The Effects of Interference on Video Quality over Wireless Sensor Networks

Houda Zeghilet\textsuperscript{a}, Yacine Bazizz, Moufida Maimour\textsuperscript{c}, Bouabdellah Kechar\textsuperscript{b}, Nadjib Badache\textsuperscript{d}

\textsuperscript{a}DTISI Laboratory, CERIST Research Center, Algiers, Algeria
\textsuperscript{b}Oran Es-Sénia University, Oran, Algeria
\textsuperscript{c}CRAN laboratory, Lorraine University, CNRS, France
\textsuperscript{d}LSI Laboratory, USTHB University, Algeria

Abstract

Multi-path routing appears to be an essential feature for supporting the characteristics of wireless multimedia sensor networks. In fact, using multiple paths between the source and the destination can provide adequate network resources required by such networks. However, the effect of wireless interferences between paths severely affects the performance of this approach. In this paper, we propose a simple interference-aware multipath routing protocol and study the effects of inter-path interferences on the received video quality in the context of wireless multimedia sensor networks. We show, through simulations, that minimizing interferences between the transmission paths can provide necessary bandwidth to support multimedia applications when small frame rates are used.

© 2013 The Authors. Published by Elsevier B.V.
Selection and peer-review under responsibility of Elhadi M. Shakshuki

Keywords: WMSNs, Multipath routing, Interference, Video

1. Introduction

In wireless multimedia sensor networks (WMSNs), multipath routing is considered as an efficient solution to provide necessary bandwidth and load balancing in order to support high volumes of data (image and video). In general, this is done through the simultaneous transmission of data flows in the network. However, when multiple adjacent paths are being used concurrently, the broadcast nature of wireless channels results in inter-path interference which significantly degrades the end-to-end throughput. This problem is known as the route coupling problem and was first introduced in [1]. In this work, the authors stated that using multipath routing in a single channel network results in negligible benefits due to severe route coupling. Moreover, using simply link or node-disjoint shortest paths is not sufficient to guarantee any throughput gains because sensor nodes may interfere beyond their communication range.
The problem of finding two non-interfering paths between a source and a destination pair is NP-complete [2]. Therefore, many heuristic algorithms have been proposed in the literature to construct paths with minimum inter-path interferences. The main ideas of these propositions is to construct paths that are physically separated to minimize the effects of wireless interferences. One solution to do so is to use the geographical information of nodes to construct physically separated paths [3]. The work in the latter reference is based on a geographic multipath routing [3] where a deviation angle adjustment method is employed to construct geographically far paths in order to minimize interferences between them. Another solution is to block the neighboring nodes of the already built paths to prevent them from belonging to the future paths. The work in [4] proposes an incremental multipath routing protocol where, for a given session, only one path is built at once. Additional paths are built when required, typically in case of congestion or bandwidth shortage. These paths are chosen to be non-interfering with the used ones by blocking the nodes neighbors of the already selected path. The work in [5] proposes an Interference-Minimized Multipath Routing (I2MR) protocol that increases throughput. In this protocol, a primary shortest path is discovered and then a secondary and a backup paths are built. This is done after marking one and two-hop neighbors of intermediate nodes of the primary path so that they do not participate in the second discovery phase. This work still uses localization information to build the three paths towards three different sinks. In [6], Wang et al. present an interference aware multipath routing without requiring localization information. The proposed protocol permits to construct two spatially disjoint and interference minimized paths using two rounds of request and reply cycles. In the first round, the protocol discovers the shortest path between and blocks the neighbor nodes along with the nodes of the shortest path. In the second round, two interference minimized paths are constructed with the nodes that are adjacent to the blocked nodes in the first round.

In these solutions, either two round of route request and route reply cycles are used and/or the nodes are explicitly blocked using control messages. Unlike the previously cited works, we address in this paper the effects of inter-path interference on video data transmission over WSNs. We propose a simple multipath routing protocol that permits to construct multiple paths between a source and a destination with minimum inter-path interferences while minimizing the overhead used to block the nodes of the network.

Our protocol is source-based and consists in one phase of route request and route reply cycle. Several transmission strategies using the built paths with different degree of interference are evaluated through simulation. The impact of interference on video data quality is then considered for these different strategies. Our simulation results show that using less-interfering paths can provide better received video quality. Transmission using less-interfering paths permits to support real-time bandwidth requirements in WMSNs when small frame rates are considered even if the shortest path is not used.

The rest of the paper is organized as follows. Section 2 gives the interference model along with the basic idea of our protocol. Section 3 describes our multipath protocol. The section 4 is dedicated to simulation results where the performances of our protocol are assessed and section 5 concludes the chapter.

2. Preliminaries

2.1. Interference Model

We employ a simple interference model, as the one proposed in [6], where the interference occurs between two edges when either the endpoint node of one edge is within the interference range of an endpoint node of the other edge. Therefore, for a given pair of edges \((i, j)\) and \((k, l)\) if \(\max(\text{dist}(i, k), \text{dist}(i, l), \text{dist}(j, k), \text{dist}(j, l)) \leq I\) then the two edges are interfering with each other, where \(\text{dist}(x, y)\) returns the distance between nodes \(x\) and \(y\) and \(I\) is the interference range. An ideal scheme would be for the two endpoints of an edge to be far away from the two endpoints of the other edge by a distance greater or equal to \(2R\) [7]. If \(I \leq R\) than the two endpoints are neighbors.

To measure the interference between two disjoint paths we simply use the number of common neighbors (one-hop) between two paths instead of using the number of links connecting them. We define the neighbors of a path as the union of the sets of the neighbors of each node belonging to this path except the source and the sink nodes. In fact, these nodes should not be blocked to permit the construction of other paths. Our goal is to simplify the implementation of the routing protocol and minimize the overhead used to block the nodes.
in the network. Therefore, we propose a source-based routing protocol consisting of one round of route request and route reply cycle. In our protocol the sink is responsible of blocking the neighboring nodes of the built paths. The constructed paths may present different degrees of interference for example:

- \( I \geq R \): The idea here is to build one path (typically the shortest one) and block the Neighbors list of this path by the sink, i.e. any route request for a path containing one of the neighbors of the already built path is ignored. A second path is chosen such that the intersection between the Neighbors lists of the two paths is an empty set.

- \( I \geq 2R \): Here the protocol permits to construct the shortest path and block the nodes of this path and the nodes of its Neighbors list. The other paths are selected using the same principle as in the previous paragraph. When other paths are found, the source node selects the paths above and below the shortest path using the locations of its neighbors.

### 2.2. Selecting a Routing Path

The routing paths are selected such as the signal strength is maximal between any two pairs of nodes. This is mainly done to: First, minimize the intra-path interference between the nodes of the same path; and second, avoid blind areas in the network. In fact, in figure 1 redrawn from [6], the area CDE is a blind area. A node located in this area is not belonging to the list of neighbors of the nodes A and B. When another path is selected using the intersection of the Neighbors list of two paths, such a node may still be exiting between the two paths. This situation may biases the route selection rule. However, a good signal strength between the nodes permits to mitigate this situation by reducing the blind areas between the nodes and by that the number of the nodes belonging to these areas.

### 3. Interference-minimized Multipath Routing

#### 3.1. Paths Discovery

Initially, each node broadcasts a HELLO message to discover its neighbors. After the HELLO step, each node maintains the list of neighbors (called Neighbors) which is used in the future steps. The Neighbors list is used by the sink to implicitly block the neighboring nodes and prevent them form belonging to the future paths. If the topology depicted in figure 2 is considered, the Neighbors lists of some nodes are: Neighbors(1) = \{2, 15, 16\}, Neighbors(2) = \{1, 3, 16, 6, 17\}, Neighbors1(3) = \{2, 4, 7\}. The source node initiates the path construction by flooding the network with an Explore Message until the sink node is reached. An Explore Message contains the request sequence number, the path ID, the list of crossed nodes, the number of hops, a Metric field, the list of neighbors of the crossed nodes and the list of the neighbors of the source node. The path ID is the ID of the first node on the path that receives the Explore Message from the source. When an intermediate node receives the Explore Message it processes it if the signal strength of the sending node is maximum. Then the intermediate node checks if it has not already processes an Explore Message for the same pathId. In this case, the intermediate node increases the number of hops, adds its ID and its Neighbors list and forward the Explore Message to its neighbors. Otherwise, the hop count is
compared and the path that presents the minimum hop count is selected. In case the hop count is greater or equal for the same path ID the Explore Message is ignored. The Neighbors list should not contain redundant nodes’ IDs. Therefore, each intermediate node delete the redundant neighbors IDs in the piggybacked list of Neighbors. When the first Explore Messages are received by the sink, it selects the shortest path (in terms of number of hops) and creates a routing entry for this path (selects path \( S, 1, 2, 3, 4, 5, D \)).

The sink node unicasts then a Build Message on the selected path. Each intermediate node unicasts the Build Message towards the source. The Neighbors list piggybacked in the Explore Message is saved ((6, 7, 8, 13, 15, 16, 17, 18, 20)) to select the future path while continuing to receive the Explore Messages for other paths for a certain time defined by the user. Here many variants can be applied depending on the interference degree tolerated by the application (user). For example, after the shortest path is selected, the set of its one hop neighboring nodes is blocked., i.e any received Explore Message for a path which contains one of its neighbors is ignored. In this way, every node in the future path will be out of the communication range of the nodes of the shortest path. In this case, the sink does not accept a path containing one or more nodes from \( \{6, 7, 8, 13, 15, 16, 17, 18, 20\} \). Therefore, paths \( S - 15 \rightarrow 21 \rightarrow 17 \rightarrow 18 \rightarrow 19 \rightarrow 20 \rightarrow D \) or \( S - 6 \rightarrow 7 \rightarrow 8 \rightarrow 12 \rightarrow 13 \rightarrow D \) are not accepted. When the sink node receives Build Messages for paths which do not contain any of the neighboring nodes of the shortest path, it begins by calculating the intersection between Neighbors list and the list of nodes forming the shortest path. The selected path is the one with minimum number of common nodes. If the paths present the same number of common nodes in the first intersection, the sink node calculates the intersection between the Neighbors lists. The sink chooses a path with minimum number of common nodes. If two paths present the same number of common nodes in the intersection, the shortest one is selected. For example, if the sink receives requests for the three paths \( S - 9 \rightarrow 10 \rightarrow 11 \rightarrow 12 \rightarrow 14 \rightarrow D, S - 15 \rightarrow 23 \rightarrow 24 \rightarrow 25 \rightarrow 26 \rightarrow 27 \rightarrow D \) and \( S - 15 \rightarrow 21 \rightarrow 24 \rightarrow 22 \rightarrow 19 \rightarrow 27 \rightarrow D \). The path \( S - 15 \rightarrow 23 \rightarrow 24 \rightarrow 25 \rightarrow 26 \rightarrow 27 \rightarrow D \) is chosen over the other two as the number of of common neighbors between it and the shortest path is the smallest. To discover more paths and based on the network connectivity, the sink node may apply the same principle when selecting other paths (if possible). That is, when another path is selected along with the first one, the nodes of its Neighbors list are blocked and the intersection is calculated between the path to be selected and the two paths already selected (including the shortest one). This route discovery phase permits to construct path with \( I \geq R \). To select paths satisfying \( I \geq 2R \), the shortest path may be blocked along with the neighbors of its Neighbors list. Other paths are selected using the same rule (comparing the neighbors list between each path and the shortest path and between each other). When the source node receives the Build Messages for these paths, the ones above and below the shortest one are selected to transmit data.

3.2. Data Transmission Phase

To study the effect of interference on the video quality, We define four transmission strategies using three paths. This latter are typically, the shortest one and the two paths which are above and below it. These paths present different degrees of interference. The first strategy refers to the single path strategy where the shortest path is used to transmit video data (stratég P0). Two transmission paths are used in the second and third strategies. The two paths used in the second one (stratég P0P1) are more interfering (typically the shortest path and another path) than the ones used in the third one (stratég P1P2). In this latter, the paths above and below the shortest path are used. Three paths are used in the third strategy (stratég P0P1P2), the shortest one and the two others which are below and above of it. The data is sent alternatively on the three paths. Figure 2 illustrates an example of these paths.

4. Video Evaluation

We implemented the different variants of our multipath routing protocol using Castalia [8], an Omnet++ based simulator for wireless sensor networks. The CC2420 radio model provided by Castalia is used for all simulations with a data rate of 250 kbps. We also used the Additive Interference model to simulate the interference effects in the network along with a contention based CSMA MAC layer. Each scenario is simulated 10 times using different simulation seeds. To evaluate the video quality, we selected one of the
standard video sequences used by a variety of video encoding and transmission studies called Hall Monitor. The video lasts 10 seconds and consists of 300 frames in QCIF resolution \((128 \times 128)\). Two types of gray-scale frames, M (Main) and D (Difference), are generated for this video using a modified version of MPEG [9]. We fixed the number of nodes in the network topology to 300 nodes and used a communication range so that the mean degree of a node is 6. One video source is assumed to capture, encode and send video sequences to a single sink node. The source and the sink are located at the left and the right side of the simulation area respectively. We consider small values for frame rates (from 1 fps to 6 fps) as high frame rates are not achievable because of the severe bandwidth limitations of WMSNs [10].

Figure 3 compares the average PSNR (Peak signal-to-noise ratio) versus the frame rate for the different transmission strategies. We can see that using less interfering paths (Strategy P1P2) permits to achieve maximum average PSNR over the other strategies. Thus, better video quality is to be expected. When the frame rate increases, the average PSNR of all the strategies decreases as more data losses are noticed in this case (figure 4). When only one path is used for transmission, losses are mainly caused by radio non-readiness (service time). Multipath routing is then used to load balance the traffic between the different paths. However, using interfering paths for data transmission does not permit to achieve good performances. The losses are mainly caused by mutual interference from the simultaneous transmissions over the adjacent paths.

The average packet loss (depicted in figure 4) does not exactly match the graph of average PSNR. This is mainly due to the fact that data has different levels of priorities. Therefore, losing more important data affects the video quality. For example, the average packet loss of the two strategies P0P1 and P0P1P2 for the case 6 FPS is respectively 50.39 and 50.32. However, the average PSNR is respectively 19.164446 and 18.165038. This can be explained by examining the average packet loss of M frames which is 51.876 and 53.634. The M packet loss is depicted in figure 5 which corresponds more to the PSNR graph depicted in
Figure 6 shows the mean throughput improvement for different transmission strategies as function of time and confirms the previous observations (concerning PSNR and video quality). Mainly, we observe that the strategy 2 clearly shows better performances compared to the others despite the fact that shorter paths achieve better delay (graph omitted due to space constraints). The achieved throughput can be used to support real-time requirements for these frame rates.

Figure 8 and figure 9 show a sample image (Frame number 54) as received by the different strategies along with the coded image (figure 7). We can notice in figure 9 the video quality achieved when the paths used for transmission are less interfering (strategy P1P2) compared with other strategies (strategy P0P1P2).

5. Conclusion

In this paper, we presented a simple interference aware multipath routing in WSNs. Also, we investigated the use of multipath routing for video data transmission in WSNs when considering inter-path interferences. We study the effects of inter-path interference on received video quality. Our simulation results show that using less interfering paths permits to achieve better video quality in terms of PSNR. In fact, multipath routing allows for load balancing and supports real-time transmission if small frame rates are used.

References

[8] [link].