingLocation*. Thus, DGR will select as the next hop node the neighbor whose *mapping coordinates* is closest to the *StrategicMappingLocation*, instead of the neighbor closest to the sink as in traditional geographical routing protocols.

The shadow area in Fig. 7 is considered to be the Mapping-Coordinates-Selection-Area. The neighboring nodes whose *mapping coordinates* are located in the Mapping-Coordinates-Selection-Area are deemed to be next hop candidates (*NHC*s). To be selected as the next hop node, an *NHC* will start a backoff timer when it receives a *PROB* message.



Fixed Attributes

SourceID	SinkID	SeqNum
DeviationAngle	SrcToSinkHopCount	

Variable Attributes

HopCount	PreviousHop	AbsolutePosition

Fig. 6. Packet structure of PROB message.

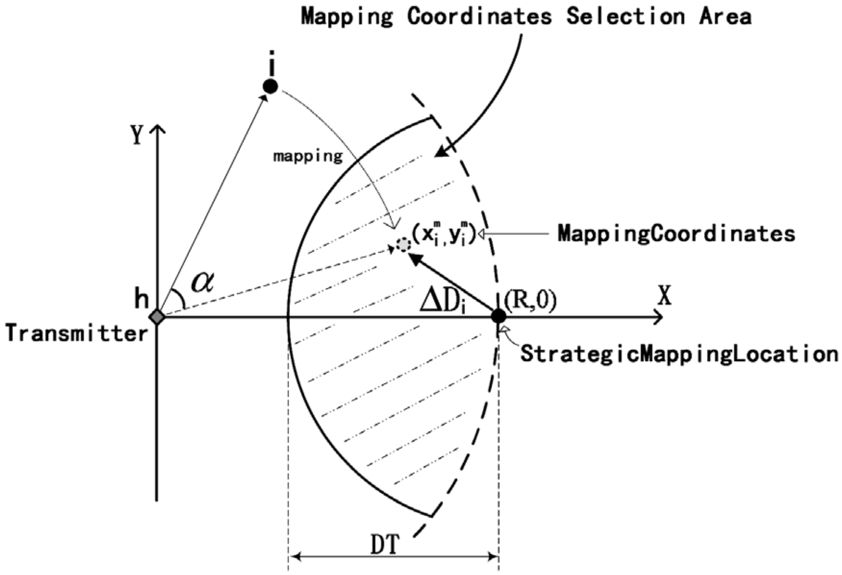


Fig. 7. Mapping coordinates selection area.

Let ΔD_i be the distance between the *mapping coordinates* of node *i* and *StrategicMappingLocation*. ΔD_i is given by

$$\Delta D_i = \sqrt{(x_i^m - R)^2 + (y_i^m)^2}. \tag{3}$$

A threshold *DT* is set to limit the selection area. Node *i* becomes an *NHC* if $\Delta D_i < DT$, and it calculates a backoff time (t_b) based on ΔD_i as follows.

$$t_b = \tau \times \Delta D_i + \mathrm{rand}(0, \mu), \tag{4}$$

where τ is a fixed time interval. $\mathrm{rand}(0, \mu)$ returns a random value uniformly distributed in $(0, \mu)$, and μ is a small constant.

Among the timers of the *NHC*s, the one with the smallest t_b value will expire first. Let such an *NHC* be node *i*. Node *i* will then unicast a "reply" message (REP) to its upstream node *h* immediately after timer expiration. If *h* receives the REP, it will broadcast a "selection" message (SEL) with the identifier of node *i*. Node *h* will only accept the first REP while ignoring the later ones. If node *i* receives the SEL, it is selected as the next hop node of *h*. When other *NHC*s receive the SEL or REP, they will cancel their backoff timers.

To avoid collision among the REPs, we set τ to a sufficiently large value. Since the next hop selection is a relatively infrequent task as compared to the periods of data transmissions, the use of a larger τ will result in a slightly longer path establishing delay, but will not increase the data latency.

When node *i* is selected as a next hop node, it will broadcast PROB to its neighbors. The above selection mechanism is repeated until the sink receives the PROB message. When the sink receives the PROB, it will broadcast a notification packet immediately to terminate the path establishment. If *i* is selected as a next-hop node of a path, it will no longer participate in next hop selections for other paths, which guarantees that all the paths are strictly disjointed.

3.4. Path repair mechanism

If an intermediate node (say, node *i*) in a path fails, the MAC layer of its previous-hop node (node *h*) will not be able to deliver any packet to it. After several retransmission attempts, the MAC layer of node *h* will notify the routing layer of the failed transmission. Then, the DGR protocol will broadcast a PROB message immediately at node *h* and selects a different next hop using the same procedure as in the previous section, and hands the packet again down to the MAC layer for forwarding. Since the path repair operation increases the transmission latency, the transmission deadline of the video packet may be exceeded, in which case the packet will be dropped at node *h* while the path repair operation continues until it is successful.

3.5. The mechanism for deviation angle adjustment

Though we can control the direction of path establishment as illustrated in Section 3.3, the PROB message may go farther and farther from the sink if the deviation angle is fixed, and the PROB may finally arrive at the network border, as shown in Fig. 8(a). Thus, after a path starting from the source has been extended for a number of hops in the specified direction, it should point back to the sink quickly. To achieve this goal, we shrink the operational value of α hop-by-hop; i.e., adjust deviation angle.

On reception of a PROB from its upstream node, neighbor *i* can determine the hop count from the source (denoted by *H*) and the minimum hop count between the source and the sink (denoted by H_s) indicated in the respective fields, *HopCount* and *SrcToSinkHopCount*, of the PROB. Assume *i* is selected as the next hop node, the adjusted

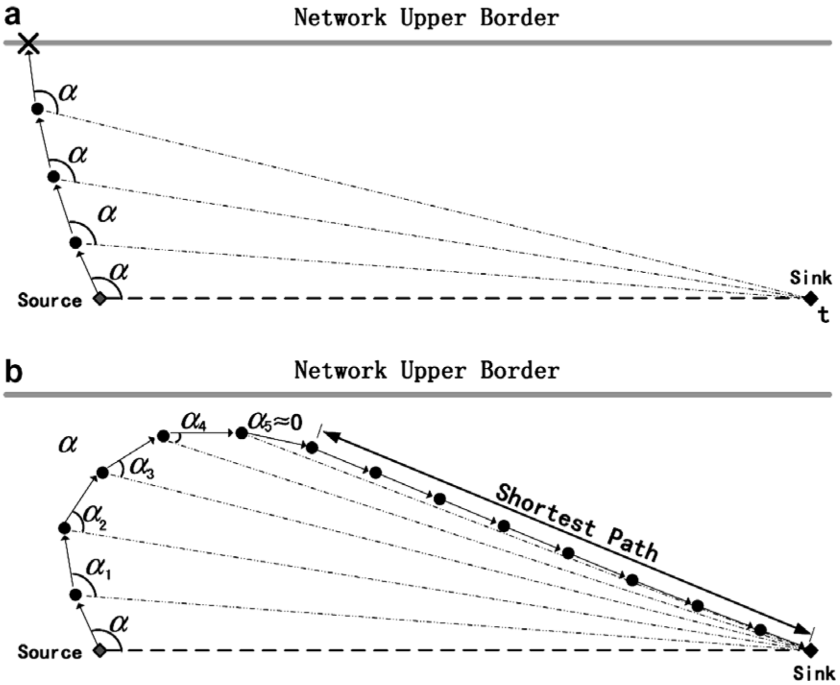


Fig. 8. Determination of position relative to the centerline.

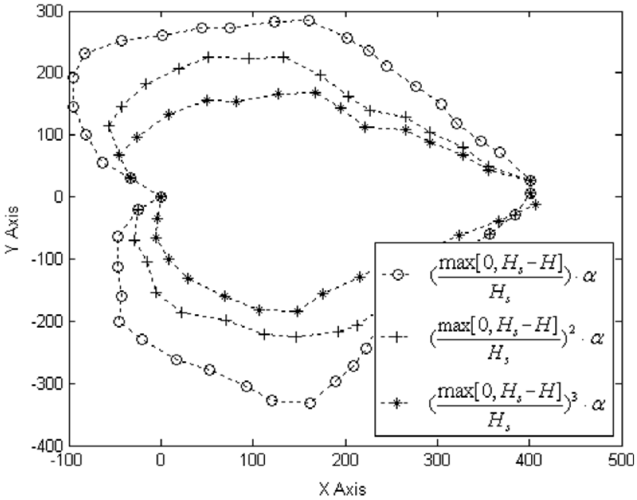


Fig. 9. Effects of different deviation angle adjusting functions with $|\alpha| = 150$.

deviation angle (α_H) is calculated using a *deviation angle adjusting function f* as $\alpha_H = f(H) \cdot \alpha$. The larger is H, the smaller is α_H, and the larger is the progress toward the sink. The adjusted mapping coordinates of node i (x_i^m, y_i^m) can be calculated as

$$\begin{cases} x_i^m = \cos(\alpha_H) \cdot x_i + \sin(\alpha_H) \cdot y_i, \\ y_i^m = \cos(\alpha_H) \cdot y_i - \sin(\alpha_H) \cdot x_i. \end{cases} \tag{5}$$

The form of the deviation angle adjusting function decides how fast a path points back to the sink. In Fig. 9, we evaluate three different functions: $\left(\frac{\max[0, H_s - H]}{H_s}\right) \cdot \alpha$, $\left(\frac{\max[0, H_s - H]}{H_s}\right)^2 \cdot \alpha$, and $\left(\frac{\max[0, H_s - H]}{H_s}\right)^3 \cdot \alpha$. α_H decreases most slowly with $\left(\frac{\max[0, H_s - H]}{H_s}\right) \cdot \alpha$, and the corresponding path length is the longest. By comparison, the path expansion is the smallest with $\left(\frac{\max[0, H_s - H]}{H_s}\right)^3 \cdot \alpha$. Generally, higher order functions will yield faster converging paths than lower order ones. Thus, the higher the order of the angle adjusting function, the more closely will the adjacent paths be packed spatially.

We exploit nine paths for concurrent transmissions in the simulations, where a third order function is selected empirically with $\alpha_H = \left(\frac{\max[0, H_s - H]}{H_s}\right)^3 \cdot \alpha$. We do not preclude the possibility that a "higher order" function may give a better performance than the selected function. A suitable choice of the order of the angle adjusting function will depend on the number of paths, the distance between the source–sink pair, and the node density. A thorough evaluation of these dependencies is left for future research. A good starting point of this future research is the work by Toumpis and Tassiulas [18], which studied the spatial distribution of wireless nodes and can give the basis for a systematic selection of the angle adjusting function.

Using the deviation angle adjustment method, a path can be established successfully using any specific initial deviation angle. In order to set up an application-specific number of paths with different initial deviation angles, the source can transmit a series of PROBs each specifying a different deviation angle. As an example, in Fig. 10 the source changes the absolute value of the deviation angle from 0° to 90°, in steps of 22.5° and sends a different PROB message with each deviation angle. Thus, in total 9 paths are established with α equal to −90°, −77.5°, −45°, −22.5°, 0°, 22.5°, 45°, 77.5°, and 90°, respectively.

Please cite this article in press as: M. Chen et al., Directional geographical routing for real-time video communications ..., Comput. Commun. (2007), doi:10.1016/j.comcom.2007.01.016

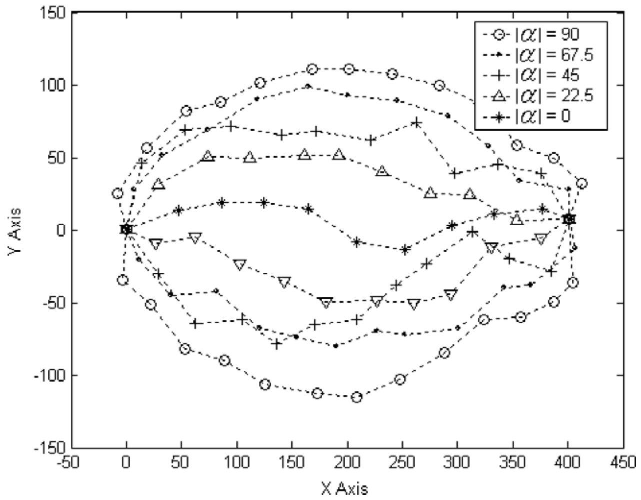


Fig. 10. Example of 9 disjointed paths using DGR.

3.6. Handling the dead end problem

The so-called dead end problem [32,33] arises when a packet is forwarded to a local optimum, i.e., a node with no neighbor that has a closer hop distance to the destination as, illustrated in Fig. 11. In DGR, if there are no *NHC*s found within the Mapping-Coordinates-Selection-Area of an intermediate node, it will enter the greedy mode to select the node among all its neighbors that is geographically closest to the sink as the next-hop node. If a node does not have any neighbor closer to the sink in the greedy mode, DGR meets the dead end problem and the packet is forwarded in recovery mode, i.e., the packet is routed according to the right-hand rule to recover from the local minimum [29]. The right-hand rule is a well-known concept for traversing mazes. To avoid loops, the packet is routed in recovery mode on the faces of a locally extracted planar subgraph, namely the Gabriel graph. The packet returns to greedy mode when it reaches a node closer to the sink than the node where the packet entered the recovery mode. Furthermore, if the node has *NHC*(s) again, the packet switches to DGR routing rather than greedy routing.

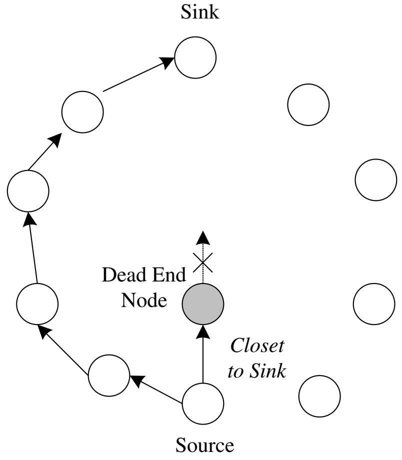


Fig. 11. Illustration of dead end problem.

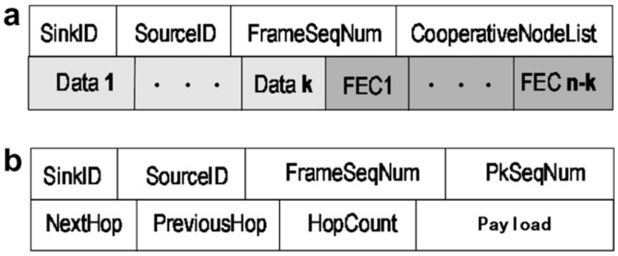
4. Video transmission strategy

In this section, a hybrid video stream broadcasting and sub-streams unicasting scheme is proposed based on multiple disjoint paths that has been pre-established between the source VN and the sink using DGR as presented in above section.

An active VN first broadcasts to its one-hop neighbors a packet concatenating all the data and FEC packets of a video frame, following the structure shown in Fig. 12(a). Those neighboring nodes that are the first intermediate nodes of individual paths to the sink are referred as *CooperativeNode*s. Upon receiving the concatenated packet broadcast by the VN, each *CooperativeNode* selects its own payload according to the *CooperativeNodeList* in the concatenated packet. *CooperativeNodeList* contains the identifiers of the *CooperativeNode*s and the sequence numbers of the corresponding packets (denoted by *PkSeqNum* in Fig. 12) assigned to these nodes. Then these *CooperativeNode*s unicast the assigned packets to the sink via the respective individual paths using the packet structure shown in Fig. 12(b).

A simple example of the proposed transmission architecture is illustrated in Fig. 13, where the multipath routing layer sets up 3 paths between the source and the sink. Each path goes through a different *CooperativeNode* of the VN. To simplify the analysis, we consider that the VN encodes one video frame into two data packets and one FEC packet, and divides the video stream into three packet sub-streams: two data flows and one FEC flow. The structure of sub-stream entry is shown in Fig. 14. The *CooperativeNodeList* is highlighted in Fig. 14 and contains the list of *NodeID*s and *PacketToSend*s.

In general, the VN can intelligently specify the number of sub-streams and assign these sub-streams according to the number of available paths, the path length, and the number of data/FEC packets to send for each video frame. If the number of data/FEC packets of a video frame is larger than the number of available paths, some paths will deliver multiple packet flows. Otherwise, the VN can select a set of shorter paths to achieve faster delivery. The length


a

SinkID	SourceID	FrameSeqNum	CooperativeNodeList		
Data 1	. . .	Data k	FEC1	. . .	FEC n-k

b

SinkID	SourceID	FrameSeqNum	PkSeqNum
NextHop	PreviousHop	HopCount	Payload

Fig. 12. Data packet formats: (a) concatenated data broadcast by VN; (b) data unicast by ordinary sensor node.

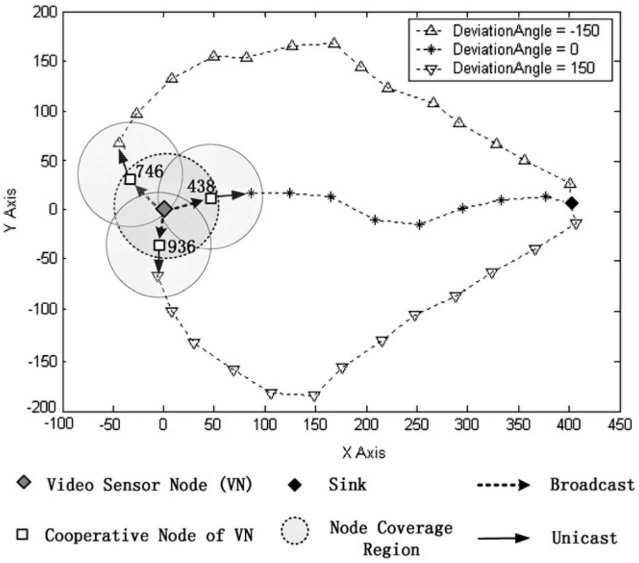
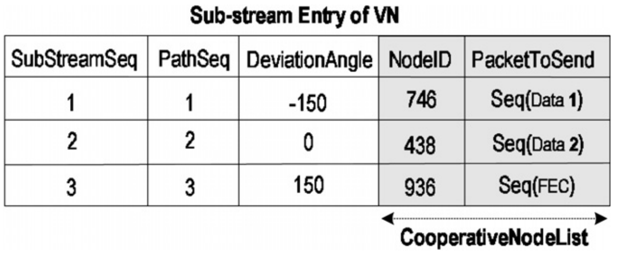


Fig. 13. Illustration of DGR-based multipath video transmission.



Sub-stream Entry of VN

SubStreamSeq	PathSeq	DeviationAngle	NodeID	PacketToSend
1	1	-150	746	Seq(Data 1)
2	2	0	438	Seq(Data 2)
3	3	150	936	Seq(FEC)

CooperativeNodeList

Fig. 14. Example of sub-stream entry of the VN (NodeID as in Fig. 13).

of a path can be estimated by the value of the deviation angle. If the number of residual paths is high, the VN can adopt a Round-Robin path scheduling algorithm among the available paths to achieve load balance. To adapt to the fluctuations in channel quality, the VN also can adjust the number of FEC packets for each video frame and the number of paths used according to the feedback information from the sink.

Assume that the FEC scheme generates $n - k$ redundant packets to protect k data packets of a video frame. If the sink has correctly received the k data packets, it may decode the frame immediately to reduce latency, while the redundant packets are subsequently ignored and dropped. However, if there are errors in the data packets, then the redundant packets are applied in an attempt to correct the errors using the FEC scheme.

In Fig. 15, the DGR-based transmission scheme is compared with the traditional scheme where the whole video stream is transmitted over the shortest path.

5. Performance analysis

In this section, we present an analysis that derives the key performance metrics of the transmission scheme based on the shortest path routing and DGR, including the number of successful frame deliveries before lifetime of WSN L, and the cumulative delay for a video frame T. To simplify the analysis, we define "lifetime" as the time until the first node dies.

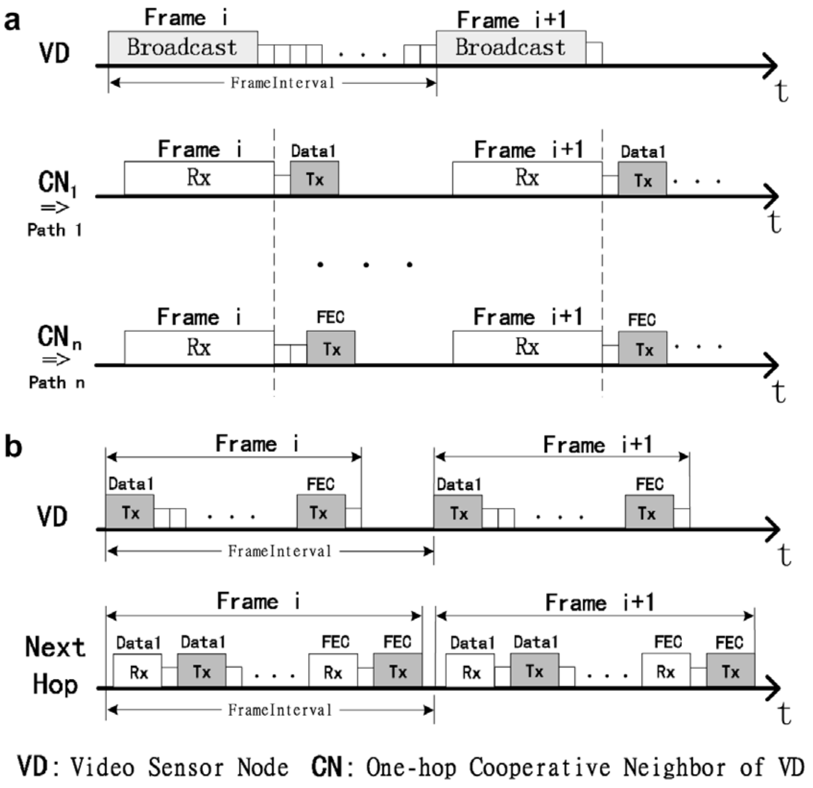


Fig. 15. Comparison of video transmissions: (a) DGR; (b) traditional scheme.

Assume link is always in good status. Assume VN packetizes a video frame into k data packets. Let S_{d} be the size of each data packets in bits. Let S_{h} be the size of packet header. Let v be the data transmission rate. Let t_{frame} be the frame interval. Let H be the number of hops along the shortest path between the VN and the sink. Let $H+h$ be the average number of hops in DGR scheme. Let t_{ctrl} be the total delay for control messages during a successful data transmission.

Let T_{sp} denote the average end-to-end frame latency in the shortest path based transmission scheme. Let t_{data} denote the time to transmit a data packet. $t_{\mathrm{data}} = \frac{S_{\mathrm{d}}+S_{\mathrm{h}}}{v}$. Then, T_{sp} is equal to:

$$T_{\mathrm{sp}} = \begin{cases} k \cdot (t_{\mathrm{data}} + t_{\mathrm{ctrl}}) \cdot H, \\ \qquad\qquad \text{if } k \cdot (t_{\mathrm{data}} + t_{\mathrm{ctrl}}) \leqslant t_{\mathrm{frame}}, \\ k \cdot (t_{\mathrm{data}} + t_{\mathrm{ctrl}} + t_{\mathrm{queue}}) \cdot H, \\ \qquad\qquad \text{if } k \cdot (t_{\mathrm{data}} + t_{\mathrm{ctrl}}) > t_{\mathrm{frame}}. \end{cases} \tag{6}$$

Note that if $k \cdot (t_{\mathrm{data}} + t_{\mathrm{ctrl}}) > t_{\mathrm{frame}}$, new frame arrives before the previous one has been delivered timely, thus, packets begin to be backlogged in the queue of the intermediate nodes along the shortest path. In Eq. 7, t_{queue} denotes the average queueing delay.

Let T_{dgr} denote the average end-to-end frame latency in DGR-based transmission scheme. Assume k paths are used to transmit each of k packet concurrently. If k is a large value and a pair adjacent paths may interfere with each other, let t_{c} be the average backoff time to access the channel. Note that $H+h-1$ numbers of hops will unicast the data and only the VN will broadcast the concatenated data. Let t_1 denote the time for the VN to broadcast the concatenated packet. $t_1 = \frac{k \cdot S_{\mathrm{d}}+S_{\mathrm{h}}}{v}$. Then, T_{dgr} is equal to:

$$T_{\mathrm{dgr}} = \begin{cases} t_1 + (t_{\mathrm{data}} + t_{\mathrm{ctrl}} + t_{\mathrm{c}}) \cdot (H+h-1), \\ \qquad\qquad \text{if } t_1 \leqslant t_{\mathrm{frame}}, \\ t_1 + t^1_{\mathrm{queue}} + (t_{\mathrm{data}} + t_{\mathrm{ctrl}} + t_{\mathrm{c}}) \cdot (H+h-1), \\ \qquad\qquad \text{if } t_1 > t_{\mathrm{frame}}. \end{cases} \tag{7}$$

In Eq. 7, t^1_{queue} denotes the queueing delay at the first hop in DGR if the VN's bandwidth is less than the required capacity for the transmission of the concatenated packet. However, there should be no the case that the capacity of a selected path for unicasting a video *sub-stream* is less than the required capacity, which means k is set to too small a value inefficiently.

Let L_{gpsr} denote the number of successful frame deliveries before lifetime in GPSR based transmission scheme. Let E be the initial energy of sensor node. Let m_{t} and m_{r} be the energy consumption for transmitting and receiving a bit, respectively. Let e be the energy consumption of control messages exchanging for a successful data transmission. Let e_{data} denote the energy to transmit and receive a data packet. $e_{\mathrm{data}} = (S_{\mathrm{d}} + S_{\mathrm{h}}) \cdot (m_{\mathrm{t}} + m_{\mathrm{r}})$. Then, L_{gpsr} can be estimated as:

$$\begin{aligned} L_{\mathrm{gpsr}} &= \left\lceil \frac{E}{k \cdot (e_{\mathrm{data}} + e)} \right\rceil, \\ &= \left\lceil \frac{E}{k \cdot (S_{\mathrm{d}} + S_{\mathrm{h}}) \cdot m_r + k \cdot (S_{\mathrm{d}} + S_{\mathrm{h}}) \cdot m_t + k \cdot e} \right\rceil. \end{aligned} \tag{8}$$

Let L_{dgr} denote the number of successful frame deliveries before lifetime in DGR-based transmission scheme. Since the cooperative nodes of the VN will receive the concatenated packet once for each video frame, the cooperative nodes will become the bottleneck of lifetime. Then, L_{dgr} can be estimated as:

$$L_{\mathrm{dgr}} = \left\lceil \frac{E}{(k \cdot S_{\mathrm{d}} + S_{\mathrm{h}}) \cdot m_{\mathrm{r}} + (S_{\mathrm{d}} + S_{\mathrm{h}}) \cdot m_{\mathrm{t}} + e)} \right\rceil. \tag{9}$$

6. Simulation methodology

6.1. Simulation model

In order to demonstrate the performance of DGR, we compare it with GPSR [29] via extensive simulation studies. In this section, we present the simulation settings and performance metrics. The simulation results will be presented in the following section.

We use OPNET [41,42] for discrete event simulation. The sensor nodes are battery-operated and have a limited energy supply, whereas the sink is assumed to have an infinite energy supply. We assume that all the nodes (VN, sensor nodes, the sink) are stationary. In each simulation, a network topology is generated with the sink located at one side of the area, the VN located at the other side, and sensor nodes randomly located over the entire area. To decrease the influence of a specific topology on the results, each simulation was performed 10 times with different random topologies for the sensor nodes. For each topology, we repeat the transmission experiment 20 times with different random seeds. Each data point presented in the next section for evaluation is give by the mean value of the 10×20 runs. Fig. 16 illustrates the topology of a set of randomly generated sensor nodes, as well as the VN and the sink node in the network.

The test video sequence is Forman that was coded in QCIF format (176 × 144 pixels/frame) at a temporal resolution of 20 frames/s by the H.26L video coding standard [3,4]. The average bit rate of video data is about 178 kbps, and the average bit rate after packet encapsulation is about 200 kbps. The first frame is intra-coded and the remaining frames are inter-coded. Each frame is packetized into 6 data packets. Three FEC packets are transmitted per video frame to protect the video data packets.

The energy consumption parameters are shown in Table 1. Every node starts with the same initial energy budget (120 W s) [43]. We can use the following equation to calculate the energy consumption of a sensor node in the three possible states (transmitting, receiving, or overhearing) [43]:

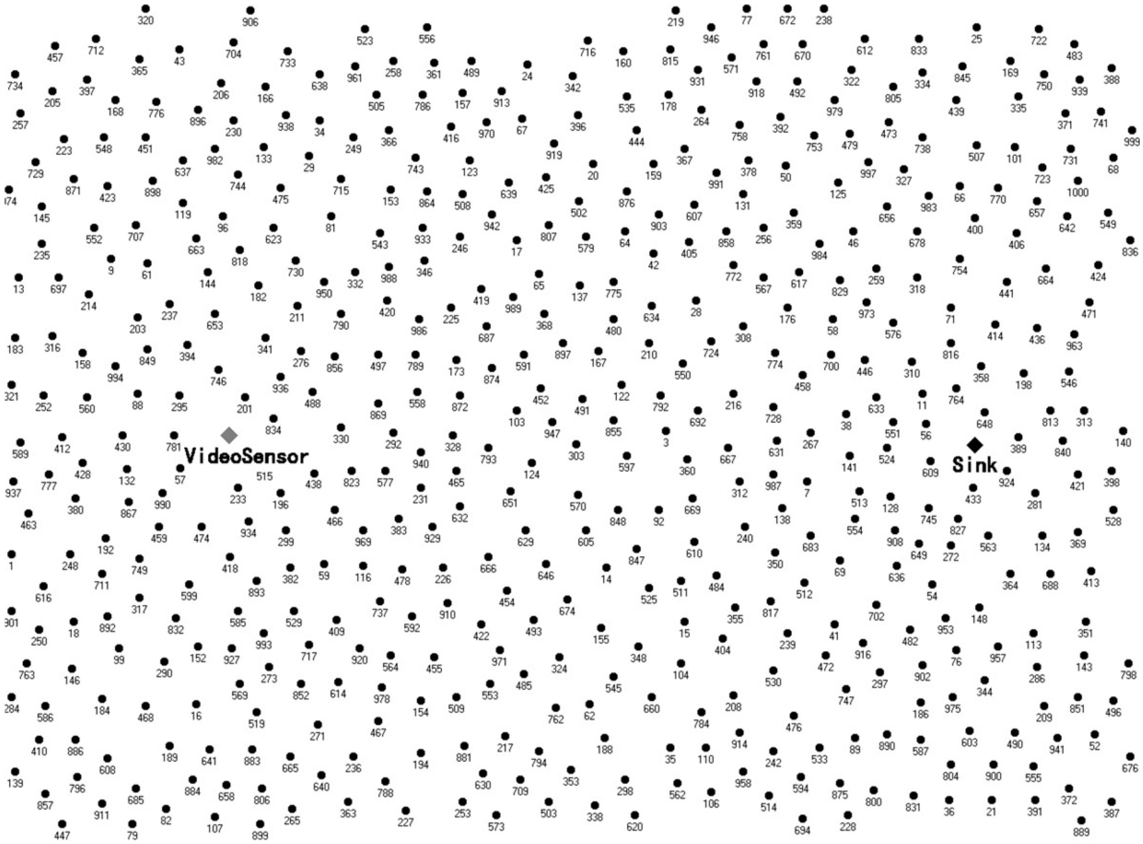


Fig. 16. Example of network topology in simulation.

Table 1
Energy consumption parameters of the Lucent IEEE802.11 2 Mbps WaveLAN card [16]

Normalized initial energy of sensor node (W s)		4500
Incremental cost (μW s/bytes)	m_{tx}	1.9
	m_{recv}	0.5
	$m_{overhearing}$	0.39
Fixed cost (μW s)	b_{tx}	454
	b_{recv}	356
	$b_{overhearing}$	140
P_{idle} (mW)		843

$$m \times PacketSize_{\mathrm{MAC}} + b + P_{\mathrm{idle}} \times t \times 1000(\mu\mathrm{W\ s}) \tag{10}$$

In (10), m represents the incremental cost compared to the power consumption in the idle state, b represents the fixed cost, t represents the duration of the state, and $PacketSize_{\mathrm{MAC}}$ represents the size of the MAC packet.

To model link failures, we simply block the channel between two nodes with a link failure rate p. Thus, a packet will be lost with a probability p. The parameter values used in the simulations are presented in Table 2. The basic settings are common for DGR and GPSR.

Table 2
Simulation settings

Basic specification	
Network size	500 m × 500 m
Topology configuration mode	Randomized
Total sensor node number	500
Data rate at MAC layer	2 Mbps
Transmission range of sensor node	50 m
Packet loss rate	Default: 0.15%
DGR specification	
Number of paths	Default: 9
τ in Eq. (4)	Default: 2.5 ms
μ in Eq. (4)	Default: 5 ms

6.2. Performance metrics

In this section, five performance metrics are evaluated:

- *Lifetime.* There is no universally agreed definition of network lifetime as it depends on the specific application. The lifetime can be measured by the time when the first node exhausts its energy, or when a certain fraction of nodes are dead, or even when all nodes are dead. Alternately, it may be reasonable to measure the network lifetime by application-specific parameters, such as the time when the network can no longer relay the video. Similar with [39], we consider two kinds of lifetime, namely *LifeTime I* and *LifeTime II*. *LifeTime I* is the time when the first node dies due to energy depletion. *LifeTime II* is the time when the VN has no available paths to the sink.

- *Number of successful frames received by sink before lifetime.* It is denoted by n_{frame}. It is the number of video frames delivered to the sink before network lifetime is met. Note that due to FEC coding, some packets of a video frame may be lost and the frame can still be correctly received. n_{frame} is an alternate measure of the network lifetime in this paper.
- *Average end-to-end packet delay.* Let T_{dgr}, T_{gpsr}, and $T_{\text{gpsr}}^{\text{fec}}$ be the average end-to-end packet delay of DGR, GPSR, and GPSR with FEC coding, respectively. They include all possible delays during data dissemination, caused by queuing, retransmission due to collision at the MAC, and transmission time.
- *Energy consumption per successful data delivery.* It is denoted by *e*. It is the ratio of network energy consumption to the number of data packets delivered to the sink before lifetime. The network energy consumption includes all the energy consumed by transmitting and receiving during a simulation. As in [44], we do not account for energy consumption in the idle state, since this part is approximately the same for all the schemes simulated.
- *PSNR.* The peak signal-to-noise ratio is a measure of the received video quality.

Among the performance metrics defined above, we believe that n_{frame} is the most important metric for WSNs, while PSNR and T_{ete} are important for real-time video transmissions.

7. Performance evaluations

In this section, the simulation results for three video transmission techniques are evaluated; i.e., DGR, GPSR (GPSR without FEC coding), and GPSR with FEC coding. In each group of experiments, we change the link failure rate from 0 to 0.3 in step size of 0.05.

In Fig. 17, T_{dgr} is always lower than T_{gpsr}, though the average path length in DGR is higher than the length of the shortest path. This is due to the fact that the average bandwidth provided by the shortest path is very close to the bandwidth required by a video stream, so that link congestion and video frame corruption due to burst packet losses are inevitable when single-path routing is employed. $T_{\text{gpsr}}^{\text{fec}}$ is much higher than T_{dgr} and T_{gpsr}, especially at low packet loss rates, which shows that the limited link bandwidth cannot accommodate the additional transmission overhead of FEC packets. As the packet loss rate increases, more packets are lost before they reach the sink. Since the packet loss helps to alleviate congestion, both T_{gpsr} and $T_{\text{gpsr}}^{\text{fec}}$ show a reduction as the packet loss rate is increased. However, congestion is not a problem for DGR due to the load balancing effect of multipath routing; therefore T_{dgr} stays relatively constant as the packet loss rate changes.

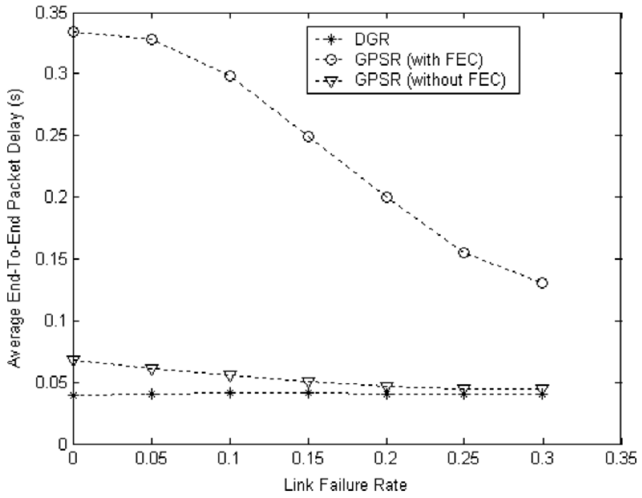


Fig. 17. Comparisons of T_{ete}.

Fig. 18 compares n_{frame} values for DGR, GPSR, and GPSR with FEC coding as the packet loss rate is varied. When the packet loss rate increases, n_{frame} of all the schemes increases since some sensor nodes save the energy of packet transmissions if they fail to receive the packets. DGR has higher n_{frame} values compared with that of GPSR and GPSR with FEC coding, because DGR distributes the traffic load of each video frame evenly over multiple paths. Thus, energy consumption of each path in DGR is much smaller than that of GPSR. In DGR, though 9 paths are exploited, only 6 paths are used to transmit data sub-streams while the remaining 3 paths are used to transmit FEC sub-streams. Ideally, DGR should achieve about 5 times more n_{frame} than that of GPSR (without FEC). However, the cooperative neighbors of the VN are the bottlenecks with respect to energy consumption, since they receive the long concatenated packets from the VN, while other intermediate sensor nodes receive much shorter pack-

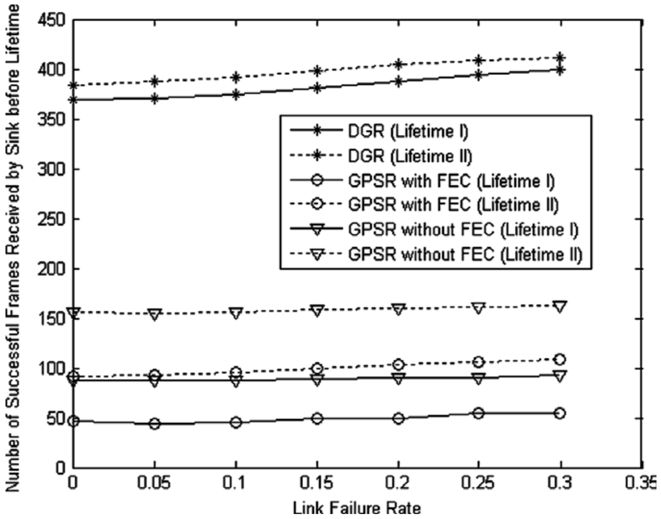


Fig. 18. Comparisons of n_{frame}.

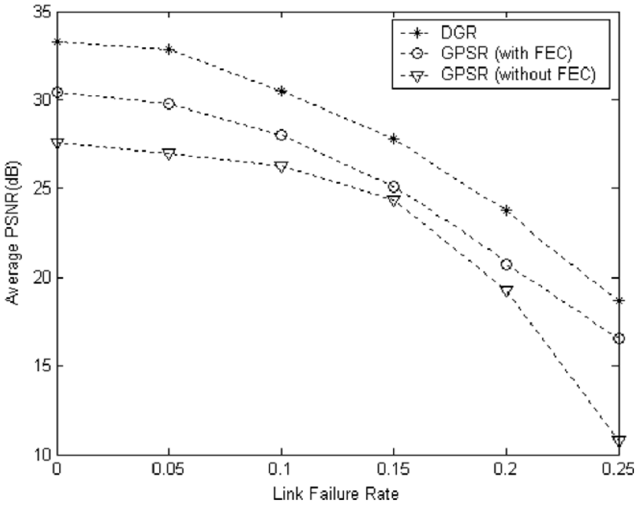


Fig. 19. Comparisons of *PSNR*.

ets with only a single data/FEC payload. The end result is that n_{frame} of DGR is about 3 times more than that of GPSR in terms of *LifeTime I*. In GPSR, if FEC coding is adopted, more energy is consumed to transmit the FEC packets. Thus, n_{frame} of GPSR with FEC is about 40 lower than that of GPSR in terms of *LifeTime I*. Note that *LifeTime II* of DGR is close to its *LifeTime I*. By comparison, *LifeTime II* of GPSR with FEC has about 50 s more than its *LifeTime I*, and *LifeTime II* of GPSR without FEC has about 70 s more than its *LifeTime I*.

Fig. 19 shows the comparison of PSNR for these three schemes. It can be seen that DGR achieves the highest PSNR, which on average is about 3 dB higher than that of GPSR with FEC and 5 dB higher than that of GPSR. Though the PSNR of GPSR with FEC is higher than GPSR, both its delay and lifetime performances are worst, as we have already described above.

Fig. 20 compares the PSNR of each frame resulting from the test sequence Foreman with packet loss rate = 0.05. Since GPSR does not take any measure to prevent error propagations, the PSNR of the reconstructed image decreases rapidly as more frames are received, and the subjective quality of the received video is poor. At a higher packet loss rate (0.2 in Fig. 21), the received video quality degrades rapidly for all the schemes due to packet losses. Nevertheless, DGR still achieves the highest perceived quality for the video frames received at the sink.

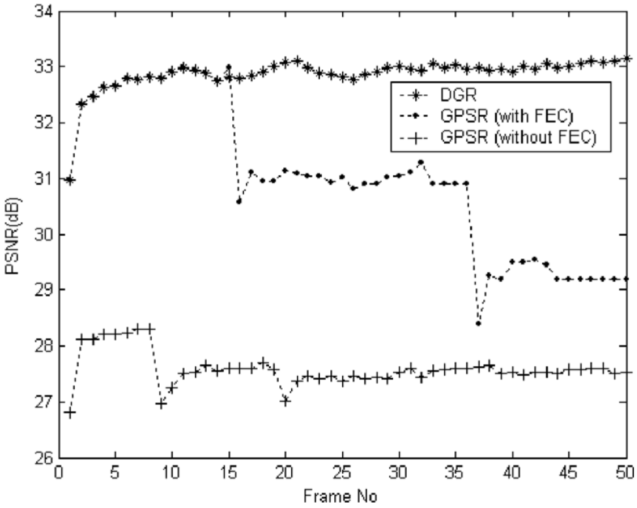


Fig. 20. Comparisons of PSNR with link failure rate = 0.05.

Fig. 22 shows that e_{dgr} is slightly higher than that of GPSR and GPSR with FEC under varying packet loss rates. This is because the average path length of DGR, which is equal to total path length (accounting for all the paths used to delivery the video stream) divided by *PathNum*, is larger than that of GPSR. GPSR with FEC has a higher *e* than GPSR due to the additional energy consumed to transmit the FEC packets. This slightly increased energy consumption is the price paid for significantly improved real-time video quality. In addition, the overall perfor-

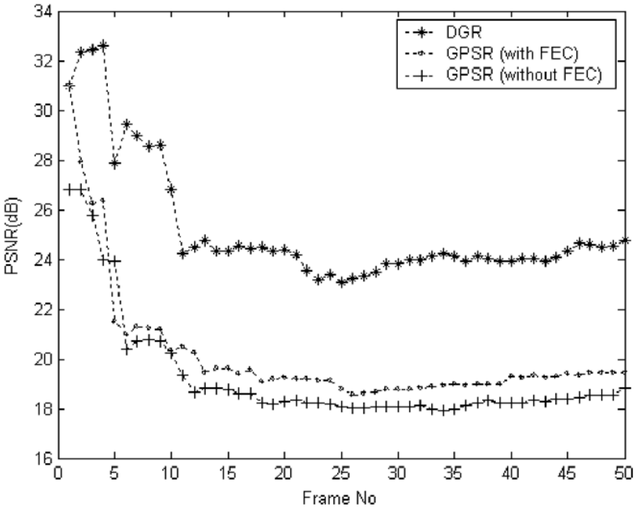


Fig. 21. Comparisons of PSNR with link failure rate = 0.2.

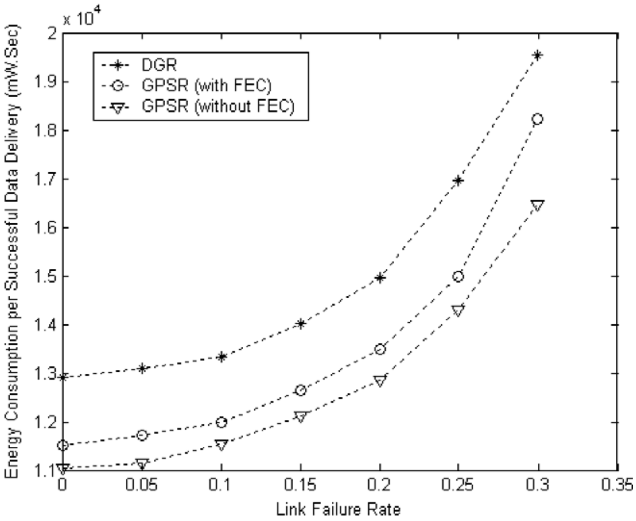


Fig. 22. Comparisons of *e*.

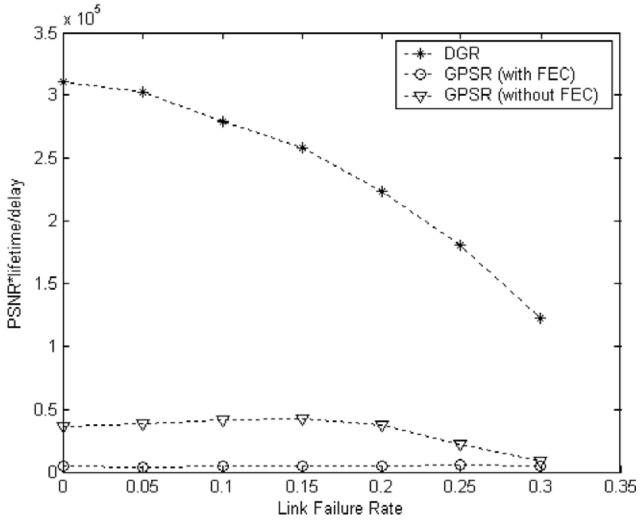


Fig. 23. Comparison of the compound metric η.

mance is greatly improved when energy, lifetime and video quality are jointly considered, as shown in Fig. 23.

By comparison, it is important to consider the lifetime, PSNR and average delay for real-time video applications. Thus, we adopt the following metric to evaluate the integrated performance of Lifetime I, PSNR and delay:

$$\eta = \frac{n_{\text{frame}} \cdot \text{PSNR}}{\text{delay}}. \tag{11}$$

The higher is η, the better is the composite QoS provided by the WSN to support real-time video services. From Fig. 23, we find that no matter whether FEC coding is adopted or not, DGR achieves a much higher η than GPSR. Since GPSR with FEC obtains small improvements of PSNR by sacrificing the energy-efficiency and delay performance, the η of GPSR with FEC is lower than that of GPSR.

Note that the above simulation does not reflect the impact of multiple active video sources. When multiple video sources are active, the complexity of our scheme is higher than that of a single-path routing scheme such as GPSR. We believe that this is a price well paid for improved video quality. Due to the bandwidth limitation of a typical WSN, it is reasonable as we have assumed in this paper that video sources do not transmit data simultaneously to the sink. Instead, they may make requests to the sink when there are video streams to send, and take turns to send video packets when instructed to do so by the sink. The protocol for making/granting these initial requests is a subject for further study.

8. Conclusion

This paper has proposed a novel architecture for video sensor networks. We investigate the problem of real-time video transmissions over WSNs in general, and the performance of H.26L video transmissions in WSNs in particular. Compressed video is susceptible to transmission errors. However, the limited bandwidth in a WSN may not allow video sensor nodes to transmit additional FEC packets to protect the video data without subjecting all packets to excessive queuing delay. It is challenging to simultaneously achieve delay guarantees and obtain a high perceived video quality at the sink. To solve this problem, we have proposed a novel video transmission scheme which efficiently combines multipath routing with FEC coding to tackle the natural unreliability of WSNs as well as their bandwidth constraints. The proposed scheme includes: (1) the directional geographical routing algorithm for the construction of an application-specific number of multiple disjointed paths; (2) the hybrid video stream broadcasting and sub-streams unicasting scheme. This study has also provided insights into novel usage of multipath transmissions in WSNs. Instead of the typical application of multipath routing in traditional WSN designs to provide path redundancy for failure recovery, DGR employs multipath routing to increase aggregate source-to-sink bandwidth and achieve better load balancing. Performance evaluations have shown that in combination with packet-level FEC coding, DGR simultaneously achieves reliability, energy-efficiency and timely packet delivery to support real-time video service over WSNs.

Acknowledgement

This work was supported in part by the Canadian Natural Sciences and Engineering Research Council under Grant STPGP 322208-05.

References

[1] J.N. Al-Karaki, A.E. Kamal, Routing techniques in wireless sensor networks: a survey, IEEE Personal Communications 11 (6) (2004) 6–28.

[2] K. Akkaya, M. Younis, A survey of routing protocols in wireless sensor networks, Ad Hoc Network Journal 3 (3) (2005) 325–349.

[3] ITU-T SG16. Draft, H.26L Video Coding[s], February 1998.

[4] G. Bjontegaard, H.26L Test Model Long Term Number 9(TML-9) Draft0[s], July 2002.

[5] Y. Wang, S. Wenger, J. Wen, A.K. Katsaggelos, Error resilient video coding techniques, IEEE Signal Processing Magazine 86 (2000) 61–82.

[6] H. Ma, M. El Zarki, Broadcast/multicast MPEG-2 video over broadband fixed wireless access networks, IEEE Network Magazine 13 (6) (1998) 80–93.

[7] M.R. Pearlman, Z.J. Haas, P. Sholander, S.S. und Tabrizi, On the impact of alternate path routing for load balancing in mobile ad hoc networks, in: ACM international symposium on mobile ad hoc networking and computing (MobiHOC), 2003.

[8] C. Intanagonwiwat, R. Govindan, D. Estrin, Directed diffusion: a scalable and robust communication paradigm for sensor networks, in: The Proceedings of the 6th Annual ACM/IEEE MobiCom, Boston, MA, August 2000.

[9] M. Etoh, T. Yoshimura, Advances in wireless video delivery, Proceedings of IEEE 93 (2005) 111–122.

[10] P. Bucciol, G. Davini, E. Masala, E. Filippi, J.D. Martin, Cross-layer perceptual ARQ for H.264 video streaming over 802.11 wireless networks, in: Proceedings of IEEE Global Telecommunications

Please cite this article in press as: M. Chen et al., Directional geographical routing for real-time video communications ..., Comput. Commun. (2007), doi:10.1016/j.comcom.2007.01.016

Conference (Globecom), vol. 5, Dallas, TX, November/December 2004, pp. 3027–3031.

[11] Q. Li, M. van der Schaar, Providing adaptive QoS to layered video over wireless local area networks through real-time retry limit adaptation, IEEE Transactions on Multimedia 6 (2) (2004) 278–290.

[12] A. Majumdar, D.G. Sachs, I.V. Kozintsev, K. Ramchandran, M.M. Yeung, Multicast and unicast real-time video streaming over wireless LANs, IEEE Transactions on Circuits and Systems for Video Technology 12 (6) (2002) 524–534.

[13] M. Chen, G. Wei, Multi-stages hybrid ARQ with conditional frame skipping and reference frame selecting scheme for real-time video transport over wireless LAN, IEEE Transactions on Consumer Electronics 50 (1) (2004) 158–167.

[14] Z. He, D. Wu, Resource allocation and performance analysis of wireless video sensors, IEEE Transactions on Circuits and Systems for Video Technology 16 (5) (2006) 590–599.

[15] Z. He, D. Wu, Performance analysis of wireless video sensors in video surveillance, Proceedings of Globecom 2005 1 (2005) 178–182.

[16] S. Mao, S. Lin, S.S. Panwar, Y. Wang, E. Celebi, Video transport over ad hoc networks: multistream coding with multipath transport, IEEE Journal on Selected Areas in Communications 21 (10) (2003) 1721–1737, Special issue on Recent Advance in Wireless Multimedia.

[17] M. Chen, X. Wang, V. Leung, Y. Yuan, Virtual coordinates based routing in wireless sensor networks, Sensor Letters 4 (2006) 325–330.

[18] S. Toumpis, L. Tassiulas, Packetostatics: deployment of massively dense sensor networks as an electrostatics problem, IEEE INFOCOM 2005 4 (2005) 2290–2301.

[19] C.Y. Wan, A.T. Campbell, L. Krishnamurthy, Pump-slowly, fetch-quickly (PSFQ): a reliable transport protocol for sensor networks, IEEE Journal of Selected Areas in Communications 23 (2005) 862–872.

[20] F. Stann, J. Heidemann, RMST: reliable data transport in sensor networks, in: Proceedings of the IEEE International Workshop on Sensor Network Protocols and Applications, May 2003, pp.102–112.

[21] Y. Sankarasubramaniam, O.B. Akan, I.F. Akyildiz, ESRT: Event-to-sink reliable transport in wireless sensor networks, in: ACM MobiHoc, June 2003, pp.177–188.

[22] N. Tezcan, E. Cayirci, M.U. Caglayan, End-to-end reliable event transfer in wireless sensor networks, in: IEEE PIMRC'04, vol.2, September 2004, pp. 989–994.

[23] Y. Yuan, Z. He, M. Chen, Virtual MIMO based cross-layer design for wireless sensor networks, IEEE Transactions on Vehicular Technology 53 (3) (2006).

[24] D. Ganesan, R. Govindan, S. Shenker, D. Estrin, Highly resilient, energy efficient multipath routing in wireless sensor networks, in: Mobile Computing and Communications Review (MC2R), vol 1., No. 2, 2002, pp. 10–24.

[25] B. Deb, S. Bhatnagar, B. Nath, ReInForM: reliable information forwarding using multiple paths in sensor networks, IEEE LCN (2003) 406–415.

[26] F. Ye, G. Zhong, S. Lu, L. Zhang, GRAdient broadcast: a robust data delivery protocol for large scale sensor networks, Wireless Networks 11 (3) (2005) 285–298.

[28] S. De, C. Qiao, H. und Wu, Meshed multipath routing with selective forwarding: an efficient strategy in wireless sensor networks, Elsevier Computer Communications Journal 26 (4) (2003).

[29] B. Karp, H.T. Kung, GPSR: greedy perimeter stateless routing for wireless networks, in: Proceedings of ACM MobiCom 2000, Boston, MA, USA, August 2000, pp. 243–254.

[30] Y. Yu, R. Govindan, D. Estrin, Geographical and energy aware routing: a recursive data dissemination protocol for wireless sensor networks, UCLA Computer Science Department Technical Report UCLA/CSD-TR-01-0023, May 2001.

[32] L. Zou, M. Lu, Z. Xiong, A distributed algorithm for the dead end problem of location-based routing in sensor networks, IEEE Transactions on Vehicular Technology 54 (2005) 1509–1522.

[33] Q. Fang, J. Gao, L.J. und Guibas, Locating and bypassing routing holes in sensor networks, IEEE Infocom (2004).

[35] T. He, J.A. Stankovic, L. Chenyang, T. Abdelzaher, SPEED: a stateless protocol for real-time communication in sensor networks, in: Proceedings of IEEE ICDCS'03, May 2003, pp. 46–55.

[36] K. Akkaya, M. Younis, Energy-aware routing of time-constrained traffic in wireless sensor networks, Journal of Communication Systems 17 (2004) 663–687, Special Issue on Service Differentiation and QoS in Ad Hoc Networks.

[37] K. Akkaya, M. Younis, Energy and QoS aware routing in wireless sensor networks, Journal of Cluster Computing Journal 2–3 (2005) 179–188.

[38] Y. Yuan, Z.K. Yang, Z.H. He, J.H. He, An integrated energy aware wireless transmission system for QoS provisioning in wireless sensor network, Elsevier Journal of Computer Communications 29 (2) (2006) 162–172, Special issue on Dependable Wireless Sensor Networks.

[39] M. Chen, T. Kwon, Y. Choi, Energy-efficient differentiated directed diffusion (EDDD) for real-time traffic in wireless sensor networks, Elsevier Journal of Computer Communications 29 (2) (2006) 231–245, Special issue on Dependable Wireless Sensor Networks.

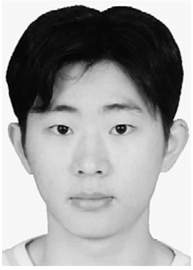
[40] L. Rizzo, Effective erasure codes for reliable computer communication protocols, ACM Computer Communication Review 27 (1997) 24–36.

[41] http://www.opnet.com.

[42] M. Chen, OPNET Network Simulation, Press of Tsinghua University, China, 2004, ISBN 7-302-08232-4, 352 pages.

[43] L.M. Feeney, M. Nilsson, Investigating the energy consumption of a wireless network interface in an ad hoc networking environment, in: Proceedings of IEEE INFOCOM'01, April 2001, pp. 1548–1557.

[44] J. Gao, L. Zhang, Load balancing shortest path routing in wireless networks, IEEE INFOCOM 2004 2 (2004) 1098–1107.



Min Chen was born on December 1980. He received BS, MS and Ph.D degree from Department of Electronic Engineering, South China University of Technology, in 1999, 2001 and 2004, respectively. He is a post-doctoral fellow in Communications Group, Department of Electrical and Computer Engineering, University of British Columbia. He was a post-doctoral researcher in Multimedia & Mobile Communications Lab., School of Computer Science and Engineering, Seoul National University in 2004 and 2005. His current research interests include wireless sensor network, wireless ad hoc network, and video transmission over wireless networks.

Victor C. M. Leung received the B.A.Sc. (Hons.) and PhD degrees, both in electrical engineering, from the University of British Columbia (UBC) in 1977 and 1981, respectively. He was the recipient of many academic awards, including the APEBC Gold Medal as the head of the 1977 graduate class in the Faculty of Applied Science, UBC, and the NSERC Postgraduate Scholarship. From 1981 to 1987, Dr. Leung was a Senior Member of Technical Staff and satellite systems specialist at MPR Teltech Ltd. In 1988, he was a Lecturer in Electronics at the Chinese University of Hong Kong. He returned to U.B.C. as a faculty member in 1989, where he is a Professor and holder of the TELUS Mobility Research Chair in Advanced Telecommunications Engineering in the Department of Electrical and Computer Engineering. His research interests are in mobile systems and wireless networks.

Dr. Leung is a Fellow of IEEE and a voting member of ACM. He is an editor of the IEEE Transactions on Wireless Communications, an associate editor of the IEEE Transactions on Vehicular Technology, and an editor of the International Journal of Sensor Networks.

Please cite this article in press as: M. Chen et al., Directional geographical routing for real-time video communications ..., Comput. Commun. (2007), doi:10.1016/j.comcom.2007.01.016

Shiwen Mao received the B.S. and the M.S. degree from Tsinghua University, Beijing, P.R. China in 1994 and 1997, respectively, both in Electrical Engineering. He received the M.S. degree in System Engineering and the Ph.D. degree in Electrical and Computer Engineering from Polytechnic University, Brooklyn, NY, in 2000 and 2004, respectively. He was a Research Member at IBM China Research Lab, Beijing from 1997 to 1998, and a research intern at Avaya Labs-Research, Holmdel, NJ in the summer of 2001. He has been a Research Scientist in the Department of Electrical and Computer Engineering, Virginia Tech, Blacksburg, VA since December 2003. Currently, he is an Assistant Professor in the Department of Electrical and Computer Engineering at Auburn University, Auburn, AL. Dr. Mao's research interests include cross-layer design and optimization in multi-hop wireless networks, as well as multimedia communications. He is a co-recipient of the 2004 IEEE Communications Society Leonard G. Abraham Prize in the Field of Communications Systems. He is the co-author of a textbook, TCP/IP Essentials: A Lab-Based Approach (Cambridge University Press, 2004).

Yong Yuan received the B.E. and M.E. degrees from the Department of Electronics and Information in Yunnan University, Kunming, P.R.China, in 1999 and 2002, respectively. Since 2002, he has been studying at the Department of Electronics and Information in Huazhong University of Science and Technology, P.R.China, as a Ph.D. candidate. His current research interests include wireless sensor network, wireless ad hoc network, wireless communication and signal processing.